Journal of Artificial Intelligence and Engineering Applications

Website: https://ioinformatic.org/

15th February 2026. Vol. 5. No. 2; e-ISSN: 2808-4519

Prediction of Clean Water Quality Using K-Nearest Neighbor (KNN) and Naïve Bayes at PDAM Kupang City

Haliim Wila Supardi[1]*, Sumarlin,[2]

[1,2]*STIKOM Uyelindo Kupang, Indonesia*
haliimsupardhi@gmail.com [1]*, sumarlin@uyelindo.ac.id[2]

Abstract

Kupang City faces significant challenges in providing clean water due to its dry geographical conditions and extreme climate. Although it has various potential water sources such as watersheds and bore wells, clean water distribution remains suboptimal. This study aims to predict clean water quality using two machine learning algorithms, namely K-Nearest Neighbor (KNN) and Naïve Bayes, based on the Water Quality Dataset which includes parameters such as pH, hardness, total dissolved solids, and turbidity. The process involves data preprocessing, algorithm implementation, and model evaluation using classification metrics. The KNN model achieved an accuracy of 56%, with an F1-score of 0.67 for the "unsafe" class and 0.36 for the "safe" class. Meanwhile, the Naïve Bayes model achieved a higher overall accuracy of 61% but failed to detect the "safe" class, showing a precision and recall of 0.00. Overall, KNN performed more balanced across classes despite its moderate accuracy, while Naïve Bayes was biased toward the majority class. These findings highlight the importance of selecting appropriate algorithms and tuning parameters for water quality prediction. The implementation of predictive models is expected to assist PDAM Kupang in making data-driven decisions to improve clean water management sustainably.

Keywords: *PDAM, water quality, K-Nearest Neighbor, Naïve Bayes, data mining.*

1. Introduction

The Regional Drinking Water Company (PDAM) of Kupang City is a government-owned enterprise established based on Regional Regulation No. 06 of 2005, dated September 19, 2005, concerning the Formation of PDAM Kupang City. In fulfilling its duty to provide clean water services to the residents of Kupang City, PDAM requires effective and efficient work productivity to achieve its organizational goals optimally. Additionally, efficient human resource management is a key factor in operational success. Optimal productivity can only be achieved if the organization's leaders apply a high-quality leadership style. Good leadership can drive employee performance, increase organizational effectiveness, and create synergy in achieving the company's objectives [1].

In this context, the province of East Nusa Tenggara (NTT), including Kupang City, faces significant challenges related to the clean water crisis. According to the Meteorology, Climatology, and Geophysics Agency (BMKG) report in 2019, NTT is recognized as the driest province in Indonesia. Kupang City, with a population of 442,758 in 2023, has various potential water sources such as 7 watersheds, 11 reservoirs, 13 springs, and 33 bore wells. Unfortunately, these sources are still insufficient to meet the clean water needs of the entire population. Data also shows that Kupang has declared a drought emergency status along with five other districts/cities in NTT. This situation emphasizes the urgency of more efficient and sustainable water resource management due to increasing pressure from population growth and climate change. Without immediate improvement, Kupang City may face more severe consequences from the clean water crisis in the future [2].

The application of data mining methods such as K-Nearest Neighbor (KNN) and Naïve Bayes holds great potential in supporting clean water management, especially in areas experiencing water crises like Kupang. Given the city's various potential water sources, the KNN algorithm can help predict water quality based on distance, while the Naïve Bayes algorithm enables fast analysis by utilizing wide-ranging water parameters such as chemical content, pollution levels, or contamination sources to predict clean water quality. According to [3], data mining is a branch of artificial intelligence that focuses on extracting patterns from large datasets and turning them into valuable information. This technology leverages various machine learning techniques to perform automatic analysis and knowledge extraction, playing an important role across industries such as finance, weather forecasting, and science and technology. [4] explained that water quality classification can be performed using various methods. [5] classified drinking water potability using a dataset that includes 10 features such as pH, hardness, and total solids. Classification was conducted using machine learning algorithms such as Decision Tree, Naïve Bayes, and KNN.

Based on the identified issues, clean water quality prediction can be effectively addressed through the application of artificial intelligence (AI) models, particularly machine learning methods. These methods, including K-Nearest Neighbor (KNN) and Naïve Bayes, are capable of classifying data accurately and efficiently, especially when processing large datasets with many parameters, such as the Water Quality Dataset. By leveraging machine learning, decision-making related to water resource management can be conducted accurately and effectively [6].

2. Methodology

a. Naïve Bayes

The Naïve Bayes Classifier is one of the methods used in data mining to classify data based on probability calculations. This method falls under classification algorithms and applies an approach based on Bayes' Theorem. Bayes' Theorem itself is a statistical principle used to calculate the likelihood of an event. In the Naïve Bayes Classifier, the algorithm computes the probability of each class for the test data based on its attributes, then selects the class with the highest probability as the optimal classification result. The main function of this method is to identify and determine the highest probability that allows a test data instance to be classified into the appropriate category. This approach uses a simple probability prediction technique based on Bayes' rule [7].P(H|X)=P(X)P(X|H)·P(H)

b. K-Nearest Neighbors (KNN)

K-Nearest Neighbor (KNN) is an algorithm used to classify data based on training datasets by referencing the *k* nearest neighbors. The parameter *k* represents the number of closest neighbors considered in the classification process [8].
d(p,q)=√[(p1−q1)2+(p2−q2)2+...+(pn−qn)2]

c. Confusion Matrix

Confusion Matrix is one of the evaluation methods used to analyze the performance of classification algorithms, including K-Nearest Neighbors (KNN) and Naïve Bayes. This matrix provides a detailed overview of the model's predictions compared to the actual outcomes based on the test data. The confusion matrix presents the results in the form of a table containing the number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). This information is essential for calculating evaluation metrics such as precision, recall, and accuracy [9].
Accuracy=TP+TN+FP+FNTP+TN
Precision=TP+FPTP
Recall=TP+FNTP
F1-Score=Precision+Recall2·Precision·Recall.

3. Result and Discussion

3.1. Dataset Description

The dataset used in this study was obtained from PDAM Kota Kupang, the government agency responsible for clean water distribution in the area. It contains various water quality parameters routinely measured to ensure water safety for public consumption. The data is stored in an Excel file (dataa.xlsx) and consists of 10 feature columns and 1 target column representing potability labels. Each row corresponds to a laboratory-tested water sample. This study applied two classification methods—K-Nearest Neighbors (KNN) and Naïve Bayes—to the dataset. KNN classifies data by calculating Euclidean distances between test samples and training data, assigning the class based on the majority of the nearest neighbors. In contrast, Naïve Bayes predicts class probabilities under the assumption that all features are independent. Prior to model training, the data was normalized using StandardScaler to ensure uniform feature scaling. The dataset was split into 70% training data and 30% testing data, and both models were evaluated using performance metrics such as confusion matrix, accuracy, precision, recall, and F1-score.
From the non-functional aspects, the system must have an adequate level of security, including support for encryption of sensitive data such as passwords, and the implementation of role-based access control to maintain data integrity and confidentiality. As an initial step in development, the integrated information system of BPAD NTT Province is designed using an object-oriented approach by utilizing the Unified Modeling Language (UML), such as use case diagrams and class diagrams as the basis for designing the system to be developed.

3.2. The implementation of the K-Nearest Neighbor (KNN)

The implementation of the K-Nearest Neighbor (KNN) model in this study aims to classify the quality of clean water based on the physical and chemical parameters available in the dataset. The KNN algorithm works by identifying the *k* nearest neighbors of the test data based on their distance to the training data, and then determining the majority class among those neighbors as the predicted result. This method was chosen for its simplicity, intuitiveness, and effectiveness, particularly in binary classification cases such as predicting the potability of drinking water.

1. K-Nearest Neighbor (KNN) Application of the KNN model

The K-Nearest Neighbor (KNN) model is applied in this study to classify clean water quality based on the physical and chemical parameters available in the dataset. The KNN algorithm works by identifying the k nearest neighbors of a test sample based on the distance to training data, then assigning the majority class among those neighbors as the predicted result. This method is selected for its simplicity, intuitiveness, and effectiveness in binary classification problems, such as predicting water potability.

2. Classification using K-Nearest Neighbor (KNN)

KNN is a non-parametric classification algorithm that operates by calculating the shortest distance between a test point and the training data. In this study, KNN is used to predict clean water quality based on parameters such as pH, hardness, total dissolved solids, chloramines, sulfate, conductivity, organic carbon, trihalomethanes, and turbidity.

3. KNN implementation using Python

The implementation process starts with data preprocessing, which includes data cleaning, normalization, and splitting into training and testing sets. The dataset is stored in an Excel file named DATAA.xlsx and loaded using pandas. Missing values are handled by filling them with the median of each column. The data is then separated into features (X) and target labels (y). Standardization is applied using StandardScaler to ensure equal scaling across features, which is crucial as KNN is sensitive to feature magnitude. The data is then split into 70 percent training and 30 percent testing using the train_test_split function.The KNN model is trained using k = 5, meaning the prediction for each test sample is based on the 5 closest training samples using Euclidean distance. The model is built using the KNeighborsClassifier from scikit-learn, then trained with the training data.After training, predictions are made on the test set. The model's performance is evaluated using classification metrics including precision, recall, F1-score, and accuracy, generated through the classification_report and accuracy_score functions.

4. Confusion Matrix for the KNN model

In this section, a confusion matrix is used to evaluate the classification performance of the KNN model. The confusion matrix shows how well the model predicts each class compared to the actual labels, providing insight into true positives, true negatives, false positives, and false negatives. This helps in calculating more detailed evaluation metrics for model assessment.

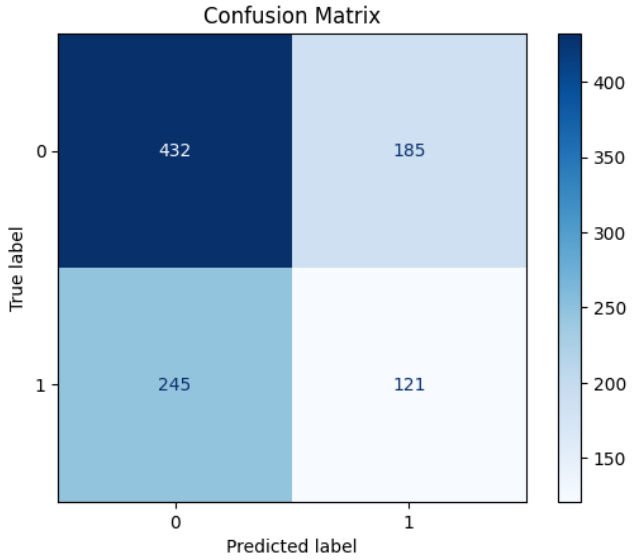

Fig. 1. Classification Report

5.Visualization of Feature Distribution Based on Potability

To better understand the characteristics of the data and observe the distribution differences of each feature between potable water (Potability = 1) and non-potable water (Potability = 0), visualization is performed using Kernel Density Estimation (KDE) plots.

6.Visualization of Feature Distribution Based on Potability

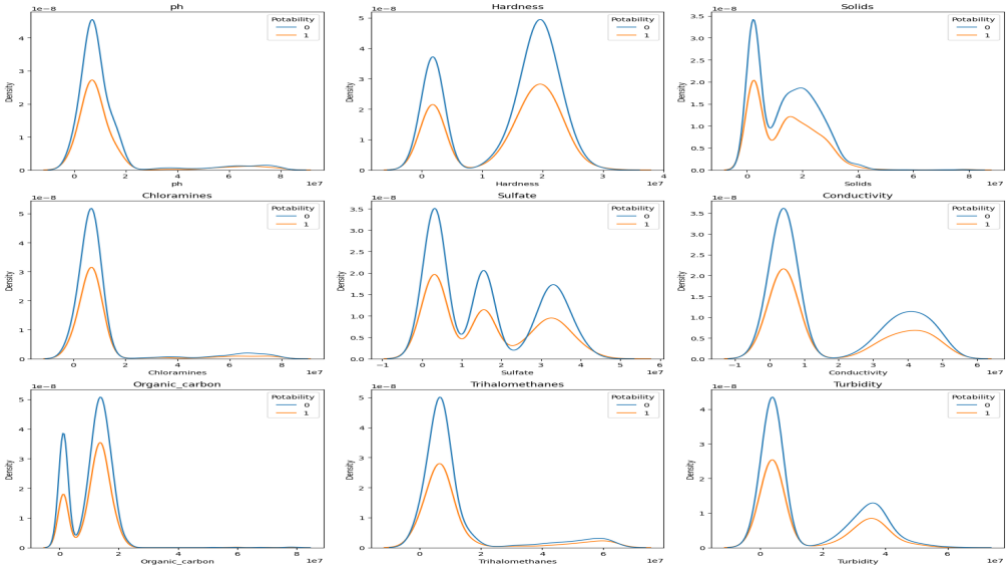

Fig.2. Visualization of Feature Distribution Based on Potability

The distribution of data is visualized using the Kernel Density Estimation (KDE) method for each numerical feature in the dataset, distinguishing between potable water (Potability = 1) and non-potable water (Potability = 0). This visualization provides a clearer overview of the value distribution patterns for each water quality parameter.

3.3. Application of the Naïve Bayes Model

Naïve Bayes is a probability-based classification algorithm used in this study as a comparison to the K-Nearest Neighbor (KNN) method. The model operates based on Bayes' Theorem with the assumption that all features are mutually independent. Although this assumption is rarely fully met in real-world data, Naïve Bayes remains effective in many classification cases due to its fast computation and reasonably accurate results. In its implementation, the data first undergoes preprocessing, including handling missing values, normalization, and separation of features and target labels. After the data is split into training and testing sets, the model is trained using the training data and evaluated using metrics such as accuracy, precision, recall, and F1-score. The results show that Naïve Bayes can serve as a simple yet reliable classification method for identifying clean water quality.

1. Naïve Bayes Implementation Process

The Naïve Bayes algorithm in this study is implemented using the Gaussian Naïve Bayes approach, which is suitable for numerical data with a near-normal distribution. The process follows similar steps as KNN, starting from loading the dataset, preprocessing, model training, and evaluating predictions

2. First, all required Python libraries are imported for data handling, modeling, evaluation, and GUI development. The dataset, named DATAA.xlsx, is then read using `pd.read_excel()` and displayed to verify its structure and content3. .
3. Next, the structure and descriptive statistics of the dataset are examined to identify data types, the number of entries, and any missing values. The preprocessing stage includes removing rows with missing values in key columns (such as pH, Sulfate, Trihalomethanes, and Potability), filling remaining missing values with the median, and deleting duplicates. Outliers are handled using the Interquartile Range (IQR) method. The features (X) and target labels (y) are separated, and all features are standardized using `StandardScaler`. The dataset is then split into training and testing sets in a 70:30 ratio with stratified sampling to maintain class balance. The cleaned dataset is saved for further analysis and interface use.
4. Finally, histograms and Kernel Density Estimation (KDE) plots are used to visualize the distribution of each feature. This helps assess how closely the data aligns with a normal distribution, which supports the assumptions of Gaussian Naïve Bayes.

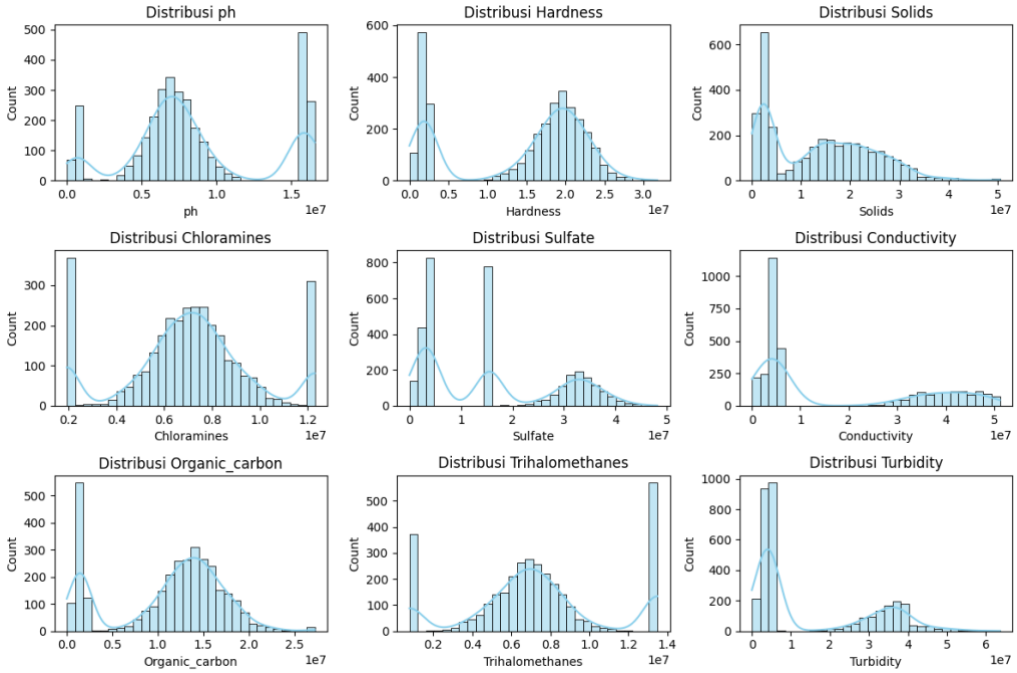

Fig. 3. Distribution Table

5. Naïve Bayes Model

This section describes the process of building and training a classification model using the Gaussian Naïve Bayes algorithm. The model is created using the GaussianNB function, which is appropriate for numerical data that follows a normal distribution. After creating the model object, it is trained using the training dataset (X_train) and the corresponding target labels (y_train). During this training phase, the model learns patterns and distributions in the data, enabling it to make predictions on new or unseen data in the next stage.

6. Evaluation of the Naïve Bayes Model

Once the Gaussian Naïve Bayes model is trained, the next step is to make predictions using the test dataset (X_test) and evaluate the results. The evaluation process begins with the generation of a confusion matrix, which is then visualized using a heatmap to illustrate the number of correct and incorrect predictions between the "Safe" and "Unsafe" water classes. A classification report is also provided, showing key performance metrics such as precision, recall, and F1-score for each class. To further assess the model's stability and reliability, five-fold cross-validation is performed using the cross-validation technique, which calculates the average accuracy across five separate training and testing cycles. This comprehensive evaluation provides an overall picture of how well the Naïve Bayes model performs in classifying clean water quality.

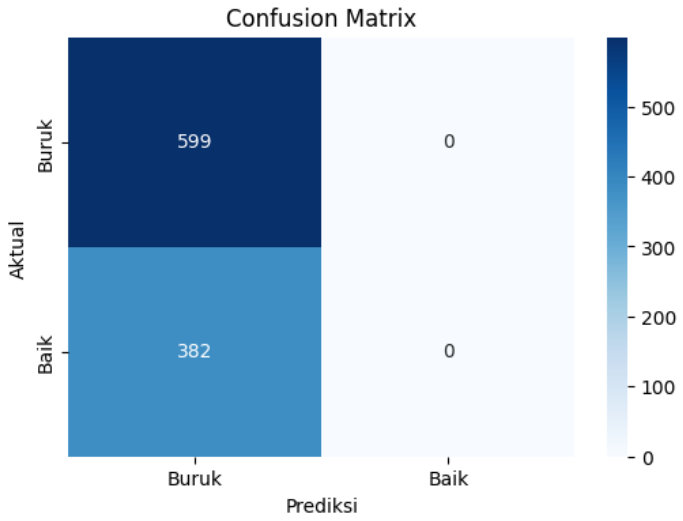

Fig. 4. Classification Report navie bayes

3.4. Model Evaluation

Model evaluation was carried out to assess the performance of classification algorithms in predicting clean water quality based on physicochemical parameters. The two models applied in this study were K-Nearest Neighbor (KNN) and Naïve Bayes. Standard classification metrics such as accuracy, precision, recall, and F1-score were used, supported by confusion matrices and cross-validation. The KNN model delivered relatively balanced results with acceptable accuracy for both "Safe" and "Unsafe" classes, depending on the chosen k value and normalization parameters. In contrast, the Naïve Bayes model showed limitations in identifying the "Safe" class, reflected in its very low precision and recall scores for that category. This performance gap is likely due to the model's assumption of feature independence and the fact that the dataset does not fully follow a Gaussian distribution, which is a core assumption of the Gaussian Naïve Bayes model. Overall, KNN proved to be more effective in detecting both classes, while Naïve Bayes remains a simpler but more sensitive approach, especially when the dataset deviates from its statistical assumptions.

3.5. Prediction GUI Implementation

To make the prediction results more accessible to users, a graphical user interface (GUI) was implemented using the Tkinter library in Python. This GUI allows users to load data files, automatically view prediction results, and display model accuracy along with the water quality status in real time. Through the GUI, users can perform predictions without needing to write any code manually. The trained models (both KNN and Naïve Bayes) are saved using Joblib, allowing them to be reloaded within the GUI application. This approach makes the clean water quality prediction system more interactive, user-friendly, and suitable for non-technical users, such as environmental agencies or the general public who want to quickly assess water potability.

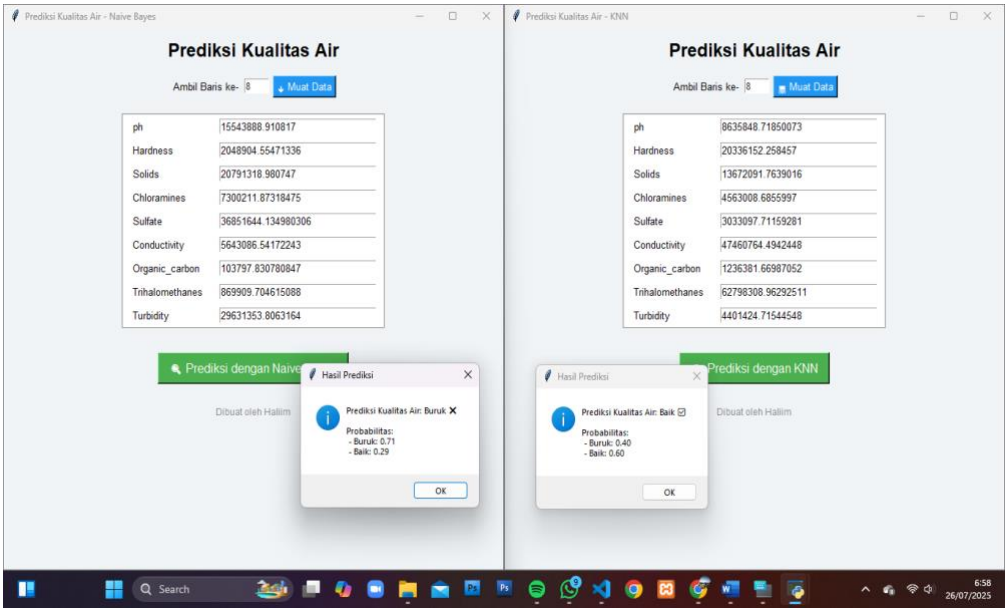

Fig. 8. Prediction GUI Implementation

3.6. Result Discussion

The results of applying both models indicate that classification approaches can be effectively used to predict water potability based on chemical and physical parameters. The KNN model provided more accurate and balanced predictions across both classes. This is because KNN compares the similarity between data points based on distance, making it well-suited for normalized data. On the other hand, the Naïve Bayes model showed limitations in recognizing the “Safe” class, likely due to the data distribution not aligning with the Gaussian assumption and class imbalance within the dataset. In terms of efficiency, Naïve Bayes is significantly faster and more lightweight compared to KNN. However, in terms of accuracy and prediction stability, KNN performs better. Therefore, selecting the appropriate method depends heavily on the specific needs of the application—whether prioritizing speed and simplicity, or focusing on accuracy and reliability in prediction.

4. Conclusion

Based on the results of this study, it can be concluded that machine learning-based classification methods can be used to predict clean water quality, with varying levels of accuracy depending on the algorithm applied. The K-Nearest Neighbor (KNN) method demonstrated better performance compared to Naïve Bayes, particularly in recognizing both classes (Safe and Unsafe) more evenly. This was reflected in the evaluation results using accuracy, precision, recall, and F1-score metrics, where KNN provided more stable and accurate predictions after proper normalization and parameter tuning. In contrast, although Naïve Bayes is simpler and faster, it showed limitations in identifying the "Safe" class, especially when the data distribution did not meet the Gaussian assumption. A graphical user interface (GUI) was also successfully developed to make the prediction system more accessible to users, making this study not only academically valuable but also practically applicable.

Acknowledgement

I would like to express my sincere gratitude to God Almighty for His blessings and guidance throughout the completion of this final project. I am deeply thankful to my supervisor, [Supervisor's Name], for the continuous support, advice, and encouragement during the research process. My appreciation also goes to the Department of Information Technology at [Your University Name] and PDAM Kota Kupang for their support and for providing the necessary data for this study. I am truly grateful to my family for their unwavering love, motivation, and prayers, as well as to my friends and classmates who have provided assistance and encouragement along the way. This project would not have been possible without their contributions and support.

References

[1]. Alvian, V., Hidayatullah, D., Nilogiri, A., Azizah, H., & Faruq, A. (2021). Klasifikasi Siswa Berprestasi Menggunakan Metode K-Nearest Neighbor (KNN) Pada SMA Negeri 2 Situbondo Classification Of Achieving Students Using K-Nearest Neighbor (KNN) Method At SMA Negeri 2 Situbondo. *Jurnal Smart Teknologi*, *1*(1), 2774–1702. **[internet]. [diakses 24 Oktober 2024].** Tersedia pada : http://jurnal.unmuhjember.ac.id/index.php/JST

[2]. Aruriansyah, S. N., Cherid, A., Santoso, H., & Rochmah, D. A. (2023). Rancang Bangun Lingkungan Pemrograman Python Dengan Metode Chatbot Pada Platform Whatsapp. *Bit (Fakultas Teknologi Informasi Universitas Budi Luhur)*, *20*(2), 82. **[internet]. [diakses 17 Oktober 2024].** Tersedia pada : https://doi.org/10.36080/bit.v20i2.2511

[3]. Azmi, B. N., Hermawan, A., & Avianto, D. (2022). Analisis Pengaruh PCA Pada Klasifikasi Kualitas Air Menggunakan Algoritma K-Nearest Neighbor dan Logistic Regression. *Jurnal Sistem Dan Teknologi Informasi*, *7*(2), 94–103. **[internet]. [diakses 17 Oktober 2024].** Tersedia pada : http://jurnal.unmuhjember.ac.id/index.php/JUSTINDO/article/view/8190%0Ahttp://jurnal.unmuhjember.ac.id/index.php/JUSTINDO/article/download/8190/4143

[4]. Fitriono, D., Wardani, S. A., Al, M. N. B., Ristyawan, A., & Daniati, E. (2024). *Perbandingan Metode Algoritma Decission Tree dan K-Nearest Neighbors untuk Memprediksi Kualitas Air yang dapat dikonsumsi*. *8*, 475–484. **[internet]. [diakses 03 November 2024].**

[5]. Imandasari, T., Irawan, E., Windarto, A. P., & Wanto, A. (2019). Algoritma Naive Bayes Dalam Klasifikasi Lokasi Pembangunan Sumber Air. *Prosiding Seminar Nasional Riset Information Science (SENARIS)*, *1*(September), 750. **[internet]. [diakses 18 November 2024].** Tersedia pada : https://doi.org/10.30645/senaris.v1i0.81

[6]. Lado, D. (2019). *Dadi Lado, Timuneno and Fanggidae/ JOURNAL OF MANAGEMENT (SME's) Vol. 10, No.3, 2019, p283-297*. *10*(3), 283–297. **[internet]. [diakses 18 November 2024].** Tersedia pada :

[7]. Mahesh, B. (2020). Machine Learning Algorithms - A Review. *International Journal of Science and Research (IJSR)*, *9*(1), 381–386. **[internet]. [diakses 18 November 2024].** Tersedia pada : https://doi.org/10.21275/art20203995

[8]. Maulidah, N., Maulidah, M., Supriyadi, R., Nalatissifa, H., Diantika, S., & Fauzi, A. (2024). Prediksi Kualitas Air Menggunakan Metode Random Forest, Decision Tree, Dan Gradient Boosting. *Jurnal Khatulistiwa Informatika*, *12*(1), 1–6. **[internet]. [diakses 18 November 2024].** Tersedia pada : https://doi.org/10.31294/jki.v12i1.16004

[9]. Natzir, S. M. (2023). Perbandingan Kinerja Model Pembelajaran Mesin dalam Prediksi Banjir menggunakan KNN, Naive Bayes, dan Random Forest. *Jurnal Teknologi Informasi*, *14*(2), 59–64. **[internet]. [diakses 18 November 2024].** Tersedia pada : https://doi.org/10.52972/hoaq.vol14no1.p59-64