JCTA Journal of Computing Theories and Applications ISSN:3024-9104

Research Article

Web Phishing Classification using Combined Machine Learning Methods

Bambang Mahardhika Poerbo Waseso, Noor Ageng Setiyanto*

Faculty of Computer Science, Dian Nuswantoro University, Semarang, Central Java 50131, Indonesia; e-mail : 111201710352@mhs.dinus.ac.id, nasetiyanto@dsn.dinus.ac..id
* Corresponding Author: Noor Ageng Setiyanto

Abstract: Phishing is a crime that uses social engineering techniques, both in deceptive statements and technically, to steal consumers' personal identification data and financial account credentials. With the new Phishing machine learning approach, websites can be recognized in real-time. K-Nearest Neighbor(KNN) and Naïve Bayes (NB) are popular machine learning approaches. KNN and NB have their own strengths and weaknesses. By combining the two, deficiencies can be covered. So this study proposes to combine K-Nearest Neighbor with Naïve Bayes to classify phishing websites. Based on the results of the accuracy test of the combination of KNN with k=8 and Naïve Bayes, a maximum accuracy of 93.44% is produced. This result is 6.25% superior compared to using only one classifier.

Keywords: Phishing detection; Phishing classification; Naïve Bases; K-Nearest Neighbor; Combined Classifier.

1. Introduction

The Internet has become one of the main needs of the world community. Today's search, transmission and storage of data depend on internet technology. This can be seen from the number of internet users increasing yearly. Currently, the world's individual internet users have reached 66%, and statistics on the development of internet users reported by the International Telecommunication Union (ITU)[1] can be seen in Fig. 1. However, internet users will be very vulnerable to facing dangers that can cause financial loss, data fraud, loss of individual data, and loss of confidence in doing business. Increased cyber-crimes were reported by [2], where losses are predicted to reach $ 10.5 trillion in 2025.

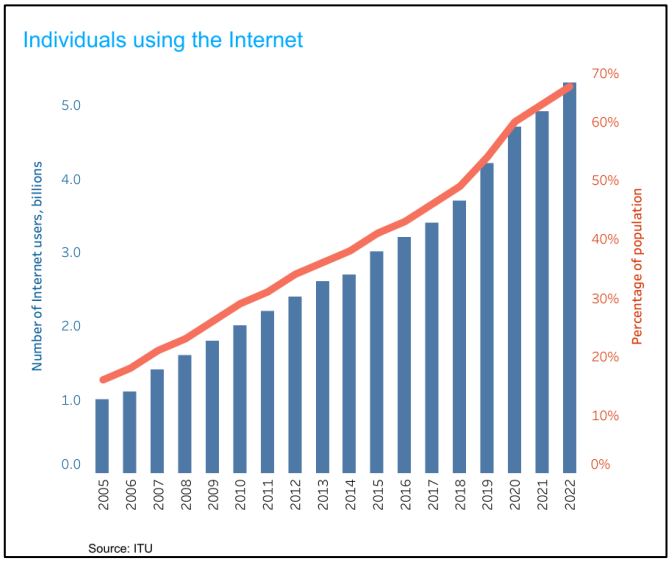


Figure 2. Individual Internet User Statistic by ITU.

Received: July, 7th 2023
Revised: August, 5th 2023
Accepted: August, 6th 2023
Published: August, 8th 2023

Crime on the internet is quite varied, one of which is phishing. Phishing is a crime that uses social engineering techniques and is a serious problem because it is easy to practice and increasingly sophisticated to imitate real websites.[3]. On research [4] it is said that theft using phishing methods from mid-March to May 2020, at which time the COVID-19 pandemic had occurred, was among the most frequent with a presentation of 86% of the total attacks. Therefore, automatic web phishing detection is needed to protect internet users. One of the ways to prove the classification of phishing websites is by using a heuristic approach, where several unique characteristics found on a website are used to identify phishing websites. [5]–[8]. The accuracy obtained from the heuristic approach depends on the uniqueness of the parameters on a website. Several parameters were analyzed in the heuristic method, namely text analysis, URL, visual content, network features, metadata and similarity to similar sites. Another way used to detect web phishing is classification with machine learning (ML). The ML approach to phishing web classification has several advantages, namely higher acquisition, better generalization, and more ability to learn new webs that have never been detected, but of course, the ML method is relatively more complex than the heuristic method.[9]–[11].

Several ML classification methods can be used to know phishing websites, such as K-Nearest Neighbor (KNN) and Naïve Bayes (NB). [11]–[16]. KNN excels because it is easy to implement and simple to understand. In addition, it does not require a complicated learning process because the training data is part of the classification process. But KNN needs to set the value of k, and it is less effective for data with high dimensions. Naïve Bayes is generally more efficient than KNN and more effective with high-dimensional datasets because it regards features as independent. A combination of two methods can cover these deficiencies. The voting method can use to combine at least three methods[17]–[20]. The voting method is one of the ensemble techniques in classification that is used to combine prediction results from several different classification models. In this method, each model provides predictions on the test data, and the final results are taken based on the majority of votes from the predictions given by each model[21]. But using too many ML methods results in more computationally expensive ones. Another technique that is simpler with simpler computations is the combination of the two methods as in research [22], [23]. This study aims to combine two methods, namely KNN and NB, with the combination technique.

2. Related Work

Several phishing studies have been conducted, one of which is research[16]. This study proposed the KNN method as a phishing web classifier. This research uses data from the phish tank public repository, which has 1,353 records with ten features, namely URL_Length, popUpWidnow, age_of_domain, hav-ing_IP_Address, web_traffic, Request_URL, SSLfinal_State, SFH, and URL_of_Anchor. The model designed in this study detects phishing attacks through URL classification and is tested empirically. The test was carried out at a value of k = 1 to 10, which resulted in an average value of 85.08%, while the best accuracy was 87.82% with k = 10.

Another study proposed by [13] discusses the use of machine learning techniques for classifying and detecting phishing websites. Various machine learning techniques such as logistic regression(LR), random forest(RF), NB, decision tree(DT), and KNN have been compared for the phishing URL classification task. The best results achieved were the Naïve Bayes Classifier with an accuracy of 0.98, a precision of 1, a recall of 0.95, and an F1-Score of 0.97. This study also discusses dataset processing, feature extraction techniques using host-based lexical analysis, and statistical analysis. The results show almost the same consistency in the various classification methods used.

Afandi et al. [24] focused more on research on COVID-19 phishing detection based on URLs using the KNN method. The research phase in this study, namely raw data input which was preprocessed, then feature extraction, training and testing were carried out and evaluated with 10 k-fold cross-validation. Two datasets are used, each consisting of 500 records, which are equally divided between phishing and legitimate classes. These two datasets were processed from several sources, namely Phishtank, SpyCloud, DomainTool and Kaggle. While the URL features used are Generic_TLD, URL_Length, Having_Sub_Domain, Prefix_Suffix and Having_Slash. Based on the test, high accuracy was produced, namely 97.80% for Dataset 1 and 99.60% for Dataset 2.

Another study was proposed by [25] which examined mobile phishing using the NB method. This research proposes a framework for detecting and distinguishing mobile applications with user permissions using the Naïve Bayesian method. With this machine learning approach, our framework effectively separates safe and dangerous applications. This approach involves a differentiating keylogger system that is completely based on the common quality behavior of all keyloggers, without relying on the keylogger's internal structure.

From some literature, it appears that these two methods are quite reliable in classifying web phishing, so this study tried to combine these methods with the combination method as described in the study[22], [23]. Further explanation of the proposed method is presented in Section 3.

3. Proposed Method

In this section, we describe the proposed method step by step and give an illustration with flow diagrams in Figure 2.

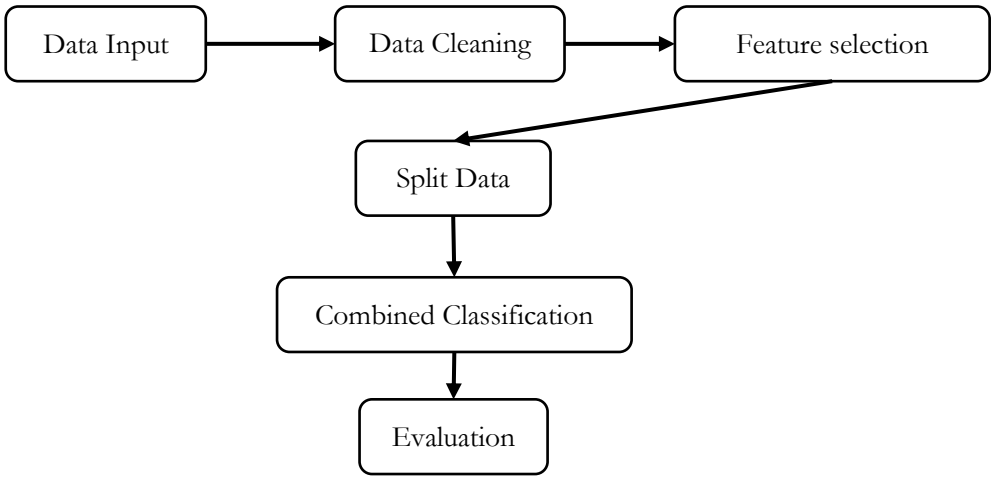


Figure 2. Proposed Method Illustration.

The first stage of input data is taken from the Kaggle website about web phishing datasets. This dataset has 31 attributes and 1354 records. Furthermore, a cleaning process is carried out, which aims to eliminate duplicate data. The number of records experienced a significant reduction, remaining only 677. Of the 31 attributes, nine attributes were selected, namely SFH, popUpWindow, SSLFinal_state, Request_URL, URL_of_Anchor, Web_Traffic, URL_Length, Age_of_domain, Having_IP_Address. Table 1 explains the explanation of each attribute. Furthermore, the data is transformed and consolidated into a form suitable for mining. The data did not go through this stage in this research because it was already numeric data with a scale of -1, 0, and 1. Where 1 means if the data includes Web Phishing characteristics and 0 or -1 if the data does not include Web Phishing characteristics.

Table 1. Attributes Explanation.

Attributes	Explanation
SFH (Server Form Handler)	Indicates whether the target URL uses a Server Form Handler (SFH) mechanism to submit data. SFH refers to the use of the POST or GET method in sending data from a web page.
popUpWindow	Indicates whether the target URL has a pop-up window appearing.
SSLFinal_state	Describes the SSL (Secure Sockets Layer) state of the target URL. SSL is used to secure communication between users and servers by encrypting data.
Request_URL	Contains the number of hyperlink elements in the target URL's HTML code. It describes the number of URL requests from the target web page.

URL_of_Anchor	Similar to the Request_URL attribute, but targets a different domain (anchor URL). It describes the number of hyperlinks pointing off the domain of the target web page.
Web_Traffic	The total number of requests and responses from the web server during the target web session.
URL_Length	Indicates the length of the target URL. This can be an indicator of phishing because often phishing URLs are of odd or suspicious length.
Age_of_domain	Indicates the age of the target domain in days. A newly created phishing domain may be more suspicious than a domain that has been around for a long time.
Having_IP_Address	Determines whether the target URL has a numeric IP address in its domain.

At the first classification stage tested using KNN and NB. For KNN, it is tested with a value of k = 1 to 10. The distance calculation uses the Eu-clidian equation, with the formula written in Eq. (1)[26]. While NB, of course, does not require a distance parameter like KNN but uses Eq. (2)[27].

$$Ed = \sqrt{\sum_{i=1}^{k}(x_i - y_i)^2} \tag{1}$$

$$P(H|E) = \frac{P(E|H) * P(H)}{P(E)} \tag{2}$$

Where x and y are two vectors or sets of numeric values that each have k elements. x and y are two points in k-dimensional space whose Euclidean distance will be calculated, then I is index. Then k is the dimension or number of elements in each x and y vector. In k-dimensional space, x and y each have k elements which will be calculated by the difference in squares in each dimension. The number of elements or dimensions is the same for both vectors, and k is a parameter that determines the dimensions of the space in which the points are located. $P(H|E)$ is the posterior probability or conditional probability of the hypothesis H occurring given the existence of evidence or information E. In a classification context, H is a class label (eg, "phishing" or "non-phishing" class in the case of email spam classification), while E is the feature or attribute vector of the data to be classified. $P(E|H)$ is the probability of occurrence of evidence or information E if the hypothesis H is true or if the data belongs to class H. In the context of classification, $P(E|H)$describes how likely certain features appear in class H. $P(H)$ is the prior probability of the H hypothesis, that is, the probability that H occurs without considering evidence or information E. This prior probability can be defined based on the frequency of occurrence of class H in the dataset. $P(E)$ is the probability of occurrence of evidence or information E in general, regardless of a particular class. This probability can be defined based on the frequency of occurrence of certain features in the dataset.

At the stage of merging the calcification method is carried out with the following stages:

1. NN classification is carried out first with predetermined input data testing and k parameters.
2. Calculate the distance between each training data and testing data using the Euclidean distance equation.
3. Sort the distances from closest to furthest.
4. Retrieve a predetermined number of k values (for example, 3 or 5) as the nearest neighbor of the testing data.
5. Determine the final class of data testing based on most classes from its nearest neighbors.
6. Then, classification is carried out using NB by using data testing results from the KNN stages as training data for Naïve Bayes classification.
7. The NB will use training data to calculate class probabilities from data testing based on existing attributes.

8. The final result of the NB classifier is the predicted class from the data testing

In brief, the above classification steps use two separate classification methods, namely KNN and NB. The process begins with the KNN stage to find the nearest neighbor from the data testing, then the results from the KNN stage are used as training data for classification using the Naïve Bayes method. The Naïve Bayes method then calculates the class probability from the testing data based on the existing attributes and gives the final prediction class.

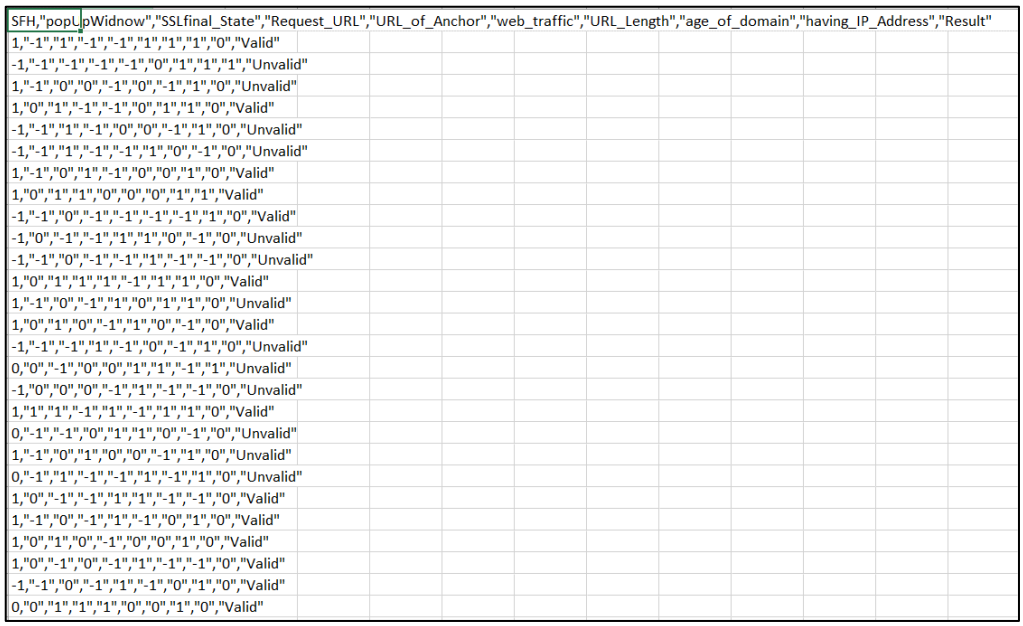
In this case, the KNN results are used as training data for NB so that the NB model can utilize information from KNN to produce more accurate predictions. KNN and NB are two different classification methods with different approaches to decision-making, so the combination of the two can provide more comprehensive information and produce more reliable predictions. The combination of KNN as training data for NB can provide benefits such as:

- They are reducing the bias and error associated with each classification method.
- Increase the accuracy and precision of predictions by combining the strengths of the two methods.
- Take advantage of variations in predictive results to improve overall performance.

In the final stage of the evaluation, it is measured using accuracy. Implementation of the method is presented in the next section.

4. Results and Discussion

As explained in the previous stage, the dataset has been cleaned and feature selection has been carried out. There are 677 records with nine features and two classes namely Valid (non-phishing web) and Invalid (phishing web). As an illustration of the data can be seen in Fig. 3.



```
SFH,"popUpWidnow","SSLfinal_State","Request_URL","URL_of_Anchor","web_traffic","URL_Length","age_of_domain","having_IP_Address","Result"
1,"-1","1","-1","-1","1","1","1","0","Valid"
-1,"-1","-1","-1","-1","0","1","1","1","Unvalid"
1,"-1","0","0","-1","0","-1","1","0","Unvalid"
1,"0","1","-1","-1","0","1","1","0","Valid"
-1,"-1","1","-1","0","0","-1","1","0","Unvalid"
-1,"-1","1","-1","-1","1","0","-1","0","Unvalid"
1,"-1","0","1","-1","0","0","1","0","Valid"
1,"0","1","1","0","0","0","1","1","Valid"
-1,"-1","0","-1","-1","-1","-1","1","0","Valid"
-1,"0","-1","-1","1","1","0","-1","0","Unvalid"
-1,"-1","0","-1","-1","1","-1","-1","0","Unvalid"
1,"0","1","1","1","-1","1","1","0","Valid"
1,"-1","0","-1","1","0","1","1","0","Unvalid"
1,"0","1","0","-1","1","0","-1","0","Valid"
-1,"-1","-1","1","-1","0","-1","1","0","Unvalid"
0,"0","-1","0","0","1","1","-1","1","Unvalid"
-1,"0","0","0","-1","1","-1","-1","0","Unvalid"
1,"1","1","-1","1","-1","1","1","0","Valid"
0,"-1","-1","0","1","1","0","-1","0","Unvalid"
1,"-1","0","1","0","0","-1","1","0","Unvalid"
0,"-1","1","-1","-1","1","-1","1","0","Unvalid"
1,"0","-1","-1","1","1","-1","-1","0","Valid"
1,"-1","0","-1","1","-1","0","1","0","Valid"
1,"0","1","0","-1","0","0","1","0","Valid"
1,"0","-1","0","-1","1","-1","-1","0","Valid"
-1,"-1","0","-1","1","-1","0","1","0","Valid"
0,"0","1","1","1","0","0","1","0","Valid"
```

Figure 3. Screenshot of CSV Dataset.

In this study the dataset was split into training and testing data with the respective percentages being 80% for training and 20% for testing. Furthermore, an evaluation is carried out with an assessment of accuracy. The results of the evaluation of each method are presented in Table 2.

Table 2. Accuracy results of KNN and NB for the proportion of 80:20.

k	KNN	NB
1	84.73 %	85.71%
2	79.80 %	
3	86.21 %	
4	85.22 %	

5	87.19 %	
6	85.71 %	
7	85.22 %	
8	86.70 %	
9	86.21 %	
10	83.74 %	
Average	85.07%	85.71%

Table 2 shows the results of testing on individual classifiers. It can be seen that the KNN method has an accuracy performance of between 79.80% and 87.19% if the average accuracy is 85.07%. This is not better with NB accuracy, which is 85.71%. In the second experiment, we tried to use the combination method, where KNN became the first classifier and NB became the second classifier, the second classifier. The implementation of the combination of these two methods uses the Rapid Miner, which is presented in Fig. 4. The results of the second and third experiments are presented in Table 3.

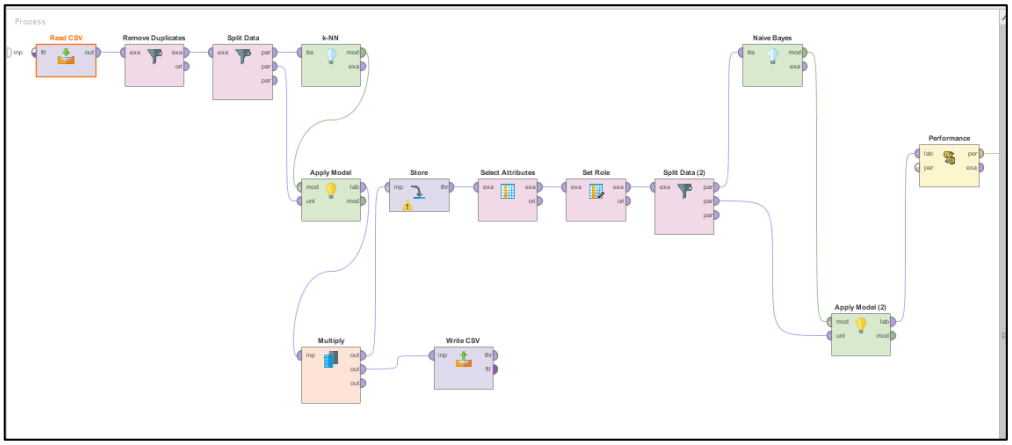


Figure 4. Implemntation of Combined Method using Rapid Miner.

Table 3. Accuracy results of Combined Method.

Classifier 1	Classifier 2	Combined Accuracy	Individual Method Accuracy	
			KNN	NB
KNN, k=1	NB	83.61 %	84.73 %	85.71%
KNN, k=2	NB	81.97 %	79.80 %	
KNN, k=3	NB	86.89 %	86.21 %	
KNN, k=4	NB	88.52 %	85.22 %	
KNN, k=5	NB	85.25 %	87.19 %	
KNN, k=6	NB	90.16 %	85.71 %	
KNN, k=7	NB	90.16 %	85.22 %	
KNN, k=8	NB	93.44 %	86.70 %	
KNN, k=9	NB	90.16 %	86.21 %	
KNN, k=10	NB	86.89 %	83.74 %	

Based on the data presented in Table 3 it appears that the random voting method does not always provide an increase in value as in k = 1, k = 5, but the majority increases accuracy, where the highest accuracy is 93.44%. This result is 6.25% superior compared to the individual classifier.

5. Conclusions

The research findings have demonstrated the effectiveness of combining classifiers in achieving improved accuracy. This technique of combining classifiers is relatively straightforward and offers several advantages, including reducing bias and errors associated with individual classification methods, enhancing accuracy and precision of predictions by leveraging the strengths of both methods, and capitalizing on variations in predictive outcomes to

enhance overall performance. These two methods have been validated and shown to increase performance accuracy.

Author Contributions: Conceptualization: B.M.P.W. and N.A.S.; methodology: B.M.P.W.; software: B.M.P.W.; validation, B.M.P.W. and N.A.S.; formal analysis, B.M.P.W.; investigation: N.A.S.; resources: B.M.P.W.; writing—original draft preparation: B.M.P.W.; writing—review and editing: N.A.S.; supervision: N.A.S,; project administration: B.M.P.W.

Funding: This research received no external funding

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

References

[1] "Statistics." https://www.itu.int/en/ITU-D/Statistics/Pages/stat/default.aspx (accessed Aug. 01, 2023).
[2] Cybersecurity Ventures, "Cybercrime To Cost The World $10.5 Trillion Annually By 2025," *Cybercrime Magazine*, Sausalito, Nov. 2020. [Online]. Available: https://cybersecurityventures.com/hackerpocalypse-cybercrime-report-2016/
[3] P. Yang, G. Zhao, and P. Zeng, "Phishing Website Detection Based on Multidimensional Features Driven by Deep Learning," *IEEE Access*, vol. 7, pp. 15196–15209, 2019, doi: 10.1109/ACCESS.2019.2892066.
[4] H. S. Lallie *et al.*, "Cyber security in the age of COVID-19: A timeline and analysis of cyber-crime and cyber-attacks during the pandemic," *Comput. Secur.*, vol. 105, p. 102248, 2021, doi: 10.1016/j.cose.2021.102248.
[5] S. Gastellier-Prevost, G. G. Granadillo, and M. Laurent, "Decisive Heuristics to Differentiate Legitimate from Phishing Sites," in *2011 Conference on Network and Information Systems Security*, May 2011, pp. 1–9. doi: 10.1109/SAR-SSI.2011.5931389.
[6] S. Bhattacharyya, C. kumar Pal, and P. kumar Pandey, "Detecting Phishing Websites, a Heuristic Approach," *Int. J. Latest Eng. Res. Appl.*, vol. 3, pp. 120–129, 2017, [Online]. Available: www.ijlera.com
[7] J. Solanki and R. G. Vaishnav, "Website Phishing Detection using Heuristic Based Approach," *Int. Res. J. Eng. Technol.*, pp. 2044–2048, 2016, [Online]. Available: www.irjet.net
[8] C. M. R. da Silva, E. L. Feitosa, and V. C. Garcia, "Heuristic-based strategy for Phishing prediction: A survey of URL-based approach," *Comput. Secur.*, vol. 88, p. 101613, 2020, doi: 10.1016/j.cose.2019.101613.
[9] A. Safi and S. Singh, "A systematic literature review on phishing website detection techniques," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 35, no. 2, pp. 590–611, 2023, doi: 10.1016/j.jksuci.2023.01.004.
[10] M. N. Alam, D. Sarma, F. F. Lima, I. Saha, R. E. Ulfath, and S. Hossain, "Phishing attacks detection using machine learning approach," *Proc. 3rd Int. Conf. Smart Syst. Inven. Technol. ICSSIT 2020*, no. Icssit, pp. 1173–1179, 2020, doi: 10.1109/ICSSIT48917.2020.9214225.
[11] S. A. Khan, W. Khan, and A. Hussain, "Phishing Attacks and Websites Classification Using Machine Learning and Multiple Datasets (A Comparative Analysis)," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12465 LNAI, 2020, pp. 301–313. doi: 10.1007/978-3-030-60796-8_26.
[12] M. Mithra Raj and J. A. Arul Jothi, "Website Phishing Detection Using Machine Learning Classification Algorithms," 2022, pp. 219–233. doi: 10.1007/978-3-031-19647-8_16.
[13] J. Kumar, A. Santhanavijayan, B. Janet, B. Rajendran, and B. S. Bindhumadhava, "Phishing Website Classification and Detection Using Machine Learning," in *2020 International Conference on Computer Communication and Informatics (ICCCI)*, Jan. 2020, pp. 1–6. doi: 10.1109/ICCCI48352.2020.9104161.
[14] A. K. Dutta, "Detecting phishing websites using machine learning technique," *PLoS One*, vol. 16, no. 10, p. e0258361, Oct. 2021, doi: 10.1371/journal.pone.0258361.
[15] S. Alnemari and M. Alshammari, "Detecting Phishing Domains Using Machine Learning," *Appl. Sci.*, vol. 13, no. 8, p. 4649, Apr. 2023, doi: 10.3390/app13084649.
[16] T. A. Assegie, "K-Nearest Neighbor Based URL Identification Model for Phishing Attack Detection," *Indian J. Artif. Intell. Neural Netw.*, vol. 1, no. 2, pp. 18–21, 2021, doi: 10.54105/ijainn.b1019.041221.
[17] A. Manconi, G. Armano, M. Gnocchi, and L. Milanesi, "A Soft-Voting Ensemble Classifier for Detecting Patients Affected by COVID-19," *Appl. Sci.*, vol. 12, no. 15, 2022, doi: 10.3390/app12157554.
[18] S. Chatterjee and Y.-C. Byun, "Voting Ensemble Approach for Enhancing Alzheimer's Disease Classification," *Sensors*, vol. 22, no. 19, p. 7661, Oct. 2022, doi: 10.3390/s22197661.
[19] F. Ülker and A. Küçüker, "Probabilistic weighted voting model using multiple machine learning methods for fault detection and classification," *COMPEL - Int. J. Comput. Math. Electr. Electron. Eng.*, vol. 41, no. 5, pp. 1542–1565, Aug. 2022, doi: 10.1108/COMPEL-06-2021-0200.
[20] S. H. Ahammad *et al.*, "Phishing URL detection using machine learning methods," *Adv. Eng. Softw.*, vol. 173, no. July, p. 103288, 2022, doi: 10.1016/j.advengsoft.2022.103288.
[21] Z. Mushtaq, M. F. Ramzan, S. Ali, S. Baseer, A. Samad, and M. Husnain, "Voting Classification-Based Diabetes Mellitus Prediction Using Hypertuned Machine-Learning Techniques," *Mob. Inf. Syst.*, vol. 2022, pp. 1–16, Mar. 2022, doi: 10.1155/2022/6521532.
[22] S. Marianingsih, F. Utaminingrum, and F. A. Bachtiar, "Road surface types classification using combination of K-nearest neighbor and Naïve Bayes based on GLCM," *Int. J. Adv. Soft Comput. its Appl.*, vol. 11, no. 2, pp. 15–27, 2019.
[23] R. G. Devi and P. Sumanjani, "Improved classification techniques by combining KNN and Random Forest with Naive Bayesian classifier," *ICETECH 2015 - 2015 IEEE Int. Conf. Eng. Technol.*, no. March, pp. 1–4, 2015, doi: 10.1109/ICETECH.2015.7274997.

[24] N. A. Afandi and Isredza Rahmi A Hamid, "Covid-19 Phishing Detection Based on Hyperlink Using K-Nearest Neighbor (KNN) Algorithm," *Appl. Inf. Technol. Comput. Sci.*, vol. 2, no. 2, pp. 287–301, 2021, [Online]. Available: https://publisher.uthm.edu.my/periodicals/index.php/aitcs/article/view/2317

[25] N. Kumar and P. Chaudhary, "Mobile phishing detection using naive bayesian algorithm," *IJCSNS Int. J. Comput. Sci. Netw. Secur.*, vol. 17, no. 7, pp. 142–147, 2017.

[26] D. R. Ignatius Moses Setiadi *et al.*, "Comparison of SVM, KNN, and NB Classifier for Genre Music Classification based on Metadata," in *2020 International Seminar on Application for Technology of Information and Communication (iSemantic)*, Sep. 2020, pp. 12–16. doi: 10.1109/iSemantic50169.2020.9234199.

[27] A. Susanto, Z. H. Dewantoro, C. A. Sari, D. R. I. M. Setiadi, E. H. Rachmawanto, and I. U. W. Mulyono, "Shallot Quality Classification using HSV Color Models and Size Identification based on Naive Bayes Classifier," *J. Phys. Conf. Ser.*, vol. 1577, no. 1, 2020, doi: 10.1088/1742-6596/1577/1/012020.