International Journal for Modern Trends in Science and Technology
Volume 12, Issue 04, pages 07-13.
ISSN: 2455-3778 online
Available online at: http://www.ijmtst.com/vol12issue04.html
DOI: https://doi.org/10.5281/zenodo.19321859

Cat Boost - Based Intelligent Threat Detection for Browser Security

Sucheta Chandra, Ankita Mandal, Sriparno Chakraborty, Sourish Chakraborty, Sohom Chakraborty, Sourav Bose, Sohan Saha

Department of Computer Application and Science Institute of Engineering & Management, University of Engineering & Management, Kolkata, India

To Cite this Article
Sucheta Chandra, Ankita Mandal, Sriparno Chakraborty, Sourish Chakraborty, Sohom Chakraborty, Sourav Bose & Sohan Saha (2026). Cat Boost - Based Intelligent Threat Detection for Browser Security. International Journal for Modern Trends in Science and Technology, 12(04), 07-13. https://doi.org/10.5281/zenodo.19321859

Article Info
Received: 28 February 2026; Revised: 18 March 2026; Accepted: 22 March 2026.

KEYWORDS

CatBoost, WebSafe, gradient boosting algorithm

ABSTRACT

In the rapidly evolving landscape of cyber threats, real-time detection and mitigation of malicious URLs are critical for securing web browsing experiences. This project introduces WebSafe, a comprehensive browser-integrated defense system that employs the CatBoost gradient boosting algorithm to classify URLs into benign, phishing, defacement, and malware categories with high accuracy. Utilizing a large-scale dataset with labeled URLs, WebSafe applies a multi-layered feature extraction process that captures URL structure, domain characteristics, and web page content attributes to enhance threat detection robust- ness. The system's design supports seamless integration within browsers, enabling proactive identification and containment of malicious links through dynamic thresholding and ensemble learning. Future developments include backend migration to the Django framework for improved scalability, deployment of browser extensions for live monitoring, model optimization to increase precision and recall, and expansion of detection capa- bilities to cover zero-day attacks. Through these enhancements, WebSafe aims to provide an adaptive, scalable, and user-friendly cybersecurity solution tailored to the challenges of modern web threats.

INTRODUCTION

Traditional browser security solutions predominantly rely on blacklisting techniques and rule-based heuristics. While these methods offer basic protection against known threats, they are inherently reactive in nature, depending on prior identification of malicious entities and static pattern-matching rules. As a result, they are ill-equipped to handle zero-day vulnerabilities, polymorphic attacks, and rapidly evolving threat vectors that increasingly characterize the modern cyber threat

landscape. With the web browser now serving as a critical access point for users—facilitating everything from social communication and online banking to enterprise cloud computing and remote work—its attack surface has expanded significantly. This ubiq- uity makes browsers an attractive target for cybercriminals who exploit even minor vulnerabilities to orchestrate phish- ing schemes, deploy malware through redirects, and perform website defacements or drive-by downloads.

Web-based threats are growing not only in frequency but also in sophistication, often leveraging obfuscation techniques, social engineering, and dynamic content manipulation to by- pass traditional defenses. For example, attackers frequently utilize visually similar domain names, dynamic JavaScript injection, shortened URLs, and redirection chains to evade blacklist detection. Moreover, legitimate-looking SSL certifi- cates and HTTPS usage can give users a false sense of security, allowing phishing sites to appear trustworthy. These challenges expose the limitations of legacy detection mechanisms and highlight the urgent need for intelligent, proactive security frameworks.

In response to these evolving threats, this project introduces WebSafe, a lightweight, browser-integrated defense architec- ture specifically designed for the real-time detection and mit- igation of malicious URLs. WebSafe employs a multi-layered detection strategy that fuses both static analysis (e.g., URL structure, domain age, presence of suspicious keywords) and behavioral features (e.g., redirection patterns, iframe usage, and login form detection) to extract comprehensive indicators of compromise. These features are fed into an ensemble machine learning pipeline, centered around the CatBoost algo- rithm—an optimized gradient boosting model well-suited for handling categorical and imbalanced data without extensive preprocessing.

The CatBoost model is trained on a heterogeneous dataset comprising benign, phishing, malware, and defaced URLs collected from reliable sources such as PhishTank, OpenPhish, and Alexa Top Sites. The model demonstrates strong gener- alization performance across unseen samples, achieving high precision, recall, and F1-scores even under conditions where malicious URLs are obfuscated or exhibit subtle mimicry of legitimate sites. Furthermore, a feature-wise thresholding mechanism is integrated into the system, allowing interpretable decision-making that enhances explainability and reduces false positives—a common challenge in real-world deployment of ML-based detection systems.

What sets WebSafe apart is its direct integration within the browser ecosystem, enabling inline scanning and real- time feedback as users interact with web content. This on- the-fly evaluation eliminates latency associated with server- side verification and reduces dependency on third-party APIs or external blacklist updates. The system is designed to be lightweight, ensuring minimal impact on browser performance, and user-centric, by offering clear risk indications without causing alert fatigue. Its modular architecture also allows for future extension to support anomaly detection, zero-day threat modeling, and integration with threat intelligence platforms.

In summary, WebSafe provides a scalable, adaptive, and high-precision approach to browser-based cybersecurity. By addressing the core limitations of traditional methods and embracing intelligent automation, WebSafe positions itself as a robust solution for mitigating the ever-evolving risks posed by malicious web content. The proposed system not only enhances real-time threat detection but also fosters a safer browsing experience for end users through proactive and explainable protection mechanisms.

LITERATURE SURVEY

Phishing attacks remain a major online threat by stealing sensitive data via impersonation of trusted entities. This paper proposes a stacking-based multilayered ensemble model com- bining boosting and deep learning for URL-based phishing detection, achieving over 99.9% accuracy and suitability for real-time use [1]. ANN ensemble-based phishing detection system using supervised and unsupervised methods is pre- sented, employing Isolation Forest and ensemble classifiers (Random Forest, LightGBM, Gradient Boosting). It supports multi-language real-time detection and maintains a phishing history database, showing high accuracy and adaptability [2]. An intelligent phishing detection model using supervised learning focuses on URL characteristics with a self-destruct detection algorithm. Tested with datasets from Phish Tank and UCI, it powers a Chrome extension to help prevent phishing in real-time [3]. Due to poor

accuracy and adaptability of past methods, this work proposes a supervised learning-based phishing detection focusing on URL features and tested in controlled environments, resulting in an effective Chrome extension for phishing prevention [4]. This paper compares multiple ML algorithms (Naive Bayes, KNN, Random Forest, etc.) on phishing URL datasets. Random Forest achieves highest accuracy (98.04%) in detecting phishing URLs after preprocessing and encoding [5]. The study trains three ML models on URL-based features to differentiate phishing sites and aims to prevent zero-day attacks. Random Forest shows strong results with precision 97%, recall 99%, and F1 score 97%, operating efficiently using URL data alone [6]. Five ML algorithms (Logistic Regression, SVM, KNN, Naive Bayes, XGBoost) are evaluated on phishing detection. XGBoost achieves best accuracy at 99.75%, highlighting its promise for cybersecurity [7]. A comparative study of ML algorithms and feature selection techniques on phishing datasets aims to reduce features while maintaining accuracy, using F1 score and execution time as evaluation metrics [8]. A literature survey on phishing detection methods highlights anti-phishing tool limitations and proposes improved URL feature defini- tions and a design framework for effective phishing detection [9]. A CatBoost regression model for IoT network intrusion detection is developed using the IDS2017 dataset, achieving 92.5% accuracy and outperforming state-of-the-art methods [10]. A hybrid approach using XGBoost optimized by an improved firefly algorithm is proposed for phishing detection with feature selection and hyperparameter tuning, outperform- ing other metaheuristics on multiple datasets [11]. A study compares Innovative SVM and Random Forest algorithms for detecting anonymous threats on social media, showing SVM's superior accuracy (90.18% vs 85.22%) with statistical significance [12]. SVM and Random Forest classifiers are applied to email phishing detection, achieving up to 99.87% accuracy in classifying emails as phished or legitimate [13]. Comparative analysis of classifiers (Na¨ıve Bayes, Logistic, Random Forest, AdaBoost, MLP) for phishing detection shows Random Forest yields the highest accuracy (97.98%) [14]. Random Forest and Decision Tree are compared for URL- based phishing detection. Random Forest outperforms with 93.27% accuracy, enhancing user security [15]. A phishing detection system using SMOTE for balancing and URL-based features trained on 11,430 URLs shows CatBoost perform- ing better than Random Forest with 97.24% accuracy [16]. Evaluation of XGBoost and CatBoost for phishing website detection shows both classifiers perform well, with XGBoost

slightly outperforming CatBoost on two datasets [17]. A study on phishing attacks via URLs, emails, and websites uses various classifiers. XGBoost achieves 94.44% accuracy on URLs, Na¨ıve Bayes 95.15% on emails, and Random Forest 96.80% on websites [18]. Using SMOTE for data balancing, this study trains ML models on 11,430 URLs with 87 features, showing XGBoost outperforming Random Forest with 97.37% accuracy for phishing detection [19].Random Forest algorithm is applied to phishing detection with data preprocessing and feature extraction, improving accuracy and adaptability for real-time defense against evolving phishing threats [20].

PROPOSED METHODOLOGY

A.Data Collection

The foundation of the system is a carefully acquired dataset consisting of 650,000 URLs, meticulously labeled into four categories: benign, defacement, malicious, and phishing. These involved splitting the dataset into training, validation, and testing subsets using stratified sampling to maintain repre- sentative distributions of all classes. Hyperparameter tuning was conducted through grid search and cross-validation to optimize model parameters such as learning rate, tree depth, and iterations, ensuring maximal predictive performance. After training, the model produces probability scores indicating the likelihood of each URL belonging to a specific threat category. To convert these probabilistic outputs into actionable classifications, detection thresholds were derived by analyzing the trade-offs between true positive and false positive rates, often using ROC curve analysis. These empirically derived thresholds replaced previously used arbitrary or dummy val- ues, allowing the backend system to perform precise and reliable classification in real time. This integration enhances WebSafe's capability to provide immediate and accurate threat detection directly within the browser environment.

URLs were gathered from multiple credible sources, including open threat intelligence platforms, cybersecurity repositories, and web crawlers targeting suspicious domains. The diver- sity and volume of the dataset are critical to capturing the broad spectrum of web-based threats encountered in real- world scenarios. This extensive dataset enables the model to learn nuanced differences between benign and various types of malicious URLs, thereby improving its generalization capability and detection accuracy. Special attention was given to balancing the dataset to avoid bias toward any particular category, ensuring that the model remains effective across all classes of threats.

B. Data Cleaning and Feature Extraction

Following data acquisition, the dataset underwent com- prehensive cleaning to enhance quality and reliability. This involved removing duplicate entries, correcting labeling incon- sistencies, and filtering out malformed or incomplete URLs that could degrade model performance. After cleaning, a detailed feature extraction process was performed to convert raw URLs into structured inputs suitable for machine learning. Key features were derived based on URL lexical analysis, such as length, frequency of special characters (e.g., -, @, %), presence of IP addresses instead of domain names, and token distribution patterns. Domain-related features were also included, such as domain age, registration status, and WHOIS information, which are known indicators of suspicious activ- ity. Additionally, content-based signals like the presence of suspicious keywords, URL encoding patterns, and path depth were incorporated to capture deeper semantic and structural properties of URLs. These comprehensive features enable the model to effectively distinguish subtle differences between legitimate and malicious URLs.

C. CatBoost Model Training and Threshold Extraction

The processed feature set served as input for training the CatBoost algorithm, a state-of-the-art gradient boosting framework known for its efficiency in handling categorical data and robustness against overfitting. The training process

Algorithm 1 Feature-Based Malicious URL Detection Using CatBoost Require: CSV file with URLs and labels (benign, phishing,
defacement, malware)
Ensure: Trained model and decision thresholds per feature
1: Load URL dataset from CSV
2: for all URL in dataset do
3: Extract features:
- URL length, presence of '@', dashes, subdomains, IP usage
- HTTPS presence, redirect count
- Login form, iframe presence (via HTML parsing)
- Suspicious word count in URL
- Domain age (via WHOIS)

4: end for
5: Encode labels as integers (e.g., benign = 0, phishing = 2, etc.)
6: Train global CatBoost classifier on all features
7: for all features fi do
8: Train single-feature CatBoost classifier on fi
9: Generate value grid across fi's domain
10: Predict class transitions over grid
11: Identify threshold boundaries where predicted class changes
12: if no transitions then
13: Assign default threshold based on dominant class
14: else
15: Define threshold functions for each class (safe, warn- ing, etc.)
16: end if
17: end for
18: return Trained model and per-feature threshold rules

IV. RESULTS AND DISCUSSION

The classification table defines the security level of a URL based on eleven key features by applying rule-based threshold-

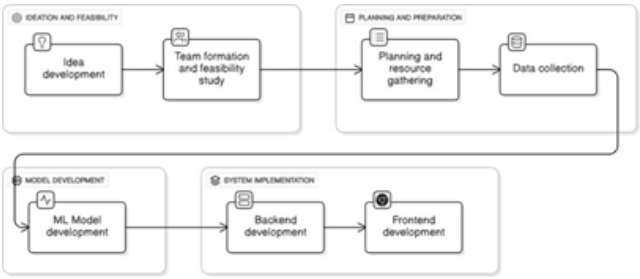

Fig. 1. PROPOSED METHODOLOGY

No. URL (Shortened) Type

1 identcode.cz/tridice-minci defacement
2 goal.com/en-us/news/1589/... benign
3 misterpoll.com/polls/300489 benign
4 masrcarcompany.homestead.com benign
5 ziraat-bireyselbankacilik.com phishing
6 googlegroups.com/stickamcoindo/ benign
7 groups.yahoo.com/OWBN-Cam phishing
8 myfundi.co.za/e/Tennis benign
9 otomoto.pl/osobowe/bmw/e46/ benign
10aba-diagnostic.com/index.html defacement
11strawberrycreek.com benign
12nhl.com/ice/schedule?team=njd benign
13twitter.com/home?status=... benign
14eslay.xyz.fozzyhost.com phishing
15wikipedia.org/wiki/USS_Edwardsbenign

TABLE 1 VIEW OF THE DATASET(SHORTENED)

No.	URL (Shortened)	Type
1	identcode.cz/tridice-minci	defacement
2	goal.com/en-us/news/1589/...	benign
3	misterpoll.com/polls/300489	benign
4	masrcarcompany.homestead.com	benign
5	ziraat-bireyselbankacilik.com	phishing
6	googlegroups.com/stickamcoindo/	benign
7	groups.yahoo.com/OWBN-Cam	phishing
8	myfundi.co.za/e/Tennis	benign
9	otomoto.pl/osobowe/bmw/e46/	benign
10	aba-diagnostic.com/index.html	defacement
11	strawberrycreek.com	benign
12	nhl.com/ice/schedule?team=njd	benign
13	twitter.com/home?status=...	benign
14	eslay.xyz.fozzyhost.com	phishing
15	wikipedia.org/wiki/USS_Edwards	benign

Each feature has specific value ranges or boolean conditions that correspond to one of four categories: Safe, Warning, Danger, and Malware. The url length feature indicates that longer URLs are generally safer, with values above 163.39 considered safe and those below 118.71 considered highly malicious. The presence of the "@" symbol (has at symbol) is unusual in legitimate domains and is treated as a phishing indicator, hence only URLs containing it are labeled safe. The use of hyphens in domains (has dash) is common in phishing attacks attempting to mimic trustworthy sites, so

Fig. 2. SNAP OF THE RESULT

TABLE II RESULT OF THE THRESHOLD-BASED CLASSIFICATION CONDITIONS FOR URL SAFETY LEVELS USING CATBOOST

Feature	Safe	Warning	Danger	Malware
has_at_symbol	True	False	False	False
has_dash	$v \leq 0$	False	$v > 0$	False
subdomain_count	$v \leq 1$	$1 < v \leq 4$	$v > 4$	False
is_https	True	False	False	False
domain_age_days	True	False	False	False
has_ip_address	True	False	False	False
redirect_count	$v > 2$	$1 < v \leq 2$	$v \leq 1$	False
has_login_form	True	False	False	False
has_iframe	$v \leq 0$	False	$v > 0$	False
suspicious_words_count	$v \leq 2$	False	$v > 2$	False

URLs without dashes are considered safer. The number of subdomains (subdomain count) helps detect overly complex URLs, with more than four subdomains suggesting danger. HTTPS usage (is https) and domain age (domain age days) are strong indicators of trustworthiness, and their presence is required for a URL to be considered safe. URLs using direct IP addresses instead of domain names (has ip address) are flagged as dangerous since this practice is often used to evade detection systems. The redirect count helps differentiate between normal redirection behavior and excessive redirec- tion, which may signal obfuscation tactics. The presence of login forms (has login form) and iframe tags (has iframe) is monitored, as they are common in phishing pages de- signed to mimic login portals or embed hidden malicious content. Finally, suspicious words count captures the number of socially engineered trigger words (like "secure," "verify," "update") present in the URL, which are often used in phishing links to incite urgency or trust. This threshold-based method provides a rule-based, explainable approach to support or override machine learning models, enhancing interpretability, minimizing false positives, and enabling

real-time URL fil- tering in practical security applications. CatBoost's classifica- tion strength comes from combining URL lexical, domain- based, and content-based features to detect patterns linked to phishing and malware. Key indicators like URL length, subdomain count, domain age, suspicious words, redirects, and login/iFrame presence enable accurate differentiation between benign mimicry and genuine malicious intent. Django enables a scalable, modular, and secure backend architecture with efficient API management, seamless model deployment, and support for real-time analytics integration.

CONCLUSION

Building upon the successful implementation of WebSafe, future work will focus on a complete migration of the backend to the Django framework to enhance modularity, scalabil- ity, and maintainability, while streamlining API development and improving integration with modern web services and databases. A critical advancement will be the development and integration of a browser extension to enable real-time URL monitoring and protection directly within users' browsers, providing immediate threat detection and alerting. Further optimization of the CatBoost model through advanced hyper- parameter tuning, feature selection, and class balancing will aim to improve precision and recall, thereby reducing false positives and negatives. To address emerging cyber threats, WebSafe will expand its detection capabilities to include zero- day attacks by incorporating anomaly detection, behavioral analysis, and external threat intelligence feeds. Additionally, deploying an interactive dashboard will facilitate real-time an- alytics, visualization of attack trends, and system performance insights to support users and administrators alike. Finally, the novel methodologies and findings will be prepared for pub- lication in peer-reviewed venues to contribute to the broader cybersecurity and machine learning research community, driv- ing continuous improvement and innovation in browser-based defense systems. WebSafe applies a hybrid decision model where CatBoost class probabilities are reinforced by ROC- derived per-feature thresholds to add interpretability and con- trol. Direct classification occurs when probabilities exceed thresholds, while borderline or conflicting cases are handled through rule reinforcement or a "Warning" state—reducing false positives, improving explainability, and enabling real- time browser filtering without arbitrary cutoffs. The system en- sures real-time performance through lightweight lexical feature extraction, a preloaded CatBoost model, fully local threshold evaluation, and no reliance on external blacklists—minimizing browsing latency. False positives are mitigated using redirect pattern analysis, multi-feature validation (e.g., HTTPS with domain age), and a threshold-based "Warning" state, pre- venting single-feature over-triggering and handling dynamic content intelligently.

FUTURE WORK

Future work will focus on migrating the backend entirely to the Django framework to enhance scalability and main- tainability while streamlining API management. A critical ad- vancement will be the development of a browser extension for real-time URL monitoring and threat detection directly within users' browsers, enabling immediate alerts and proactive de- fense. The CatBoost model will undergo further optimization to improve precision and recall, minimizing false positives and negatives. Additionally, detection capabilities will be expanded to identify zero-day threats by incorporating anomaly detection and behavioral analysis techniques. To support user engage- ment and system transparency, an interactive dashboard will be deployed for comprehensive analytics and visualization of detected threats and system performance. Finally, the project's methodology and findings will be prepared for publication to contribute to the wider cybersecurity research community and foster further innovation.

Conflict of interest statement

Authors declare that they do not have any conflict of interest.

REFERENCES

[1] A. R. Revathi, A. Arun, and K. Sebastian, "Ensemble learning frame- work for phishing url detection and leveraging llms for explainability," in 2025 11th International Conference on Communication and Signal Processing (ICCSP), 2025, pp. 566–571.

[2] S. Gowroju, S. Choudhary, K. J. Chakravarthy, N. S. Reddy, S. P. Sai, and V. Rahul, "Smart phishing detection: Integrating neural networks and ensemble learning for enhanced cybersecurity," in 2025 International Conference on Innovations in Intelligent

Systems: Advancements in Computing, Communication, and Cybersecurity (ISAC3), 2025, pp. 1–5.

[3] M. A. Syafiq Rohmat Rose, N. Basir, N. F. Nabila Rafie Heng, N. Juana Mohd Zaizi, and M. M. Saudi, "Phishing detection and prevention using chrome extension," in 2022 10th International Symposium on Digital Forensics and Security (ISDFS), 2022, pp. 1–6.

[4] G. Rangasamy, P. T, G. Kumar, H. Sivakumar, and K. G, "Phishion: Phishing detection application," in 2024 Second International Confer- ence on Emerging Trends in Information Technology and Engineering (ICETITE), 2024, pp. 1–6.

[5] A. Chandra, Gregorius, M. S. J. Immanuel, A. A. S. Gunawan, and Anderies, "Accuracy comparison of different machine learning models in phishing detection," in 2022 5th International Conference on Infor- mation and Communications Technology (ICOIACT), 2022, pp. 24–29.

[6] N. F. Abedin, R. Bawm, T. Sarwar, M. Saifuddin, M. A. Rahman, and S. Hossain, "Phishing attack detection using machine learning classification techniques," in 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS), 2020, pp. 1125–1130.

[7] A. K. Sharma, Anushree, N. Rakesh, and P. K. Verma, "An evaluation and comparison for phishing attack detection using machine learning approaches," in 2024 3rd International conference on Power Electronics and IoT Applications in Renewable Energy and its Control (PARC), 2024, pp. 464–468.

[8] S. R. Sharma, R. Parthasarathy, and P. B. Honnavalli, "A feature selection comparative study for web phishing datasets," in 2020 IEEE Inter- national Conference on Electronics, Computing and Communication Technologies (CONECCT), 2020, pp. 1–6.

[9] S. Patil and S. Dhage, "A methodical overview on phishing detection along with an organized way to construct an anti-phishing framework," in 2019 5th International Conference on Advanced Computing & Com- munication Systems (ICACCS), 2019, pp. 588–593.

[10] R. Latha and R. Bommi, "Hybrid catboost regression model based intru- sion detection system in iot-enabled networks," in 2023 9th International Conference on Electrical Energy Systems (ICEES), 2023, pp. 264–269.

[11] L. Jovanovic, D. Jovanovic, M. Antonijevic, B. Nikolic, N. Bacanin, M. Zivkovic, and I. Strumberger, "Improving phishing website detection using a hybrid two-level framework for feature selection and xgboost tuning," Journal of Web Engineering, vol. 22, no. 3, pp. 543–574, 2023.

[12] N. N. Yaswanth, R. Balamanigandan, R. Tarunprasad, and R. Palaniap- pan, "Anonymous threat creators detection in social media using support vector machine technique with improved accuracy compared with ran- dom forest technique," in 2024 IEEE 9th International Conference on Engineering Technologies and Applied Sciences (ICETAS), 2024, pp. 1–5.

[13] P. Saraswat and M. Singh Solanki, "Phishing detection in e-mails using machine learning," in 2022 2nd International Conference on Techno- logical Advancements in Computational Sciences (ICTACS), 2022, pp. 420–424.

[14] M. F. A. Razak, M. I. Jaya, F. Ernawan, A. Firdaus, and F. A. Nugroho, "Comparative analysis of machine learning classifiers for phishing detection," in 2022 6th International Conference on Informatics and Computational Sciences (ICICoS), 2022, pp. 84–88.

[15] G. Ramkumar and L. Prasanna P, "Enhancing user safety through url- based phishing detection with random forest and decision tree," in 2025 International Conference on Emerging Technologies in Engineering Applications (ICETEA), 2025, pp. 1–5.

[16] P. Singh, T. Hasija, and K. Ramkumar, "Machine learning algorithms for phishing detection: A comparative analysis of svm, random forest, and catboost models," in 2024 Second International Conference on Intelligent Cyber Physical Systems and Internet of Things (ICoICI), 2024, pp. 1421–1426.

[17] K. Sadaf, "Phishing website detection using xgboost and catboost classifiers," in 2023 International Conference on Smart Computing and Application (ICSCA), 2023, pp. 1–6.

[18] S. P. Ripa, F. Islam, and M. Arifuzzaman, "The emergence threat of phishing attack and the detection techniques using machine learning models," in 2021 International Conference on Automation, Control and Mechatronics for Industry 4.0 (ACMI), 2021, pp. 1–6.

[19] P. Singh, T. Hasija, and K. Ramkumar, "Integrated machine learning approach to phishing detection: Comparing svm, random forest, and xgboost models," in 2024 4th International Conference on Technological Advancements in Computational Sciences (ICTACS), 2024, pp. 739–744.

[20] S. Chandra, P. Das, A. Mandal, S. Chakroborty, I. Chowdhury, and K. Ghosh, "Advancing phishing detection with random forest: A ma- chine learning approach for enhanced cybersecurity," in 2025 3rd International Conference on Intelligent Systems, Advanced Computing and Communication (ISACC), 2025, pp. 74–79.