Performance Comparison of the Kernels of Support Vector Machine Algorithm for Diabetes Mellitus Classification

Dimas Aryo Anggoro, Dian Permatasari
Informatics Engineering Department, Universitas Muhammadiyah Surakarta, Surakarta, Indonesia

***Abstract*—Diabetes Mellitus is a disease where the body cannot use insulin properly, so this disease is one of the health problems in various countries. Diabetes Mellitus can be fatal and can cause other diseases and even lead to death. Based on this, it is important to have prediction activities to find out a disease. The SVM algorithm is used in classifying Diabetes Mellitus diseases. The purpose of this study was to compare the accuracy, precision, recall, and F1-Score values of the SVM algorithm with various kernels and data preprocessing. The data preprocessing used included data splitting, data normalization, and data oversampling. This research has the benefit of solving health problems based on the percentage of Diabetes Mellitus and can be used as material for accurate information. The results of this study are that the highest accuracy was obtained by 80% obtained from the polynomial kernel, the highest precision was obtained by 65% which was also obtained from the polynomial kernel, and the highest recall was obtained by 79% obtained from the RBF kernel and the highest f1-score was obtained by 70% obtained from RBF kernel.**

***Keywords*—Diabetes mellitus; kernel; normalization; oversampling; SVM**

I. Introduction

Diabetes Mellitus is a disease where blood sugar levels are overly high because the body cannot use insulin properly. Currently, Diabetes Mellitus becomes a serious health problem in various countries, including Indonesia [1]. The International Diabetes Federation (IDF) explained that in 2021 the number of people with Diabetes Mellitus in Indonesia reached 19.5 million people, while in 2019 the figure was 10.7 million. This means that there has been an increase of nearly 9 million cases in just 2 years, or just during the COVID-19 pandemic. With almost 2 times the addition, makes Indonesia ranked fifth in the world. Not only in Indonesia, but this upward trend in cases also occurs in the world. According to IDF data, currently, at least 1 in 10 people or as many as 537 million people in the world live with Diabetes Mellitus. If not treated properly immediately, Diabetes Mellitus can be fatal and can cause other diseases and even lead to death. Based on this, it is important to have prediction activities to find out a disease. This activity is carried out so that a disease can be detected quickly and can be treated immediately.

Activities in predicting various diseases have been carried out in various scientific fields, one of which is the field of computer science. Along with the development of information and communication technology, it can be used to improve the ability of the system to help detect Diabetes Mellitus disease[2]. Data mining is part of the Knowledge Discovery in Database (KDD) process that can classify, predict, and get a lot of information from large data sets[3]. Classification is an important stage in data mining; classification is carried out by looking at variables from existing data groups and aims to predict the class of an object that was not previously known [4].

II. Literature Review

Previous research conducted by Andi Maulida Argina regarding the application of the K-Nearest Neighbour classification model to the diabetes patient dataset explained that the study had the highest accuracy of 39%[5]. Another study conducted by Noviandi on the implementation of the Decision Tree C4.5 algorithm for diabetes prediction resulted in a prediction model that had the highest accuracy of 70.32% [6]. The shortcoming of the previous study is that the accuracy of the prediction model is still below 80%, so there is a need to improve accuracy performance. In research [7] that compared accuracy, recall, and precision classification on the C4.5 algorithm, Random Forest, Support Vector Machine (SVM), and Naïve Bayes resulted in the C4.5 algorithm obtaining accuracy of 86.67%, the Random Forest algorithm obtained accuracy of 83.33%, the SVM algorithm obtained accuracy by 95%, and the Naive Bayes algorithm obtained an accuracy of 86.67%. The highest accuracy algorithm is the SVM algorithm, therefore in this study applying the SVM algorithm for the classification of Diabetes Mellitus disease. This research is expected to provide accuracy results reaching 80%, so that it can improve deficiencies in previous studies.

The SVM algorithm was chosen because it is reliable in processing large amounts of data by optimizing hyperplanes in high-dimensional space that maximizes margins between data [8]. The use of the kernel in SVM is carried out to determine kernel parameters and produce the best accuracy in the classification process. Linear kernels are used when classified data can be easily separated by a hyperplane, while non-linear kernels are used when the data used is separated using curved lines or a plane in space that has high dimensions [9].

This study aims to compare the accuracy, precision, recall, and F1-Score values of the SVM algorithm with various kernels and preprocessing data in the classification of Diabetes Mellitus disease. It has the benefit of solving health problems based on the percentage of Diabetes Mellitus and can be an accurate information material. The output of this study is to

imply that the SVM algorithm is expected to show better performance values than previous studies.

III. METHODOLOGY

A. Data Collection

The first stage in this study is the collection of Diabetes Mellitus datasets. The dataset used is the Pima Indian diabetes dataset obtained from the UCI Machine Learning Repository. Several variables and attributes can facilitate the research process in data mining. The Pima Indian diabetes dataset consists of 768 data and 9 attributes. The variables and attributes used are shown in Table I.

TABLE I. VARIABLES AND ATTRIBUTES OF PEOPLE WITH DIABETES MELLITUS

Variable	Attribute
X1	*Pregnancies*, the number of pregnancies during life in the *range of* 0-17 times.
X2	*Glucose*, glucose/blood sugar levels. Normal blood sugar levels are below 120 mg/dL, while the sugar levels of diabetics are more than 120 mg/dL. *The* data range in the *dataset* is 0-199 mg/dL.
X3	*Blood Pressure*, blood pressure with mmHg units, the data *range* in *the dataset* is 0-112 mmHg.
X4	*Skin Thickness*, skin fold thickness with *a data range of* 0-99 mm. The norm is about 12.5 mm.
X5	Insulin, insulin levels in the blood with *a data range of* 0-846 U / ml.
X6	BMI, body mass weight with a data *range of* 0-67.1 BMI
X7	*Diabetes Pedigree Function*, History of diabetes Mellitus disease in the family with a data range of 1.001-2.42.
X8	*Age*, age of the patient (years) with *a data range of* 21-81 years.
Y	*Outcome*, negative and positive class variables (0 and 1). 0 are indicators of non-diabetics while 1 is an indicator of diagnosed diabetics.

B. Data Preprocessing

1) Data splitting: The next stage is the data splitting stage which is carried out by separating training data and testing data. Training data is used to create models that are applied to testing data[10] and testing data cannot be used for the training process, so that the model really learns from the new data[11]. The determination of training data and testing data is carried out randomly, so that the proportion between categories remains balanced [12]. In this study, splitting data was divided into 80% training data and 20% testing data.

2) Normalization data: Normalization of data in datasets aims to create data in the same range of values [13]. This study used the min-max and z-score normalization methods.

a) Min-max normalization: Normalization of min-max can overcome non-uniform data forms with a range of values greater than 0-1 [14]. Min-max normalization was chosen because it has the advantage that the data is balanced between before and after normalization [15]. The normalization of min-max is presented in (1).

$$x_{new} = \frac{x_{old} - x_{min}}{x_{max} - x_{min}} \quad (1)$$

x_{new} represents the min-max value, x_{old} is the value to be normalized, x_{min} is the lowest value of the overall data and x_{max} is the highest value of the entire data.

b) Z-Score normalization: Z-Score normalization is used to compare the performance or quality of data goals with the average distribution of data across groups based on standard deviation values [16]. Z-score normalization was chosen because it is a good normalization method for balancing data scale [17]. (2) is a formula for knowing the z-score.

$$x_{new} = \frac{x_{old} - \mu}{\sigma} \quad (2)$$

x_{new} is the z-score value, x_{old} is the value to be normalized, μ is the average value of the whole data and σ is the standard deviation value.

3) Oversampling

a) SMOTE (Synthetic Minority Over-sampling Technique): The SMOTE method can handle dataset class imbalances by working to make data replication of minor classes to be equivalent to major classes [18]. The diabetes dataset used in this study had a total of 268 positive classes and 500 negative classes so that there was an imbalance between the positive class and the negative class. Therefore, the SMOTE method was used in this study to balance between positive classes and negative classes. (3) is the formula for SMOTE.

$$x_{syn} = x_i + (x_{knn} - x_i)\gamma \quad (3)$$

x_{syn} is the resulting new class data, x_i is the approach to i, x_{knn} is the x closest to x_i and γ is a random number between 0-1.

C. Data Processing

1) Support Vector Machine (SVM): SVM is a good algorithm for data classification [19] with the principle of finding the best hyperplane that serves as a separator of two data classes [20]. The best hyperplane is determined by measuring the hyperplane margin and finding its maximum point, margin is the distance between the hyperplane and the nearest point of each class and this closest point is called the support vector [21]. The following is a description of SVM, there is data $\overrightarrow{xi} \in (\overrightarrow{x1}, \overrightarrow{x2}\ \overrightarrow{x3} \ldots xn)$ xi is data consisting of n attributes and two classes $y_i \in +1, -1$. Suppose that the two classes can be perfectly separated by a d-dimensional hyperplane defined by (4).

$$\vec{w}.\overrightarrow{xi} + b = 0 \quad (4)$$

Data $\overrightarrow{xi}$ which belonging to the positive class (+1) are shown in (5).

$$\vec{w}.\overrightarrow{xi} + b \geq -1 \quad (5)$$

Meanwhile, data $\overrightarrow{xi}$ belonging to the negative class (-1) are shown in (6).

$$\vec{w}.\overrightarrow{xi} + b \leq +1 \quad (6)$$

The maximum margin can be obtained by maximizing the value of the distance between the hyperplane and its closest point or support vector which $\frac{1}{||\vec{w}||}$[22]. It is formulated as Quadratic Programming (QP) by looking for a minimum point based on (7).

$$min\ \tau(w) = \frac{1}{2}||\vec{w}||^2 \quad (7)$$

By paying attention to the constraints on (8).

$$y_i(\vec{xi}, \vec{w} + b) - 1 \geq 0 \quad (8)$$

y_i is the target class to i, $\vec{xi}$ is the input data to i, $\vec{w}$ is the weight, and b is the relative field position.

2) Kernel SVM: To work around high-dimensional data can use a kernel that transforms the input space into a feature space[23]. Kernel functions commonly used in SVM are Linear[24], Radial Basic Function (RBF) and Polynomial [25]. The parameters possessed by kernel functions are used in the testing process[26]. There is no definite conclusion about the best kernel, therefore this study will compare 4 kernel functions, namely linear, RBF, polynomial and sigmoid.

a) Kernel linear: The Linear kernel was chosen because it is the simplest kernel and is used when the data is linearly overstretched.

$$K(x.y) = x.y \quad (9)$$

b) Kernel polynomial: The Polynomial kernel was chosen because it can be used when the data is not linearly separated and is suitable for use in solving classification problems in all training data that has been normalized.

$$K(x.y) = (\gamma(x.y) + C)^d \quad (10)$$

3) Kernel Radial Basic Function (RBF): The RBF kernel is used when the data is not linearly separated, it is chosen because it performs well with certain parameters, and the result of the training has a small error value.

$$K(x.y) = exp(-\gamma|x - y|^2) \quad (11)$$

a) Kernel sigmoid: This sigmoid kernel was chosen because it is similar to the two-layer perceptron model of the neural network, which works as an activation function for neurons.

$$K(x.y) = tanh(\gamma(x.y) + C) \quad (12)$$

4) Evaluation: Confusion matrix is an evaluation method that provides information comparing the classification of prediction results with the actual classification [27]. There are 4 terms of value from the confusion matrix, namely True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). Based on these values, accuracy, precision, recall, and F1-Score values can be generated. Accuracy is the ratio of predicted correct values of all data [28].

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \quad (13)$$

Precision indicates a correctly classified prediction of positive values divided across positive classified data [28].

$$Precision = \frac{TP}{TP+FP} \quad (14)$$

Recall shows the comparison of the positive correct predicted value with the entire positive correct value [29].

$$Recall = \frac{TP}{TP+FN} \quad (15)$$

The F1-Score shows the average comparison of precision and recall values[29].

$$F1 - Score = \frac{2\times Precision\times Recall}{Precision+Recall} \quad (16)$$

IV. Results and Discussions

This stage is a decipherment of the research obtained and its explanation.

A. Data Preprocessing

The dataset used is the Pima Indian diabetes dataset which consists of 768 data and 9 attributes. The initial stage carried out in this study is the process of collecting and processing datasets. In this study, data preprocessing was divided into three steps. The first step is the data splitting process, where the Diabetes Mellitus dataset will be divided into training data and testing data. The second step is the data normalization process to create data in the same range. The third step is an oversampling process to balance the dataset class by using the SMOTE method. Data processing in the study uses the Python programming language in the Google colab application.

1) Data splitting results: After getting the dataset, the next step is to divide the dataset into training data and testing data. The Diabetes Mellitus dataset totaled 768 data consisting of 8 variables and one target/class. Then the dataset is divided into 80% training data, totaling 614 data and 20% testing data, totaling 154 data. The diagnosis of Diabetes Mellitus is divided into two, namely non-diabetics who are denoted by 0 and diabetics who are denoted by 1. Obtained diabetics totaled 268 data and non-diabetics amounted to 500 data.

2) Data normalization results: The normalization methods used are min-max and z-score.

Fig. 1 shows the comparison of variables in the dataset, variables compared to 2, namely, pregnancies and insulin, the data has a fairly high range of values. For example, in the insulin variable, where the range of values is between 0 to above 200, this is considered unbalanced. The min-max normalization method is used to process values into the range of 0-1. Fig. 2 shows the results after normalization of min-max, where the range of values in the insulin variable becomes smaller, namely, 0-1.

In addition to using the min-max method, data normalization is also carried out using the z-score method. Z-score is performed by processing the mean and standard deviation from the values of its attributes. Fig. 3 shows the results after normalizing the z-score.

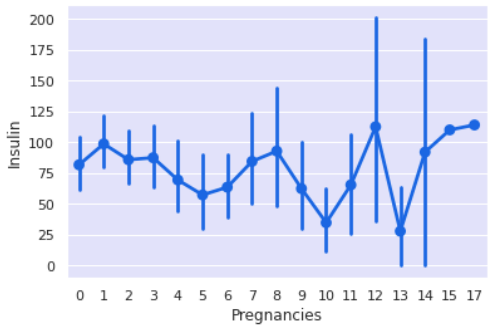


Fig. 1. Before normalization.

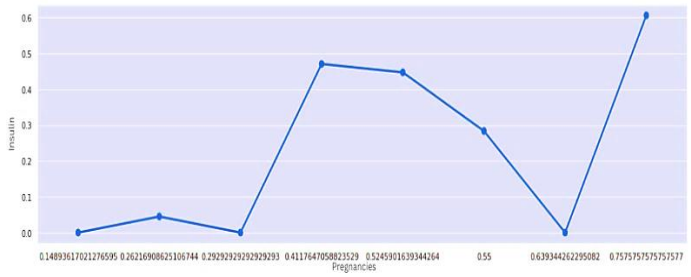


Fig. 2. After min-max normalization.

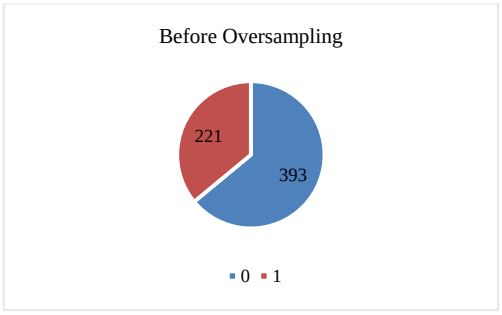
Fig. 3. After z-score normalization.

3) Oversampling results: In the dataset there is a difference between the number of positive classes and negative classes, therefore there is a need for class balancing. Class balancing is done by oversampling using the SMOTE method and is carried out on training data only. Oversampling is carried out after splitting data so that data replication does not appear in data training and data testing [30]. It can be seen in Fig. 4, before oversampling the number of positive classes was 221 and the number of negative classes was 393. Meanwhile, after oversampling, the number between the positive class and the negative class becomes the same, which is 393 so that it becomes balanced.

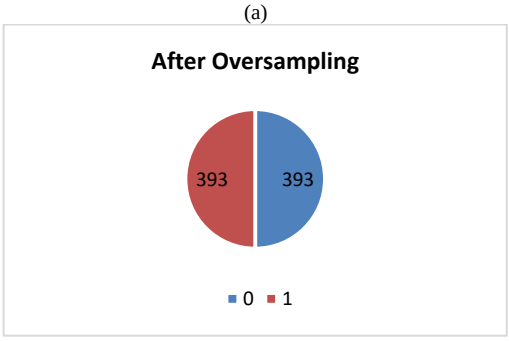
B. Data Preprocessing and Evaluation

This study compared the performance of the SVM algorithm kernels for the classification of Diabetes Mellitus diseases. SVM kernels used include linear kernels, polynomial kernels, RBF kernels, and sigmoid kernels. Evaluation is carried out using the confusion matrix method to calculate the values of accuracy, precision, recall, and f1-score by optimizing the best parameters for each kernel. Each kernel on SVM has a specific parameter, the cost parameter (C) being the most commonly used value for all kernels. The gamma (γ) parameter is used to determine the degree of proximity between 2 points to make it easier to find γ hyperplanes that are consistent with the data. The gamma parameter is used by polynomial, RBF, and sigmoid kernels. Next is the degree (d) parameter used to map data from the input space to the higher dimension space in the feature space, only the polynomial kernel uses this parameter [31]. Determination of the best parameters on the kernel is carried out by trial and error. Table II is the result of evaluating the classification models of various SVM kernels before various data preprocessing is carried out.

For this experiment in Table II, all parameter values in each kernel use auto parameters from python. The highest accuracy is obtained from the polynomial and RBF kernels, which is 77%. The highest precision was obtained from the RBF kernel, which was 69%, the highest recall was obtained from the linear kernel, which was 57% and the highest f1-score was obtained from linear and polynomial kernels, which was 61%. Table III is the result of evaluating the classification models of various SVM kernels after preprocessing data with min-max normalization and SMOTE oversampling. Meanwhile, Table IV is the result of evaluating the classification models of various SVM kernels after preprocessing data with normalization of z-score and oversampling SMOTE.



(a)



(b)

Fig. 4. (a) Data before oversampling, (b) Data after oversampling.

TABLE II. RESULTS OF EVALUATION OF VARIOUS SVM KERNELS BEFORE DATA PREPROCESSING

	Kernel			
	Linear	***Polynomial***	***RBF***	***Sigmoid***
Accuracy	76%	77%	77%	51%
Precision	66%	68%	69%	12%
Recall	57%	55%	53%	8%
F1-Score	61%	61%	60%	10%

TABLE III. RESULTS OF EVALUATION WITH MIN-MAX AND SMOTE

	Kernel			
	Linear	***Polynomial***	***RBF***	***Sigmoid***
Accuracy	77%	79%	79%	76%
Precision	61%	64%	63%	59%
Recall	72%	72%	79%	68%
F1-Score	66%	68%	70%	63%

TABLE IV. RESULTS OF VALUATION WITH Z-SCORE AND SMOTE

	Kernel			
	Linear	***Polynomial***	***RBF***	***Sigmoid***
Accuracy	79%	80%	77%	78%
Precision	62%	65%	61%	62%
Recall	74%	74%	72%	72%
F1-Score	68%	69%	66%	67%

Based on Tables III and IV, it can be seen that the highest accuracy is obtained by applying z-score normalization and SMOTE oversampling, which is obtained by 80% using a polynomial kernel. The polynomial kernel using the parameter value C=1 γ =0.1 d=1.5 is obtained through trial and error so that it can produce margin optimization values that are used to maximize the hyperplane by mapping the data into higher dimensions. The highest precision is also obtained from the polynomial kernel, which is 65%. This shows that the higher the accuracy value, the higher the precision value will be. The highest recall was obtained at 79% which is from the RBF kernel shown in Table III. The RBF kernel uses the parameter value C=2.5 γ =1.5. The highest F1-score is also obtained from the RBF kernel shown in Table III which is 70%. The values in the parameters C, γ, and d are the most optimal values in order to get the maximum accuracy value. If the value is increased or decreased, the accuracy value will decrease.

V. CONCLUSION AND FUTURE WORK

This research produces the highest accuracy of up to 80% which is obtained from polynomial kernels. So that the shortcomings of previous research have been resolved in this study. By optimizing the use of the kernel on the SVM algorithm it is proven to be able to maximize performance. So it can be concluded that the SVM algorithm shows a better performance value in classifying Diabetes Mellitus. Where in this study it was found that the performance of the SVM algorithm kernel to produce the highest accuracy was obtained from the polynomial kernel. The accuracy produced in this study can be used as an accurate and useful information material for overcoming health problems based on the percentage of Diabetes Mellitus.

For further research, you can use other datasets that have more data and also use other algorithms such as Xgboost, Bayesian Classification and other algorithms to get better accuracy. In addition, the results of this study can also be used in making applications to detect Diabetes Mellitus which can be web-based or mobile.

REFERENCES

[1] N. Shamsiyah, "Mengenal diabetes melitus." In *Berdamai dengan diabetes*, pp. 1-12. Jakarta : Bumi Media, 2022.

[2] S. Wiyono, "Perbandingan kinerja rule zeroR dan function SMO dengan T-test dalam pengklasifikasian diagnosis penyakit diabetes mellitus." *Jurnal Teknik Elektro*, vol. 16, no. 01, pp. 23–25, 2016.

[3] Y. Mardi, "Data mining : Klasifikasi menggunakan algoritma C4.5." *Jurnal Edik Informatika*, vol. 2, no. 2, pp. 213–219, 2017.

[4] D. A. Nasution, H. H. Khotimah, and N. Chamidah, "Perbandingan normalisasi data untuk klasifikasi wine menggunakan algoritma K-NN." *CESS (Journal of Computer Engineering System and Science)*, vol. 4, no. 1, pp. 78–82, 2019.

[5] A. M. Argina, "Penerapan metode klasifikasi k-nearest neigbor pada dataset penderita penyakit diabetes." *Indonesian Journal of Data Sciencs*, vol. 1, no. 2, pp. 29–33, 2020.

[6] Noviandi, "Implementasi algoritma decision tree C4.5 untuk prediksi penyakit diabetes." *Jurnal INOHIM*, vol. 6, no. 1, pp. 1–5, 2018.

[7] M. Azhari, Z. Situmorang, and R. Rosnelly, "Perbandingan akurasi, recall, dan presisi klasifikasi pada algoritma C4.5, random forest, SVM dan naive bayes." *Jurnal Media Informatika Budidarma*, vol. 5, no. 2, pp. 640–651, 2021.

[8] Y. Zhao, C. Zhang, Y. Zhang, Z. Wang, and J. Li, "A review of data mining technologies in building energy systems : Load prediction, pattern identification , fault detection and diagnosis." *Energy and Built Environment*, vol. 1, no. 2, pp. 149–164, 2020.

[9] U. P. Harapan, D. E. Ratnawati, and A. W. Widodo, "Klasifikasi penyakit gigi dan mulut menggunakan metode support vector machine." *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 2, no. 2 2018.

[10] A. F. Rina Kurniasari, "Penerapan algoritma C4.5 untuk penjurusan siswa sekolah menengah atas." *Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)*, vol. 8, no. 1, 2019.

[11] B. A. H. and F. A. S. B. Helmi Imaduddin, "Arison of support vector machine and decision tree methods in the classification of breast cancer." *Jurnal Pendidikan Teknologi Informasi*, vol. 5, pp. 22–30, 2021.

[12] R. A. Helena Nurramdhani Irmanda, "Klasifikasi jenis pantun dengan metode support vector machines (SVM)." *JURNAL RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 4, no. 5, pp. 915–922, 2020.

[13] D. M. Ahmad Harmain, Paiman, Henri Kurniawan, Kusrini, "Normalisasi data untuk efisiensi k-means pada pengelompokan wilayah berpotensi kebakaran hutan dan lahan berdasarkan sebaran titik panas." *TEKNIMEDIA*, vol. 2, no. 2, pp. 83–89, 2021.

[14] H. E. Wahanani, M. H. P. Swari, and F. A. Akbar, "Case based reasoning prediksi waktu studi mahasiswa menggunakan metode euclidean distance dan normalisasi min-max." *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 7, no. 6, pp. 1279–1288, 2020.

[15] R. Fatwa, I. Cholissodin, and Y. A. Sari, "Penerapan metode extreme learning machine untuk prediksi konsumsi batubara sektor pembangkit listrik tenaga uap." *Jurnal Pengembangan Teknolologi Informasi dan Ilmu Komputer*, vol. 3, no. 11, pp. 10749–10755, 2019.

[16] U. Al, A. Mandar, and S. Basri, "Novelty ranking approach with z-score and fuzzy multi-attribute decision making combination." *International Journal of Engineering & Technology*, vol. 7, no. 7, pp. 476–480, 2018.

[17] T. M. Fahrudin, P. A. Riyantoko, K. M. Hindrayani, and M. H. P. Swari, "Cluster analysis of hospital inpatient service efficiency based on BOR, BTO, TOI, AvLOS indicators using agglomerative hierarchical clustering." *Telematika*, vol. 18, no. 2, p. 194, 2021.

[18] R. Siringoringo, "Klasifikasi data tidak seimbang menggunakan algoritma SMOTE dan k-nearest neighbor." *Journal Information System Development (ISD)*, vol. 3 no. 1, 2018.

[19] N. Nurajijah, D. A. Ningtyas, and M. Wahyudi, "Klasifikasi siswa SMK berpotensi putus sekolah menggunakan algoritma decision tree, support vector machine dan naïve bayes." *Jurnal Khatulistiwa Informatika*, vol. 7, no. 2, 2019.

[20] H. Nalatissifa, W. Gata, S. Diantika, and K. Nisa, "Perbandingan kinerja algoritma klasifikasi naive bayes, support vector machine (SVM), dan

random forest untuk prediksi ketidakhadiran di tempat kerja." *Jurnal Informatika Universitas Pamulang*, vol. 5, no. 4, p. 578, 2021.

[21] A. A. Kasim and M. Sudarsono, "Algoritma support vector machine (SVM) untuk klasifikasi ekonomi penduduk penerima bantuan pemerintah di kecamatan simpang raya sulawesi tengah." *SEMNASTIK*, pp. 568–573, 2019.

[22] S. A. Naufal, A. Adiwijaya, and W. Astuti, "Analisis perbandingan klasifikasi support vector machine (SVM) dan k-nearest neighbors (KNN) untuk deteksi kanker dengan data microarray." *JURIKOM (Jurnal Riset Komputer)*, vol. 7, no. 1, pp. 162–168, 2020.

[23] S. Widodo, R. N. Rohmah, B. Handaga, L. Dyah, and D. Arini, "Lung diseases detection caused by smoking using support vector machine." *TELEKOMUNIKA*, vol. 17, no. 3, pp. 1256–1266, 2019.

[24] R. Wati and S. Ernawati, " Analisis sentimen persepsi publik mengenai PPKM pada twitter berbasis SVM menggunakan python." *Jurnal Pendidikan Teknologi Informasi*, vol. 06, pp. 240–247, 2021.

[25] E. Anindika Sari, M. Thereza Br. Saragih, I. Ali Shariati, S. Sofyan, R. Al Baihaqi, and R. Nooraeni, "Klasifikasi kabupaten tertinggal di kawasan timur indonesia dengan support vector machine." *JIKO (Jurnal Informatika dan Komputer)*, vol. 3, no. 3, pp. 188–195, 2020.

[26] R. H. Muhammadi, T. G. Laksana, and A. B. Arifa, "Combination of support vector machine and lexicon-based algorithm in twitter sentiment analysis." *Khazanah Informatika: Journal Ilmu Komputer dan Informatika*, vol. 8 no. 1, 2021.

[27] M. Syukron, R. Santoso, and T. Widiharih, "Perbandingan metode smote random forest dan smote xgboost untuk klasifikasi tingkat penyakit hepatitis C pada imbalance class data." *Jurnal Gaussian*, vol. 9, no. 3, pp. 227–236, 2020.

[28] M. Rangga, A. Nasution, and M. Hayaty, " Perbandingan akurasi dan waktu proses algoritma K-NN dan SVM dalam analisis sentimen twitter." *Jurnal Inform*atika, vol. 6, no. 2, pp. 212–218, 2019.

[29] A. Ridhovan, A. Suharso, "Penerapan metode residual network (RESNET) dalam klasifikasi penyakit pada daun gandum." *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 7, no. 1, pp. 58–65, 2022.

[30] A. A. Arifiyanti and E. D. Wahyuni, "Smote : Metode penyeimbangan kelas pada klasifikasi data mining." *SCAN-Jurnal Teknologi Informasi dan Komunikasi*, vol. 15 no. 1, pp. 34–39, 2020.

[31] I. M. Yulietha and S. Al Faraby, "Klasifikasi sentimen review film menggunakan algoritma support vector machine." *eProceedings of Engineering*, vol. 4, no. 3, pp. 4740–4750, 2017.