DOI:https://doi.org/10.52756/ijerr.2023.v36.009 *Int. J. Exp. Res. Rev.*, Vol. 36: 99-108(2023)

Original Article | Peer Reviewed | Open Access

International Journal of Experimental Research and Review (IJERR)
© Copyright by International Academic Publishing House (IAPH)
ISSN: 2455-4855 (Online)
www.iaph.in

Phishing Detection: A Hybrid Model with Feature Selection and Machine Learning Techniques

Rekha Pal[1], Mithilesh Kumar Pandey[1], Saurabh Pal[1*] and Dyhan Chandra Yadav[2]

[1]Department of Computer Applications, VBS Purvanchal University, Jaunpur, Uttar Pradesh, India;
[2]Department of Computer Science, Maharshi University, Lucknow, India

E-mail/Orcid Id:

RP, peehupal08@gmail.com;*MKP*, mithileshkumarmca@gmail.com ;*SP*, drsaurabhpal@yahoo.co.in, https://orcid.org/0000-0001-9545-7481;
DC, dc9532105114@gmail.com, https://orcid.org/0000-0003-0084-0360

Article History:
Received: 14th March, 2023
Accepted:09th Dec., 2023
Published: 30thDec., 2023

Keywords:
Principal Component Analysis, Logistic Regression, Random Forest, Logit LR model, Pearson Correlation

How to cite this Article:
Rekha Pal, Mithilesh Kumar Pandey, Saurabh Pal, Dhyan Chandra Yadav(2023). Phishing Detection: A Hybrid Model with Feature Selection and Machine Learning Techniques. *International Journal of Experimental Research and Review*, *36*, 99-108.
DOI:https://doi.org/10.52756/ijerr.2023.v36.009

Abstract: Various phishing problems increase in cyber space with the progress of information technology. One of the prominent cyber-attacks rooted in social engineering is known as phishing. This malicious activity aims to deceive individuals into divulging sensitive information, including credit card details, login credentials, and passwords. The main importance of this research is finding the best outcome by various machine learning (ML) techniques. This paper uses a Tree Classifier (ETC), Forward Selection, Pearson correlation, Logit-LR model and Principal_Component_Analysis for feature selection. The Logistic_regression (LR), Naïve_Bayes (NB), Decision_Tree (DT), K-Nearest Neighbor (K-NN), Support_Vector_Machine (SVM), Random_Forest (RF), AdaBoost and Bagging classifiers are used for developing the phishing detection model. We have studied the model in four cases. Case 1 has 6 commonly selected features by ET, forward selection and Pearson's correlation, case 2 has 25 features by logit model, case 3 has all features, and case 4 has principal component analysis (3 and 5 components). We find the highest accuracy of 97.3% in case 2 with the random forest model.

Introduction

Phishing continues to increase due to the increasing digital system. Phishing crimes primarily use social design and innovative misdirection to find out customer protection data (Jamil et al., 2018). The customer then cheats the customer by entering private data without confirmation. So far, phishing attacks have appeared on the PC and general stage as frequently as possible. With an eye toward reducing the risk of phishing attacks, some procedures have been proposed to plan and teach end clients to view and differentiate phishing URLs. However, they still focus on client practices and information on using the basic framework. Product-based programmatic strategies are generally used to differentiate phishing attacks due to their high accuracy and effectiveness (Zhu et al., 2019 ; Jain and Gupta, 2016; Sharfuddin et al., 2023). The benefit of this technique is that very few assets are required on the basic framework since there is little responsibility for dissecting the site's content. Nonetheless, this technique has difficulties managing recent phishing attacks because the repository that holds highly contrastive records is built from recently identified URLs. Heuristic phishing localization methods can be used to enhance high-contrast recordings (Babagoli et al., 2019; Chaurasia and Pal, 2021). Then, at that point, because of the separated elements, the fundamental AI classifiers are prepared to recognize the phishing sites. Classifiers are usually built from LR, SVM models, NB models, etc. Using AI methods, phishing sites can be effectively identified (Chaurasia et al., 2022). In the meantime, it can also accommodate recent phishing sites. The way to implement this technique is to obtain highly qualified elements from phishing URLs and their associated sites. Still, unwisely identifying sensitive elements will make it impossible for basic classifiers to identify phishing sites with certainty. At the same time, some useless or ineffective highlights will cause AI technology to fall into

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.

the problem of overfitting (Cawley and Talbot, 2010; Dawn et al., 2023).

This paper proposes a hybrid model incorporating selection and artificial intelligence, a powerful phishing attack detection model. This model selects highlights by utilizing feature selection to compute ET, forward selection, Pearson correlation, and a measurable logit model to evaluate the impact of fine elements on phishing site identification. Then, considering these partial guarantees that prominent features are derived, the plan estimates to pick ideal components from various URLs. Finally, we trained the selected LR, NB, DT, KNN, SVM, RF, AdaBoost and Bagging algorithms by important features to find phishing attacks. Taking everything into account, the commitments of this paper are recorded as follows-

(1) Model in view of 3-feature selection methods. To more likely evaluate the impact of subtle elements of selection on identifying phishing attacks, this paper proposes a 3-highlight selection model as a synthetic source. The addition of positive and negative highlights of the URL characterizes common elements. By evaluating 3 - including selection (ET, forward selection and Perason correlation), some useless or insignificant highlights can be disposed of for the presentation of the entire model.

(2) Model based on logit (Logistic Regression) feature selection algorithm. The calculation determines from the outset the strengths of all the highlights of the information URL and its associated sites. Then, at this point, set an edge (p-value < 0.05) to select fine elements to develop the ideal element-wise vector. With this calculation, many useless and insignificant highlights are pruned away. Since these repeated highlights are not exacerbated, the overfitting problem of the base classifier is reduced. ML classifiers outperform many existing frameworks in phishing site discovery.

(3) Analysis based on all features. Through the ML classifiers, the phishing dataset was evaluated with full features to compare with other models. The comparison with other models shows the importance of features in various models.

(4) Model based on Principal Component Analysis (3 and 5 component). Through a selection of sensitive highlights and numerous experimental investigations, the ideal design of the ML classifier is prepared and constructed as the model's final classifier.

In addition to a large number of works, SMOTE (Synthetic Minority Oversampling Technique) is used to similarly propagate information between classes, and the problem of representation is likely to be the main problem to solve to prevent misclassification due to heavily skewed classes. Destroyed is an oversampling strategy to increase the minority class in the dataset (Guan et al., 2021).

The rest of this paper organized into various sections as: Related work is represented in section 2. Features selection is represented in section 3. Training ideas are generated in sections 4 and section 5, and 6, representing discussion and conclusion.

Related Work

Although there are procedures for preparing and presenting to the end customer, the programming-based custom zone approach is the most notable technique for addressing the risk of phishing attacks. Mohammed et al. proposed an intelligent self-coordinated neural association for identifying phishing regions (Mohammad et al., 2014). They showed 17 components of 600 real and 800 phishing destinations accumulated documents, isolated with the help of outcasts. Their tests demonstrate neural tissue's high generalization and power in phishing identification (Mohammad et al., 2013). 18 components are demonstrated for 859 valid and 969 phishing locales, respectively. Considering the whitelist, Kang and Lee (2007) proposed a system to distinguish phishing destinations. This method determines the client's authority over the site by distinguishing between URL proximity. Jain and Gupta (2018) proposed an AI-based strategy to identify phishing sites using only client-side elements. Towards detecting phishing websites on the client using a machine learning approach. They removed some web-based sections, and banking locales evaluated their methods. Sharifi and Siadati (2008) proposed a counter-generator strategy to identify phishing sites. This strategy determines if a site is phishing by matching its zone name with Google's listing. While the abovementioned checks recommend various elements to identify phishing sites, some may not fully characterize phishing incidents (Rajab, 2018). Essentially, Babagoli et al.use a comparable informational index and propose include determination utilizing choice trees and the covering strategy, which brings about choosing 20 highlights (Han et al., 2016 ; Babagoli et al., 2019). They assess the phishing location execution utilizing a novel meta-heuristic-based nonlinear relapse calculation. All things considered, the component determination strategies proposed by these checks are informed by information and require client-indicated edge values. Lee et al. (2014) proposed the PhishTrack structure for subsequent recovery of phishing against blacklists. High-contrast recording consumes very few assets on the base

DOI:https://doi.org/10.52756/ijerr.2023.v36.009

frame. Nonetheless, it cannot schedule recent phishing attacks as expected (Aleroud and Zhou, 2017). Khonji et al. (2013) explored a few element determination strategies for distinguishing email phishing, including Relief and relationship-based component choice. Attempted computations included RF, SVM and DT strategies; irregular out-of-the-way acquisition techniques beat the others. AI technology is widely examined to identify phishing sites due to its dynamic learning capabilities. Zhang et al. (2007) is a phishing attack discovery model based on 27 sensitive removed from URLs. The model uses TF-IDF calculations to identify phishing attacks. The research computation can identify many kinds of phishing attacks, but it is not conducive to the large time cost of consuming hidden frameworks. Some legitimate websites were considered phishing during this period (Li et al., 2016). Basnet et al. (2012) tried two notable element selection strategies: overlay and CFS. Insatiable forward guarantees and regular computations are used to evaluate components removed from web pages and web crawlers. To evaluate the feasibility of the component selection technique, the creators used three AI computations, specifically, LR, RF, and NB. The outcomes show that the covering highlight choice strategy performed better than CFC regarding exactness. Compared to Cantina, CANTINAC adds 10 additional elements. Meanwhile, phishing revealed that SMV replaced TF-IDF computation. With these upgrades, Cantina's deficiencies can be addressed. In any case, the new CANTINAC has a strict scope of use (Nguyen et al., 2014). Chiew et al. (2019) proposed another component determination technique called crossover gathering highlight choice (HEFS). The creators separated their proposed strategy into two stages: they utilized the aggregate conveyance work slope (CDF-G) in the main stage to create essential highlights and these elements were taken care of in the subsequent stage addressed by an information bother ensemble with the high capacity perturbation ensemble to deliver the other part of elements. A bunch of standard elements evaluated in these two stages are processed into the AI calculations used to differentiate phishing. The chosen highlights were taken care of in a few AI calculations; the best calculation regarding exactness was an RF that, when utilized with the standard elements, acquired a precision of 94.6%. The creators utilized two tests to approve their proposed strategy. As a programmed phishing discovery model, PhishStorm (Marchal et al., 2014) is carried out as a connection point between informal communication apparatuses and email servers. This section prepares RF algorithms by separating 12 significant URL elements. Still, it fails to identify multiple phishing attacks as it rarely contains sensitive highlights (Leskovec et al., 2010).

Table 1. Comparison of some learning algorithms of previous research.

Reference	Dataset Used 1= legitimate ,0= phishing	Method proposed	Classification Accuracy
Fette et al., 2007	6950 (1) & 860 (0)	LIBSVM	99%
Abu-Nimeh et al., 2007	1700 (0) & 1700(1)	LR, CART, BART, SVM, RF & NN	95.11%
Chandrasekaran et al., 2006	100 (0) & 100 (1)	SVM	95%
Jameel and George, 2013	(3000 (0) & 3000 (1)	FFNN	98.72%
Rathod and Pattewar, 2015	2500 (1) & 2100 (0)	NB	96.46%
Rawal et al., 2017	1605 (0) & 414 (1)	RF & SVM	99.87%
Shyni et al., 2016	5260 e-mails	SVM, RF & LB	96.3%
Smadi et al., 2015	5000 (0) & 5000 (1)	J48	98.11%
Mbah, 2017	6951(1) & 2357 (0)	KNN & J48	93.11%
Hota et al., 2018	1824 (0) & 1604 (1)	C4.5 & CART	99.27%
Fang et al., 2019	7781(1) & 999(0)	RCNN	99.848%
Aljofey et al., 2020	3000(1) & 3000(0)	RCNN	95.02%
Sonowal, 2020	1824 (0) & 1604 (1)	BSFS	97.41%
Bagui et al., 2021	3416 (0) & 14950 (1)	CNN	95.97%

In this section, we have studied the performance of anti-phishing and machine-learning algorithms. The results of this analysis move and represent more values of precision when websites with different Classifications are detected. Various algorithms have been used in Table 1 to enhance the accuracy, and the comparative study of the literature is presented.

Material and Methods

A URL is the address used for the presentation locale. A typical URL contains four parts: program, region name, recording method, and query limit (Leskovec et al.,

DOI:https://doi.org/10.52756/ijerr.2023.v36.009

2010). Internet resources can be accessed based on the address displayed by the URL. By eliminating subtle highlights in real and phishing URLs and their associated regions, critical ML classifiers are ready to identify phishing attacks. As shown in Figure 1, the calculation is first used to eliminate the features and its huge objection in the information test URL. The picking estimate is then used to create the optimal parts. Techniques for picking ideal components can reduce the obligation to set up basic ML classifiers. A test set of revised URLs that contain ideal components is the final step in the process of developing a new classifier.

Forward Selection

Forward selection is a coverage model that typically investigates a component's foresight capabilities and returns many components that perform surprisingly well (Bokrantz et al., (2020).

Pearson Correlation

Pearson Correlation is used to develop an association matrix that activates a direct connection between two features and gives a value between -1 and 1, showing how correlated the two components are to each other (Seo and Shneiderman, 2005).

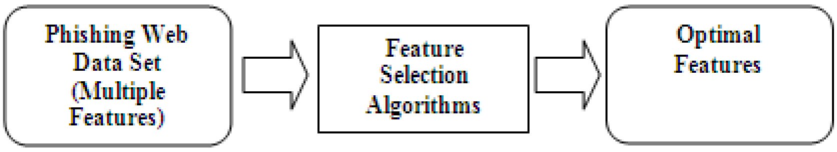
Phishing Web Data Set (Multiple Features)
Feature Selection Algorithms
Optimal Features

Figure 1. Workflow for generating optimal features from a phishing website dataset.

Feature Selection

All illegal URLs are treated as authenticated URLs by phishing attackers. By doing this, they can trick customers into sending phishing attacks quickly (Gupta et al., 2018). Luckily, not quite the same as legitimate URLs, phishing URLs have clear, recognizable elements. Following techniques are used as feature selection techniques.

Extra Tree Classifier

Extra Tree Classifier or Extremely Random Tree Classifier is a social event computation that seeds tree models built from a planning dataset for arbitrary reasons and sorts out the best-fit components (Tama and Lim, 2020). In Figure 2, out of 30 features (30 independent and 1 dependent), only 10 prominent ones are selected.

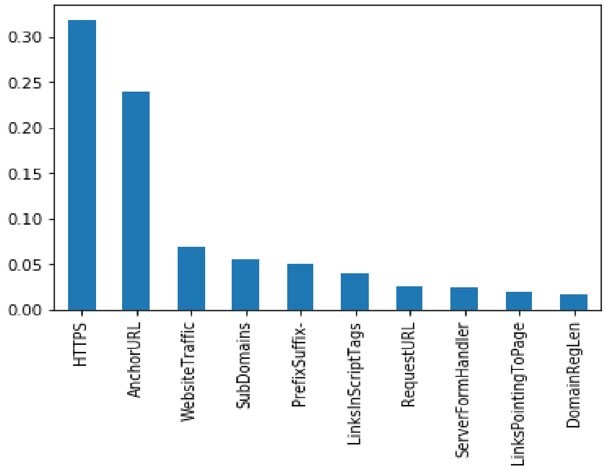

Figure 2. Feature selected by Extra Tree Classifiers

Statistical logit (LR) Model

Logit, or computed recurrence, is a quantifiable basic fundamental ability to display dual dependent variables in its basic design (Mahapatra et al., 2022). In a backward check, the logit model is investigated to determine the limits of the model. We only consider those features whose p-value is less than 0.05. The five features (Redirecting//, DomainRegLen, Favicon, AbnormalURL and UsingPopupWindow) have a p-value greater than 0.05, so we can drop those features and consider prominent 25 features for further analysis.

Principal Componant Analysis (PCA)

PCA is a dimensionality descent strategy that spends most of its time reducing the dimensionality of a huge heuristic list, by changing a variable huge plan to a more humble plan that really contains large in huge set (Chaurasiaet al., 2021). Eliminating elements or features of a file often leads to a weakness in precision. No need to deal with unnecessary variables due to more inconspicuous guiding classifications that are less easy to examine and imagine and make the separation of data estimated by AI clearer and faster.

The selected features and their numbers are presented in the following table 2 from the above-mentioned feature selection techniques.

Table 2. Selected features and their numbers by different feature selection techniques

Algorithms	No. of selected Features
Extra Tree Classifier	10
Forward Selection	6
Pearson Correlation	10
Logit (LR)	25

From ET classifier, Forward Selection and Pearson correlation, the hypothesis suggests that PrefixSuffix-, SubDomains, HTTPS, AnchorURL, ServerFormHandler and Website Traffic are 6 important features. At the same time, the logit (LR) model suggests 25 most significant features.

The following 4 cases were studied to discover the features most significant for predicting the phishing websites.

Case-1: Analysis of common features shared by Extra Tree classifier, Forward Selection and Pearson correlation.

Case-2: Analysis of features selected by the logit (LR) model.

Case-3: Analysis of all features.

Case -4: Analysis of features selected by PCA (3 and 5 components).

The AI calculations are planned so that they gain as a matter of fact and their presentation improves as they feed on an ever-increasing amount of information. Each calculation has its own specific manner of learning and anticipating the information. In this section, we will discuss the working of following AI calculations and a portion of the numerical conditions carried out in those calculations that are used in the learning system.

Logistic Regression

LR is an action estimation procedure that measures the after-effects of total variables considering independent elements. It is basically used for analysing and fitting data to fundamental constraints. The likelihood increase depends on the free factors' coefficients within the determined limits. Gradient descent values decide the cost limit.

$$cost(h_\theta(x), y) = -\log\big(h_\theta(x)\big)\ if\ y = 1$$

$$cost(h_\theta(x), y) = -\log(1 - h_\theta\,(x))\ if\ y =\ 0$$

Cost function of LR

$$P = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots \beta_n X_n)}}$$

LR Equation

Naive Bayes

NB is a planning computation that relies on Bayes' theorem. The assessment acknowledges that no relationship exists between the autonomous components. That is the specific condition in which a part in one selection is independent in the presence of another section in a similar class. We make a repeating table for all tags against the class and calculate the probability of a large number of pointers. With NB conditions, the backs of all courses are not set. The highest probability of all classes evaluates the result of the NB classifier.

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)}$$

$$P(c|x) = P(x_1|c) \times P(x_1|c) \times \ldots \times P(x_n|c) \times P(c)$$

Where, c→class, x→predictor

Decision Tree

The decision tree is basically used for action or decision for regrate. Decision tree helps evaluate the dataset quality by calculating entropy and information gain. We have used techniques to split the dataset for quality observation. The DT use the Gini Index as an important model.

$$Entropy = \sum_{i=1}^{c} -p_i \times log_2(p_i)$$

Where, c→No. of classes

$$Gini\ = 1 - \sum_{i=1}^{c} (p_i)^2$$

K-Nearest Neighbor

Regression and classification are both techniques used for data quality estimation. The algorithm helps in estimating the value of Euclidean distance. The algorithms are also used for the highest distance as the manhattam distance.

$$\sqrt{\sum_{i=1}^{k} (x_i - y_i)^2}\ \text{Euclidean}$$

$$\sum_{i=1}^{k} |x_i - y_i|\ Manhattan$$

$$\left(\sum_{i=1}^{k} (|x_i - y_i|)^q\right)^{1/q}\ Minkowski$$

Distance Metrics

SVM

Furthermore, SVM analyzed the learning scheme, gave details of the problem, and accessed an ideal hyperplane in N-layer space. The ideal plan is augmenting edge distances between class views using hinge disaster work. The part of the hyperplane depends on how much information is highlighted. If N represents features, N-1 will represent hyperplane.

$$l(y) = \max\,(0, 1 + \overset{max}{y \neq t} w_y x - w_t x)$$

We calculate the loss function as t, which represents the target variable, w, which represents the model parameter and x, which is the input variable.

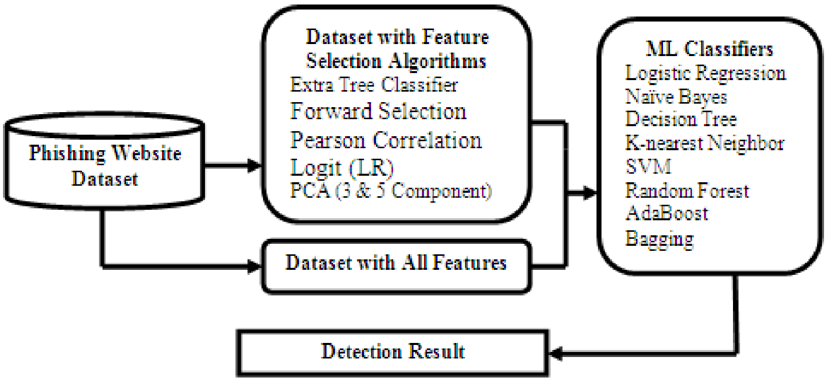

Figure 3. Flow diagram of working method

Random Forest

Random forests collect various decision trees and work as an ensemble for a special model. In RF, the output prepared as special class and votting is converted into a gauge for RF for each decision. Some techniques for ensuring this are through bagging and feature selection. Storage is the process of picking out sporadic instances of insight from a dataset.

AdaBoost

AdaBoost computation, short for Adb is used as an ensemble technique in AI. It is called adaptive boosting because the heap is reallocated to each case, with a higher burden assigned to the badly requested model. Support is used to reduce tendencies and shakes in supervised learning. It applies the rules that the learner gradually grows up. With the exception of the principal, each subsequent learner is made up of learners who actually develop. In clear words, weak learners become strong learners. AdaBoost is estimated to work on a comparison rule as a slightly qualified support.

Bagging

Bagging produces extra information for preparing the dataset. This is accomplished by irregular inspecting with substitution from the first dataset. Inspecting with substitution might rehash a few perceptions in each new preparation informational collection. Each component in Bagging is similarly likely to show up in a new dataset.

These multi-datasets are utilized to prepare different models in equal. The normal of the multitude of expectations from various group models is determined. A significant portion of the votes obtained through the democratic system are considered to be as follows: when the order is established, showing reduces the difference and adjusts the forecast to something that is more typical.

Flow model of work

Fig. 3 shows the work process of recognizing phishing assaults of the mode. As is displayed here, the phishing site dataset is separated by two strategies. In the first method, several important algorithms are used, and they only select prominent and sensitive features. And in the second method, the dataset carry forward with full features in the next level, where several machine learning classifiers were applied. Following that, a comparison was made between the accuracy results provided by the machine learning classifiers in each method. We see that the feature selection method is more appropriate.

Description of Datasets

The information classification is the pick structure UCI Phishing Guide Classification (UCI Repository). Instructive records come from 11,054 models, with 55.69% legitimate URLs and 44.31% phishing URLs. Meanwhile, 80% of the direct 11054 models are used to set the classifier, and 20% of the models are used to evaluate the presentation of the model. In fact, a modest number of tests can cause conclusive classifiers to suffer from underfitting and weak hypothesis constraints. Contrary to the norm, when all instances of instructive classification are used for preparation, the classifier will fall into the problem of overfitting and powerless action results.

Results and Discussion

In case 1, the Extra Tree classifier, Forward Selection and Pearson relationship model chose the elements (Prefix Suffix-, Sub Domains, HTTPS, Anchor URL, Server Form Handler, Website Traffic). A few ML classifiers (LR, NB, DT, KNN, SVM, RF, AdaBoost and Bagging) were applied for precision assessment. As shown in Table 3, the most elevated exactness (93.86%) attracts between two classifiers, for example, Random forest and Bagging.

In case 2, 25 features (Table 1) were chosen by the logit (LR) model. The same classifiers were applied to this situation. In precision astute, we can find in Table 2 that random forest again got higher exactness (97.30%) among every one of the classifiers.

In case 3, all dataset features were taken to extract the precision of the classifiers. Again Random forests gain higher precision, for example, 97.10% (table 3).

DOI:https://doi.org/10.52756/ijerr.2023.v36.009

Table 3. Obtained accuracy in different cases.

Algorithms	Case-1 (6 Features)	Case-2 (Logit) (25 Features)	Case-3 (All Features)	Case-4 (PCA)	
				3-componants	5-componants
Logistic Regression	0.9137	0.9271	0.9267	0.8141	0.9242
Naïve Bayes	0.9082	0.9144	0.9097	0.8211	0.9025
Decision Tree	0.9362	0.9425	0.9293	0.8580	0.9384
K-Nearest Neighbor	0.9176	0.9650	0.9598	0.8978	0.9510
SVM	0.9361	0.9522	0.9483	0.8317	0.9278
Random Forest	0.9386	0.9730	0.9710	0.9544	0.9748
AdaBoost	0.9310	0.9390	0.9131	0.8280	0.9230
Bagging	0.9386	0.9697	0.9457	0.9442	0.9703

In case 4, try is led with Principal component analysis (3 and 5 parts). The Random forest again got the higher exactness (95.44% with 3-component and 97.48% with 5-component). Figure 4 is drawn for better comprehension of the results created by Table 2.

Since in all cases in general, the case-2 logit (LR) model with 25 highlights has performed well. So, we have drawn a ROC (AUC) curve inside different classifiers. The RF, LR, NB, DT, SVC, AdaBoost and Bagging classifier get a higher accuracy (AUC) of approximately 100% (Figure 5).

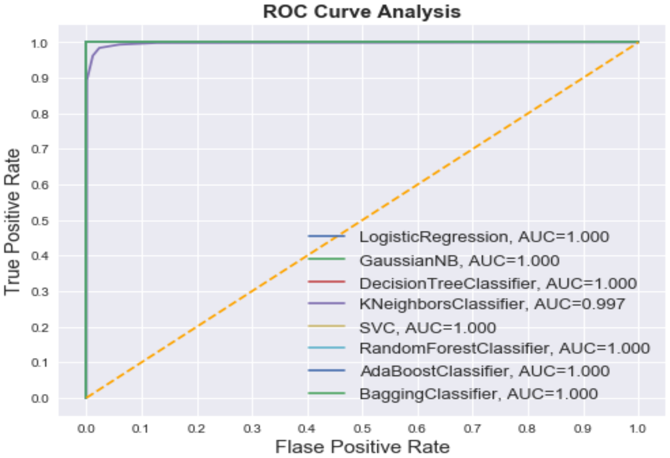

Figure 5. ROC (AUC) curve by logit (LR) model with 25 features.

Conclusion and Future Work

In this research, a four-feature selection calculation and principal inspection are first characterized to analyse the impact of the impact of fine highlights on phishing recognition. Then, at this point, given the results of these component selection processes, the ideal element determination computation aims to find a specific idea value of the vector for machine learning techniques. The evaluation can handle a large number of fishing-sensitive

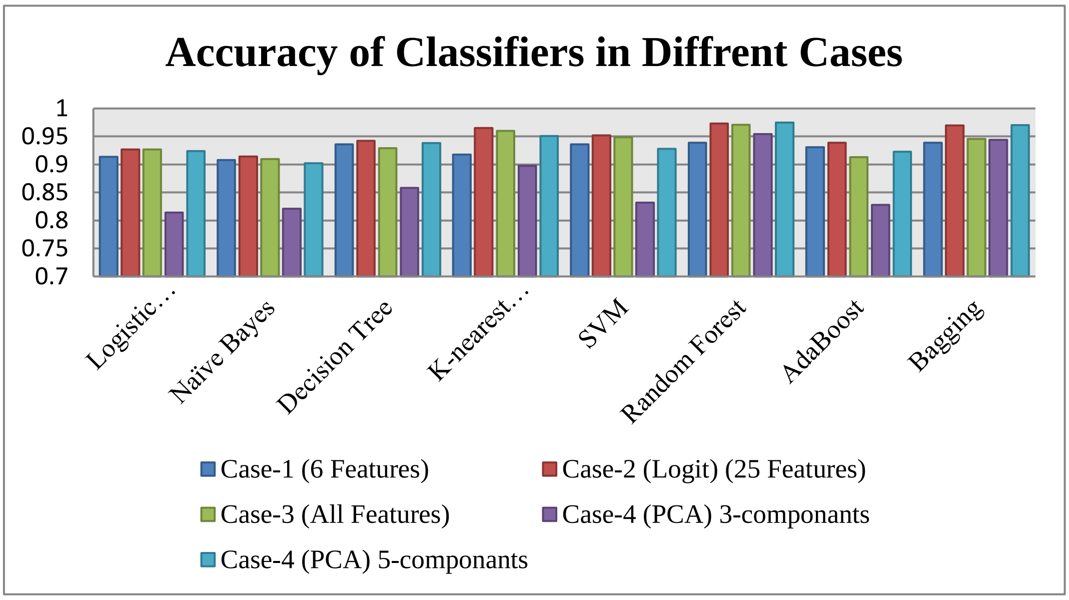

Figure 4. Accuracy of classifiers in different cases

elements and changing highlights. Subsequently, it can alleviate the over-fitting problem of ML classifiers. Finally, through trial-and-error investigation, the ideal machine learning classifier is ready to identify phishing attacks. Two important outcomes resulted from this experiment. First, we can conclude that among the entire feature selection model, the Logit (LR) model with 25 features in case 2 performed well (Table 3), and secondly, RF calculated a high score among all the classifiers in entire cases (Table 3). So, the Logit (LR) model for feature selection and Random forest for accuracy measurement could be more appropriate for detecting phishing websites. As the subtle elements of a phishing attack continue to change, collecting more elements later for ideal element determination is important.

Acknowledgement

I would like to thank my supporting researchers from the bottom of my heart for their assistance during this study project.

Conflict of Interests

The authors announce that there is no conflict of interest concerning the distribution of this paper.

References

Abu-Nimeh, S., Nappa, D., Wang, X., & Nair, S. (2007, October). A comparison of machine learning techniques for phishing detection. In *Proceedings of the Anti-phishing Working Groups 2nd Annual eCrime Researchers Summit*, pp. 60-69. https://doi.org/10.1145/1299015.1299021

Aleroud, A., & Zhou, L. (2017). Phishing environments, techniques, and countermeasures: A survey. *Computers & Security*, *68,* 160-196.https://doi.org/10.1016/j.cose.2017.04.006.

Aljofey, A., Jiang, Q., Qu, Q., Huang, M., & Niyigena, J. P. (2020). An effective phishing detection model based on character level convolutional neural network from URL. *Electronics*, *9*(9), 1514. https://doi.org/10.3390/electronics9091514

Babagoli, M., Aghababa, M. P., & Solouk, V. (2019). Heuristic nonlinear regression strategy for detecting phishing websites. *Soft Computing*, *23*(12), 4315-4327. https://doi.org/10.1007/s00500-018-3084-2

Babagoli, M., Aghababa, M. P., & Solouk, V. (2019). Heuristic nonlinear regression strategy for detecting phishing websites. *Soft Computing*, *23*(12), 4315-4327. https://doi.org/10.1007/s00500-018-3084-2.

Bagui, S., Nandi, D., Bagui, S., & White, R. J. (2021). Machine learning and deep learning for phishing email classification using one-hot encoding. *Journal of Computer Science*, *17*, 610-623. https://doi.org/10.3844/jcssp.2021.610.623

Basnet, R. B., Sung, A. H., & Liu, Q. (2012). Feature selection for improved phishing detection. In *Advanced Research in Applied Artificial Intelligence: 25th International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2012, Dalian, China, June 9-12, 2012.* Springer Berlin Heidelberg,*Proceedings 25*, pp. 252-261. https://doi.org/10.1007/978-3-642-31087-4_27.

Bokrantz, J., Skoogh, A., Berlin, C., Wuest, T., & Stahre, J. (2020). Smart Maintenance: a research agenda for industrial maintenance management. *International Journal of Production Economics*, *224*, 107547. https://doi.org/10.1016/j.ijpe.2019.107547.

Cawley, G. C., & Talbot, N. L. (2010). On over-fitting in model selection and subsequent selection bias in performance evaluation. *The Journal of Machine Learning Research*, *11*, 2079-2107.

Chandrasekaran, M., Narayanan, K., & Upadhyaya, S. (2006). Phishing email detection based on structural properties. In *NYS Cyber Security Conference*, *3*, 2-8. https://doi.org/10.4236/aces.2019.94023

Chaurasia, V., & Pal, S. (2021). Ensemble Technique to Predict Breast Cancer on Multiple Datasets, *The Computer Journal*, bxab110, https://doi.org/10.1093/comjnl/bxab110.

Chaurasia, V., Pandey, M. K., & Pal, S. (2021, March). Prediction of presence of breast cancer disease in the patient using machine learning algorithms and SFS. IOP Publishing,In *IOP conference series: Materials Science and Engineering*, *1099*(1), 012003. https://doi.org/10.1088/1757-899X/1099/1/012003.

Chaurasia, V., Pandey, M. K., & Pal, S. (2022). Chronic kidney disease: a prediction and comparison of ensemble and basic classifiers performance. *Human-Intelligent Systems Integration*, pp. 1-10. https://doi.org/10.1007/s42454-022-00040-y.

Chiew, K. L., Tan, C. L., Wong, K., Yong, K. S., & Tiong, W. K. (2019). A new hybrid ensemble feature selection framework for machine learning-based phishing detection system. *Information Sciences*, *484*, 153-166. https://doi.org/10.1016/j.ins.2019.01.064.

Dawn, N., Ghosh, T., Ghosh, S., Saha, A., Mukherjee, P., Sarkar, S., Guha, S., & Sanyal, T. (2023). Implementation of Artificial Intelligence, Machine Learning, and Internet of Things (IoT) in revolutionizing Agriculture: A review on recent trends and challenges. *International Journal of Experimental Research and Review*, *30*, 190-218. https://doi.org/10.52756/ijerr.2023.v30.018

Fang, Y., Zhang, C., Huang, C., Liu, L., & Yang, Y. (2019). Phishing email detection using improved

RCNN model with multilevel vectors and attention mechanism. *IEEE Access*, *7*, 56329-56340. https://doi.org/10.5120/ijca2022921868

Fette, I., Sadeh, N., & Tomasic, A. (2007, May). Learning to detect phishing emails. In *Proceedings of the 16th International Conference on World Wide Web*, pp. 649-656. https://doi.org/10.1145/1242572.1242660

Guan, H., Zhang, Y., Xian, M., Cheng, H. D., & Tang, X. (2021). SMOTE-WENN: Solving class imbalance and small sample problems by oversampling and distance scaling. *Applied Intelligence*, *51*(3), 1394-1409. https://doi.org/10.1007/s10489-020-01852-8.

Gupta, S. S., Thakral, A., & Choudhury, T. (2018). Social media security analysis of threats and security measures. IEEE, In *2018 International Conference on Advances in Computing and Communication Engineering (ICACCE),* pp. 115-120. https://doi.org/10.1109/ICACCE.2018.8441710.

Han, X., Kheir, N., & Balzarotti, D. (2016, October). Phisheye: Live monitoring of sandboxed phishing kits. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security,* pp. 1402-1413. https://doi.org/10.1145/2976749.2978330.

Hota, H. S., Shrivas, A. K., & Hota, R. (2018). An ensemble model for detecting phishing attack with proposed remove-replace feature selection technique. *Procedia Computer Science*, *132*, 900-907. https://doi.org/10.1016/j.procs.2018.05.103.

Jain, A. K., & Gupta, B. B. (2016). A novel approach to protect against phishing attacks at client side using auto-updated white-list. *EURASIP Journal on Information Security*, *2016*(1), 1-11. https://doi.org/10.1186/s13635-016-0034-3.

Jain, A. K., & Gupta, B. B. (2018). Towards detection of phishing websites on client-side using machine learning based approach. *Telecommunication Systems*, *68*, 687-700. https://doi.org/10.1007/s11235-017-0414-0.

Jameel, N. G. M., & George, L. E. (2013). Detection of phishing emails using feed forward neural network. *International Journal of Computer Applications*, *77*(7). https://doi.org/10.5120/13405-1057

Jamil, A., Asif, K., Ghulam, Z., Nazir, M. K., Alam, S. M., & Ashraf, R. (2018). Mpmpa: A mitigation and prevention model for social engineering based phishing attacks on facebook. IEEE,*In 2018 IEEE International Conference on Big Data (Big Data),* pp. 5040-5048. https://doi.org/10.1109/BigData.2018.8622505.

Kang, J., & Lee, D. (2007, November). Advanced white list approach for preventing access to phishing sites. IEEE, In *2007 International Conference on Convergence Information Technology (ICCIT 2007)*, pp. 491-496. https://doi.org/10.1109/ICCIT.2007.50.

Khonji, M., Jones, A., & Iraqi, Y. (2013). An empirical evaluation for feature selection methods in phishing email classification. *International Journal of Computer Systems Science & Engineering*, *28*(1), 37-51. https://doi.org/10.1109/SURV.2013.032213.00009.

Lee, L. H., Lee, K. C., Chen, H. H., & Tseng, Y. H. (2014, November). Poster: Proactive blacklist update for anti-phishing. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1448-1450. https://doi.org/10.1145/2660267.2662362.

Leskovec, J., Huttenlocher, D., & Kleinberg, J. (2010, April). Predicting positive and negative links in online social networks. In *Proceedings of the 19th International Conference on World wideWe,b* pp. 641-650. https://doi.org/10.1145/1772690.1772756.

Li, X., Geng, G., Yan, Z., Chen, Y., & Lee, X. (2016, December). Phishing detection based on newly registered domains. IEEE, In *2016 IEEE International Conference on Big Data (big data),* pp. 3685-3692. https://doi.org/10.1109/BigData.2016.7841036.

Mahapatra, M., Majhi, S. K., & Dhal, S. K. (2022). Mrmr-ssa: a hybrid approach for optimal feature selection. *Evolutionary Intelligence*, *15*(3), 2017-2036.https://doi.org/10.1007/s12065-021-00608-8.

Marchal, S., François, J., State, R., & Engel, T. (2014). Phish Storm: Detecting phishing with streaming analytics. *IEEE Transactions on Network and Service Management*, *11*(4), 458-471. https://doi.org/10.1109/TNSM.2014.2377295.

Mbah, K. (2017). A phishing e-mail detection approach using machine learning techniques. Computer and Information Engineering, vol. 3(1), pp. 2333. https://doi.org/10.1080/01430750.2021.1953590.

Mohammad, R. M., Thabtah, F., & McCluskey, L. (2014). Predicting phishing websites based on self-structuring neural network. *Neural Computing and Applications*, *25*, 443-458. https://doi.org/10.1007/s00521-013-1490-z.

Mohammad, R., McCluskey, T. L., & Thabtah, F. (2013). Predicting phishing websites using neural network trained with back-propagation. *World Congress in Computer Science, Computer Engineering, and Applied Computing,25*, 443–458. https://doi.org/10.1007/s00521-013-1490-z.

Nguyen, L. A. T., To, B. L., Nguyen, H. K., & Nguyen, M. H. (2014, October). An efficient approach for phishing detection using single-layer neural network. IEEE, In *2014 International Conference on Advanced Technologies for Communications (ATC 2014),* pp. 435-440. https://doi.org/10.1109/ATC.2014.7043427.

Rajab, M. (2018, February). An anti-phishing method based on feature analysis. In *Proceedings of the 2nd International Conference on Machine*

DOI:https://doi.org/10.52756/ijerr.2023.v36.009

Learning and Soft Computing, pp. 133-139.https://doi.org/10.1145/3184066.3184082.

Rathod, S. B., & Pattewar, T. M. (2015). Content based spam detection in email using Bayesian classifier. IEEE, In *2015 International Conference on Communications and Signal Processing (ICCSP)*, pp. 1257-1261. https://doi.org/10.1109/ICCSP.2015.7322709

Rawal, S., Rawal, B., Shaheen, A., & Malik, S. (2017). Phishing detection in e-mails using machine learning. *International Journal of Applied Information Systems*, *12*(7), 21-24. https://doi.org/10.5120/ijais2017451713

Seo, J., & Shneiderman, B. (2005). A rank-by-feature framework for interactive exploration of multidimensional data. *Information Visualization*, *4*(2), 96-113. https://doi.org/10.1057/palgrave.ivs.9500091.

Sharfuddin, N., Anwer, F., & Ali, S. (2023). A Novel Cryptographic Technique for Cloud Environment Based on Feedback DNA. *International Journal of Experimental Research and Review*, *32*, 323-339. https://doi.org/10.52756/ijerr.2023.v32.028

Sharifi, M., & Siadati, S. H. (2008, March). A phishing sites blacklist generator. IEEE, In *2008 IEEE/ACS International Conference on Computer Systems and Applications*, pp. 840-843. https://doi.org/10.1109/AICCSA.2008.4493625.

Shyni, C. E., Sarju, S., & Swamynathan, S. (2016). A multi-classifier based prediction model for phishing emails detection using topic modelling, named entity recognition and image processing. *Circuits and Systems*, *7*(9), 2507-2520. https://doi.org/10.4236/cs.2016.79217.

Smadi, S., Aslam, N., Zhang, L., Alasem, R., & Hossain, M. A. (2015, December). Detection of phishing emails using data mining algorithms. IEEE, In *2015 9th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)*, pp. 1-8. https://doi.org/10.1109/SKIMA.2015.7399985.

Sonowal, G. (2020). Phishing email detection based on binary search feature selection. *SN Computer Science*, *1*(4), 191.https://doi.org/10.1007/s42979-020-00194-z.

Tama, B. A., & Lim, S. (2020). A comparative performance evaluation of classification algorithms for clinical decision support systems. *Mathematics*, *8*(10), 1814. https://doi.org/10.3390/math8101814.

UCI Machine Learning Repository (2022). Center for Machine Learning and Intelligent Systems. Accessed: 2022. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/phishing+website.

Xiang, G., Hong, J., Rose, C. P., & Cranor, L. (2011). Cantina+ a feature-rich machine learning framework for detecting phishing web sites. *ACM Transactions on Information and System Security (TISSEC)*, *14*(2), 1-28. https://doi.org/10.1145/2019599.2019606.

Zhang, Y., Hong, J. I., & Cranor, L. F. (2007). Cantina: a content-based approach to detecting phishing web sites. In *Proceedings of the 16th International Conference on World Wide Web*, pp. 639-648. https://doi.org/10.1145/1242572.1242659.

Zhu, E., Chen, Y., Ye, C., Li, X., & Liu, F. (2019). OFS-NN: an effective phishing websites detection model based on optimal feature selection and neural network. *IEEE Access*, *7*, 73271-73284. https://doi.org/10.1109/ACCESS.2019.2920655.

How to cite this Article:

Rekha Pal, Mithilesh Kumar Pandey, Saurabh Pal, Dhyan Chandra Yadav (2023). Phishing Detection: A Hybrid Model with Feature Selection and Machine Learning Techniques.*International Journal of Experimental Research and Review*, *36*, 99-108.

DOI: https://doi.org/10.52756/ijerr.2023.v36.009

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.