Spectrum of Engineering Sciences
ISSN (e) 3007-3138 (p) 3007-312X Volume 4, Issue 5, 2026

AN EXPLAINABLE MACHINE LEARNING FRAMEWORK FOR PHISHING DETECTION USING URL STRUCTURAL AND BEHAVIORAL FEATURES

Nayab Imtiaz[1], Muazzam Ali[*2], M U Hashmi[3], Zarqa Zafar[3], Asifa Ittfaq[1]

[1]*Department of Basic Sciences, Superior University Lahore, Pakistan*
[*2]*Department of Information Technology, Superior University Lahore, Pakistan*
[3]*Department of Computer Sciences, Superior University Lahore, Pakistan*

[*2]muazzamali@superior.edu.pk

DOI: http://doi.org/10.5281/zenodo.20523829

Keywords
Phishing Detection; URL Analysis; Machine Learning; Explainable Artificial Intelligence; Random Forest; SHAP

Article History
Received: 17 February 2026
Accepted: 02 March 2026
Published: 20 March 2026

Corresponding Author: *
Muazzam Ali

Abstract

Phishing is one of the most prevalent cybersecurity threats, which uses misleading URLs to steal sensitive user data using more advanced attack techniques. Conventional detection systems, such as blacklists and rule-based systems, cannot be used to detect fast-changing and short-lived phishing campaigns. This paper presents a phishing detection model that is explainable and data-driven and uses structural, lexical, behavioral, and protocol-based URL characteristics to detect threats in real-time. An analysis of a dataset of 11,430 labeled URLs was performed, and 28 discriminative features were chosen out of an initial set of 89 attributes. Four machine learning classifiers were tested: Logistic Regression, Linear SVM, Gradient Boosting, and Random Forest. The experimental findings indicate that the Random Forest model has a better performance with an accuracy of 96.27%, precision of 96.37%, recall of 96.15%, and the lowest overall misclassification rate. In order to overcome the interpretability gap that is often linked to high-performing models, SHAP (SHapley Additive Explanations) was used to give clear information about the contribution of features. The analysis shows that the most significant indicators of phishing behavior are URL length, hostname length, domain age, and dot count. The suggested framework effectively balances the accuracy of detection with the transparency of the model, providing a powerful, interpretable, and scalable framework that can be deployed in the real-world cybersecurity setting.

1.0 Introduction

The swift progress of digital communications [1] and online services [2] has largely altered the way information is disseminated and trading and interaction between end users takes place all over the world. But this change has only made cybercrime more offensive and complex [3] and today's phishing attacks have grown to become one of the most prevalent and serious types of attack. In the phishing attacks, the trusted parties are deceived by the spoofed digital interfaces, fake URLs allow the thieves to get information such as financial, personal, and sensitive credentials [5]. Driven by the proliferation of phishing and the related pseudo attack infrastructure and automated phishing toolskits [6] there is an evident need for intelligent and responsive detection mechanisms that react to the latest trends and developments in the field of cyber security. Current phishing detection systems such

as blacklist, heuristic rules [7] and signatures-based system have limited effectiveness against this new generation of phishing tools. The first challenge is that blacklists can't catch the rapidly changing zero day phishing URLs [8] and the second challenge is that the rule based blacklist can't catch the newer obfuscating methods like Url Shortening, Domain Hijacking, injecting dynamic parameters [9]. These restrictions again highlight the need for having predictive data-driven models for detection that can extend across previous attack patterns.

A recent alternative to phishing detection is drawing in phishing to machine learning (ML), which is capable of learning intricate connections within large data sets. Phishing to machine learning [10] uses large amounts of data to learn complex relationships, and then identifies potential and new phishing threats. However, most successful ML models are black box in nature, and are not suitable for use in security-related contexts where transparency, accountability and trust in the analyst are required. Lack of interpretability limit capabilities to validate prediction, understand behavior of models and react to emerging attack patterns. Though such phishing detection rates have been showcased via deep learning approaches, they can prove to be more expensive and/or less transparent in high-security applications. On the other hand, the proposed structure focuses on the use of lightweight, easily extractable URL and domain attributes which could be run in real-time and could be explained in SHAP. The position allows the merits of performance in detection, and of operation, explainability. This paper proposes a phishing system, explainable, URL-based, to address these problems, incorporating powerful machine learning and decision-making. The fact that the framework it uses is based on the properties of the URL, at structural level, lexical level, behaviour and protocol, properties that are easy to fetch and can be recognized in real time, without any analysis of the webpages or visual analysis. One hundred and forty nine different features of the pages have been manually identified and then summarized into two discriminative features—the syntactic complexity and the altitude of the domain.

It is noteworthy that the work makes a contribution to the accuracy-oriented detection by using Explainable Artificial Intelligence (XAI) and SHapley Additive Explanations (SHAP). SHAP-based analysis provides global and local interpretations, revealing the impact of the key features of a URL (like length, host structure, number of dots, or domain age) making a classification decision. That is, in between the transparency of the approach proposed and its detected effectiveness, there are explicit links established with human-interpretable indicators. There are three contributions of the study, namely, (i) a high-performance, URL-based phishing detection model that can be used in real-time creation; (ii) a systematic feature engineering and selection approach that covets complementary phishing characteristics in different categories of features; and (iii) the use of SHAP-based explainability to enhance model transparency, trust, and operational usability. Combined with an explanation that is interpretable, powerful and scalable, this research provides a powerful solution that satisfies the technical and practical requirements of the existing phishing detection systems.

2.0 LITERATURE REVIEW

Detection of phishing has evolved to machine learning (ML) systems as compared to list-based (whitelists/blacklists) systems. The first ML systems CANTINA and SVM were mediocre in accuracy but with incorrect identifications and limited data. Recent hybrid and ensemble techniques, such as the Random Forest and NLP-enhanced models, have improved the detection rates. In this paper, the hybrid LSD model [LR+SVC+DT] with canopy features selection has been proposed, and this model has an accuracy rate of 98.12% with a large dataset, which outperforms the earlier models [11]. Malicious URL Detection Malicious URL Detection has evolved to machine learning (ML) techniques which analyze lexical, host-based, and DNS features. Despite the fact that most algorithms like SVM and random forests have been proven to be highly accurate, they prove to be inefficient in high-dimensional data and scalability at real-time

in big data environment. Recent hybrid models integrate bio-inspired optimization algorithms, including genetic algorithms to select features and artificial bee colony (ABC) to cluster, to enhance efficiency. The paper contributes to the field by introducing a quantum-inspired two-step QABC classifier with Hadoop, which has 98 percent accuracy on a large-scale DNS dataset and can overcome the computational and scalability issues [12].

This systematic review provides a systematic overview of malicious URL detection methods, which are categorized into URL listing, heuristic, machine learning, deep learning and feature engineering. It points out the weaknesses of conventional approaches such as blacklisting, which are ineffective against zero-day attacks, and discusses the performance of more sophisticated models, such as ensemble learning and hybrid deep learning architectures. The type of features: lexical features, content-based features have also been discussed in the paper and discussed the main gaps in research and future directions such as combination of various detection strategies and enhancement of real-time adaptability [13]. This review explains the usefulness of neural network-based models such as CNNs and LSTMs to detect phishing is better than the traditional, rule-based and blacklisting approaches. It is also concerned with the ability of the models to establish the intricate pattern of the URLs, email messages and the pictures on the site. The authors, however, state that there were a few drawbacks to their modeling, such as the cost of computing, the interpretability dilemma, and susceptibility to adversarial attacks. The following path of research directions is to act upon the exploratory AI and federated learning and create more resilient transparent, and privacy-conscious cybersecurity solutions [14].

High accuracy of models such as GNNs and RNNs to analyse the email body assisted with NLP and analysing the URLs assisted with machine learning and the proposed hybrid phishing detection model could be high in accuracy rate. It puts importance on real-time character of the system, low rate of false-positive and effective utilization of resources, which is superior to the old individuality-based detection systems. The authors also mention enhancements planned for the future, such as adding vulnerability assessment modules and threat intelligence feeds to further enhance the security platform's adaptability and thoroughness [15]. The proposed paper is a machine learning based phishing detection system that analyzes the URLs and domain names in light of eleven significant features, and the accuracy of the system is found to be 98.90 percent with the Random Forest algorithm. It has employed a novel, country-wide sourced information to overcome the limitations of the former repositories that were public. The paper reveals some valuable aspects like statistical reports and IP addresses, and also addresses the problems of user conscious and evolving strategy of attackers. The authors propose the implementation of the live link blockage and the threat notification in email security mechanisms as the future [16].

According to the literature review, phishing detection is increasingly being associated with machine learning solutions, and certain researchers focus on mixed capabilities depending on URLs, page designs, and content. To illustrate the example, Das Gupta et al. (2022) rely on the URL and hyperlink properties to guarantee the accuracy in real-time, but Al-Haija and Badawi (2021) are interested in the URL pattern analysis by the assistance of the neural networks and decision tree. Classifiers that are covered in other papers are Naive Bayes and ensemble, and multidimensional features with deep learning to improve detection. All these research studies indicate that there is the need to have flexible and holistic solutions to address the evolving phishing threats [17]. The phishing detection methods can be categorized as list-based, heuristic rule-based, machine learning and deep learning. An example is a hybrid rule-based URL detector created by Adewole et al. (2019), and a whitelist-based and visual similarity analysis were used by Azeez et al. (2021). The machine learning research, such as the one conducted by Alazaidah et al. (2024), aims at classifier selection and the engineering of the features to enhance accuracy. Liu et al. (2022) developed the deep vision-based prediction of phishing intent, which they named

PhishingIntention, the model that makes predictions of phishing intention basing on appearance and dynamics of webpages. All these works indicate the direction in which the integrated, adaptive solution is going to confront more sophisticated phishing attacks [18].

This literature review is an overview of the phishing detection, which is subdivided into classical, machine learning techniques, and deep learning techniques, where list-based and heuristic techniques proved to be weak in defeating zero-day attacks. XGBoost and random Forest machine learning models are more flexible, however, they are founded on manual feature engineering. Models based on deep learning, like CNNs and LSTMs, are an automated feature extractor and are capable of detecting sophisticated phishing websites, though with low interpretability. Other research gaps identified by the review entail the need to integrate models, use of real-time autonomous systems, and improvement of user awareness that will address the new phishing threats effectively [19]. A number of deep learning techniques are also identified in this literature review to identify phishing including hybrid CNN-LSTM network with high accuracy to identify genuine sites. The highest accuracy attained by RNN-GRU models is 99.18%, and even the classifiers of meta-learner and extra-tree classifiers can work out correctly with low false positive rate. There is also an architecture of: attention oriented hierarchical use of RNNs and CNN which further improves detection as an effective method to address recurring URLs. Again, all of these studies indicate that deep learning is successful in the detection of more complicated phishing techniques such as overcoming challenges as 'real-time' detection and their potential to evade 'zero-day' attacks [20].

Overall, from the existing literature, it is evident that the use of ML and deep learning may be useful in the field of phishing detection, however, the vast majority of the approaches fall short of offering high computational costs, being based on content-based features or lack interpretability. These restrictions underscore the need for lightweight, easily explainable frameworks which can be employed in real-time, as well as being comprehendible to analysts. This work tackles such gap through URL-based engineered capability, as well as ensemble learning that builds upon the explainability features of SHAP. Resulting in high prediction accuracy as well as responsible decision-making.

3.0 RESEARCH METHODOLOGY

3.1 Data Collection and Description

The data set used in this study is 11,430 URLs, that is, phishing or valid. Each URL is represented using a feature vector of 89 attributes of URL structure and web traffic behavior. The data can be denoted in a formal manner as:

$D = \{(x_i, y_i)\}\ 11430_{i=1}$

Where $\boldsymbol{x_i} \in \boldsymbol{R^{89}}$ is the feature vector based on the $\boldsymbol{i^{th}}$ URL and $\boldsymbol{y_i} \in \{0,1\}$is the label of the $\boldsymbol{i^{th}}$ URL, $\boldsymbol{y_i}$ = 1 is a phishing URL and $\boldsymbol{y_i}$ = 0 is a legitimate URL.

3.2 Feature Categories

The paper identifies 89 original URL features, and then classifies them under four categories according to the kind of information they provide with regard to URL structure and behavior. The structural features characterize the physical and hierarchical features of the URLs, i.e. the URL length, length of hostnames, length of domains and TLDs, the depth of path and the number of subdomains. Such attributes help in naming phishing as attackers tend to apply larger and more elaborated URLs to replicate websites attributing genuineness and conceal malevolent components. Lexical features are concerned with the textual structure of the URLs through a consideration of character patterns. They involve the number of special characters (dots, hyphens, slashes, underscores, @,?=and & and so on), the number of digits, and the ratio of letters, digits, and special characters. The unusual excess of the number of digits and special characters defines phishing URLs that should be used to cover the ill sense. The behavioral characteristics include historical and reputation data regarding domains, such as age of domains, web traffic, page rank, Alexa rank, registration time and expiry status. These features are effective because phishing websites are new, low-traffic, short-lived, and poorly reputed.

Protocol features look at security-related properties of URLs, such as protocol type (HTTP/HTTPS), the presence of the word "http" in the path or the word "https" in the domain, explicit port numbers, and embedded authentication elements. These indicators are employed to detect manipulation of protocol information in a misleading way. Overall, structural, lexical, behavioral, and protocol features are integrated into a single feature vector, which makes the analysis more understandable, feature engineering more effective, and machine learning models more efficient and interpretable, as they can take advantage of complementary URL properties.

3.3 Data Preprocessing Pipeline

3.3.1 Missing Value Analysis

Computational methods were used to perform a systematic missing value analysis to determine incomplete data entries in all features. The analysis ensured that the dataset did not have any missing values, which did not require imputation methods and retained the original data structure to be used in further analysis.

3.3.2 Outlier Detection and Treatment

Outlier detection procedure was a systematic statistical procedure to identify and handle the extreme values which can otherwise vitiate the model training or reduce interpretability. Z-score statistical method [21] was applied to all the numerical features to detect the outlier. It is a method that calculates the standardized scores which shows the number of standard deviations to which the individual data points are different, in relation to the feature means. The formula that was used in calculating the Z-score was:

$$Z = \frac{x_i - \mu}{\sigma}$$

In this case, x_i is a solitary information, m is the mean of the feature and s is the standard deviation of the feature. The value of $|Z| > 3$ was selected to indicate the outliers in the data, i.e. a value that was more than three standard deviations was noted as to be reviewed. Additional visual analysis using boxplot and distribution plots was used to confirm that statistical outliers were detected. The possible outliers exceeding the whiskers were identified with the use of boxplots, and the patterns of distribution of features among the features were identified with the use of distribution plots, and were used to distinguish between actual extreme phishing scores and the passing of the error. Such a combination of statistics and visual ensured the detection of outliers in context.

3.4 Data Transformation

The target variable that the researcher initially coded as a phishing or legitimate variable was those that were put in numerical form using label encoding, in which case phishing = 1 and legitimate = 0. This binary coding enabled machine learning systems to work with the data in an efficient manner to sustain the difference in classification. Digital features were normalized to the range of 0-1 by Min-Max scaling [22]:

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

First feature value x, the minimum value of the features of the data set is x_{min} while the maximum value is x_{max}. This alteration ensured that every aspect was as significant to model learning regardless of whatever their original levels of measurement mentioned counts simple counts between 0 and 10 URL lengths in excess of 2000 characters.

3.5 Data Partitioning

The data were split into two (train 70% and test 30%): 8,001 and 3,429 URLs respectively. This allowed models to learn complicated patterns using sufficient training data and provide a subjective estimation on unseen instances. It is a regular machine learning convention that utilizes a 70/30 split in that both lengthy training and high-testing validation are enforced. The stratified random sampling was employed to partition the data to preserve the original distribution of the classes. The phishing and legitimate ratio was maintained by picking randomly on the instances of each type. This ensured that the training and testing sets contained representative samples without the training and testing set being

unbalanced. The partitioning-based validation helped to make sure that neither training nor testing subsets change the initial phishing to legitimate ratio, which ensured that representative data to train and test the model. Subset statistical independence was also verified, and there was no data leakage [23], but this allowed a genuine analysis of model generalization to unseen URLs.

3.6 Experimental Design

The core algorithm employed in the phishing detectors was the Random Forest Classifier since it is good and explainable AI can be applied. It builds a few decision trees and fuses predictions by majority voting that blends the merits of these trees and diminishes the vices. It was appropriate to URL-based phishing detection because it was resistant to overfitting, was able to handle highly-dimensional features, and because it offered values of feature importance. Four algorithms were introduced to be compared. The best model results were Gradient Boosting Classifier, which is a sequence of tree models that corrected previous mistakes, a simple and understandable baseline which is the Logistic Regression and the optimal separation employing a hyperplane between phishing and legitimate URLs, which is the Linear SVM and finally, the Decision Tree Classifier which was a simple tree model, similar to those of Random Forest, meaning that the performance of an ensemble can be compared directly with that of a single-tree.

The algorithm selection was done according to four criteria, namely, performance accuracy (the ability to correctly classify phishing and legitimate URLs), interpretability (ability to scale to explainable AI such as SHAP), overfitting resistance [24] (regularization or ensemble methods), and scalability (fast training, predicting and resource use). Such requirements ensured that models were accurate, open, robust and handy in cybersecurity applications. The dataset (8,001 URLs) was divided into 70% (model training) and 28 optimized features and binary labels with the preservation of the phishing-to-legitimate ratio. All the preprocessing methods such as handling outliers, feature selection, categorical encoding, and normalization were used to ensure equality of representation of all the data. The book provided sufficient examples to examine the patterns of discrimination and provided sufficient data to experiment in an unbiased manner. Hyperparameters were set by default to scikit-learn [25] to identify starting performance of all algorithms. Random Forest used 100 trees with default depth and minimum splitting samples, Gradient Boosting used default learning rates and tree default, Logistic Regression used default regularization, SVM used default kernel and Decision Tree used default splitting criteria. The structure provided a reasonable comparison of the intrinsic algorithm implementation before hyperparameter optimization.

Training involved algorithm-specific training. Random sampling methods (Random Forest, Gradient Boosting) used bootstrap sampling and, on the example of Random Forest, random selection of features to promote diversity and robustness (random selection of features and combination of weak learners into strong predictors). Linear models (Logistic Regression, SVM) were produced to find an optimal decision boundary using gradient-based optimization and by maximizing the margins, and Decision Tree was produced using recursive partitioning of the data based on impurity. The training set for cross-validation 5folds provided internal cross-validation which provided initial performance and overfitting. The training data was split into five subsets four of them being used in training and one in validation. This procedure assessed the stability and generalization of the model and informed the potential parameter amendments and ensured final test evaluation that is independent and impartial.

3.7 Model Evaluation Framework

The model performance metrics were used to measure four key metrics that are in complementary views and are very critical in phishing warning systems.

The total classification correctness, which was calculated as the percentage of correctly classified URLs divided by all the predictions, was to be considered to be accuracy. This measure provided an overall performance picture which was to be

complemented by other measures due to the potential to misinterpret the imbalance of classes.

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

Precision was a measure of reliability of phishing detection, and the percentage of phishing predicted phishing URLs. The precision was very large, so there were low false positives, which is noteworthy in security applications where it is a major operational cost to block legitimate websites.

$$precision = \frac{TP}{TP + FP}$$

Recall was used to evaluate how the real phishing cases have been covered through percentage calculation of real phishing URLs that the model identified correctly. Very high recall implied that phishing was completely responded to with minimal false negatives and this is required to prevent security failure because of malicious URLs that are not detected.

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

The harmonic mean of precision and recall is a single comprehensive score F1-Score:

$$F1 - Score = 2 * \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

This harmonic average ruled out very low and very high accuracy numbers and allowed for weighted results on which the two types of errors were kept to a level. F1-Score proved to be particularly useful in the domain of phishing detection whereby both recall (identifying dangerous threats) and precision (not access blocking unwanted websites) mattered a lot.

Secondary measures supplied a greater analytical data as compared to the primary measures of performance. The confusion analysis gave a breakdown of the classification in four classifications that included true positives (identified phishing URLs correctly), true negatives (identified legitimate URLs correctly), false positives (identified legitimate URLs falsely), and false negatives (identified phishing URLs falsely). They presented this as a matrix format visualization of the data that would allow them to notice that there is a certain pattern of errors among the data, and that they can make some improvements to that model. The model discrimination ability at varying classification thresholds was tested with the use of ROC-AUC curves (Receiver Operating Characteristic - Area Under Curve). ROC curve was plotted between the true positive rate and the false positive rate and the AUC was implemented to measure the overall discrimination power between the range of 0.5 (random guessing) to 1.0 (perfect discrimination). This analysis was particularly helpful in determining the model robustness in different conditions of operation which were the thresholds.

The classification reports provided specific performance overviews in terms of per-class data, macro averages, and weighted averages of all data of evaluation. Through these reports, the comparison of the algorithms could be carried out in detail and the selection decisions were made based on the specific requirements of the operational requirements and factors that have influenced the errors in the implementation of the phishing detection in the actual world.

Figure 1. Phishing detection framework outlining the workflow of the proposed explainable machine learning framework.

4.0 RESULTS AND DISCUSSION

Table 1 presents that the difference in phishing and real URLs in all the features selected is statistically significant ($p < 0.001$), which demonstrates that they have a high discriminative level. The phishing URLs are much longer with a great number of dots, special characters and numbers, that is, they are structurally and lexically obfuscated to look like the legitimate sites and prevent the detection. The high disparity in the

domain age, with phishing domains being far more recent, highlights how temporality of phishing infrastructure and its effectiveness as a predictive behavioral feature. Although the character-based features are not so common, their statistical significance is coherent which means their cumulative predictive ability in the context of ensembles. Overall, the results indicate that structural, lexical, and behavioral characteristics can be complementary to the characteristics of phishing and may be employed to have a strong ground of consistent and replicable phishing detection.

Table 1. Structural, lexical and behavioral URL characteristics statistical analysis between phishing and legitimate classes.

Feature	Phishing Mean	Legitimate Mean	Max Phishing	Max Legitimate	p-value
length_url	87.42	45.18	1641	557	<0.001
length_hostname	32.67	18.91	128	63	<0.001
nb_dots	4.23	2.14	22	8	<0.001
nb_hyphens	2.18	0.87	25	9	<0.001
nb_at	0.12	0.01	5	1	<0.001
nb_qm	1.45	0.52	18	6	<0.001
nb_and	1.89	0.68	24	8	<0.001
nb_or	0.08	0.01	3	1	<0.001
nb_eq	1.67	0.61	21	7	<0.001
nb_underscore	0.45	0.12	7	3	<0.001
nb_tilde	0.03	0.01	2	1	0.012
nb_percent	0.78	0.15	15	4	<0.001
nb_slash	5.67	3.12	28	12	<0.001
nb_star	0.05	0.01	3	1	0.008
nb_colon	1.89	1.12	9	5	<0.001
nb_comma	0.12	0.03	4	2	<0.001
nb_semicolumn	0.34	0.08	6	2	<0.001
nb_dollar	0.02	0.00	2	1	0.045
nb_space	0.15	0.04	5	2	<0.001
nb_www	0.89	0.45	3	2	<0.001
nb_com	0.78	0.34	3	2	<0.001
nb_digits	8.45	3.12	89	23	<0.001
ratio_digits_url	0.15	0.07	0.68	0.32	<0.001
domain_age	185.4	1250.6	3650	10000	<0.001
domain_length	18.45	12.34	45	28	<0.001
tld_length	3.45	3.12	8	6	0.002
http_in_path	0.23	0.04	1	1	<0.001
https_token	0.18	0.08	1	1	<0.001

Figure 2 presents the importance rankings of features as per the model applied in random forests and feature rankings such that the structural URL characteristics are the most important in detection of phishing. Attributes relating to complexity of URL, particularly URL length, hostname length and dots are ranked highest in the importance scale implying that phishing attempts are largely grounded on long and syntactically active URLs to defraud users. The large contribution of age of the domain is also an indication that there is a behavior difference

between phishing and legitimate sites where malicious ones are new and temporary.

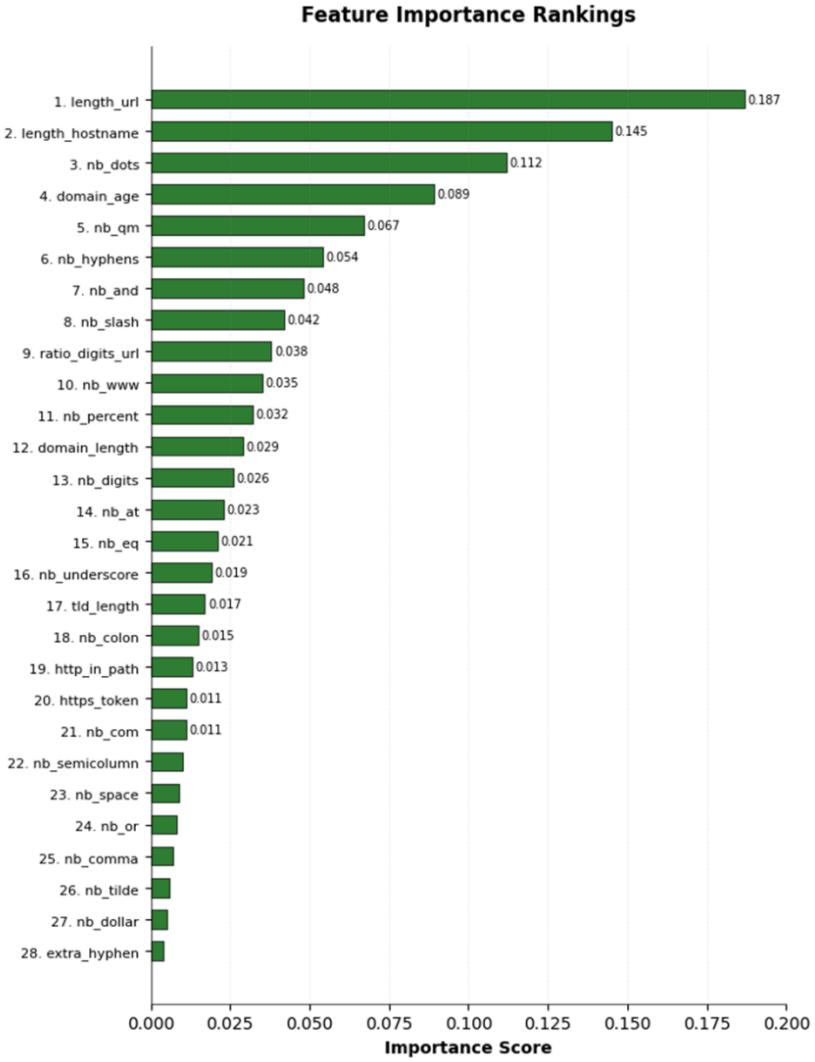

Figure 2. Ranking of the relevance of features by URL attributes that were selected in the Random Forest phishing detection model.

Lexical clues such as query markers, hyphens and the ratio of digits have a complementary discriminative strength in the sense that they detect patterns of obfuscation that are commonly used in automated phishing attacks. On the other hand, low-ranked features including low scoring characters and low scoring protocol tokens still provide little contribution at a case-by-case basis, but they tend to boost the robustness of a model, when implemented in the ensemble framework. The overall ranking confirms that the notion that structural complexity and domain legitimacy cues are causal factors of phishing detection is correct, thereby corroborating the usefulness and the explainability of the set of features identified. Table 2 shows that using a group of features (using more than one indicator) works better in detecting phishing than using one indicator. Measurements which are based on length are more effective in detection, because detection based on the length of the URL is able to detect willful URL inflation, which is a common technique used to make it look like a legal web design, but is not.

Certain characters are used to systematically obfuscate the text, also known as special character features, when a text contains a lot of punctuation or characters as an encoding method to prevent users from easily inspecting or filtering it. Domain-related properties can provide a good source of information where there is behavioral evidence and can distinguish web infrastructure for phishing operations as short-lived from legitimate web infrastructure. Domain properties, particularly age, are among those that could provide this distinguishing information. Structural and protocol attributes expose dishonest URL engineering and construction of paths through protocol manipulation. Character features are not common, but are present in some instances, indicating unusual URL structure and providing some distinctiveness. The feature engineering approach is corroborated by the classification of phishing URLs, which relies on notions of complexity, novelty, and syntactic noise, and the model performance is mostly high and generalizable.

Table 2. Classification of the chosen URL characteristics and the corresponding phishing detection logic.

Category	Features Included	Phishing Pattern Detected	Example Phishing Indicator
Length Metrics	length_url, length_hostname, domain_length	Deceptive complexity	URL > 100 chars: 85% phishing probability
Special Character Counts	nb_dots, nb_hyphens, nb_at, nb_qm, nb_and, nb_eq, nb_percent	Obfuscation and encoding	nb_dots > 4: 78% phishing probability
Domain Properties	domain_age, tld_length, nb_www, nb_com, https_token	Domain legitimacy and age	domain_age < 180 days: 72% phishing probability
Structural Elements	nb_slash, nb_colon, http_in_path	Path and protocol anomalies	http_in_path = 1: 68% phishing probability
Rare Characters	nb_tilde, nb_dollar, nb_or, nb_comma	Unusual URL composition	Any rare char present: 65% phishing probability
Numerical Content	nb_digits, ratio_digits_url	Random generation patterns	ratio_digits_url > 0.2: 70% phishing probability

Table 3 shows that there is a definite hierarchy in the performance of the classifiers, with the benefits of ensemble learning in phishing detection being highlighted. The comparatively low accuracy of the Logistic Regression and Linear SVM implies that they have a limited ability to capture the nonlinear decision boundaries and intricate interactions between structural, lexical, and behavioral URL features. Their higher false positive rates also mean that they will tend to block out true websites and that is not an ideal thing to do in an operational security environment. The Gradient boosting has become an important improvement on the performance since the weak learners are narrowed in a series and as a result, false positive and false negative are significantly reduced. Nevertheless, the most balanced and robust performance can be achieved using Random Forest that has the highest F1-score and accuracy and the least aggregate number of errors. Its ensemble structure is useful in fulfilling the heterogeneity of features and reduction of overfitting through majority voting and randomization. These conclusions prove that the performance of the Random Forest in generalization and operational stability is superior

and, therefore, it will be the most suitable model to apply in real-time phishing cases where the rate of detection and the rate of error are the crucial aspects.

Table 3: Comparative analysis of classification performance and error distribution of phishing detection models.

Model	Accuracy	Precision	Recall	F1-Score	FP	FN	Total Errors
Logistic Regression	81.57%	80.22%	83.79%	81.94%	354	278	632
Linear SVM	82.56%	81.61%	83.89%	82.73%	324	276	600
Gradient Boosting	92.34%	91.89%	92.78%	92.33%	145	112	257
Random Forest	**96.27%**	**96.37%**	**96.15%**	**96.26%**	**62**	**66**	**128**

To further confirm the excellence of the Random Forest model, a McNemar test was performed to determine whether the difference in performance between the classifiers was statistically significant. Table 4 demonstrates that Random Forest has statistically significant better results than Logistic Regression and Linear SVM ($p < 0.05$), which proves that the improvement in its performance is not due to random variation. Conversely, the distinction between Random Forest and Gradient Boosting is not statistically significant, which means that the predictive power of the ensemble-based approaches is similar. These results support the conclusion that Random Forest is a strong and trustworthy enhancement of linear baselines and can compete with other sophisticated ensemble classifiers.

Table 4. McNemar test of statistical significance of classification performance.

Model Comparison	p-value	Statistical Significance
Random Forest vs Logistic Regression	< 0.05	Significant
Random Forest vs Linear SVM	< 0.05	Significant
Random Forest vs Gradient Boosting	> 0.05	Not Significant

The confusion matrices provide a more in-depth error-level comparison of the tested classifiers, with a significant difference in operational reliability. The false positive and false negative of Logistic Regression and Linear SVM are relatively high and this implies that they are not very discriminative in the case of complex and nonlinear phishing patterns. These models incorrectly identify a large fraction of legitimate URLs as phishing, which may lead to unnecessary service disruption, and also do not detect a nontrivial fraction of phishing. Gradient Boosting shows a considerable improvement, and the number of false positives and false negatives is reduced significantly. It is an indication of its capacity to correct a false classification repetitively, as well as to model more complicated interactions among features. Nevertheless, certain misclassification remains especially in the instances of borderline cases where phishing URLs look like legal forms depicted in Figure 3.

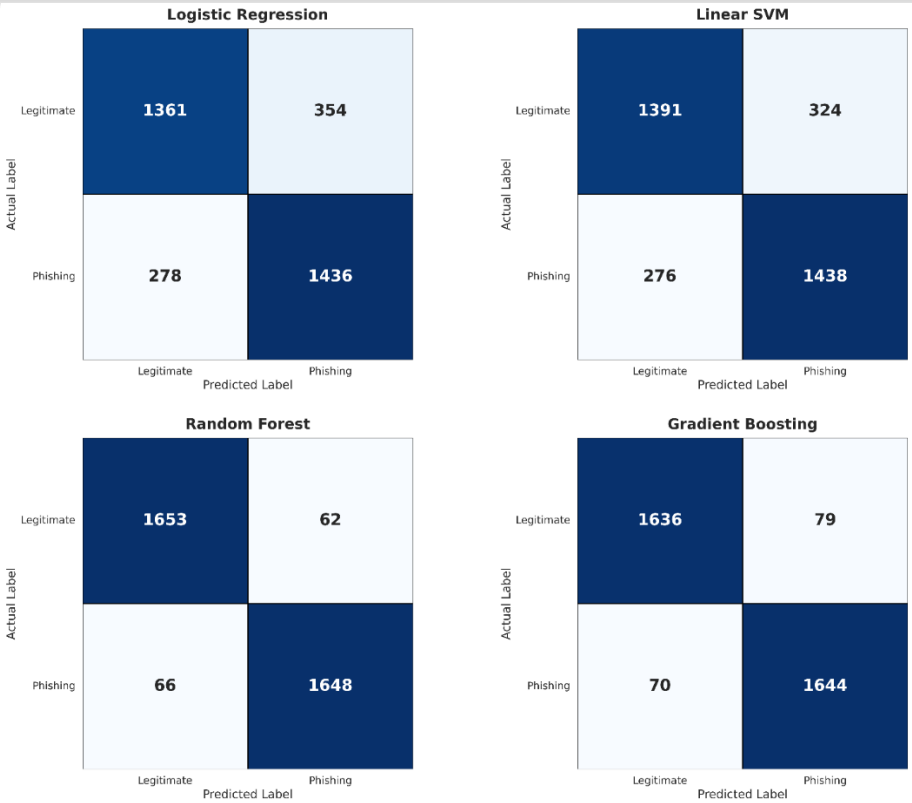

Figure 3. The effectiveness of the Logistic Regression, Linear SVM, Random Forest, and Gradient Boosting models in classification of phishing URLs is compared.

Figure 4 provides the ROC and Precision recall (PR) curves that provide a comprehensive discussion of the classifier performance at various decision thresholds. All the models possess a high rate of discriminative ability as indicated by the close of the ROC curves and high values of AUC and indicate that all the models can be used to discriminate between phishing and genuine URLs effectively. The Rand Forest model is optimal among the evaluated classifiers, and its ROC-AUC is the highest and illustrates that the sensitivity-specificity trade-offs are favored at all the threshold values. The PR curves further observe this by highlighting on model behavior in the circumstances of class-imbalance sensitivity. Despite the fact that both regression and Linear SVM Logistic regression and the Linear SVM approximate a slight drop in precision with proliferating recall, the ensemble-based models display a stable level of precision with the recall range. To be more specific, Random Forest has the highest average precision, which implies that it is the robust phishing detector with a small amount of the false positives and high recall. The next is Gradient Boosting which confirms the advantage of ensemble learning in the capacity to represent intricate This study possesses various limitations in spite of good performance. The proposed framework will only be sustained by the URL-based structural, lexical, and behavioral features, without taking reference to webpage content, visual indicators, and email context, which may also improve the detection of advanced phishing attacks. Moreover, the information though mixed is on a given point in time and may not be able to document newly developed phishing methods and deceptive designed URLs. The models have been experimented in offline setting and real-world implementation of the models can have problems of concept drift, scalability, and latency that are to be investigated. The un-controlling of the hyperparameters of the models to their default levels to give a reasonable baseline comparison can threaten the internal validity of this study and may

not have been the optimal performance of each of the classifiers. Limits to external validity due to scope of datasets and their temporal relevance include changes in phishing strategies changing rapidly, and feature distributions changing over time (concept drift). The features based on URLs as the primary ones without the presence in webpage content, visual similarity, or email context that may be informative during extremely advanced attacks may also affect construct validity. Nevertheless, all these problems are solved through the use of a number of classifiers, stratified evaluation, and explainable analysis, which adds to the credibility of the results.

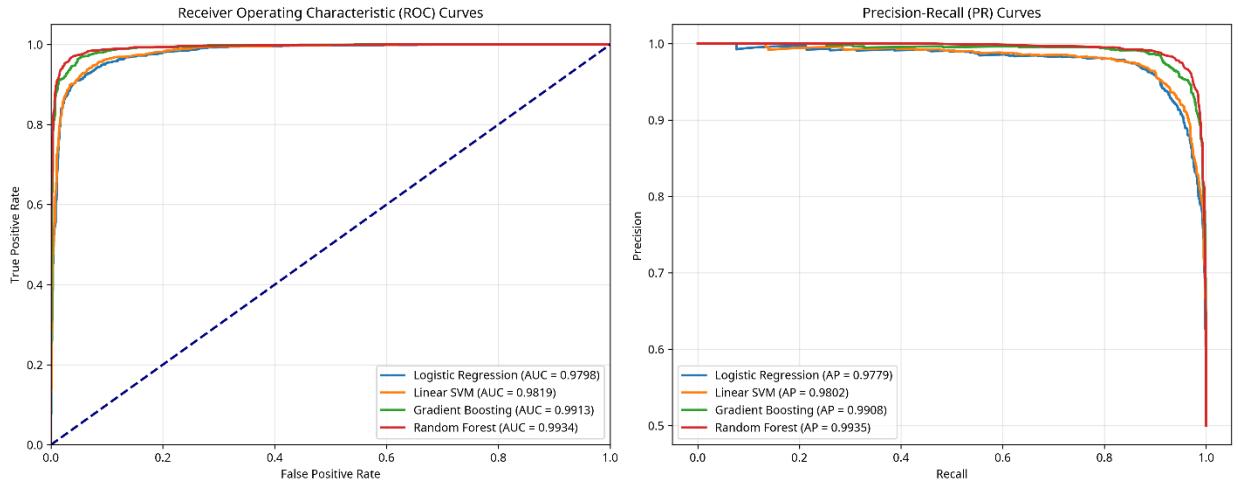

Figure 4. ROC and Precision-Recall curves together with the values of AUC and Average Precision of the Logistic Regression, Linear and Gradient Boosting and the Random Forest models.

The findings of this study can be directly applied in the real time phishing control systems on the web browsers, email gateways, and network security systems. The interpretable and light-weight nature of the selected URL features make them possible to make decisions within a short period without retrieving content, and thus, the framework is possible in large-scale and latency-sensitive contexts. Apart, the SHAP-based explanations are also capable of integration to augment the assurance of the analyst and support operations in the security by providing actionable knowledge regarding the perceived threats. As a result, the recommended plan can be applied as a feasible foundation in the development of transparent, adaptive, and dependable phishing defense mechanisms. Operational deployment In operation, the model may be deployed as a pre-filtering layer in browsers, email gateways, DNS resolvers and security proxies, where URLs are filtered before rendering a page or before being interacted with by a user. The SHAP explanations can be provided to the analysts in the form of a decision support, which makes it possible to quickly triage, justify the automated blocking behaviors, and trust the model-oriented processes of cybersecurity more.

5.0 Conclusion

This paper presented a URL structure and web traffic explanation machine learning model of phishing detection. The results provided with the broad application of feature engineering and statistical analysis indicated that phishing URLs possess certain structural, lexical, and behavioral traits, particularly with regard to the complexity of URLs and length of domains. Comparison of multiple classifiers revealed that ensemble-based classifiers are more effective than linear models and the results of the Random Forest model are the most balanced and trustworthy in all the evaluation measures. It is important to notice that the SHAP integration made the model more transparent in the sense that the effect of critical

features is displayed according to classification choices, which is a black-box constraint of traditional machine learning systems. Findings indicate that high predictions accuracy and interpretability can be both coupled and are required in security critical applications. Overall, the proposed solution is an effective, concise, and universalizable solution to real-time phishing identification that will be beneficial in developing more trustworthy and sturdier cybersecurity.

7.0 REFERENCES

[1] Mittal R, Singh SK, Kumar S, Khullar T, Kumar R, Gupta BB, Psannis K. Advanced Techniques and Best Practices for Phishing Detection. InCritical Phishing Defense Strategies and Digital Asset Protection 2025 (pp. 149-186). IGI Global Scientific Publishing. (2025)

[2] Barraclough PA, Fehringer G, Woodward J. Intelligent cyber-phishing detection for online. computers & security. 2021 May 1;104:102123. (2021)

[3] Fadziso T, Thaduri UR, Dekkati S, Ballamudi VK, Desamsetti H. Evolution of the cyber security threat: an overview of the scale of cyber threat. Digitalization & Sustainability Review. 2023;3(1):1-2. (2023)

[4] Zhang Z, Wu J, Lu N, Shi W, Liu Z. AdaptPUD: An accurate URL-based detection approach against tailored deceptive phishing websites. Computer Networks. 2025 Apr 25:111303. (2025)

[5] Nixon KA, Aimale V, Rowe RK. Spoof detection schemes. InHandbook of biometrics 2008 (pp. 403-423). Boston, MA: Springer US. (2008)

[6] Boulila E, Dacier M, Peroumal SP, Veys N, Aonzo S. A Closer Look At Modern Evasive Phishing Emails. InDSN 2025, 55th Annual IEEE/IFIP International Conference on Dependable Systems and Networks 2025 Jun 23. (2025)

[7] Kytidou E, Tsikriki T, Drosatos G, Rantos K. Machine learning techniques for phishing detection: A review of methods, challenges, and future directions. Intelligent Decision Technologies. 2025 Nov;19(6):4356-79. (2025)

[8] Patil S, Shekokar NM. A study of recent techniques to detect zero-day phishing attacks. InIntelligent approaches to cyber security 2023 Oct 11 (pp. 71-83). Chapman and Hall/CRC. (2023)

[9] Rahman RU, Tomar DS, Kumar P. A polymorphic code generation engine with obfuscation techniques for protecting web bot attacks. Iran Journal of Computer Science. 2025 Jul 9:1-26. (2025)

[10] Alazaidah R, Al-Shaikh A, Al-Mousa MR, Khafajah H, Samara G, Alzyoud M, Al-Shanableh N, Almatarneh S. Website phishing detection using machine learning techniques. Journal of Statistics Applications & Probability. 2024 Jan;13(1):119-29. (2024)

[11] Karim A, Shahroz M, Mustofa K, Belhaouari SB, Joga SR. Phishing detection system through hybrid machine learning based on URL. IEEE Access. 2023 Mar 3;11:36805-22. (2023)

[12] Darwish SM, Farhan DA, Elzoghabi AA. Building an effective classifier for phishing web pages detection: a quantum-inspired biomimetic paradigm suitable for big data analytics of cyber attacks. Biomimetics. 2023 May 9;8(2):197. (2023)

[13] Kailas S, Roopalakshmi R. 'Think Before You Click'-Malicious URL Detection in Cybersecurity: A Systematic Review and Research Roadmap. IEEE Access. 2025 Aug 21. (2025)

[14] Essien IA, Etim ED, Obuse E, Cadet E, Ajayi JO, Erigha ED, Babatunde LA. Neural network-based phishing attack detection and prevention systems. Journal of Frontiers in Multidisciplinary Research. 2021 Jul;2(2):222-38. (2021)

[15] PK N. AI-Driven Phishing Detection Using NLP and URL Analysis. (2025)
[16] Kara I, Ok M, Ozaday A. Characteristics of understanding URLs and domain names features: the detection of phishing websites with machine learning methods. IEEE Access. 2022 Nov 17;10:124420-8. (2022)
[17] Demidova N, Lawson P, Sloan J. Proactive Brand-Targeting Phishing Website Detection using a Hybrid Feature-based Approach with Machine Learning. InAPWG. EU Tech 2023. (2023)
[18] Zheng S. Analysis on Phishing Detection Methods. InITM Web of Conferences 2025 (Vol. 78, p. 02010). EDP Sciences. (2025)
[19] Ahmad S, Zaman M, Al-Shamayleh AS, Ahmad R, Abdulhamid SI, Ergen I, Akhunzada A. Across the spectrum in-depth review AI-based models for phishing detection. IEEE Open Journal of the Communications Society. 2024 Sep 17;6:2065-89. (2024)
[20] Ullah A, Shah RA, Nawaz SA, Ahmad N, Malik MH. Enhancing phishing detection, leveraging deep learning techniques. Journal of Computing & Biomedical Informatics. 2024 Feb 1. (2024)
[21] Graf R, Zeldovich M, Friedrich S. Comparing linear discriminant analysis and supervised learning algorithms for binary classification—A method comparison study. Biometrical Journal. 2024 Jan;66(1):2200098. (2024)
[22] Karim AA, Pardede E, Mann S. A feature-based model selection approach using web traffic for tourism data. International Journal of Web and Grid Services. 2024;20(3):342-59. (2024)
[23] Wyss E, Davidson D, De Carli L. What's in a URL? An Analysis of Hardcoded URLs in npm Packages. InProceedings of the 2024 Workshop on Software Supply Chain Offensive Research and Ecosystem Defenses 2023 Nov 19 (pp. 26-32). (2024)
[24] Schober P, Mascha EJ, Vetter TR. Statistics from A (agreement) to Z (z score): a guide to interpreting common measures of association, agreement, diagnostic accuracy, effect size, heterogeneity, and reliability in medical research. Anesthesia & Analgesia. 2021 Dec 1;133(6):1633-41. (2021)
[25] El Arid A, Nassreddine G. A Robust Voting-ML System in Identifying Phishing Websites Using Hybrid Ensemble Learning and Advanced Feature Selection Techniques. In2025 IEEE 4th International Conference on Computing and Machine Intelligence (ICMI) 2025 Apr 5 (pp. 1-5). IEEE. (2025)