Applied Mathematics & Information Sciences
*An International Journal*

http://dx.doi.org/10.18576/amis/180624

# Enhanced Phishing Detection: An Ensemble Stacking Model with DT-RFECV and SMOTE

*Mangayarkarasi Ramaiah*[1], *Vanmathi Chandrasekaran*[1], *Vikash Chand*[1], *Asokan Vasudevan*[2,3], *Suleiman Ibrahim Mohamma*[4,5,*], *Eddie Eu Hui Soon*[2], *Qusai Shambour*[6], *and Muhammad Turki Alshurideh*[7]

[1]School of Computer Science Engineering and Information Systems, Vellore Institute of Technology, 632014 Vellore, India
[2]Faculty of Business and Communications, INTI International University, Persiaran Perdana BBN Putra Nilai, 71800 Nilai, Negeri Sembilan, Malaysia
[3]Wekerle Business School, Budapest, Jázmin u. 10, 1083 Hungary
[4]Electronic Marketing and Social Media, Economic and Administrative Sciences Zarqa University, 13110 Zarqa, Jordan
[5]INTI International University, 71800 Negeri Sembilan, Malaysia
[6]Software Engineering Department, Hourani Center for Applied Scientific Research, Al-Ahliyya Amman University, 19111 Amman, Jordan
[7]Department of Marketing, School of Business, The University of Jordan, Amman 11942, Jordan

Received: 15 Aug. 2024, Revised: 5 Oct. 2024, Accepted: 10 Oct. 2024
Published online: 1 Nov. 2024

**Abstract:** Phishing websites are a significant threat, constantly evolving to deceive users into revealing sensitive information. While current anti-phishing systems rely on URLs, website content, and third-party data, they often struggle to keep pace with these dynamic scams. This study addresses these challenges by introducing a novel approach that analyzes the effectiveness of URL-based features, JavaScript characteristics, and anomaly-based indicators in detecting malicious web links. To overcome the issues of data imbalance and feature selection, our approach incorporates SMOTE oversampling and a Decision Tree-Recursive Feature Elimination cross-validation (DT-RFECV) wrapper method. The selected features are then used to train an ensemble stacking model that combines Decision Trees, Random Forests, and Bagging. The framework was rigorously evaluated on two benchmarking datasets and achieved impressive accuracy rates of 97.7% on Dataset-1 and 97.5% on Dataset-2 using ten features, underscoring the effectiveness of our approach. Our proposed framework significantly contributes to the internet community's defense against phishing scams with its unique features, ensemble model construction, and promising results.

**Keywords:** URLs, DT-RFECV, Machine learning, ensemble stacking model, phishing scam, financial inclusion

## 1 Introduction

The Internet has become a hostile environment where attacks can be launched rapidly and are difficult to prevent, detect, and trace. Safeguarding the core security principles—privacy, integrity, and accessibility is challenging. Mutual trust, once a cornerstone of the Internet's decentralized nature, has eroded due to the prevalence of malicious activities. However, the importance of Internet security is undeniable, especially for the growth of e-commerce. Phishing is a prominent network attack that capitalizes on human trust by mimicking legitimate websites. These fraudulent sites often replicate the appearance of reputable companies like eBay, Facebook, Amazon, and Microsoft to deceive users. By combining social engineering and technical expertise, attackers extract sensitive personal information. Phishing campaigns are typically initiated through deceptive emails, SMS messages, or social media posts, enticing victims to click on malicious links. Despite having existed for over three decades, phishing remains a widespread threat, resulting in significant annual financial inclusion. Phishing attacks are the leading cause of online security breaches, with their frequency increasing. According to Astra Security, phishing emails constitute approximately 1.2% of all emails sent, equating to 3.4 billion daily. The similarity between phishing and legitimate websites makes detection difficult for users, who often overlook URL details. Consequently, many phishing crimes go unreported. Anti-phishing tools primarily fall into four

* Corresponding author e-mail: dr_sliman@yahoo.com

categories: whitelist/blacklist, deep learning, machine learning, and heuristics [1,2,3,4,5].

Although blacklisting and whitelisting are the most widely used anti-phishing techniques, their ability to withstand zero-day attacks is uncertain because they rely on a centralized database to verify the legitimacy of the website [6,7,8,9]. Heuristic-based anti-phishing solutions rely on a third party to assess the website's validity. Although the web page's content, page ranking, and other aspects are included in the heuristic-based approach, the reliability of the data, which is taken from a third party, is controversial [10,11,12]. To combat the new phishing and to alleviate technical hindrances, machine learning-based anti-phishing solutions have been presented in various venues [13,14,15]. The main highlight of using AI techniques for the candidate task is to learn the hidden pattern to detect unseen fake information on the phishing web link. Over the past few years, ransomware has become the most prevalent type of cybercrime, with phishing being the most widely employed distribution method. Even while an ML-based anti-phishing solution might lessen the impact of a zero-day attack, it requires well-designed features from both legitimate and malicious URLs that are updated. An XGB-based anti-phishing solution has been built upon URL character order, hyperlink-specific and TF-IDF plaintext, and noisy character features of HTML. As we move forward and look at innovative rule-based techniques has been presented by [16,17,18] for detecting phishing scams in online banking. The candidate SVM-based phish-detector has been upon different features to detect the fake information. However, the prospective phish detector independently determines its capabilities from other sources such as search engines, network browser histories, and blacklists. Additionally, the features are language-dependent because they were taken from the webpage's content. Effective features maximize the detection rate of phishing crimes. Filter-based feature selection [19,20,21] and demonstrate encouraging outcomes [22,23,24].

Filter-based metrics [21,25] used statistical tools requiring less computing power. However, there is some uncertainty over its ability to forecast suitable features dynamically. In contrast to the filter tool, wrapped-based methods take advantage of the machine learning model's capacity to identify compelling features. Attempts to extract highly influential features [21,26,27,28] introduce a novel feature selection method based on the wrapper-method and yield superior results. When the number of features is huge, it takes longer to define them, but in the end, it improves the classifier's performance. Phishing scam detection presented [29,30,31] used diverse categories, URLs, domains, HTML and JavaScript, and abnormal features. Sixteen machine-learning models were trained using two datasets. One dataset had balanced classes (equal numbers of benign and malignant samples), while the other was imbalanced. Top ten significant features are extracted to train various machine learning models. While comparing the results in terms of accuracy, the XGBoost and Random Forest (RF) models' results are better than those of others. The challenges in developing Ml models are sufficient samples against the considered output class labels and suitable feature selection methodology. The presented work in this paper demonstrates the ML model's efficiency for phishing website detection using suitable data sampling techniques and efficient feature selection methods. To reduce response time, the language-independent phishing webpage detection mechanism operates without relying on external data. The presented novel approach utilizes features from multiple sources, including URL, address domain, JavaScript, URL file, and directory attributes. To optimize efficiency, a minimal set of features from diverse categories is used to train various machine learning models. Feature significance was determined using DT-RFECV (Decision-Tree-Recursive Feature Elimination).

Decision-Tree-Recursive Feature Elimination offers a valuable tool for feature selection, providing benefits such as feature ranking, improved model performance, computational efficiency, versatility, and enhanced interpretability. The selected features were then used to develop an ensemble stacking model. The key contributions of this research are summarized below:

- Predominant feature sets are computed using DT-RFECV.
- The significance of these derived features is analyzed through various ML models.
- An ensemble stacking model is designed to mitigate cyber-threats posed by phishing scams.
- The model's resilience is evaluated using two datasets.
- Results are compared to a recently released phishing detection framework to assess its competence.

The rest of the manuscript is organized in the following way. Section 2 examines earlier research and methods for identifying phishingscams. Section 3 depicts the information about the dataset used and features selection, Section 4 elaborates on the methodology applied in this research. The results are reported in Section 5, and the study's conclusions are presented in Section 6.

## 2 Related Works

This section analyses the various aspects of the ML-basedphishing detection frameworks demonstrated in multiple venues—an ML-based anti-phishing solution [32,33,34] to mitigate phishing scams. Experimentation was conducted using collected samples from various sources. Nineteen significant features were selected using Pearson correlation analysis. Various ML models were trained on features extracted from URLs, login forms, hyperlinks, CSS, and web identity. The detailed results

demonstrate the effectiveness of different feature types for the problem. Response time is low since the model is entirely built on client-side features. Similarly, Al-Shanableh et al. [35] and Jain & Gupta [36] utilized client-side features, mainly hyperlink information. ML models were constructed based on twelve features extracted from hyperlinks. Consequently, a positive aspect of the presented solution is its applicability to websites in any human language. Conversely, its effectiveness is contingent on the website being designed using HTML. To strengthen phishing scam detection, a two-phase enabled framework was built based on URL and source code features [37,38,39]. In the first phase, similarity-based attributes are used to generate a fingerprint, which is then compared to stored fingerprints to identify potential malicious websites. In the second phase, approximately 21 features extracted from URLs and source code are used to train an ensemble model employing Random Forest, XGBoost, and Extra Trees classifiers. Since no webpage exists independently on the internet, every webpage is connected to various resources, such as forwarding pages. In most cases, phishing attacks may neglect to conceal this information.

The frameworks presented in [40] use heterogeneous information networks (HIN) to understand the semantic and syntactic relationship among the various objects that constitute the web page and compute the Phish score for the nodes and nodes attributes [39] are used to train ML models. A deep learning-based phishing detection framework was presented by [41] based on the merits of character level and word level embedding for the input URL information. The study did not focus on data imbalance, which might lead to overfitting issues.

Recent research indicates that employing optimization techniques for hyperparameter tuning [42,43,44,45] and feature selection (Ramaiah et al., 2024) significantly enhances the performance of machine learning models. Similarly, in the context of phishing detection, authors in [46] utilize Genetic Algorithms (GA) to optimize hyperparameters for various machine learning models. The study employs three datasets to demonstrate its robustness against evolving phishing attacks. Although the inclusion of GA improves results, the iterative nature of the process increases computational time. To infer deeper insights into the syntactic and semantic information in the text extracted from source code, the authors in [47] employ various word embedding algorithms to enhance detection accuracy. The resulting word embeddings are used to generate feature vectors, subsequently engaged to design ensemble and multimodal phishing detectors. The phishing framework presented in [48] proposes eight features extracted from the URL, one from the plaintext character level and six from hyperlinks. An additional seven features from the literature were incorporated, resulting in 15 features used by the authors to compile the customized dataset. To mitigate performance degradation over time, a vanilla neural network (VNN) model trained using continual learning algorithms, Learning Without Forgetting (LWF), and Elastic Weight Consolidation (EWC). These CL algorithms enable the VNN to acquire new information while preserving previously learned knowledge.

To enhance the robustness of phishing website detection, the authors in [49] considered a dataset comprising 112 attributes. The study explores various scenarios to assess the resilience of the phishing detection framework. Data imbalance was addressed using SMOTEENN, and 13 constant features were identified and removed to improve response time. Subsequently, Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) were employed to reduce feature dimensionality. Based on the results, ML-based phishing detection models were unaffected by PCA and LDA, butremoving constant features significantly improved detection accuracy. Similarly, authors in [50] also employed PCA for dimensionality reduction on a balanced dataset. SVM and DNN models were trained and evaluated. It would be beneficial to include details about the PCA parameters, such as the number of principal components retained, and consider exploring additional ML models for a more comprehensive comparison. To address the technical challenges of small datasets, a large dataset containing more samples for both phishing and legitimate categories should be presented [51]. An Optimal Feature Vectorization Algorithm (OFVA) was introduced to extract 41 features, including 10 novel ones, effectively detecting phishing scams. Content-related features were excluded to reduce response time. Authors in [52] developed an SVM-based phishing detection model utilizing URL-based features. A chi-square metric was employed to select nine significant features from an initial set of sixteen—the SVM model with a polynomial kernel function performed better than its radial basis function counterpart.

The literature offers numerous robust solutions for detecting phishing websites using ML models (Table 1), but there's a need to cultivate the comprehensive nature of such models. This involves enabling ML models to understand phishing websites and develop resilience against emerging threats deeply. One of the challenges associated with the candidate problem is insufficient labeled samples. Very few publicly available datasets [50] maintain equal samples for benign and malign labels. In the cited literature, the work presented in [40,48], and [49] used datasets where the number of benign samples is higher than the number of malign samples. Conversely, Ejaz et al. [53], Bahaghigha et al. [49], Tamal et al. [51], and Shombot et al. [52] utilize datasets with more malign samples than benign samples.

They were training a machine learning model to detect phishing websites, andhaving an imbalanced dataset with significantly more benign than malicious samples can pose challenges. The model might become overly focused on recognizing benign patterns, leading to false negatives where phishing websites are incorrectly classified as safe. With fewer malicious samples, the

**Table 1:** Cutting-edge phishing detection modules

| Approach | Description | Feature Selection | Dataset | Limitations |
|---|---|---|---|---|
| Jain AK and Gupta BB [34] | Uses the URL and source code features. | Statistical tool | 1918 benign and 2141 phishing websites. | Model can detect websites built exclusively with HTML. |
| Jain and Gupta [36] | ML models uses the hyperlink features. | Uses the domain expertise. | 1116 benign and 1428 phishing websites. | Model can detect websites built solely with HTML |
| Rao and Pais [39] | Two-phase framework included similarity based and ensemble model. | Uses the domain expertise | 4097 phishing websites and 5438 benign websites. | Relies on source-code. |
| Guo et al. [40] | HinPhish utilizes link information from webpage objects to construct a HIN | Not used | 30,649 benign samples and 29,496 phish samples. | Webpage with more link may hinders the performance of the model. |
| Zheng et al. [41] | HDP-CNN identifies phishing URLs by leveraging both character and word-level representations. | Not used | 71,556 phishing URLs and 344,794 benign URLs. | The model's performance on large datasets is likely compromised due to severe class imbalance, potentially leading to overfitting. |
| Sarem,et.al [46] | Ensemble models hyper-parameters are computed using GA | Not used | UCI and two datasets from Mendeley | Response time is high. |
| Rao et.al [47] | Word embedding method outputs are used to train the ensemble and multimodal enabled phishing detector | Not used | 5076 benign 5438 phishing | Language dependant. Word embedding demands more storage. |
| Aljofey, and Qingshan [48] | The ML model employs a rich feature set including URL, hyperlink, and character-level TF-IDF features extracted from HTML's plain text. | Not used | 32,972 safe sites and 27,280 phishing URLs along with HTML codes. | To unveil the language-specific plain text content of a webpage, accessing its underlying HTML source code is essential. |
| Ejaz et. al [53] | VNN with CL has been used upon the samples collected from 2018-2020 | Not used | 99504 phishing and 81,082 benign | As new information learned increases the size of the learning parameters |
| Bahaghighat et. al [49] | ML models built upon PCA and LDA were presented | Constant features are removed using statistical tool | 58,000 benign and 30,647 phishing | Without feature extraction, the number of features are high |
| Elumalai[50] | SVM and DNN models trained upon the public dataset | PCA | 5000 benign, 5000 malign samples | Exploration of other ML models could be established. |
| Tamal et al. [51] | Intra URL features are used to train 15 ML models | Optimal feature vectorization algorithm | 247950 instances, of which 128541 are from phishing URLs and 119409 are from legitimate URLs. | Content related features were discarded. |
| Shombot et. al [52] | SVM based anti-phishing framework built upon the URL based features. | Chi-Square | 548 benign, 805 phishing URLs | Dataset size is small |

model might struggle to learn subtle phishing characteristics, reducing its ability to identify them accurately. Conversely, if the dataset has too many malicious samples, the model might become overly sensitive to malicious patterns, leading to false positives where benign websites are incorrectly flagged as phishing. In both cases, a high accuracy score might not accurately reflect the model's performance, as the model could predict everything as benign or malicious to achieve high accuracy. The presented phishing detection framework employs appropriate data sampling techniques to ensure the generalization of machine learning models. Contrary to the literature that utilizes statistical tools [34, 52] for feature selection, the studies presented in [36, 39] rely on domain expertise to identify the most significant features. In contrast, the research in [40, 41] employ all features from the dataset, potentially leading to increased prediction time. Hence, to mitigate the mentioned technical hindrances, the presented ML model is built upon the features derived through the DT-REF algorithm.

## 3 Proposed Methodology

This section provides a detailed description of the presented anti-phishing solution. Exploratory Data Analysis (EDA) techniques are applied in the pre-processing phase to analyze the features. Data sampling is employed to balance the class distribution. DT_RFECV is used to select the most essential features. Next, machine learning models are trained on 80% of the data. The trained models are then evaluated on the remaining 20% of the data for testing. After assessing the performance of the individual models, an ensemble stacking model is built that combines Decision Trees (DT), Random Forests (RF), and Bagging Classifiers (BC) for improved accuracy. The candidate proposal's architecture view is shown in Figure 1.

### *3.1 Dataset Description*

The proposed framework is experimented on two datasets https://data.mendeley.com/datasets/h3cgnj8hft/1: DS-1 is

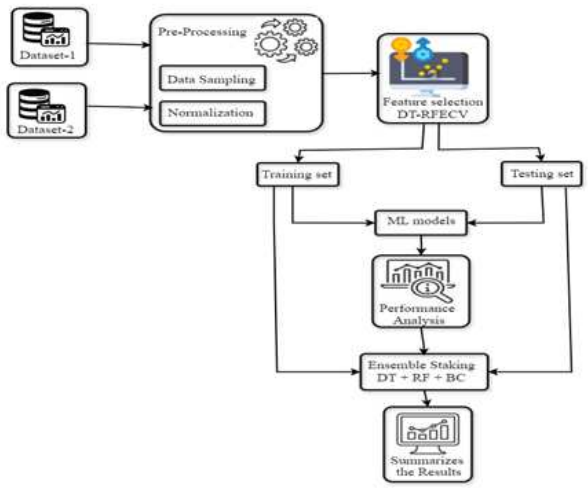



**Fig. 1:** The architecture of the Proposed Anti-phishing Model.

a compilation of characteristics extracted from both phishing and legitimate websites. 48 features and 10,000 samples are available on both phishing and legitimate labels. The graphical representation is displayed in Figure 2. Four categories of features constitute the dataset, offering better insight into web pages: sixteen address-based features, four domain-based features, twenty-one Abnormal-based features, and six JavaScript-based features.

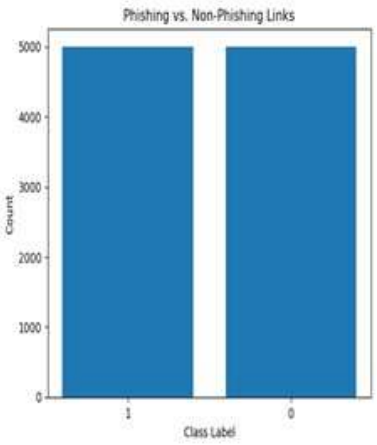



**Fig. 2:** Sample distribution statistics for dataset-1 (DS-1).

https://data.mendeley.com/datasets/72ptz43s9v/1: DS2 is an extensive collection of examples of both authentic and phishing websites. The collection contains 88,647 cases, 58,000 instances of legitimate websites, and 30,647 instances of phishing websites. Sample distribution statistics of dataset-2 can be found in Figure 3. 111 distinct features are provided to differentiate the anomaly entities uniquely. Nineteen URL properties, twenty-one Domain properties, eighteen URL file properties, eighteen URL directory properties, and fifteen more properties make up the distinct feature categories in the dataset.

### *3.2 Pre-processing*

The dataset DS-1 has an equal number of phishing and legitimate samples. It has been ensured that the dataset is devoid of null values. This indicates that each instance's features have valid values and no missing data. The absence of null values enables continuous analysis and modeling without the need for imputation or managing missing values. Each instance in the dataset is unique, indicating no duplicate records are present. The uniqueness of tuples ensures that each instance contributes independently to the machine learning process, preventing any duplication bias that could distort the results. A box plot analysis was conducted on the dataset to resolve this issue. The box plot analysis revealed that no outliers were identified in the dataset. This suggests that the data points don't contain extreme values that would skew the analysis or affect the model's functionality, and instead fall within a reasonable range. In dataset 2, all instances with null values have been eliminated. This procedure verifies that the remaining data is comprehensive and contains no missing valuesor redundant recordings. The dataset DS-2 had an imbalanced distribution of legitimate and phishing instances, with 58,000 legitimate instances and 30,647 phishing instances, respectively. Synthetic Minority Over-Sampling Technique (SMOTE) analysis addressed this class imbalance and provideda more balanced dataset. To match the number of phishing cases in the majority class, SMOTE creates fictional instances of the minority class, which, in this case, are valid instances. Consequently, the dataset was rebalanced to contain 58,000 instances of phishing and legitimate connections. The sample statistics before and after applying the SMOTE can be found in Figures 3(a) and 3(b). This phase ensures the quality and integrity of the dataset prior to analysis.

### *3.3 Features Selection*

Feature selection is a crucial piece of information to have before training the model. The redundant or highly correlated independent features in extremely high-dimensional data usually generate problems for models. This could increase the training time needed for the machine learning model, worsening the over fitting issues. This section describes selecting the most

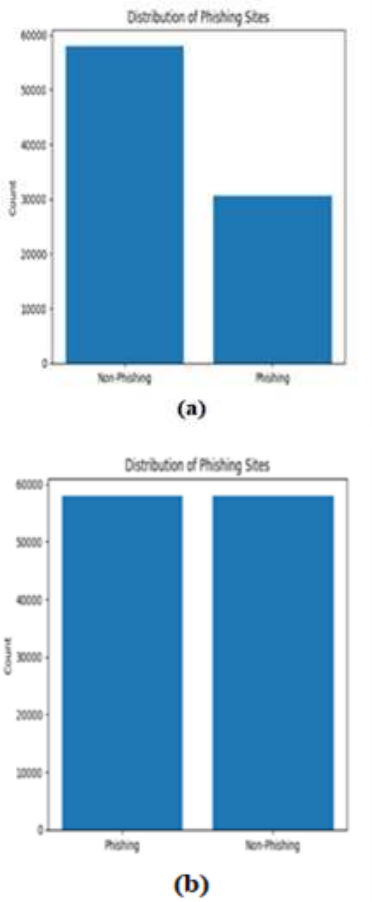



**Fig. 3:** (a) Sample Distribution Statistics of DS-2, (b) Sample Distribution of DS-2 after Applying SMOTE.

important features from the data. The method of Recursive feature elimination (RFE) is used along with a decision tree (DT) model and cross-validation (CV) to evaluate feature importance. DT-RFECV utilized 10-fold cross-validation with StratifiedKFold. This method ensures each data split maintains the original class distribution, which is crucial for classification tasks to avoid biases caused by class imbalance. During the recursive feature elimination (RFE) process, features are iteratively eliminated, and accuracy metrics are computed at each step to measure the impact on model performance. This aids in understanding the contribution of individual features, and the optimal feature set is chosen based on the highest overall accuracy. Subsequently, the selected feature set undergoes evaluation using 10-fold cross-validation. Here, the decision tree (DT) model is trained and evaluated ten times, each time with a different fold as the validation set. The average and standard deviation of accuracy across these iterations provide insights into model robustness and generalization to unseen data. Throughout the cross-validation, internal accuracy metrics from the RFE process are used to assess feature selection effectiveness in classifying attack and non-attack classes. Notably, the accuracy values determining the best model within each training sample set are based on final assessments using the test dataset, not internal metrics from RFE or cross-validation iterations. The top ten features derived using DT-REFCV upon dataset-1 and dataset-2 are furnished in Table 2 and Table 3.

**Table 2:** Top Ten features from Dataset-1 (DS-1)

| | |
|---|---|
| Feature_set | PctExtHyperlinks(AF6), PctExtNullSelfRedirectHyperlinksRT(JF6), FrequentDomainNameMismatch(AF14) , InsecureForms(AF9), PctNullSelfRedirectHyperlinks(AF13), NumDash(AdF5), PctExtResourceUrls(AF7), SubmitInfoToEmail(AF18), PathLevel(AdF3), IframeOrFrame(AF19) |

**Table 3:** Top Ten features from Dataset-2 (DS-2)

| | |
|---|---|
| Feature_set | qty_dot_directory(UDF1),time_domain_activation(OF5), directory_length(UDF18),asn_ip(OF4),time_response(OF2), length_url(UF19),qty_dot_domain(DF1),ttl_hostname(OF10), time_domain_expiration(OF6), qty_nameservers(OF6) |

## *3.4 Machine Learning Models*

Once the dominant features were selected, various machine learning models are trained upon them. To test their efficacy, all the trained ML models are tested using a test dataset. This ensures they can adapt to unseen threats, a must in this ever-changing battle. Then, the ML models' performance is analysed to select the promising models. After analyzing the models' results, an ensemble stacking model has been designed using DT, RF, and Bagging.

### 3.4.1 Decision Tree (DT)

Regression and classification are two applications of nonparametric supervised learning techniques, such as decision trees. By learning decision rules derived from the features of the data, a decision tree classifier creates a model that predicts when the target variable will be estimated. The if-then-else decision rule is linked to decision tree algorithms. Deeper decision rules and a more suitable model result from the deeper tree. Classifiers construct an adjudication tree Tree-like structures. The method separates the dataset into smaller, more manageable chunks and simultaneously enhances the decision tree that goes along with it. The end product will be a tree with leaf and decision nodes. The leaf nodes are used to convey a classification or judgment. A decision node is any node that has two or more branches.

3.4.2 Random Forest (RF)

Using bootstrap aggregation or bagging, several classification and regression trees (CART) are combined to create the supervised learning technique known as random forest (RF). Several regression trees referred to as 'ntree' were built, and random subsets of independent variables (referred to as "mtry") were used for each split of a tree. The dependent variable, phishing or legitimate,is predicted using the average of all the trees. The out-of-bag Samples not included in the bootstrap set were used for an internal cross-validation accuracy and variable significance evaluation.

3.4.3 Bagging Classifier (BC)

Bagging is an ensemble learning method incorporating multiple base classifiers, typically decision trees, by training them on distinct bootstrap samples from the original dataset. The ultimate classification is determined by aggregating the predictions of these primary classifiers. By incorporating diversity among the base models, bagging decreases overfitting and enhances generalization.

3.4.4 Proposed Ensemble Stacking Model

One way to create effective classifiers that outperform conventional ML classifiers in classification accuracy is to use the ensemble technique. Stacking stands out as a potent ensemble learning method that amalgamates the predictions generated by several individual models, culminating in a final prediction that is more resilient and accurate. The training ensemble model includes two levels. Firstly, Base Learnersare the individual models trained on the original data. The candidate experiment uses three different models DT, RF, and Bagging. The other level of the model is Meta-Learner.Upon the completion of training for the base learners, their predictions are employed as input to facilitate the training of a meta-learner. The meta-learner learns how to combine the predictions of the base learners to make a final prediction. ELNet is used in this with meta-learner. ELNet, also known as Elastic Net Regularization, is a regression and classification technique that combines L1 and L2regularization. Combining these two techniques can lead to better performance than using either alone. ELNet regularization can help reduce the final model's variance, making it more robust to noise and outliers. Because ELNet models produce coefficients indicating the relative relevance of each feature, they are easier to interpret than other ensemble models like random forests. The presented ensemble stacking model steps are furnished in Figure 4.

**Input:**
Dataset $D_s = \{(x_1,y_1),(x_2,y_2),.....(x_n,y_n)$
Base Learners $f_1$ = RF, $f_2$ = XGBoost, $f_3$ = NN
Meta Learner f = ELNET
**Algorithm:**
for t = 1,2,3:
first stage is to train the $f_1$, $f_2$, and $f_3$
$g_t = f_t(D_s)$;
end
for i = 1,2,3….n:
for t = 1,2,3:
for each sample, generating the new feature vector
$V_{it} = g_t(x_t)$;
end
$D_s' = D_s' \cup ((V_{i1},V_{i2},V_{i3}),y_t)$;
end
second stage, training the meta learner
$g' = f(D_s')$;
end
**Output**:
$H(x) = g'(g_1(x),g_2(x),g_3(x))$

**Fig. 4:** Algorithm for Ensemble Stacked Classifier.

## 4 Experimented Results and discussion

The experiment is conducted on a Machine booted with Windows 11 operating system and powered with processor: Intel(R) Xeon(R) CPU @ 2.20GHz, RAM 31GB. The proposed framework is implemented Python Version: 3.10.10. To design the machine models, Pandas, NumPy, Scikit-Learn (sklearn) libraries were used. For training and testing the models 80:20 ratio is used for providing the samples. Performance measures including recall, accuracy, precision, F1 score, TPR, and FPR are employed to assess the recommended anti-phishing solution. The accuracy score (A) is the most commonly used metric to evaluate model performance in binary and multi-class classification problems. The corresponding mathematical expression is given in equation 1. A machine learning model precision is its ability to identify true positives. The model's recall accuracy is also measured. Sensitivity is real positive interest. The mathematical expression for precision and recall are furnished in equation 2 and 3. The trade-off between recall and precision evaluates the ML model's accuracy while accounting for FP and FN, which can be assessed by the F1 score (equation 4).

$$A = \frac{T^+ + T^-}{T^+ + T^- + F^+ + F^-} \tag{1}$$

$$P = \frac{T^+}{T^+ + F^+} \tag{2}$$

$$R = \frac{T^+}{T^+ + F^-} \tag{3}$$

$$F1score = 2\frac{P*R}{P+R} \tag{4}$$

**Table 4:** Significant feature through DT_RFECV from Dataset-1 (DS-1)

| Type_of_features | Feature_number |
|---|---|
| Address _based | F3,F5 |
| Abnormal-based | F6,F7,F9,F13,F14,F18,F19 |
| JavaScript-based | F6 |

**Table 5:** Tested Results of various ML models upon (DS-1)

| Table 4DT_RFECV | | | | |
|---|---|---|---|---|
| Methods | A | P | R | F1_Score |
| DT | 0.971 | 0.963 | 0.981 | 0.972 |
| RF | 0.976 | 0.973 | 0.980 | 0.976 |
| LR | 0.911 | 0.903 | 0.923 | 0.913 |
| GRB | 0.970 | 0.965 | 0.975 | 0.970 |
| ADB | 0.962 | 0.960 | 0.964 | 0.962 |
| SVM | 0.911 | 0.901 | 0.925 | 0.913 |
| KNN | 0.940 | 0.943 | 0.937 | 0.940 |
| GNB | 0.794 | 0.917 | 0.652 | 0.762 |
| BC | 0.976 | 0.974 | 0.978 | 0.976 |
| P-Stacking | 0.977 | 0.975 | 0.979 | 0.977 |

True positives (TPs) are those situations in which the model accurately predicts positive outcomes. True negatives (TN) are instances in which the model accurately predicts a negative result. False positives, or FP, are instances in which the model predicts positive outcomes inaccurately. Last but not least, FN denotes false negatives—situations in which the model predicts negative outcomes inaccurately. The metrics TPR, FPR, and AUC are also computed to evaluate the binary classification models. The comparable mathematical expression of FPR (False Positive Rate) and TPR (True Positive Rate) is represented in Equation 5 and Equation 6. ROC reveals the relationship among TPR and FPR. A model with minimum value compared to FPR is broadly accepted. In contrast, a higher value is appreciated for TPR. AUC is the measurement of the area underneath the ROC curve.

$$TPR = \frac{T^+}{T^+ + F^-} \tag{5}$$

$$FPR = \frac{F^+}{F^+ + T^-} \tag{6}$$

The proposed framework is evaluated on two Mendeley datasets. The first experiment 1, considered 48 features from Dataset-1. DT-RFECV identified the top ten features, which are listed in Table 4. by their numbers and in Table 2 by their names. Then various machine learning (ML) models are trained on these selected features. The results are reported in Table 5. As shown in Table 5, decision tree (DT), random forest (RF), and bagging classifier (BC) models performed better than others. Notably, the ensemble stacking model did a better job in terms of accuracy, and F1-score than that of the other models. This experiment suggests that the features selected by DT-RFECV significantly improve the ability of ML models to accurately classify phishing web links. ROC curves obtained through diverse ML models can be found in Figure 5.

**Table 6:** Tested Results with the cutting-edge method's results (DS-1)

| Methods | A | P | R |
|---|---|---|---|
| [11]RF | 0.965 | 0.962 | 0.974 |
| [11]BC | 0.963 | 0.962 | 0.973 |
| [11]DT | 0.957 | 0.960 | 0.962 |
| [19]GA-ADB | 0.972 | 0.971 | 0.972 |
| [19]GA-BC | 0.975 | 0.972 | 0.978 |
| [24]DNN | 0.997 | 0.998 | 0.996 |
| [24]SVM | 0.999 | 1.0 | 0.998 |
| P-Stacking | 0.977 | 0.975 | 0.979 |

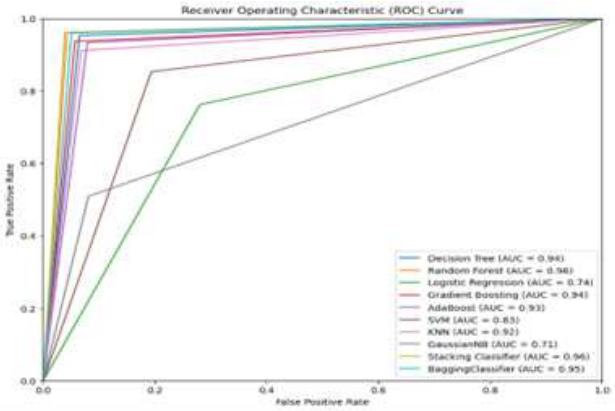



**Fig. 5:** Comparison of AUC values of various ML models.

The Figure 5 reveals RF and Stacking model's results are comparable and better results than the other models. These studies also evaluated anti-phishing solutions on the same datasets used by the proposed experiment. According to Table 5, the frameworks in [54] used all 48 features for their machine learning models. In Almomani et al. [54] and Al-Sarem et al. [46], sixteen classifiers were evaluated, with RF, BC, and DT models performing best. The anti-phishing solution in [46] employed Genetic Optimization to fine-tune hyperparameters, achieving the best results with GA-ADB (AdaBoost), GA-BC (Bagging), and GA-enabled stacking models Notably, the proposed P-Stacking model outperforms the solutions in [45] despite using only the top ten features from Dataset-1.

In the point of security analysis, the candidate framework become resistant to the iframe-attack (IframeOrFrame). Although phishing scams are typically

**Table 7:** Significant feature through DT_FS from Dataset-2 (DS-2)

| Type_of_features (DS-2) | Feature_number |
|---|---|
| URL directory | F1,F18 |
| URL features | F19 |
| Domain features | F1 |
| Others | F2,F4,F5,F6,F8,F10 |

**Table 8:** Tested Results of various ML models upon (DS-2)

| Methods | A | P | R | F1_S |
|---|---|---|---|---|
| DT | 0.96 | 0.96 | 0.960 | 0.96 |
| RF | 0.975 | 0.969 | 0.981 | 0.975 |
| LR | 0.910 | 0.922 | 0.894 | 0.908 |
| GRB | 0.954 | 0.942 | 0.966 | 0.954 |
| ADB | 0.935 | 0.925 | 0.945 | 0.935 |
| SVM | 0.762 | 0.733 | 0.816 | 0.772 |
| KNN | 0.893 | 0.875 | 0.914 | 0.894 |
| GNB | 0.829 | 0.927 | 0.710 | 0.804 |
| BC | 0.971 | 0.971 | 0.971 | 0.971 |
| P-Stacking | 0.975 | 0.971 | 0.978 | 0.975 |

triggered by the presence of an external link or an empty link, the work that is being presented has features such as PctExtHyperlinks, PctExtNullSelfRedirectHyperlinksRT, and PctNullSelfRedirectHyperlinks that strengthen its defences against malicious resources and the effects of self-redirection. The FrequentDomainNameMismatch vulnerability could lead to detect man-in-the-middle attack by hackers.

In Experiment 2, the Dataset-2 has 111 features. The results obtained through various conventional ML models along with the ensemble stacking model using the significant feature through DT-RFECV are furnished in Table 7. Features derived through the DT-RFECV are provided in Table 7 and the corresponding feature names are furnished in Table 3. Table 8 displays the results obtained through the presented ensemble stacking model along with the other baseline models. While observing the results presented in Table 8, the presented model showing better performance than the baseline models. The ROC curves obtained by the various ML models can be found in Figure 6.

To portray the superiority of the presented work, the experimented results are compared with the results in [46, 49] and furnished in Table 9. For the candidate dataset, the frameworks in [46] apply optimization techniques to derive the hyper-parameters of various machine learning models. Notably, the anti-phishing frameworks in [46] utilizes all 111 features, whereas the presented solution achieves better results using only ten features. The phishing scam detection method proposed in [49] employs SMOTEENN to address the class imbalance issue. To reduce dimensionality, statistical methods were used to eliminate 13 constant features and used PCA. The

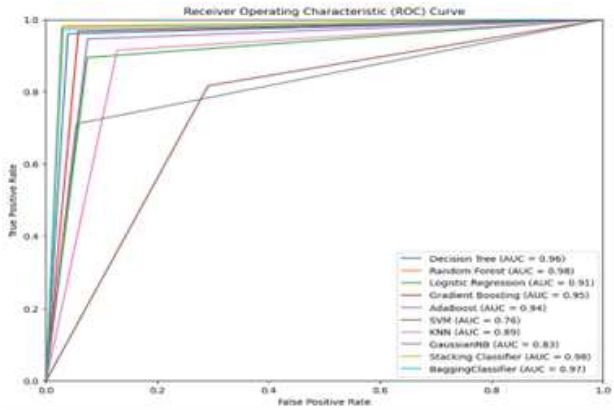



**Fig. 6:** Comparison of AUC values of various Machine learning models.

**Table 9:** Comparative results with the cutting-edge methods (DS-2)

| Methods | A | P | R |
|---|---|---|---|
| [19]GA-ADB | 0.940 | 0.909 | 0.920 |
| [19]GA-RF | 0.964 | 0.946 | 0.950 |
| [19]GA-XGB | 0.973 | 0.962 | 0.961 |
| [19]GA-BC | 0.969 | 0.953 | 0.959 |
| [23]LR | 0.953 | 0.941 | 0.968 |
| [23]NB | 0.930 | 0.909 | 0.961 |
| [23]XGB | 0.992 | 0.991 | 0.994 |
| P-Stacking | 0.975 | 0.971 | 0.978 |

results obtained in [49] were then compared with the proposed ensemble stacking model. Compared to XGBoost, the performance of the proposed method is better interms of number of features and method used. The graphical representation of the same is shown in Figure 7.

In terms of security analysis, the primary component that has the biggest impact on the outcome is the features that DT_RFECV listed in Table 7. Few of the semantic 10 features from DS-2 relate to URLs, URL directories, and domain-based networks. The highly significant features from six other categories time_domain_activation (OF5), asn_ip (OF4), time_response (OF2), ttl_hostname (OF10), time_domain_expiration (OF6), and qty_nameservers(OF6) become the cause of the guaranteed accuracy in phishing scam detection. The proposed fine-tuned stacking ensemble method achieves superior performance on Dataset 1 compared to recent works [46,49]. This was evident in both accuracy and recall metrics. Additionally, for Dataset 2, our method surpassed the solution presented in [46,49].

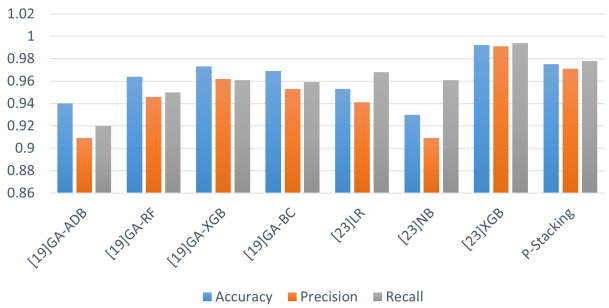



**Fig. 7:** Tested results using DS-2 along with its counterpart methods.

## 5 Conclusion

Phishing, a pervasive and harmful cyberattack, employs deception to obtain sensitive information. This research leverages machine learning to address the evolving nature of phishing threats. To overcome the challenges of sample imbalance and feature selection, the proposed approach incorporates SMOTE oversampling and a Decision Tree-Recursive Feature Elimination (DT-RFECV) wrapper method. DT-RFECV calculates the importance of features and utilizes cross-validation to prevent overfitting. The study identified two promising feature subsets for each dataset using DT-RFECV. To evaluate their effectiveness, various ML models are trained and tested. Random forests, decision trees, and bagging models demonstrated reliable predictive capabilities. Subsequently, a stacking ensemble model is developed to improve performance further. The proposed model's results are compared with recent anti-phishing solutions, demonstrating superior performance using fewer features according to quantitative metrics. Furthermore, the framework is designed to resist common cyberattacks in the IoT environment, including iframe attacks, man-in-the-middle attacks, and vulnerabilities related to null and external links. As future work, the solution can be extended to mitigate phishing scams on blockchain platforms. Additionally, we will establish generalizability metrics for the models to ensure their adaptability to new contexts.

## Acknowledgement

The authors thank all the respondents who provided valuable responses and support for the survey. They offer special gratitude to INTI International for publishing the research work, particularly to INTI International University for funding its publication, and acknowledge the partial funding support provided by the Electronic Marketing and Social Media Department, Economic and Administrative Sciences, Zarqa University.

## Funding

The authors offer special gratitude to INTI International University for the opportunity to conduct research and publish the research work. In particular, the authors would like to thank INTI International University for funding the publication of this research work. Also, we extend our heartfelt gratitude to all research participants for their valuable contributions, which have been integral to the success of this study.

## Conflict of Interest

The authors have no conflict of interest to declare.

## References

[1] A.M. Al-Adamat, M.K. Alserhan, L.S. Mohammad, D. Singh, S.I.S. Al-Hawary, A.A. Mohammad, M.F. Hunitie, The Impact of Digital Marketing Tools on Customer Loyalty of Jordanian Islamic Banks. In *Emerging Trends and Innovation in Business and Finance* (pp. 105-118). Singapore: Springer Nature Singapore (2023).

[2] M.S. Al-Batah, E.R. Al-Kwaldeh, M. Abdel Wahed, M. Alzyoud, N. Al-Shanableh, Enhancement over DBSCAN Satellite Spatial Data Clustering. *Journal of Electrical and Computer Engineering*,**2024**, 2330624 (2024).

[3] M.S. Al-Batah, M.S. Alzboon, M. Alzyoud, N. Al-Shanableh, Enhancing Image Cryptography Performance with Block Left Rotation Operations. *Applied Computational Intelligence and Soft Computing*,**2024**, 3641927 (2024).

[4] M.M. Alani, H. Tawfik, Phishnot: A cloud-based machine-learning approach to phishing url detection. *Computer Networks*,**218**, 109407 (2022).

[5] A. Adwan, M. Alsoud, The impact of brand's effectiveness on navigating issues related to diversity equity and inclusion. *Uncertain Supply Chain Management*,**12**, 2101-2112 (2024).

[6] F.M. Aldaihani, A.A. Mohammad, H. AlChahadat, S.I.S. Al-Hawary, M.F. Almaaitah, N.A. Al-Husban, A. Mohammad, Customers' perception of the social responsibility in the private hospitals in Greater Amman. In *The effect of information technology on business and marketing intelligence systems* (pp. 2177-2191). Cham: Springer International Publishing (2023).

[7] F.A. Al-Fakeh, M.S. Al-Shaikh, S.I.S. Al-Hawary, L.S. Mohammad, D. Singh, A.A. Mohammad, M.H. Al-Safadi, The Impact of Integrated Marketing Communications Tools on Achieving Competitive Advantage in Jordanian Universities. In *Emerging Trends and Innovation in Business and Finance* (pp. 149-165). Singapore: Springer Nature Singapore (2023).

[8] A.S. Al-Adwan, H. Berger, Exploring physicians' behavioural intention toward the adoption of electronic health records: an empirical study from Jordan. *International Journal of Healthcare Technology and Management*,**15**, 89-111 (2015).

[9] S. Purkait, Examining the effectiveness of phishing filters against DNS based phishing attacks. *Information & Computer Security*,**23**, 333-346 (2015).

[10] R. Rao, S.T. Ali, Phishshield: a desktop application to detect phishing webpages through heuristic approach. *Procedia Computer Science*,**54**, 147-156 (2015).

[11] Q.Y. Shambour, M.M. Abualhaj, A. Abu-Shareha, A.H. Hussein, Q.M. Kharma, Mitigating Healthcare Information Overload: a Trust-aware Multi-Criteria Collaborative Filtering Model. *Journal of Applied Data Sciences*,**5**, 1134-1146 (2024).

[12] R. Al Khouri, M. Al Fauri, The Impact of Working Capital Management on the Profitability of Jordanian Companies Listed on the Amman Stock Exchange. *Al-Balqa Journal for Research and Studies*,**26**, 77-97 (2023).

[13] D.A. Al-Husban, S.I.S. Al-Hawary, I.R. AlTaweel, N.A. Al-Husban, M.F. Almaaitah, F.M. Aldaihani, D.I. Mohammad, The impact of intellectual capital on competitive capabilities: evidence from firms listed in ASE. In *The effect of information technology on business and marketing intelligence systems* (pp. 1707-1723). Cham: Springer International Publishing (2023).

[14] M.I. Alkhawaldeh, F.M. Aldaihani, B.A. Al-Zyoud, S.I.S. Al-Hawary, N.A. Shamaileh, A.A. Mohammad, O.A. Al-Adamat, Impact of internal marketing practices on intention to stay in commercial banks in Jordan. In *The effect of information technology on business and marketing intelligence systems* (pp. 2231-2247). Cham: Springer International Publishing (2023).

[15] R. Yang, K. Zheng, B. Wu, C. Wu, X. Wang, Phishing website detection based on deep convolutional neural network and random forest ensemble learning. *Sensors*,**21**, 8281 (2021).

[16] M.S. Alshura, S.S. Tayeh, Y.S. Melhem, F.N. Al-Shaikh, H.M. Almomani, F.L. Aityassine, A.A. Mohammad, Authentic leadership and its impact on sustainable performance: the mediating role of knowledge ability in Jordan customs department. In *The effect of information technology on business and marketing intelligence systems* (pp. 1437-1454). Cham: Springer International Publishing (2023).

[17] A.A. Mohammad, I.A. Khanfar, B. Al Oraini, A. Vasudevan, I.M. Suleiman, M. Ala'a, User acceptance of health information technologies (HIT): an application of the theory of planned behavior. *Data and Metadata*,**3**, 394-394 (2024).

[18] M. Moghimi, A.Y. Varjani, New rule-based phishing detection method. *Expert systems with applications*,**53**, 231-242 (2016).

[19] N. Al-shanableh, M. Alzyoud, R.Y. Al-husban, N.M. Alshanableh, A. Al-Oun, M.S. Al-Batah, S. Alzboon, Advanced Ensemble Machine Learning Techniques for Optimizing Diabetes Mellitus Prognostication: A Detailed Examination of Hospital Data. *Data and Metadata*,**3**, 363-363 (2024).

[20] N. Al-shanableh, M.S. Alzyoud, E. Nashnush, Enhancing Email Spam Detection Through Ensemble Machine Learning: A Comprehensive Evaluation Of Model Integration And Performance. *Communications of the IIMA*,**22**, 2 (2024).

[21] M. Ramaiah, V. Chandrasekaran, V. Ravi, N. Kumar, An intrusion detection system using optimized deep neural network architecture. *Transactions on Emerging Telecommunications Technologies*,**32**, e4221 (2021).

[22] A.S. Al-Adwan, M. Alsoud, N. Li, T.E. Majali, J. Smedley, A. Habibi, Unlocking future learning: Exploring higher education students' intention to adopt meta-education. *Heliyon*,**10**, e29544 (2024).

[23] M. Alsharaiah, M. Abualhaj, L. Baniata, A. Al-saaidah, Q. Kharma, M. Al-Zyoud, An innovative network intrusion detection system (NIDS): Hierarchical deep learning model based on Unsw-Nb15 dataset. *International Journal of Data and Network Science*,**8**, 709-722 (2024).

[24] R. Mangayarkarasi, C. Vanmathi, V. Ravi, A robust malware traffic classifier to combat security breaches in industry 4.0 applications. *Concurrency and Computation: Practice and Experience*, e7772 (2023).

[25] A.A. Mohammad, I.A. Khanfar, B. Al-Oraini, A. Vasudevan, I.M. Suleiman, Z. Fei, Predictive analytics on artificial intelligence in supply chain optimization. *Data and Metadata*,**3**, 395-395 (2024).

[26] S. Abusaleh, M. Arabasy, M. Abukeshek, T. Qarem, Impacts of E-learning on the Efficiency of Interior Design Education (A comparative study about the efficiency of interior design education before and during the novel Coronavirus (COVID-19) pandemic). *Al-Balqa Journal for Research and Studies*,**27**, 47-63 (2024).

[27] H. Hmoud, A.S. Al-Adwan, O. Horani, H. Yaseen, J. Al Zoubi, Factors influencing business intelligence adoption by higher education institutions. *Journal of Open Innovation: Technology, Market, and Complexity*,**9**, 100111 (2023).

[28] A.A. Mohammad, F.L. Aityassine, Z.N. Al-fugaha, M. Alshurideh, N.S. Alajarmeh, A.A. Al-Momani, A.M. Al-Adamat, The Impact of Influencer Marketing on Brand Perception: A Study of Jordanian Customers Influenced on Social Media Platforms. In *Business Analytical Capabilities and Artificial Intelligence-Enabled Analytics: Applications and Challenges in the Digital Era* (pp. 363-376). Cham: Springer Nature Switzerland (2024).

[29] A.A. Mohammad, M.Y. Barghouth, N.A. Al-Husban, F.M. Aldaihani, D.A. Al-Husban, A.A. Lemoun, S.I.S. Al-Hawary, Does Social Media Marketing Affect Marketing Performance. In *Emerging Trends and Innovation in Business and Finance* (pp. 21-34). Singapore: Springer Nature Singapore (2023).

[30] M.M. Abualhaj, Q.Y. Shambour, A. Alsaaidah, A. Abu-Shareha, S. Al-Khatib, M.O. Hiari, Enhancing Spam Detection Using Hybrid of Harris Hawks and Firefly Optimization Algorithms. *Journal of Applied Data Sciences*,**5**, 901-911 (2024).

[31] R. Ghoneim, M. Arabasy, The Role of Artworks of Architectural Design in Emphasizing the Arab Identity. *Al-Balqa Journal for Research and Studies*,**27**, 1-14 (2024).

[32] A.A. Mohammad, M.M. Al-Qasem, S.M. Khodeer, F.M. Aldaihani, A.F. Alserhan, A.A. Haija, S.I.S. Al-Hawary, Effect of Green Branding on Customers Green Consciousness Toward Green Technology. In *Emerging Trends and Innovation in Business and Finance* (pp. 35-48). Singapore: Springer Nature Singapore (2023).

[33] M. Odeh, S.S. Badrakhan, N. Flayyih, M.O. Sabri, Z. Abdijabar, H. Alsabatin, S. Hammad, Quantifying the Impact of the COVID-19 Pandemic on Quality Assurance Practice. *Appl. Math.*,**18**, 989-996 (2024).

[34] A.K. Jain, B.B. Gupta, Towards detection of phishing websites on client-side using machine learning based approach. *Telecommunication Systems*,**68**, 687-700 (2018).
[35] n. Al-Shanableh, M. Al-Zyoud, R.Y. Al-Husban, N. Al-Shdayfat, J.F. Alkhawaldeh, N.S. Alajarmeh, S.I.S. Al-Hawary, Data Mining to Reveal Factors Associated with Quality of life among Jordanian Women with Breast Cancer. *Appl. Math.*,**18**, 403-408 (2024).
[36] A.K. Jain, B.B. Gupta, A machine learning based approach for phishing detection using hyperlinks information. *Journal of Ambient Intelligence and Humanized Computing*,**10**, 2015-2028 (2019).
[37] L. Mobaideen, A. Adaileh, The Impact Of Organizational Culture On Improving Institutional Performance In Aqaba Special Economic Zone Authority In Jordan. *Al-Balqa Journal for Research and Studies*,**27**, 1-21 (2024).
[38] A.S. Al-Adwan, M.M. Al-Debei, The determinants of Gen Z's metaverse adoption decisions in higher education: integrating UTAUT2 with personal innovativeness in IT. *Education and Information Technologies*,**2S**, 7413-7445 (2024).
[39] R.S. Rao, A.R. Pais, Two level filtering mechanism to detect phishing sites using lightweight visual similarity approach. *Journal of Ambient Intelligence and Humanized Computing*,**11**, 3853-3872 (2020).
[40] B. Guo, Y. Zhang, C. Xu, F. Shi, Y. Li, M. Zhang, HinPhish: An effective phishing detection approach based on heterogeneous information networks. *Applied Sciences*,**11**, 9733 (2021).
[41] F. Zheng, Q. Yan, C.M. Victor, F. Leung, Y.U. Richard, Z. Ming, HDP-CNN: High way deep pyramid convolution neural network combining word-level and character-level representations for phishing website detection. *Computers & Security*,**114**, 102584 (2022).
[42] N. Al-shanableh, S. Anagreh, A.A. Haija, M. Alzyoud, M. Azzam, H.M. Maabreh, S.I.S. Al-Hawary, The Adoption of RegTech in Enhancing Tax Compliance: Evidence from Telecommunication Companies in Jordan. In *Business Analytical Capabilities and Artificial Intelligence-enabled Analytics: Applications and Challenges in the Digital Era* (pp. 181-195). Cham: Springer Nature Switzerland (2024).
[43] F.Y. Al-Kasassbeh, S.M. Awaisheh, M.A. Odeibat, S.M. Awaesheh, L. Al-Khalaileh, M. Al-Braizat, Digital Human Rights in Jordanian Legislation and International Agreement. *International Journal of Cyber Criminology*,**18**, 37-57 (2024).
[44] N. Al-Dabbas, The Scope and Procedures of the Expert Recusal in the Arbitration Case: A Fundamental Analytical Study in Accordance with Jordanian Law. *Al-Balqa Journal for Research and Studies*,**27**, 291-306 (2024).
[45] A.M. Vincent, P. Jidesh, An improved hyperparameter optimization framework for AutoML systems using evolutionary algorithms. *Scientific Reports*,**13**, 4737 (2023).
[46] M. Al-Sarem, F. Saeed, Z.G. Al-Mekhlafi, B.A. Mohammed, T. Al-Hadhrami, M.T. Alshammari, T.S. Alshammari, An optimized stacking ensemble model for phishing websites detection. *Electronics*,**10**, 1285 (2021).
[47] R.S. Rao, A. Umarekar, A.R. Pais, Application of word embedding and machine learning in detecting phishing websites. *Telecommunication Systems*,**79**, 33-45 (2022).
[48] A. Aljofey, Q. Jiang, A. Rasool, H. Chen, W. Liu, Q. Qu, Y. Wang, An effective detection approach for phishing websites using URL and HTML features. *Scientific Reports*,**12**, 8842 (2022).
[49] M. Bahaghighat, M. Ghasemi, F. Ozen, A high-accuracy phishing website detection method based on machine learning. *Journal of Information Security and Applications*,**77**, 103553 (2023).
[50] K. Elumalai, D. Bose, Advancement of Phishing Attack Detection Using Machine Learning. *Journal of Electrical Systems*,**20**, 1208-1213 (2024).
[51] M.A. Tamal, M.K. Islam, T. Bhuiyan, A. Sattar, N. Prince, Unveiling suspicious phishing attacks: enhancing detection with an optimal feature vectorization algorithm and supervised machine learning. *Frontiers in Computer Science*,**6**, 1428013 (2024).
[52] E.S. Shombot, G. Dusserre, R. Bestak, N.B. Ahmed, An application for predicting phishing attacks: A case of implementing a support vector machine learning model. *Cyber Security and Applications*,**2**, 100036 (2024).
[53] A. Ejaz, A.N. Mian, S. Manzoor, Life-long phishing attack detection using continual learning. *Scientific Reports*,**13**, 11488 (2023).
[54] A. Almomani, M. Alauthman, M.T. Shatnawi, M. Alweshah, A. Alrosan, W. Alomoush, B.B. Gupta, Phishing website detection with semantic features based on machine learning classifiers: a comparative study. *International Journal on Semantic Web and Information Systems*,**18**, 1-24 (2022).

**M. Ramaiah** received her Ph.D. Degree in Information Technology and Engineering from Vellore Institute of Technology, M.E. in Computer Science from Anna University. She is working as a Professor in the School of Computer Science Engineering and Information Systems at VIT University, Vellore, India. She has attended many national and international conferences and published articles in reputed journals. Her research interest includes cyber-security, Blockchain, Image Processing, Machine Learning, and Artificial Intelligence. Her Orcid ID is: https://orcid.org/0000-0003-3088-6001.

**V. Chandrasekaran** is a Senior Professor at the School of Computer Science Engineering and Information Systems, Vellore Institute of Technology (VIT), Vellore Campus, India. She holds a Ph.D. in Information Technology and Engineering from VIT University, a Master's

degree in Information Technology from Sathyabama University, and a Bachelor's degree in Computer Science from the University of Madras. With 21 years of experience in teaching and research, her expertise spans Image Processing, Deep Learning, Computer Vision, Blockchain, Cyber-Physical Systems, and IoT. She is also an active member of the Computer Society of India and the Soft Computing Research Society. Her Orcid ID is: https://orcid.org/0000-0001-5833-8803.

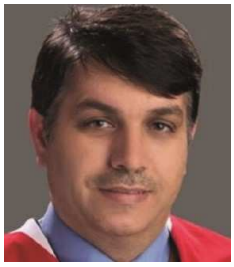
**Vikash Chand** is pursuing an M.Tech in Software Engineering at Vellore Institute of Technology. He is currently working as a cloud engineer intern at Signify. During his course tenure, he has participated in many technical events and presented technical papers at international conferences. His Orcid ID is orcid.org/0007-4835-1036.


**Asokan Vasudevan** is a distinguished academic at INTI International University, Malaysia. He holds multiple degrees, including a PhD in Management from UNITEN, Malaysia, and has held key roles such as Lecturer, Department Chair, and Program Director. His research, published in esteemed journals, focuses on business management, ethics, and leadership. Dr. Vasudevan has received several awards, including the Best Lecturer Award from Infrastructure University Kuala Lumpur and the Teaching Excellence Award from INTI International University. His ORCID ID is orcid.org/0000-0002-9866-4045.

**Suleiman Ibrahim Mohammad** is a Professor of Business Management at Al al-Bayt University, Jordan (currently at Zarqa University, Jordan), with more than 17 years of teaching experience. He has published over 100 research papers in prestigious journals. He holds a PhD in Financial Management and an MCom from Rajasthan University, India, and a Bachelor's in Commerce from Yarmouk University, Jordan. His research interests focus on supply chain management, Marketing, and total quality (TQ). His ORCID ID is orcid.org/0000-0001-6156-9063.

**Eddie Eu Hui Soon** is a Senior Lecturer at INTI International University with over 20 years of experience in academia and the animation industry. Before academia, he worked as a Technical Director in Malaysian production houses, contributing to TV commercials, series, feature films, and corporate videos. He continues to consult in the animation and gaming industry, specializing in 3D cinematic design. His research spans transdisciplinary topics, including Graph Theory, Systems Design, and digital frameworks. Dr. Soon is also involved in prototyping and visualization at the university's fabrication lab and supports research initiatives through journal and website management.

**Qusai Shambour** is affiliated with the Laboratory of Decision Systems and e-Service Intelligence, within the Centre for Quantum Computation and Intelligent Systems at the University of Technology Sydney. He is part of the School of Software in the Faculty of Engineering and Information Technology. His research primarily focuses on recommender systems, collaborative filtering, multi-criteria decision-making, and fuzzy logic. Dr. Shambour explores topics such as recommendation accuracy, semantic similarity, user preferences, and the cold-start problem in recommendation approaches. His work is pivotal in addressing issues like information overload and enhancing the quality of personalized recommendations in online services and social networks. His ORCID ID is orcid.org/0000-0002-3026-845X.



**Muhammad Turki Alshurideh** is a faculty member at the School of Business at the University of Jordan and the College of Business Administration, at the University of Sharjah, UAE. He teaches a variety of Marketing and Business courses to both undergraduate and postgraduate students. With over 170 published papers, his research focuses primarily on Customer Relationship Management (CRM) and customer retention. His ORCID ID is orcid.org/0000-0002-7336-381X.