Available online at www.garmian.edu.krd

Passer Journal of Basic and Applied Sciences

Passer 7 (Issue 1) (2025) 370-377

Original Article

Enhancing Phishing Detection: A Comparative Analysis of Machine Learning Techniques

Peshraw Salam Abdalqadir*[1], Sirwan Mohammed Aziz[1], Brzu Tahir Muhammed[2], Ardalan Hussein Awlla[3]

[1]*Department of Computer Science, Darbandikhan Technical Institute, Sulaimani Polytechnic University, Kurdistan Region, Iraq.*
[2]*Department of Computer Science, Kurdistan Technical Institute, Sulaimani, Kurdistan Region, Iraq.*
[3]*Department of Computer Science, Cihan University Sulaimani, Sulaimani, Kurdistan Region, Iraq.*

Received 24 June 2024; revised 30 MArch 2025;
accepted 10 April 2025; available online 08 June 2025

DOI: 10.24271/PSR.2025.464499.1640

ABSTRACT

Phishing attacks trick people into giving away personal information on fake websites. These attacks are still a big problem for cybersecurity because traditional methods often can't keep up with new tricks. This study evaluated various supervised machine learning (ML) classifiers to identify phishing websites and examined how feature selection can enhance their performance. Seven ML classifiers were tested: Naive Bayes, Multilayer Perceptron (MLP), AdaBoost, Bagging, Support Vector Machine (SVM), JRip, and REP Tree. These classifiers were trained on a dataset consisting of 2,500 instances, comprising 1,220 legitimate websites and 1,280 phishing sites, using 10-fold cross-validation. Feature selection techniques, specifically Principal Component Analysis (PCA) and Incremental Wrapper Subset Selection (IWSS), were employed to improve the input features. The results showed that PCA significantly improved detection rates through dimensionality reduction while zretaining key discriminative features, achieving an accuracy of 99.8%. Among the classifiers, MLP outperformed the others with an accuracy of 92.94%, and when PCA was applied to MLP, the accuracy increased to 97.5%. These findings highlight the importance of feature optimization in identifying phishing websites and demonstrate that MLP is a suitable model for real-world applications.

Keywords: Phishing detection, machine learning, feature selection, Principal Component Analysis (PCA), cybersecurity.

1 Introduction

In the current context of web pages, a web phishing attack means cybercriminals deceiving people over the internet by sending emails or messages under the name of real companies such as banks. Such messages mainly offer links that lead to look-alike sites of genuine ones which can be malicious. The moment that the clients give login information into these fake sites, the attackers can use the obtained information in order to embezzle the money or other important data [1, 2]. While spoofing refers to the act of impersonating a legitimate third party with the intent of gaining unauthorized access to secure data. Website phishing attacks manipulate semantics by focusing on users rather than computers. In spoofing, the hackers ensure that they develop exact and professional look alike sites which the client takes to be the original site which is rather tricky. In some ways it is even more risky for internet crime than the virus or piracy kind of crimes. Phishers utilize webpage templates, similar characters, etc., to mimic authentic sites and deceive visitors into divulging personal information like usernames and passwords as illustrated in Figure 1, precisely identifying phishing sites within allocated timeframes is crucial for the advancement of phishing website detection methods [3].

The Anti-Phishing Working Group (APWG) has reported that the total number of confirmed phishing attacks has more than quadrupled compared to this time last year. The second quarter of 2022 saw between 68,000 and 94,000 attacks per month, this subsequently reached 1,097,811, making it the most pronounced quarter in history for phishing, according to APWG [4], these numbers show the growing danger of phishing attempts and the need for better security measures, which is what this research attempts to do.

* Corresponding author
E-mail address Peshraw.salam@garmian.edu.krd (Instructor).
Peer-reviewed under the responsibility of the University of Garmian.

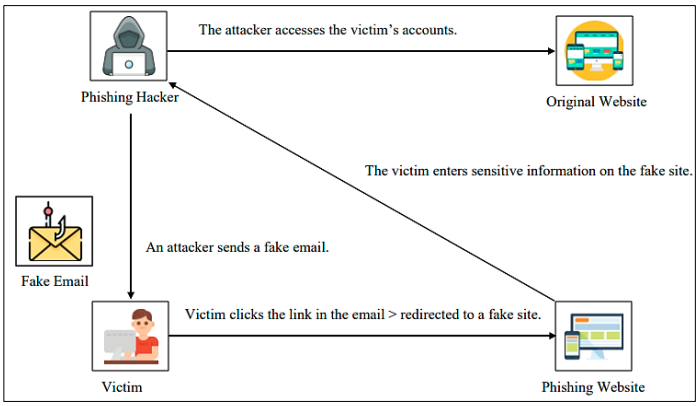

Figure 1: A common phishing scenario.

Phishing identification depends on analyzing website content or utilizing external data from servers to classify URLs as legitimate or fraudulent. Attack targets are selected by monitoring phishing website URLs, scouring subdirectories, employing well-known standard identification techniques, and eliminating variations.

This study employed various supervised machine learning classification algorithms such as Multilayer Perceptron (MLP), Support Vector Machine (SVM), Naive Bayes, Bagging, etc., to construct an automated model for identifying phishing websites. Also, feature selection algorithms used for selecting the optimal subset of features from the general feature set of the phishing datasets were employed with the intention to boost the model's performance and accuracy. Model performance was assessed by means of such criteria as Accuracy, Error Rate, F-score, etc.

The rest of the paper is organized as follows: Section 2 discusses the related work. Section 3 presents the methodology that will be used in support of this study. Section 4 elaborates on modelling and experimental processes. Section 5 finally gives the findings and conclusions reached from our analysis.

The text may contain up to 10 short subheadings in total and can include a maximum of 6 display items (figures or tables). Passer allows up to 60 references per article. All sections and subsections must strictly adhere to the formatting guidelines specified in this template, with consistent sizes and styles applied throughout the document.

The title should be in 14-point Times New Roman, bold, and must not exceed two lines of 90 characters (including spaces). It should avoid numbers, abbreviations, acronyms, or punctuation.

Authors' names should be formatted in 12-point Times New Roman and written in full, avoiding abbreviations (e.g., "Hossein Mehranfar" instead of "H. Mehranfar"). Affiliations must adhere to the structure outlined in the example provided in the template. Additionally, the corresponding author should be clearly identified, with a single e-mail address provided in the footnote for correspondence.

The spacing between lines should be set to 1 inch, with a 1-inch margin on all sides (top, bottom, right, and left). Each paragraph should begin flush with the left margin, without any indentation. Bold and italic text should be avoided within paragraphs. Instead of using quotation marks (" "), single quotes (' '), or brackets (), maintain a consistent style throughout the document.

The Introduction should provide a concise overview of the study's context, highlighting its significance and purpose, with up to 500 words of referenced background. It must include a thorough review of the current research field, citing key publications and addressing conflicting hypotheses if relevant.

2 Related work

Phishing attacks represent one of the most prevalent cybersecurity threats and have been addressed through a variety of detection strategies, such as blacklist-based, heuristic, and machine learning-based techniques. The dynamic nature of phishing URLs renders conventional approaches like blacklists and rule-based systems ineffective against newly deployed phishing websites. Interestingly, recent studies have shown that machine learning models such as Decision Trees (DT), Support Vector Machines (SVM), Random Forest (RF), and Naïve Bayes (NB) are capable of classifying URLs as phishing websites. Recent approaches have better accuracy by leveraging advanced ensemble techniques or creating hybrid models involving machine learning and deep learning. A hybrid LSD model (Logistic Regression + SVM + Decision Tree) with both soft and hard voting is further improved by the application of Canopy feature selection, cross-fold validation, and Grid Search Hyperparameter Optimization. This study [5] shows that such a hybrid classifier approach has higher accuracy, achieving a detection rate of 98.12%, on par with traditional machine learning classifiers. These results denote a gradual attestation to the efficiency of ensemble and hybrid models, paving the way towards robust detection of phishing threats. In addition, this studies [6, 7] used decision trees and neural networks for the classification of URLs in the context of phishing research. and obtained a good result. This is a clear indication of Trend Micro and other cybersecurity firms embedding machine learning to combat phishing attacks of this sophistication. function classifiers such as support vector machines [8, 9] decision tree classifiers like REPTree, and rules classifiers. The studies exclude classifiers with low precision or those similar to others. Sufficient data is essential for scrutinizing the dataset to identify phishing occurrences. Throughout the research, emphasis is placed on determining the most effective classifier algorithm. Barraclough [10] innovatively employed a neuro-fuzzy methodology to identify fraudulent websites. From its innovation in collecting data from various sources such as Phish Tank, Legitimate Website Rules, User-specific Websites and User mobile-to-online profiles, the approach of the model did, in fact, reach a 98.5% accuracy rate. In addition to this, the inclusion of extra attributes that accounted for a minor error, the model's technique was still quite fruitful. The examination of Barraclough [11] parameters reinvented the standardization method, which is

based on the utilization of six credit providers to include the 352 credits. Furthermore, this framework gave a most laudable percentage of 97% program accuracy in accurately identifying rogue websites.

Nawafleh and Hadi [12] introduce an innovative associative classification rule for identifying phishing websites. Their study demonstrates the effectiveness of this approach through observational data analysis. The results indicate that employing classification proves to be a practical and effective method, showcasing competitive performance when compared against alternative estimations like SVM, Prim's, Manslayer, and Naive Bayes (NB). Pradeep and Ravendra [13] endeavored to devise a model capable of discerning phishing websites. Employing six distinct machine learning algorithms, they achieved classification accuracies of 92.7846%, 95.11%, 96.57%, 96.3%, 93.85%, and 93.4039%, respectively. These algorithms encompassed Nave Bayes, J48, SVM, and Random Forest. They want to improve the current methods and, with the help of algorithms, to find out if they can get more accurate results of the diagnosis. In addition, they come forward with one more improved way of classification, the Neural Net, to further beef up their analysis equipment.

A research team led by He et al. [14] came up with a 12-feature framework for the phish website. They implement the classification methods that come into their approach such as decision trees, random forests, and Naive Bayes (NB) to efficiently detect the attackers [15]. Many researchers focus on the malware learning approach they have outlined in the section "Data Mining: Various methods to reduce the threat of phishing" [16] which shows the effort for phishing threat mitigation through such computational techniques.

3 Methodology

Improving the accuracy of recognizing web pages is important for effective detection of phishing. Machine learning algorithms can learn to recognize real web pages by predicting if they are genuine. These algorithms use a technique called URL filtering, which checks the structure of web addresses to stop phishing sites. This method has been shown to decrease errors in phishing detection systems and make them more accurate. In this study, we developed an application equipped with machine learning algorithms to differentiate between normal and phishing websites and find the accuracy of each algorithm, then apply the best feature extraction technique to improve accuracy.

The system diagram Figure 2, illustrates the process of phishing detection. A classifier within the system evaluates various attributes of web addresses to determine if a website is legitimate or potentially malicious. Five main steps are involved in recognizing phishing websites: dataset selection, attribute extraction, training machine learning classifiers, and evaluating the best-performing classifiers for accuracy.

The experiments use groups consisting of various easily recognizable classifiers, including Naïve Bayes, Multilayer Perceptron (MLP), Ada Boost, Bagging, Support Vector Machine (SVM), Repeated Incremental Pruning to Produce Error Reduction (RIPPER) and Reduced Error Pruning Tree (REP). Algorithms were chosen for balance of speed/accuracy (Naïve Bayes, with no hyperparameter tuning, serves as a simple baseline. MLP (two hidden layers, ReLU) captures non-linear patterns, while SVM (RBF kernel, C = 1.0) handles high-dimensional data. AdaBoost and Bagging (50 estimators, decision tree base) reduce overfitting. JRip and REP Tree (min. two instances per leaf) offer interpretable rule-based detection). Correspondingly, model accuracy in the context of feature selection can be improved through contemporary methods like PCA, Particle swarm optimization (PSO), Insect Search, IWSS, each using different optimization techniques to identify the most relevant features.

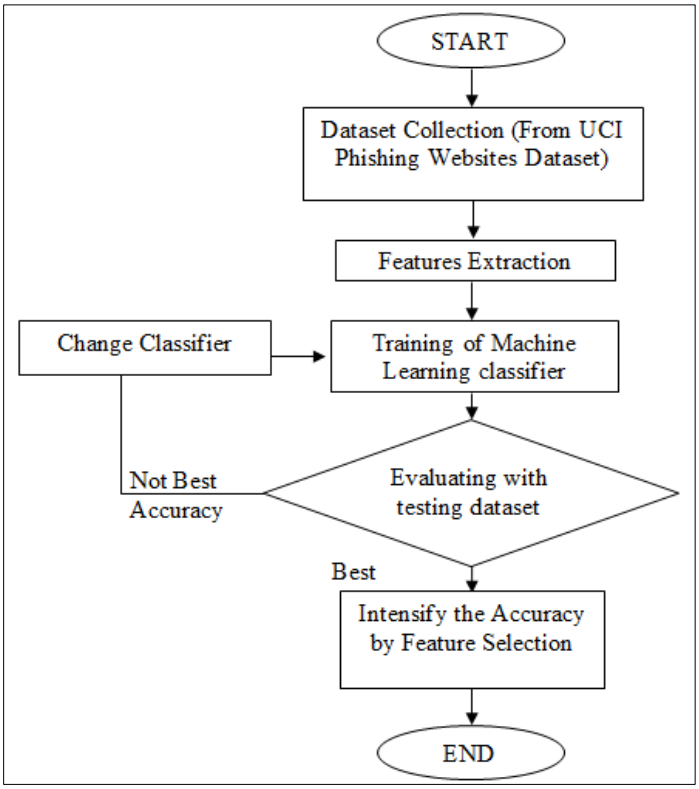

Figure 2: System diagram of phishing detection.

3.1 Dataset

The dataset used for these experiments consists of phishing addresses collected from different repositories such as Phishing documents, Miller Smiles archives, and Google. The dataset contains 30 main features to discriminate the websites as the phishing ones and the legitimate ones, the dataset contains 2500 phishing URLs, with phishing URLs 1220 and legitimate URLs 1280, sourced from PhishTank, Miller Smiles, and verified legitimate URLs from Google Safe Browsing. There are attributes that have been sorted into groups according to their address bar characteristics, irregularities, HTML and JavaScript elements, and domain properties [17].

Table 1: Attributes of the dataset.

No.	Attribute Name	Values	Details
1	Having an IP Address	{-1,1}	Utilizing the IP Address
2	Length of URL	{1,0,-1}	The Long Web address to Cover the Uncertain Section
3	Shortening Service	{1,-1}	Using URL Shortening Services “TinyURL”
4	At Symbol (@)	{1,-1}	The "@" sign is used in Web addresses.
5	redirecting with a double-slash(//)	{-1,1}	Using the "//" redirection symbol
6	Prefix and Suffix	{-1,1}	attaching a prefix or suffix to the domain, separated by (-)
7	Subdomain	{-1,0,1}	Using Sub Domain with Multiple Subdomains
8	SSLfinal State	{-1,1,0}	HTTPS (Hyper Text Transfer Protocol with Secure Sockets Layer)
9	Length of Domain Registration	{-1,1}	Length of Domain Registration
10	Favicon	{1,-1}	Favicon
11	Port	{1,-1}	Bypassing a Standard Port
12	HTTPS Token	{-1,1}	The presence of a "HTTPS" Token in the URL's Domain
13	URL Request	{1,-1}	URL Request
14	the Anchor's URL	{-1,0,1}	the Anchor's URL
15	Links in tags	{1,-1,0}	Links in <Meta>, <Script> and <Link> tags
16	SFH	{-1,1,0}	Server Form Handler (SFH)
17	Email Submission	{-1,1}	Information Submission via Email
18	Unusual URL	{-1,1}	Unusual URL
19	Redirect	{0,1}	Website Forwarding
20	When a Mouseover	{1,-1}	Customizing the Status Bar
21	RightClick	{1,-1}	Deactivating Right Click
22	Window pop-up	{1,-1}	Utilizing Pop-Up Windows
23	Iframe	{1,-1}	IFrame Redirection
24	Domain's Age	{-1,1}	Age of Domain
25	DNSRecord	{-1,1}	DNS Record
26	Web Traffic	{-1,0,1}	Website Traffic
27	Page Rank	{-1,1}	PageRank
28	Google Index	{1,-1}	Google Index
29	Links pointing to page	{1,0,-1}	Number of Links Pointing to Page
30	Statistical report	{-1,1}	Statistical-Reports Based Feature
31	Result	{-1,1}	Phishing Website Detection Result
Attributes based on the Address Bar (1–12), attributes based on Abnormality (13–18), attributes based on HTML and JavaScript (19–23), and attributes based on the Domain (24–30).			

3.2 Features Selection

From a website, a lot of things can be found to mark the difference between phishing sites and original sites, with the quality of the features being of the utmost importance for the success of phishing detection mechanisms. In the phishing websites dataset, 30 key features have been identified as efficient and influential in predicting phishing and legitimate websites, these features are from the phishing websites dataset, which is made publically accessible via the UCI Machine Learning Repository [17]. categorized into four groups: address bar-based absolute options (12 attributes), functions with irregular bases (6 attributes), HTML and JavaScript-based options (5 attributes), and domain-based options (7 attributes). Table 1, provides a detailed

breakdown of these attributes along with their corresponding column names, offering insight into the features utilized for phishing website classification. Feature selection is actually a part of the whole problem involving filtering and wrapper methods. This procedure is to search for and figure out the most relevant characteristics from the dataset in order to upgrade the accuracy of predictive models as portrayed in fig.3. Furthermore, PSO, Bee Colony Optimization (BEE), IWSS, Ranker Search Gain Ratio (GAIN), PCA are among some of the feature selection methods that were implemented to enhance the accuracy of the machine learning algorithm. These algorithms methodically score subsets of input data according to the features and then select those which are the most informative for phishing website classification.Because PCA can preserve variance while reducing dimensionality, a crucial feature for high-dimensional datasets, it was selected, however it may introduce biases by discarding features that might be relevant to phishing detection, and IWSS uses an iterative wrapper technique that ensures optimum feature relevance by evaluating feature subsets depending on classifier performance, it was chosen.

Attributes based on the Address Bar (1–12), attributes based on Abnormality (13–18), attributes based on HTML and JavaScript (19–23), and attributes based on the Domain (24–30).

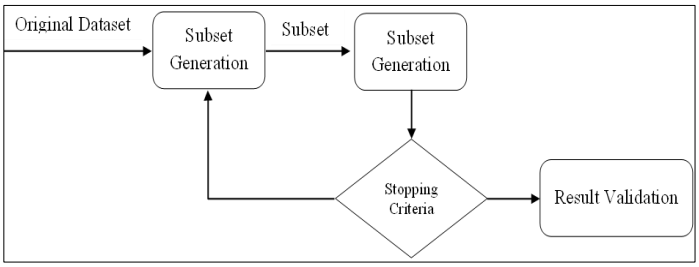

Figure 3: Feature selection working flow.

Figure 3, shows how the process feature selection works: The feature extract process is all about a systematic and iterative process to identify the most relevant features from an original dataset. It starts with preprocessing the raw dataset, normalizing features as needed. In the subset generation step, algorithms like IWSS or GAIN are used to construct candidate feature subsets based on how much they help the model perform. Each subset is assessed using performance metrics like accuracy or F-score. The iterative process continues until predefined stopping criteria are met, and the best-performing subset is selected for validation. The final result validation step ensures robustness through 10-fold cross-validation, with the final output being an optimal feature subset with documented metrics. This workflow balances efficiency and rigor, ensuring only the most discriminative features are retained while mitigating overfitting risks.Using algorithms such as IWSS or GAIN, subsets are (1) created, (2) assessed using metrics like accuracy/F-score, (3) stopped when no progress is seen, and (4) verified using 10-fold cross-validation.

4 Evaluation

Our objective is to showcase the efficacy of classification algorithms and underscore the significance of feature selection in enhancing their performance. Furthermore, we build a test dataset comprising phishing websites and employ 10-fold cross-validation to train the classifiers [18]. We also define the evaluation metrics used to gauge the quality of the classification algorithms.

4.1 Evaluation Measures

A confusion matrix, also known as an error matrix, provides insights into how well a classifier's rules perform on a set of test data. Table 2, presents the confusion matrix for binary classification, and we compute evaluation metrics based on it.

Table 2: Binary–class Confusion–Matrix.

	Predicted-Positive	**Predicted-Negative**
Phishing-Website	True-Positive (TP)	False-Negative (FN)
Genuine - Websites	False-Positive (FP)	True-Negative (TN)

True positives occur when the model correctly predicts positive cases (Phishing Websites) with actual positive values. True negatives represent correct predictions of negative cases (Genuine Websites) with actual positive values. False positives and false negatives occur when the model incorrectly predicts positive and negative cases, respectively.

Based on these, we compute several evaluation measures such as accuracy, error rate, true positive rate, true negative rate, positive predictive value, and F-score.

1. Accuracy or Correct Classification Rate: measures the proportion of correctly classified instances among all instances evaluated as shown in equation 1.

$$Accuracy=((TP+TN))/((TP+FP+FN+TN)) \quad (1)$$

2. Error–Rate or Incorrect Classification Rate: indicates the proportion of incorrectly classified instances in relation to the total instances assessed as shown in equation 2.

$$ErrorRate=((FP+FN))/((TP+FP+FN+TN) \quad (2)$$

3. True–Positive Rate (TPR)/Sensitivity/Recall: reflects the ability of a test to identify positive instances accurately as shown in equation 3.

$$TPR=TP/((TP+FN)) \quad (3)$$

4. True–Negative Rate (TNR) or Specificity: demonstrates the effectiveness of a test in correctly identifying negative instances as shown in equation 4.

$$TNR=TN/((TN+FP)) \quad (4)$$

5. Positive–Predictive Value (PPV) or Precision: indicates the proportion of instances classified as positive that are truly positive as shown in equation 5.

$$PPV=TP/((TP+FP)) \quad (5)$$

6. F–Score or F–Measure: provides a balanced evaluation of a classifier's precision and recall using their harmonic mean as shown in equation 6.

$$FScore=(2*precision*recall)/((precision+recall)) \quad (6)$$

4.2 Evaluation of Feature Search Algorithm

Five well-known feature selection methods are showing in Table 3, are used by us. The methods are evaluated by a multilayer perceptron classifier, and their performance is gauged using a number of metrics.

Table 3: Search Algorithm.

Search Algorithm	Short
Particle Swarm Optimization	PSO
Bee Search	BEE
Iterative Weighted Subset Selection	IWSS
Gain Ratio & Attribute Evaluation	GAIN
Principal Components Analysis	PCA

It can be seen in Table 4, that each feature selection method's accuracy is given separately and the graphical depiction is also shown in Figure 4.

Table 4: 10-Fold Cross Validation Evaluation Measure.

	TP-Rate	FP-Rate	Precision	Recall	F-Score
PSO	0.91	0.08	0.91	0.91	0.91
BEE	0.93	0.07	0.93	0.93	0.93
IWSS	0.94	0.06	0.94	0.94	0.94
GAN	0.93	0.07	0.93	0.93	0.93
PCA	0.99	0.002	0.99	0.99	0.99

Figure 4, depicts the accuracy of each feature selection method, showing PCA as the most effective, the dataset is reduced to only include essential features that are necessary for identifying malicious websites using the classifier algorithms.

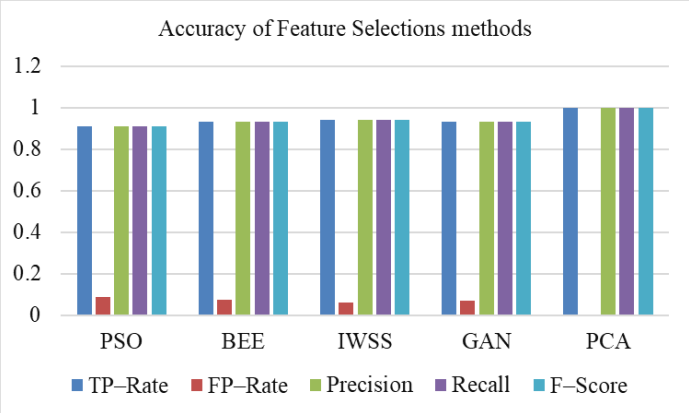

Figure 4: Accuracy of feature selections methods.

4.3 Machine Learning Classifier Training

Our objective is to precisely recognize phishing websites by training classifiers to predict which group belongs to the disguised dataset.

4.3.1 Classifiers Assessment

The training phase is the phase where a learning algorithm uses training data to build a classification model or classifier. For ensuring that the classifier is able to achieve the classification accuracy that we are expecting, the trained model needs to be tested on a separate dataset. If the classification accuracy is good enough on the testing dataset, the trained classifier can be used in real-life situations as well.

4.3.2 Classifier Performance based on Accuracy & Error Rate

All the 30 features are employed in our experiments. We are practicing 10-fold cross-validation technique. We use this method to run the experiment both in case of the training phase and to test. Experimentation showing a table of results in Table 5 and Figure 5, show the accuracy outcomes of the classification.

Table 5: Classifiers accuracy.

Classification Algorithms	10-fold Cross validation	Accuracy
Naive Bayes (NB)	92.9806	91.7684
Multilayer Perceptron (MLP)	96.9064	92.9444
AdaBoost (ADA)	92.5825	92.0850
Bagging (BAG)	96.2008	92.3564
Support Vector Machine (SVM)	93.8037	92.4921
Repeated Incremental Pruning to Produce Error Reduction (RIPPER)	95.0158	91.6327
Reduced Error Pruning Tree (REP)	95.3324	91.4066

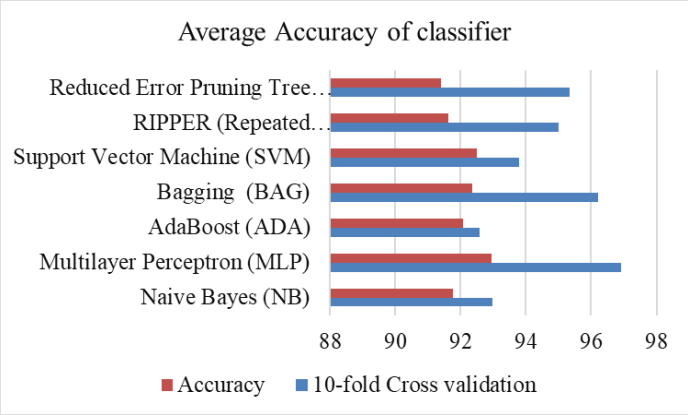

Figure 5: Average accuracy of classifier.

The Table 5 and Figure 5, inform that the Multilayer Perceptron (MLP) has incredible accuracy. Thus, because of the good performance of the Multilayer Perceptron (MLP), we give our preference to it for the detection of phishing.

4.4 Applying Principal Component Analysis (PCA) To Multilayer Perceptron (MLP)

By applying PCA to the dataset, the most contributive traits were found based on their contributions to the principal components and the initial set of 30 attributes. PCA, with decent feature selection, is effective in reducing the dimension of the feature space thus at the same time retaining as much variance as existed in the data.

The collection of the most relevant features through PCA and its associated high information carry was a result of the aggregate properties that covered several dimensions inclusive of the high variance of the features and the high discriminatory power in separating phishing and legitimate websites. These features likely include traits like the length of the URL, presence of HTTPS, domain age, presence of IP address, and other clues referred to as phishing attempts.

This will help the MLP classifier to take the optimal decision effectively within the feature-based training and classification process by providing the better and concise representation of the dataset after retaining important features using the PCA.

The discovery of these key features demonstrates the usefulness of PCA as a means of selecting features for machine learning data, especially when encountering datasets which have many features, both informative and irrelevant, with varying degrees of statistical importance with respect to the target variable.

Higher accuracy and improved computational effectiveness were generally observed for PCA and MLP combination in all the test cases in terms of accuracy and computational efficiency of the performance results. These same methods could be used to enhance the results of PCA and MLP, which are the most suitable solutions to increase the recall performance in the case of phishing attack detection.

We observed a significant boost in performance metrics after merging PCA-enhanced features to the MLP classifier as compared to the initial results of 10-fold cross-validation accuracy of 96.90% and 92.94%, respectively. The accuracies, then, rose to 97.5% and 94.5%, respectively, owing to the superior discriminatory information contained in the feature subset chosen by PCA.

Conclusion and Future Works

The general idea of this study is to enhance security measures regarding phishing, in which the machine learning models predict whether a specific website is either legitimate or phishing. We did this by comparing the accuracy of various classifiers using a publicly available dataset. Our results highlight that the appropriate choice of classifier is vital for phishing detection and the mitigation of related risks.

The experimental outcomes indicate that PCA achieves the best performance in terms of the among five feature selection techniques with F-score of 99%. Back to the classification algorithms, MLP offered the best results for 10-fold cross-validation and accuracy, and got 96.90% and 92.94% respectively.

By adding features improved by PCA to the MLP classifier, we saw a big improvement in how well it worked compared to the first test results. The first test had an accuracy of 96.90% and 92.94%, but with the PCA-improved features, the accuracy went up to about 97.5% and 94.5%.

In the future, we can try using PCA with other types of classifiers like CNNs. For real-time phishing detection, this approach may be included into business cybersecurity products, email security filters. The focus should be on explaining how PCA improves detection claims about PCA's potential with other classifiers should be tested with empirical benchmarks against deep learning-based feature selection methods. By seeing how PCA affects different classifiers that we haven't tested yet, we hope to find even better ways to spot phishing attacks. Using PCA more widely could give us a better understanding of phishing and help us create stronger ways to catch and stop these attacks.

Funding

None.

Acknowledgments

None.

Authors contribution

The authors were equally involved in the project's application, research strategy, outcome evaluation, and manuscript writing.

Conflict of interests

None.

References

[1] Basit, A., Zafar, M., Liu, X. et al. A comprehensive survey of AI-enabled phishing attacks detection techniques. *Telecommun Syst* **76**, 139–154, doi.org/10.1007/s11235-020-00733-2 (2021).

[2] Awlla, Ardalan Husin. A Hybrid Simulated Annealing and Back-propagation Algorithm for Feed-forward Neural Network to Detect Credit Card Fraud. *UHD J Sci Technol* **1**, 31–36, doi: doi.org/10.21928/uhdjst.v1n2y (2017).

[3] Jain, A. K. & Gupta, B. B. Phishing Detection: Analysis of Visual Similarity Based Approaches. *Secur Commun Netw* **2017**, 5421046, doi.org/10.1155/2017/5421046 (2017).

[4] Anti-Phishing Working Group (APWG). Phishing Activity Trends Report. https: apwg.org/trendsreports/ (Accessed: 23 May 2024).

[5] Karim, A. et al. Phishing detection system through hybrid machine learning based on. URL *IEEE Access* **11**, 36805–36822, doi: 10.1109/ACCESS.2023.3252366 (2023).

[6] Robert, C. Machine learning, a probabilistic perspective. *CHANCE* **27**, 62-63, doi.org/10.1080/09332480.2014.914768 (2014).

[7] Zhu, Erzhou, et al. DTOF-ANN: an artificial neural network phishing detection model based on decision tree and optimal features. *App Soft Comput* **95**, 106505, doi: 10.1016/j.asoc.2020.106505 (2020).

[8] Al Sukhni, B. et al. Investigating the Security Issues of Multi-layer IoT Attacks Using Machine Learning Techniques. *Centered Cogn Syst (HCCS)* 1–9, doi: 10.1109/HCCS55241.2022.10090400 (2022).

[9] Kumar, S. et al. Credit Card Fraud Detection Using Support Vector Machine. *Lect Notes Netw Syst* **237**, doi.org/10.1007/978-981-16-6407-6_3 (2022).

[10] Barraclough, P. A. et al. Parameter optimization for intelligent phishing detection using Adaptive Neuro-Fuzzy. *Int J Adv Res Artif Intell* **3**, doi: 10.14569/IJARAI.2014.031003 (2014).

[11] Fehringer and P. A. Barraclough. Intelligent Security for Phishing Online using Adaptive Neuro Fuzzy Systems, *Int J Adv Comput Sci Appl* **8,** doi: 10.14569/IJACSA.2017.080601 (2017).

[12] Nawafleh, Sahem, and Wa'el Hadi. Multi-class associative classification to predicting phishing websites. *Int J Acad Res Part* A **4**, 68–73, doi: 10.7813/2075-4124.2012/4-6/A.43 (2012).

[13] Tiwari, P. & Singh, R. R. Comparison of Classifier for Anti-Phishing Techniques. *Int J Eng Trends Technol* **34**, 53-58, doi:10.14445/22315381/IJETT-V34P258 (2016).

[14] He, M. et al. An efficient phishing webpage detector. *Expert Syst Appl* **38**,12018-12027, doi: 10.1016/j.eswa.2011.01.046 (2011).

[15] Nguyen, H. H. & Nguyen, D. T. Machine learning based phishing web sites detection. *AETA 2015 Recent Adv Electr Eng Relat Sci* 123–131, doi.org/10.1007/s11042-023-14731 (2016).

[16] Mohammad, R. M., Thabtah, F. & McCluskey, L. Predicting phishing websites based on self-structuring neural network. *Neural Comput Appl* **25**, 443–458, doi:10.1007/s00521-013-1490-z (2014).

[17] Mohammad, R., Thabtah, F., & McCluskey, L. Phishing website dataset. *UCI Mach Learn Repos* doi.org/10.1016/j.eswa.2022.118010 (2022).

[18] Peter, F., G. George, K. Mohammed, and U. B. Abubakar. Evaluation of classification algorithms on Locky ransomware using Weka tool. *Open J Phys Sci* **3,** 23-34, doi: 10.52417/ojps.v3i2.382 (2022).