Spat. Inf. Res. (2023) 31:157–166
https://doi.org/10.1007/s41324-022-00489-8

Prediction of phishing websites using machine learning

Mithilesh Kumar Pandey[1] · Munindra Kumar Singh[1] · Saurabh Pal[1] · B. B. Tiwari[2]

Received: 29 July 2022 / Revised: 21 September 2022 / Accepted: 22 September 2022 / Published online: 6 October 2022

Abstract With the growing popularity of the information science, more application is being integrated with websites that can be accessed directly through the internet. This has increased the possibility of attack by ill-legal persons to steal personal information. To identify a phishing assault, several strategies have been presented. However, there is still opportunity for progress in the fight against phishing. The objective of this research paper is to develop a more accurate prediction model using Decision Tree (DT), Random Forest (RF) and Gradient Boosting Classifiers (GBC) with three features selection techniques Extra Tree (ET), Chi-Square and Recursive Feature Elimination (RFE). Since phishing websites dataset contains 89 features, therefore we have applied extra tree and chi-square, feature selection method to identify the limited important features and then recursive features elimination technique has been used to reduce the dataset up-to optimum important features. We have compared the performance of the developed model using machine learning algorithms and find the best prediction performance using GBC, followed by RF and DT. These algorithmic models capture the trends from various cases of phishing with over R-square, Root Mean Square Error (RMSE), and Mean Absolute Error (MAE), in each case.

Keywords Machine learning · Decision tree algorithm · Random forest algorithm · Gradient boosting and phishing websites

✉ Saurabh Pal
drsaurabhpal@yahoo.co.in

1 Department of Computer Applications, VBS Purvanchal University, Jaunpur, Uttar Pradesh 222001, India

2 Department of Electronics and Communication, VBS Purvanchal University, Jaunpur, Uttar Pradesh 222001, India

1 Introduction

Phishing is a type of cybercrime that involves establishing a fake website that seems like a real website in order to collect vital or private information from consumers. Phishing detection method deceives the user by capturing a picture from a reputable website. Image comparison, on the other hand, takes more time and requires more storage space. Provides a high percentage of false negatives and fails to detect minor changes in visual appearance. Phishing detection method works well with huge datasets. Phishing detection also eliminates the disadvantages of the current technique and allows for the detection of zero-day attacks. As a result, the suggested method will focus on detecting phishing websites using tree-based classifiers [1].

Hackers used better way their phony websites to gain personal information. We find some signs and aspects that can help to judge the difference between a real and a fake website.

We can avoid phishing websites by using direct websites from the URL address or using real websites Pop-Ups windows. If we find any warning message which shows harm computer Non-Secured Sites then left the URL or if we find lacks https Pay Close Attention to the URL or Web Address insecure. If the Content and Design of the Website for some are below standard then it will be phishing website. Community people already provide credit score so we can judge on the basis of Online Reviews.

The Table 1 represents total number of unique phishing reports (campaigns) received, according to Anti-Phishing Working Group (APWG). With the study, we find on July 15, 2020, various twitter suffered a strong break that combined elements of security and phishing. With the previous study, we find various people targeted on identifying malicious URLs from the massive set of URLs [2]. The

Table 1 Total number of unique phishing reports (campaigns) received, according to anti phishing working group

Years	Jan–Mar	Apr–June	Jul–Sep	Oct–Dec	Total
2005	39,196	44,448	41,473	47,946	173,063
2006	53,520	66,170	71,956	76,480	268,126
2007	78,393	75,959	88,055	85,407	327,814
2008	85,630	76,837	91,196	82,302	335,965
2009	96,011	108,370	115,370	92,641	412,392
2010	86,985	85,062	73,814	67,656	313,517
2011	74,955	65,376	65,844	78,270	284,445
2012	85,443	84,125	74,390	76,123	320,081
2013	74,127	76,483	180,012	160,777	491,399
2014	171,792	171,801	163,333	197,252	704,178
2015	221,211	417,472	395,015	380,280	1,413,978
2016	557,964	315,524	229,251	211,032	1,313,771
2017	318,940	273,395	296,208	233,613	1,122,156
2018	262,704	264,483	270,557	239,910	1,037,654
2019	112,393	112,163	118,260	132,553	475,369

main objectives of this study to focus each and every angle of phishing dataset by various features selection methods and features elimination method of machine learning. The Sects. 1, 2, 3, 4, 5 and 6 organized Introduction, background related literature, methodology of the research, results, and discussion and concludes respectively.

Jain and Gupta considered Naïve Bayes and support vector machine with malicious websites. They found both learners do not store previous results in the memory. Finally, authors found efficiency of URL detector may be reduced [3].

Purbay and Kumar [4] examined multiple classifiers with URL websites. Authors measured the performance of multiple classifiers but they did not support retrieval capacity of the algorithms.

Gandotra and Gupta [5] used multiple predictors for analyzing malicious URLs. After all the examination they found the performance of the system was better compare to other classifiers, but a drawback was run with the organized classifier, this system did not support large volume dataset.

Le et al. [6] organized a deep learning system based on URL detector applied on lexical features for examined phishing websites. They found more time requirements for produce an output by deep learning.

Hong et al. [7] organized a system for URLs sites to identify lexical features in phishing websites. They evaluated crawker based dataset and found no assurance of URL detector with real time.

Kumar et al. [8] examined URL detector blacklisted dataset. They used a system on lexical features and classified malicious and legitimate websites. In the examination authors find the performance of the detector reduced with time.

Abutair and Belghith [9] discussed for classifying websites and predicts the phishing websites. They used GA techniques to measure the performance of time for huge and complex dataset.

Rao and Pais [10] experimented with logo, favicon, scripts and styles attributes of page. They update page attributes that helps in performance reduction in detecting system.

Aljofy et al. [11] discussed about identifying the phishing page using CNN algorithm. They found organized system easy retrieve image rather than text. Finally, authors detect CNN results are better compare to another classifier.

AlEroud and Karabatis [12] organized a system of neural network for observe adversarial network. The system easily identifies the impression of advert network compare to other algorithms.

Althobaiti et al. [13] have discussed total URL features in six categories: lexical, host, rank, redirection, certificate, search engines and black/white lists. All these six categories of features make the 89 features of the UCI machine learning phishing website dataset.

Gupta et al. [14] have applied the features selection technique as choosing the lexical feature only and obtained the highest accuracy of 99.57% in the case of random forest. Since the author has chosen only a smaller number of features so they obtained to much high accuracy, which is not justifiable.

Sahoo et al. [15] have presented a review paper in which they have discussed total phishing website features in five categories as black list, lexical, host, content-based features and other features.

In this study ensemble classification approach for detecting Phishing Websites. Training, feature optimization, and testing are the three primary steps in this process. The classifiers (DT, RF, and Gradient Boosting) were first trained using training websites dataset. There was no optimization strategy used in this stage. In the second stage, a hybrid features selection approach is utilized to optimize these classifiers that may be used to improve the classifiers' overall accuracy. Following that, depending on their ranking, optimized classifiers were used as the chi-square, extra tree, recursive features elimination techniques. The result obtained by the proposed model shows a high improvement in terms of accuracy as the results of research studied in literature reviewed.

2 Methods

In this study, we have applied three different feature selection techniques: Extra Tree, Chi-Square and Recursive Feature Elimination on phishing website dataset obtained by UCI machine learning repository. Phishing website dataset consist of 89 variables, by applying these three feature selection techniques we obtained 29 most important features (attributes) and obtained new optimum subsets of phishing website dataset. Then we have applied three machine learning techniques: Decision Tree, Random Forest and Gradient Boosting Classifier to train the optimum subset of phishing website dataset.

The predictions obtained by three different feature selection methods are compared to choose the best feature selection techniques and best prediction accuracy. The whole proposed methodology used in this research paper is described in Fig. 1.

Following classifiers and feature selection techniques are used to evaluate the performance of proposed model.

2.1 Gradient boosting

Regularization strategies that punish various sections of the algorithm and overall enhance the algorithm's performance by decreasing over-fitting might help it. GB is a non-parametric supervised machine learning technique [16]. Boosting is the method for converting weak learners into strong learner. In gradient boosting, each new tree is a fit on a modified version of the original data set. The gradient boosting algorithm (GB) can be most easily explained by first introducing the AdaBoost Algorithm. The AdaBoost Algorithm begins by training a decision tree in which each observation is assigned an equal weight.

2.2 Random forest

A random forest classifier is a supervised learning technique and can be used for classification and regression analysis. This algorithm is most simple and flexible to use. A forest is collection of various trees. If high number of trees is present, then the forest is more robust. Random forests randomly select data to create decision trees, and give prediction from each tree and choose the best solution by use of voting technique. It also provides an attractive excellent display of the feature importance [17].

2.3 Decision tree

A decision tree is a supervised learning based predictive modeling tool [18]. This tool works on the principle of multivariate analysis, that can help in predicting, explaining, describing, classifying the

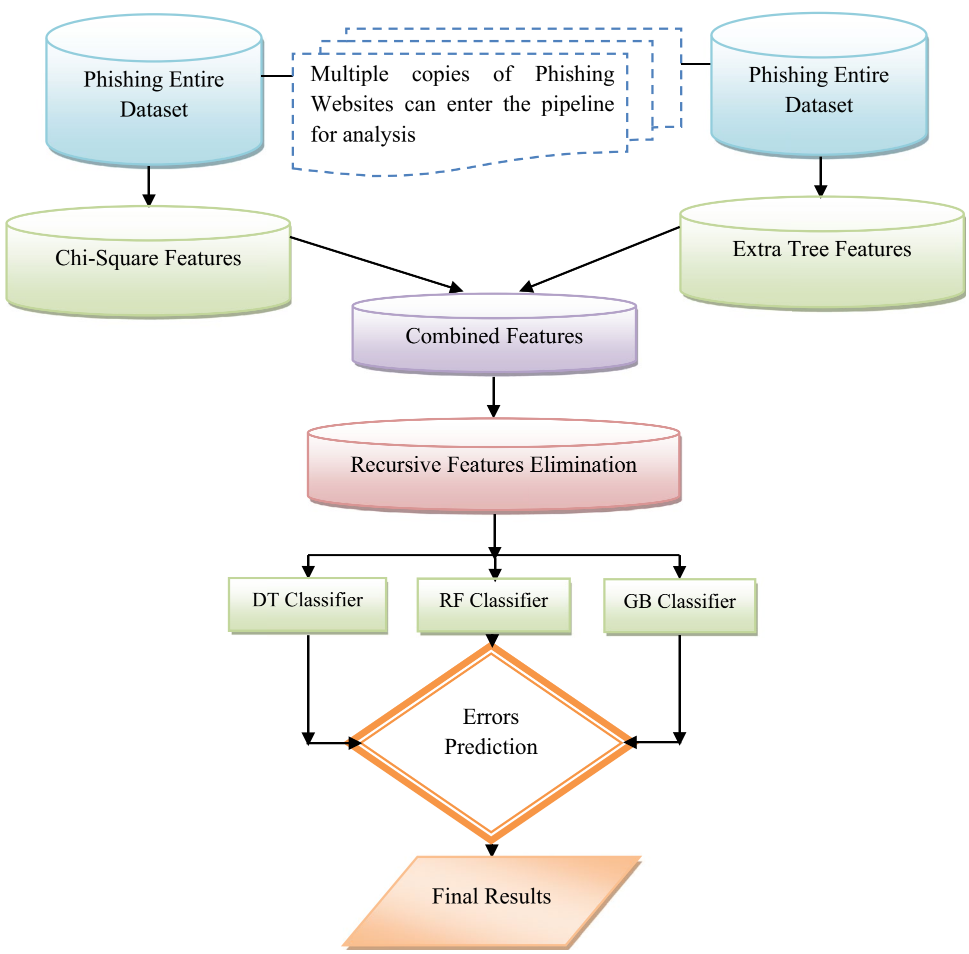


Fig. 1 Represents proposed method for phishing dataset

outcome. It splits the dataset based on multiple conditions, thus help in describing beyond one cause cases and help us describe the condition based on multiple influences. Quinlan created Iterative Dichotomiser version 3 (ID3) algorithms, which was used for generation of decision trees. A decision tree is generated from root following top-down approach that involves partitioning of data, entropy is used to calculate homogeneity of data samples, if the sample data is completely homogeneous, the entropy value is 0 or if sample data is not homogeneous, the entropy value is 1. Entropy can be calculated using Eq. (1).

$$E(S) = \sum_{i=1}^{c} -p_i log_2 p_i \tag{1}$$

2.4 Dataset analysis

We have used phishing website dataset collected from UCI machine learning repository, which consists 89 features as shown in Table 2. There is total 11,430 numbers of instances out of which 5715 are Legitimate and 5715 are Phishing. The categorical variables "Legitimate" and "Phishing" in the gathered dataset have been changed to numerical values by substituting the values "1" and "− 1" for "Legitimate" and "Phishing," respectively.

3 Results

The feature selection techniques are very important for improving the performance of a developed model. We have applied three feature selection techniques extra tree, chi-square and recursive feature elimination technique to find the 29 relevant features, which play on important role to improve the results of developed model.

3.1 Extra trees

Extra Trees is an ensemble machine learning approach that aggregates the predictions of many decision trees (see Fig. 2).

Table 2 Phishing website data attributes

No.	Attributes	No.	Attributes	No.	Attributes
1	url	31	tld_in_path	61	ratio_nullHyperlinks
2	length_url	32	tld_in_subdomain	62	nb_extCSS
3	length_hostname	33	abnormal_subdomain	63	ratio_intRedirection
4	ip	34	nb_subdomains	64	ratio_extRedirection
5	nb_dots	35	prefix_suffix	65	ratio_intErrors
6	nb_hyphens	36	random_domain	66	ratio_extErrors
7	nb_at	37	shortening_service	67	login_form
8	nb_qm	38	path_extension	68	external_favicon
9	nb_and	39	nb_redirection	69	links_in_tags
10	nb_or	40	nb_external_redirection	70	submit_email
11	nb_eq	41	length_words_raw	71	ratio_intMedia
12	nb_underscore	42	char_repeat	72	ratio_extMedia
13	nb_tilde	43	shortest_words_raw	73	sfh
14	nb_percent	44	shortest_word_host	74	iframe
15	nb_slash	45	shortest_word_path	75	popup_window
16	nb_star	46	longest_words_raw	76	safe_anchor
17	nb_colon	47	longest_word_host	77	onmouseover
18	nb_comma	48	longest_word_path	78	right_clic
19	nb_semicolumn	49	avg_words_raw	79	empty_title
20	nb_dollar	50	avg_word_host	80	domain_in_title
21	nb_space	51	avg_word_path	81	domain_with_copyright
22	nb_www	52	phish_hints	82	whois_registered_domain
23	nb_com	53	domain_in_brand	83	domain_registration_length
24	nb_dslash	54	brand_in_subdomain	84	domain_age
25	http_in_path	55	brand_in_path	85	web_traffic
26	https_token	56	suspecious_tld	86	dns_record
27	ratio_digits_url	57	statistical_report	87	google_index
28	ratio_digits_host	58	nb_hyperlinks	88	page_rank
29	punycode	59	ratio_intHyperlinks	89	status
30	port	60	ratio_extHyperlinks		

Extra Trees ensemble is a decision tree ensemble that is similar to random forest. This is a model-based technique to picking characteristics that uses tree-based supervised models to make judgments about their relevance. Instead of using a bootstrap replica, it fits each decision tree to the whole dataset and splits the nodes at random. Random Forest selects the best split, whereas Extra Trees choose it at random [19]. The greatest and lowest feature significance levels are represented by the extra tree. Once the split points are chosen, the two algorithms determine which of the subsets of characteristics the best is.

$$\begin{bmatrix} 0.00370036 & 0.0048668 & 0.0056781 & 0.0022648 & 0.00534266 & 0.00406504 & 0.00465011 \\ 0.00287578 & 0.01078235 & 0.00704151 & 0.01275825 & 0.00806429 & 0.00399562 & 0.01562412 \\ 0.00212852 & 0.01431833 & 0.00257516 & 0.24535652 & 0.05775128 & 0.5861604 & \end{bmatrix}$$

3.2 Chi-square

We want to pick features that are heavily dependent on the reaction while we're selecting features in Table 3. This test is based on frequencies rather than factors like mean and standard deviation (as a non-parametric test). The test is only useful for hypothesis testing and not for estimate. As previously stated, this test contains the additive property [20].

3.3 Recursive feature elimination

Recursive Feature Elimination is popular because it is easy to configure and use and because it is effective at selecting those features (columns) in a training dataset that are more or most relevant in predicting the target variable. There are two important configuration options when using RFE: the choice in the number of features to select and the choice of the algorithm used to help choose features. Both of these hyper parameters can be explored, although the performance of the method is not strongly dependent on these hyper parameters being configured well.

The resultant features are shown in Fig. 3.

Fig. 2 Represents Extra tree features selection method for phishing dataset

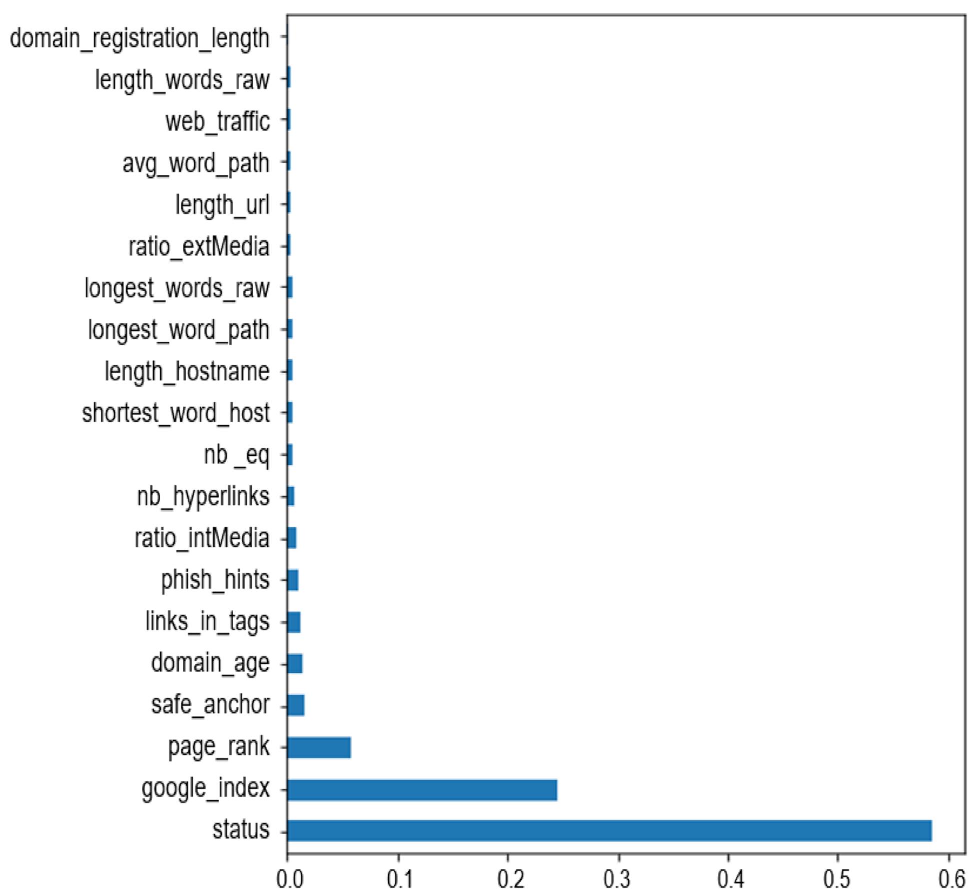


Table3 Represents Chi-Square features selection method for phishing dataset

Feature No.	Specs	Score
16	web_traffic	1.937304e+08
15	domain_age	2.992713e+06
9	nb_hyperlinks	4.279212e+05
14	domain_registration_length	4.028754e+05
0	length_url	3.532804e+04
6	longest_word_path	2.607634e+04
11	ratio_intMedia	2.131494e+04
5	longest_words_raw	1.450434e+04
12	ratio_extMedia	1.428797e+04
13	safe_anchor	1.415479e+04
10	links_in_tags	1.289123e+04
18	page_rank	6.032517e+03
19	status	5.715000e+03
7	avg_word_path	4.460650e+03
1	length_hostname	3.574909e+03
17	google_index	2.847872e+03
8	phish_hints	2.785077e+03
2	nb_eq	2.116255e+03
3	length_words_raw	2.099191e+03
4	shortest_word_host	1.760361e+03

After applying the three base classifiers decision tree, gradient boosting and random forest on data subset obtained after feature selection techniques, the obtained results are shown in Tables 4 and 5.

With the results, we found Table 4 Represents Computational table of Training Model (70%) Phishing Websites dataset by DT, GB and RF algorithms. The experimental results Random Forest calculated highest values for sensitivity and accuracy as 0.9761, 0.9655 respectively.

After the experiment, we found test results as test model (30%) Phishing Websites dataset, the Table 5 represents Computational table for DT, GB and RF algorithms. The experimental results Random Forest calculated highest values for sensitivity and accuracy as 0.9905, 0.9862 respectively.

Recursive Feature Elimination is a feature selection algorithm. Like an excel spreadsheet, a machine learning dataset for classification or regression is made up of rows and columns. Feature selection refers to methods for selecting a subset of a dataset's most important characteristics (columns). Using the feature importance property of the model in Fig. 4 we can extract the feature importance of each feature in the dataset. The feature significance score assigns a value to each data feature; the higher the score, the more essential or relevant the feature is to the output variable [21].

The Table 6 represents analysis (Training Set = 70%) for phishing dataset using classifiers. The results indicated that

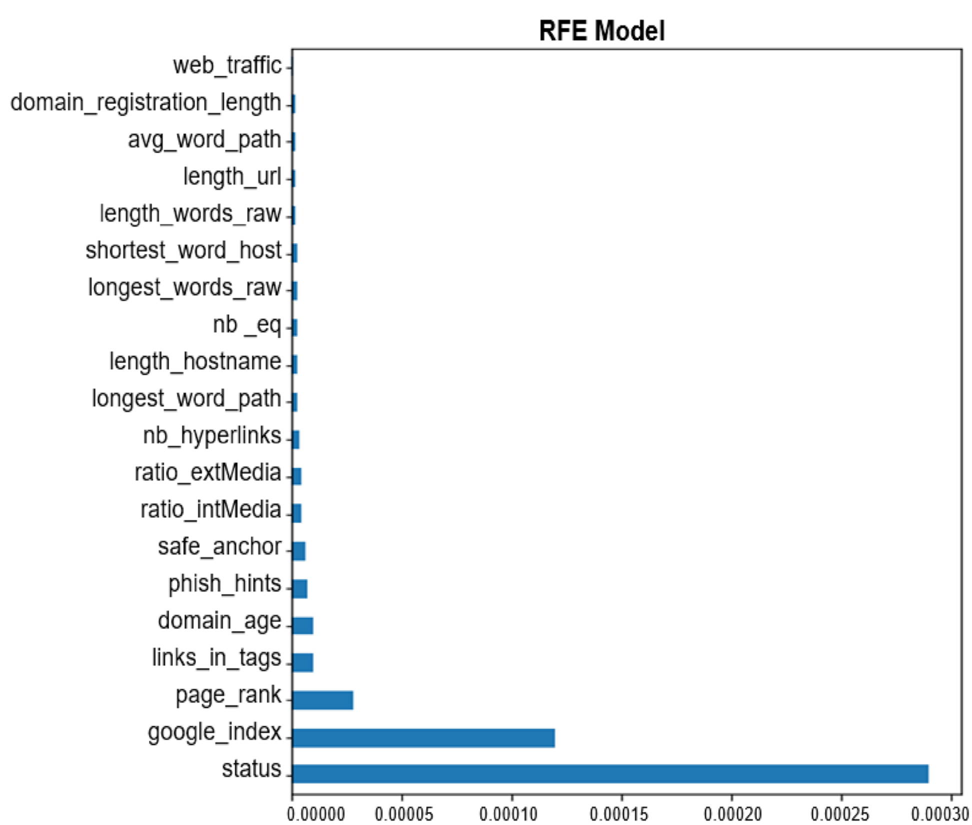


Fig. 3 Represents RFE features extraction method for phishing dataset

Table 4 Represents computational table of training model (70%) phishing websites dataset

Measure	DT	GB	RF
Sensitivity	0.958	0.965	0.976
Specificity	0.769	0.899	0.952
Precision	0.769	0.914	0.962
Accuracy	0.853	0.933	0.966
F1 Score	0.853	0.938	0.969

Table 5 Represents computational table of test model (30%) phishing websites dataset

Measure	DT	GB	RF
Sensitivity	0.816	0.942	0.991
Specificity	0.918	0.885	0.959
Precision	0.976	0.957	0.993
Accuracy	0.836	0.927	0.986
F1 Score	0.889	0.95	0.992

random forest classifiers had achieved the highest Correlation coefficient result of 0.9317% when compared to Decision Tree, Random Forest and Gradient Boosting [22].

The Table 7 represents analysis of test Set on 30% phishing dataset using classifiers. The results indicated that random forest classifiers had achieved the highest Correlation coefficient result of 0.9816% and lowest error, when compared to Decision Tree, Random Forest and Gradient Boosting. The random forest performs better compare to other selected classifiers in phishing website. The features selection methods determine effective of phishing website in Table 7.

4 Discussion

Correlation coefficients are used to determine the strength of the link between two variables [23]. Correlation involves determining the correlation between two variables. By the experiment, we find (Training Set = 70%) for Decision Tree, Random Forest, Gradient Boosting calculated as Correlation coefficient 0.8593, 0.9317, 0.7311; Analysis (Test Set = 30%), Decision Tree, Random Forest, Gradient Boosting calculated as Correlation coefficient 0.9281, 0.9816, 0.8014 in Fig. 4.

MAE is calculated [24] as:

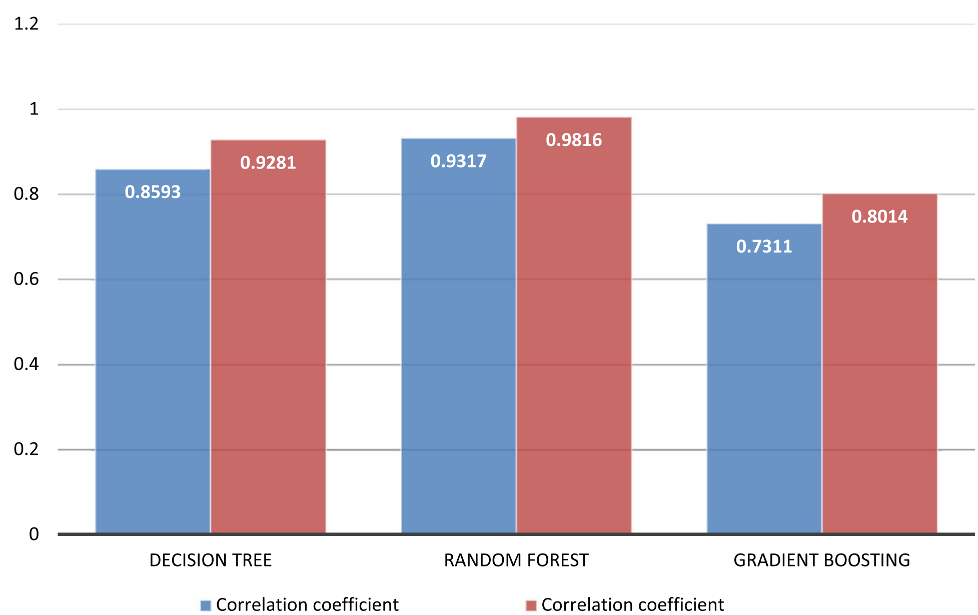


Fig. 4 Represents analysis Correlation method for phishing dataset

Table 6 Represents analysis (Training Set = 70%) for phishing dataset using classifiers

Analysis (Training Set = 70%)	Decision tree	Random forest	Gradient boosting
Correlation coefficient	0.8593	0.9317	0.7311
Mean absolute error	0.0703	0.0822	0.2327
Root mean squared error	0.2652	0.1825	0.3412
Relative absolute error (%)	14.07	16.44	46.54
Root relative squared error (%)	53.04	36.49	68.23

Table 7 Represents analysis (Test Set = 30%) for phishing dataset using classifiers

Analysis (Test Set = 30%)	Decision tree	Random forest	Gradient boosting
Correlation coefficient	0.9281	0.9816	0.8014
Mean absolute error	0.064	0.0751	0.1625
Root mean squared error	0.1964	0.1126	0.271
Relative absolute error (%)	12.19	14.41	42.50
Root relative squared error (%)	46.28	29.52	61.19

$$MAE = \frac{\sum_{i=1}^{n} \left| y_i - x_i \right|}{n} = \frac{\sum_{i=1}^{n} \left| e_i \right|}{n} \tag{2}$$

By the experiment (Training Set = 70%) for Decision Tree, Random Forest and Gradient Boosting calculated as Mean absolute error 0.0703, 0.0822, 0.2327 and analysis for (Test Set = 30%), Decision Tree, Random Forest and Gradient Boosting evaluated as Mean absolute error 0.064, 0.0751, 0.1625 in Fig. 5.

The relative absolute error [25] is calculated as:

$$E_i = \frac{\sum_{j-1}^{n} \left| P_{(ij)} - T_j \right|}{\sum_{j=1}^{n} \left| T_j - \overline{T} \right|} \tag{3}$$

where $P(ij)$ = predicted value and Tj = target value

$$\overline{T} = \frac{1}{n} \sum_{j=1}^{n} T_j \tag{4}$$

By the experiment, we find (Training Set = 70%), Decision Tree, Random Forest and Gradient Boosting calculated as Relative absolute error 14.0673%, 16.4381%, 46.5433% and analysis for (Test Set = 30%), Decision Tree, Random Forest and Gradient Boosting evaluated as Relative absolute error 12.1931%, 14.4138%, 42.499% respectively in Fig. 6.

By the experiment, we find (Training Set = 70%), Decision Tree, Random Forest and Gradient Boosting calculated as Root relative squared error 53.0401%, 36.4876%, 68.2275% and analysis for (Test Set = 30%)**,** Decision Tree, Random Forest and Gradient Boosting evaluated as Root relative squared error**,** 46.2762%, 29.5203%, 61.1923% respectively in Fig. 7.

RMSE [26] Formulated as:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (y_i - \hat{y})^2}{n}} \tag{5}$$

With the results, we find (Training Set = 70%)**,** Decision Tree**,** Random Forest and Gradient Boosting calculated as Root mean squared error 0.2652, 0.1825, 0.3412 and analysis for (Test Set = 30%)**,** Decision Tree, Random Forest and Gradient Boosting examined as Root mean squared error 0.1964, 0.1126, 0.271 respectively.

Because the sample data set has labels, this study uses supervised machine learning (phishing and legitimate). Furthermore, supervised machine learning produces good

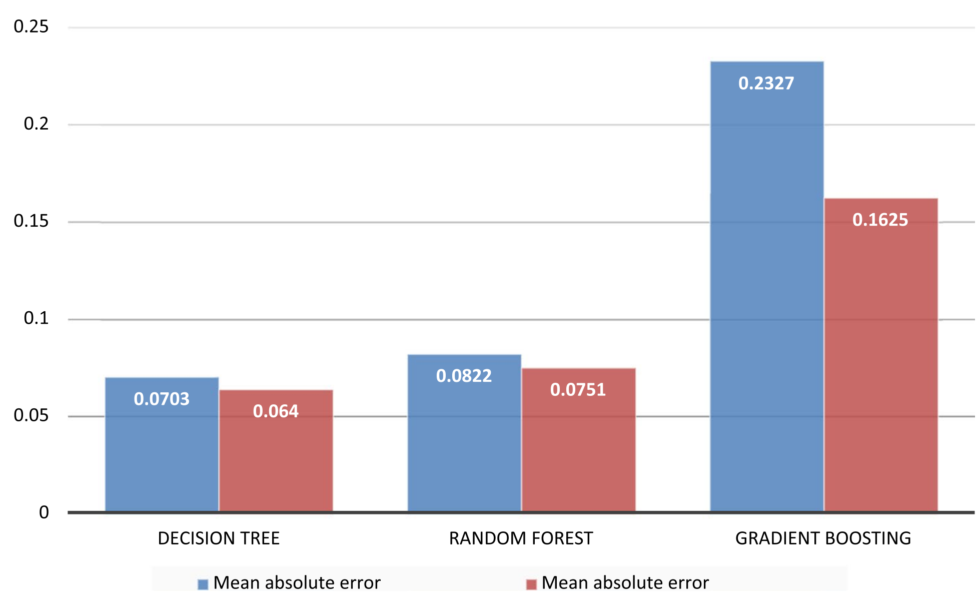


Fig. 5 Represents analysis MAE for phishing dataset

Fig. 6 Represents analysis RAE for phishing dataset

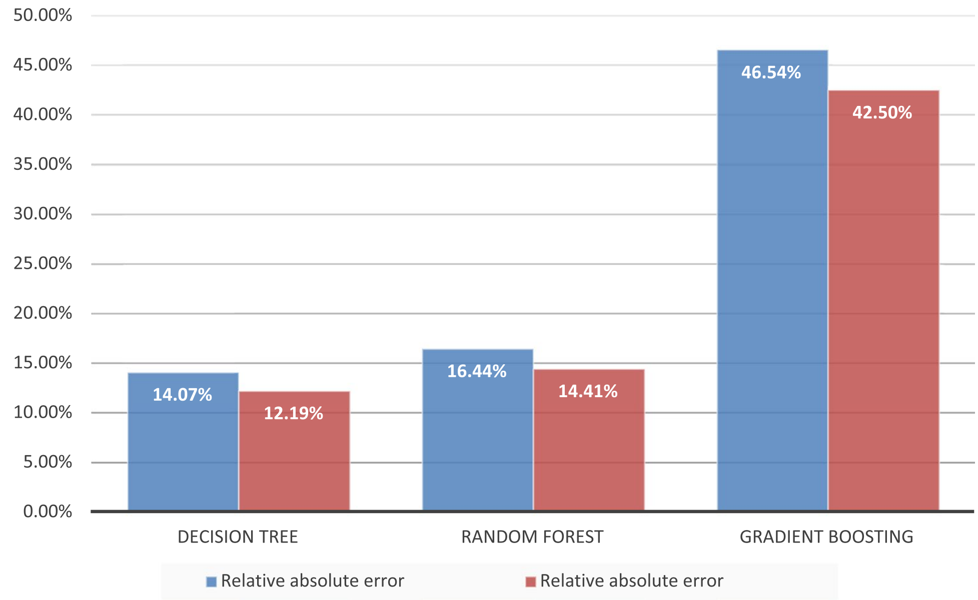
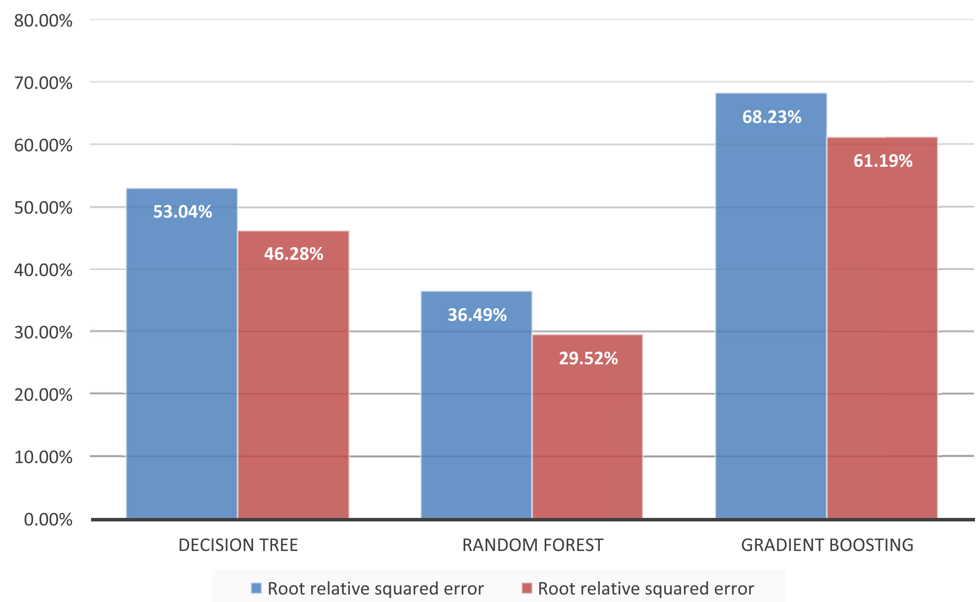


Fig. 7 Represents analysis RSE for phishing dataset



outcomes by reducing mistakes. In this research paper we have used three classifiers is RF, DT and GB and evaluated "R-square, Root Mean Square Error, and Mean Absolute Error". Tables 5 and 6 shows the Random Forest algorithm perform best compare to decision tree and gradient boosting classifiers in training and testing phase for phishing datasets.

5 Conclusion

In this research paper, we used Chi-Square and Extra Tree features selection techniques for organizing complex dataset and extract import features by Recursive Features Elimination as pipeline model, then trained three different machine learning method as Random Forest, Decision Tree and Gradient boosting on 70% phishing dataset and test on 30% dataset. In all experiment, we find Random calculated Correlation coefficient 0.9317, Mean absolute error 0.0822, Root mean squared error 0.1825, Relative absolute error 16.4381%, and Root relative squared error 36.4876%. Analysis (Test Set = 30%)**,** Random calculated correlation coefficient 0.9816,Mean absolute error 0.0751, Root mean squared error 0.1126, Relative absolute error 14.4138%,Root relative squared error 46.2762%. Finally, we concluded Random Forest classifier performs better results compare to another classifier. In the future, we plan to extend this by using real online real dataset using various ensemble models and predict user beneficial results.

Acknowledgements This work was supported by the VBS Purvanchal University, Jaunpur. I am indebted to the people who supported to the research and shared their ideas. I appreciate Prof. Surjeet Kumar due to his scientific advice related to the subject of this research.

Declarations

Conflict of Interest The authors declare no conflicts of interest.

References

1. Lam, I. F., Xiao, W. C., Wang, S. C., & Chen, K. T. (2009, June). Counteracting phishing page polymorphism: An image layout analysis approach. In *International conference on information security and assurance* (pp. 270–279). Springer.
2. Krombholz, K., Hobel, H., Huber, M., & Weippl, E. (2015). Advanced social engineering attacks. *Journal of Information Security and applications, 22*, 113–122.
3. Jain, A. K., & Gupta, B. B. (2018). PHISH-SAFE: URL features-based phishing detection system using machine learning. In *Cyber security* (pp. 467–474). Springer.
4. Purbay, M., & Kumar, D. (2021). Split behavior of supervised machine learning algorithms for phishing URL detection. In *Advances in VLSI, communication, and signal processing* (pp. 497–505). Springer.
5. Gandotra, E., & Gupta, D. (2021). An efficient approach for phishing detection using machine learning. In *Multimedia security* (pp. 239–253). Springer.
6. Le, H., Pham, Q., Sahoo, D., & Hoi, S. C. (2018). *URLNet: Learning a URL representation with deep learning for malicious URL detection*. arXiv preprint, arXiv:1802.03162.
7. Hong, J., Kim, T., Liu, J., Park, N., & Kim, S. W. (2020). Phishing URL detection with lexical features and blacklisted domains. In *Adaptive autonomous secure cyber systems* (pp. 253–267). Springer.
8. Kumar, J., Santhanavijayan, A., Janet, B., Rajendran, B., & Bindhumadhava, B. S. (2020, January). Phishing website classification and detection using machine learning. In *2020 international conference on computer communication and informatics (ICCCI)* (pp. 1–6). IEEE.
9. Abutair, H. Y., & Belghith, A. (2017). Using case-based reasoning for phishing detection. *Procedia Computer Science, 109*, 281–288.
10. Rao, R. S., & Pais, A. R. (2019). Jail-Phish: An improved search engine based phishing detection system. *Computers & Security, 83*, 246–267.
11. Aljofey, A., Jiang, Q., Qu, Q., Huang, M., & Niyigena, J. P. (2020). An effective phishing detection model based on character level convolutional neural network from URL. *Electronics, 9*(9), 1514.
12. AlEroud, A., & Karabatis, G. (2020, March). Bypassing detection of URL-based phishing attacks using generative adversarial deep neural networks. In *Proceedings of the Sixth international workshop on security and privacy analytics* (pp. 53–60).
13. Althobaiti, K., Rummani, G., & Vaniea, K. (2019, June). A review of human-and computer-facing URL phishing features. In *2019 IEEE European symposium on security and privacy workshops (EuroS&PW)* (pp. 182–191). IEEE.
14. Gupta, B. B., Yadav, K., Razzak, I., Psannis, K., Castiglione, A., & Chang, X. (2021). A novel approach for phishing URLs detection using lexical based machine learning in a real-time environment. *Computer Communications, 175*, 47–57.
15. Sahoo, D., Liu, C., & Hoi, S. C. (2017). *Malicious URL detection using machine learning: A survey*. arXiv preprint, arXiv:1701.07179.
16. Chaurasia, V., & Pal, S. (2020). Applications of machine learning techniques to predict diagnostic breast cancer. *SN Computer Science, 1*(5), 1–11.
17. Yadav, D. C., & Pal, S. (2020). Prediction of thyroid disease using decision tree ensemble method. *Human-Intelligent Systems Integration, 2*(1), 89–95.
18. Chaurasia, V., & Pal, S. (2014). Performance analysis of data mining algorithms for diagnosis and prediction of heart and breast cancer disease. *Review of Research, 3*(8), 1–13.
19. Kharwar, A. R., & Thakor, D. V. (2022). An ensemble approach for feature selection and classification in intrusion detection using extra-tree algorithm. *International Journal of Information Security and Privacy (IJISP), 16*(1), 1–21.
20. Aggrawal, R., & Pal, S. (2020). Sequential feature selection and machine learning algorithm-based patient's death events prediction and diagnosis in heart disease. *SN Computer Science, 1*(6), 1–16.
21. Chaurasia, V., & Pal, S. (2022). An ensemble framework-stacking and feature selection technique for detection of breast cancer. *International Journal of Medical Engineering and Informatics, 14*(3), 240–251.
22. Pandey, M. K., & Pal, S. (2022). Evaluation of chronic myelogenous leukemia (CML) as the chronic phase of disease using machine learning techniques. *International Journal of Mechanical Engineering, 6*, 198–206.
23. Chaurasia, V., Pandey, M. K., & Pal, S. (2021, March). Prediction of presence of breast cancer disease in the patient using machine learning algorithms and SFS. In *IOP conference series: Materials science and engineering* (Vol. 1099, No. 1, p. 012003). IOP Publishing.
24. Shu, M., Zuo, J., Shen, M., Yin, P., Wang, M., Yang, X., Tang, J., Li, B., & Ma, Y. (2021). Improving the estimation accuracy of SPAD values for maize leaves by removing UAV hyperspectral image backgrounds. *International Journal of Remote Sensing, 42*(15), 5862–5881.
25. Yadav, D. C., & Pal, S. (2021). Performance based evaluation of algorithms on chronic kidney disease using hybrid ensemble model in machine learning. *Biomedical and Pharmacology Journal, 14*(3), 1633–1645.
26. Stančič, L., Oštir, K., & Kokalj, Ž. (2021). Fluvial gravel bar mapping with spectral signal mixture analysis. *European Journal of Remote Sensing, 54*(sup1), 31–46.