Al-Mustansiriyah Journal of Science

Al-Mustansiriyah Journal of Science
ISSN: 1814-635X (print) 2521-3520 (electronic)

ORIGINAL ARTICLE *OPEN ACCESS*

Performance and Explainability of Machine Learning Models in Phishing Detection Using SHAP

Dhurgham Kareem Gharkan

Department of Cybersecurity Techniques, Technical Institute Kut, Middle Technical University, Baghdad, Iraq

CORRESPONDENCE
Dhurgham Kareem Gharkan
dhurgham-kareem@mtu.edu.iq

ARTICLE INFO
Received: Jul. 12, 2025
Revised: Sep. 28, 2025
Accepted: Oct. 19, 2025
Published: Dec. 30, 2025

ABSTRACT: ***Background:*** *Phishing is a common cybercrime attack, and it is also considered a social crime that has been going on for more than two decades. Phishing aims to trick users into revealing their private information, including banking information, passwords, and account credentials. Phishing remains a real threat and usually occurs via instant messages, email, or phone calls.* ***Objective:*** *Used shape analysis on the system to uncover the most important features that contribute to phishing detection. A set of key features was identified. Many phishing detection methods have been used recently, but they do not provide a complete understanding of the impact of different features on predictions.* ***Methods:*** *Several machine learning strategies based on SHAP (Shappley Additive Explanations) were applied, which enhanced the classification model. This paper proposes a fast model based on a set of contemporary machine learning techniques.* ***Results:*** *Experiments showed that the proposed model achieved a maximum accuracy of 99.1% for K-NN and 98.5% for XG-Boost on the Phishing_Legitimate_full dataset. K-NN has demonstrated superior performance and interpretability, which is critical for security-critical applications.* ***Conclusions:*** *The results highlight the balance between predictive performance and interpretability. This provides valuable transparency into the decision-making process. This makes it a more practical choice for real-world phishing detection systems, where reliability and interpretability are critical.*

KEYWORDS: *Phishing attack; Machine learning; SHAP; Feature selection; Cyber-attack detection*

INTRODUCTION

Due to the importance of the internet in business and commerce today [1] and the abundance of services it provides that make our lives easier in general. The use of cyberspace is steadily increasing. We can get information anywhere thanks to the internet services. For example, online banking services have increased because many people have become accustomed to them [2]. There are a number of risks associated with the extensive use of internet technology, including denial of service, masquerade, replay, and phishing [3]. It has been proven that insufficient security measures have greatly increased due to weak network security [4], in addition to anonymity [5]. A single cyberattack can cause the loss of important data. Security is of the utmost importance in our society today. The problem is exacerbated by the use of smart devices [6]. Phishing websites are commonly used as entry points for online social engineering attacks, which constitute the majority of cyber breaches that target both individuals and businesses [7]. In a phishing attack, the attacker uses spam, emails, SMS, or online social networking sites to give the target audience questionable URLs after stealing pages from reliable websites, as shown in Figure 1. The hacker entices the victim to reveal extremely private or sensitive data, including bank account details, government savings numbers, login credentials, and passwords [8]. Phishing is an extremely versatile attack method [9]; by promoting fake pages or delivering massive waves of messages under the names of reputable companies, malicious customers boost their chances of success in their quest for innocent people's certifications [10]. There are three different kinds of phishing [11]: (1) Electronic phishing, where consumers are tricked into transmitting personal information by imitating a website, (2) Email-based phishing: Under this strategy, an attacker assumes that there is an issue with a customer's records and sends an email to an infinite number of customers, some of whom become victims. (3) Malware-based phishing: when a customer visits a respectable website that has been infiltrated with dubious code, their computer becomes infected with malware.

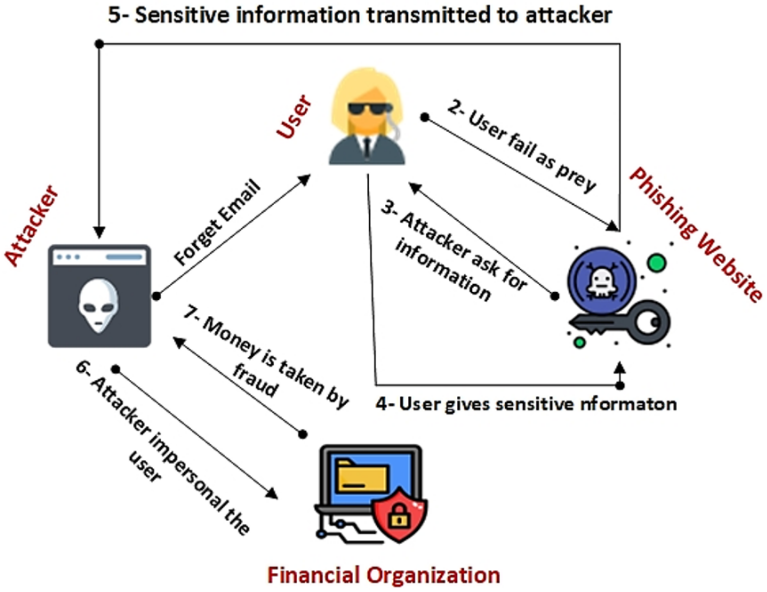
5- Sensitive information transmitted to attacker
User
2- User fail as prey
Phishing Website
Attacker
Forget Email
3- Attacker ask for information
7- Money is taken by fraud
4- User gives sensitive nformaton
6- Attacker impersonal the user
Financial Organization

Figure 1. Phishing attack scenario [12]

Numerous anti-phishing methods and solutions have been created. To get around the recently created methods, phishers have improved their tactics [13]. Many confidential and valuable details have been made public by these cyberattacks [14]. The blacklist has historically been the most used technique for phishing detection. Although phishing websites are listed on the blacklist, it takes a lot of work to maintain a current blacklist in order to identify and verify dubious websites. Additionally, maintaining a global blacklist is extremely difficult due to the rapid emergence of new phishing sites [15]. On the other hand, the whitelist consists of legitimate websites; nevertheless, similar to the blacklist, it is challenging to keep up a global whitelist due to the sheer volume of websites and their quick expansion. [16] This makes it impossible to create a database with all authentic and legitimate websites, so a customized whitelist is required. In this research, we used the whitelist approach, which provides reliable and effective protection against phishing attempts of all types by enabling automatic updates [17]. Anti-phishing software aims to prevent sensitive and private information about people and organizations from being made public, which usually leads to property loss and the release of private information [18]. Therefore, the most important contributions of this research are as follows: An anti-phishing system architecture is proposed using state-of-the-art techniques based on SHAP analysis; the K-NN and XGBoost models successfully increase the detection rate of phishing attacks; the accuracy and efficiency of the suggested model are assessed in relation to the latest technologies; and the suggested algorithm was tested on a recent data set. Proposing a new, scientific, and useful defense against phishing efforts. This study aims to overcome the aforementioned limitations and stop the threat posed by this electronic attack. This is how the rest of the document is organized: Section 2 provides a quick overview of the literature review. Section 3 displays the recommended system design, while Section 4 presents a phishing detection technique. The experimental data and an explanation of the evaluation measures are also presented in Section 5. Section 6 discusses the paper's conclusion and future directions

RELATED WORKS

In the past, a number of methods have been employed to identify phishing assaults on websites. Discuss some of these techniques in this area. According to A. Basit *et al.* in [19], in scenarios during COVID-19, phishing attacks have emerged as one of the biggest hazards that internet consumers, businesses, and service providers must deal with. Phishing attempts can be reliably detected by machine learning algorithms before they do harm to a customer. This study offers a brand-new ensemble approach to website phishing assault detection. Choose three machine learning classifiers-Decision Tree (C4.5), K-Nearest Neighbors (KNN), and Artificial Neural Network (ANN)-to employ in an ensemble setting using the Random Forest Classifier (RFC). Compared to previous studies, our ensemble technique detects online phishing attacks with greater accuracy. Experimental results show that the KNN and RFC ensemble detects phishing attacks with an accuracy rate of 97.33%. ANN and RFC have the largest area under the curve, at 0.997. After its ensemble with C4.5, the ROC area is 0.996. When paired with ANN, RFC offers 97.16% accuracy in ensembles. The KNN with RFC ensemble has a 97.33% accuracy rate. This ensemble classifier scored the highest out of all the ones evaluated in this experiment. Phishing attack detection accuracy using ensemble RFC with C4.5 was 96.36%. These experimental findings show that the ensemble model improves the detection accuracy of phishing attacks and that the suggested model is scalable, reliable, and overhead-free. COVID-19-related phishing attempts have increased recently (as seen between March 1 and March 23, 2020). Researchers' interest in this field of study is growing as a result of recent attacks on online collaboration platforms like Zoom and Microsoft Teams. Our experimental results pave the way for further research and the development of workable implementation strategies for this concept. In [20], L. Fernandez-Sanz *et al.* An automated whitelist method for phishing attack detection is presented in this research study. The visible connection and the link are carefully compared to establish the whitelist. Before a decision is made, the domain name and whitelist contents are compared, as well as the IP address. The similarities of the recognized reliable website are then calculated by further analyzing the visual connection and the real link. The information obtained from the hyperlink using this method is also accessible through the website address provided by the user. Tests were conducted using six different datasets, achieving very high accuracy even with very small dataset sizes. The method achieved an average accuracy of 96.17% in detecting phishing sites, with a true positive rate of 95.0%. Researchers in reference [21] found that integrating text, graphics, frames, and artificial intelligence algorithms contributes to developing a comprehensive approach for managing both genuine and fake websites. This study presents an integrated security solution for diagnosing online phishing, based on an Adaptive Fuzzy Neural Inference System (ANFIS), which integrates text, image, and frame data. The proposed solution achieved an accuracy of 98.3%. Furthermore, the authors of reference [22] used the K-means algorithm, a machine learning technique, to select valuable features. After selecting the features, the study uses an artificial neural network (ANN) as a classifier to distinguish between fraudulent and non-fraudulent emails. To achieve high accuracy, the proposed method modifies the parameters of the artificial neural network. The accuracy of the proposed system reached 99.4% [23]. After adding more samples, a series of tests were conducted to evaluate the accuracy of the proposed classifier; five tests were performed for each trained target site to confirm the effectiveness of the detection technique. All input URLs were correctly classified as expected, demonstrating the classifier's high accuracy in almost all categories, particularly ABSA and DHL URLs. With the PayPal URLs, the classifier's accuracy was lower (85.71%), but its precision was exceptionally high (100%), and it met the necessary accuracy rate in real-world applications with a high total detection accuracy of 94.16%. The approach demonstrates a high level of general confidence in pattern recognition with a classification accuracy rating of 95.83%. For phishing detection, this accuracy rate is considered a respectable and acceptable outcome. Because of the PayPal results, which require more research in subsequent experiments, the recall rate was lower than the precision rate (87.50%). Three studies-phishing website detection, phishing kit sample familiarity analysis, and phishing website source identification-received baseline PhiKitA results in [24]. These studies covered MD5 hashes, fingerprints, and graph representation DOM techniques. We discover signs of a minor phishing campaign and different kinds of phishing kits in the familiarity study. As demonstrated by the graph representation technique's 92.50% accuracy in the binary classification problem for phishing detection, the phishing kit data contains valuable information to detect phishing. Lastly, the MD5 hash representation received an F1 score of 39.54%. S. Uplenchwar *et al.* in [25] use machine learning (ML) methods such as KNN, Random Forest Classifier, Support Vector Classification, and Naive Bayes Classifier to detect the phished communications. PADSTM focuses on a blacklist of URLs and a range of targeted keywords in text messages to accurately identify phishing attempts. According to experimental results, the Random Forest Classifier outperforms other machine learning algorithms in terms of accuracy and F1-score for identifying phishing messages. The predictions from the estimators at the current layer are used as input for the next layer in [26], which proposes a multilayered stacked ensemble learning technique utilizing estimators across different layers. Experimental results demonstrate that the proposed model performs well when evaluated on multiple datasets, achieving accuracies ranging from 96.79% to 98.90%. The methodology was evaluated on datasets from UCI (D1), Mendeley 2020 (D3, D4), and Mendeley 2018 (D2). The proposed model achieved an accuracy of 98.9% and a detection rate of 97.76% using the D1 and D2 datasets. Finally, the method was tested on D3 and D4, which yielded accuracies of 96.79% and 98.43%, respectively. In [27], the article presents a deep learning-based phishing detection system employing five different algorithms: artificial neural networks, convolutional neural networks, recurrent neural networks, bidirectional recurrent neural networks, and attention networks. The primary objective of the system is to rapidly classify online pages using URLs. To evaluate the system's effectiveness, a large dataset of labeled URLs-comprising more than five million entries-was collected and shared. According to the experimental results, convolutional neural networks outperformed the other models, detecting phishing attacks with an accuracy of 98.74%.

MATERIALS AND METHODS

This section presents a proposed solution for building a model that enhances the detection capabilities of phishing attacks on websites. Figure 2 illustrates the framework's structure, which includes several stages to complete the process, from data collection to model interpretation. The phishing dataset consists of 50 distinct features, which form the basis of the model. The framework begins with data collection, including the collection and organization of phishing-related features. This is followed by a preprocessing step to clean and prepare the data for model training. The processed data is then fed into the detection model, which classifies websites as either phishing or legitimate. After training, the model's reliability and accuracy are evaluated using standardized metrics. Finally, SHAP analysis is used to explain the model's predictions by determining the impact of each feature on the classification results. The proposed method achieves improved detection performance while providing clear insights into how the model works.

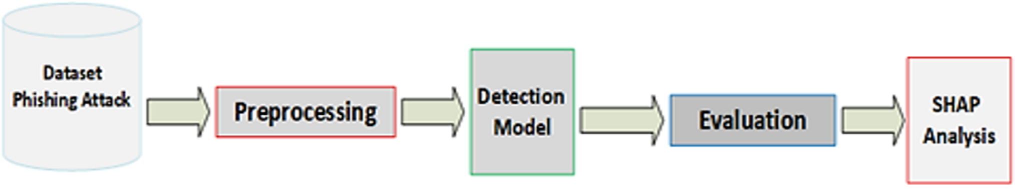

Figure 2. Proposed architecture

The three main components of the proposed model are the detection model, preprocessing, and data acquisition.

Data Description

In this study, we used the Phishing_Legitimate_full dataset, which was collected from the Kaggle website. This dataset contains data on over 10,000 phishing websites with diverse characteristics. The dataset comprises 50 features and 10,000 cases. This dataset has been used in numerous previous studies on phishing detection, making it a standard reference for this purpose. The dataset consists of approximately 11,055 cases, with a balanced distribution between phishing and legitimate websites. Each case is represented by 48-50 numerical and categorical features extracted from URL, HTML, and JavaScript properties. These features cover multiple dimensions, including: URL structure indicators (such as URL length, presence of "@" or "//", and proportion of external links), domain-related attributes (such as repeated domain mismatch, use of subdomains, and redirection patterns), page resource properties (such as proportion of external resources, metascript links, and embedded frames), and user interaction properties (such as insecure forms, email submission, and right-click disabling).

This dataset was chosen because it combines lexical and host-related features, making it suitable for phishing detection research. To ensure consistency, we preprocessed the dataset by standardizing feature values, addressing missing data, and encoding categorical variables into a numerical format suitable for machine learning. This dataset is publicly available and has been used in several recent studies (2020–2023), ensuring comparability of results and enhancing the reliability of our empirical assessment.

Data Preprocessing

To implement any machine learning model, the input data must be preprocessed to organize it and make it suitable for predictive modeling. The dataset must be cleaned to remove all noise before preprocessing. The dataset used in this study contained no missing values, providing an ideal starting point for training and evaluating the model.

1 Data Cleaning

This study focused on identifying and removing duplicate elements to ensure data quality. Automated methods were used to find duplicate rows or inputs while preserving the core dataset as a first step in the preprocessing process to guarantee model performance accuracy and reduce the time required for model training. Verification of the final dataset to ensure that all duplicates were effectively eliminated without compromising the data's quality.

2 Features Selection

Determining how many relevant and essential features are required to employ fundamental classification techniques is a crucial stage in the feature selection process [28]. Identifying every potential subset of features is conceptually equivalent to defining a subset of features. The search approach must be computationally cheap and find feature sets that are either optimal or nearly ideal, while there are numerous additional search strategies that can be employed. Since it is often impossible to satisfy both needs, trade-offs are necessary [29]. Filter, wrapper, and embedding models are the three types of supervised features, which can be defined in a number of ways. Based on the value of the extra tree's classifier, we selected the best nine features for our evaluation out of the 50 features in the dataset, as indicated in Table 1.

Table 1. Feature selection

No	Feature name	Weight	Interpretation
1	PctExtNullSelfRedirect-HyperlinksRT	0.815	Highest impact: Indicates strong association with phishing sites due to suspicious redirect patterns
2	FrequentDomainName-Mismatch	0.667	Key indicator of domain spoofing - common phishing technique
3	PctExtHyperlinks	0.434	High percentage of external links suggests malicious resource loading
4	PctNullSelfRedirect-Hyperlinks	0.413	Indicates presence of self-redirecting links (common in phishing traps)
5	ExtMetaScriptLinkRT	0.393	Suspicious external scripts loaded through meta tags
6	PctExtResourceUrlsRT	0.364	External resources (images/styles) hosted on untrusted domains
7	SubmitInfoToEmail	0.316	Form submissions to email addresses instead of secure endpoints
8	InsecureForms	0.298	HTTP form submissions without encryption (vs. HTTPS)
9	IframeOrFrame	0.255	Hidden frames used for credential harvesting

3 Feature Weighting

An essential substitute for keeping or eliminating the feature is feature weighting. It assigns less weight to the less significant qualities and greater weight to the more significant ones. Large-weighted features are crucial to the model's construction and increase accuracy. These weights are frequently established using the domain's understanding of the relative significance of features. On the other hand, it might be chosen by accident [30]. The mechanism by which the feature selection approach operates is depicted in Figure 3.

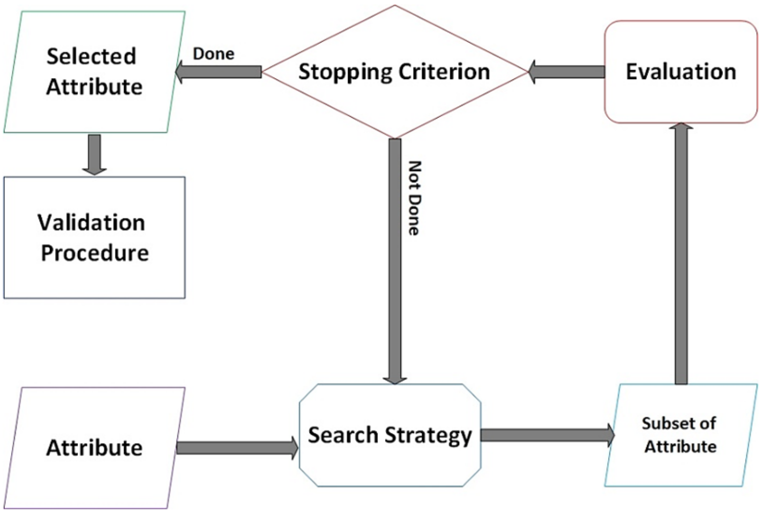

Figure 3. Proposed architecture

Detection Model

There are several algorithms that transform inputs into knowledge that can be put to good use. As the name suggests, predictive models perform tasks that involve making predictions about one value based on other values in the dataset [31]. Finding and simulating the relationship between the objective feature and the other features is the goal of the learning algorithm. Supervised learning, also referred to as classification, is the process of developing a prediction model. Among the available supervised learning techniques are K-Nearest Neighbors (K-NN), Random Forests (RF), Support Vector Machine (SVM), Logistic Regression (LR), and Extreme Gradient Boosting (XGBoost). Five models were constructed in order as illustrated in the Figure 4 to arrive at the research's conclusions. The best model was selected after they were compared. K-Nearest Neighbors is a popular method for data exploration, analysis, and predictive modeling.

1 K-Nearest Neighbors

Represents a supervised machine learning approach that solves both classification and regression tasks. The algorithm determines new data item classes or values by examining its ”k” closest neighbors from the training dataset. The method uses regression averaging and majority vote to achieve this [32].

2 Extreme Gradient Boosting, or XGBoost

The development of Extreme Gradient Boosting (XGBoost) as an open-source machine learning software enabled successful implementation of gradient-boosted decision trees. The software has gained widespread adoption because of its exceptional performance and scalability features and its ability to adapt to various applications in real-world scenarios and machine learning competitions [33].

3 Random Forest

A random forest is a technique consisting of multiple decision trees that predict a response variable based on the majority decision. In standard decision trees, each node is split to achieve the best-performing model. In random forests, nodes are randomly split. Random forests not only consider the mean and variance structure but also encompass deeper aspects of the data [34].

4 Support Vector Machine

Strong supervised learning techniques called support vector machine (SVM) are frequently employed for problems including regression, classification, and outlier detection. They function by identifying the best hyperplane in a high-dimensional space that maximally divides classes. The data points nearest to the decision boundary, known as the support vectors, determine this hyperplane [35].

5 Logistic Regression

Logistic regression is a fundamental statistical method for binary classification tasks, where the result variable is categorical with two alternative outcomes. It determines the probability that a particular class or event will occur using one or more predictor variables. Random forests, or RFs, are an ensemble learning technique used for tasks that combine classification and regression. During training, they construct several decision trees and separately generate the class mode (classification) or mean prediction (regression) for each tree. Prediction accuracy is increased and overfitting is decreased with this technique [36].

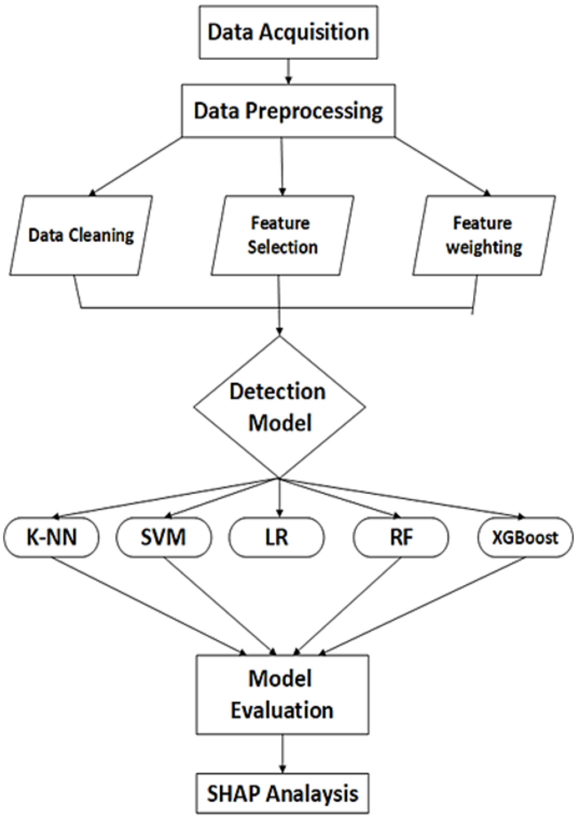

Figure 4. Workflow of detection model development and analysis

6 SHAP (Shappley Additive Explanations)

SHAP is a unified framework for explaining the output of any machine learning model. SHAP is based on cooperative game theory (Shapley values) and assigns each feature an importance value for a specific prediction. SHAP values satisfy key properties: local accuracy (the explanation matches the model output for that instance), absence (absent features have no effect), and consistency. They also provide additional, interpretable attribution for features (e.g., prediction = base value + sum of SHAP values for features). SHAP enables model-independent local explanations (e.g., force plots) and global insights (e.g., summary plots), enhancing transparency and confidence in complex models such as deep neural networks [37].

RESULTS AND DISCUSSION

Model Performance

This section presents experimental results for the implemented detection models (K-NN, SVM, RF, LR, XGBOOST). Shape analysis is then used to interpret the contributions of critical features and the model's decision logic. Feature weights were calculated using an additive tree classifier with a Gini parameter for impurity. Higher weights (scale 0–1) indicate greater importance in distinguishing phishing sites. The top 9 features were selected from the original 50 through iterative feature elimination.

Performance Metrics

The confusion matrix displays the accuracy performance table in relation to the dataset's actual classifications. Recall, accuracy, precision, and F1-score all of which were calculated using the confusion matrix to assess performance. Using the following calculations, the confusion matrix predicted performance for a specific table arrangement:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{4}$$

True Positive (TP): The real phishing was found to be the accurately anticipated phishing. False Negative (FN): The real phishing was mistakenly identified as authentic and classed as such. False Positive (FP): The true positives were marked as false values after being recognized as phishing. True Negative (TN): The legitimate classes were correctly predicted to be legitimate since the actual and expected classes were the same. Figures 5 and 6 display the confusion matrix results for the K-NN, SVM, LR, RF, and XGBoost.

The results of our evaluation measures' performance tests for the five most popular machine learning algorithms, K-NN, XGBoost, RF, SVM, and LR, are displayed in Table 2. Together with the performance measurement Equations 1-4, these results also rely on the confusion matrices (Figure 5).

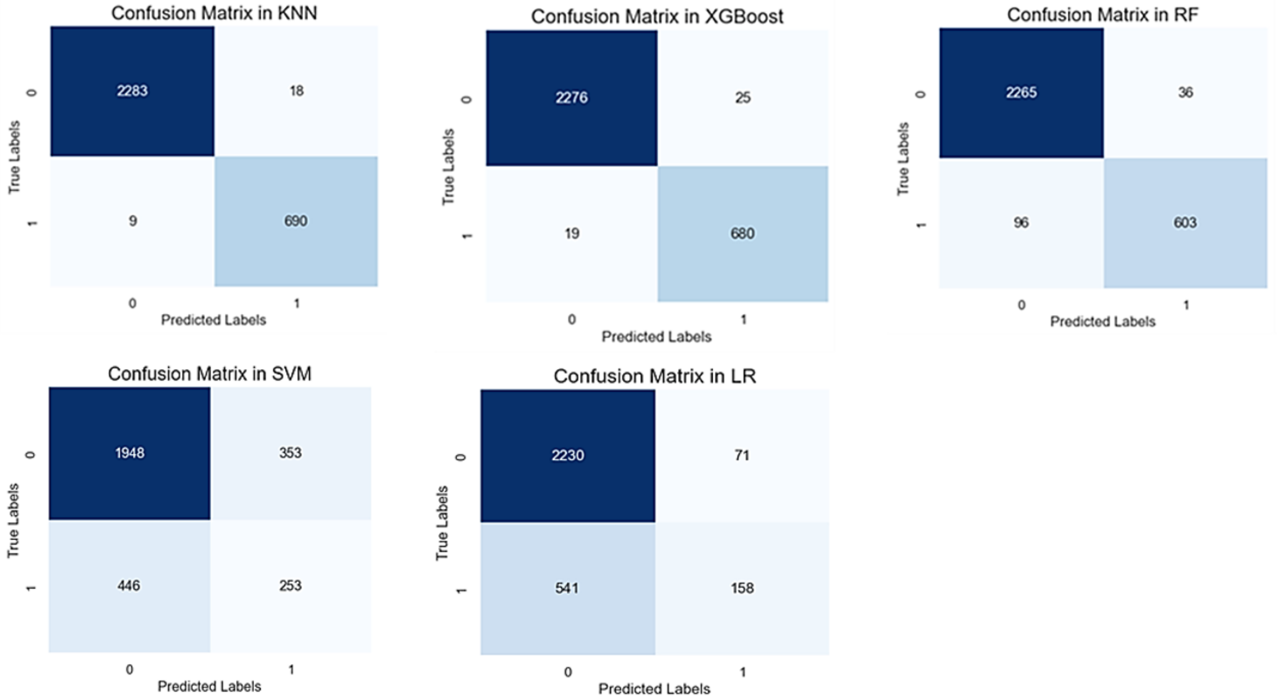

Figure 5. The confusion matrix for the five algorithms

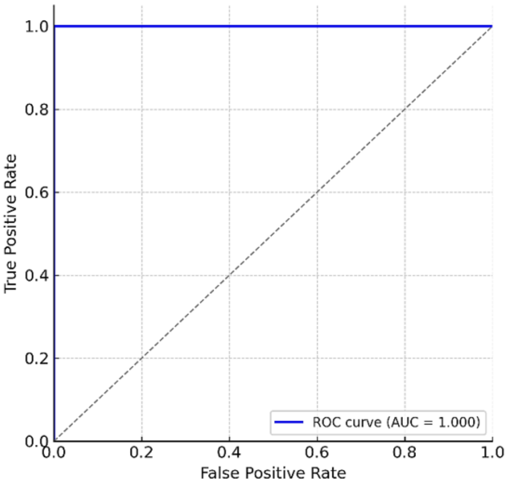

Figure 6. ROC curve for phishing detection (K-NN)

Table 2. The performance examination results

Model	Accuracy	Recall	Precision	F1 score	Time consumer
K-NN	99.1	0.99	1.00	0.99	35 S
XGBoost	98.5	0.97	0.96	0.97	59 S
RF	95.6	0.98	0.96	0.97	130 S
LR	79.6	0.97	0.80	0.88	2 M
SVM	73.3	0.85	0.81	0.83	4.63 M

Among the five classification techniques for dealing with this data evaluated, K-NN and XGBoost are ahead of others with accuracy rates of 99.1% and 98.5%, respectively, and for them the probability of success F1 score for K-NN and XGBoost is 99% and 97%, respectively, which means that it indicates that this heuristic method is the most optimal. Table 3 shows the comparisons with previous research.

Table 3. Comparison of related works and our proposed framework

Ref.	Approach / Model	Accuracy
Basit *et al.* [19]	Ensemble ML Method	≈ 97.0
L. Fernandez-Sanz *et al.* [20]	Automated Whitelist Approach	≈ 96.5
L. Barlow *et al.* [23]	Binary Visualization + ML)	≈ 98.2
L. R. Kalabarige *et al.* [26]	Stacked Ensemble Learning	≈ 98.9
Our Work (2025)	K-NN (Optimized)	≈ 99.1

SHAP Performance Analysis

Among the machine learning models applied, the K-Nearest Neighbors (K-NN) classifier achieved the highest accuracy of 99%, slightly outperforming XGBoost's 98%. While K-NN demonstrated superior performance in terms of raw classification accuracy, it is known for its poor model interpretability due to its non-parametric, instance-based nature. To better understand the internal decision logic, we applied a SHAP (Shapley Additional Interpretations) analysis to the model. The SHAP summary plot revealed that a set of features had the greatest impact on phishing predictions. These results provide a clear rationale for the model's output, which is essential in practical applications where decision transparency is critical, as shown in Figure 7. Conversely, it is therefore a better choice when achieving both performance and interpretability. This balance between accuracy and interpretability highlights the importance of selecting models based on performance metrics, as well as their suitability for use in sensitive areas such as cybersecurity.

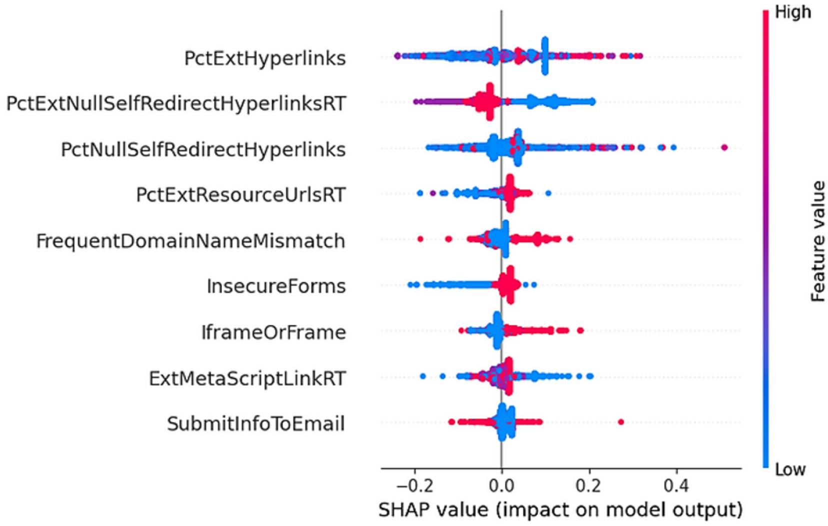

Figure 7. SHAP summary plot of phishing detection model

The SHAP summary plot highlights that hyperlink-based features (PctExtHyperlinks, PctExtNullSelfRedirectHyperlinksRT, PctNullSelfRedirectHyperlinks) and external resource indicators (PctExtResourceUrlsRT, ExtMetaScriptLinkRT) have the highest influence, reflecting attackers' reliance on manipulated links and external content. In addition, form-related features (InsecureForms, SubmitInfoToEmail) and structural elements (IframeOrFrame, FrequentDomainNameMismatch) contribute significantly, confirming known phishing patterns. Overall, SHAP not only validates the statistical importance of these features but also provides actionable insights, ensuring that the K-NN model is both highly accurate and interpretable for real-world phishing detection. Table 4 shows a comparison model between traditional explanation methods and SHAP methods in phishing detection.

Table 4. SHAP vs. traditional explainability methods in phishing detection

Feature	**Traditional Methods**	**SHAP (Our Implementation)**	**Security Impact**
Decision Interpretation	Vague feature rankings (”Important: URL length”)	Quantifiable risk attribution (“RedirectHyperlinks strongly increases phishing likelihood”)	Enables precise threat scoring for SOC tiering
SOC Integration	Manual investigation required	Automated forensic reports with feature-level attack signatures	Accelerates incident triage
Compliance Alignment	Partial GDPR/NIST coverage	Full audit trails with prediction provenance	Meets Article 22 ”right to explanation” requirements
Adversarial Analysis	Blind to evasion techniques	Detects manipulation attempts through SHAP value anomalies	Hardens models against poisoning attacks

CONCLUSION

This study highlights the urgent need for effective phishing detection systems to prevent fraudulent activities. Significant progress has been made in developing such systems thanks to feature selection based on an extra tree classifier, along with other innovative techniques that improve accuracy and computational efficiency. The proposed methods have achieved substantial performance improvements across multiple evaluation criteria. The feature selection method improved upon traditional machine learning performance metrics, including accuracy, fine-tuning, and recall, while reducing the number of features to fewer than ten. Combining feature selection with the Nearest Neighbor (K-NN) algorithm achieved a high accuracy rate of 99.1%. The algorithm also delivered superior performance in terms of execution speed and low latency, while achieving high detection rates. The research provides important solutions for efficient and accurate phishing detection while demonstrating the need for continuous innovation to confront evolving cyber threats. The research shows how predictive performance is related to ease of interpretation in detection systems. SHAP (additional interpretations) provides a balanced solution that enables both efficient detection and interpretable decisions. Future research should investigate hybrid approaches that unite the strengths of both models or implement explainability methods with non-transparent algorithms.

SUPPLEMENTARY MATERIAL

No supplementary material is provided for this study.

AUTHOR CONTRIBUTIONS

Dhurgham Kareem Gharkan: Conceptualization, investigation, methodology, software, data analysis and interpretation, visualization, writing, review, and editing.

FUNDING

This research received no external funding.

DATA AVAILABILITY STATEMENT

The dataset used in this study is publicly available on Kaggle at: https://www.kaggle.com/datasets/shashwatwork/phishing-dataset-for-machine-learning.

ACKNOWLEDGMENTS

The author wishes to acknowledge the editors and reviewers of this manuscript.

CONFLICTS OF INTEREST

The author declares no conflicts of interest.

DECLARATION OF GENERATIVE AI USE

The author declares that no generative Al or Al-assisted technologies were used in the preparation of this manuscript.

REFERENCES

[1] K. Adane, B. Beyene, and M. Abebe, “Single and hybrid-ensemble learning-based phishing website detection: Examining impacts of varied nature datasets and informative feature selection technique,” *Digital Threats: Research and Practice*, vol. 4, no. 3, pp. 1–27, 2023, doi: 10.1145/3611392.

[2] A. A. Athulya and K. Praveen, “Towards the detection of phishing attacks,” in *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184)*, IEEE, Jun. 2020, pp. 337–343, doi: 10.1109/icoei48184.2020.9142967.

[3] A. Alharbi, A. Alotaibi, L. Alghofaili, M. Alsalamah, N. Alwasil, and S. Elkhediri, “Security in social-media: Awareness of phishing attacks techniques and countermeasures,” in *2022 2nd International Conference on Computing and Information Technology (ICCIT)*, IEEE, Jan. 2022, pp. 10–16, doi: 10.1109/iccit52419.2022.9711640.

[4] S. Dangwal and A.-N. Moldovan, “Feature selection for machine learning-based phishing websites detection,” in *2021 International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA)*, IEEE, Jun. 2021, pp. 1–6, doi: 10.1109/cybersa52016.2021.9478242.

[5] G. A. Kothamasu, S. K. Angara Venkata, Y. Pemmasani, and S. Mathi, “An investigation on vulnerability analysis of phishing attacks and countermeasures,” *International Journal of Safety and Security Engineering*, vol. 13, no. 2, pp. 333–340, 2023, doi: 10.18280/ijsse.130215.

[6] A. R. Javed, M. O. Beg, M. Asim, T. Baker, and A. H. Al-Bayatti, “AlphaLogger: Detecting motion-based side-channel attack using smartphone keystrokes,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 14, no. 5, pp. 4869–4882, 2020, doi: 10.1007/s12652-020-01770-0.

[7] M. Mittal, C. Iwendi, S. Khan, and A. Rehman Javed, "Analysis of security and energy efficiency for shortest route discovery in low-energy adaptive clustering hierarchy protocol using Levenberg-Marquardt neural network and gated recurrent unit for intrusion detection system," *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 6, Art no. e3997, 2020, doi: 10.1002/ett.3997.

[8] A. Rehman Javed, Z. Jalil, S. Atif Moqurrab, S. Abbas, and X. Liu, "Ensemble Adaboost classifier for accurate and fast detection of botnet attacks in connected vehicles," *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 10, Art no. e4088, 2020, doi: 10.1002/ett.4088.

[9] A. Rasool and Z. Jalil, "A review of web browser forensic analysis tools and techniques," *Researchpedia Journal of Computing*, vol. 1, no. 1, pp. 15–21, 2020, [Online]. Available: https://www.researchgate.net/publication/358975880_A_Review_of_Web_Browser_Forensic_Analysis_Tools_and_Techniques.

[10] Z. Dong, A. Kapadia, J. Blythe, and L. J. Camp, "Beyond the lock icon: Real-time detection of phishing websites using public key certificates," in *2015 APWG Symposium on Electronic Crime Research (eCrime)*, IEEE, May 2015, pp. 1–12, doi: 10.1109/ecrime.2015.7120795.

[11] C. Iwendi, Z. Jalil, A. R. Javed, T. Reddy G., R. Kaluri, G. Srivastava, and O. Jo, "KeySplitWatermark: Zero watermarking algorithm for software protection against cyber-attacks," *IEEE Access*, vol. 8, pp. 72 650–72 660, 2020, doi: 10.1109/access.2020.2988160.

[12] D. K. Gharkan and A. A. Abdulrahman, "Construct an efficient distributed denial of service attack detection system based on data mining techniques," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 29, no. 1, Art no. 591, 2022, doi: 10.11591/ijeecs.v29.i1.pp591-597.

[13] R. Basnet, S. Mukkamala, and A. H. Sung, "Detection of phishing attacks: A machine learning approach," in *Soft Computing Applications in Industry.* Springer Berlin Heidelberg, 2008, vol. 226, pp. 373–383, doi: 10.1007/978-3-540-77465-5_19.

[14] S. Salloum, T. Gaber, S. Vadera, and K. Shaalan, "Phishing email detection using natural language processing techniques: A literature survey," *Procedia Computer Science*, vol. 189, pp. 19–28, 2021, doi: 10.1016/j.procs.2021.05.077.

[15] S. Bell and P. Komisarczuk, "An analysis of phishing blacklists: Google safe browsing, OpenPhish, and PhishTank," in *Proceedings of the Australasian Computer Science Week Multiconference*, ser. ACSW '20, ACM, Feb. 2020, pp. 1–11, doi: 10.1145/3373017.3373020.

[16] R. Goenka, M. Chawla, and N. Tiwari, "A comprehensive survey of phishing: mediums, intended targets, attack and defence techniques and a novel taxonomy," *International Journal of Information Security*, vol. 23, no. 2, pp. 819–848, 2024, doi: 10.1007/s10207-023-00768-x.

[17] A. Karim, M. Shahroz, K. Mustofa, S. B. Belhaouari, and S. R. K. Joga, "Phishing detection system through hybrid machine learning based on URL," *IEEE Access*, vol. 11, pp. 36 805–36 822, 2023, doi: 10.1109/access.2023.3252366.

[18] M. Sanchez-Paniagua, E. F. Fernandez, E. Alegre, W. Al-Nabki, and V. Gonzalez-Castro, "Phishing URL detection: A real-case scenario through login URLs," *IEEE Access*, vol. 10, pp. 42 949–42 960, 2022, doi: 10.1109/access.2022.3168681.

[19] A. Basit, M. Zafar, A. R. Javed, and Z. Jalil, "A novel ensemble machine learning method to detect phishing attack," in *2020 IEEE 23rd International Multitopic Conference (INMIC)*, IEEE, Nov. 2020, doi: 10.1109/inmic50486.2020.9318210.

[20] N. A. Azeez, S. Misra, I. A. Margaret, L. Fernandez-Sanz, and S. M. Abdulhamid, "Adopting automated whitelist approach for detecting phishing attacks," *Computers & Security*, vol. 108, Art no. 102328, Sep. 2021, doi: 10.1016/j.cose.2021.102328.

[21] M. Adebowale, K. Lwin, E. Sánchez, and M. Hossain, "Intelligent web-phishing detection and protection scheme using integrated features of images, frames and text," *Expert Systems with Applications*, vol. 115, pp. 300–313, Jan. 2019, doi: 10.1016/j.eswa.2018.07.067.

[22] M. Jasim and L. E. George, "Phishing attacks detection by using artificial neural networks," *Iraqi Journal for Computer Science and Mathematics*, vol. 4, no. 3, pp. 159–166, 2023, doi: 10.52866/ijcsm.2023.02.03.013.

[23] L. Barlow, G. Bendiab, S. Shiaeles, and N. Savage, "A novel approach to detect phishing attacks using binary visualisation and machine learning," in *2020 IEEE World Congress on Services (SERVICES)*, IEEE, Oct. 2020, pp. 177–182, doi: 10.1109/services48979.2020.00046.

[24] F. Castaño, E. F. Fernandez, R. Alaiz-Rodríguez, and E. Alegre, "PhiKitA: Phishing kit attacks dataset for phishing websites identification," *IEEE Access*, vol. 11, pp. 40 779–40 789, Apr. 2023, doi: 10.1109/access.2023.3268027.

[25] S. Uplenchwar, V. Sawant, P. Surve, S. Deshpande, and S. Kelkar, "Phishing attack detection on text messages using machine learning techniques," in *2022 IEEE Pune Section International Conference (PuneCon)*, IEEE, Dec. 2022, pp. 1–5, doi: 10.1109/punecon55413.2022.10014876.

[26] L. R. Kalabarige, R. S. Rao, A. Abraham, and L. A. Gabralla, "Multilayer stacked ensemble learning model to detect phishing websites," *IEEE Access*, vol. 10, pp. 79 543–79 552, 2022, doi: 10.1109/access.2022.3194672.

[27] O. K. Sahingoz, E. BUBEr, and E. Kugu, "DEPHIDES: Deep learning based phishing detection system," *IEEE Access*, vol. 12, pp. 8052–8070, 2024, doi: 10.1109/access.2024.3352629.

[28] T. Dokeroglu, A. Deniz, and H. E. Kiziloz, "A comprehensive survey on recent metaheuristics for feature selection," *Neurocomputing*, vol. 494, pp. 269–296, Jul. 2022, doi: 10.1016/j.neucom.2022.04.083.

[29] F. Karimi, M. B. Dowlatshahi, and A. Hashemi, "SemiACO: A semi-supervised feature selection based on ant colony optimization," *Expert Systems with Applications*, vol. 214, Art no. 119130, Mar. 2023, doi: 10.1016/j.eswa.2022.1191 30.

[30] Z. Liu, H. Qiu, S. Letchmunan, M. Deveci, and L. Abualigah, "Multi-view evidential c-means clustering with view-weight and feature-weight learning," *Fuzzy Sets and Systems*, vol. 498, Art no. 109135, Jan. 2025, doi: 10.1016/j.fss.2 024.109135.

[31] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to data mining*. Pearson Education India, 2016, [Online]. Available: https://elibrary.pearson.de/book/99.150005/9780273775324.

[32] S. Uddin, I. Haque, H. Lu, M. A. Moni, and E. Gide, "Comparative performance analysis of K-nearest neighbour (KNN) algorithm and its different variants for disease prediction," *Scientific Reports*, vol. 12, no. 1, Art no. 6256, 2022, doi: 10.1038/s41598-022-10358-x.

[33] S. K. Alam, P. Li, M. Rahman, M. Fida, and V. Elumalai, "Key factors affecting groundwater nitrate levels in the Yinchuan Region, Northwest China: Research using the eXtreme Gradient Boosting (XGBoost) model with the SHapley Additive exPlanations (SHAP) method," *Environmental Pollution*, vol. 364, Art no. 125336, Jan. 2025, doi: 10.1016/j.envpol.2024.125336.

[34] S. Wali, Y. A. Farrukh, and I. Khan, "Explainable AI and Random Forest based reliable intrusion detection system," *Computers & Security*, vol. 157, Art no. 104542, Oct. 2025, doi: 10.1016/j.cose.2025.104542.

[35] M. Daviran, A. Maghsoudi, and R. Ghezelbash, "Optimized AI-MPM: Application of PSO for tuning the hyperparameters of SVM and RF algorithms," *Computers & Geosciences*, vol. 195, Art no. 105785, Feb. 2025, doi: 10.1016/j.ca geo.2024.105785.

[36] D. Dey, M. S. Haque, M. M. Islam, U. I. Aishi, S. S. Shammy, M. S. A. Mayen, S. T. A. Noor, and M. J. Uddin, "The proper application of logistic regression model in complex survey data: A systematic review," *BMC Medical Research Methodology*, vol. 25, no. 1, Art no. 15, 2025, doi: 10.1186/s12874-024-02454-5.

[37] K. Merabet, F. Di Nunno, F. Granata, S. Kim, R. M. Adnan, S. Heddam, O. Kisi, and M. Zounemat-Kermani, "Predicting water quality variables using gradient boosting machine: global versus local explainability using SHapley Additive Explanations (SHAP)," *Earth Science Informatics*, vol. 18, no. 3, Art no. 298, 2025, doi: 10.1007/s12145-025-01 796-y.