https://doi.org/10.1007/s42979-023-02084-6

ORIGINAL RESEARCH

A System Review on Fraudulent Website Detection Using Machine Learning Technique

P. Saraswathi[1] · J. V. Anchitaalagammai[2] · R. Kavitha[1]

Received: 8 June 2023 / Accepted: 26 June 2023

Abstract

At present, scams and malicious websites are one of the most widespread and dangerous problems on the website. It brings enormous economic suffering and irretrievable losses to companies and individuals. This approach can strengthen the Internet's legitimacy and impose sanctions on criminals who engage in prohibited or malicious activities. However, governments still need a derivation to classify websites as dangerous or non-dangerous. However, several malicious and counterfeit goods are published on fraudulent websites to cheat consumers and make high and unfair profits. Due to the proliferation of such fraudulent websites, it is difficult to detect and identify them through manual inspection. Phishing attacks include various attacks, including spoofing malicious-based, DNS-based, data theft, email/spam, web-based delivery, and telephone-based phishing. We propose an integrated machine learning (ML) framework for fraudulent website detection to solve this problem. Artificial neural networks (ANN), support vector machine (SVM), random forests (RF), and K-nearest neighbor (K-NN) are algorithms to detect phishing websites accurately. Some URLs can be used to classify them as appropriate or phishing. Data from publicly available phishing websites can be collected from the UCIrvine ML repository for training and testing. Then, the results can be predicted using the features of the dataset. We conduct an in-depth literature review and propose methods for detecting phishing websites using ML methods.

Keywords ML · ANN · SVM · RF · KNN · UCIrvine · Phishing · DNS · URL

Introduction

Nowadays, the current machinery and the ease of communication have made fraudsters and criminals vulnerable to various attacks. It costs billions of dollars worldwide every year. Despite their efforts, several techniques can be realized to detect and investigate fraudulent websites due to the adversative effects of fraudulent websites. However, these methods have limited functionality, and keeping up with the growth and divergence of fraud sites is challenging. Fraudulent websites often masquerade as legitimate online data sources, goods, properties, and facilities [1].

Phishing works as a method to steal sensitive information from users, and phishing sites can be used to lure users away from the site. Attackers use these to gain access to an online service's website and steal sensitive information to earn money and reputation. Phishing works by impersonating website pages and tricking online users into providing confidential information. The term victim phishing comes from "fishing" for complex data. Phishing is one of the most potent and destructive attacks to deceive users. Additionally, sensitive activities such as passwords and credit card material can be used to compromise peculiar information [2, 3].

Figure 1 represents the basic structure of website fraud detection. These include the basic types of phishing techniques and types of attacks. Later, features of the URLs can

This article is part of the topical collection "Advances in Computational Approaches for Image Processing, Wireless Networks, Cloud Applications and Network Security" guest edited by P. Raviraj, Maode Ma and Roopashree H R.

✉ P. Saraswathi
psaraswathimtech@gmail.com

J. V. Anchitaalagammai
jva@vcet.ac.in

R. Kavitha
rka@vcet.ac.in

[1] Department of IT, Velammal College of Engineering and Technology, Madurai, India

[2] Department of Computer Science Engineering, Velammal College of Engineering and Technology, Madurai, India

Published online: 13 September 2023

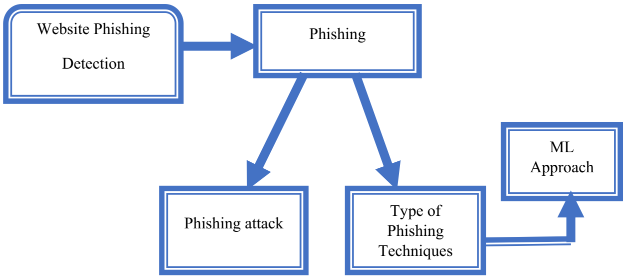

Fig. 1 Basic Architecture of Website Phishing Detection

be used to classify them as legitimate or phishing. Next, the fraudulent website can be detected using the ML framework.

A recent Anti-Phishing Working Group (APWG) report showed that APWG members detected 250,000 phishing attacks between 2015 and 2016 using 195,475 domains. Currently, phishing detection methods are divided into three main groups based on the visual comparison of web pages: blacklisting and whitelisting practices, URL-based systems, and attribute-based web content. But phishing concerns caused by spyware and email scams have led to non-profit industry groups working to combat impersonation and fraud. Phishing is a severe problem due to its widespread disruption to target industries such as payments, financial institutions, and email. However, phishing crimes are estimated to cost the US economy between $61 million and $3 billion in direct economic losses annually [4, 5].

This section proposes an integrated ML framework for fraudulent website detection. ANN, SVM, RF, and K-NN work as algorithms to accurately detect phishing websites. Also, they can be used to classify specific URLs as legitimate or phishing. Data from publicly available phishing websites can be collected from the UC Irvine ML repository for training and testing.

Literature Survey

The author investigates malicious user channel gain feedback falsification (CGFF) attacks using non-orthogonal multiple access (NOMA) approaches to generalize spectral performance and define new malicious threats. However, even a tiny amount of damage to the receiving channel significantly reduces the performance of NOMA, and detecting malicious users is an essential task in NOMA [6]. Long short-term memory (LSTM) method is proposed to obtain information by detecting malware based on attack detection. In addition, two attack models should be considered: correlated and noncorrelated. Often, when an attack occurs, one of the platoon members may use attack patterns to attack the platoon. However, the potential for malicious attacks on the Cooperative Adaptive Cruise Control (CACC) scheme to disrupt driving comfort, traffic flow, and fuel economy benefits is high [7].

A technique can be proposed to guarantee low power consumption and detect malicious attacks in a typical functional provider wake-up radio (WUR) mode. Later, these also defined operational procedures for responding to malicious attacks. However, malicious attacks trick the WUR receiver into accidentally activating it, such as waking up the main radio and putting it into sleep mode [8]. A malicious mobile can design and implement KAYO to differentiate malicious actions from web pages. Using KAYO, multiple iframes up to known invalid phone numbers can be resolved based on static page attributes. However, mobile web pages considerably change from desktop web pages' content, layout, and functionality [9].

The author presents phishing attacks involving stealing user data and downloading and installing malicious software. Similarly, attackers can create phishing emails that appear legitimate users but are challenging to detect. Attackers use social media sites and emails to trick users into sending false information. It takes place as part of a social engineering attack [10]. An approach such as K-nearest neighbor (kNN) and location-based service (LBS) is introduced to crawl all website items through an LBS interface efficiently. Additionally, crawling algorithms can be developed for two-dimensional and high-dimensional spaces. Overhead is defined by theoretical analysis as a function of the algorithm dimension and the number of objects crawled [11].

In the novel approach, various tests can be carried out in the detection mode using classification algorithms to verify the recommended convolutional neural network (CNN) performance and deep neural network (DNN) type. Although many techniques can be presented to detect malicious websites, it is challenging to achieve satisfactory results in a proven manner [12]. The proposed machine learning (ML) algorithms can be used to detect malicious websites and even get personal information to help malicious websites become available websites. These algorithms detect conflicting information hidden in high traffic volumes [13]. MalJPEG is proposed to provide the first ML-based solution tuned to detect unknown malicious JPEG images transparently and efficiently. MalJPEG can exploit this technique to systematically extract recognizable features from the JPEG file system with ten simple methods and identify benign and malicious JPEG images. However, some ideas contain malicious payloads when performing malicious operations [14].

A proposed artificial intelligence (AI)-based meta-learner can be installed on a dataset of phishing websites to define a performance evaluation. The given model can achieve high detection accuracy with a false positive ratio of less than 0.028. However, the consequences of phishing attacks are often dangerous and devastating problems [15]. Phishing detection programs, in particular, can provide software-based programs for systematic review. Reputational datasets, detection capabilities, detection techniques, and indicators can be learned through the taxonomy of phishing detection [16]. The multiobjective evolution/random forest (MOE/RF) approach offers a new phishing attack detection typically based on a revised objective MOE optimization algorithm. The MOE/RF model is designed to accurately detect and reduce phishing sites with a high probability of false positives [17].

A featureless method can be introduced by proposing normalized compression distance (NCD) to detect phishing websites. NCD is used to combine two websites to assess similarity and eliminate the need for feature extraction. A parameter-free similarity measure, in particular, removes dependencies between website feature sets [18]. They proposed that it could be implemented as a browser plug-in to detect phishing websites using a deep learning (DL)-based environment. It can detect in real-time if a user is at risk of phishing while viewing a web page and notify a warning message. However, stolen personal information, legitimate websites, and the rupture of trust in financial institutions are beyond illegal gain [19]. A multilayer stacking ensemble learning method with estimators in different layers is proposed to feed the current layer's estimator predictions to the subsequent layers. The models were sequenced and evaluated using the UCI (D1), Mendeley 2018 (D2), and Mendeley 2020 (D3, D4) datasets. However, phishing uses fake or impersonates legitimate websites to trick online users into revealing delicate data [20].

A novel approach using particle swarm optimization (PSO) is proposed to effectively measure different website features and increase the revealing accuracy of phishing websites. Phishing web detection can be improved by introducing a weighted PSO function for phishing detection [21]. A design-based neuro-fuzzy framework (Fi-NFN) can provide similar resource location and network traffic capabilities in phishing websites. Based on fog computing (FC), a new approach developed an anti-phishing model that Cisco recommends to track and protect. However, fog users from phishing attacks are expensive generic hardware routines that work against different attacks [22]. Overfitting neural networks (OFS-NN) can propose effective phishing website exposure models based on OFS methods and NNs. However, NN models have many useless, low-impact features in the training data set, causing overfitting problems [23].

The novel uses ML and DL techniques to analyze URLs to compare how to detect phishing websites. Most modern solutions that handle phishing detection can offer a canonical class homepage without a login form. Additionally, the base model can be trained on the old dataset and tested with the new URL using datasets from different years [24]. They simplify the feature extraction process by considering URLs and domain names and reduce processing overhead by parsing HTML, DOM, and URL-based features. Among them, 12,134 non-phishing data and 20,614 phishing data could be coded according to 11 predefined attributes [25]. A proposed dataset of 11,000 websites can be combined into a phishing URL-based dataset in vector format pulled from a dataset repository accessible by Phishing and legitimate URL attributes. After pre-processing, multiple ML algorithms block phishing URLs and protect users. Various studies have highlighted research on phishing attack's prevention, detection, and awareness. However, there must be a perfect and adequate solution to the present problem [26].

The proposed two methods can be offered based on generative adversarial networks (GANs). In addition, these methods can integrate phishing and legitimate models to inform real-world websites. Synthetic data can be generated from 10 publicly available phishing datasets used by adversarial autoencoder (AAE) and Wasserstein GAN (WGAN) to obtain information about real-world datasets [27]. Risky implements the domains classifier based on the risky websites (DOCRIW) structure and is based on two essential techniques that help identify domains that contain potentially fraudulent or malicious content. The first statement is that the pre-constructed knowledge base has information on risky websites. The second statement is that the system could be supplemented with a labelable binary classifier to classify a website as malicious or non-malicious based on its domain [28].

The paper claims that ML techniques can be applied to URL patterns, and a new linguistic URL classification approach can be proposed. Additionally, a system based on language processing natural abilities, word vector representations, and n-gram models of black-labelled words used as salient features can be introduced. However, it is ineffective against unknown attacks, most of the episodes are launched from malicious URLs, and attackers trick users into clicking on malicious URLs [29]. A detailed analysis of malicious URL detection techniques and a structural understanding of ML can be detected. Enabling proper malicious URL detection is an ML approach. In addition, literature studies addressing different aspects of the problem can be reviewed to categorize contributions (functional representation, algorithm design, etc.). However, deniers could be better, and detecting newly created malicious URLs is a complex process [30].

Phishing Approach Detection

Phishing can be considered a social engineering technique in this category. Although it looks legitimate, it tricks users into clicking on malicious links that contain malware. Some criminals use this technique to obtain sensitive information, such as credit card numbers and

Table 1 Assessment study of literature

Author name	Previous update	Category of model	Methods	Advantage of method	Disadvantage of method
Zhauniarovich et al. [31]	2018	Malicious detection	DNS (Domain Name System)	A general context for malicious domain detection techniques can be proposed using DNS data. Furthermore, existing approaches can be characterized using several orthogonal perspectives, including DNS data sources and abundance, data analysis methods, and evaluation strategies	However, it requires thoroughly formalizing methods and carefully examining all team strengths and limitations
Joshi et al. [32]	2019	Malicious detection	ML	Malicious URLs in emails can be detected using the ML-based ensemble classification technique. URLs can be extracted from strings using computer types of standard lexical features. Additionally, expectations are high for malicious and non-malicious URLs to differ significantly	Clicking or crawling such URLs enables compromised email accounts, phishing campaigns, and significant financial losses
Kumi et al. [33]	2021	Malicious detection	Classification Based On Association (CBA)	Most malicious attacks aim to gain network access, steal sensitive information, and spy on targeted computer systems. Malicious URLs can be detected using URLs and web content attributes based on the data mining method of association-based CBA classification	However, attackers exploit browser vulnerabilities to install malware and gain remote access to a victim's computer
SagarPatil et al. [34]	2020	Phishing detection	ML	Phishing sites can be detected based on essential characteristics like URL, domain ID, and cryptographic criteria for the ultimate phishing detection rate	However, many websites maliciously ask users to provide sensitive data like usernames, passwords, and credit card details
Cohen et al. [35]	2018	Malicious detection	ML	New static descriptor features extracted from all email elements (headers and parts) can be developed using ML methods to detect malicious emails	Such attacks often cause substantial damage to an organization, including loss or disclosure of confidential or sensitive information

Table 2 Category of Model based on contribution and disadvantage

Author Name	Category of model	Technique	Contributions	Disadvantage
Yang et al. [36]	Phishing detection	DL	They proposed a fast DL-based multidimensional feature phishing detection method. Millions of Phishing and legitimate URLs can be used to test the accuracy of the dataset	Phishing seriously threatens people's online environments, where organizations are trying to protect their privacy
Li et al. [37]	Phishing detection	URL and HTML	Layered models that use URL and HTML features to detect phishing web pages can be provided. Functionally, a summary of insubstantial URL and HTML formatting capabilities and HTML sequence inserting make it potential to build real-time sensing applications without third-party services	As phishing sites continue to operate, more and more people and businesses face issues such as invasion of privacy and loss of funds
Verma et al. [38]	Phishing detection	Neural Network (NN)	The ANN-based approach was developed to capture different types of phishing website data on previously and recently captured datasets	Sophisticated clients use page evasion methods, obfuscation and JavaScript to create complicated interactions between potential victims and phishing sites
Somesha et al. [39]	Phishing detection	ML	The recommended ML method is based on heuristic features for phishing exposure and utilizes 18 factors to complete accuracy	However, social network commerce attacks can steal valuable and private data from email addresses and websites
Ramana et al. [40]	Phishing detection	Random Forest (RF) and Decision Tree (DT)	It is claimed that phishing sites with significant performance can be identified by introducing an intelligent classic with a group of different methods. Different ML algorithms RF and DT ensemble models can be developed to determine the best classifier	However, Phishing is one of the most coordinated attacks. Similarly, it tricks users into retrieving malicious content information

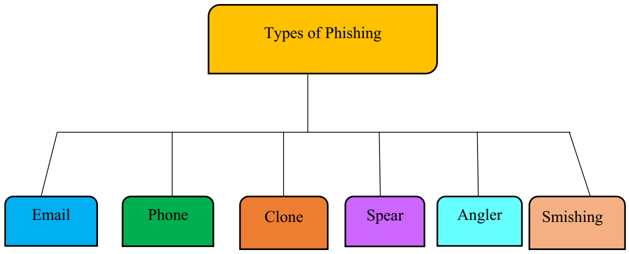

Fig. 2 Architecture types of phishing attack

login credentials. Based on these, phishing techniques and phishing types can be well defined (Tables 1, 2).

A. Types of phishing attacks

Figure 2 shows that phishing attacks mainly aim to trick the target into revealing personal information. However, different types of attacks take place among these. Phishing attacks are one of the main ones to watch out for.

Table 3 shows that spoofing phishing attacks are undesirable to appear more legitimate. For example, attackers can spoof phone numbers or email domains to make them appear more credible.

2 Phishing website detection approaches

This section introduces several anti-phishing methods that can detect and prevent phishing attacks. Phishing attacks are then divided into five groups based on different techniques.

Figure 3 shows that phishing website detection techniques analysis. It is classified as the five phases of approaches in this group.

Table 4 illustrates that phishing website detection techniques can be developed using signatures from phishing websites. Then this approach also helps to distinguish between phishing and simple websites. Further, internet detection techniques are divided into five categories.

Table 5 describes the phishing model for detecting Internet fraud and the data set used to understand the pattern of the data set model. Identification can also be done using a systematic sampling of the dataset.

Website Fraudulent Detection for Machine Learning (ML) Approaches

In this section, we identify several algorithms, SVM, ANN, RF, and K-NN, to accurately detect phishing websites and then describe some of these methods. It can activate URLs and classify them as legitimate or Phishing. A dataset of phishing sites retrieved from the UCI ML repository can be used for training and testing, improving the capabilities of the dataset to predict outcomes.

In this section, we initially collect a dataset from UCI (ML) website fraud detection to investigate the performance of website fraud detection described in Fig. 4. Then, several algorithms for SVM, ANN, RF and K-NN can be used to accurately detect phishing websites using the ML approach and describe some of these methods.

UCI (ML) Dataset

This section initially collects datasets from the UCI ML repository. Phishing is then seen as identity theft when a malicious website impersonates a legitimate website to obtain sensitive information such as passwords, account details, credit card numbers and more. Potential phishing sites can be identified by distinguishing legitimate sites. This dataset can identify critical features of phishing websites and learn website fraud detection by using this dataset to identify ten such features.

Artificial Neural Networks

In this sense, an ANN acts as a series of neurons of interconnected nodes and can be inspired by biological neural networks. Each neuron receives input from subsequent layers and exploits the behavioral transfer function to calculate weights and nonlinear output. Neuron weights can be randomly set at the beginning of training and gradually adjusted using gradient descent to provide an optimal solution. Different layers can be manipulated to change the information they contain. The power of NNs works due to the linear nature of hidden nodes. Thus, introducing nonlinearity into the network is essential to learn functions. Optimal inputs for the classifier can be identified using URL vectors. Then, the primary tasks of forward and backward propagation of the classifier can be handled.

a. Forward propagation

The phishing dataset can be fed into ANN and selected with the best features. The input data set can be allocated into training and test sets. Among them, the training set of phishing websites can be implemented, and the optimal structure of ANN can be obtained. Choose a test suite and see the overall performance of the phishing website detection model.

The activation function detects contributions from the input layer. It computes the hidden layer neuron unit output matrix, the number of hidden layer neurons with randomly created weights and offsets. Let's assume y is the detecting

Table 3 Analysis of the types of phishing attack

S. no	Type of phishing attack	Contribution of phishing
1	Vishing	Vishing campaigns include malicious activities such as phone calls, voicemails, and collecting and using people's data for financial gain
2	Angler	Angler phishing is done through social media, tricking consumers into stealing data posted on the site or revealing personal information
3	Spear	Sophisticated phishing attacks send malicious emails to specific targets
4	Clone	These types of attacks are whaling attacks that target corporate executives
5	Phone	Malicious attacks can also be carried out through phishing phones
6	Email	Attackers send emails to launch attacks to carry out online attacks
7	Smishing	Phishing is a scam that uses text messages

result, G is hidden layer, z is weight, m is no of input points, d is activation function, and a is output level of the matrix.

$$G_{sum} = \begin{bmatrix} d(z_1 * y_1 + b_1) \ldots & d(z_1 * y_1 + a_G) \\ \vdots & \vdots \\ d(z_1 * y_m + b_1) \ldots & d(z_1 * y_m + a_G) \end{bmatrix} \quad (1)$$

The output layer of an ANN is referred to as the randomly generated weight vector. The number of neuron units calculated by an activation function is the entire quantity of input data points, where P is output, i is neuron units in the hidden layer, weight, m is no of input points, f is activation function, a_i is randomly no of neuron units, and β_i is connecting the importance of the hidden layer.

$$P_i = \sum_{i=1}^{G} \beta_i (z_i * y_i + a_i) i = 1, 2, 3, \ldots .. m \quad (2)$$

b. Backward propagation

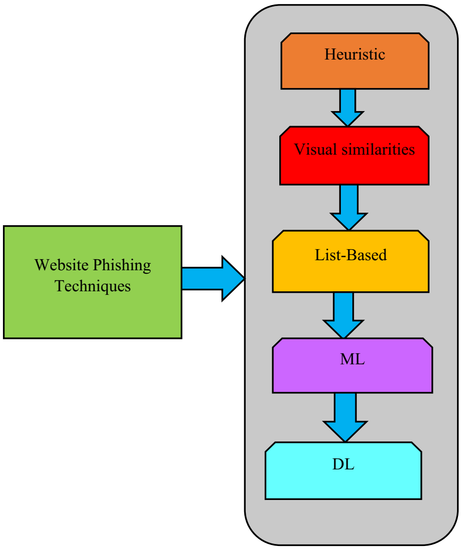

Fig. 3 Analysis of the website phishing detection techniques

The ANN outputs the identified data arguments in the training set in this section. If the ANN makes the correct prediction for that data point, the ANN will remain unchanged. Then calculate the mean square error of the data points of the ANN.

Equation 3 is the neural network outputs that can be calculated with the data points. Let's assume the $\hat{c}_i^l$ is output neural network, i is hidden layer, d is function, and l is data points.

$$\hat{c}_i^l = d(\beta_i - a_i) \quad (3)$$

Equation 4 is the mean square error of the data argument. Let's assume, F is error and l is data points.

$$F_l = \frac{1}{2} \sum_{i=1}^{1} (\hat{c}_i^l - \hat{c}_i^l)^2 \quad (4)$$

Equations 5 and 6 update the data points' weights for accurate predictions, where i is hidden layer, z is weight vector, γ is rate, and $c_1 - \hat{c}_l$ is ANN that does not change.

$$z_i \leftarrow z_i + \Delta z_i \quad (5)$$

$$\Delta z_i = \gamma (c_1 - \hat{c}_l) \quad (6)$$

In this category, the primary functions of forward and backward propagation of the classifier can be handled. However, when calculating the activation function, we identify the output matrix of a unit as the contributions in the input layer and the neuron in the hidden layer. The output layer of an ANN is referred to as the randomly generated weight vector, and the data points for accurate prediction are updated.

Support Vector Machine (SVM)

In this segment, linear and nonlinear data can be classified using SVM. Additionally, given the original training data, a nonlinear graph can transform the algorithm into higher dimensions. In this dimension, optimal linear hyperplanes are used to separate any two types of data, and then, SVMs can be

Table 4 Primary conduct Phishing detection technique approach

S. no	Phishing approach	Algorithm techniques usages	Dataset accuracy	Most important findings
1	Heuristic	RF, Multilayer Perceptron	UCI ML Repository	The system detects phishing emails and spam with 97.7% and 89.2% accuracy
2	Visual similarity	DT, KNN	UCI ML Repository Mendeley 21,055 occurrences	This tentative study achieved an accuracy of 97.51% using a dataset from Mendeley's Phishing dataset for UCI and ML
3	List –based	The DNS Blacklist	PhishTank, Google	Detects phishing and zero-day phishing attacks with an accuracy of 98.90%
4	ML	SVM	Standard crawl accomplish (5000 URL)	This method can detect Phishing and legitimate websites with 95.66% accuracy
5	DL	RNN	Phish Tank and Phish Storm ISCX-URL-2016222,541 URL	The sensitivity is 3.98%, UP from the previous work

Table 5 Performance comparison with Phishing dataset

S. no	Author name	Model	Methods	Data set description
1	Zhu et al. [41]	Phishing detection	Optimal Feature Selection—Neural Network (OFS-NN)	With the capability to acquire intensively from large datasets, NNs are essential heuristic ML techniques in phishing website detection and blocking
2	Liu et al. [42]	Phishing detection	Multistage Phishing Detection (MSPD)	One is a qualified test of CASE's altered features and detection models, which contain traditional ML and DL algorithms based on the construction of complex data sets
3	Zhang et al. [43]	Phishing detection	Client-Side Cloaking (CSC)	The dataset was collected from 112,005 phishing websites over the 14 months from 2018 to 2019. And can run CrawlPhish to analyze them entirely
4	Hazim Alkawazet al. [44]	Phishing detection	ML	Kaggle's dataset includes 86 features and 11,430 complete URLs, half Phishing and half legitimate
5	Aljofey et al. [45]	Phishing detection	Convolutional neural network (CNN)	Experiments indicate that the proposed framework performs better than phishing URL models on benchmark datasets

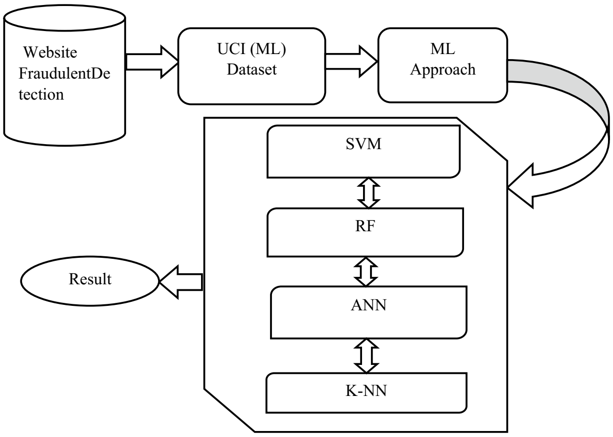

Fig. 4 An Overview of Proposed Model in Website Fraudulent Detection

used for classification and numerical prediction. A simple form of SVM is a complex binary classifier in which the classes are linearly separated. Also, the data can be transformed into higher dimensions using an appropriate kernel function to implement a linear discriminant process. Segmentation is not possible using kernels, and the goal is to reduce the error rate of SVM.

Equation 7 is a parameter consisting of input vectors of input features that can be calculated to determine the size and Model’s bias-variance trade-off. Q is minimum, z is weight vector, e is parameter, j is class, a is scalar quantity,ξ_j is positive slack variable, and m is no of the input vector.

$$\underset{z,a,\xi}{Q}\frac{1}{2}\parallel z\parallel^2+e\sum_{j=1}^{m}\xi_j \tag{7}$$

In this Eq. 8, derive the Lagrangian equation in its dual problem and calculate using Karush–Kuhn–Tucker conditions by substituting values, where o is maximum, α is Lagrangian vector, b and c is class, l is kernel function, d_b, d_j is no of feature mapping, and d is higher dimension.

$$ {}^{o}_{\alpha}z\,(\alpha) = \sum_{\mathrm{b}=1}^{\mathrm{m}} \alpha_{\mathrm{i}} - \frac{1}{2}\sum_{\mathrm{b}=1}^{\mathrm{m}}\sum_{\mathrm{c}=1}^{\mathrm{m}} \mathrm{y}_{\mathrm{b}}\mathrm{y}_{\mathrm{c}}\alpha_{\mathrm{b}}\alpha_{\mathrm{c}}\,\mathrm{l}\left(d_{\mathrm{b}}, d_{\mathrm{j}}\right) \tag{8} $$

In Eq. 9, calculate the equivalently expressed dual in vectors. Let us assume the S is function, f is equivalently double vector, v represents the dual form vector, and Q is min.

$$ {}^{Q}_{\alpha}\frac{1}{2}\,\alpha^{\mathrm{S}}\mathrm{v}_{\alpha} - \mathrm{f}^{\mathrm{S}}_{\alpha} \tag{9} $$

This indicates that partitioning is not possible in the kernel. This includes trade-offs between input feature vector size and model-dependent variance. These values can then be interpolated to achieve a vector form of equivalent representation in dual vectors for later calculation.

Random Forest (RF)

In this section, packing can be combined with random attribute selection to generate RF. These are simple decision trees adding inputs or checks at the top and ending with smaller subsets of the tree. RF follows an ensemble learning approach and can use strategies derived from these to improve performance. A clustering mechanism combines random subsets of different trees into RF. The accuracy of RF depends on the degree of dependence between classifiers and the strength of individual classifiers. RF does not need cross-validation or a separate test set to acquire a fair estimate of test set error.

Calculate the standard error in RF accuracy using Eq. 10. Let's assume the J is test set, E is error, a and b is the average number of votes, x is predictor vector, y is classification, and mh is margin function.

$$ \mathrm{JE} *= \mathrm{P}_{\mathrm{AB}}(\mathrm{mh(AB)}) < 0 \tag{10} $$

In Eq. 11, the margin function measures how much the average vote for the appropriate class is greater than the average calculated for the other categories, where Q is max, avk is average value, h is tree structure classifier, and k is sensitive parameter,

$$ mh(A,B) = avkI\left(h_k(A) = Y\right) - Q_j \neq b \tag{11} $$

In Eq. 12, the expected value of the edge function gives the RF intensity calculation. Let's assume, R is strength and E is error.

$$ R = E_{A,B}(\mathrm{mh(AB)}) \tag{12} $$

In Eq. 13, calculate the generalization error of the constrained ensemble classifier as a function of the average correlation between the base classifiers and their average strength, where JE is test set error and *the*ρ is mean value of correlation.

$$ JE^* \leq \rho\left(1 - R^2\right)/R^2 \tag{13} $$

In this section, a reasonable estimate of the test set error can be obtained using the degree of inter-classifier dependence and the strength of individual classifiers to determine the accuracy of the RF. Furthermore, ensemble classification can be constrained as a function of the average correlation between the base classifiers and their average strength to account for the generalization error.

K-nearest Neighbor (K-NN)

In this category, distance-based contrasts assign equal weight to each attribute, which can lead to noise or irrelevant data errors. However, editing and pruning techniques can be used to solve the problems of wasted data tuples and noisy data tuples, respectively. Each tuple can experimentally determine the optimal number of neighbors for a point in n-dimensional space.

Minkowski, Manhattan, and Euclidean distance functions can be used in slow classifiers because the entire training dataset must be optimized for classification. Three mathematical expressions of the algorithm can be found under Eqs. 14, 15 and 16. Let's assume k-nearest neighbor, u is value, and c_{u} and d_{u} is attribute variable.

$$ \sqrt{\sum_{\mathrm{u}=1}^{\mathrm{K}} \left(c_{\mathrm{u}} - d_{\mathrm{u}}\right)^2} \tag{14} $$

$$ \sqrt{\sum_{\mathrm{u}=1}^{\mathrm{K}} \left|c_{\mathrm{u}} - d_{\mathrm{u}}\right|^2} \tag{15} $$

$$ \sqrt{\left[\sum_{\mathrm{u}=1}^{\mathrm{K}} (\left|c_{\mathrm{u}} - d_u\right|)^2\right]} \tag{16} $$

In this section, distance functions can be used in slow classifiers since the entire training dataset can be implemented in three mathematical expressions of the algorithm to be optimized for classification.

Result and Discussion

This section evaluates the model's performance for detecting phishing website datasets published from the UCI ML repository. The ML methods techniques can test and define

Table 6 Evaluation of matrix

Parameters	Formula
Sensitivity	$\frac{TP}{TP+FN}$
Specificity	$\frac{TN}{TN+FP}$
Accuracy	$\frac{TP+TN}{TP+FN+TN+FP} * 100\%$
Precision	$\frac{TP}{TP+FP}$
Recall	$\frac{TP}{TP+FN}$
F-Measure	$2 * \frac{Precision*Recall}{Precision+Recall}$

each method's precision, F-measure, sensitivity, specificity, precision, and recall. Different data processing techniques can be used for tenfold cross-validation classification; TP, FP, TN and FN each have multiple bits.

Table 6 shows that various ML techniques, namely ANN, k-NN, SVM and RF, are used as classifiers for phishing detection, and the results are presented. Each method can be tested and defined using its TP, FP, TN and FN values.

Table 7 demonstrates that confusion matrices are used to estimate the efficacy of ML methods for detecting website fraud. Valid and predicted values and contributions can be compared with a defined confusion matrix with phishing detection in percentage.

Sensitivity

In this category, Fig. 5 shows that ANN and SVM techniques have lower accuracy when compared to sensitivity analysis. Comparing these two approaches, the RF method achieves a higher accuracy of 79%.

Specificity

Figure 6 illustrates that, compared to specificity analysis, ANN and SVM methods are 65% and 69% less accurate, respectively. The RF method achieves a higher accuracy of 81% compared to the two performance methods in specificity.

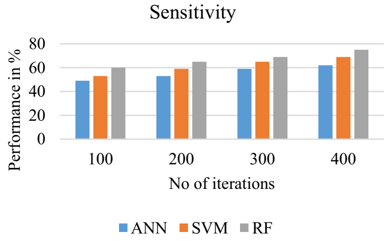

Fig. 5 Analysis of Performance in Sensitivity

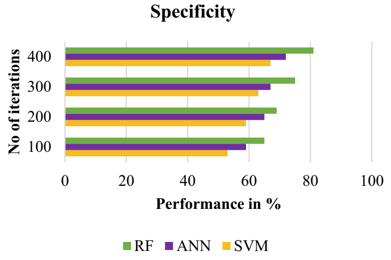

Fig. 6 Analysis of Performance in Specificity

Accuracy

In this section, Fig. 7 demonstrates that the RF method achieves 93% higher accuracy compared to the two performance methods in terms of accuracy. Compared to the precision methods, the ANN and SVM analysis methods obtained 71% and 81% lower accuracy, respectively.

In this category, in the precision and recall model shown in Table 8, comparing the two methods such as PSO and CBA, their accuracy has risen to 92 and 95.8%, and their number has reached the highest accuracy of 98.99% when dealing with another URL model F-measure method.

Conclusion

In this section, the phishing techniques behind the classification work to automatically classify fraudulent website detection into predefined class values based on certain features and class variables. Phishing sites can be detected by

Table 7 A performance evaluation of phishing website detection

Real value	Forecast value	Contribution
Positive (Phishing website)	TP	Some correctly recognized phishing websites among phishing sites
Negative (Legal Website)	FP	Wrongly detected as a legal site
Positive (Phishing website)	FN	Several websites can be misdiagnosed as phishing sites
Negative (Legal website)	TN	A legitimate website has been correctly identified as fair

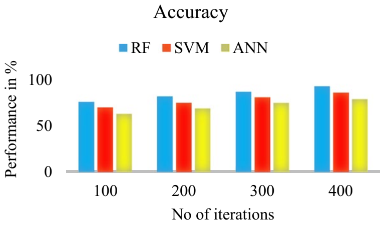

Fig. 7 Analysis of Accuracy Performance

Table 8 Comparison of precision, recall, and F-measure model

Authors	Method	Model	Analysis of accurateness
Ali et al. [21]	PSO	Precision	92%
Kumi et al. [33]	CBA	Recall	95.8%
Yang et al. [36]	URL	F-Measure	98.99%

relying on ML-based phishing techniques to gather information to help organize websites. Nevertheless, the damage can be mitigated by developing embattled anti-phishing programs and technologies and refining the public on spotting and identifying fraudulent phishing websites. Also, they include precision and F1 measures, sensitivity, specificity, accuracy, and recall that can be improved using algorithms. In this regard, their assessment achieved 91% accuracy in sensitivity and specificity. The precision and recall models outperformed PSO and CBA at 92% and 95.8% accuracy. This number was higher at 98.99% when dealing with the F-Measurement method for another URL model. In addition, research can be extended to generate more expansive network results and protect individual privacy.

Funding No funding received for this research.

Declarations

Conflict of interest No conflict of interest.

References

1. Maktabar M, Zainal A, Maarof M, Kassim M. Content-based fraudulent website detection using supervised machine learning techniques. Adv Intell Syst Comput. 2018;734:294–304. https://doi.org/10.1007/978-3-319-76351-4_30.
2. Yang R, Zheng K, Wu B, Wu C, Wang X. Phishing website detection based on deep convolutional neural network and random forest ensemble learning. Sensors (Basel). 2021;21(24):8281. https://doi.org/10.3390/s21248281.PMID:34960375;PMCID:PMC8709380.
3. Rao RS, Pais AR. Detection of phishing websites using an efficient feature-based machine learning framework. Neural Comput Appl. 2019;31:3851–73. https://doi.org/10.1007/s00521-017-3305-0.
4. Aljofey A, Jiang Q, Qu Q, Huang M, Niyigena JP. An effective phishing detection model based on character level convolutional neural network from URL. Electronics. 2020;9:1514. https://doi.org/10.3390/electronics9091514.
5. Karthick K, et al. Iterative dichotomiser posteriori method-based service attack detection in cloud computing. Comput Syst Sci Eng. 2023;44(2):1099–107.
6. Xia S, Tao X, Li N, Wang S. Malicious user detection in non-orthogonal multiple access based on spectrum analysis. IEEE Signal Process Lett. 2020;27:1390–4. https://doi.org/10.1109/LSP.2020.3012826.
7. Ko B, Son SH. An approach to detecting malicious information attacks for platoon safety. IEEE Access. 2021;9:101289–99. https://doi.org/10.1109/ACCESS.2021.3095480.
8. Park H. Anti-malicious attack algorithm for low-power wake-up radio protocol. IEEE Access. 2020;8:127581–92. https://doi.org/10.1109/ACCESS.2020.3008431.
9. Amrutkar C, Kim YS, Traynor P. Detecting mobile malicious webpages in real time. IEEE Trans Mobile Comput. 2017;16(8):2184–97. https://doi.org/10.1109/TMC.2016.2575828.
10. Asiri S, Xiao Y, Alzahrani S, Li S, Li T. A survey of intelligent detection designs of HTML URL phishing attacks. IEEE Access. 2023;11:6421–43. https://doi.org/10.1109/ACCESS.2023.3237798.
11. Yan H, Gong Z, Zhang N, Huang T, Zhong H, Wei J. Crawling hidden objects with kNN queries. IEEE Trans Knowl Data Eng. 2016;28(4):912–24. https://doi.org/10.1109/TKDE.2015.2502947.
12 Liu D, Lee JH. CNN based malicious website detection by invalidating multiple web spams. IEEE Access. 2020;8:97258–66. https://doi.org/10.1109/ACCESS.2020.2995157.
13. Dhiyanesh B, Rameshkumar M, Karthick K, Radha R. Cloud computing and machine learning for analysis of health care data based on neuro fuzzy logistic regression. J Intell Fuzzy Syst: Appl Eng Technol. 2023;44(6):9955–64. https://doi.org/10.3233/JIFS-223280.
14. Cohen A, Nissim N, Elovici Y. MalJPEG: machine learning based solution for the detection of malicious JPEG images. IEEE Access. 2020;8:19997–20011. https://doi.org/10.1109/ACCESS.2020.2969022.
15. Alsariera YA, Adeyemo VE, Balogun AO, Alazzawi AK. AI meta-learners and extra-trees algorithm for the detection of phishing websites. IEEE Access. 2020;8:142532–42. https://doi.org/10.1109/ACCESS.2020.3013699.
16 Dou Z, Khalil I, Khreishah A, Al-Fuqaha A, Guizani M. Systematization of knowledge (SoK): a systematic review of software-based web phishing detection. IEEE Commun Surv Tutor. 2017;19(4):2797–819. https://doi.org/10.1109/COMST.2017.2752087.
17 Zhu E, Chen Z, Cui J, Zhong H. MOE/RF: a novel phishing detection model based on revised multiobjective evolution optimization algorithm and random forest. IEEE Trans Netw Serv Manage. 2022;19(4):4461–78. https://doi.org/10.1109/TNSM.2022.3162885.
18. Purwanto RW, Pal A, Blair A, Jha S. PhishSim: aiding phishing website detection with a feature-free tool. IEEE Trans Inf Forensics Secur. 2022;17:1497–512. https://doi.org/10.1109/TIFS.2022.3164212.
19. Tang L, Mahmoud QH. A deep learning-based framework for phishing website detection. IEEE Access. 2022;10:1509–21. https://doi.org/10.1109/ACCESS.2021.3137636.

20. Kalabarige LR, Rao RS, Abraham A, Gabralla LA. Multilayer-Multilayer stacked ensemble learning model to detect phishing websites. IEEE Access. 2022;10:79543–52. https://doi.org/10.1109/ACCESS.2022.3194672.
21. Ali W, Malebary S. Particle swarm optimization-based feature weighting for improving intelligent phishing website detection. IEEE Access. 2020;8:116766–80. https://doi.org/10.1109/ACCESS.2020.3003569.
22. Pham C, Nguyen LAT, Tran NH, Huh E-N, Hong CS. Phishing-aware: a neuro-fuzzy approach for anti-phishing on fog networks. IEEE Trans Netw Serv Manage. 2018;15(3):1076–89. https://doi.org/10.1109/TNSM.2018.2831197.
23. Zhu E, Chen Y, Ye C, Li X, Liu F. OFS-NN: an effective phishing websites detection model based on optimal feature selection and neural network. IEEE Access. 2019;7:73271–84. https://doi.org/10.1109/ACCESS.2019.2920655.
24. Sánchez-Paniagua M, Fernández EF, Alegre E, Al-Nabki W, González-Castro V. Phishing URL detection: a real-case scenario through login URLs. IEEE Access. 2022;10:42949–60. https://doi.org/10.1109/ACCESS.2022.3168681.
25. Sakthivel S. UBP-trust: user behavioral pattern based secure trust model for mitigating denial of service attacks in software as a service (SaaS) cloud environment. J Comput Theor Nanosci. 2016;13(10):7649–54.
26. Karim A, Shahroz M, Mustofa K, Belhaouari SB, Joga SRK. Phishing detection system through hybrid machine learning based on URL. IEEE Access. 2023;11:36805–22. https://doi.org/10.1109/ACCESS.2023.3252366.
27 Shirazi H, Muramudalige SR, Ray I, Jayasumana AP, Wang H. Adversarial autoencoder data synthesis for enhancing machine learning-based phishing detection algorithms. IEEE Trans Serv Comput. 2023. https://doi.org/10.1109/TSC.2023.3234806.
28. Prieto A, Fernández-Isabel A, De Diego IM, Ortega F, Moguerza JM. Knowledge-based approach to detect potentially risky websites. IEEE Access. 2021;9:11633–43. https://doi.org/10.1109/ACCESS.2021.3051374.
29. Hai QT, Hwang SO. Detection of malicious URLs based on word vector representation and ngram. J Intell Fuzzy Syst. 2018;35(6):5889–900. https://doi.org/10.3233/jifs169831.
30. Sahoo D, Liu C, Hoi S. Malicious URL detection using machine learning: a survey. Arxiv.org, 2021. [Online]. Available.
31 Zhauniarovich Y, Khalil I, Yu T, Dacier M. A survey on malicious domains detection through DNS data analysis. ACM Comput Surv. 2018;51(4):1–36. https://doi.org/10.1145/3191329.
32. Joshi A, Lloyd L, Paul Westin P, Seethapathy S. Using lexical features for malicious URL detection - A machine learning approach. Think India J. 2019;22(11). arXiv:1910.06277
33. Kumi S, Lim C, Lee S-G. Malicious URL detection based on associative classification. Entropy. 2021;23(2):182. https://doi.org/10.3390/e23020182.
34. Patil S, Shetye Y, Shendage N. Detecting phishing websites using machine learning. Think India J. 2019;22(11).
35. Cohen A, Nissim N, Elovici Y. Novel set of general descriptive features for enhanced detection of malicious emails using machine learning methods. Expert Syst Appl. 2018;110:143–69.
36. Yang P, Zhao G, Zeng P. Phishing website detection based on multidimensional features driven by deep learning. IEEE Access. 2019;7:15196–209.
37. Li Y, Yang Z, Chen X, Yuan H, Liu W. A stacking model using URL and HTML features for phishing Webpage detection. Future Gener Comput Syst. 2019;94:27–39.
38. Verma MK, Yadav S, Goyal BK, Prasad BR, Agarwal S. Phishing website detection using neural network and deep belief network. In: Recent findings in intelligent computing techniques. Singapore: Springer; 2019. p. 293–300.
39. Somesha M, Pais AR, Rao RS, Rathour VS. Efficient deep learning techniques for the detection of phishing websites. Sādhanā. 2020;45(1):1–18.
40. Ramana AV, Rao KL, Rao RS. Stop-phish: an intelligent phishing detection method using feature selection ensemble. Soc Netw Anal Mining. 2021;11(1):1–9.
41. Zhu E, Ye C, Liu D, Liu F, Wang F, Li X. An effective neural network phishing detection model based on optimal feature selection. Proc 16th IEEE IntSymp Parallel Distrib Process Appl (ISPA). 2018;pp. 781–787
42. Liu D-J, Geng G-G, Jin X-B, Wang W. An efficient multistage phishing website detection model based on the CASE feature framework: aiming at the real web environment. Comput Secur. 2021;110.
43. Zhang P, Oest A, Cho H, Sun Z, Johnson R, Wardman B et al. CrawlPhish: large-scale analysis of client-side cloaking techniques in phishing. Proc IEEE Symp Secure Privacy (SP). 2021;pp. 1109–1124.
44. HazimAlkawaz M, Steven SJ, Hajamydeen AI. Detecting phishing website using machine learning. 16th IEEE International Colloquium on Signal Processing its Applications (CSPA 2020). 2020;pp. 28–29
45. Aljofey A, Jiang Q, Qu Q, Huang M, Niyigena JP. An effective phishing detection model based on the character level convolutional neural network from URL. Electronics. 2020;9(9):1514.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.