IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, October 2026
Impact Factor: 8.2

AI-Enabled Phishing Detection: A Comparative Study of Learning-Based Approaches

Akanksha B. Pawar[1], Prof. P. T. Dhotre[2], Prof. G. R. Sarode[3], Dr. H. B. Jadhav[4]
Student, Department of Computer Engineering [1]
Guide, Department of Computer Engineering [2]
HOD, Department of Computer Engineering[3]
Adusul Technical Campus, Chas, India

Abstract: *Phishing remains a major cybersecurity problem because deceptive websites, messages, and links can imitate legitimate services while changing rapidly. This paper presents an expanded comparative review of learning-based approaches for phishing detection. It examines conventional blacklists, whitelists, and rule-based mechanisms alongside machine learning, deep learning, natural-language processing, screenshot analysis, and hybrid approaches. Particular attention is given to URL structure, domain characteristics, webpage content, textual signals, visual similarity, feature extraction, dataset construction, and evaluation methodology. The discussion shows that learning-based methods can generalize beyond previously reported malicious addresses, but their practical effectiveness depends on representative and current datasets, meaningful features, balanced evaluation, and resilience to changing attack patterns. Deep and multimodal systems may provide richer representations, while lightweight models remain useful for low-latency browser and mobile deployment. The paper also discusses concept drift, adversarial manipulation, shortened URLs, external feature dependencies, explainability, privacy, and computational cost. Finally, a conceptual adaptive framework is proposed in which multiple evidence sources are combined and confirmed observations are used for controlled model updates.*

Keywords: Artificial Intelligence, Phishing Detection, Machine Learning, Deep Learning, URL Analysis, Feature Extraction, Webpage Classification, Cybersecurity

I. INTRODUCTION

The rapid growth of online banking, digital payments, e-commerce, education portals, social networks, cloud platforms, and communication services has increased the number of situations in which users interact with links and webpages. This expansion also creates opportunities for cybercriminals to imitate trusted services. Phishing is a form of social engineering in which an attacker attempts to make a malicious communication or website appear legitimate so that a user reveals information or performs an unsafe action. Phishing is difficult to detect because the visible appearance of a fraudulent resource may be convincing even when its underlying URL, domain, source code, hosting information, or page behavior is suspicious. Attackers can also register new domains, change URL paths, use redirections, shorten links, or repeatedly modify page templates. Consequently, a detection method that depends only on previously observed addresses may not recognize a newly created threat. Traditional mechanisms such as blacklists and manually written rules remain useful because they are relatively simple and can provide fast decisions for known indicators. However, their coverage depends on how quickly new threats are discovered and added. Learning-based detection attempts to complement these methods by learning statistical patterns from labelled examples. Depending on the design, a model may analyze lexical URL features, domain information, HTML structure, visible text, scripts, screenshots, or combinations of these signals. The purpose of this paper is to organize these approaches into a practical research view. The paper discusses the threat landscape, conventional detection, machine learning, deep learning, feature engineering, datasets, evaluation, practical challenges, research gaps, and anadaptive conceptual framework.

Copyright to IJARSCT
www.ijarsct.co.in
DOI: 10.48175/IJARSCT-38842

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, October 2026
Impact Factor: 8.2

The objective is not to claim a new experimentally validated algorithm, but to identify design principles that can support future phishing detection implementations.

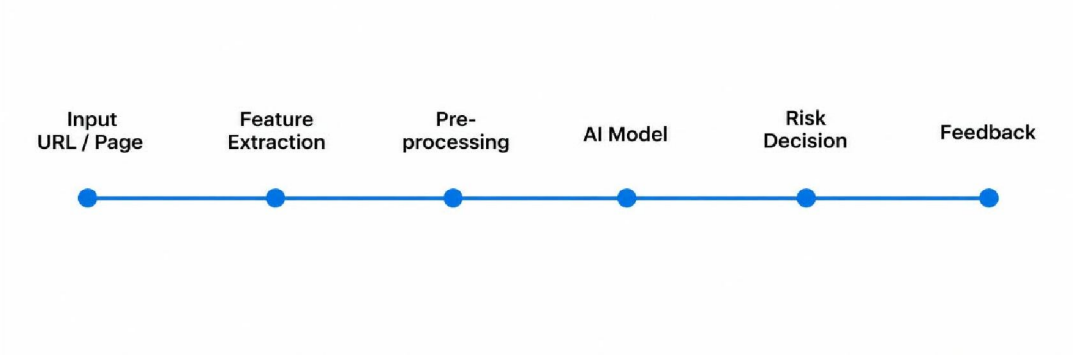


Fig. 1. Conceptual pipeline for learning-based phishing detection.

II. BACKGROUND AND THREAT LANDSCAPE

Phishing can be delivered through email, social media platforms, SMS, messaging applications, fraudulent websites, and other digital channels. The supplied literature review discusses web phishing together with related forms such as spear phishing, smishing, vishing, malicious applications, cloned pages, and deceptive URLs. Although the delivery channels differ, the central objective is similar: create enough trust or urgency that the victim treats a malicious resource as genuine. A phishing campaign can be considered a sequence of stages. The attacker first identifies a target or a group of targets, prepares a deceptive message or webpage, distributes the link through a suitable channel, waits for interaction, and attempts to collect information or influence a decision. Different stages expose different detection evidence. URL syntax can provide lexical signals, domain information can provide contextual signals, webpage source can expose structural characteristics, and screenshots can reveal visual similarity. The changing nature of the threat is important for model development. A detector trained only on historical examples may learn patterns that are specific to one period. When attackers change their infrastructure or page design, those patterns may become less reliable. Therefore, phishing detection should be considered a moving-target problem in which both data and model behavior require periodic review.

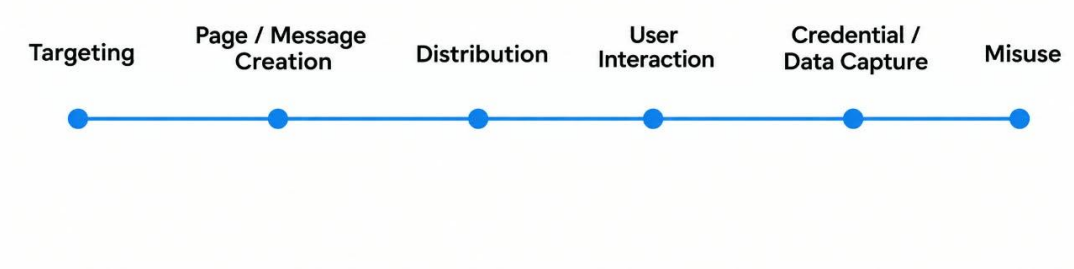


Fig. 2. Simplified lifecycle of a phishing campaign.

III. LIMITATIONS OF CONVENTIONAL DETECTION

Blacklist or blocklist detection stores known malicious addresses and compares new requests against the stored collection. Its main advantage is simplicity and speed. Its main limitation is that it relies on prior identification. A newly created phishing URL may not be present when a user encounters it, producing a gap between attack creation

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, October 2026
Impact Factor: 8.2

and list update. Whitelisting uses the opposite strategy: known legitimate resources are treated as trusted, while unknown resources may receive additional scrutiny. This can be useful in controlled environments, but a broad Internet environment contains many legitimate domains that cannot be maintained manually. Rule-based and heuristic systems improve flexibility by checking suspicious patterns such as unusual URL structures or page properties, but rules must be maintained as attacker behavior changes. Another difficulty is that a malicious URL canbe modified without changing the underlying objective. Attackers may change path names, subdomains, parameters, redirects, or shortening services. A fixed rule can therefore become outdated or produce false alarms when legitimate websites use similar structures. These limitations motivate data-driven methods. However, learning-based systems should not be treated as a complete replacement for conventional mechanisms. In practice, a layered design can use known-threat lists for fast decisions while applying learned models to uncertain or previously unseen cases.

IV. MACHINE LEARNING APPROACHES

Machine learning frames phishing detection as a classification problem. A dataset contains examples associated with classes such as legitimate and phishing, and a model learns relationships between the class labels and selected characteristics. Common approaches discussed in the literature include Support Vector Machines, Random Forests, Naive Bayes, gradient-boosting methods, and ensemble classifiers. Support Vector Machines can be effective when the selected features provide useful separation between classes. Tree-based methods can combine heterogeneous indicators and are often convenient when features have different scales or meanings. Random Forest models use multiple decision trees and can provide useful feature-importance information. Gradient-boosting and other ensemble methods build collections of weak or moderate learners to improve classification behavior. The strength of a machine-learning detector depends strongly on the representation supplied to the model. A URL can be converted into numerical features such as length, token composition, number of special characters, suspicious words, subdomain count, or redirection-related properties. Domain and hosting information can provide additional evidence. Webpage analysis can include forms, hyperlinks, HTML elements, scripts, and visible text. Preprocessing is also important. Duplicate records, missing values, inconsistent labels, and highly correlated variables can distort evaluation. Feature scaling may be required by some algorithms, while tree-based methods can operate differently. The training process should also be separated from final evaluation so that the reported performance represents unseen examples rather than memorized samples.

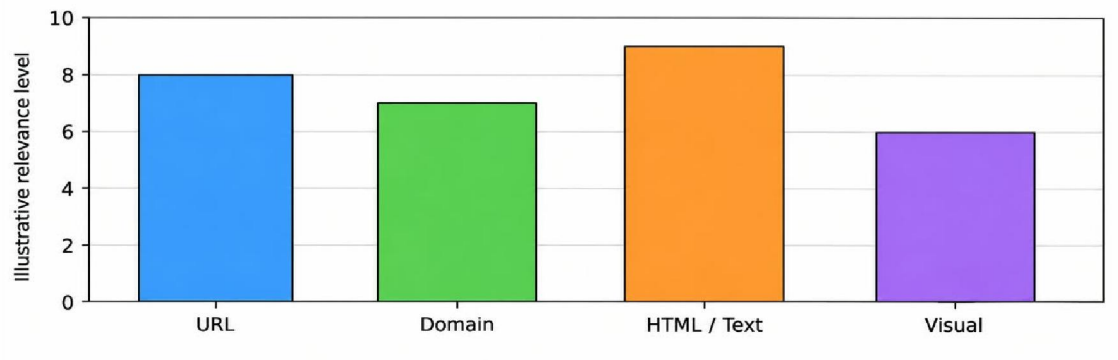


Fig. 3. Illustrative grouping of evidence used for phishing classification; values are conceptual and not experimental results.

V. DEEP LEARNING AND MULTIMODAL DETECTION

Deep learning uses neural networks to learn more complex representations from data. In phishing research, neural models may process character or token sequences from URLs, webpage text, source-code patterns, or images representing webpage screenshots. The main attraction is that useful representations can be learned from raw or lightly

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, October 2026
Impact Factor: 8.2

processed input, reducing reliance on manually defining every feature. Screenshot-based detection is particularly useful when a fraudulent webpage attempts to reproduce the visual identity of a trusted service. Layout, forms, logos, typography, spacing, and the arrangement of elements can provide visual evidence. Such evidence can complement URL and domain analysis because an attacker may hide suspicious lexical patterns while still producinga visually deceptive page. A multimodal system combines more than one evidence source. For example, URL characteristics can be evaluated together with domain signals, textual content, HTML structure, and visual information. This approach can improve coverage because weaknesses in one modality may be compensated for by another. At the same time, multimodal systems introduce additional engineering complexity, including data alignment, computational cost, missing modalities, and latency. Deep models also require careful evaluation. A high score on a large but homogeneous dataset does not automatically mean that the system will perform well on current Internet traffic. Data diversity, temporal separation, and realistic testing are therefore important when comparing deep-learning approaches.

VI. FEATURE EXTRACTION AND SELECTION

Feature extraction converts raw observations into measurable inputs. URL-based extraction may consider length, token patterns, character composition, subdomain structure, suspicious terms, and domain attributes. Content-based extraction can examine HTML tags, forms, hyperlinks, scripts, embedded resources, and visible text. Textual analysis can represent words, characters, or semantic patterns. Visual extraction can represent the layout or appearance of a webpage screenshot. Feature selection attempts to retain informative variables while reducing irrelevant or redundant information. The supplied review discusses techniques including recursive feature elimination, principal component analysis, information gain, Relief-based ranking, fuzzy rough sets, and other selection procedures. Selection can reduce computation and may improve generalization, but it must be validated carefully. A feature that is highly informative in one dataset may lose value when the threat of the environment changes. External services may provide domain, DNS, registration, hosting, or reputation information. These signals can strengthen a detector, but they also create dependencies. Network calls can increase response time, services can become unavailable, and information may become stale. A practical system should therefore identify which features can be computed locally, and which require external resources. Feature design should also consider robustness. If a detector depends heavily on one easily changed URL token or threshold, an attacker may alter that element without changing the malicious nature of the page. Combining independent evidence sources is therefore preferable to relying on one narrow indicator.

VII. DATASETS AND EVALUATION

Dataset design is one of the most important factors in phishing research. The supplied review describes multiple public and research datasets containing phishing URLs and legitimate webpages. These collections differ in size, source, age, feature availability, class balance, and update frequency. Such differences make direct comparisons between published models difficult. A model trained on old or narrowly sampled URLs may produce impressive results while failing to generalize to newer campaigns. Duplicate or near-duplicate records can also lead to overly optimistic scores if related samples appear in both training and testing sets. Data leakage should therefore be checked carefully. Evaluation should be done with multiple metrics. Accuracy describes the overall proportion of correct predictions, but it can be misleading when class distributions are imbalanced. Precision indicates how many predicted phishing cases are phishing. Recall indicates how many phishing cases have been successfully identified. F1-score balances precision and recall. False-positive rates are operationally important because repeatedly blocking legitimate websites can reduce user trust. For phishing, temporal evaluation is especially useful. A model can be trained on earlier observations and tested on later observations to approximate deployment on future threats. Reporting latency, memory requirements, update frequency, and robustness alongside classification metrics can provide a more realistic picture of practical usefulness.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, October 2026
Impact Factor: 8.2

VIII. COMPARATIVE DISCUSSION

No single detection family is universally optimal. Conventional lists are efficient when the threat has already been identified, but their coverage is limited for unseen URLs. Rule-based systems are understandable and can be updated by experts, yet they require continuing maintenance. Machine-learning models provide greater flexibility by learningrelationships among features, but their performance depends on data and feature quality. Deep-learning models can learn richer representations and may be especially useful for webpage text, source patterns, and screenshots. Their disadvantages can include greater data requirements, computational cost, training complexity, and difficulty explaining individual predictions. Hybrid systems combine different approaches and may provide broader coverage, but integration increases system complexity. The practical choice should therefore depend on the deployment environment. A browser extension or mobile application may prioritize low latency and low memory usage. A security operations environment may tolerate more computation in exchange for deeper analysis. A layered architecture can use lightweight screening first and apply more expensive analysis only to suspicious or uncertain cases.

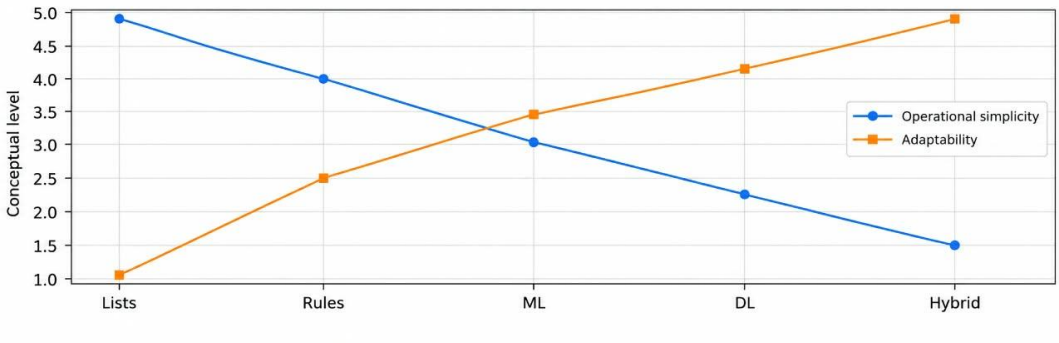


Fig. 4. Conceptual comparison of operational simplicity and adaptability; values are illustrative.

IX. PRACTICAL CHALLENGES

The first major challenge is concept drift. Phishing campaigns change their wording, domains, page templates, hosting infrastructure, redirection behavior, and distribution channels. A model that performs well on historical data can gradually lose effectiveness when the data distribution changes. Monitoring and controlled retraining are therefore important. The second challenge is adversarial behavior. Attackers may deliberately modify observable characteristics to avoid learned patterns. Robustness testing should examine whether small changes in URLs, content, or page structure cause unstable predictions. Systems should avoid depending on a single easily manipulated feature. The third challenge is the deployment cost. Real-time protection requires decisions to be produced quickly. External DNS or registration queries can increase latency, while large deep models may require more processing resources. Practical design must balance detection quality, response time, memory use, and network dependency. The fourth challenge is explainability. When a security tool blocks a website, users and administrators benefit from understandable reasons. Explanations can reference suspicious URL structure, abnormal domain properties, unusual page elements, or model confidence. Explainability can support investigation and increase confidence in automated decisions. Privacy is another consideration. Detection systems should collect only information necessary for security analysis and should avoid unnecessary retention of personal data. This is especially relevant when analyzing email text, browsing behavior, or webpage content that may contain sensitive information.

X. RESEARCH GAPS AND FUTURE DIRECTIONS

The reviewed literature identifies recurring gaps in adaptability, dataset quality, shortened URLs, external feature dependencies, and real-world deployment. These gaps indicate that phishing detection should be designed as an evolving security service rather than a static classifier. Dynamic learning is a promising direction. Confirmedphishing

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, October 2026
Impact Factor: 8.2

and legitimate samples can be monitored and incorporated into carefully controlled update cycles. Such updates should include validation so that incorrect or noisy labels do not degrade the model. Multimodal fusion is another important direction. URL, domain, text, HTML, and screenshot evidence can be evaluated together. Future benchmark datasets should include realistic temporal variation and should be refreshed regularly. Evaluation should report not only accuracy but also precision, recall, F1-score, false-positive rate, latency, resource requirements, and robustness. Privacy-aware detection should minimize unnecessary collection of user information. Lightweight models may be useful for browsers and mobile devices where computation and network access are constrained. Explainable AI can also help security administrators understand why a resource was classified as suspicious. Another research direction is uncertainty-aware decision-making. Instead of forcing every input into a strict phishing or legitimate class, a system can identify uncertain cases for deeper analysis or human review. This can reduce the risk of aggressive automated blocking when evidence is weak.

XI. PROPOSED CONCEPTUAL FRAMEWORK

Based on the reviewed approaches, a practical phishing-detection framework can be organized into five stages. First, the system receives a URL, message, webpage, or related artifact. Second, it collects evidence from multiple groups such as lexical URL characteristics, domain information, content properties, text, and optional visual signals. Third, the features are cleaned, normalized, and selected according to the requirements of the chosen model. Fourth, one or more classifiers produce a risk estimate. Fifth, the result is converted into an operational decision such as allowing, warning, blocking, or sending for additional analysis. The framework is conceptual and is not presented as a newly experimentally validated algorithm. Its purpose is to show how complementary techniques can be organized into a coherent pipeline. The design can begin with a lightweight URL classifier and later add webpage content or screenshot analysis. A confidence-aware mechanism can be included so that highly suspicious cases are handled quickly while uncertain cases receive additional checks. Confirmed outcomes can then be stored as carefully labelled feedback for future model updates. Human review may remain useful for unusual cases, especially when false positives have a high operational cost. Such a staged design also supports modular development. Each component can be evaluated independently, making it easier to identify whether errors arise from data collection, feature extraction, classification, or decision policy.

XII. CONCLUSION

Phishing detection is a moving target cybersecurity problem. Traditional lists and rules remain useful for known threats and fast screening, but they cannot by themselves provide broad protection against rapidly changing malicious links. Machine learning and deep learning offer more adaptive approaches by learning patterns from URL, domain, webpage, textual, and visual evidence. The review shows that model selection is only one part of a reliable detection system. Dataset freshness, feature quality, class balance, evaluation design, robustness, latency, explainability, privacy, and deployment constraints strongly influence practical performance. A high reported accuracy is therefore not sufficient evidence of real-world effectiveness unless the testing methodology reflects the conditions under which the system will operate. Future systems should emphasize adaptive learning, multimodal evidence, temporal evaluation, efficient deployment, and controlled model updating. Combining complementary evidence and continuously validating the detector against new attack behavior provides a stronger foundation for practical phishing defense. Technical detection should also remain complementary to user awareness and secure browsing practices.

ACKNOWLEDGMENT

I would like to express my sincere gratitude to everyone who supported and encouraged me throughout the completion of this research paper. I am thankful to our institution for providing the necessary resources and facilities. I also appreciate the valuable support and encouragement received during this work, which helped me complete the research successfully.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, October 2026
Impact Factor: 8.2

REFERENCES

[1] S. Ahmad, M. Zaman, A. S. Al-Shamayleh, R. Ahmad, S. M. Abdulhamid, I. Ergen, and A. Akhunzada, "Across the Spectrum In-Depth Review AI-Based Models for Phishing Detection," IEEE Open Journal of the Communications Society, 2025.

[2] D. Kalla, F. Samaah, S. Kuraku, and N. Smith, "Phishing detection implementation using Databricks and artificial intelligence," International Journal of Computer Applications, vol. 185, no. 11, 2023

[3] A. Safi and S. Singh, "A systematic literature review on phishing Website detection techniques," Journal of King Saud University—Computer and Information Sciences, vol. 35, no. 2, pp. 590–611, 2023.

[4] B. Naqvi, K. Perova, A. Farooq, I. Makhdoom, S. Oyedeji, and J. Porras, "Mitigation strategies against phishing attacks: A systematic literature review," Computers & Security, vol. 132, 2023.

[5] B. Gupta, K. Yadav, I. Razzak, K. Psannis, A. Castiglione, and X. Chang, "A novel approach for phishing URL detection using lexical-based machine learning in a real-time environment," Computer Communications, vol. 175, pp. 47–57, 2021.

[6] C. Singh and Meenu, "Phishing Website detection based on machine learning: A survey," Proc. 6th International Conference on Advanced Computing and Communication Systems, pp. 398–404, 2020.

[7] A. Zamir, "Phishing website detection using diverse machine learning algorithms," The Electronic Library, vol. 38, no. 1, pp. 65–80, 2020.

[8] M. Vijayalakshmi, S. Shalinie, M. Yang, and U. Meenakshi, "Web phishing detection techniques: A survey on the state-of-the-art, taxonomy and future directions," IET Networks, vol. 9, no. 5, pp. 235–246, 2020.

[9] A. Jain and B. Gupta, "A survey of phishing attack techniques, defence mechanisms and open research challenges," Enterprise Information Systems, vol. 16, no. 4, pp. 527–565, 2022.

[10] M. Almousa, T. Zhang, A. Sarrafzadeh, and M. Anwar, "Phishing Website detection: How effective are deep learning-based models and hyperparameter optimization?" Security and Privacy, vol. 5, no. 6, 2022.

[11] S. Salloum, T. Gaber, S. Vadera, and K. Shaalan, "Phishing email detection using natural language processing techniques: A literature survey," Procedia CIRP, vol. 189, pp. 19–28, 2021.

[12] M. Korkmaz, O. Sahingoz, and B. Diri, "Feature selection for the classification of Webpages to detect phishing attacks: A survey," Proc. 2nd International Congress on Human–Computer Interaction, Optimization and Robotic Applications, 2020.

[13] H. R. M. Gowda, M. V. Adithya, S. G. Prasad, and S. Vinay, "Development of anti-phishing browser based on random forest and rule of extraction framework," Cybersecurity, vol. 3, no. 1, p. 20, 2020. Copyright to IJARSCT DOI: