futureinternet-14-00229.pdf

by

Submission date: 16-Dec-2022 02:44PM (UTC+0700)
Submission ID: 1982690986
File name: futureinternet-14-00229.pdf (1.22M)
Word count: 8981
Character count: 55889

future internet

MDPI

Article

CCrFS: Combine Correlation Features Selection for Detecting Phishing Websites Using Machine Learning

Jimmy Moedjahedy [1,*], Arief Setyanto [2,*], Fawaz Khaled Alarfaj [3] and Mohammed Alreshoodi [4]

[1] Computer Science Department, Universitas Klabat, Minahasa Utara 95371, Indonesia
[2] Magister of Informatics Engineering, Universitas AMIKOM Yogyakarta, Yogyakarta 55281, Indonesia
[3] Department of Computer and Information Sciences, Imam Mohammad Ibn Saud Islamic University, Riyadh 11564, Saudi Arabia; fkarfaj@imamu.edu.sa
[4] Unit of Scientific Research, Applied College, Qassim University, Buraydah 52362, Saudi Arabia; mo.alreshoodi@qu.edu.sa
* Correspondence: jimmy@unklab.ac.id (J.M.); arief_s@amikom.ac.id (A.S.); Tel.: +62-81244957223 (J.M.); +62-81316024569 (A.S.)

Abstract: Internet users are continually exposed to phishing as cybercrime in the 21st century. The objective of phishing is to obtain sensitive information by deceiving a target and using the information for financial gain. The information may include a login detail, password, date of birth, credit card number, bank account number, and family-related information. To acquire these details, users will be directed to fill out the information on false websites based on information from emails, adverts, text messages, or website pop-ups. Examining the website's URL address is one method for avoiding this type of deception. Identifying the features of a phishing website URL takes specialized knowledge and investigation. Machine learning is one method that uses existing data to teach machines to distinguish between legal and phishing website URLs. In this work, we proposed a method that combines correlation and recursive feature elimination to determine which URL characteristics are useful for identifying phishing websites by gradually decreasing the number of features while maintaining accuracy value. In this paper, we use two datasets that contain 48 and 87 features. The first scenario combines power predictive score correlation and recursive feature elimination; the second scenario is the maximal information coefficient correlation and recursive feature elimination. The third scenario combines spearman correlation and recursive feature elimination. All three scenarios from the combined findings of the proposed methodologies achieve a high level of accuracy even with the smallest feature subset. For dataset 1, the accuracy value for the 10 features result is 97.06%, and for dataset 2 the accuracy value is 95.88% for 10 features.

Keywords: feature selection; phishing detection; machine learning; correlation; feature elimination

check for updates

Citation: Moedjahedy, J.; Setyanto, A.; Alarfaj, F.K.; Alreshoodi, M. CCrFS: Combine Correlation Features Selection for Detecting Phishing Websites Using Machine Learning. *Future Internet* **2022**, *14*, 229. https://doi.org/10.3390/fi14080229

Academic Editor: Vijayakumar Varadarajan

Received: 16 June 2022
Accepted: 23 July 2022
Published: 27 July 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

Phishing is a sort of fraud that is one of the greatest threats on the Internet; it refers to false webpages that look and behave like actual webpages. Cybercriminals typically prey on individuals who lack essential digital security awareness through social engineering [1]. The objective is to dupe people into transmitting sensitive data such as their username, password, bank account number, or credit card number [2]. This form of crime can jeopardize users' and organizations' credibility and financial security; in some situations, phishing is the initial event that can lead to economic losses and more significant disruptions [3]; it can result in millions of dollars in losses every day [4]. According to the latest quarterly report from the Anti-Phishing Working Group (APWG) [5], since early 2020, the number of recent phishing assaults has more than doubled. As reported in the third quarter of 2021, the number of new unique phishing websites detected was 730,372, representing a rise of 30 percent from the second quarter of 2021. Moreover, between years, the number

of phishing websites increased. Meanwhile, according to Phislabs's data [6], during 2019 83.9% of phishing attacked services are financial, email, cloud, payment, and SaaS.

According to Abutair & Belghith [7], no single approach or strategy can detect all phishing websites ideally. The problem is that the website's content is liable to change, and the website's operational lifespan is limited. Machine learning (ML) is a promising and intelligent approach. This approach detects new phishing websites by analyzing a range of indicators, better known as features. Research on this topic attempts to implement several methods and algorithms using machine learning. The Hybrid Ensemble Feature Selection method uses a Cumulative Distribution Function gradient (CDF-g) algorithm to identify the automated feature cut-off rank.

Additionally, Chiew et al. [8] employ an ensemble technique coupled function perturbation. Due to this approach, Random Forest (RF) outperformed Naïve Bayes (NB), JRip, PART Classifier, and Support Vector Machines (SVM). With hybrid Natural language processing (NLP) and words vector-based features, RF receives the most excellent accuracy rating [9]. Furthermore, Nazz [10] uses a way that ranks the features based on the maximum variance using Principal Component Analysis (PCA). Next, Nazz applies RF, SVM, and Logistic Regression (LR) machine learning algorithms; the result is RF and SVM make better performance by accuracy. Each of these studies used a unique method for selecting features. Recent dataset comes with several features, and the assumption is some researchers use 50–100% of the total features, with complex computational time. For instance, Alotaibi and Alotaibi [11] used 23 from 48 features, Naaz [10] used all the features for 49, Hutchinson et al. [12] used 16 from 30 features, Karabatak and Mustafa [13] used 27 from 30 features, and Zaini et al. [14] used 15 from 30 features

This paper's hypothesis is that reducing low-impact features lead to simpler computation with an insignificant decline in the recognition rate for the web phishing dataset. In this paper, three scenarios of feature selection followed by machine learning classification. Hence, the objectives and contributions of this paper are to propose the optimal feature selection scenario for detecting phishing websites using machine learning. We observe feature selection, and recursive feature elimination by gradually decreasing and making new subset features. Additionally, this paper conducts a comparative analysis of three correlation methods combined with a machine learning algorithm: Spearman, Power Predictive Score (PPS), and Maximal Information Coefficient (MICe) with Total Information Coefficient (TICe) on selecting features from a dataset. Moreover, a performance comparison of four Random Forest (RF), Support Vector Machine (SVM), Decision Tree (DT) and AdaBoost algorithms on selected features is evaluated.

The remaining paper is organized as follows: the second section discusses material and methods, the result is presented in Section 3, Section 4 presents the discussion, and we draw our conclusion in Section 5.

2. Materials and Methods

2.1. Data

This study utilized two public datasets; Tan's [15] for the first dataset and Hannousse and Yahiouche [16] for the second dataset. We use this dataset because some studies on similar topics use it, such as Chiew et al. [8] on 2019, Khan et al. [17] on 2020, Dangwal and Moldova [18] on 2021, Al-Sareem et al. [19] on 2021, and Haynes et al. [20] on 2021.The first dataset selected phishing websites with the PhishTank and OpenPhish URLs and legitimate websites with the Alexa and General Archives URLs. For building the dataset, Tan gathered webpages in two different sessions between January and May and the other session in May and June for two years. The GNU Wget utility and Python programs automatically grab webpages from those two sessions. Apart from entire HTML texts, these datasets download associated resources—for example, images, CSS, and JavaScript—to guarantee that downloaded webpages appear correctly in the browser.

The downloaded dataset is further processed to eliminate broken webpages that failed to load or sites that returned an Error 404 in phishing and legal datasets. Additionally,

screenshots of each webpage are preserved for examination and filtering purposes. There are 48 features on this dataset, consisting of 10.000 data, balance for phishing and non-phishing, as seen in Table 1. The second dataset was built on 2021 and gathered URLs for the non-phishing dataset with Alexa and Yandex by crawling URLs from the top domain listed. URLs Phishing data are collected from PhisTank and OpenPhish by removing duplicates and URL that are not active anymore. Document Object Model is used for limited lifetime URLs and stores the data on a different file. There are 87 features on this dataset, and it consists of 11.430 data, balance for phishing and non-phishing. The dataset is divided into 70%, 30% for training and testing purposes.

Table 1. Dataset information.

Dataset/Category	Features	Total Data	Non-Phishing	Phishing	Published
Tan [15]/URL	48	10.000	5000	5000	2018
Hannousse and Yahiouche [16]/URL	87	11.430	5715	5715	2021

2.2. Correlation

According to Susanti et al. [21], correlation analysis is a statistical approach used to determine the strength of a relationship between two variables or features. Additionally, this analysis assesses the direction and degree of correlations between variables or features. There are various correlation measures, the first being Spearman's product-moment. Because it can quantify correlations on interval scales and ratios, this measure is sometimes referred to as parametric statistics. By comparison, other correlation coefficients, such as the spearman rank correlation coefficient, the Kendall correlation coefficient, and the gamma correlation coefficient, are referred to as nonparametric statistics.

Other research on correlation measurement includes those by D. N. Reshef et al. [22], who introduced a method called the Maximal Information Coefficient (MIC). As Simon & Tibshirani [23] observe, MIC is both popular and contentious; it asserted that MIC lacks state-of-the-art. Researchers enhanced and refined their study with Reshef et al. [24] and Reshef et al. [25], which is a combination approach of MIC and Total Information Coefficient (TIC). Their strategy resulted in establishing a library, which was explored in works by Albanese et al. [26]. Wetschoreck et al. [27] also proposed the Power Predictive Score (PPS) method for assessing the level of relationships. This method arose due to various problems with correlation; there are some cases of relationships that are not found when using conventional correlation methods. This method is open-source, and they provide a library that can be downloaded and used for free.

This study used correlation to determine the correlation between features as the preliminary stages in eliminating the feature. At this stage, the correlation value between two features is rated, and then ten features with a high correlation value between features are eliminated. This method, also known as Correlation Based Feature Selection (CSF), evaluates more than one attribute and analyzes the relationship between features. In CSF, a good feature combination has substantially linked characteristics with output but not with one another [28]. Eliminating correlation between features from existing features is expected to result in unique features. Whitley et al. [29] define multicollinearity as the presence of a strong relationship between two or more variable predictors. According to Senawi et al. [30], feature selection is a process that entails identifying the most meaningful subset of characteristics for targeted concepts by excluding unnecessary features. Besides that, it can result in decision-making errors, but it can also jeopardize the legitimacy of the conclusions or outcome. The most straightforward method for determining this is building and examining correlation matrix tables.

2.3. Feature Selection

Feature selection is an activity to remove irrelevant features; the selection of features may be accomplished in two methods. The first is to rank features according to several

criteria and then choose the top k features. The second approach is to select a minimal selection of characteristics without investigating performance degradation. In other words, the subset selection method may decide the number of factors to be picked automatically. In comparison, the feature rating algorithm must rely on a predefined threshold to determine the existence of a feature [31]. There are various types of feature selection, such as traditional, hybrid, and ensemble. Scikit, Boruta, MLFeaturesSelection, ITMO FS, ReBATE, MLxtend, Caret, and MLR are libraries that are frequently used for this type of study. According to studies by Pilnenskiy and Smetannikov [32], ITMO FS has superior and quicker performance.

This study chose the last ten features using Recursive feature elimination for the last stage. RFE is based on recursively developing a model by deleting features, modelling the model with the remaining features, and modelling the model's correctness. It is a greedy optimization technique for discovering the best subset of attributes and then ranking them based on their elimination period [33]. In other words, the goal behind RFE is to train a model using all of the features in the dataset and then delete one feature at a time using recursive train models. This procedure is continued until the dataset's components are depleted. After the process, it is possible to determine the relative value of each attribute [34].

2.4. Related Works

The studies described have a relationship with the study undertaken in feature selection. Similar to reference research by Chiew et al. [8], this study presents a novel framework for selecting Hybrid Ensemble Feature Selection (HEFS) features. The existing characteristics are then eliminated using HEFS, leaving ten baseline features. A test involving ten baseline features showed that random forest outperformed the accuracy of the other five machine learning algorithms, which was 94.6%. The subsequent study was conducted by Rao and Pais [35]. They proposed a classification model based on heuristic features taken from URLs, source code, and third-party services obtained from the analysis with eight machine learning algorithms. Random forest outperformed the accuracy value by 99.31% of the first experiment comparing eight machine learning methods. In the second experiment comparing principal component analysis (PCA-RF) with oblique Random Forests (oRFs), PCA-RF had a higher accuracy rating of 99.551%.

Salihovicd et al. [36] utilized the UCI phishing dataset to test the list of spam and phishing using six machine learning algorithms. They then conducted three software experiments: the first without changing the dataset, the second with BestFirst + CfsSubsEvaluation, and the third with Ranker + PrincipalComponents Optimization, WEKA. Random forest outperformed other machine learning algorithms on the first, second, and third trials with phishing accuracy scores of 97.26%, 94.77%, and 97.33%, respectively. Sahingoz et al. [9] researched the detection of phishing websites using random forests. Based on the results of testing seven algorithms, they discovered that only one algorithm, Random Forest combined with NLP, produced the best results, achieving an accuracy rate of 97.98%. A. Butnaru et al. [37] employed five supervised machine learning methods, including NB, DT, RF, SVM, and MLP, on a 100,315-dataset. The features utilized are a combination of ten features that are typically employed in this type of study and the two new features proposed by the authors. The hyperparameter tuning approach yields just three of the four features, namely RF, SVM, and MLP, with great performance values.

Alotaibi & Alotaibi [11] presented two approaches for selecting and eliminating features: consensus and majority voting. They use two datasets to evaluate the method. Consensus provided 17 features from the first dataset, while most votes yielded 23 features. For the second dataset, consensus yields three features, while majority voting yields thirteen. Voting is the approach with the highest accuracy, with a value of 98.63%. Haynes et al. [20] conducted similar research utilizing Artificial Neural Network (ANN), DT, Gaussian NB, Gradient Boosting (GB), KNN, RF, SVC, and SV. They also used three feature selection strategies, including URL and HTML based, URL based only, Transformer (BERT, ELEC-

TRA) NLP. The BERT and ELECTRA methods using ANN outperform others. Previous similar research was undertaken by Al-Sarem et al. [19]. They use a genetic algorithm (GA) to optimize the parameters of several ensemble machine learning algorithms, such as XGBoost, LightGBM, random forests, AdaBoost, GB, and Bagging. The optimized classifiers were then ranked, and the three top models were selected as the foundational classifiers for a stacking ensemble technique. The trials were conducted using two datasets of phishing websites. In the first dataset, the stacking ensemble approach yields an accuracy value of 97.16%, whereas, in the second dataset, the accuracy value is 98.50%.

Table 2 provides a brief description of the associated works. The research listed is related to the work done on feature selection and machine learning methods. According to the researchers' findings, the machine learning algorithm that outperforms others is RF [8,9,35,36]. As a result, the method employed in this work is the RF algorithm.

Table 2. Brief description of related works.

Authors/Year	Dataset	Number of Features	Machine Learning Algorithm	Feature Selection Algoritm	Accuracy
Rao & Pais [35]/2019	Build their own dataset using Phistank and Alexa	Not stated	RF, J48 Tree Decision, LR, Bayesian Network (BN), SVM, Sequential Minimal Optimization (SMO), AdaBoostM1, Multi-Layer Perceptron (MLP)	PCA-RF, obligate RF. PCF-RF performed the best accuracy	Experiment 1: 99.3% Experiment 2: 93.19%
Chiew et al. [8]/2019	Tan [15]	10	PART, JRip, RF, SVM, NB, C4.5	Hybrid Ensemble Feature Selection (HEFS), When HEFS is integrated with RF, the result outperforms another algorithm	94.6%
Salihovic et al. [36]/2019	UCIs Phishing Websites Data Set and Spam Emails Dataset	10 16	RF, K-Nearest Neighbor (KNN), SVM, LR, Artificial Neural Network (ANN), NB	BestFirst, CfsSubsEvaluation and Ranker, PrincipalComponents Optimization. Ranker + PrincipalComponents optimization using RF has the best accuracy	97.33% 94.24%
Sahingoz et al. [9]/2019	construct their own dataset	40	RF, DT, AdaBoost, SMO, NB, K-Star, KNN	Feature reduction mechanism using Natural Language Processing (NLP) features based. RF with NLP features outperform the other algorithms	97.98%
Alotaibi & Alotaibi [11]/2021	Tan [15]	23	SVM, NB, AdaBoost, LightGBM	Consensus and majority voting feature selection. Consensus and voting with Adaboost and LightGMB get the best result	98.63%

Table 2. *Cont.*

Authors/Year	Dataset	Number of Features	Machine Learning Algorithm	Feature Selection Algoritm	Accuracy
Haynes et al. [20]/2021	Shirazi Dataset	15	Artificial Neural Network (ANN), DT, Gaussian NB, Gradient Boosting (GB), KNN, RF, SVC, SVM	URL and HTML based, URL based only, Transformer (BERT, ELECTRA) NLP. BERT and ELECTRA using ANN outperform another method	More than 96%
Al-Sarem et al. [19] /2021	UCI dataset Tan [15] Mendeley dataset	Not stated	XGBoost, AdaBoost, RF, GradientBoost, Bagging, LightGBM	Ensemble, GA-based ensemble. GA-SVM outperforms another method	97.16%, 98.58%, 97.39%

2.5. Methods

The analytic approach of this method is removing features from the two datasets in two steps. Firstly, eliminate intercorrelation between features using three scenarios, Pearson, MICe TICe, and PPS, and remove ten characteristics with the highest correlation value ranking, the remaining features will process in the second round with recursive feature elimination (RFE). Secondly, removing other features gradually by the ranked using RFE, as shown in Figure 1. Then, the dataset is subjected to machine learning tests utilizing Random Forest (RF), Decision Tree (DT), AdaBoost, and Support Vector Machine (SVM). As shown by this methodology, three scenarios are conducted. The first scenario is PPS + RFE, the second scenario is MICe TICe + RFE and the third use Spearman + RFE.

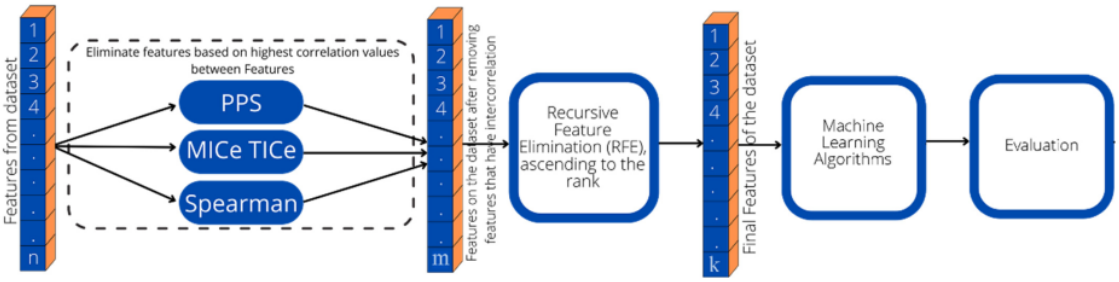

Figure 1. Feature selection and elimination approach from this method.

In Figure 1, n represents the number of full features, m the remaining features after the first removal using inter-features correlation, and k represents the remaining features after recursive elimination.

2.5.1. Scenario Using PPS + RFE

1. Step one: Choosing an algorithm; By default, this method computes results using a decision tree. A decision tree classifier will be used when obtaining data with categorical values. The decision tree regressor will then be used when numerical values are obtained. Data pre-processing: Use Label Encoding if the column whose value will be predicted or the target column has a definite value. However, if the column that denotes the alias value of the feature column has a substantial value, a one-hot encoding should be used. The metric value for evaluation: F1 score weights will be used as an evaluation metric if the chosen method is a classification method. F1 scores range from 0 to 1 on a scale of 1 to 10. The PPS value between two features

is rated at this stage, and then ten features with a high correlation value between features are eliminated.

2. Step two: After deleting ten features, there are 38 remainings for dataset 1 and 77 for dataset 2. The remaining characteristics are obtained by conducting a ranking search using RFE with python sci-kit-learn and selecting rankings 1 to 10 by gradually reducing features.
3. Step three: The highest-ranked features are evaluated for accuracy using the RF algorithm.

2.5.2. Scenario Using MIC_e TIC_e + RFE

1. Step one: Using the technique illustrated in Figure 2, search for MICe TICe values in 48 and 87 existing features. Additionally, the TICe value for each pair of x, y features was determined using *p*-values, followed by a twofold correction test to get the MICe value for the existing feature pair. At this stage, the MICe TICe value between two features is rated, and then ten features with a high correlation value between features are eliminated. Eliminating the ten features with the highest values has a negligible effect on the output feature. Because there are up to 10,000 samples, the parameter utilized is 0.45 [26].
2. Step two: After deleting ten features, there are 38 remainings for dataset 1 and 77 for dataset 2. The remaining characteristics are obtained by conducting a ranking search using RFE with python sci-kit-learn and selecting rankings 1 to 10 by gradually reducing features.
3. Step three: The highest-ranked features are evaluated for accuracy using the RF algorithm.

Figure 2. Procedures for finding MICe TICe values.

2.5.3. Scenario Using Spearman + RFE

1. The first step is to find Spearman correlation coefficients ranging from 0 to 1 for each of the 48 features in the dataset. Furthermore, the Spearman correlation value is determined using MICtools provided by [26]. Then, depending on the correlation value's results, issue ten features. The Spearman rank correlation coefficient is calculated by first rating the X elements in the paired sample data from 1 to n and then ranking the Y elements separately from one to n, assigning rank 1 to the minor component and n to the largest while retaining the original pairings. As a consequence, [rank (Xi), rank (Yi)] pairings are obtained. Then, compute a difference d equal to the difference between the X and Y variables' ranks for each pair. As mentioned in Equation (1), the test statistic, rs, is defined as a function of the sum of squares of these differences, d [38]. At this stage, the correlation value between two features is rated, and then ten features with a high correlation value between features are eliminated.

$$r_s = 1 - \frac{6\sum_{i=1}^{n} d_i^2}{(n^3 - n)} \quad (1)$$

2. Step two: After deleting ten features, there are 38 remaining features for dataset 1 and 77 for dataset 2. The remaining characteristics are obtained by conducting a ranking search using RFE and selecting rankings 1 to 10.
3. Step three: The ten highest-ranked features are evaluated for accuracy using the RF algorithm.

3. Results

3.1. PPS + RFE Scenario Result

After obtaining the correlation results using this method, ten features with a negligible influence on the output variable are eliminated. The value fragments are indicated in Table 3 for dataset 1 and Table 4 for dataset 2.

Table 3. PPS value between variables for dataset 1.

No	Variable 1	Variable 2	PPS Value
1	UrlLengthRT	UrlLength	1
2	NumHash	UrlLength	1
3	SubdomainLevelRT	SubdomainLevel	0.93
4	NumAmpersand	NumQueryComponents	0.84
5	NumHash	NumAmpersand	0.83
6	ExtMetaScriptLinkRT	PctExtResourceUrls	0.76
7	RightClickDisabled	PctNullSelfRedirectHyperlinks	0.7
8	PctExtNullSelfRedirectHperlinksRT	PctExtHyperlinks	0.69
9	UrlLengthRT	PathLength	0.68
10	RightClickDisabled	PctExtHyperlinks	0.65
11	PctExtResourceUrlsRT	PctExtResourceUrls	0.65
12	FrequentDomainNameMismatch		0.63
13	FrequentDomainNameMismatch	PctExtResourceUrls	0.63
14	AbnormalExtFormActionR	ExtFormAction	0.63
...	...	...	...
24	UrlLengthRT	NumDash	0.42
25	DomainInPaths	NumDash	0.42
26	SubdomainLevelRT	HostNameLength	0.42
27	PathLength	UrlLength	0.41
28	QueryLength	NumQueryComponents	0.41
29	RandomString	UrlLength	0.4

Table 4. PPS value between variables on Dataset 2.

No	Variable 1	Variable 2	PPS Value
1	domain_age	domain_in_brand	0.749341935
2	ratio_intHyperlinks	ratio_extHyperlinks	0.72978711
3	nb_eq	nb_and	0.720697676
4	nb_eq	nb_qm	0.702171885
5	longest_word_path	avg_word_path	0.655215372
6	shortest_word_host	nb_www	0.640106706
7	avg_word_path	longest_word_path	0.611298231
8	char_repeat	nb_www	0.607295021
9	longest_word_path	longest_words_raw	0.605856693
10	ratio_extHyperlinks	ratio_intHyperlinks	0.592877237
11	nb_subdomains	nb_dots	0.569833309
12	avg_word_host	longest_word_host	0.556752327
13	length_url	length_words_raw	0.55302947
14	length_words_raw	length_url	0.529359974
...	...	...	...
37	web_traffic	page_rank	0.411068239
38	nb_hyperlinks	empty_title	0.408773788
39	ratio_extHyperlinks	status	0.406332414
40	ratio_intHyperlinks	status	0.406029351
41	ratio_intHyperlinks	external_favicon	0.403947811
42	shortest_word_host	avg_word_host	0.400271455

The deleted feature is generated by ranking the results according to the PPS correlation value that is the highest. Then, as seen in Table 5, the deleted features from those findings and their effect on the output variable for dataset 1 are RightClickDisabled, NumHash,

SubdomainLevel, PctExtResourceUrls, UrlLength, SubdomainLevelRT, PathLength, NumAmpersand, NumQueryComponents, and PctExtNullSelfRedirectHyperlinksRT. For dataset 2, as seen in Table 6, the deleted features from those findings and their effect on the output variable for dataset 2 are ratio_extHyperlinks, shortest_word_path, web_traffic, char_repeat, nb_extCSS, domain_in_brand, external_favicon, domain_registration_length, links_in_tags, and ratio_intHyperlinks.

Table 5. List of removed features using PPS and their effect on the output variable for dataset 1.

Removed Features	Effect on the Output Variable
RightClickDisabled	0.0749
NumHash	0.048
SubdomainLevel	0.0431
PctExtResourceUrls	−0.0219
UrlLength	−0.0745
SubdomainLevelRT	−0.0758
PathLength	−0.0761
NumAmpersand	−0.0814
NumQueryComponents	−0.1474
PctExtNullSelfRedirectHyperlinksRT	−0.5405

Table 6. List of removed features using PPS and their effect on the output variable on dataset 2.

Removed Features	Effect on the Output Variable
ratio_extHyperlinks	0.0834
shortest_word_path	0.0744
web_traffic	0.0604
char_repeat	0.0147
nb_extCSS	−0.0836
domain_in_brand	−0.0982
external_favicon	−0.1466
domain_registration_length	−0.1617
links_in_tags	−0.1844
ratio_intHyperlinks	−0.244

3.2. MICe TICe + RFE Scenario Result

After obtaining the correlation results using this method, ten features with a negligible influence on the output variable are eliminated. The value fragments are indicated in Table 7 for dataset 1 and Table 8 for dataset 2.

Table 7. MICe TICe correlation value between features on dataset 1.

No	Variable 1	Variable 2	MICe TICe Value
1	UrlLength	UrlLengthRT	1.000
2	NumQueryComponents	QueryLength	0.638
3	PctExtResourceUrls	ExtMetaScriptLinkRT	0.550
4	PctExtResourceUrls	PctExtResourceUrlsRT	0.549
5	PctExtHyperlinks	PctExtNullSelfRedirectHyperlinksRT	0.508
6	UrlLength	PathLength	0.468
7	ExtFormAction	AbnormalExtFormActionR	0.461
8	PathLength	UrlLengthRT	0.422
9	NumQueryComponents	NumAmpersand	0.401
10	PctExtHyperlinks	FrequentDomainNameMismatch	0.336
11	NumAmpersand	QueryLength	0.320
12	PathLevel	PathLength	0.318
13	PctExtResourceUrls	FrequentDomainNameMismatch	0.307

Table 7. *Cont.*

No	Variable 1	Variable 2	MICe TICe Value
14	PctExtHyperlinks	PctExtResourceUrls	0.296
15	NumDots	SubdomainLevel	0.296
...	...	...	...
770	NumHash	InsecureForms	0.0006
771	TildeSymbol	ExtFavicon	0.0006
772	IpAddress	DomainInSubdomains	0.0006

Table 8. MICe TICe correlation value between features on dataset 2.

No	Variable 1	Variable 2	MICe TICe Value
1	nb_dots	nb_subdomains	0.93006
2	ratio_intHyperlinks	ratio_extHyperlinks	0.928648
3	shortest_word_path	longest_word_path	0.857185
4	shortest_word_path	avg_word_path	0.857185
5	longest_word_path	avg_word_path	0.857185
6	longest_word_host	avg_word_host	0.655094
7	shortest_words_raw	shortest_word_path	0.645462
8	length_url	length_words_raw	0.637978
9	nb_www	shortest_word_host	0.592451
10	nb_www	char_repeat	0.591922
11	longest_words_raw	longest_word_host	0.582465
12	ratio_intMedia	ratio_extMedia	0.556993
13	length_words_raw	longest_word_path	0.556662
14	length_words_raw	avg_word_path	0.545439
15	length_words_raw	shortest_word_path	0.542383
...	...	...	...
2511	nb_star	nb_redirection	0.000528
2512	path_extension	empty_title	0.000526
2513	nb_tilde	whois_registered_domain	0.000515

The deleted feature is generated by ranking the results according to the MICe TICe correlation value that is the highest. Then, as seen in Table 9, the deleted features from those findings and their effect on the output variable for dataset 1 are as follows: PathLevel, UrlLengthRT, PctExtResourceUrlsRT, PctExtResourceUrls, UrlLength, PathLength, QueryLength, NumAmpersand, NumQueryComponents, and PctExtNullSelfRedirectHyperlinksRT. For dataset 2, as seen in Table 10, the deleted features from those findings and their effect on the output variable for dataset 2 are are as follows: ratio_extHyperlinks, shortest_word_path, char_repeat, shortest_words_raw, nb_extCSS, nb_hyphens, ratio_extMedia, external_favicon, links_in_tags, and ratio_intMedia.

Table 9. List of removed features using MICe TICe and their effect on the output variable on dataset 1.

Removed Features	Effect on the Output Variable
PathLevel	0.2295
UrlLengthRT	0.1695
PctExtResourceUrlsRT	0.0521
PctExtResourceUrls	−0.0219
UrlLength	−0.0745
PathLength	−0.0761
QueryLength	−0.0805
NumAmpersand	−0.0814
NumQueryComponents	−0.1474
PctExtNullSelfRedirectHyperlinksRT	−0.5405

Table 10. List of features removed using MICe TICe and their effect on the output variable on dataset 2.

Removed Features	Effect on the Output Variable
ratio_extHyperlinks	0.0834
shortest_word_path	0.0744
char_repeat	0.0147
shortest_words_raw	−0.0394
nb_extCSS	−0.0836
nb_hyphens	−0.1001
ratio_extMedia	−0.1404
external_favicon	−0.1466
links_in_tags	−0.1844
ratio_intMedia	−0.1933

3.3. Spearman + RFE Scenario Result

After obtaining the correlation results using this method, ten features with a negligible influence on the output variable are eliminated. The value fragments are indicated in Table 11 for dataset 1 and Table 12 for dataset 2.

Table 11. Spearman correlation value between features on dataset 1.

No	Variable 1	Variable 2	Spearman Value
1	NumQueryComponents	QueryLength	0.959329
2	NumQueryComponents	NumAmpersand	0.750978
3	UrlLength	PathLength	0.724112
4	NumAmpersand	QueryLength	0.708677
5	PathLevel	PathLength	0.652628
6	UrlLength	NumNumericChars	0.564596
7	NumDash	PathLength	0.550183
8	NumDots	SubdomainLevel	0.542126
9	PctExtHyperlinks	PctExtResourceUrls	0.520385
10	UrlLength	NumDash	0.501986
11	PctExtResourceUrlsRT	ExtMetaScriptLinkRT	0.487458
12	PctExtResourceUrls	FrequentDomainNameMismatch	0.465619
13	SubdomainLevel	HostnameLength	0.459177
14	NumNumericChars	RandomString	0.457105
15	RelativeFormAction	AbnormalFormAction	0.429697
...	...	...	...
770	RandomString	PctExtResourceUrls	0.000528
771	NumUnderscore	MissingTitle	0.000107
772	PctExtHyperlinks	UrlLengthRT	0.000058

Table 12. Spearman correlation value between features on Dataset 2.

No	Variable 1	Variable 2	Spearman Value
1	nb_dots	nb_subdomains	0.981132
2	longest_word_path	avg_word_path	0.928796
3	nb_qm	nb_eq	0.916903
4	length_url	length_words_raw	0.907651
5	longest_word_host	avg_word_host	0.878764
6	length_url	longest_word_path	0.831591
7	ratio_intHyperlinks	links_in_tags	0.787153
8	longest_words_raw	avg_words_raw	0.756151
9	length_hostname	longest_word_host	0.748802
10	nb_percent	nb_space	0.748207
11	length_words_raw	longest_word_path	0.747163
12	shortest_word_path	avg_word_path	0.742056
13	nb_extCSS	external_favicon	0.737071

Table 12. *Cont.*

No	Variable 1	Variable 2	Spearman Value
14	google_index	status	0.731171
15	nb_slash	length_words_raw	0.720081
...	...	...	...
2511	page_rank	status	−0.546889
2512	nb_hyperlinks	status	−0.551603
2513	length_words_raw	shortest_words_raw	−0.621968

The deleted feature is generated by ranking the results according to the MICe TICe correlation value that is the highest. Then, as seen in Table 13, the deleted features from those findings and their effect on the output variable for dataset 1 are as follows: PathLevel, SubdomainLevel, NumNumericChars, PctExtResourceUrls, UrlLength, PathLength, QueryLength, NumAmpersand, NumQueryComponents, and NumDash. For dataset 2, as seen in Table 14, the deleted features from those findings and their effect on the output variable for dataset 2 are as follows: ratio_extHyperlinks, tld_in_path, shortest_word_path, nb_dslash, http_in_path, nb_underscore, nb_percent, char_repeat, nb_space, and shortest_words_raw.

Table 13. List of removed features using Spearman and their effect on the output variable on dataset 1.

Removed Features	Effect on the Output Variable
PathLevel	0.2295
SubdomainLevel	0.0431
NumNumericChars	0.0191
PctExtResourceUrls	−0.0219
UrlLength	−0.0745
PathLength	−0.0761
QueryLength	−0.0805
NumAmpersand	−0.0814
NumQueryComponents	−0.1474
NumDash	−0.3722

Table 14. List of features removed using Spearman and their effect on the output variable on dataset 2.

Removed Features	Effect on the Output Variable
ratio_extHyperlinks	0.08336
tld_in_path	0.07915
shortest_word_path	0.07436
nb_dslash	0.0726
http_in_path	0.07078
nb_underscore	0.03809
nb_percent	0.0281
char_repeat	0.01473
nb_space	−0.00419
shortest_words_raw	−0.03936

4. Discussion

The remaining feature in each scenario then tested for classification purposes using Machine Learning algorithms. Figures 3 and 4 show the performance of machine learning algorithms on various number of selected features on the first scenario using Power Predictive score (PPS) and Recursive Feature Elimination (RFE). Random forest (RF) consistently achieves the best performance of each features subset on both dataset (blue line). Therefore, the rest of the paper evaluated on random forest algorithms only. Our finding identify RF is the best performing algorithms as reported in [8,9,36]. Therefore, the rest of the experiment was carried out on random forest (RF).

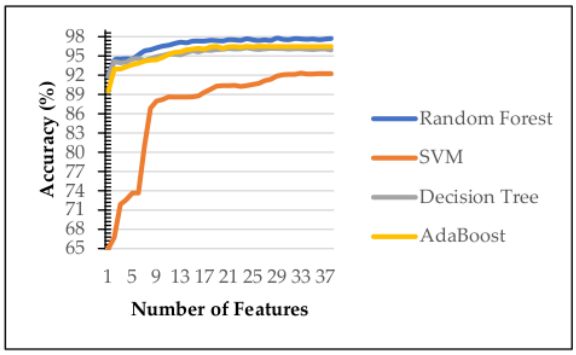

Figure 3. Comparison of various machine learning algorithms for dataset 1.

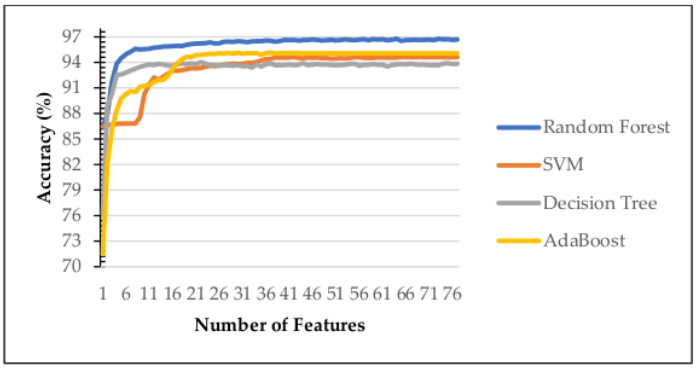

Figure 4. Comparison of various machine learning algorithms for dataset 2.

Table 15 shows the accuracy and execution time comparison of the selected features of each scenario for dataset 1. In light of the analysis of the accuracy value comparison in the graph in Figure 5, we can see that the accuracy using the Random Forest algorithm the accuracy value is maintained even with the reduction of features using three existing methods. The accuracies on the full features are achieved at 98.1%, and it declines insignificantly from 97% to 96% on 38, 30, and 20 features to 10 features. This data is in line with the hypothesis that reducing low-impact features leads to simpler computation with an insignificant decline in the web phishing dataset recognition rate.

Table 15. Feature subset accuracy and execution time on Dataset 1.

Features Subset	Accuracy			Execution Time (Second)		
	PPS + RFE	MICe + RFE	Spearman + RFE	PPS + RFE	MICe + RFE	Spearman + RFE
48 (full)	98.1%	98.1%	98.1%	0.983028	0.983028	0.983028
38 (correlation only)	98.16%	97.86%	97.86%	0.983681	0.978557	0.977316
30 (correlation + RFE)	97.96%	97.53%	97.83%	0.981095	0.972850	0.977302
20 (correlation + RFE)	97.63%	97.53%	97.6%	0.977835	0.974691	0.975958
10 (correlation + RFE)	96.96%	97.06%	96.63%	0.970703	0.969539	0.969281

Table 16 shows the accuracy and execution time comparison of the selected features of each scenario for dataset 2. In light of the analysis of the accuracy value comparison in

the graph in Figure 6, we can see that the accuracy using the Random Forest algorithm the accuracy value is maintained even with the reduction of features using three existing methods. The accuracies on the full features are achieved at 96.76%, and it declines insignificantly from 96% to 95% on 77, 60, 50, 40, 30, and 20 features to 10 features. This data is in line with the hypothesis that reducing low-impact features leads to simpler computation with an insignificant decline in the web phishing dataset recognition rate.

Table 16. Feature subset accuracy using Spearman on Dataset 2.

Features Subset	Accuracy			Execution Time (Second)		
	PPS + RFE	MICe + RFE	Spearman + RFE	PPS + RFE	MICe + RFE	Spearman + RFE
87 (full)	96.76%	96.76%	96.76%	0.969786	0.969786	0.969786
77 (correlation only)	96.93%	96.58%	96.32%	0.973809	0.968564	0.967281
60 (correlation + RFE)	96.79%	96.61%	96.73%	0.973166	0.970815	0.967000
50 (correlation + RFE)	96.61%	96.52%	96.35%	0.970255	0.967971	0.963443
40 (correlation + RFE)	96.61%	96.58%	96.38%	0.971377	0.968564	0.962919
30 (correlation + RFE)	96.47%	96.23%	96.38%	0.965171	0.963905	0.961832
20 (correlation + RFE)	95.97%	95.88%	95.82%	0.960969	0.959810	0.958136
10 (correlation + RFE)	95.88%	95.21%	95.04%	0.957109	0.950617	0.948853

As we can see in Tables 15 and 16, the accuracy is decrease as the number of features reduced, however the gap is small. For the first scenario—PPS + RFE, for example—the gap between full feature and the 10 features is less than 1% for reducing 78 features as shown on Table 16 on the second column. All experiments scenarios with two datasets shown no more than 2% gap of accuracy between full and 10 features set. The experimental result shows that reducing the features for binary classification (normal/phihing) class do not affect too much for the recognition ability of the model. Figures 5 and 6 also show that reducing the number of feature do not suffer to much on recognition rate.

To decide the class, machine learning algorithms do not need to calculate all the impact of each feature in input space. Not all features share the similar impact to the class label, low correlation between input variable and the class label indicate that the data less powerful to decide the class of the input. Removing low correlation input will help to speed up the classification process. This effect clearly presented in the execution time in Tables 15 and 16 where smaller number of features need smaller amount of time.

According to Tables 15 and 16, removing the 10 weakest features leads to a better recognition rate because involving features with low or negative correlation cause the model to learn from disrupting information. The proof that in both datasets, accuracy improved from 96.76 to 96.93 and 98.1 to 98.16 at the first removal of the 10 weakest features in dataset1 and dataset2, respectively. In the first removal of 10 features, correlation between a pair of features is evaluated. A pair of features with correlation indicates redundant information in input side. Removing one of a pair of corelated features will not suffer the recognition rate. Some of removed features with negative correlation to the target labels as shown in Tables 13 and 14 indicate that the features do not support the output class and therefore removing those features gives positive effect to the accuracy.

Three studies on this topic utilize the same dataset, namely the Tan dataset [15]. The results of comparing the accuracy values with the number of features extracted for each study's methodology are presented in Table 17. According to the Table 17, our method outperformed the accuracy value from another method that uses the same dataset.

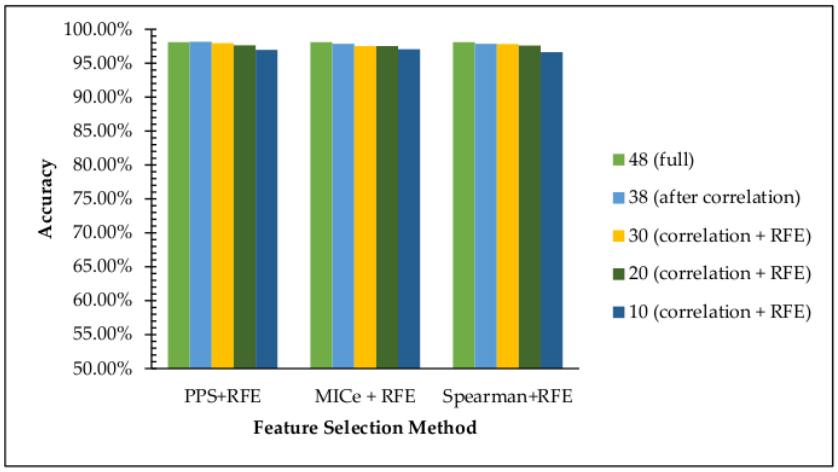

Figure 5. Comparison of all feature subset scenarios for dataset 1.

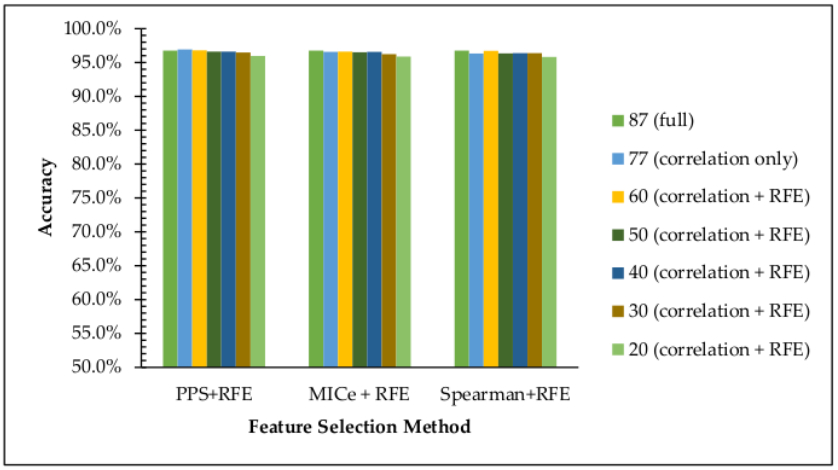

Figure 6. Comparison of all feature subset scenarios for dataset 2.

Table 17. The studies feature set and accuracy comparison on Tan's dataset.

No	Feature Typeset	Features	Accuracy
Chiew et al. [8]	Full Features	48	96.17%
	Baseline	10	94.60%
	Full Features HEFS	30	92.40%
	Baseline HEFS	5	93.22%
Khan et al. [17]	Full Features	48	97.87%
	Using PCA	30	94.90%
Dangwal et al. [18]	RF	30	93.7%
Ours	Full features	48	98.1%
	PPS Correlation Only	38	98.16%
	PPS+RFE	30	97.96%
	PPS+RFE	20	97.63%
	PPS+RFE	10	96.96%
	MICe TICe Correlation Only	38	97.86%

Table 17. *Cont.*

No	Feature Typeset	Features	Accuracy
	MICe TICe + RFE	30	97.53%
	MICe TICe + RFE	20	97.53%
	MICe TICe + RFE	10	97.06%
	Spearman Correlation Only	38	97.86%
	Spearman + RFE	30	97.83%
	Spearman + RFE	20	97.6%
	Spearman + RFE	10	96.63%

Table 17 shows the achievement of our proposed approaches compared to the existing work. Refs. [8,17] utilize the Tan's dataset [15] with similar proportion of training and testing at 70% and 30%.

There is only one study on this topic utilize the second dataset, namely the Hannousse and Yahiouche [16] dataset. The results of comparing the accuracy values with the number of features extracted for each study's methodology are presented in Table 18. According to the Table 18, our method outperformed the accuracy value from another method that uses the same dataset.

Table 18. The studies feature set and accuracy comparison on Hannousse and Yahiouche dataset.

No	Feature Typeset	Features	Accuracy
Hannousse and Yahiouche [16]	URLs based + Content based + External features (Full features)	87	96.61%
	URLs based + Content based	80	94.10%
	URLs based + External features	63	96.65%
	Content based + External features	31	95.13%
	URLs based	56	91%
	Content based	24	89.90%
	External Features	7	94.09%
Ours	Full features	87	96.76%
	PPS Correlation Only	77	96.93%
	PPS+RFE	10	**95.58%**
	MICe TICe Correlation Only	77	96.58%
	MICe TICe + RFE	10	**95.21%**
	Spearman Correlation Only	77	96.32%
	Spearman + RFE	10	**95.04%**

Table 18 shows the achievement of our proposed approaches compared to the existing work. Hannousse and Yahiouche [16] utilize dataset 2 with similar proportion of training and testing at 70% and 30%. The accuracy of RF classification of 10 selected features slightly better than the result reported in [16] with more features considered in their approach.

After feature selection, we learn about what is important: redundant features on both datasets. Although both datasets have different set of data, they share some high rank features such as length of the host name, path level and the number of hyperlink. Those high ranking features remain in the 10 most important features in the final feature selection. The first 10 removed features are redundant and it was proven removing them lead to improvement of the classification performance. For example, both "nb_dots" (number of dots) and "nb_subdomains" (number of sub domain) are highly correlated since number dots (.) are the separators between domain and subdomains and therefore removing one of them will not suffer the quality of the model, since they represent the same thing.

5. Conclusions

Removal of inter-correlated features and low and negative correlation features to the output label leads to a better recognition rate of phishing dataset. The subset of phishing datasets selected with PPS+RFE scenario slightly over-performs compared to MICe

TICe+RFE and Spearman+RFE. According to the experimental result, Random Forest (RF) achieves better accuracy in recognizing the phishing website in Tan's [15] and Hannousse and Yahiouche 's [16] dataset. Our approach to feature selection and classification using random forest achieves slightly better accuracy on Tan's [15] dataset at 96,96% of accuracy, compared to the result reported in [7] and [18] at 94,6% and 93.7%, respectively. On Hannousse and Yahiouche's [16] dataset, our method of feature selection and classification using random forest also produces marginally better accuracy at 97.96%. This assumes that unimportant features can be removed without the recognition rate suffering too much. Therefore, we implement feature selection. We are aware that reducing the feature will definitely reduce the amount of information and our experiment result shows that consequent although the gap between full features and minimum subset (10 features) less than 2% of accuracy. However, as far as accuracy is concerned, the option of using dimensionality reduction such as principal component analysis (PCA) or autoencoder would be an interesting exploration. Another remaining problem of phishing is the evolving technique of phishing itself; therefore, exercising the new evidence by providing the latest dataset is always a challenge in the future.

Author Contributions: Conceptualization, J.M. and A.S.; methodology A.S. and J.M.; software, J.M.; investigation, J.M. and F.K.A.; writing—original draft preparation J.M., A.S., M.A. and F.K.A.; data curation, M.A. and F.K.A.; visualization, J.M.; validation, A.S.; supervision, A.S.; project administration, F.K.A.; funding acquisition, F.K.A.; resources, M.A. All authors have read and agreed to the published version of the manuscript.

Funding: The authors extend their appreciation to the Deanship of Scientific Research at Imam Mohammad Ibn Saud Islamic University for funding this work through Research Group no. RG-21-51-01.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors are grateful to LPPM Universitas Amikom Yogyakarta for their administrative and technical support.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Alabdan, R. Phishing Attacks Survey: Types, Vectors, and Technical Approaches. *Futur. Internet* **2020**, *12*, 168. [CrossRef]
2. Ding, Y.; Luktarhan, N.; Li, K.; Slamu, W. A keyword-based combination approach for detecting phishing webpages. *Comput. Secur.* **2019**, *84*, 256–275. [CrossRef]
3. Alkhalil, Z.; Hewage, C.; Nawaf, L.; Khan, I. Phishing Attacks: A Recent Comprehensive Study and a New Anatomy. *Front. Comput. Sci.* **2021**, *3*. [CrossRef]
4. Sonowal, G.; Kuppusamy, K. PhiDMA—A phishing detection model with multi-filter approach. *J. King Saud Univ.-Comput. Inf. Sci.* **2020**, *32*, 99–112. [CrossRef]
5. APWG. *Phishing Activity Trends Report 3rd Quarter 2021*; Anti Phishing Working Group: Lexington, KY, USA, 2021. Available online: https://docs.apwg.org/reports/apwg_trends_report_q3_2021.pdf (accessed on 20 February 2021).
6. Phislabs. 2019 Phishing Trends and Intelligence Report the Growing Social Engineering Threat. 2019. Available online: https://info.phishlabs.com/hubfs/2019%20PTI%20Report/2019%20Phishing%20Trends%20and%20Intelligence%20Report.pdf (accessed on 26 April 2020).
7. Abutair, H.; Belghith, A.; AlAhmadi, S. CBR-PDS: A case-based reasoning phishing detection system. *J. Ambient Intell. Humaniz. Comput.* **2019**, *10*, 2593–2606. [CrossRef]
8. Chiew, K.L.; Tan, C.L.; Wong, K.; Yong, K.S.; Tiong, W.K. A new hybrid ensemble feature selection framework for machine learning-based phishing detection system. *Inf. Sci.* **2019**, *484*, 153–166. [CrossRef]
9. Sahingoz, O.K.; Buber, E.; Demir, O.; Diri, B. Machine learning based phishing detection from URLs. *Expert Syst. Appl.* **2019**, *117*, 345–357. [CrossRef]
10. Naaz, S. Detection of Phishing in Internet of Things Using Machine Learning Approach. *Int. J. Digit. Crime Forensics* **2021**, *13*, 1–15. [CrossRef]
11. Alotaibi, B.; Alotaibi, M. Consensus and majority vote feature selection methods and a detection technique for web phishing. *J. Ambient Intell. Humaniz. Comput.* **2021**, *12*, 717–727. [CrossRef]

12. Hutchinson, S.; Zhang, Z.; Liu, Q. Detecting Phishing Websites with Random Forest. In *International Conference on Machine Learning and Intelligent Communications, Proceedings of the Third International Conference, MLICOM 2018, Hangzhou, China, 6–8 July 2018*; Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering (LNICST); Springer: Cham, Switzerland, 2018; pp. 470–479. [CrossRef]
13. Karabatak, M.; Mustafa, T. Performance comparison of classifiers on reduced phishing website dataset. In Proceedings of the 6th International Symposium on Digital Forensic and Security (ISDFS), Antalya, Turkey, 22–25 March 2018; pp. 1–5. [CrossRef]
14. Zaini, N.S.; Stiawan, D.; Ab Razak, M.F.; Firdaus, A.; Din, W.I.S.W.; Kasim, S.; Sutikno, T. Phishing detection system using nachine learning classifiers. *Indones. J. Electr. Eng. Comput. Sci.* **2020**, *17*, 1165–1171. [CrossRef]
15. Tan, C.L. Mendeley Data—Phishing Dataset for Machine Learning: Feature Evaluation. 2018. Available online: https://data.mendeley.com/datasets/h3cgnj8hft/1 (accessed on 13 May 2020).
16. Hannousse, A.; Yahiouche, S. Towards benchmark datasets for machine learning based website phishing detection: An experimental study. *Eng. Appl. Artif. Intell.* **2021**, *104*, 104347. [CrossRef]
17. Khan, S.A.; Khan, W.; Hussain, A. Phishing Attacks and Websites Classification Using Machine Learning and Multiple Datasets (A Comparative Analysis). In *International Conference on Intelligent Computing, Proceedings of the 16th International Conference, ICIC 2020, Bari, Italy, 2–5 October 2020*; Springer: Cham, Switzerland, 2020; pp. 301–313. [CrossRef]
18. Dangwal, S.; Moldovan, A.-N. Feature Selection for Machine Learning-based Phishing Websites Detection. In Proceedings of the International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA), Dublin, Ireland, 14–18 June 2021; pp. 1–6. [CrossRef]
19. Al-Sarem, M.; Saeed, F.; Al-Mekhlafi, Z.; Mohammed, B.; Al-Hadhrami, T.; Alshammari, M.; Alreshidi, A.; Alshammari, T. An Optimized Stacking Ensemble Model for Phishing Websites Detection. *Electronics* **2021**, *10*, 1285. [CrossRef]
20. Haynes, K.; Shirazi, H.; Ray, I. Lightweight URL-based phishing detection using natural language processing transformers for mobile devices. *Procedia Comput. Sci.* **2021**, *191*, 127–134. [CrossRef]
21. Susanti, D.S.; Sukmawaty, Y.; Salam, N. *Analisis Regresi dan Korelasi*, 1st ed.; CV IRDH: Malang, Indonesia, 2019; pp. 49–50.
22. Reshef, D.N.; Reshef, Y.A.; Finucane, H.K.; Grossman, S.R.; McVean, G.; Turnbaugh, P.J.; Lander, E.S.; Mitzenmacher, M.; Sabeti, P.C. Detecting Novel Associations in Large Data Sets. *Science* **2011**, *334*, 1518–1524. [CrossRef] [PubMed]
23. Simon, N.; Tibshirani, R. Commen on Detection Novel Associations in Large Data Sets by Reshef et al, Science Dec 16, 2011. *Science* **2011**, *334*, 1521–1522.
24. Reshef, D.N.; Reshef, Y.A.; Sabeti, P.C.; Mitzenmacher, M.M. An Empirical Study of Leading Measures of Dependence. *arXiv* **2015**, arXiv:1505.02214.
25. Reshef, Y.A.; Reshef, D.N.; Finucane, H.K.; Sabeti, P.C.; Mitzenmacher, M. Measuring dependence powerfully and equitably. *J. Mach. Learn. Res.* **2016**, *17*, 1–63.
26. Albanese, D.; Riccadonna, S.; Donati, C.; Franceschi, P. A practical tool for maximal information coefficient analysis. *GigaScience* **2018**, *7*, 1–8. [CrossRef] [PubMed]
27. Wetschoreck, F.; Krabel, T.; Krishnamurthy, S. *8080labs/ppscore: Zenodo release*; Zenodo: London, UK, 2020. [CrossRef]
28. Raza, M.S.; Qamar, U. *Understanding and Using Rough Set Based Feature Selection: Concepts, Techniques and Applications*; Springer: Singapore, 2017; pp. 42–45.
29. Whitley, B.E.; Kite, M.E.; Adams, H.L. *Principles of Research in Behavioral Science*, 3rd ed.; Routledge: New York, NY, USA, 2013.
30. Senawi, A.; Wei, H.-L.; Billings, S.A. A new maximum relevance-minimum multicollinearity (MRmMC) method for feature selection and ranking. *Pattern Recognit.* **2017**, *67*, 47–61. [CrossRef]
31. Liu, H.; Motoda, H. *Computational Methods of Feature Selection*; CRC Press: Boca Raton, FL, USA, 2007.
32. Pilnenskiy, N.; Smetannikov, I. Feature Selection Algorithms as One of the Python Data Analytical Tools. *Futur. Internet* **2020**, *12*, 54. [CrossRef]
33. Das, S.; Cakmak, U.M. *Hands-On Automated Machine Learning A Beginner's Guide to Building Automated Machine Learning Systems Using AutoML and Python*; Packt Publishing Ltd.: Birmingham, UK, 2018.
34. Mishra, A. *Machine Learning for IOS Developers*; Wiley: Oxford, UK, 2020. [CrossRef]
35. Rao, R.S.; Pais, A.R. Detection of phishing websites using an efficient feature-based machine learning framework. *Neural Comput. Appl.* **2019**, *31*, 3851–3873. [CrossRef]
36. Salihovicd, I.; Serdavic, H.; Kevric, J. The Role of Feature Selection in Machine Learning for Detection of Spam and Phishing Attacks. In *International Symposium on Innovative and Interdisciplinary Applications of Advanced Technologies (IAT)*; Springer: Cham, Switzerland, 2019; Volume 3, pp. 476–483. [CrossRef]
37. Butnaru, A.; Mylonas, A.; Pitropakis, N. Towards Lightweight URL-Based Phishing Detection. *Futur. Internet* **2021**, *13*, 154. [CrossRef]
38. Podgor, M.J.; Gibbons, J.D. Nonparametric Measures of Association. *J. Am. Stat. Assoc.* **1994**, *89*, 719. [CrossRef]

futureinternet-14-00229.pdf

ORIGINALITY REPORT

13%	10%	8%	5%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	api.crossref.org Internet Source	3%
2	Zhiqiang Dai, Gaochao Xu, Ziqi Liu, Jiaqi Ge, Wei Wang. "Energy Saving Strategy of UAV in MEC Based on Deep Reinforcement Learning", Future Internet, 2022 Publication	1%
3	Submitted to Eastern Institute of Technology Student Paper	1%
4	www.mdpi.com Internet Source	1%
5	irep.ntu.ac.uk Internet Source	1%
6	Behnaz Abbasgholi-NA, Seyed Reza Nokhbeh, Osamah A. Aldaghri, Khalid Hassan Ibnaouf, Nawal Madkhali, Humberto Cabrera. "Thermal Diffusivity and Conductivity of Polyolefins by Thermal Lens Technique", Polymers, 2022 Publication	<1%
7	Submitted to University Of Tasmania	

	Student Paper	<1%
8	Andie Setiyoko, T. Basaruddin, Aniati Murni Arymurthy. "Minimax Approach for Semivariogram Fitting in Ordinary Kriging", IEEE Access, 2020 Publication	<1%
9	Erzhou Zhu, Zhile Chen, Jie Cui, Hong Zhong. "MOE/RF: A Novel Phishing Detection Model based on Revised Multi-Objective Evolution Optimization Algorithm and Random Forest", IEEE Transactions on Network and Service Management, 2022 Publication	<1%
10	Vasileios Cheimaras, Athanasios Trigkas, Panagiotis Papageorgas, Dimitrios Piromalis, Emmanouil Sofianopoulos. "A Low-Cost Open-Source Architecture for a Digital Signage Emergency Evacuation System for Cruise Ships, Based on IoT and LTE/4G Technologies", Future Internet, 2022 Publication	<1%
11	nottingham-repository.worktribe.com Internet Source	<1%
12	Submitted to Glyndwr University Student Paper	<1%

13	"Machine Learning and Autonomous Systems", Springer Science and Business Media LLC, 2022 Publication	<1 %
14	Kang Leng Chiew, Choon Lin Tan, KokSheik Wong, Kelvin S.C. Yong, Wei King Tiong. "A new hybrid ensemble feature selection framework for machine learning-based phishing detection system", Information Sciences, 2019 Publication	<1 %
15	Submitted to Melbourne Institute of Technology Student Paper	<1 %
16	link.springer.com Internet Source	<1 %
17	Emtethal K. Alamri, Abdullah M. Alnajim, Suliman A. Alsuhibany. "Investigation of Using CAPTCHA Keystroke Dynamics to Enhance the Prevention of Phishing Attacks", Future Internet, 2022 Publication	<1 %
18	www.hindawi.com Internet Source	<1 %
19	minerva.usc.es Internet Source	<1 %

20	www.igi-global.com Internet Source	<1%
21	Submitted to American Public University System Student Paper	<1%
22	T. Vaisakh, R. Jayabarathi. "Analysis on intelligent machine learning enabled with meta-heuristic algorithms for solar irradiance prediction", Evolutionary Intelligence, 2020 Publication	<1%
23	doi.org Internet Source	<1%
24	res.mdpi.com Internet Source	<1%
25	Dania Aljeaid, Amal Alzhrani, Mona Alrougi, Oroob Almalki. "Assessment of End-User Susceptibility to Cybersecurity Threats in Saudi Arabia by Simulating Phishing Attacks", Information, 2020 Publication	<1%
26	Khalid Alsubhi, Bander Alzahrani, Nikos Fotiou, Aiiad Albeshri, Mohammed Alreshoodi. "Reliable Application Layer Routing Using Decentralized Identifiers", Future Internet, 2022 Publication	<1%

27	research-management.mq.edu.au Internet Source	<1%
28	"Information and Communications Security", Springer Science and Business Media LLC, 2020 Publication	<1%
29	klme.nuist.edu.cn Internet Source	<1%
30	Submitted to Turun yliopisto Student Paper	<1%
31	Submitted to University of Northumbria at Newcastle Student Paper	<1%
32	Wang, Deng, Wang, Sangaiah, Cai, Almakhadmeh, Tolba. "Securing Cryptographic Chips against Scan-Based Attacks in Wireless Sensor Network Applications", Sensors, 2019 Publication	<1%
33	csrid.potensi-utama.ac.id Internet Source	<1%
34	ebin.pub Internet Source	<1%
35	"Frontier Computing", Springer Science and Business Media LLC, 2020 Publication	<1%

36	"Inventive Communication and Computational Technologies", Springer Science and Business Media LLC, 2020 Publication	<1 %
37	Areti Nagendra Soma Charan, Yu-Hung Chen, Jiann-Liang Chen. "Phishing Websites Detection using Machine Learning with URL Analysis", 2022 IEEE World Conference on Applied Intelligence and Computing (AIC), 2022 Publication	<1 %
38	Mohammed M. Alani, Hissam Tawfik. "PhishNot: A Cloud-Based Machine-Learning Approach to Phishing URL Detection", Computer Networks, 2022 Publication	<1 %
39	Routhu Srinivasa Rao, Alwyn Roshan Pais. "Two level filtering mechanism to detect phishing sites using lightweight visual similarity approach", Journal of Ambient Intelligence and Humanized Computing, 2019 Publication	<1 %
40	publications.eai.eu Internet Source	<1 %
41	"Applied Informatics", Springer Science and Business Media LLC, 2022 Publication	<1 %

42	"Emergent Converging Technologies and Biomedical Systems", Springer Science and Business Media LLC, 2022 Publication	<1%
43	"Advanced Technologies, Systems, and Applications III", Springer Science and Business Media LLC, 2019 Publication	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		