Scientific Journal of Informatics
Vol. 11, No. 2, May 2024

p-ISSN 2407-7658 https://journal.unnes.ac.id/journals/sji/index e-ISSN 2460-0040

Performance Analysis of Support Vector Classification and Random Forest in Phishing Email Classification

Chaerul Umam[1*], Lekso Budi Handoko[2], Folasade Olubusola Isinkaye[3]
[1,2]Department of Informatics Engineering, Faculty of Computer Science, University of Dian Nuswantoro Semarang, Indonesia
[3]Department of Computer Science, Faculty of Science, Ekiti State University, Ado-Ekiti, Nigeria

Abstract.
Purpose: This study aims to conduct a performance analysis of phishing email classification system using machine learning algorithms, specifically Random Forest and Support Vector Classification (SVC).
Methods/Study design/approach: The study employed a systematic approach to develop a phishing email classification system utilizing machine learning algorithms. Implementation of the system was conducted within the Jupyter Notebook IDE using the Python programming language. The dataset, sourced from kaggle.com, comprised 18,650 email samples categorized into secure and phishing emails. Prior to model training, the dataset was divided into training and testing sets using three distinct split percentages: 60:40, 70:30, and 80:20. Subsequently, parameters for both the Random Forest and Support Vector Classification models were carefully selected to optimize performance. The TF-IDF Vectorizer method was employed to convert text data into vector form, facilitating structured data processing.
Result/Findings: The study's findings reveal notable performance accuracies for both the Random Forest model and Support Vector Classification across varying data split percentages. Specifically, the Support Vector Classification consistently outperforms the Random Forest model, achieving higher accuracy rates. At a 70:30 split percentage, the Support Vector Classification attains the highest accuracy of 97.52%, followed closely by 97.37% at a 60:40 split percentage.
Novelty/Originality/Value: Comparisons with previous studies underscored the superiority of the Support Vector Classification model. Therefore, this research contributes novel insights into the effectiveness of this machine learning algorithm~~s~~ in phishing email classification, emphasizing its potential in enhancing cybersecurity measures.

Keywords: Email phishing, Classification, Support vector classification, Random forest.

Received April 2024 / **Revised** April 2024 / **Accepted** May 2024

INTRODUCTION

Technological developments that occur in the current era mean that information can be obtained from anywhere easily and efficiently [1], [2]. With the development and ease of access to information, a system is needed that can secure our data and privacy [3], [4]. Because, with the ease of obtaining information, cybercrime can also occur that threatens our digital privacy via the internet [5]. One example of cybercrime that can occur is phishing. Phishing is a way of exploiting internet users to obtain important and sensitive information from these users [6], which can be used irresponsibly [7]. One way of spreading phishing that can occur is through email [8], in the process of spreading email using malicious attachments or links sent via email which, if clicked, can steal user data [9], [10]. Therefore, it is important to carry out the process of sorting phishing emails so as not to cause harm to ourselves both in terms of privacy and finances.

Machine learning is a family of artificial intelligence (AI) that allows computers to learn independently to find patterns from data without having to be programmed first [11]–[15]. So, by using machine learning, valuable information from data can be easily extracted to help the decision-making process [16]. Examples of methods or algorithms found in the machine learning process are random forest and support vector classification. Random forest is an ensemble learning method or learning method that combines several models where the model is built based on a combination of decision tree models to form a forest scheme

[*] Corresponding author.
Email addresses: chaerul@dsn.dinus.ac.id (Umam), handoko@dsn.dinus.ac.id (Handoko), folasade.isinkaye@eksu.edu.ng (Isinkaye)
DOI: 10.15294/sji.v11i2.3301

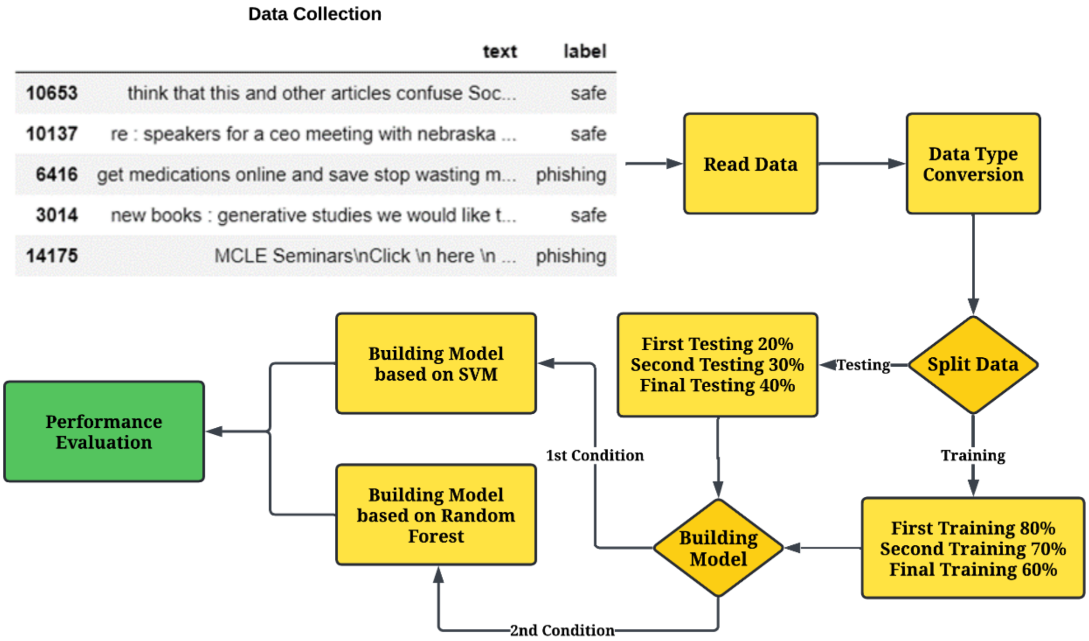
[17], [18]. Meanwhile, support vector classification (SVC) is an implementation of the SVM algorithm which is specifically used in the data classification process, which is conceptually the same as creating and determining optimal support vectors which are useful for assisting the data classification process [19], [20]. In processing text data, so that the model that is built can carry out the pattern recognition process from the data, the process of converting the data into vector form is carried out, so that it can change the form of data from unstructured to structured [21]. To carry out the process of converting text data into vector form, a method can be used, namely TF-IDF Vectorizer, which in this method will convert text into vector form using statistical methods [22].

In previous research conducted by Rao et. al in 2021 [25] discussed the process of classifying spam or spam messages using the Logistic Regression algorithm. The aim of this research is to build a machine learning model with a Logistic Regression algorithm to be able to classify ham or spam messages. The results obtained from this research were that after the testing process was carried out, the model testing accuracy was 98%. In previous research conducted by Ma et. al in 2020 [26], discussed the process of classifying spam emails using naïve Bayes classifiers and support vector machines. The aim of this research is to build a machine learning-based spam email classification model and compare the NBC and SVM methods to see which one produces better testing accuracy. The results obtained in this research are that after the testing process was carried out, the NBC algorithm obtained the best accuracy.

In this research, the process of building a phishing email classification model will be carried out using the Random Forest and Support Vector Classification algorithms. The aim of this research is to build a classification model that can help many people sort out phishing emails so they don't suffer losses. The purpose of this research process is also to compare the performance of Random Forest and Support Vector Classification to carry out the phishing email classification process and determine which algorithm is best in carrying out the classification process. In this research, testing will also be carried out using 3 different data division schemes, namely 50:10:40 (50% training, 10% validation and 40% testing), 60:10:30 (60% training, 10% validation and 30% % testing) and 70:10:20 (70% training, 10% validation, and 20% testing), which aims to find out with what kind of data sharing scheme the model can produce optimal accuracy. The purpose of using random forest is because random forest is a form of ensemble learning which is composed of decision tree models, so it is hoped that it can provide maximum results [23]. Meanwhile, the purpose of using the SVC algorithm is because this algorithm is an implementation of the SVM algorithm which is very good in data processing and this SVC algorithm is specifically used for the classification process [24].

METHODS

Classification is a method where an algorithm model carries out a pattern recognition process and can carry out the process of determining labels from a given dataset [27]. This method is a supervised learning method where the model carries out a training process to guess the target data [28]. The flow of the classification process carried out is presented in Figure 1. Figure 1 shows the flow of the classification process carried out. As seen in Figure 1, the first thing to do is read the data or dataset that will be used for the training, validation and model testing processes. After the data is converted into vector form, the next process is to divide the data, which in this study the data will be divided into 3 variations, namely 60:40 (60% training and 40% testing), 70:30 (70% training and 30% testing) and 80:20 (80% training and 20% testing). The purpose of dividing data into several variations is to find out what percentage of data division can obtain optimal accuracy values. Then, after the data sharing process is carried out, the process of building a machine learning model will be carried out using the Random Forest and Support Vector Classification algorithms.

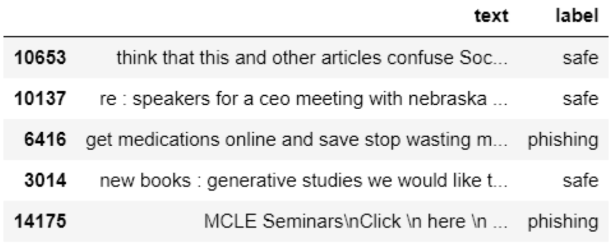


	text	label
10653	think that this and other articles confuse Soc...	safe
10137	re : speakers for a ceo meeting with nebraska ...	safe
6416	get medications online and save stop wasting m...	phishing
3014	new books : generative studies we would like t...	safe
14175	MCLE Seminars\nClick \n here \n ...	phishing

Figure 1. Proposed scheme

Data Collection

In this research, the dataset used is a CSV dataset obtained from the kaggle.com website, which has a total of 18650 data and has 2 main classes, namely secure emails and phishing emails. With data distribution, 11322 email data is safe and 7328 data is phishing email. For visualization of the dataset used, it is given in Figure 2.



	text	label
10653	think that this and other articles confuse Soc...	safe
10137	re : speakers for a ceo meeting with nebraska ...	safe
6416	get medications online and save stop wasting m...	phishing
3014	new books : generative studies we would like t...	safe
14175	MCLE Seminars\nClick \n here \n ...	phishing

Figure 2. Data collection

Figure 2 shows a sample of the dataset used in this research. After the dataset for the machine learning development process is prepared, the next step is to convert the text data into vector data (Vector Space Model) so that the data can be used for the model training and testing process. The purpose of data conversion is also to change data from unstructured to structured [21]. In this research, the process of converting text into vector form will use the TF-IDF Vectorizer method, where this method will convert text into vector form using statistical methods [22]. So, with this method the weight value of the words obtained is searched based on the importance of the data in the document [29]. In the TF-IDF calculation, the TF value is estimated or the value of how often the word appears and the IDF or how uniquely the word appears in the document. The calculation of the TF-IDF value is given in equations (2) until (3).

$$\mathrm{TF(W, Doc)} = \frac{\sum \ (\text{Word appears in Doc})}{\sum \ (\text{All Words in Doc})} \tag{1}$$

$$\mathrm{IDF\ (W, Corp)} = \mathrm{Log}\ \frac{\mathrm{N}}{(1 + \mathrm{DF\ (W, Corp)})} \tag{2}$$

$$\mathrm{TF \cdot IDF\ (W, Doc, Corp)} = \mathrm{TF\ (W, Doc)} * \mathrm{IDF\ (W, Corp)} \tag{3}$$

Random Forest

This research introduces a novel approach of random forest method to ensemble learning by aggregating the predictions of multiple decision trees, thereby enhancing the model's robustness and accuracy [30]. One key innovation lies in the definition of the number of trees, denoted as N. Following initialization, the algorithm randomly selects a subset of features from the dataset. For each feature, entropy is calculated based on the probability distribution of classes. Subsequently, the algorithm computes the information gain for each feature [31]. The node with the highest information gain is selected as the splitting criterion, leading to the creation of sub-nodes. This process iterates until a single decision tree is formed, meeting the minimum sample requirement. Notably, this process repeats N times, generating a forest of decision trees. The random forest formula is given in equation (4).

The term "N Trees" refers to the number of decision trees in the Random Forest algorithm, where each tree is constructed through a series of steps. Initially, a subset of features (X) is randomly selected from the dataset. For each feature, the algorithm calculates the entropy (Ent) of the target variable (Z) using the formula $Ent(Z) = -\sum(x = 1)^n P_x \, log2(P_x)$, where P_x represents the probability of each class. Then, the conditional entropy ($Ent(Z, I)$) is computed based on the selected feature, and the information gain ($InformationGain(C, Z)$) is determined as the difference between the entropy of the target variable and the conditional entropy. The algorithm selects the feature that yields the highest information gain to split the nodes and form sub-nodes. This process continues recursively until a tree is fully grown, and the minimum number of samples required at each leaf node is reached. Finally, the algorithm repeats this process N times, where N represents the desired number of trees, to form a Random Forest with N decision trees.

The equation used is as in (4)

$$\mathrm{Prediction(Z)} = \frac{(\mathrm{decTree\_1(Z)} + \mathrm{decTree\_2(Z)} + \ldots + \mathrm{decTree\_N(Z)})}{\mathrm{N}} \quad (4)$$

Where Z is a prediction feature, while $decTree\_N(Z)$ is a decision tree model used to predict feature Z.

Support Vector Classification (SVC)

Support Vector Classification (SVC) introduces a novel approach in determining the margin (Margin) for classification [24], [32], [33]. This margin calculation incorporates the $weight$ (w), $data\ point$ ($n(i)$), and $bias$ (b), which are iteratively adjusted until convergence [34]. Notably, when the margin falls below a threshold, adjustments are made to both the weight and bias parameters to ensure proper classification. This iterative refinement mechanism enhances the algorithm's capability to classify data points accurately, thus improving the overall performance of the SVC algorithm [35], [36]. The hyperplane calculation formula is given in equation (5). Determine w ($weight$), b ($bias$), ($learning\ rate$), Z ($regularization\ param$):

Initially, the margin (Margin) is computed using the equation $m(i) * (w * n(i) * b)$, where $m(i)$ represents the data point's class label, w denotes the weight vector, $n(i)$ signifies the input data point, and b is the bias term. Subsequently, if the margin falls below 1, indicating misclassification, the algorithm updates the $weight$ (w) and $bias$ (b) parameters using the learning $rate$ (α) and the gradient descent approach. Specifically, w is adjusted by $\alpha * (m(i)n(i) - 2Zw)$, where Z is the regularization parameter, and b is updated by $\alpha m(i)$. Conversely, if the margin is greater than or equal to 1, no updates are made to w, and b remains unchanged. The decision $function$ ($f(x)$) is then computed as $w * x * b$, where x represents the input data point. Finally, the algorithm classifies the data point as 1 if $f(x)$ is greater than 0, indicating it belongs to the positive class, otherwise, it is classified as 0.

With hyperplane calculations as in (5)

$$\omega * \mathrm{Z} + \mathrm{b} = 0 \quad (5)$$

Where ω is the hyperplane weight vector, X is the feature used and b is the bias (shift) value. After building the classification model that will be used, the next process is to train, validate and test the model. So, after the model that has been built can be tested, the next process can calculate the model's performance using a metric called the confusion matrix. Confusion Matrix is a useful metric for evaluating the performance of the classification model built [37]. To carry out the process of calculating model performance, you can use the values resulting from the confusion matrix, namely the precision or accuracy of the model, recall or the model's ability to guess all data classes well and f1-score. or the balance value between precision and recall. Calculation of precision, recall and f1-score values are given in equation (6) until (8).

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Postive} + \text{False Positive}} \tag{6}$$

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Postive} + \text{False Negative}} \tag{7}$$

$$\text{F1} - \text{Score} = \frac{\text{True Positive}}{\text{True Postive} + \text{False Positive}} \tag{8}$$

RESULTS AND DISCUSSIONS

The classification process in this research uses the Jupyter Notebook IDE for writing program code and the Python programming language to be able to carry out the process of implementing a phishing email classification system. To carry out the process of developing a classification model, first after the model has been prepared, a model training process is carried out using training data which has previously been divided into 3 percentage data division schemes. The parameters used in the Random Forest and Support Vector Classification models are given in Table 1.

Table 1. Analysis of parameter per model

Model	Random Forest	Support Vector Classification (SVC)

Parameters	n_estimators=500,
	criterion='entropy',
	max_depth=None,
	min_samples_split=2,
	min_samples_leaf=1,
	min_weight_fraction_leaf=0.0,
	max_leaf_nodes=None,
	min_impurity_decrease=0.0,
	bootstrap=True,
	oob_score=False,
	n_jobs=-1,
	random_state=42,
		C=1.0,
		kernel='sigmoid',
		degree=3,
		gamma=1.0,
		coef0=0.0,
		shrinking=True,
		probability=False,
		tol=0.001,
		class_weight='balanced',
		verbose=False,
		max_iter=-1,
		decision_function_shape='ovr',
		random_state=None

Table 1 shows the parameters used for the training and testing process of the classification model that was built. After the data is carried out, the training process is carried out using varying percentages of training data, namely 60%, 70% and 80%, then the testing process was carried out using a model that has been trained with test data which has varying percentages of data, namely 40%, 30% and 20 %. The results of the data classification testing process that has been carried out are presented in Table 2.

Table 2. Experiment results

Split Percentage	**Testing Accuracy**	
	Random Forest	**Support Vector Classification**
60:40	96.35%	97.37%
70:30	96.57%	97.52%
80:20	96.41%	97.40%

Table 2 shows the results obtained after carrying out the model testing process. As seen in Table 2, the Random Forest model gets the best testing accuracy when carried out with a scheme of dividing 70% training data and 30% testing, namely 96.57%. Meanwhile, the model with the Support Vector Classification algorithm gets the most optimal accuracy of 97.52% when using a data distribution scheme of 70% training and 30% testing. From these results, it can be seen that the random forest and support vector classification models that are built can obtain the most optimal accuracy values when the training and testing processes are carried out with a data sharing percentage scheme of 70% training and 30% testing.

From table 2 it can also be seen that the best model that gets the most optimal accuracy value is Support Vector Classification (SVC), which is 97.52%. In this study, although the accuracy value obtained was close to the maximum value of 100%, it still could not produce 100% accuracy. The reason is, in the classification process, the data used is original and realistic email message data that we usually encounter in our emails. Therefore, it is possible that the dataset used still contains data that is quite similar so that it can affect the accuracy of the model being built.

Previously there were several studies that discussed the process of classifying phishing emails using machine learning methods. Previous research that has been carried out is research conducted by Mathur et. al in 2022 [35]. In this research, the email classification process was carried out using the Naïve Bayes Classifier algorithm, Support Vector Machine, Random Forest, XGBoost. The results gotten in this research were that using the NBC algorithm obtained a testing accuracy of 94%, using the SVM algorithm obtained a testing accuracy of 97%, using the RF algorithm obtained a testing accuracy of 97%, while using the XGBoost algorithm obtained a testing accuracy of 92%. From these results, we had been achieved high result where this research succeeded in increasing the accuracy of the SVM algorithm used, namely 97.52% and for the Random Forest algorithm it also got almost the same value, rounded up to around 97%.

CONCLUSION

After the process of building a machine learning classification model through the process of training and testing the model using the Random Forest and Support Vector Classification algorithms on a split dataset, the result showed that the Random Forest model produced maximum accuracy when the training process was carried out using a split data scheme of 70% training and 30% test with an accuracy value of 96.57%, while the Support Vector Classification model achieved an optimal accuracy results when carried out with a data sharing process of 70% training and 30% test, namely 97.52%. These findings indicate that the SVC model outperforms the Random Forest algorithm by 1.15% in accuracy. From these results, it could be concluded that Support Vector Classification model that was built can perform classification tasks better when compared ~~to~~ with the Random Forest algorithm.

REFERENCES

[1] L. Weili, H. Khan, I. khan, and L. Han, "The impact of information and communication technology, financial development, and energy consumption on carbon dioxide emission: evidence from the Belt and Road countries", *Environmental Science and Pollution Research*, vol. 29, no. 19, pp. 27703–27718, 2022, doi: 10.1007/s11356-021-18448-5.

[2] K. Okundaye, S. K. Fan, and R. J. Dwyer, "Impact of information and communication technology in Nigerian small-to medium-sized enterprises", *Journal of Economics, Finance and Administrative Science*, vol. 24, no. 47, pp. 29–46, 2019, doi: 10.1108/JEFAS-08-2018-0086.

[3] S. Mbonihankuye, A. Nkunzimana, and A. Ndagijimana, "Healthcare Data Security Technology: HIPAA Compliance", *Wirel Commun Mob Comput*, vol. 2019, pp. 1–7, 2019, doi: 10.1155/2019/1927495.

[4] P. Yang, N. Xiong, and J. Ren, "Data Security and Privacy Protection for Cloud Storage: A Survey", *IEEE Access*, vol. 8, pp. 131723–131740, 2020, doi: 10.1109/ACCESS.2020.3009876.

[5] Y. Li and Q. Liu, "A comprehensive review study of cyber-attacks and cyber security; Emerging trends and recent developments", *Energy Reports*, vol. 7, pp. 8176–8186, 2021, doi: 10.1016/j.egyr.2021.08.126.

[6] S. Salloum, T. Gaber, S. Vadera, and K. Shaalan, "A Systematic Literature Review on Phishing Email Detection Using Natural Language Processing Techniques", *IEEE Access*, vol. 10, pp. 65703–65727, 2022, doi: 10.1109/ACCESS.2022.3183083.

[7] P. Sharma, B. Dash, and M. F. Ansari, "Anti-Phishing Techniques – A Review of Cyber Defense Mechanisms", *IJARCCE*, vol. 11, no. 7, 2022, doi: 10.17148/IJARCCE.2022.11728.

[8] S. Salloum, T. Gaber, S. Vadera, and K. Shaalan, "Phishing Email Detection Using Natural Language Processing Techniques: A Literature Survey", *Procedia Comput Sci*, vol. 189, pp. 19–28, 2021, doi: 10.1016/j.procs.2021.05.077.

[9] J. Petelka, Y. Zou, and F. Schaub, "Put Your Warning Where Your Link Is", *in Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, New York, NY, USA: ACM, pp. 1–15, 2019, doi: 10.1145/3290605.3300748.

[10] S. R. Martin, J. J. Lee, and B. L. Parmar, "Social distance, trust and getting 'hooked': A phishing expedition", *Organ Behav Hum Decis Process*, vol. 166, pp. 39–48, 2021, doi: 10.1016/j.obhdp.2019.08.001.

[11] A. Kilic, "Artificial Intelligence and Machine Learning in Cardiovascular Health Care", *Ann Thorac Surg*, vol. 109, no. 5, pp. 1323–1329, 2020, doi: 10.1016/j.athoracsur.2019.09.042.

[12] A. Kilic, "Artificial Intelligence and Machine Learning in Cardiovascular Health Care", *Ann Thorac Surg*, vol. 109, no. 5, pp. 1323–1329, 2020, doi: 10.1016/j.athoracsur.2019.09.042.

[13] S. J. MacEachern and N. D. Forkert, "Machine learning for precision medicine", *Genome*, vol. 64, no. 4, pp. 416–425, 2021, doi: 10.1139/gen-2020-0131.

[14] I. H. Sarker, A. S. M. Kayes, S. Badsha, H. Alqahtani, P. Watters, and A. Ng, "Cybersecurity data science: an overview from machine learning perspective", *J Big Data*, vol. 7, no. 1, pp. 41, 2020, doi: 10.1186/s40537-020-00318-5.

[15] Q. Bi, K. E. Goodman, J. Kaminsky, and J. Lessler, "What is Machine Learning? A Primer for the Epidemiologist", *Am J Epidemiol*, vol. 188, no. 12, pp. 2222–2239, 2019, doi: 10.1093/aje/kwz189.

[16] D. Shah, S. Patel, and S. K. Bharti, "Heart Disease Prediction using Machine Learning Techniques", *SN Comput Sci*, vol. 1, no. 6, p. 345, 2020, doi: 10.1007/s42979-020-00365-y.

[17] A. Parmar, R. Katariya, and V. Patel, "A Review on Random Forest: An Ensemble Classifier", *in Lecture Notes on Data Engineering and Communications Technologies*, Springer Science and Business Media Deutschland GmbH, vol. 26, pp. 758–763, 2019, doi: 10.1007/978-3-030-03146-6_86.

[18] T. N. Rincy and R. Gupta, "Ensemble Learning Techniques and its Efficiency in Machine Learning: A Survey", *in 2nd International Conference on Data, Engineering and Applications (IDEA)*, IEEE, pp. 1–6, 2020, doi: 10.1109/IDEA49133.2020.9170675.

[19] J. Fan, J. Lee, and Y. Lee, "A Transfer Learning Architecture Based on a Support Vector Machine for Histopathology Image Classification", *Applied Sciences*, vol. 11, no. 14, pp. 6380, 2021, doi: 10.3390/app11146380.

[20] J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, and A. Lopez, "A comprehensive survey on support vector machine classification: Applications, challenges and trends", *Neurocomputing*, vol. 408, pp. 189–215, 2020, doi: 10.1016/j.neucom.2019.10.118.

[21] M. R. Akbar, I. Slamet, and S. S. Handajani, "Sentiment analysis using tweets data from Twitter of Indonesian's Capital City changes using classification method support vector machine", *in AIP Conference Proceedings*, American Institute of Physics Inc., pp. 020041, 2020, doi: 10.1063/5.0030357.

[22] Z. Qi, "The Text Classification of Theft Crime Based on TF-IDF and XGBoost Model", *in 2020 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, IEEE, pp. 1241–1246, 2020, doi: 10.1109/ICAICA50127.2020.9182555.

[23] C. M. Yeşilkanat, "Spatio-temporal estimation of the daily cases of COVID-19 in worldwide using random forest machine learning algorithm", *Chaos Solitons Fractals*, vol. 140, pp. 110210, 2020, doi: 10.1016/j.chaos.2020.110210.

[24] A. Kurani, P. Doshi, A. Vakharia, and M. Shah, "A Comprehensive Comparative Study of Artificial Neural Network (ANN) and Support Vector Machines (SVM) on Stock Forecasting", *Annals of Data Science*, vol. 10, no. 1, pp. 183–208, 2023, doi: 10.1007/s40745-021-00344-x.

[25] G. Siva, N. Rao, P. Madhuri, D. Sudheer, and D. Meghana, "Spam or Ham Text Classification using Logistic Regression", *Turkish Journal of Computer and Mathematics Education*, vol. 12, no. 9, pp. 426–433, 2021, doi: https://doi.org/10.17762/turcomat.v12i9.3097.

[26] T. M. Ma, K. YAMAMORI, and A. Thida, "A Comparative Approach to Naïve Bayes Classifier and Support Vector Machine for Email Spam Classification", *in 2020 IEEE 9th Global Conference on Consumer Electronics (GCCE)*, IEEE, pp. 324–326, 2020, doi: 10.1109/GCCE50665.2020.9291921.

[27] J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, and A. Lopez, "A comprehensive survey on support vector machine classification: Applications, challenges and trends", *Neurocomputing*, vol. 408, pp. 189–215, 2020, doi: 10.1016/j.neucom.2019.10.118.

[28] T. Jiang, J. L. Gradus, and A. J. Rosellini, "Supervised Machine Learning: A Brief Primer", *Behav Ther*, vol. 51, no. 5, pp. 675–687, 2020, doi: 10.1016/j.beth.2020.05.002.

[29] J. Wang, W. Xu, W. Yan, and C. Li, "Text Similarity Calculation Method Based on Hybrid Model of LDA and TF-IDF", *in Proceedings of the 2019 3rd International Conference on Computer Science and Artificial Intelligence*, New York, NY, USA: ACM, pp. 1–8, 2019, doi: 10.1145/3374587.3374590.

[30] A. Parmar, R. Katariya, and V. Patel, "A Review on Random Forest: An Ensemble Classifier", *in Lecture Notes on Data Engineering and Communications Technologies*, Springer Science and

Business Media Deutschland GmbH, vol. 26, pp. 758–763, 2019, doi: 10.1007/978-3-030-03146-6_86.

[31] E. H. Rachmawanto, D. R. I. M. Setiadi, N. Rijati, A. Susanto, I. U. W. Mulyono, and H. Rahmalan, “Attribute Selection Analysis for the Random Forest Classification in Unbalanced Diabetes Dataset,” *in 2021 International Seminar on Application for Technology of Information and Communication (iSemantic)*, pp. 82–86, 2021, doi: 10.1109/iSemantic52711.2021.9573181.

[32] A. Susanto, C. A. Sari, H. Rahmalan, and M. A. S. Doheir, “Support vector machine based discrete wavelet transform for magnetic resonance imaging brain tumor classification,” *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 21, no. 3, pp. 592–599, 2023, doi: 10.12928/TELKOMNIKA.v21i3.24928.

[33] N. R. D. Cahyo, C. A. Sari, E. H. Rachmawanto, C. Jatmoko, R. R. A. Al-Jawry, and M. A. Alkhafaji, “A Comparison of Multi Class Support Vector Machine vs Deep Convolutional Neural Network for Brain Tumor Classification,” *in 2023 International Seminar on Application for Technology of Information and Communication (iSemantic)*, IEEE, pp. 358–363, 2023, doi: 10.1109/iSemantic59612.2023.10295336.

[34] E. Y. Boateng, J. Otoo, and D. A. Abaye, "Basic Tenets of Classification Algorithms K-Nearest-Neighbor, Support Vector Machine, Random Forest and Neural Network: A Review", *Journal of Data Analysis and Information Processing*, vol. 08, no. 04, pp. 341–357, 2020, doi: 10.4236/jdaip.2020.84020.

[35] S. Muthukrishnan, H. Krishnaswamy, S. Thanikodi, D. Sundaresan, and V. Venkatraman, "Support vector machine for modelling and simulation of heat exchangers", *Thermal Science*, vol. 24, no. 1 Part B, pp. 499–503, 2020, doi: 10.2298/TSCI190419398M.

[36] Z. Soumaya, B. Drissi Taoufiq, N. Benayad, K. Yunus, and A. Abdelkrim, "The detection of Parkinson disease using the genetic algorithm and SVM classifier", *Applied Acoustics*, vol. 171, pp. 107528, 2021, doi: 10.1016/j.apacoust.2020.107528.

[37] Y. Li, J. Nie, and X. Chao, "Do we really need deep CNN for plant diseases identification?", *Comput Electron Agric*, vol. 178, pp. 105803, 2020, doi: 10.1016/j.compag.2020.105803.