Journal of Statistics Applications & Probability
An International Journal

http://dx.doi.org/10.18576/jsap/130108

Website Phishing Detection Using Machine Learning Techniques

R. Alazaidah[1,], A. Al-Shaikh[2], M. R. AL-Mousa[2], H. Khafajah[3], G. Samara[1], M. Alzyoud[4], N. Al-Shanableh[4], and S. Almatarneh[3]*

[1]Department of Computer Science, Faculty of Information Technology, Zarqa University, Zarqa, Jordan
[2]Department of Cyber Security, Faculty of Information Technology, Zarqa University, Zarqa, Jordan
[3]Department of Data Science and Artificial Intelligence, Faculty of Information Technology, Zarqa University, Jordan
[4] Department of Computer Science, Faculty of Information Technology, Al al-Bayt University, Mafraq, Jordan

Received: 25 Nov. 2022, Revised: 4 Jan. 2023, Accepted: 15 Jan. 2023.
Published online: 1 Jan. 2024.

Abstract: Phishing is a cybercrime that is constantly increasing in the recent years due to the increased use of the Internet and its applications. It is one of the most common types of social engineering that aims to disclose or steel users sensitive or personal information. In this paper, two main objectives are considered. The first is to identify the best classifier that can detect phishing among twenty-four different classifiers that represent six learning strategies. The second objective aims to identify the best feature selection method for websites phishing datasets. Using two datasets that are related to Phishing with different characteristics and considering eight evaluation metrics, the results revealed the superiority of RandomForest, FilteredClassifier, and J-48 classifiers in detecting phishing websites. Also, InfoGainAttributeEval method showed the best performance among the four considered feature selection methods.

Keywords: Classification, Feature selection, Learning strategies, Machine learning, Phishing.

1 Introduction

Internet applications, cloud computing, and mobile computing have gained a tremendous momentum during this century [1, 2], making them important pillars of remarkable businesses that control the markets, with billions of users and devices that connect with each other to dominate the economy, and to create new, non-traditional business, investment, and employment opportunities. Consequently, the world is witnessing a pervasive increase access to e-commerce, on-line banking, and digital and social media[3], which proportionally increased phishing attacks accordingly [4].

Phishing attacks are social engineering crimes [5] in which users receive scam e-mails asking them to provide their credentials to a trust-worthy entity, which turns out to be a bad actor who disguised that trust-worthy entity to steal users' financial or personal information [6]. More than million phishing attacks have been recorded and observed in the last 3 months of 2022 according to the Anti-Phishing Work Group (APWG). The APWG numbers show that more than 23% of those attacks were against the financial sector.

In fact, phishing attacks represent more than 50% of all cybercrimes that target users [7]. Therefore, those attacks are of a high risk to several crucial modern sectors such as banks, payment systems, financial, webmail, and cloud storage [8]. The Federal Bureau of Investigation (FBI) categorizes business e-mail compromise (BEC) as "the most financially damaging online crimes", since they target businesses [9]. According to the APWG, BEC continued to be troublesome with a 0.59% increase compared to the second quarter of 2022.

Phishing detection techniques can be categorized into two main categories. The first is referred to as user awareness, which aims at training the users to identify e-mails as phishing or non-phishing. The second approach is referred to as software detection, which incorporates the use of blacklists, heuristics, visual similarity, and machine learning (ML) to detect phishing [10].

Machine Learning is a significant branch of computer science and artificial intelligence (AI) that attempts to imitate humans learning to gain significant hidden knowledge from data and use specific algorithms[11]. Practically, ML has been utilized in several domains such as medical diagnosis[12], malware prediction[13], weather forecasting[14], fraud detection[15], scene classification [16, 17, 18] and several other domains.

Practically, traditional methods to protect computer networks, systems, and users and to prevent attacks and intrusions are become less effective, due to the increasing number of cyberattacks as well as the changing nature of attacks that makes it harder for those traditional protection systems to function as desired [19].

In this context, this paper aims to contribute to the global effort of fighting phishing through utilizing the high capabilities of machine learning techniques in predicting phishing websites.

Two main objectives are considered in this paper. The first objective aims to identify the best classifier in predicting phishing

*Corresponding author e-mail: razaidah@zu.edu.jo

website. To achieve this objective, twenty-four different classifiers that belong to six well-known learning strategies have been selected for evaluation. The evaluation phase considers eight different popular metrics such as Accuracy, Precision, Recall, F-score, and several other metrics. Regarding this objective, another implicit objective is being considered. That is, to identify the best learning strategy among the six considered strategies and use four evaluation metrics: Accuracy, True Positive (TP), F-score, and ROC (Receiver Operating Characteristics) metrics.

The second objective considered in this paper is the identification of the best feature selection method to be used in the prediction of website phishing. To accomplish this task, four popular feature selection methods have been evaluated and compared using same classifiers in the first objective with respect to three evaluation metrics, namely, Accuracy, Precision, and Recall.

The rest of the paper is organized as follows: Section 2 surveys the most recent literature in the domain of utilizing ML techniques in phishing. Section 3 presents the methodology, results, and discussion. Section 4 concludes and suggests future directions.

2 Related Work

Jain and Gupta [20] introduced a ML-based approach to detect phishing attacks. The approach checks all hyperlinks in a website, analyses those links, and uses a logistic regression (LR) classifier to identify phishing websites. The approach utilizes features selection as an important step that aims to increase the prediction accuracy of the ML model that identifies phishing website based on 12 features. Accordingly, a website is either classified as phishing or legitimate. The approach achieved 98.42% accuracy, 98.8% precision, and 98.59% f1 score.

Random Forest classifier (RF) achieved 98.11% accuracy in the work that was conducted by Almseidin et al [21] to detect phishing using ML. The work was conducted on a dataset that contains 10 thousand phishing and legitimate Webpages that are divided evenly in the dataset. The dataset consists of 48 features; from which, only 20 were selected in the feature selection process. The work concluded that feature selection contributed to improving the accuracy of the detection approach.

Rashid et al. [22] proposed an approach to detect phishing Webpages using ML. Principal component analysis (PCA) was used as a feature selection approach to select the most appropriate features form the dataset that was already collected. The results showed that the support vector machine (SVM) classifier was the best in terms of accuracy which was 95.66%. The importance of using feature selection alongside the classification approach was also highlighted. Practically, PCA selected 5 features for the SVM classifier, and the result was an improved accuracy that surpassed the accuracies of the remaining approaches.

The performances of several ML approaches were compared by Gandotra and Gupta [23]using a 30-feature dataset that contains nearly 5000 phishing Webpages and around 6000 legitimate Webpages. The authors concluded that ML-based phishing detection models can be built faster when feature selection is used and, in the same time, the accuracy of the model is maintained. Their results showed that the random forest classifier (RF) achieved the best accuracy with or without using feature selection.

Lexical analysis of URLs was also used in conjunction with ML to detect phishing. The technique, which was introduced by Abutaha et al. [24], was proposed to be utilized as a web browser plug-in that alerts the users when they try to reach a webpage by analyzing the URL of that webpage. The technique was experimented on a dataset that contains more than a million of phishing and legitimate URLs. Initially, the technique extracts 22 features, which were then reduced to 10 selected features. The results showed that the SVM classifier achieved 99.89% accuracy that outperformed the other three classifiers, namely, RF, gradient boosting classifier (GBC), and neural network.

A framework to reduce the number of features was proposed by Wei and Sekiya [25]. The main target of their work was to find the minimal set of features while maintaining the phishing detection accuracy. In the beginning, 11 ML algorithms were compared, then RF was selected in terms of performance. It was concluded that only 14 features can be selected from the dataset that consists of 111 features while still achieving 97% accuracy. As a result, accuracy was not the only contribution of feature selection, memory usage was optimized, and the training time of the model was reduced.

The capabilities of seven ML approaches to detect phishing attacks were compared using three datasets by Mughaid et al. [26]. The first dataset is imbalanced and contains more than half a million instances 98% of which are legitimate and consists of 22 features. Nearly 17 thousand instances were selected from the dataset to extract a subset of almost evenly distributed phishing and legitimate instances. The accuracy of all the seven approaches was relatively low, where the boosted decision tree (BDT) achieved only 89%. The second dataset consists of 50 features and contains 10 thousand instances that are evenly distributed between phishing and legitimate instances. The best accuracy achieved when running the seven approaches on the second dataset was 100% from the BDT. The third dataset consists of 2500 ham and 500 spam emails, i.e., this dataset is with text features only. Averaged perceptron and neural network achieved the best accuracy this time with 99.7%.

The focus in the work of Yadav and Panda [27] was on extracting the features from a dataset that contains 1500 spam and ham e-mails. In the preprocessing phase, the e-mails were pared, and stemming, stop word elimination, and tokenization techniques were carried out. Then, feature selection resulted in 15 features that were selected and fed to the ML techniques, that showed the

superiority of the RF approach over DT and LR with more than 99% accuracy.

3 Research Methodology

In this section, the datasets, the methodology, the analysis of the 24 classifiers and the four feature selection methods are presented. Section 3.1 provides description for the two website phishing datasets that are used in this paper. Section 3.2 introduces the methodology, while Section 3.3 identifies the best classifiers. Section 3.4 evaluates and identifies the best feature selection method among four well-known methods. Section 3.5 discusses the main finding.

3.1 Description of Datasets

Two datasets are considered in this paper. The first dataset represents a binary classification dataset. It consists of 30 integer features with majority of these features are binary. The number of instances in this dataset is 11055. The second dataset represents a multiclass dataset with three class labels and consists of 9 integer-type features and 1353 instances. Table 1 lists the characteristics of both datasets. These datasets could be downloaded from UCI repository.

Table 1. Datasets characteristic

Name	Instances	Features	No. of Classes	Ref.
Dataset1	11055	30	2	[28]
Dataset2	1353	9	3	[29]

3.2 Methodology

Figure 1 depicts the methodology that is followed in this paper. The first step is the data collection step, where two website phishing datasets are collected and downloaded from UCI repository. More information regarding the considered datasets are provided in the following subsection. The second step is the pre-processing step which aims to prepare data for analysis through applying several tasks such as data cleaning, handling missing data, data reduction and several other tasks. It is noteworthy that the datasets that are used in this paper are well-formatted and do not need any pre-processing efforts.

The third step is the core of this paper which aims to identify the most appropriate classifier to use with website phishing datasets. This step considers several evaluation metrics such as Accuracy, True Positive (TP) ratio, False Positive (FP) ratio, Precision, Recall, and several other metrics as described in Section 3.3. Moreover, this step considers 24 different classifiers that belong to six well-known learning strategies. More description regarding these classifiers could be found also in Section 3.3.

The next step aims to identify the most appropriate feature selection method to use with website phishing datasets. This step considers four well-known feature selection methods which have been compared and evaluated based on three evaluation metrics, namely, Accuracy, Precision, and Recall. The final step is the discussion.

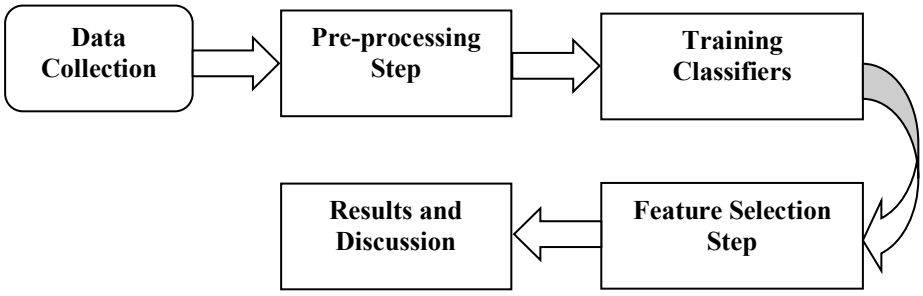

Fig. 1. Research methodology

3.3 Identifying the Best Classifier

In order to identify the best classifier that can effectively handle the considered datasets, 24 different classifiers that belong to six learning strategies have been evaluated and compared. These classifiers are categorized as follows:

- BayesNet [30], NaiveBayes [31], and NaiveBayesUpdateable [31] from Bayes learning strategy.
- Functions learning strategy is represented by four classifiers, namely, Logistic [32], MultilayerPerceptron [33], SimpleLogistic [34], and SMO [35].
- The lazy learning strategy is represented through IBK [36], KStar [37], and LWL classifiers [38].

- The meta learning strategy is represented by five classifiers, namely, AdaBoostM1[39], FilteredClassifier [33], LogitBoost [40], MultiClassClassifier [33], and RandomCommittee [33].
- Four classifiers are used to represent the rules learning strategy, namely, DecisionTable [41], JRip [42], PART [43], and ZeroR [33].
- Finally, the trees' learning strategy is represented by five classifiers, namely, DecisionStump [33], J-48 [44], LMT [33], RandomForest [45], and RandomTree [33].

All of these classifiers have been used with their default implementations in Waikato Evolutionary Knowledge Analysis (WEKA) which is a data mining studio that contains a collection of algorithms and tools that are used in the data analysis [33].

The evaluation phase of the 24 classifiers considers eight metrics, namely, Accuracy, TP rate, FP rate, Precision, Recall, F1-score, Matthews Correlation Coefficient (MCC), and Receiver Operating Characteristics (ROC)area. More information regarding these metrics could be found in [46]. These metrics are computed using the following equations:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$TP\ rate = \frac{TP}{TP + FN} \quad (2)$$

$$FP\ rate = \frac{FP}{FP + TN} \quad (3)$$

$$Precision = \frac{TP}{TP + FP} \quad (4)$$

$$Recall = \frac{TP}{TP + FN} \quad (5)$$

$$F_1 - score = 2 \times \frac{Precison \times Recall}{Precision + Recall} \quad (6)$$

The ROC metric is a graph that evaluates the performance of the classifier using all thresholds by plotting the FP rate at the x-axis and the TP rate at the y-axis.

Table 2 lists the performance of the 24 classifiers with the first input dataset. According to Table 2, Both FilteredClassifier from Meta learning strategy and J48 from Trees learning strategy have the best identical performance on the first dataset in terms of Accuracy and TP rate.

The same performance metrics for Dataset2 are measured and are listed in Table 3. Random Forest from Trees learning strategy showed the best results in terms of Accuracy and TP rate with 97.259% and 0.973 respectively. RandomForest is the second-best classifier on Dataset1 in terms of Accuracy and TP rate with a little difference from the best classifiers on Dataset1.

It is worth mentioning that for both Accuracy and TP rate, the higher the value, the best the performance of the classifiers. Therefore, these metrics should always be maximized.

Furthermore, according to Table 2, FilteredClassifier from the Meta learning strategy group of classifiers shows the best predictive performance on Datasets1 in terms of FP rate, 0.070, and Precision, 0.908. Table 3 lists the same performance metrics for Dataset2. RandomCommittee classifier from meta learning strategy group of classifiers shows the best FP rate, 0.029, while RandomForest from Trees learning strategy shows the best Precision result, 0.973.

It is worth mentioning that for the FP rate, the lower the value, the best the performance. While for Precision, the higher the value, the best the performance.

Table 2. Performance of the 24 classifiers using Dataset1 as the input dataset

Strategy	Classifier	Accurac	TP	FP	Precisio	Recall	F1	MCC	ROC
Bayes	BayesNet	84.331	0.843	0.118	0.820	0.843	0.828	0.727	0.948
	NaiveBayes	84.109	0.841	0.120	0.817	0.841	0.825	0.722	0.948
	NaiveBayesUpdateable	84.109	0.841	0.120	0.817	0.841	0.825	0.722	0.948
Function s	Logistic –R	85.735	0.857	0.106	0.842	0.857	0.847	0.755	0.962
	MultilayerPerceptron –L	88.766	0.888	0.085	0.888	0.888	0.888	0.801	0.959
	SimpleLogistic –I	85.809	0.858	0.110	0.835	0.858	0.841	0.752	0.961

	SMO –C	86.031	0.860	0.109	0.843	0.860	0.846	0.757	0.900
Lazy	IBk –K	88.322	0.883	0.085	0.884	0.883	0.883	0.796	0.952
	KStar –B	87.805	0.878	0.100	0.875	0.878	0.874	0.786	0.971
	LWL -U	81.744	0.817	0.161	0.836	0.817	0.873	0.726	0.943
Meta	AdaBoostM1 –P	81.744	0.817	0.161	0.836	0.817	0.873	0.726	0.900
	FilteredClassifier –F	**90.761**	**0.908**	**0.070**	**0.908**	**0.908**	**0.908**	**0.834**	0.960
	LogitBoost –P	85.514	0.855	0.118	0.827	0.855	0.833	0.742	0.957
	MultiClassClassifier –M	86.031	0.860	0.113	0.835	0.860	0.839	0.752	0.957
	RandomCommittee –S	89.061	0.891	0.080	0.893	0.891	0.891	0.806	0.943
Rules	DecisionTable –X	84.479	0.845	0.110	0.835	0.845	0.839	0.737	0.954
	JRip –F	90.244	0.902	0.070	0.904	0.902	0.903	0.826	0.933
	PART –C	90.022	0.900	0.074	0.901	0.900	0.900	0.823	0.957
	ZeroR	51.885	0.519	0.519	0.519	0.519	0.683	0.000	0.496
Trees	DecisionStump	81.744	0.817	0.161	0.836	0.817	0.873	0.726	0.818
	J48 –C	**90.761**	**0.908**	**0.070**	**0.908**	**0.908**	**0.908**	**0.834**	0.960
	LMT –I	89.357	0.894	0.079	0.894	0.894	0.894	0.813	**0.972**
	RandomForest –P	89.948	0.899	0.078	0.900	0.899	0.900	0.821	0.969
	RandomTree –K	87.435	0.874	0.092	0.877	0.874	0.875	0.778	0.915

Moreover, both FilteredClassifier and J48 show the best results on Dataset1 with 0.908 Recall and 0.908 F1-score, as listed in Table 2. For Dataset2, whose performance metrics are recorded in Table 3, RandomForest shows the best performance considering both Recall and F-score metrics. Both Recall and F1-score metrics should be always maximized.

Based on Table 2, FilteredClassifier and J48 show the best MCC result, 0.834, on Dataset1, while LMT classifier from Trees learning strategy shows the best ROC Area result, 0.972, on the same dataset.

For Dataset2, RandomCommittee from Meta learning strategy and RandomForest show the best results considering MCC metric, while KStar shows the best ROC Area result, 0.997, on the same dataset, as listed in Table 3. Both MCC and ROC Area metrics should be always maximized.

Table 4 summarizes the results in Tables 2 and 3 in order to identify the best classifier that could handle effectively the considered datasets with respect to the previously mentioned eight metrics. In Table 4, RF refers to RandomForest, FC refers to FilteredClassifier, RC refers to RandomCommittee, and KS stands for KStar.

Based on Table 4, it can be clearly concluded that FC and J-48 classifiers are the best choices to handle the considered website phishing datasets. The second-best choice is the RF classifier.

Table 3. Performance of the 24 classifiers using Dataset2 as the input dataset

Strategy	**Classifier**	**Accuracy**	**TP**	**FP**	**Precision**	**Recall**	**F-score**	**MCC**	**ROC Area**
Bayes	BayesNet	92.990	0.930	0.075	0.93	0.93	0.93	0.858	0.981
	NaiveBayes	92.981	0.930	0.076	0.93	0.93	0.93	0.858	0.981
	NaiveBayesUpdateable	92.981	0.930	0.076	0.93	0.93	0.93	0.858	0.981
Functions	Logistic –R	93.994	0.940	0.064	0.940	0.940	0.940	0.878	0.987
	MultilayerPerceptron –L	96.780	0.968	0.034	0.968	0.968	0.968	0.935	0.995
	SimpleLogistic –I	93.876	0.939	0.065	0.939	0.939	0.939	0.876	0.987
	SMO –C	93.804	0.938	0.066	0.938	0.938	0.938	0.874	0.936
Lazy	IBk –K	97.178	0.972	0.03	0.972	0.972	0.972	0.943	0.989
	KStar –B	97.196	0.972	0.032	0.972	0.972	0.972	0.943	**0.997**
	LWL -U	88.892	0.889	0.118	0.889	0.889	0.889	0.774	0.974
Meta	AdaBoostM1 –P	92.583	0.926	0.076	0.926	0.926	0.926	0.85	0.981
	FilteredClassifier –F	95.875	0.959	0.045	0.959	0.959	0.959	0.916	0.984
	LogitBoost –P	92.736	0.927	0.078	0.927	0.927	0.927	0.853	0.981
	MultiClassClassifier –M	93.993	0.940	0.064	0.94	0.94	0.94	0.878	0.987
	RandomCommittee –S	97.241	0.972	**0.029**	0.972	0.972	0.972	**0.944**	0.992
Rules	DecisionTable –X	93.242	0.932	0.075	0.933	0.932	0.932	0.863	0.979

	JRip –F	95.015	0.950	0.054	0.95	0.95	0.95	0.899	0.961
	PART –C	96.761	0.968	0.035	0.968	0.968	0.968	0.934	0.988
	ZeroR	55.694	0.557	0.557	0.557	0.557	0.715	0	0.5
Trees	DecisionStump	88.891	0.889	0.118	0.889	0.889	0.889	0.774	0.882
	J48 –C	95.875	0.959	0.045	0.959	0.959	0.959	0.916	0.984
	LMT –I	96.924	0.969	0.033	0.969	0.969	0.969	0.938	0.99
	RandomForest –P	**97.259**	**0.973**	0.03	**0.973**	**0.973**	**0.973**	**0.944**	0.946
	RandomTree –K	96.372	0.964	0.039	0.964	0.964	0.964	0.926	0.976

Table 4. Results summary

Dataset	**Accuracy**	**TP**	**FP**	**Precision**	**Recall**	**F-score**	**MCC**	**ROC**
Dataset1	FC	FC	FC	FC	FC	FC	FC	LMT
	J-48	J-48	J-48	J-48	J-48	J-48	J-48	
Dataset2	RF	RF	RC	RF	RF	RF	RF	KS

To get the complete picture, it has been decided to evaluate and identify the best learning strategy to handle the phishing datasets based on the performance metrics of the classifiers representing these learning strategies. Figures 2-5 depict the Average of the six learning strategies considered in this research with respect to the corresponding metrics and dataset.

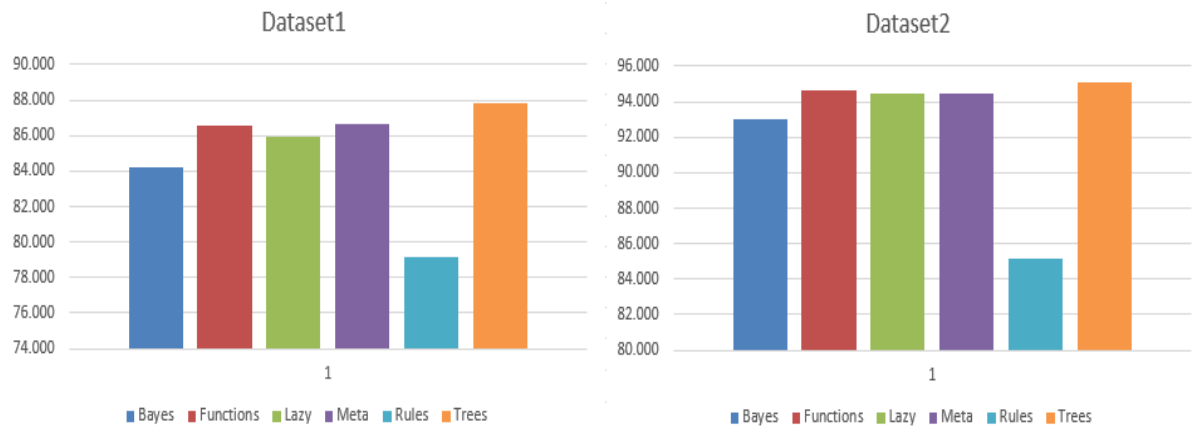

Fig. 2. Average Accuracy for the six learning strategies in the two considered datasets

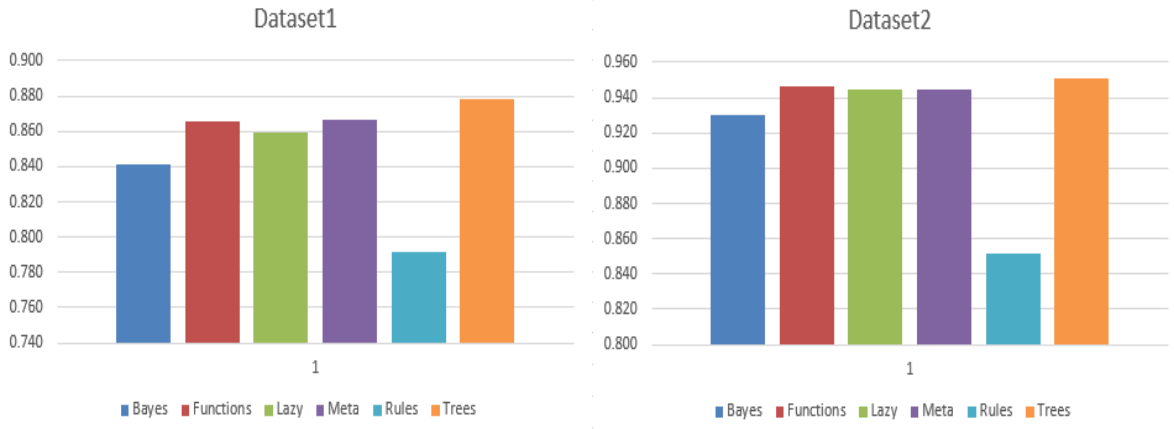

Fig. 3. Average TP rate for the six learning strategies in the two considered datasets

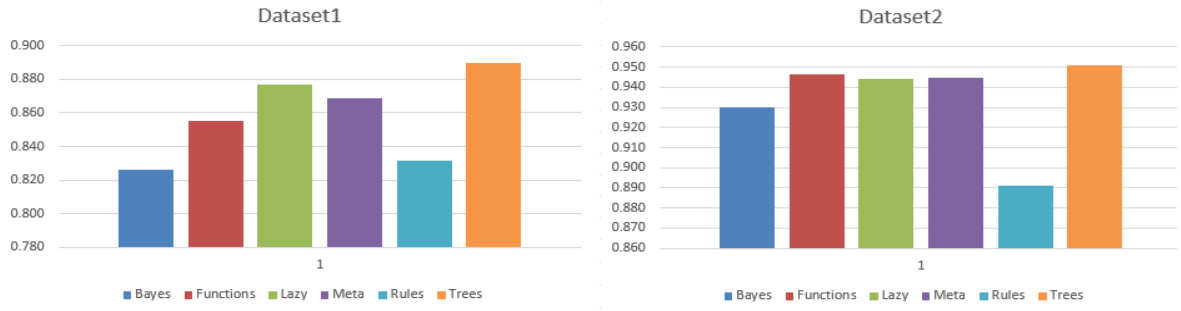

Fig. 4. Average F1-score for the six learning strategies in the two considered datasets

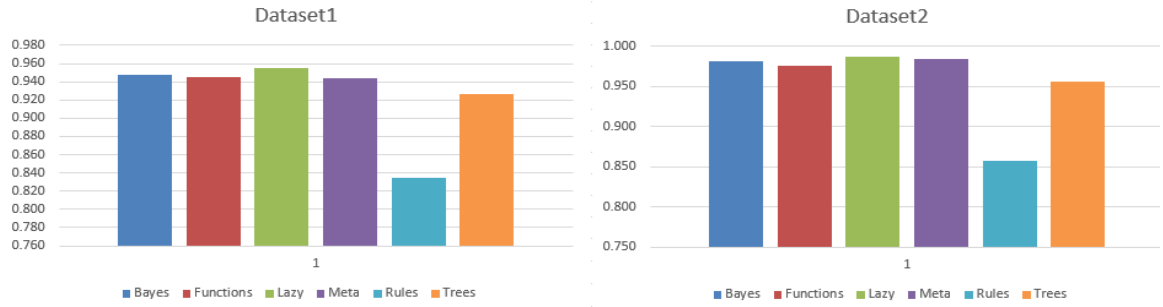

Fig. 5. Average ROC Area for the six learning strategies in the two considered datasets

According to Figures 2-4, the trees learning strategy shows the best performance on the two considered datasets with respect to Accuracy, TP rate, and F1-score. Figure 5 shows that the lazy learning strategy is the best strategy when there is a need to optimize the ROC metric. Moreover, the rules learning strategy shows the worst performance on the two datasets on almost all metrics. Therefore, it is highly recommended to avoid the rules learning strategy in the domain of phishing detection.

3.4 Identifying the Best Feature Selection Method

In order to identify the best feature selection method that suits the domain of websites phishing, four different feature selection methods have been chosen for comparison and evaluation using three evaluation metrics, namely, Accuracy, Precision, and Recall. These methods have been selected due to their popularity in the ML domain. The selected feature selection methods are InfoGainAttributeEval, GainRatioAttributeEval, CorrelationAttributeEval and CFsSubsetEval. All these feature selection methods have been used with their default settings as they have been implemented in WEKA.

Table 5 lists the evaluation results for the four feature selection methods considered in this paper using 50% of the features of Dataset1. The same 24 classifiers used in the previous section are considered in the evaluation phase of the feature selection methods. Also, three metrics are being considered. These metrics are Accuracy (A), Precision (P), and Recall (R).

Table 5. Evaluation results for the four feature selection methods

Classifier	**InfoGain**			**GainRatio**			**CorrelationAttribute**			**CfsSubsetEval**		
	A	**P**	**R**	**A**	**P**	**R**	**A**	**P**	**R**	**A**	**P**	**R**
BayesNet	92.764	0.928	0.928	92.782	0.928	0.928	92.745	0.928	0.927	92.637	0.927	0.926
NaiveBayes	92.791	0.928	0.928	92.782	0.928	0.928	92.745	0.928	0.927	92.646	0.927	0.926
NBUpdateable	92.791	0.928	0.928	92.782	0.928	0.928	92.745	0.928	0.927	92.646	0.927	0.926
Logistic	93.379	0.934	0.934	93.388	0.934	0.934	93.216	0.932	0.932	93.198	0.932	0.932
MLP	95.758	0.958	0.958	95.740	0.957	0.957	95.730	0.957	0.957	94.564	0.946	0.946
SimpleLogistic	93.388	0.934	0.934	93.351	0.934	0.934	93.189	0.932	0.932	93.143	0.931	0.931
SMO	93.541	0.935	0.935	93.324	0.933	0.933	93.261	0.933	0.933	93.288	0.933	0.933
IBk	96.119	0.961	0.961	95.830	0.958	0.958	95.984	0.960	0.960	94.491	0.945	0.945

KStar	96.138	0.962	0.961	95.984	0.960	0.960	96.110	0.961	0.961	93.912	0.940	0.939
LWL	89.010	0.890	0.890	88.973	0.890	0.890	88.973	0.890	0.890	89.335	0.893	0.893
AdaBoostM1	92.583	0.926	0.926	92.583	0.926	0.926	92.583	0.926	0.926	92.583	0.926	0.926
FilteredClassifier	95.423	0.954	0.954	95.631	0.956	0.956	95.323	0.953	0.953	94.310	0.943	0.943
LogitBoost	92.736	0.927	0.927	92.736	0.927	0.927	92.736	0.927	0.927	92.736	0.927	0.927
MultiClassClassifier	93.379	0.934	0.934	93.388	0.934	0.934	93.216	0.932	0.932	93.198	0.932	0.932
RandomCommittee	96.391	0.964	0.964	96.119	0.961	0.961	96.337	0.963	0.963	94.663	0.947	0.947
DecisionTable	93.089	0.931	0.931	92.999	0.930	0.930	55.694	0.557	0.557	93.053	0.931	0.931
JRip	94.365	0.944	0.944	94.527	0.945	0.945	94.401	0.944	0.944	93.650	0.937	0.936
PART	95.749	0.958	0.957	95.459	0.955	0.955	95.369	0.954	0.954	94.346	0.943	0.943
ZeroR	55.694	0.557	0.557	55.694	0.557	0.557	55.694	0.557	0.557	55.694	0.557	0.557
DecisionStump	88.892	0.889	0.889	88.892	0.889	0.889	88.892	0.889	0.889	88.892	0.889	0.889
J48	95.423	0.954	0.954	95.631	0.956	0.956	95.323	0.953	0.953	94.310	0.943	0.943
LMT	95.920	0.959	0.959	95.830	0.958	0.958	95.857	0.959	0.959	94.645	0.946	0.946
RandomForest	**96.581**	**0.966**	**0.966**	**96.255**	**0.963**	**0.963**	**96.355**	**0.964**	**0.964**	**94.772**	**0.948**	**0.948**
RandomTree	95.911	0.959	0.959	95.649	0.956	0.956	95.712	0.957	0.957	94.663	0.947	0.947

According to Table 5, RandomForest achieves the best results considering the three metrics and the four feature selection methods.

To better understand the results that are listed in Table 5, Table 6 summarizes the results that are presented in Table 5, considering the highest obtained results of the three-performance metrics on the four feature selection methods.

Table 6. Identifying the best feature selection method by determining the highest obtained results for the considered metrics

Method	**Accuracy**	**Precision**	**Recall**
InfoGainAttributeEval.	**96.581**	**0.966**	**0.966**
GainRatioAttributeEval.	96.255	0.963	0.963
CorrelationAttributeEval.	96.355	0.964	0.964
CFsSubsetEval.	94.772	0.948	0.948

Based on the results that are shown in Table 6, it can be clearly concluded that the highest results for the three metrics have been obtained using InfoGainAttributeEval method. Therefore, InfoGainAttributeEval is the best feature selection method among the four considered methods.

Table 7 shows the average of the 24 classifiers considering the three-evaluation metrics on the four feature selection methods.

Table 7. Identifying the best feature selection method by determining the highest performance average for the considered metrics

Method	**Accuracy**	**Precision**	**Recall**
InfoGainAttributeEval.	**92.409**	**0.924**	**0.924**
GainRatioAttributeEval.	92.347	0.923	0.923

CorrelationAttributeEval.	90.758	0.908	0.908
CFsSubsetEval.	91.724	0.917	0.917

According to Table 7, InfoGainAttributeEval method achieved the best average for the three metrics. Hence, it can be clearly concluded that InfoGainAttributeEval is the best feature selection method that suit the domain of websites phishing.

Accordingly, and based on the last two tables, InfoGainAttributeEval method is the optimal choice for feature selection when considering the datasets that are related to websites phishing.

3.5 Discussion

Two main objectives have been considered in this paper. Regarding identifying the best classifier to deal with the problem of phishing, three classifiers dominated. For Dataset1, FilteredClassifier and J-48 show the best results while RandomForest classifier showed the best results in Dataset2. In Dataset1, the number of instances is 11055, which is relatively high, compared with the number of features, that is, 30. Also, in Dataset1, there are only two class labels, and most of the features are binary, hence, the training phase is easier, and consequently, two classifiers showed high performance. RandomForest was among the best classifiers considering Dataset1.

For Dataset2, the total number of instances is 1353, which is relatively low compared with the total number of features, that is, 9. Also, Dataset2 contains 3 class labels. Therefore, the training phase in Dataset2 was much more complicated than the training phase in Dataset1. Therefore, RandomForest showed the best performance in Dataset2. Consequently, it can be concluded that RandomForest is more suitable to datasets with higher number of class labels and lower number of instances and features. In other words, RandomForest suits complex datasets better than any other classifier does. For datasets with higher number of instances and lower number of features, or datasets in which most of the features are binary, several classifiers may show high predictive performance.

For identifying the best feature selection method, it has been shown that this step could be crucial. Therefore, the features selection step must always be considered as a main step in any classification task. The InfoGainAttributeEval method showed a consistent high performance considering the three metrics, namely, Accuracy, Precision, and Recall. Therefore, it is recommended to use the InfoGainAttributeEval method in the websites phishing.

Finally, regarding the best learning strategies, and according to the evaluation results on both datasets, it can be concluded that the trees and meta learning strategies are the best choices to use with websites phishing datasets. Also, the rules learning strategy showed the worst performance considering the eight-evaluation metrics and the two datasets.

4 Conclusion and Future Work

In this paper, two main objectives have been achieved. The first is the identification of the best classifier that suits the domain of websites phishing, while the second is the identification of the best feature selection method in order to reduce the dimensionality of the datasets and thus to improve the performance. Regarding the first objective, three classifiers showed the best results: FilteredClassifier, J-48, and RandomForest. Considering the second objective, InfoGainAttributeEval method showed the best performance. Hence, it is highly recommended to consider an ensemble model that consists of the three best classifiers to solve the problem of website phishing as a future work. Metaheuristic algorithms can be used in the future to design feature selection algorithms with greater performance.

Conflicts of Interest Statement

The authors certify that they have NO affiliations with or involvement in any organization or entity with any financial interest (such as honoraria; educational grants; participation in speakers' bureaus; membership, employment, consultancies, stock ownership, or other equity interest; and expert testimony or patent-licensing arrangements), or non-financial interest (such as personal or professional relationships, affiliations, knowledge or beliefs) in the subject matter or materials discussed in this manuscript.

References

[1] R. Alazaidah, G. Samara, S. Almatarneh, M. Hassan, M. Aljaidi, & H. Mansur, Multi-Label Classification Based on Associations. Applied Sciences, **13**(8), 5081, (2023).

[2] M. Al-Khateeb, M. Al-Mousa, A. Al-Sherideh, D. Almajali, M. Asassfeha, & H. Khafajeh, Awareness model for minimizing the effects of social engineering attacks in web applications. *International Journal of Data and Network Science*, 7(2), 791-800. (2023).

[3] A. Al-Shaikh, R. Al-Sayyed, and A. Sleit, A Case Study for Evaluating Facebook Pages with Respect to Arab Mainstream

News Media, *Jordanian Journal of Computers and Information Technology*, **vol. 3**, no. 3, pp. 142-156, 2017.

[4] N. Q. Do, A. Selamat, O. Krejcar, E. Herrera-Viedma, & H. Fujita, Deep learning for phishing detection: Taxonomy, current challenges and future directions. *IEEE Access*, (2022).

[5] B. B. Gupta, N. A. Arachchilage, & K. E. Psannis, Defending against phishing attacks: taxonomy of methods, current issues and future directions. *Telecommunication Systems*, **67**, 247-267. (2018).

[6] M. Alluwaici, A. K. Junoh, W. A. AlZoubi, R. Alazaidah, & W. Al-luwaici, New features selection method for multi-label classification based on the positive dependencies among labels. *Solid State Technology*, **63**(2s). (2020).

[7] A. K. Jain, & B. B. Gupta, A survey of phishing attack techniques, defence mechanisms and open research challenges. *Enterprise Information Systems*, **16**(4), 527-565. (2022).

[8] O. K. Sahingoz, E. Buber, O. Demir, & B. Diri, Machine learning based phishing detection from URLs. *Expert Systems with Applications*, **117**, 345-357. (2019).

[9] H. F. Atlam, & O. Oluwatimilehin, Business Email Compromise Phishing Detection Based on Machine Learning: A Systematic Literature Review. *Electronics*, **12**(1), 42. (2022).

[10] M. R. Al-Mousa, Analyzing cyber-attack intention for digital forensics using case-based reasoning, *arXiv preprint,* arXiv:2101.01395. (2021).

[11] R. Alazaidah, M. A. Almaiah, & M. Al-Luwaici, Associative Classification in Multi-label Classification: an Investigative Study, *Jordanian Journal of Computers and Information Technology*, **vol. 7**, no. 2, pp. 166-179, (2021).

[12] M. S. A. Batah, M. Alzyoud, R. Alazaidah, M. Toubat, H. Alzoubi, & A. Olaiyat, Early Prediction of Cervical Cancer Using Machine Learning Techniques, *Jordanian Journal of Computers and Information Technology* , **vol. 8**, no. 4, pp. 357-369, (2022).

[13] H. Takci, F. Nusrat, & B. Women, Highly Accurate Spam Detection with the Help of Feature Selection and Data Transformation. *International Arab Journal of Information Technology*, **20**(1), 29-37. (2023).

[14] R. Alazaidah, F. K. Ahmad, & M. F. M. Mohsin, Multi label ranking based on positive pairwise correlations among labels. *The International Arab Journal of Information Technology*, **17**(4), 440-449. (2020).

[15] M. Alluwaici, A. K. Junoh, F. K. Ahmad, M. F. M. Mohsen, & R. Alazaidah, *Open research directions for multi label learning*. In 2018 IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE) (pp. 125-128). IEEE. (2018).

[16] R. Alazaidah, F. K. Ahmad, & M. F. M. Mohsen, A comparative analysis between the three main approaches that are being used to solve the problem of multi label classification, *International Journal of Soft Computing*, **12**(4), 218-223. (2017).

[17] R. Alazaidah, F. K. Ahmad, M. F. M. Mohsen, & A. K. Junoh, Evaluating conditional and unconditional correlations capturing strategies in multi label classification, *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, **10**(2-4), 47-51. (2018).

[18] F. Alshraiedeh, S. Hanna, & R. Alazaidah, An approach to extend WSDL-based data types specification to enhance web services understandability, *International Journal of Advanced Computer Science and Applications*, **6**(3), 88-98. (2015).

[19] I. N. Nti, O. Narko-Boateng, A. F. Adekoya, & A. R. Somanathan, Stacknet Based Decision Fusion Classifier for Network Intrusion Detection, *International Arab Journal of Information Technology*, **19**(3 A), 478-490. (2022).

[20] A. K. Jain & B. B. Gupta, A machine learning based approach for phishing detection using hyperlinks information, *Journal of Ambient Intelligence and Humanized Computing*, **10**, 2015-2028. (2019).

[21] M. Almseidin, A. A. Zuraiq, M. Al-Kasassbeh, & N. Alnidami, Phishing detection based on machine learning and feature selection methods, *International Association of Online Engineering*, Retrieved July 9, 2023, (2019).

[22] J. Rashid, T. Mahmood, M. W. Nisar & T. Nazir, *Phishing detection using machine learning technique*, In 2020 first international conference of smart systems and emerging technologies (SMARTTECH) *(pp. 43-46). IEEE*, (2020).

[23] E. Gandotra & D. Gupta, An efficient approach for phishing detection using machine learning, *Multimedia Security: Algorithm Development, Analysis and Applications*, 239-253. (2021).

[24] M. Abutaha, M. Ababneh, K. Mahmoud & S. A. H. Baddar, *URL phishing detection using machine learning techniques based on URLs lexical analysis,* In 2021 12th International Conference on Information and Communication Systems (ICICS) (pp. 147-152). IEEE, (2021).

[25] Y. Wei & Y. Sekiya, *Feature selection approach for phishing detection based on machine learning*, In International Conference on Applied CyberSecurity (pp. 61-70). Cham: Springer International Publishing (2021).

[26] S. K. Birthriya, P. Ahlawat & A. K. Jain, An Efficient Spam and Phishing Email Filtering Approach using Deep Learning and Bio-inspired Particle Swarm Optimization, *International Journal of Computing and Digital Systems*, **13**(1), 189-199. (2023).

[27] N. Yadav S. P. & Panda, Feature selection for email phishing detection using machine learning, In *International Conference on Innovative Computing and Communications: Proceedings of ICICC 2021*, **Volume 2** (pp. 365-378). Springer Singapore. (2022).

[28] A. Asuncion, UCI machine learning repository, university of california, irvine, school of information and computer sciences. http://www. ics. uci. edu/~ mlearn/MLRepository. html. (2007).

[29] N. Abdelhamid, A. Ayesh & F. Thabtah, Phishing detection based associative classification data mining. *Expert Systems with Applications*, **41**(13), 5948-5959. (2014).

[30] C. Zhang & P. Wang, *A new method of color image segmentation based on intensity and hue clustering*. In Proceedings 15th International Conference on Pattern Recognition. ICPR-2000 (**Vol. 3**, pp. 613-616), IEEE, (2000).

[31] P. Langley & G. H. John, *Estimating continuous distributions in bayesian classifiers*. In Proc. Uncertainty in Artificial Intelligence, (1995).

[32] S. L. Cessie & J. V. Houwelingen, Ridge estimators in logistic regression. *Journal of the Royal Statistical Society Series C: Applied Statistics*, **41**(1), 191-201. (1992).

[33] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann & I. H. Witten, The WEKA data mining software: an update, *ACM SIGKDD explorations newsletter*, **11**(1), 10-18. (2009).

[34] N. Landwehr, M. Hall & E. Frank, Logistic model trees, *Machine learning*, **59**, 161-205. (2005).

[35] J. Platt, Using analytic QP and sparseness to speed training of support vector machines, *Advances in neural information processing systems*, **11**. (1998).

[36] D. W. Aha, D. Kibler & M. K. Albert, Instance-based learning algorithms, *Machine learning*, 6, 37-66. (1991).

[37] J. G. Cleary & L. E. Trigg, *K*: An instance-based learner using an entropic distance measure*. In Machine Learning Proceedings 1995 (pp. 108-114). Morgan Kaufmann. (1995).

[38] E. Frank, M. Hall & B. E. Pfahringer, *Locally weighted naive bayes*, In Proceedings of the Conference on Uncertainty in Artificial Intelligence, (2003).

[39] Y. Freund & R. E. Schapire, *Experiments with a new boosting algorithm*, In icml (**Vol. 96**, pp. 148-156), (1996).

[40] J. Friedman, T. Hastie & R. Tibshirani, Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors), *The annals of statistics*, **28**(2), 337-407. (2000).

[41] R. Kohavi, *The power of decision tables*, In Machine Learning: ECML-95: 8th European Conference on Machine Learning Heraclion, Crete, Greece, April 25–27, 1995 Proceedings 8 (pp. 174-189). Springer Berlin Heidelberg. (1995).

[42] W. W. Cohen, *Fast effective rule induction*, In Machine learning proceedings 1995 (pp. 115-123). Morgan Kaufmann. (1995).

[43] E. Frank & I. H. Witten, Generating accurate rule sets without global optimization, Working paper 98/2). Hamilton, New Zealand (1998).

[44] J. R. Quinlan, Program for machine learning C4.5, (1993).

[45] L. Breiman, Random forests, *Machine learning*, **45**, 5-32. (2001).

[46] M. Alluwaici, A. K. Junoh & R. Alazaidah, New problem transformation method based on the local positive pairwise dependencies among labels, *Journal of Information & Knowledge Management*, **19**(01), 2040017. (2020).