2nd International Conference on Information Science and Technology Innovation (ICoSTEC)
February, 25 2023. Yogyakarta, Indonesia

Analysis of Machine Learning Algorithms in Predicting the Flood Status of Jakarta City

Irwan Daniel[1], Hartono[2], Zakarias Situmorang[3]

[1,2]*Magister of Computer Science, Potensi Utama University*
JL. KL. Yos Sudarso Km. 6,5 No. 3-A, Medan

[3]*Departement of Computer Science, Universitas Katolik Santo Thomas*
Jl. Setia Budi, Kp. Tengah, Kec. Medan Tuntungan, Medan

[1]irwandaniel@gmail.com
[2]hartonoibbi@gmail.com
[3]zakarias65@yahoo.com

***Abstract*— By mining the information in the dataset, we can solve a prediction problem, especially flood status prediction based on floodgate levels, using machine learning algorithms. This research employs three machine learning algorithms (K-Nearest Neighbor, Naive Bayes, and Support Vector Machine) for predicting the flood status using a dataset containing the data of DKI Jakarta's floodgate levels. Using a 5-fold, 10-fold, and 20-fold cross-validation evaluation, we get the highest accuracy (85.096%), f-score (85.1%), precision (85.641%), and recall (85.096%) from the model using the SVM algorithm with a polynomial kernel. Average performance-wise, the K-NN algorithm performs better than the other algorithm with an average accuracy of 83.147%, an average f-score of 83.156%, an average precision of 83.566%, and an average recall of 83.147%.**

***Keywords*— flood gate, flood prediction, k-nearest neighbors, naïve bayes, support vector machine**

I. INTRODUCTION

Floods are one of the most common disasters in Indonesia. This disaster often comes during the rainy season and causes disturbance to the community due to the damage and losses they cause [1]. By determining whether the water level exceeds or remains below the predetermined threshold, we can utilize river level and flow to determine if flooding will occur in a specific location [2]. Using flood incidents and water level data from DKI Jakarta as a source of knowledge, data mining is a suitable tool to categorize the flood status by classifying the data into specified categories [3].

Data mining analyzes and extracts knowledge from a dataset and uses that knowledge to solve problems such as association, clustering, prediction, estimation, and classification [4]. Some popular data mining algorithms used in classification problems are K-nearest neighbors (K-NN), support vector machine (SVM), and naive Bayes (NB) for their high processing speed, easy implementation, and good performance [5]. This research implements these algorithms in building models used to classify flood status based on a dataset of floodgate heights in DKI Jakarta province.

We use recent studies about water level classification, rainfall prediction, and flood prediction as references in this research. In the rainfall prediction using the SVM algorithm, the study shows that the best model produces a Root Mean Square Error (RMSE) value of 88.426 [6]. In flood prediction in Bangladesh using the K-NN algorithm, the study shows that the K-NN algorithm yields an average accuracy value of 94.91%, an average precision value of 92%, and an average recall value of 91% [7]. In the Sleman regency's rainfall intensity prediction using the NB algorithm, the study shows that the most influential parameter on rainfall intensity is the average temperature with an entropy value of 0.047811028 [8]. Using these references, we studied how machine learning solves flood prediction and classification problems based on the water level. We used three machine learning algorithms in the classification process in this study, namely the K-NN, SVM, and Naive Bayes algorithms.

In the K-NN algorithm, the K value (number of nearest neighbors) plays an essential role in the performance of the classification results; since it determines how much data should have similar characteristics [9]. As shown in the optimization of the K-NN algorithm using the certainty factor in determining students' careers, with 12 K values (K = 1 to K = 12), this study shows that the highest performance came from the K values of 3 and 4, while the worst from the K values of 12 [10]. Research about road damage identification using K values of 5, 8, and 15 shows that the K value of 5 yields the best performance, while the worst is from the K value of 15 [11]. Another research on student graduation classification using six K values (K = 1, K = 3, K = 5, K = 7, K = 9, and K = 11) shows that the K value of 7 generates the best model, while the K value of 1 the worst [12]. With these studies as references, this research implements two K values (K = 3 and K = 5) in the classification process and compares each performance.

The effectiveness of the K-NN algorithm's classification results is also influenced by selecting the proper distance metric, such as Euclidean and Manhattan since it will change how the clusters forms [13]. Studies in this discussion about textual data classification show that the Euclidean distance metric yields the best performance (accuracy value of 85.5%) compared to the Manhattan distance (accuracy value of (85.48%) [14]. Research about stroke disease detection shows that the Manhattan distance performs better in the

classification than the Euclidean distance, with an accuracy value of 96.03% against 95.93% [15]. Another implementation of the K-NN algorithm for students' academic performance classification shows that the Euclidean distance generates higher accuracy (98.42%) when compared to the Manhattan distance (97.76%) [16]. In this research, we use the Euclidean and Manhattan distance metrics in combination with the K values of 3 and 5 to show how each combination can affect the classification result.

The SVM algorithm utilizes a kernel function, such as the polynomial, radial basis function (RBF), and sigmoid, to solve the problem with non-linearly separable data by locating the optimum hyperplane into a high-dimension feature space [17]. In the classification of contraceptive use, the RBF kernel yields better performance with an Apparent Error Rate (APER) value of 73.83 compared to the polynomial (7.36) and sigmoid (55.57) [18]. These kernels comparison in a cloud environment's intrusion detection shows that the RBF kernel performs better accuracy (88.81%) when compared to the polynomial (52.58%) and sigmoid (88.2%) kernels [19]. The sigmoid kernel performs better than the other kernels in the brain tumor diagnosis using MRI images; this study shows that the sigmoid kernel generates an average accuracy of 87%, followed by the polynomial kernel (80%) and the RBF kernel (75%) [20]. This research utilizes the polynomial, RBF, and sigmoid kernel functions in flood status prediction using the SVM algorithm and displays the influence of hyperparameters on each kernel function.

In the end, we found the best model to predict the flood status of Jakarta City based on the level of Katulampa, Depok Post, Manggarai, Istiqlal, Jembatan Merah, Flushing Ancol, and Marina Ancol floodgates. To determine this best model, we used the accuracy, precision, and recall values of each model evaluated using 5-fold, 10-fold, and 20-fold cross-validation.

II. METHODS

A. Dataset

This study uses water level data covering seven floodgates in DKI Jakarta, such as Katulampa, Pos Depok, Manggarai, Istiqlal, Jembatan Merah, Flushing Ancol, and Marina Ancol. We obtained the data from kaggle.com [21], which contains a history of water levels in the seven regions from January 1 to December 7, 2020, with 624 data measured on a centimeter scale. Table I shows the 10 data set samples used.

TABLE I
DATASET SAMPLES

K	PD	M	I	JM	FA	MA	FS
A4	A4	A4	A4	A2	A4	A4	No Flood
A4	A4	A4	A4	A2	A4	A4	No Flood
A4	A4	A4	A4	A1	A4	A4	No Flood
A4	A4	A3	A4	A1	A4	A3	No Flood
A4	A4	A2	A3	A1	A4	A3	Flood
A4	A4	A2	A3	A1	A4	A3	Flood
A4	A3	A2	A3	A1	A4	A3	Flood
A4	A3	A2	A3	A1	A4	A3	Flood
A3	A3	A2	A3	A1	A4	A2	Flood
A3	A2	A2	A3	A1	A4	A2	Flood

Notes: K = Katulampa flood gate, PD = Pos Depok flood gate, M = Manggarai flood gate, I = Istiqlal flood gate, JM = Jembatan Merah flood gate, FA = Flushing Ancol flood gate, MA = Marina Ancol flood gate, FS = Flood status

Each flood gate in Table I above (K, PD, M, I, JM, FA, and MA) consists of four flood alert sequences, namely Alert 4 (A4), Alert 3 (A3), Alert 2 (A2), and Alert 1 (A1), where the smaller the alert value, the higher the risk of flooding. The normalized data from Table II shows the results of altering the A4 value to 4, A3 to 3, A2 to 2, and A1 to 1 from Table I.

TABLE II
NORMALIZATION RESULTS

K	PD	M	I	JM	FA	MA	FS
4	4	4	4	2	4	4	No Flood
4	4	4	4	2	4	4	No Flood
4	4	4	4	1	4	4	No Flood
3	4	3	3	1	4	3	No Flood
3	4	2	3	1	4	3	Flood
3	4	2	3	1	4	3	Flood
3	3	2	3	1	4	3	Flood
3	3	2	3	1	4	3	Flood
3	3	2	3	1	4	2	Flood
3	2	2	3	1	4	2	Flood

The normalization results shown in Table II will be the final dataset we use to predict the flood status with the K-NN, SVM, and NB algorithms.

B. K-Nearest Neighbors

In data mining, the K-Nearest Neighbor (K-NN) algorithm works without needing prior knowledge (unsupervised learning), where new data labels are generated based on their nearest neighbors (K value) and the majority voting process [22]. K-NN uses a distance metric to measure the distance between two points in the training and testing data in its classification process [23]. In the classification process, both distance metrics will calculate the distance between each data and classify them using equations (1) and (2) [24].

$$Euclidean\ Distance = \sqrt{\sum_{j=1}^{N} |x - y|^2} \quad (1)$$

$$Manhattan\ Distance = \sum_{j=1}^{N} |x - y| \quad (2)$$

In the classification of EGG signals using the K-NN algorithm, the results show that with different values of K=3, K =4, and K =5, the model using K=3 performance is better

than the model using K =5 [25]. The research result of the effect of distance metric on the K-NN algorithm performance shows that the Manhattan distance produces a model with better performance than the Euclidean one [26]. Looking at these results, we tried to combine the variation of the K value and the distance metric to show its effect on the classification results of the K-NN algorithm.

We use both distance metrics, the Euclidean and Manhattan distances, and two K values (K = 3 and K = 5) to build four prediction models, with a configuration shown in Table III.

TABLE III
K-NN MODEL CONFIGURATION

Model	Distance Metric	K Value
KNN I	Euclidean	3
KNN II	Euclidean	5
KNN III	Manhattan	3
KNN IV	Manhattan	5

The KNN I model uses the Euclidean distance as the distance metric and a K value of 3, while the KNN II uses the same distance metric but with a different K value (K = 5). The KNN III model uses a combination of Manhattan distance as the distance metric and a K value of 3, while the KNN IV uses the same distance metric but with a different K value (K = 5).

C. Naïve Bayes

Naive Bayes is a classic probabilistic-based data mining algorithm widely used to solve various classification problems [27]. This algorithm uses the probability value of class membership in classifying data, with a simplified Bayes theorem formula, as shown in equation (3) [28].

$$P(H|D) = \frac{P(D|H) * P(H)}{P(D)} \quad (3)$$

Where:

D = Data with unknown class
H = Hypothesis on D in specific classes
P(H|D) = Probability of H based on condition D (posterior probability)
P(D|H) = Probability of D based on condition Q (prior probability)
P(H) = Probability of H
P(D) = Probability of D

We use equation (3) to build a model that utilizes the NB algorithm in predicting flood status in this study.

D. Support Vector Machine

The support vector machine (SVM) is an algorithm that utilizes a hyperplane as a boundary to separate the data into positive or negative classes [29]. In solving classification problems using non-linearly separable datasets, this algorithm uses a kernel function (kernel trick) for mapping the data into a high-dimensional feature space to obtain a hyperplane that separates the data into two classes [30]. Some of the more popular kernel functions often used in SVM are the polynomial, RBF, and sigmoid functions; these kernel functions use equations (4) to (6) to generate a hyperplane in the classification process [31].

$$Polynomial = (g * x * y + c)^d \quad (4)$$

$$RBF = exp(-g|x - y|^2) \quad (5)$$

$$Sigmoid = tanh(g * x * y + c) \quad (6)$$

In this research, we use these kernel functions to build three classification models, with the configuration shown in Table IV.

TABLE IV
SVM MODEL CONFIGURATION

Model	Kernel Function	Parameters
SVM I	Polynomial	g = 0.1 c = 0.85 d = 3
SVM II	RBF	g = 0.1
SVM III	Sigmoid	g = 0.1 c = 0.46

The SVM I model uses the polynomial kernel and parameters such as g (gamma constant in the kernel function) = 0.1, c (c0 constant in the kernel function) = 0.1, and d (the degree of the kernel) = 0.3. The SVM II model uses the RBF kernel and parameter g = 0.1, while the SVM III uses the sigmoid kernel and parameters such as g = 0.1 and c = 0.46.

E. Cross-Validation Evaluation

With a total of 8 models built, each model predicts the flood status using the dataset (as shown in Figure 1), and we evaluate the results using 5-fold, 10-fold, and 20-fold cross-validation.

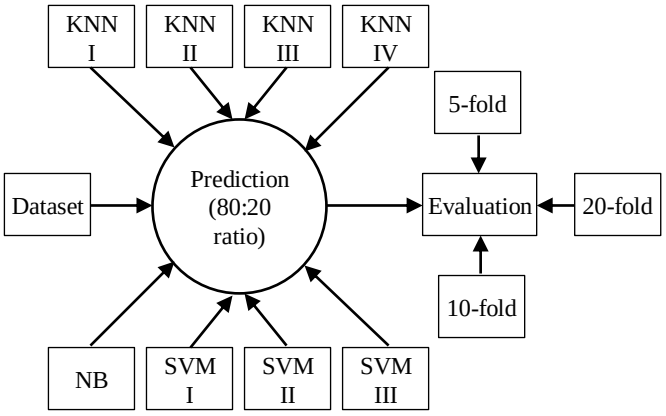

Fig. 1 Prediction Model

With a total of 8 models built, each model predicts the flood status using the dataset (as shown in Figure 1), and we evaluate the results using 5-fold, 10-fold, and 20-fold cross-validation. We use equations (7) to (10) to evaluate and analyze each model's prediction using the accuracy, f-score, precision, and recall values [32].

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (7)$$

$$Precision = \frac{TP}{TP+FP} \quad (8)$$

$$Recall = \frac{TP}{TP+FN} \quad (9)$$

$$F - Score = 2 * \frac{Precision*Recall}{Precision+Recall} \quad (10)$$

The comparative analysis research result on the classification of heart disease using the K-NN, Naive Bayes, and SVM algorithms shows that the SVM algorithm performs better than the other algorithms, with 92% accuracy; while the Naive Bayes performance is the worst with 88% [33]. In the performance comparison of predicting gold price movements, the K-NN algorithm performs better than other algorithms, with an accuracy of 61.9% [34]. The comparison of the Naïve Bayes, K-NN, and SVM algorithms in social media sentiment classification, shows that the Naive Bayes algorithm yields the highest performance with an accuracy of 79.8%, better than the K-NN (50.23%), and SVM (75.29%) algorithms [35]. The difference in the comparison results of the three machine learning algorithms is the basis for analyzing the model's performance built in this study to evaluate which algorithm succeeded in producing the best model for predicting the flood status of Jakarta City based on the level of floodgates using the cross-validation evaluation method.

III. RESULTS AND DISCUSSIONS

With an 80:20 ratio between training and testing data, we get the prediction results from each model in the form of a confusion matrix. Table V to Table VIII summarizes the confusion matrix from the K-NN, NB, and SVM evaluation.

TABLE V
K-NN I & K-NN II CONFUSION MATRIX EVALUATION

Model	K-Fold	Actual	Predicted	
			Flood	No Flood
KNN I	5	Flood	262	69
		No Flood	37	256
	10	Flood	262	69
		No Flood	42	251
	20	Flood	261	70
		No Flood	37	256
KNN II	5	Flood	262	69
		No Flood	39	254
	10	Flood	264	67
		No Flood	35	258
	20	Flood	267	64
		No Flood	34	259

TABLE VI
K-NN III & K-NN IV CONFUSION MATRIX EVALUATION

Model	K-Fold	Actual	Predicted	
			Flood	Flood
KNN III	5	Flood	262	69
		No Flood	37	256
	10	Flood	262	69
		No Flood	42	251
	20	Flood	261	70
		No Flood	37	256
KNN IV	5	Flood	263	68
		No Flood	39	254
	10	Flood	265	66
		No Flood	35	258
	20	Flood	267	64
		No Flood	34	259

TABLE VII
NAÏVE BAYES CONFUSION MATRIX EVALUATION

Model	K-Fold	Actual	Predicted	
			Flood	Flood
NB	5	Flood	269	62
		No Flood	50	243
	10	Flood	268	63
		No Flood	50	243
	20	Flood	268	63
		No Flood	47	246

TABLE VIII
SVM CONFUSION MATRIX EVALUATION

Model	K-Fold	Actual	Predicted	
			Flood	Flood
SVM I	5	Flood	267	64
		No Flood	29	264
	10	Flood	265	66
		No Flood	28	265
	20	Flood	266	65
		No Flood	31	262
SVM II	5	Flood	269	62
		No Flood	42	251
	10	Flood	270	61
		No Flood	41	252
	20	Flood	268	63
		No Flood	41	252
SVM III	5	Flood	239	92
		No Flood	39	254
	10	Flood	245	86
		No Flood	42	251
	20	Flood	245	86
		No Flood	38	255

From the values in Table V to Table VIII above, we calculate each model's accuracy, f-score, precision, and recall, resulting in the summary shown in Table IX.

TABLE IX
PERFORMANCE EVALUATION

Model	Accuracy	F-Score	Precision	Recall
KNN I (5-Fold)	83.013	83.021	83.467	83.013
KNN I (10-Fold)	82.212	82.225	82.547	82.212
KNN I (20-Fold)	82.853	82.860	83.332	82.853
KNN II (5-Fold)	82.692	82.703	83.096	82.692
KNN II (10-Fold)	83.654	83.662	84.111	83.654
KNN II (20-Fold)	84.295	84.305	84.704	84.295
KNN III (5-Fold)	83.013	83.021	83.467	83.013
KNN III (10-Fold)	82.212	82.225	82.547	82.212
KNN III (20-Fold)	82.853	82.860	83.332	82.853
KNN IV (5-Fold)	82.853	82.864	83.234	82.853
KNN IV (10-Fold)	83.814	83.823	84.246	83.814
KNN IV (20-Fold)	84.295	84.305	84.704	84.295
NB (5-Fold)	82.051	82.066	82.141	82.051
NB (10-Fold)	81.891	81.906	81.992	81.891
NB (20-Fold)	82.372	82.388	82.512	82.372
SVM I (5-Fold)	85.096	85.100	85.641	85.096
SVM I (10-Fold)	84.936	84.936	85.568	84.936
SVM I (20-Fold)	84.615	84.621	85.130	84.615
SVM II (5-Fold)	83.333	83.349	83.535	83.333
SVM II (10-Fold)	83.654	83.669	83.856	83.654
SVM II (20-Fold)	83.333	83.348	83.571	83.333
SVM III (5-Fold)	79.006	78.964	80.073	79.006
SVM III (10-Fold)	79.487	79.473	80.255	79.487
SVM III (20-Fold)	80.128	80.104	81.035	80.128

Next, we calculate the average values of accuracy, f-score, precision, and recall from each algorithm, as shown in Table X, to compare which algorithm has the best average performance.

TABLE X
AVERAGE PERFORMANCE EVALUATION

Algorithm	Average Accuracy	Average F-Score	Average Precision	Average Recall
K-NN	83.147	83.156	83.566	83.147
NB	82.105	82.120	82.215	82.105
SVM	82.621	82.618	83.185	82.621

From the results shown in Tables IX, we analyze each model's accuracy, f-score, precision, and recall to find the best and worst model. The SVM I model, with the polynomial kernel and 5-fold cross-validation evaluation, shows the highest accuracy (85.096%), f-score (85.1%), precision (85.641%), and recall (85.096%) values. The SVM III model, with the sigmoid kernel and 5-fold cross-validation, shows the lowest accuracy (79.006%), f-score (78.964%), precision (80.073%), and recall (79.006%) values.

Models using the K-NN algorithm with Euclidean and Manhattan distance (both with the K value of 5) produced the best performance in 20-fold cross-validation evaluation, producing an accuracy value of 84.295%, f-score value of 84.305%, a precision value of 84.704%, and a recall value of 84.295%. The models using the K-NN algorithm with Euclidean and Manhattan distance (both with the K value of 3) produced the worst performance in 10-fold cross-validation evaluation, producing an accuracy value of 82.212%, f-score value of 82.225%, a precision value of 82.547%, and a recall value of 82.212%.

We summarize the analytical results of this flood status prediction research based on floodgate levels in Table XI, based on the performance of the resulting models.

TABLE XI
EVALUATION SUMMARY

Parameter	Evaluation	
	Result	Description
Best performance	SVM I	Algorithm: SVM Kernel: Polynomial Cross-validation: 5-fold
Worst performance	SVM III	Algorithm: SVM Kernel: Sigmoid Cross-validation: 5-fold
Best K-NN	K-NN II	Distance metric: Euclidean K value = 5 Cross-validation: 20-fold
	K-NN IV	Distance metric: Manhattan K value = 5 Cross-validation: 20-fold
Worst K-NN	K-NN I	Distance metric: Euclidean K value = 3 Cross-validation: 10-fold
	K-NN III	Distance metric: Manhattan K value = 3 Cross-validation: 10-fold
Best NB	NB	Cross-validation: 20-fold
Worst NB	NB	Cross-validation: 10-fold
Best SVM	SVM I	Kernel: Polynomial Cross-validation: 5-fold
Worst SVM	SVM III	Kernel: Sigmoid Cross-validation: 5-fold

IV. CONCLUSIONS

From the research result, we conclude that the best algorithm to predict the flood status using the floodgate dataset is the SVM algorithm with a polynomial kernel and a 5-fold cross-validation evaluation. We also concluded that the poorest performance algorithm in predicting flood status is the SVM algorithm with a sigmoid kernel and 5-fold cross-validation. The best K-NN algorithm configuration to use to predict the flood status using this database is the Euclidean and Manhattan distance metrics, with a K value of 5. Six out of eight models perform better when being evaluated using the 20-fold cross-validation. We conclude that this is the best K value for the cross-validation evaluation. From the average accuracy, f-score, precision, and recall value of each algorithm used in this research, we found that, on average, the K-NN algorithm performs better than the NB and SVM algorithm.

REFERENCES

[1] N. G. Saputra, M. Rifai, and P. Marsingga, "Flood Disaster Management Strategy of Karawang Regency in Karangligar Village as a Disaster Resilient Village," *J. Anal. Kebijak. dan Pelayanan Publik*, vol. 8, no. 1, pp. 62–76, 2021.

[2] H. Ihsudha *et al.*, "Measurement of River Water Level and Flow as Flash Flood Detection," in *PROSIDING SKF 2018*, 2018, pp. 23–29.

[3] M. A. Hasanah, S. Soim, and A. S. Handayani, "Implementation of CRISP-DM Model Using Decision Tree Method with CART Algorithm for Flood Potential Rainfall Prediction," *J. Appl. Informatics Comput.*, vol. 5, no. 2, pp. 103–108, Oct. 2021.

[4] I. Firmansyah, J. T. Samudra, D. Pardede, and Z. Situmorang, "Comparison Of Random Forest And Logistic Regression In The Classification Of Covid-19 Sufferers Based On Symptoms," *J. Sci. Soc. Res.*, vol. 5, no. 3, p. 595, Oct. 2022.

[5] A. C. Khotimah and E. Utami, "Comparison Naïve Bayes Classifier, K-Nearest Neighbor And Support Vector Machine In The Classification Of Individual On Twitter Account," *J. Tek. Inform.*, vol. 3, no. 3, pp. 673–680, 2022.

[6] F. Yulianto, W. F. Mahmudy, and A. A. Soebroto, "Comparison of Regression, Support Vector Regression (SVR), and SVR-Particle Swarm Optimization (PSO) for Rainfall Forecasting," *J. Inf. Technol. Comput. Sci.*, vol. 5, no. 3, pp. 235–247, 2020.

[7] N. Gauhar, S. Das, and K. S. Moury, "Prediction of Flood in Bangladesh using k-Nearest Neighbors Algorithm," in *2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST)*, Jan. 2021, no. January, pp. 357–361.

[8] I. G. Sena, J. WDillak, P. Leunupun, and A. J. Santoso, "Predicting Rainfall Intensity using Naïve Bayes and Information Gain Methods (Case Study: Sleman Regency)," *J. Phys. Conf. Ser.*, no. 2nd 2019 ICERA, p. 012011, Mar. 2019.

[9] L. Al-Shalabi, "Comparative study of data mining classification techniques for detection and prediction of phishing websites," *J. Comput. Sci.*, vol. 15, no. 3, pp. 384–394, 2019.

[10] Nunsina, Tulus, and Z. Situmorang, "Analysis Optimization K-Nearest Neighbor Algorithm with Certainty Factor in Determining Student Career," in *2020 3rd International Conference on Mechanical, Electronics, Computer, and Industrial Technology (MECnIT)*, Jun. 2020, pp. 306–310.

[11] I. A. Angreni, S. A. Adisasmita, M. I. Ramli, and S. Hamid, "The Effect of K Value on the K-Nearest Neighbor (Knn) Method on the Accuracy Level of Road Damage Identification," *Rekayasa Sipil*, vol. 7, no. 2, p. 63, Jan. 2019.

[12] N. Hidayati and A. Hermawan, "K-Nearest Neighbor (K-NN) algorithm with Euclidean and Manhattan in classification of student graduation," *J. Eng. Appl. Technol.*, vol. 2, no. 2, pp. 86–91, 2021.

[13] D. M. Saputra, D. Saputra, and L. D. Oswari, "Effect of Distance Metrics in Determining K-Value in K-Means Clustering Using Elbow and Silhouette Method," in *Proceedings of the Sriwijaya International Conference on Information Technology and Its Applications (SICONIAN 2019)*, 2020, vol. 172, no. Siconian 2019, pp. 341–346.

[14] W. Wahyono, I. N. P. Trisna, S. L. Sariwening, M. Fajar, and D. Wijayanto, "Comparison Of K-Nearest Neighbor Distance Calculation In Textual Data Classification," *J. Teknol. dan Sist. Komput.*, vol. 8, no. 1, pp. 54–58, 2020.

[15] I. Iswanto, T. Tulus, and P. Sihombing, "Comparison of Distance Models on K-Nearest Neighbor Algorithm in Stroke Disease Detection," *Appl. Technol. Comput. Sci. J.*, vol. 4, no. 1, pp. 63–68, 2021.

[16] S. Sugriyono and M. U. Siregar, "Preprocessing kNN algorithm classification using K-means and distance matrix with students' academic performance dataset," *J. Teknol. dan Sist. Komput.*, vol. 8, no. 4, 2020.

[17] I. S. Al-Mejibli, J. K. Alwan, and D. H. Abd, "The effect of gamma value on support vector machine performance with different kernels," *Int. J. Electr. Comput. Eng.*, vol. 10, no. 5, pp. 5497–5506, 2020.

[18] M. B. Johra, "Application of Non-Linear Support Vector Machine on Contraceptive Use in North Maluku Province," *J. Mat. "MANTIK,"* vol. 4, no. 2, pp. 137–142, 2018.

[19] N. Nirmalajyothi, K. G. Rao, B. B. Rao, and K. Swathi, "Performance of Various SVM Kernels for Intrusion Detection of Cloud Environment," *Int. J. Emerg. Trends Eng. Res.*, vol. 8, no. 10, pp. 7532–7539, 2020.

[20] W. A. K. Naser, E. A. Kadim, and S. H. Abbas, "SVM Kernels comparison for brain tumor diagnosis using MRI," *Glob. J. Eng. Technol. Adv.*, vol. 7, no. 2, pp. 026–036, 2021.

[22] R. M. F. Lubis, Z. Situmorang, and R. Rosnelly, "Analisis Variation K-Fold Cross Validation On Classification Data Method K-Nearest Neighbor," *J. Ipteks Terap.*, vol. 15, no. March, pp. 68–73, 2020.

[23] D. Noviana, Y. Susanti, and I. Susanto, "Analysis Of Scholarship Recipient Recommendations Using The K-Nearest Neighbor (K-Nn) Algorithm And C4.5 Algorithm," 2019.

[24] K. F. Margolang, M. M. Siregar, S. Riyadi, and Z. Situmorang, "Distance Metric Analysis of K-Nearest Neighbor Algorithm on Bad Debt Classification," *J. Inf. Syst. Res.*, vol. 3, no. 2, pp. 118–124, Feb. 2022.

[25] B. Kaliraman and M. Duhan, "Feature Extraction and Classification of EEG Signals Using Machine Learning Algorithms for Biometric Systems," *Int. J. Comput. Inf. Syst. Ind. Manag. Appl.*, vol. 12, pp. 339–348, 2020.

[26] H. A. Abu Alfeilat *et al.*, "Effects of Distance Measure Choice on K-Nearest Neighbor Classifier Performance: A Review," *Big Data*, vol. 7, no. 4, pp. 221–248, Dec. 2019.

[27] E. S. Negara and D. Triadi, "Topic modeling using latent dirichlet allocation (LDA) on twitter data with Indonesia keyword," *Bull. Soc. Informatics Theory Appl.*, vol. 5, no. 2, pp. 124–132, 2021.

[28] K. L. Kohsasih and Z. Situmorang, "Comparative Analysis of C4.5 and Naïve Bayes Algorithms in Predicting Cerebrovascular Disease," *J. Inform.*, vol. 9, no. 1, pp. 13–17, 2022.

[29] Hartono, O. S. Sitompul, Tulus, and E. B. Nababan, "Biased support vector machine and weighted-SMOTE in handling class imbalance problem," *Int. J. Adv. Intell. Informatics*, vol. 4, no. 1, pp. 21–27, 2018.

[30] J. Kusuma, B. H. Hayadi, and R. Rosnelly, "Comparison of Multi-Layer Perceptron (MLP) and Support Vector Machine (SVM) Methods for Breast Cancer Classification," *MIND (Multimedia Artif. Intell. Netw. Database) J.*, vol. 7, no. 1, pp. 51–60, 2022.

[31] P. K. Intan, "Comparison of Kernel Function on Support Vector Machine in Classification of Childbirth," *J. Mat. "MANTIK,"* vol. 5, no. 2, pp. 90–99, 2019.

[32] D. Pardede, I. Firmansyah, M. Handayani, M. Riandini, and R. Rosnelly, "Comparison Of Multilayer Perceptron's Activation And Optimization Functions In Classification Of Covid-19 Patients," *JURTEKSI (Jurnal Teknol. dan Sist. Informasi)*, vol. 8, no. 3, pp. 271–278, Aug. 2022.

[33] S. Likmi, "Comparative Analysis of Naive Bayes , K-Nearest Neighbors (KNN), and Support Vector Machine (SVM) Algorithms for Classification of Heart Disease Patients," *JOIN (Jurnal Online Inform.*, vol. 7, no. 2, pp. 219–225, 2022.

[34] Y. Suryana and T. W. Sen, "The Prediction of Gold Price Movement by Comparing Naive Bayes, Support Vector Machine, and K-NN," *JISA(Jurnal Inform. dan Sains)*, vol. 4, no. 2, pp. 112–120, 2021.

[35] M. H. Asnawi, I. Firmansyah, R. Novian, and R. S. Pontoh, "Comparison of Naïve Bayes, K-NN, and SVM Algorithms in Social Media Sentiment Classification," *Semin. Nas. Stat. X*, vol. 10, no. 1, 2021.