Manuscript Submitted	12.10.2022
Accepted	28.12.2022
Published	31.12.2022

Phishing Websites Detection using Machine Learning Approaches

Raja Azlina Raja Mahmood & Tan Jun Ren
Department of Communication Technology and Network
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
raja_azlina@upm.edu.my
jun.ren.work@gmail.com

Abstract

Phishing is a form of fraud that attempts to obtain sensitive information via email, website, phone, or other forms of communication. The number of phishing attacks has increased significantly in recent years as more online services are being offered such as the online banking. The attackers design a phishing website, with similar appearance to the genuine website to steal victims' credentials account information that could lead to identity theft and financial loss. This study aims to detect phishing websites using supervised machine learning algorithms. Six classifiers which include Random Forest, K-Nearest Neighbors, Support Vector Machine, Decision Tree, Logistic Regression and Multilayer Perceptron have been implemented. The performance of the classifiers with 30 baseline features and different subsets of important features have been studied. In this study, a wrapper-based feature selection method was implemented to reduce the number of features to 15, 4 and 2 features respectively. The performance results show that Random Forest classifier using 30 features is the most accurate model to detect phishing websites with 97.41% of accuracy score, 97.14% of precision score, 98.25% of recall score and 97.69% of F1-score value respectively.

***Keywords*:** *Phishing websites detection, machine learning, feature selection.*

1. Introduction

Phishing attacks are fraud activities that can lead to severe losses to an individual or organization or even a country. Individuals are duped into disclosing sensitive and personal information either via phone, email, websites, mobile applications, or social media that can cause identity theft, financial loss and even mental health problems. Based on the 1st Quarter 2022 report by the Anti-Phishing Working Group, a total of alarming 1,025,968 phishing attacks has been observed (APWG, 2022) and it was the first time that the quarterly total attacks have exceeded one million. Phishing attacks can be classified into deceptive and technical-based methods with their techniques vary as shown in Figure 1.

Many solutions have been proposed in recent years to overcome and mitigate these phishing attacks (Miyamoto et al., 2008; Bin et al., 2010; Boddy, 2018; Chanti & Chitralekha, 2020), yet more efforts are still required to curb such variety of attacks. Our work focuses on detecting phishing websites, also known as spoofed websites, in order to prevent the attacks at early stages. With early detection, appropriate countermeasures can be taken to prevent more people fall victims to such attacks. The applications of machine learning have been growing very rapidly in many areas including cyber threats detection due to the highly accurate performance of the algorithms. In this study, few of the commonly used machine learning algorithms namely Random Forest, K-Nearest Neighbors, Support Vector Machine, Decision Tree, Logistic Regression and Multilayer Perceptron have been implemented to detect such websites. However, high feature dimensions of the websites data can affect the detection accuracy, due to the non-important and noisy features. Thus, a wrapper-based feature selection method

has been incorporated in our proposed system. The performance of machine learning algorithms is then analyzed.

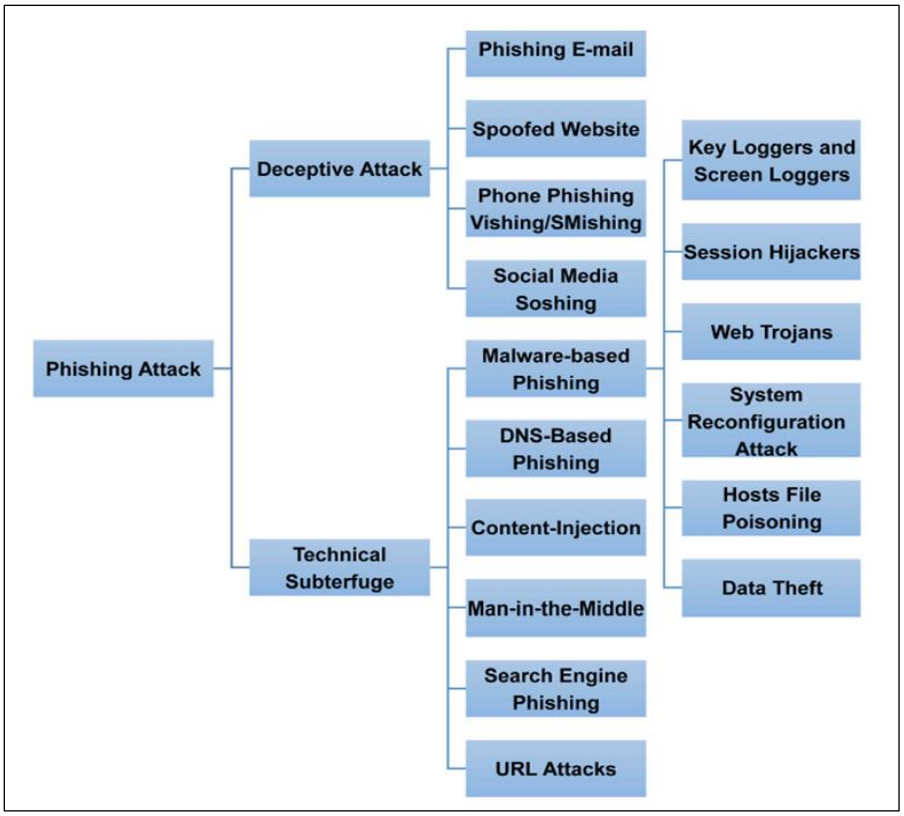

Figure 1: Phishing attack types and techniques (Alkhalil et al.,2021)

2. Feature Selection

The machine learning algorithms used in this work have been extensively reviewed (Safavian & Landgrebe, 1991; Breiman, 2001; Dhanda et al., 2019; Ray, 2019; Sarker, 2021) and hence will not be discussed here. These supervised machine learning algorithms, also known as classifiers, predict the new website as genuine or spoofed based on the training data sets that they have learnt from. The data sets contain features or attributes of a website, such as port information, length of an URL, domain's age and page's rank, defined by the domain experts. A total of 30 human-engineered features have been included in the UCI publicly available datasets by Mohammad et. al (2015). However, all these 30 features may not be significant or important in detecting phishing websites. Feature selection (FS) selects the most relevant features and hence minimizes the data dimensionality, reduces the model complexity, and also improves the performance of classifiers (Dash & Liu, 1997). It plays a vital part in data preprocessing and hence been used widely.

FS can be classified into three categories, which are the filter, wrapper, and hybrid (Miao & Niu, 2017). Filter methods ranks the features using statistical techniques such as correlation co-efficient between features and class labels. Wrapper methods incorporate the learning algorithm into the feature selection process. It follows the greedy search approach by evaluating all the possible combinations of features against the evaluation criterion (refer Figure 2 and Figure 3). Filter-based FS methods are faster and less computationally expensive than wrapper methods, however wrapper methods usually achieve better predictive accuracy than filter methods. Hybrid methods combine both filtering and wrapping methods to obtain the best of both techniques. A good example of a hybrid method when a Decision Tree paired with the Naïve Bayes classifier (Cateni et al., 2017). In this study, we implemented a wrapper-based

FS method, using Random Forest learning algorithm to reduce the number of features to 15, 4 and 2 features respectively.

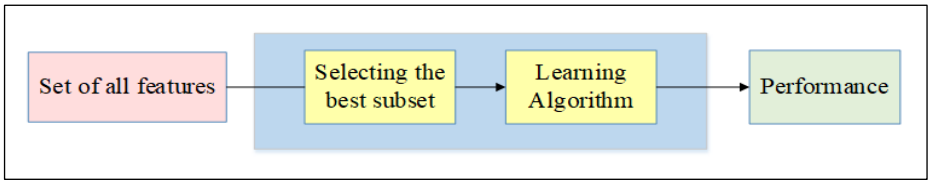

Figure 2: Filter method (Khalid et al., 2017)

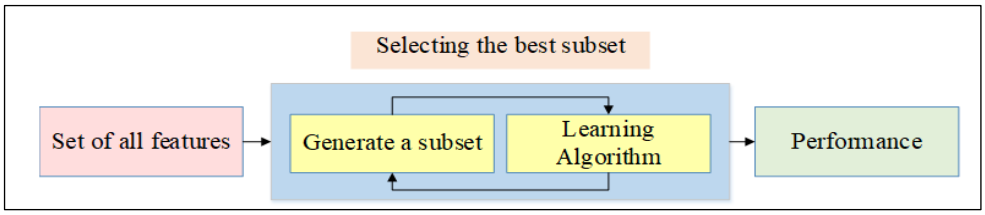

Figure 3: Wrapper method (Nolan & Lally, 2018)

3. Related Works

In this section, four studies that adopt machine learning to detect phishing websites were reviewed. All these studies use the same website phishing data set by Mohammad et. al (2015) available from the UCI repository. The data set contains information of 11,055 web sites, both legit and phishing websites, with 30 human-engineered features have been extracted from each website. Hutchinson (2018) and Zaini et al. (2020) used different sets of important features to improve the performance of their respective classifiers. Meanwhile Subasi et al. (2017) and Lokesh and BoreGowda (2020) trained the classifiers and detected the spoofed websites using the predetermined 30 features.

Hutchinson et al. (2018) improved the detection accuracy using a wrapper-based method with Random Forests classifier. The features were broken down into few different subsets to test the classifier. Set A consists of 4 web-presence related features, Set B consists of 22 features with only 2 possible outcomes (-1, 1), Set C consists of 8 features with 3 possible outcomes (-1, 0, 1), Set D consists of all 30 features and Set E consists of 17 features with their feature_importance values are 0.01 and above. We focus on Set D and Set E in our discussion. This work becomes our baseline work due to the detailed implementation information provided. The results show that set with 17 features outperformed the 30-feature set in metrics shown in Table 1.

Table 1: Performance comparison between different number of features

Number of features	Accuracy	Precision	Recall	F-Score	AUC
30 features	95.5%	95.59%	94.14%	94.86%	95.38%
17 features	96.5%	97.11%	94.79%	95.93%	96.30%

Zaini et al. (2020) implemented Random Forests, J48 Decision Tree, Multi-Layer Perceptron and K-Nearest Neighbors classifiers to detect the phishing websites. After training the classifiers with the 30 features, 15 significant features were then selected. The used feature selection method however was not stated in the paper. These features were trained and tested again with different classifiers. The results show all four classifiers achieved at least 93% accuracy with Random Forest has the highest accuracy of 94.79%.

Subasi et al. (2017) proposed an intelligent system based on Random Forest classifier to detect phishing websites. They implemented six classifier models with 10-fold cross validations, which include

Artificial neural Networks, K-Nearest Neighbors, Support Vector Machine, C4.5 Decision Tree, Random Forest and Rotation Forest. The performance of these classifiers was evaluated using accuracy, F-measure and Area under the Curve (AUC) metrics. Random Forest outperforms other classifiers in every evaluation metric, with 97.36% in accuracy, 97.4% in F-measure and 99.6% in AUC.
Lokesh and BoreGowda (2020) implemented Random Forest, K-Nearest Neighbor, Decision Tree, Linear Support Vector Classifier and Support Vector Machine in their phishing classification systems. Random Forest obtained the best accuracy score with 96.87%.

4. Methodology

Following Hutchison (2018) and Zaini et al. (2020), we aim to investigate the performance of the classifiers with different feature sets. We used Python scikit-learn machine learning library to perform the experiments including splitting the data, performing feature selection as well as training and evaluating Random Forest, K-Nearest Neighbors, Support Vector Machine, Decision Tree, Logistic Regression and Multilayer Perceptron classifiers. Figure 4 shows the detection framework used in our work.

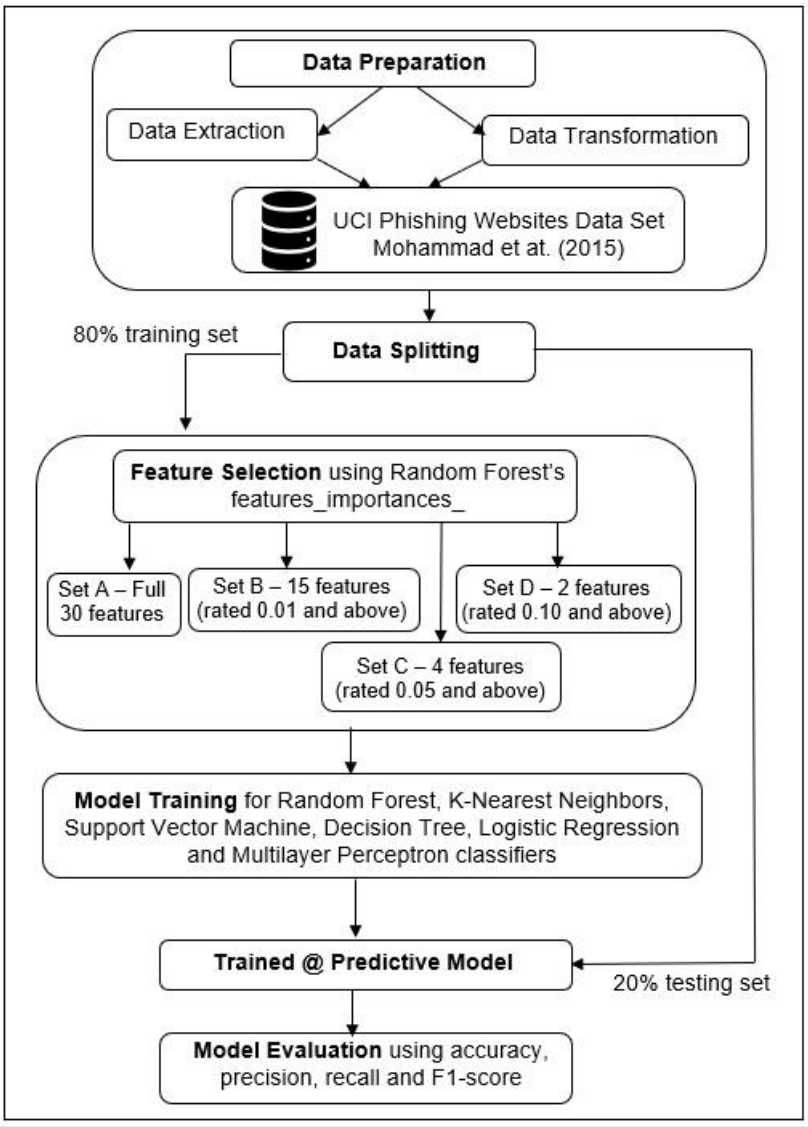

Figure 4: Proposed detection framework model

4.1 Data Preparation

The phishing website dataset from the UCI Machine Learning Repository by Mohammad et al. (2015) contains 11, 055 websites which include 6218 legit websites and 4837 phishing websites. These sites were gathered from multiple sources including the Phistank archive, MillerSmiles archive and Google search operators. A total of 30 features has been extracted from each website with each has been categorized either into 1) features based on address bar, 2) abnormal-based features, 3) HTML and and Javascript based features and 4) domain-based features. Figure 6 lists those 30 features provided by the UCI phishing websites data set. All the feature values have been transformed accordingly to be easily used for model training and prediction. Each feature is represented by either 1 to indicate phishing, -1 to indicate legitimate and 0 to indicate suspicious. The last column, the label column used 1 to represent phishing website and -1 to represent a legit website.

4.2 Data Splitting

The data set was split into training and testing sets for the supervised learning algorithms to learn from the labelled training data and hence able to predict the testing data correctly. All of the abovementioned researchers split the data set into 80:20 ratio and therefore, we implemented similar approach and randomly split the data set accordingly. We also used stratify parameter in the scikit-learn's train_test_split() function to ensure both sets contain approximately similar proportion of legit and phishing websites (refer Table 2).

Table 2: Statistics of the split data set

Label	Training (80%)	Testing (20%)	Total
Legit	4987	1231	6218
Phishing	3857	980	4837

4.3 Feature Selection

We selected the important features using the Random Forest's attribute called feature_importances_ that is readily available in scikit-learn module. This method returns the rank or values in an array of each feature's importance. We have run 5 different tests on the training data and average these feature importance values. Figure 5 shows the features' rank result. SSLfinal_State and URL_of_Anchor features outperformed others with their importance scores rate more than 0.24 score respectively. We then classified these features into different sets based on their scores.

We identified the following four feature sets to be used in our work: 1) Set A – contains full 30 features, 2) Set B – contains 15 features, with importance rated 0.01 and above, 3) Set C – contains 4 features, with importance rated 0.05 and above and 4) Set D – contains 2 features, with importance rated 0.10 and above. Figure 6 shows the list of features in each respective set.

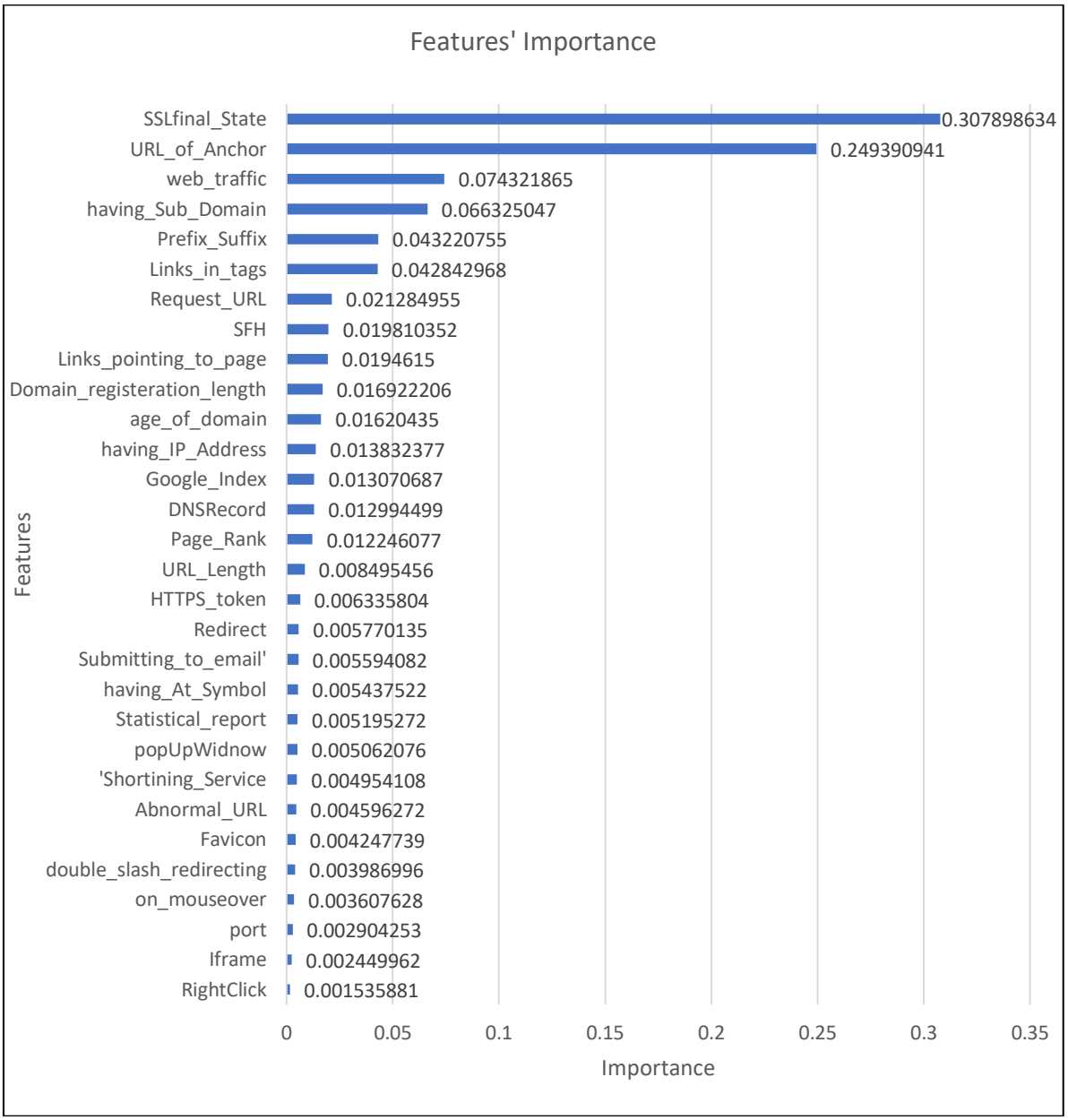

Figure 5: Feature importance rank and scores

No	Full Features Used	Selected Significant Features		
	Set A: 30 Features	Set B: 15 Features	Set C: 4 Features	Set D: 2 Features
1	having_IP_Address	having_IP_Address	having_Sub_Domain	SSLfinal_State
2	URL_Length	Prefix_Suffix	SSLfinal_State	URL_of_Anchor
3	Shortining_Serveice	having_Sub_Domain	URL_of_Anchor	
4	having_At_Symbol	SSLfinal_State	web_traffic	
5	double_slash_redirecting	Domain_registeration_length		
6	Prefix_Suffix	Request_URL		
7	having_Sub_Domain	URL_of_Anchor		
8	SSLfinal_State	Links_in_tags		
9	Domain_registeration_length	SFH		
10	Favicon	age_of_domain		
11	port	DNSRecord		
12	HTTPS_token	web_traffic		
13	Request_URL	Page_Rank		
14	URL_of_Anchor	Google_Index		
15	Links_in_tags	Links_pointing_to_page		
16	SFH			
17	Submitting_to_email			
18	Abnormal URL			
19	redirect			
20	on_mouseover			
21	RightClick			
22	popUpWindow			
23	Iframe			
24	age_of_domain			
25	DNSRecord			
26	web_traffic			
27	Page_Rank			
28	Google_Index			
29	Links_pointing_to_page			
30	Statistical_report			

Figure 6: List of feature sets

4.4 Model Training and Prediction

All the six classifiers have been trained with 80% of the data using full 30 features values and only Random Forest has been trained with different sets of feature values as well. Table 3 shows the configuration used in this experiment to train these classifiers. The trained models then predicted the results for the rest of 20% of the websites data set and their performance was then evaluated.

Table 3: Configuration parameters

Configuration Parameters	**Values**
scikit_learn classifiers modules	RandomForest(n_jobs=-1), KneighborsClassifier(n_neighbors=1), SVC(kernel='poly', probability=True), DecisionTreeClassifier(), LogisticRegression(), MLPClassifier(max_iter=1000)

4.5 Performance Evaluation

The performance of the classifiers was evaluated using accuracy, precision, recall and F1-score metrics. Table 4 lists the formulas used to calculate the respective measures using the following representations: True Positive (TP) that indicates the phishing websites been identified correctly, False Positive (FP) specifies the legit websites incorrectly identified as spoofed, True Negative (TN) indicates legit websites correctly classified and False Negative (FN) specifies spoofed websites incorrectly identified as legit.

Table 4: The classification measures

Classification Measure	Formula
Accuracy	$Accuracy = \frac{TP + TN}{TP + FP + TN + TP}$
Precision	$Precision = \frac{TP}{TP + FP}$
Recall	$Recall = \frac{TP}{TP + FN}$
F1-score	$F1 - score = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}}$

5. Results and Discussion

In this section, we assessed the performance of the different classifiers using the 30 predetermined features followed by the detection performance of Random Forest using different feature sets. Table 5 and Figure 7 show the overall scores and comparison graph of the six classifiers with Random Forest (RF) outperformed other classifiers in all metrics. RF obtained the highest accuracy score of 97.41% demonstrating its superior ability to classify genuine and phishing websites correctly. Moreover, the precision and recall values are both very high, with 97.14% and 98.25% respectively, hence show its good performance on the positive or phishing cases detection. Consequently, the weightage average of precision and recall values of RF, also known as F1-score is very high, with score of 97.69%. The Multilayer Perceptron has the second highest percentages after RF with 96.76% of accuracy score, 96.82% of precision score, 97.38% of recall score and 97.10% of F1-score value respectively. On contrary, Logistic Regression achieved the lowest percentages in all metrics, perhaps due to the non-linearity of the websites data used in the experiments. Finally, Decision Tree, K-Nearest Neighbors and Support Vector Machine classifiers performed moderately well in detecting phishing websites.

Table 5: Performance metrics of classifiers using 30 features

Classifier	Accuracy	Precision	Recall	F1-Score
Random Forest	97.41%	97.14%	98.25%	97.69%
Multilayer Perceptron	96.76%	96.82%	97.38%	97.10%
Decision Tree	96.45%	96.63%	96.99%	96.81%
K-Nearest Neighbors	96.07%	95.98%	96.99%	96.48%
Support Vector Machine	95.61%	94.97%	97.27%	96.11%
Logistic Regression	92.84%	92.53%	94.80%	93.65%

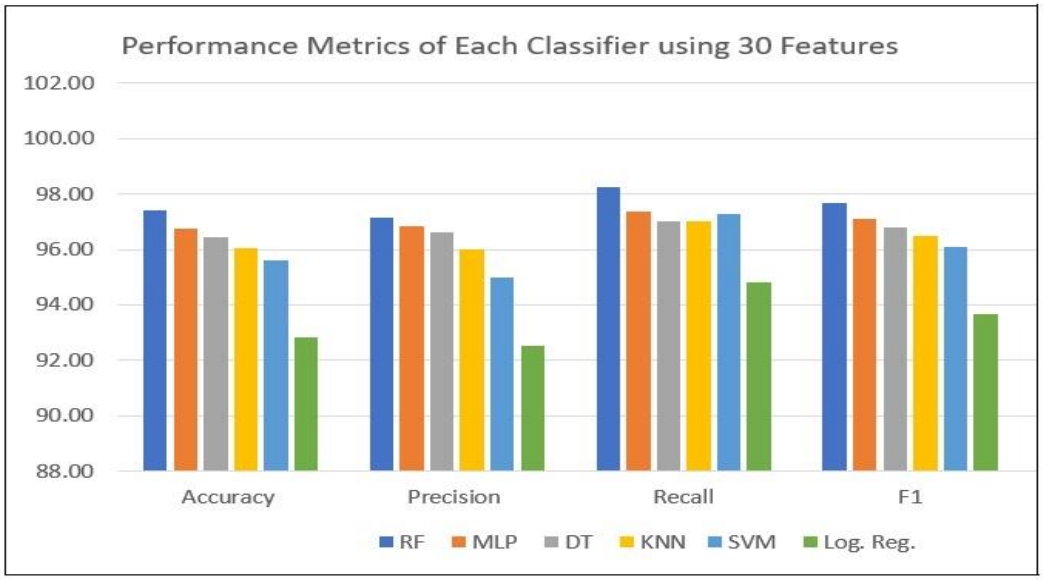

Figure 7: Performance metrics of classifiers using 30 features

The following discussion solely focuses on the performance of Random Forest (RF) classifier as it attained the best scores in the previous experiments, in comparison to other classifiers. The RF classifier was trained using different features and its performance is depicted in Table 6. Figure 8 shows that Set A or using full 30 features achieved the best scores followed by the reduced feature sets. Using 15 features with feature_importances_ threshold value set to 0.01, RF was also able to perform considerably good with 96.84% of accuracy score, 96.41% of precision score, 97.99% of recall score and 97.19% of F1-score value respectively. As the number of features reduced, the metrics scores also decreased proportionally. However, unlike Hutchinson (2018), their reduced feature set attained better results than that of using full feature set. Perhaps the feature_importances_ threshold value used in this experiment needs to be further fine-tuned to attain best possible results from the reduced feature sets. It is also worth noting that the scores between the full 30 features and reduced 15 features were close, with difference of 0.57% in accuracy score, 0.73% in precision score, 0.26% in recall score and 0.5% in F1-score value respectively. Finally, even with only 2 highly ranked features been used in Set D to create the predictive model, RF was able to perform considerably good with all the metrics scores attained were above 90%.

Table 6: Performance metrics of Random Forest classifier

Feature Set	Number of Features	Accuracy	Precision	Recall	F1-Score
Set A	30	97.41%	97.14%	98.25%	97.69%
Set B	15	96.84%	96.41%	97.99%	97.19%
Set C	4	92.97%	95.06%	92.17%	93.59%
Set D	2	91.55%	90.87%	94.30%	92.55%

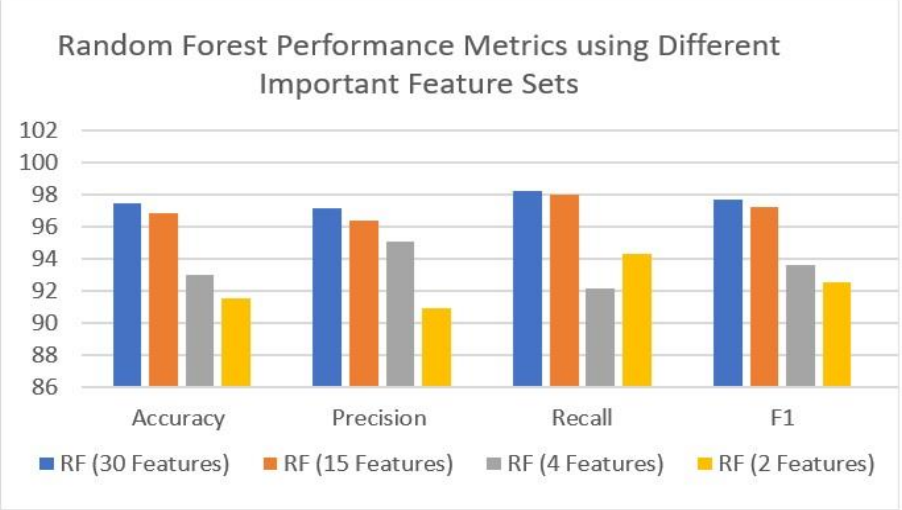

Figure 8: Performance metrics of classifiers using different feature sets

6. Conclusion and Future Recommendation

We implemented six different supervised machine learning algorithms to detect phishing websites using the publicly available UCI data set. A total of predetermined 30 features was used to train these classifiers and Random Forest (RF) achieved the best accuracy, precision, recall and F1-score results. Based on the obtained feature importance values, different subsets of features were then selected and used to train RF classifier. The classification results show that using full 30 features to train the classifier led to better performance scores than that of reduced feature sets, although with considerably small scores differences. The feature importance threshold values can be further improved to achieve better results in the future. A more robust feature selection method such as Recursive Feature Elimination (RFE) and application of cross-validation technique will be also considered in future work.

References

Alkhalil, Z., Hewage, C., Nawaf, L. & Khan, I. (2021). Phishing attacks: A recent comprehensive study and a new anatomy, Frontiers in Computer Science, 3, 3060–3389.

Bin, S., Qiaoyan, W., & Xiaoying, L. (2010). A DNS based anti-phishing approach. IEEE 2010 Second International Conference on Networks Security, Wireless Communications and Trusted Computing, Wuhan, 262–265.

Boddy, M. (2018). Phishing 2.0: The new evolution in cybercrime. Comput. Fraud Security, 8–10.

Breiman, L. (2001). Random Forests. Machine Learning, 45, 5–32.

Cateni, S., Colla, V., & Vannucci, M. (2017). A hybrid feature selection method for classification purposes. Proceedings - UKSim-AMSS 8th European Modelling Symposium on Computer Modelling and Simulation. Manchester, 39–44.

Chanti, S., and Chithralekha, T. (2020). Classification of anti-phishing solutions. SN Comput. Sci. 1, 11.

Dash, M. & Liu, H. (1997). Feature selection for classification. Intelligent Data Analysis, 1(3), 131–156.

Dhanda, N., Datta, S. S., & Dhanda, M. (2019). Machine learning algorithms. *Journal of Communications and Information Networks*, 210–233.

Hutchinson, S., Zhang, Z., & Liu, Q. (2018). Detecting phishing websites with Random Forest. Machine Learning and Intelligent Communications, 470–479.

Khalid, S., Khalil, T., & Nasreen, S. (2017). A survey of feature selection and feature extraction techniques in machine learning. Procedia Computer Science, 372–378.

Lokesh, G. H & BoreGowda, G. (2021). Phishing website detection based on effective machine learning approach, *Journal of Cyber Security Technology*, 5:1, 1-14.

Miao, J., & Niu, L. (2017). A survey on feature selection. Procedia Computer Science, 91, 919–926.

Miyamoto, D., Hazeyama, H., & Kadobayashi, Y. (2009). An evaluation of machine learning-based methods for detection of phishing sites. International conference on neural information processing ICONIP 2008: advances in neuro-information processing lecture notes in computer science. Editors M. Köppen, N. Kasabov, and G. Coghill (Berlin, Heidelberg: Springer Berlin Heidelberg), 539–546.

Mohammad, R. M. A., McCluskey, L., & Thabtah, F. (2015). UCI Machine Learning Repository: Phishing Websites Data Set. UCI Machine Learning Repository. https://archive.ics.uci.edu/ml/datasets/Phishing+Websites

Nolan, D. R., & Lally, C. (2018). Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model. *Journal of Computational Science*, 24, 132–142.

Ray, S. (2019). A quick review of machine learning algorithms. 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing. Faridabad, 35-39.

Safavian, S.R., & Landgrebe, D.A. (1991). Survey of decision tree classifier methodology. *IEEE Transactions on Systems, Man and Cybernetics*. 21, 660-674.

Sarker, I.H. (2021). Machine learning: algorithms, real-world applications and research directions. SN Comput. Sci. 2, 160.

Subasi, A., Molah, E., Almkallawi, F., & Chaudhery, T. J. (2017). Intelligent phishing website detection using random forest classifier. 2017 International Conference on Electrical and Computing Technologies and Applications (ICECTA), United Arab Emirates, 1-5.

Zaini, N. S., Stiawan, D., Razak, M. F. A., Firdaus, A., Wan Din, W. I. S., Kasim, S., & Sutikno, T. (2020). Phishing detection system using machine learning classifiers. *Indonesian Journal of Electrical Engineering and Computer Science*, 17(3), 1165.

APWG (2022). APWG phishing attack trends reports. 2021 anti-phishing work. Group, Inc. Retrieved 20 August, 2022, from https://apwg.org/trendsreports/