Journal of Artificial Intelligence and Engineering Applications

Website: https://ioinformatic.org/

15th October 2024. Vol. 4. No. 1; e-ISSN: 2808-4519

Implementation of K-Nearest Neigbors for Prediction of Motorcycle Service Waiting Times in Develop Tech

Dionicho Tri Huandito[1]*, Sri Lestanti[2], Filda Febrinita[3]

[1,2,3] *Universitas Islam Balitar*
dionichotrihuandito06@gmail.com

Abstract

This research examines the application of the K-Nearest Neighbors (KNN) algorithm to predict wait times at Develop Tech, a workshop that offers affordable services but has longer spare parts wait times compared to official workshops. The research method used is descriptive quantitative, involving data collection and processing. In this study, the KNN algorithm is utilized to predict wait times by measuring distances between data points using and performing voting based on the K value to determine the final prediction. Testing on 100 data points demonstrated that KNN could predict wait times very accurately, achieving 100% accuracy, precision, and recall at certain K values. The data was split into 80% for training and 20% for testing, a method commonly used in machine learning research to ensure a balance between training data and validation. The results indicate that the KNN algorithm is reliable for predicting wait times with optimal performance at K values between 3 and 10. These findings support the conclusion that the KNN algorithm functions effectively in predicting wait times at the workshop.

Keywords: *K-Nearest Neighbors (KNN), service wait time, confusion matrix, wait time prediction*

1. Introduction

The motorcycle industry has seen substantial growth in recent years, particularly in Indonesia, where the number of motorcycles reached 160,652,675 units in 2024, according to the Central Statistics Agency. This growth is driven by economic development and the need for affordable and efficient transportation, leading to an increased demand for repair and maintenance services. Develop Tech, a general motorcycle workshop, provides a variety of services for all brands at more affordable prices compared to authorized workshops. Despite these advantages, general workshops often struggle with accurately estimating service wait times due to factors like delayed availability of parts. Accurate wait time predictions are crucial, as overly long or short estimates can negatively impact customer satisfaction and loyalty. To enhance service quality and customer satisfaction, accurate service wait time prediction is essential. Machine learning techniques, such as the K-Nearest Neighbors (KNN) Classifier, have been effective in various predictive tasks [1]. The KNN method works by identifying the closest cases in historical data based on features like service type, reported issues, and time of arrival. This research aims to develop a model for predicting motorcycle service wait times using the KNN Classifier method, leveraging historical workshop data to provide accurate estimates categorized into classes like "fast" (0-1 week) and "slow" (over 2 weeks).

Accurate predictions can optimize resource allocation, improve operational efficiency, and ultimately enhance customer satisfaction. A confusion matrix will be used to evaluate the classification process by comparing actual data with predictions, providing insight into the model's accuracy [2].This study seeks to create a reliable prediction model that can be practically implemented in workshops like Develop Tech, contributing positively to service quality and customer satisfaction in the motorcycle repair industry.

2. Theoretical Framework

2.1. Data Mining

Data mining is the process of exploring and uncovering patterns, relationships, and hidden information in large data sets using statistical techniques and database systems. It is a key stage in the "Knowledge Discovery in Databases" (KDD) process, which includes steps such as data analysis, data management, pre-processing, modeling, inference, measurement, clustering, result processing, visualization, and online updates [3].
The primary goal of data mining is to extract valuable information from various data sources, including data marts and data warehouses, for analysis. This process employs algorithms and techniques like association analysis, artificial neural networks, and classifiers to identify relevant patterns and anomalies in large volumes of data.

2.2. K-Nearest Neighbors (KNN)

The K-Nearest Neighbor (KNN) method is a non-parametric approach used for classification based on the proximity of the nearest neighbors, and it can also be used for regression. The KNN algorithm works by calculating the distance between a test data point and a set of k nearest neighbors in the training data. When new, unknown data is introduced for classification, its category is determined based on the majority category of the k nearest neighbors in the test data [4].

Determining the k-value in the KNN algorithm involves finding the closest values k1, k2, ..., ks. The larger the data, the smaller the k-value chosen; however, if the data dimensions are larger, a higher k-value should be selected. It is preferable to use odd numbers for k, such as k = 1, 3, 5, etc. The k-value must meet the condition k < N, where N is the number of training datasets since k is used to find the majority class/label in the training data and should not exceed the training dataset size [5], [6], [7].

$$\text{Dist}(X,Y) = \sqrt{\sum_{i=1} D}\ (Xi\text{-}Yi)2$$

Note:
$Dist\ (X, Y)$: Represents the Euclidean distance between two objects or points, X and Y, in a multidimensional space.
Xi: The ith feature value of the sample data point X.
Yi: The ith feature value of the test data point.
D: The number of dimensions or features in the data.
i: The index variable that iterates over each feature or dimension in the data.
The Euclidean distance is calculated as the square root of the sum of the squared differences between corresponding feature values of the two data points across all dimensions.

2.3. Confusion Matrix

A confusion matrix is a crucial evaluation tool in machine learning, particularly in classification problems. It is a table used to evaluate the performance of a classification model by comparing the model's predictions with the actual values of the test data.
The confusion matrix consists of four main cells:

1. True Positive (TP): Cases where the model correctly predicts the positive class.
2. True Negative (TN): Cases where the model correctly predicts the negative class.
3. False Positive (FP): Cases where the model incorrectly predicts the positive class when it is actually negative (Type I error).
4. False Negative (FN): Cases where the model incorrectly predicts the negative class when it is actually positive (Type II error).

From these values, various evaluation metrics can be calculated, such as accuracy, precision, recall (sensitivity), and F1-score.

Formulas for Parameters in the Confusion Matrix:
Accuracy measures how close the predicted values are to the actual values. It is calculated by the proportion of correctly classified data. The formula for accuracy is:

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{Total}}$$

True Positive (TP): Number of correctly predicted positive cases.
True Negative (TN): Number of correctly predicted negative cases.
Total: Total number of actual data points.

Precision Precision assesses the relevance of the information retrieved by comparing the number of relevant results to the total number of results retrieved, regardless of their relevance. The formula for precision is:

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}}$$

True Positive (TP): Number of correctly predicted positive cases.
False Positive (FP): Number of incorrectly predicted positive cases.

Recall Recall provides a comprehensive view of a model’s performance in classifying data. It helps understand where the model tends to make errors and can aid in adjusting the model or assessing its suitability for specific needs.

3. Research Method

This research is classified as quantitative descriptive research. It focuses on analyzing the results of the K-Nearest Neighbors algorithm applied to quantitative numerical data, specifically historical service data from the workshop in 2023. The data used is secondary data, obtained from the workshop’s service records from 2023. The data includes categories such as Type of Motor, Type of Service, Motor Condition, Spare Parts Availability, and Service Wait Time, totaling 100 data entries [8], [9], [10].

Table 1: Data from workshop

No	Type of Motor	Type of Service	Motor Condition	Spare Parts	Waiting Time
1	Honda GL 100	Piston replacement, valve seal replacement	Running	Not Available	0
2	Honda Tiger	Clutch plate replacement, porting	Running	Not Available	0
3	Honda Revo	Oil change, carburetor service	Not Running	Available	1
4	Yamaha Vixion	Chain replacement, major service	Running	Not Available	0
5	Suzuki Satria	Exhaust replacement, tire replacement	Not Running	Available	0
6	Kawasaki Ninja	Porting, valve polish, spark plug replacement	Running	Available	1
7	Honda Supra X	Oil change, light replacement	Not Running	Not Available	0
8	Yamaha Mio	Injection service, brake pad replacement	Running	Available	0
9	Suzuki Smash	Piston replacement, brake pad replacement	Running	Available	0
10	Honda Beat	Routine service, oil change	Not Running	Not Available	0

11	Honda Vario	CVT replacement, CVT service	Not Running	Available	0
12	Yamaha Nmax	Oil change, throttle service	Running	Not Available	1
13	Suzuki Address	Tire replacement, light service	Running	Not Available	0
14	Kawasaki KLX	Suspension replacement, engine service	Not Running	Available	1
15	Honda PCX	Periodic service, brake oil change	Running	Not Available	0
16	Yamaha Aerox	Spark plug replacement, CVT service	Not Running	Available	0
17	Honda CB150R	Exhaust replacement, engine service	Running	Not Available	0
18	Suzuki GSX	Brake pad replacement, injection service	Running	Available	1
19	Yamaha R15	Periodic service, tire replacement	Not Running	Not Available	0
20	Kawasaki ZX25R	Oil change, spark plug replacement	Running	Available	0
21	Honda CBR150	Chain replacement, throttle service	Running	Not Available	0
22	Honda GL 100	Piston replacement, valve seal replacement	Running	Not Available	0
23	Honda Tiger	Clutch plate replacement, porting	Running	Not Available	0
24	Honda Revo	Oil change, carburetor service	Not Running	Available	1
25	Yamaha Vixion	Chain replacement, major service	Running	Not Available	0
26	Suzuki Satria	Exhaust replacement, tire replacement	Not Running	Available	0
27	Kawasaki Ninja	Porting, valve polish, spark plug replacement	Running	Available	1
28	Honda Supra X	Oil change, light replacement	Not Running	Not Available	0
29	Yamaha Mio	Injection service, brake pad replacement	Running	Available	0
30	Suzuki Smash	Piston replacement, brake pad replacement	Running	Available	0
31	Honda Beat	Routine service, oil change	Not Running	Not Available	0
32	Honda Vario	CVT replacement, CVT service	Not Running	Available	0
33	Yamaha Nmax	Oil change, throttle service	Running	Not Available	1
34	Suzuki Address	Tire replacement, light service	Running	Not Available	0
35	Kawasaki KLX	Suspension replacement, engine service	Not Running	Available	1
36	Honda PCX	Periodic service, brake oil change	Running	Not Available	0
37	Yamaha Aerox	Spark plug replacement, CVT service	Not Running	Available	0
38	Honda CB150R	Exhaust replacement, engine service	Running	Not Available	0
39	Suzuki GSX	Brake pad replacement, injection service	Running	Available	1
40	Yamaha R15	Periodic service, tire replacement	Not Running	Not Available	0
41	Kawasaki ZX25R	Oil change, spark plug replacement	Running	Available	0
42	Honda CBR150	Chain replacement, throttle service	Running	Not Available	0
43	Honda GL 100	Piston replacement, valve seal replacement	Running	Not Available	0
44	Honda Tiger	Clutch plate replacement, porting	Running	Not Available	0
45	Honda Revo	Oil change, carburetor service	Not Running	Available	1
46	Yamaha Vixion	Chain replacement, major service	Running	Not Available	0
47	Suzuki Satria	Exhaust replacement, tire replacement	Not Running	Available	0
48	Kawasaki Ninja	Porting, valve polish, spark plug replacement	Running	Available	1
49	Honda Supra X	Oil change, light replacement	Not Running	Not Available	0
50	Yamaha Mio	Injection service, brake pad replacement	Running	Available	0
51	Suzuki Smash	Piston replacement, brake pad replacement	Running	Available	0
52	Honda Beat	Routine service, oil change	Not Running	Not Available	0
53	Honda Vario	CVT replacement, CVT service	Not Running	Available	0
54	Yamaha Nmax	Oil change, throttle service	Running	Not Available	1
55	Suzuki Address	Tire replacement, light service	Running	Not Available	0
56	Kawasaki KLX	Suspension replacement, engine service	Not Running	Available	1
57	Honda PCX	Periodic service, brake oil change	Running	Not Available	0
58	Yamaha Aerox	Spark plug replacement, CVT service	Not Running	Available	0
59	Honda CB150R	Exhaust replacement, engine service	Running	Not Available	0
60	Suzuki GSX	Brake pad replacement, injection service	Running	Available	1
61	Yamaha R15	Periodic service, tire replacement	Not Running	Not Available	0
62	Kawasaki ZX25R	Oil change, spark plug replacement	Running	Available	0
63	Honda CBR150	Chain replacement, throttle service	Running	Not Available	0
64	Honda GL 100	Piston replacement, valve seal replacement	Running	Not Available	0
65	Honda Tiger	Clutch plate replacement, porting	Running	Not Available	0
66	Honda Revo	Oil change, carburetor service	Not Running	Available	1
67	Yamaha Vixion	Chain replacement, major service	Running	Not Available	0
68	Suzuki Satria	Exhaust replacement, tire replacement	Not Running	Available	0
69	Kawasaki Ninja	Porting, valve polish, spark plug replacement	Running	Available	1
70	Honda Supra X	Oil change, light replacement	Not Running	Not Available	0
71	Yamaha Mio	Injection service, brake pad replacement	Running	Available	0
72	Suzuki Smash	Piston replacement, brake pad replacement	Running	Available	0
73	Honda Beat	Routine service, oil change	Not Running	Not Available	0
74	Honda Vario	CVT replacement, CVT service	Not Running	Available	0
75	Yamaha Nmax	Oil change, throttle service	Running	Not Available	1
76	Suzuki Address	Tire replacement, light service	Running	Not Available	0
77	Kawasaki KLX	Suspension replacement, engine service	Not Running	Available	1
78	Honda PCX	Periodic service, brake oil change	Running	Not Available	0
79	Yamaha Aerox	Spark plug replacement, CVT service	Not Running	Available	0
80	Honda CB150R	Exhaust replacement, engine service	Running	Not Available	0
81	Suzuki GSX	Brake pad replacement, injection service	Running	Available	1
82	Yamaha R15	Periodic service, tire replacement	Not Running	Not Available	0
83	Kawasaki ZX25R	Oil change, spark plug replacement	Running	Available	0
84	Honda CBR150	Chain replacement, throttle service	Running	Not Available	0
85	Honda GL 100	Piston replacement, valve seal replacement	Running	Not Available	0
86	Honda Tiger	Clutch plate replacement, porting	Running	Not Available	0
87	Honda Revo	Oil change, carburetor service	Not Running	Available	1
88	Yamaha Vixion	Chain replacement, major service	Running	Not Available	0

89	Suzuki Satria	Exhaust replacement, tire replacement	Not Running	Available	0
90	Kawasaki Ninja	Porting, valve polish, spark plug replacement	Running	Available	1
91	Honda Supra X	Oil change, light replacement	Not Running	Not Available	0
92	Yamaha Mio	Injection service, brake pad replacement	Running	Available	0
93	Suzuki Smash	Piston replacement, brake pad replacement	Running	Available	0
94	Honda Beat	Routine service, oil change	Not Running	Not Available	0
95	Honda Vario	CVT replacement, CVT service	Not Running	Available	0
96	Yamaha Nmax	Oil change, throttle service	Running	Not Available	1
97	Suzuki Address	Tire replacement, light service	Running	Not Available	0
98	Kawasaki KLX	Suspension replacement, engine service	Not Running	Available	1
99	Honda PCX	Periodic service, brake oil change	Running	Not Available	0
100	Yamaha Aerox	Spark plug replacement, CVT service	Not Running	Available	0

4. Implementation

At this stage, the implementation of Google Colab is carried out to facilitate the use of the k-nearest neighbors algorithm with the Python programming language. This implementation is supported by Python's data processing capabilities for machine learning cases, specifically for wait time classification cases. The k-nearest neighbors algorithm provided by this library works as follows:

1. First, the data will be split into two parts: the variable X or features, and the variable Y or label. In this study, the variable Y is the wait time itself.
2. The k-nearest neighbors algorithm works by calculating the distance between data points using the Euclidean distance formula.
3. The data will be divided into training data and testing data. At this stage, the data will undergo a process of repetition to calculate the distance with each training data point.
4. The results of these distance calculations will be used to perform voting according to the desired number of k. The final prediction will be obtained through the voting process, where each data point will be chosen based on the result of the voting. The class with the most votes will be the prediction result.

Below is an example source code for implementing the k-nearest neighbors algorithm:

```
# Membuat model k-nearest neighbors dengan jarak Euclidean
model = KNeighborsClassifier(n_neighbors=1, metric='euclidean')
model

KNeighborsClassifier
KNeighborsClassifier(metric='euclidean', n_neighbors=1)
```

Fig. 1: Source code

After the model training process is complete, the KNN algorithm will be used to make predictions on new data. In this case, the prediction results are classified into two categories:

1. Long Service Time**:** Categorized with a value of 1
2. Quick Service Time**:** Categorized with a value of 0

In other words, if the prediction result is 1, it indicates that the service time for the data is estimated to be long. Conversely, if the prediction result is 0, it indicates that the service time is estimated to be quick. This classification process allows users to obtain a clear estimate of how long the service time will be based on the features present in the data.

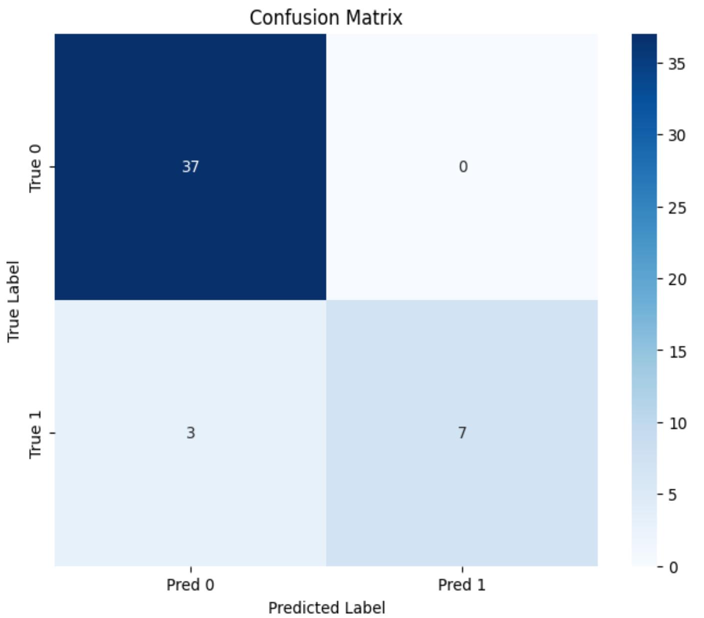

Fig. 2: Confusion Matrix

The Confusion Matrix in Figure 4.2 illustrates the model's classification results on the test data. This matrix consists of two rows and two columns, each representing the number of correct and incorrect predictions for each class (Positive and Negative). From these results, we can calculate additional evaluation metrics such as precision, recall, and accuracy:

1. Precision: This is the ratio of True Positives (TP) to the total number of positive predictions (TP + FP). Precision measures the accuracy of the model in predicting the positive class.

$$\text{Precision}=\frac{\text{TAPI}}{\text{TAPI+FP}}=\frac{7}{7+0}=\frac{7}{7}=1$$

2. Recall: This is the ratio of True Positives (TP) to the total number of actual positive cases (TP + FN). Recall measures the model's ability to identify all positive examples

$$\text{Recall}=\frac{\text{TP +TN}}{\text{TP+TN+FP+FN}}=\frac{7}{7+3}=0.7$$

3. Accuracy: This is the ratio of correct predictions (TP + TN) to the total number of predictions.

$$\text{Accuracy}=\frac{\text{TP}}{\text{TP+FN}}=\frac{7+37}{7+37+0+3}=0.92$$

This model has high accuracy and is very precise in predicting the positive class, as there are no False Positives. However, some positive examples are missed by the model, which is reflected in the lower Recall value. Overall, this model is reliable for classification tasks that require high precision.

To determine the optimal performance, running 10 different testing scenarios provides a comprehensive view of the model's performance under various conditions. Figure 4.3 shows the comparison results from the 10 testing scenarios (k1-k10), offering clear insights into the model's performance across different conditions. The testing results shown in Figure 4.3 suggest that the KNN algorithm performs exceptionally well across various scenarios with K values ranging from 1 to 10.

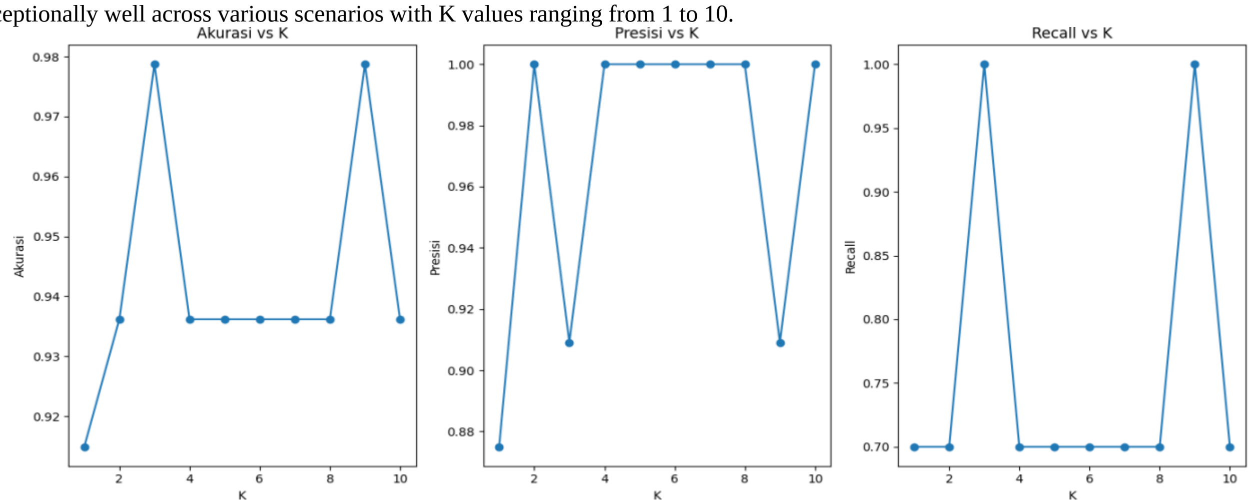

Fig. 3: K value testing

Figure 4.3 shows the comparison results for the K values across different scenarios. Based on the testing results depicted in Figure 4.3, it can be concluded that the KNN algorithm demonstrates excellent performance across various scenarios with K values ranging from 1 to 10.

5. Conclusion

Based on the discussion regarding the use of the K-Nearest Neighbors (KNN) algorithm for predicting wait times, the following conclusions can be drawn:

The implementation of the K-Nearest Neighbors (KNN) algorithm in predicting service wait times at Develop Tech's motorcycle workshop has been successfully accomplished. By using Euclidean distance to measure the distance between data points, this algorithm is capable of making accurate wait time predictions. The testing results indicate that the optimal K value ranges between 3 and 10, with accuracy reaching 100% at certain K values. Testing involving 100 data points, with an 80% training and 20% testing split, as suggested by [6]. provides a good balance between model training and validation, resulting in highly accurate predictions. The evaluation results from Figure 4.2 indicate that the model achieved an accuracy of over 92% in several scenarios. Precision was also high, exceeding 88%, while recall ranged above 70%. The F1-Score, reflecting the balance between precision and recall, was above 0.76. According to Joyce & Sepp (2021), such a high accuracy (92%) signifies strong model performance. From the tests depicted in Figure 4.3, it is concluded that the KNN algorithm performed exceptionally well across various scenarios with K values from 1 to 10. Specifically, accuracy exceeded 92%, precision surpassed 88%—reaching 100% for certain K values (3, 4, 5, 6, 7, and 10)—and recall was above 70%, also achieving 100% for K values 3 and 9.

References

[1] E. F. Rayo, A. C. P. Inaray, and B. Lule, "Capacity Strategies a Comparative Perspective in Manufacturing vs Service Industries," *J. Inform. Ekon. Bisnis*, vol. 5, pp. 1445–1452, 2023, doi: 10.37034/infeb.v5i4.759.

[2] L. Firdaus and T. Setiadi, "Perbandingan Algoritma Naive Bayes, Decision Tree, dan KNN untuk Klasifikasi Produk Populer Adidas US dengan Confusion Matrix," *J. Sist. Komput. dan Inform.*, vol. 5, no. 2, pp. 185–195, 2023, doi: 10.30865/json.v5i2.6124.

[3] A. M. Irfan, "Analisis Perbandingan Metode Naïve Bayes dan K-NN dalam Penentuan Lokasi Layanan Administrasi BPJS Kesehatan di Provinsi Maluku .," vol. 4, no. 2, 2024.

[4] F. Liantoni, "Klasifikasi Daun Dengan Perbaikan Fitur Citra Menggunakan Metode K-Nearest Neighbor," *J. Ultim.*, vol. 7, no. 2, pp. 98–104, 2016, doi: 10.31937/ti.v7i2.356.

[5] C. A. Rachma *et al.*, "Implementasi algoritma k-nearest neighbor dalam penentuan klasifikasi tingkat kedalaman kemiskinan provinsi jawa timur," 2022.

[6] Y. Lin, H. Chen, W. Xia, F. Lin, Z. Wang, and Y. Liu, "A comprehensive survey on deep learning techniques in educational data mining," pp. 1–21.

[7] R. Habibi, I. G. Prahmana, I. Ambarita, and L. A. N. Kadim, "Prediction Analysis of Literacy Numeracy and Technology Adaptation Abilities of Students Who Participate in Teaching Campuses Using the KNN Algorithm", *j. of artif. intell. and eng. appl.*, vol. 3, no. 2, pp. 590–594, Feb. 2024.

[8] A. Pratiwi, A. M. H. Pardede, and I. G. Prahmana, "Categorying Sugarcane Production Based On Factors Affecting Productivity With The K-Nearest Neighbor Algorithm", *j. of artif. intell. and eng. appl.*, vol. 3, no. 1, pp. 234–238, Oct. 2023.

[9] E. Br Milala, "Grouping of Outstanding Students at Abdi Negara Vocational School Using the K-Nearest Neighbor Method", *j. of artif. intell. and eng. appl.*, vol. 3, no. 1, pp. 428–432, Oct. 2023.

[10] E. Sulistio, "Classification For Predicting Heart Disease Using The K Nearest Neighbor Method Sylvani General Hospital Binjai City", *j. of artif. intell. and eng. appl.*, vol. 3, no. 1, pp. 521–528, Oct. 2023.