Vol. 3 No. (1) (June 2025) 56-64

Journal of Data Insights

http://journalnew.unimus.ac.id/index.php/jodi

K-Nearest Neighbor (KNN) Method for Weather Data Prediction

Agata Dwi Putri[1], M Al Haris[2], Fatkhurokhman Fauzi[3], Saeful Amri[4]
[1234]Universitas Muhammadiyah Semarang, Indonesia

DOI: https://doi.org/10.26714/jodi.v3i1.214

Article Information

Artiicle History:
Submitted 23th July 2023
Revised 23th June 2025
Accepted 27th June 2025

Keywords:
KNN; Weather; Predictions.

Abstract

The weather tends to change frequently every day, so weather forecasts are made to be used as an early warning if sudden weather changes occur. By forecasting the weather, losses can be minimized and people are alert to carry out outdoor activities. From this problem, the K-Nearest Neighbor (KNN) method was applied. This method is expected to provide accurate and efficient information to obtain weather predictions for existing conditions. The data used is secondary data. After conducting research on training data (old data) amounting to 80% and test data (new data) amounting to 20%. The accuracy results from the testing data predictions are 75% with a value of k = 8.

✉ Corresponding Author:
E-mail: agatadwip@gmail.com

e-ISSN: 2988 - 2109

INTRODUCTION

In terms of life, weather significantly affects outdoor activities. Weather is an atmospheric phenomenon that lasts for some time and is measured, including changes, developments, and the emergence or disappearance of an air conditioner (Fadholi, 2013). This weather condition is influenced by temperature, air pressure, wind speed, humidity, and rainfall. To find out the weather forecast, early identification needs to be used as initial information in data mining techniques to identify problems early. The classification technique used aims to predict a label category that distinguishes one object from another; this technique has a classification algorithm that can predict, which is called K-Nearest Neighbor.

Weather is an important factor that directly affects various sectors of life, such as agriculture, transportation, tourism, and disaster mitigation. Therefore, accurate weather predictions are needed to support fast and precise decision-making. Along with the development of information technology, the use of artificial intelligence algorithms and machine learning in weather prediction is becoming increasingly relevant and growing rapidly.

One method that is quite simple but effective in data processing and prediction is K-Nearest Neighbor (KNN). The KNN method works based on the principle of historical data similarity, namely by comparing new input data to previous data to determine predictions based on its nearest neighbors (Zhang & Wang, 2013). In other studies, KNN has been shown to be able to provide competitive results in the classification and regression of weather data such as temperature, rainfall, and humidity (Bhat & Quadri, 2016; Kisku & Raju, 2020).

Several studies have shown the success of applying the KNN method in the context of short— and medium-term weather forecasts. For example, Waghmare and Patil (2014) used KNN to predict rainfall in India with a fairly high level of accuracy. Similarly, Salah and Ali (2021) compared KNN with the Support Vector Machine (SVM) method and concluded that KNN provided more stable results for weather condition classification.

In addition, this method also has advantages in terms of ease of implementation. It does not require a complex model training process, making it suitable for historical data-based applications such as weather predictions based on BMKG data (Nurhasanah & Wijaya, 2020). The reliability of KNN is also shown in various studies that compare it with other algorithms, such as decision trees and random forests (Adams & McNish, 2014; Garg & Garg, 2019).

Looking at the various research results, the application of the KNN method in weather data prediction still has excellent potential to be explored further, especially in the context of local data in Indonesia. This study aims to implement the KNN method in predicting weather data based on historical data and evaluate the level of accuracy and reliability in producing daily weather forecasts.

METHODS

Has the research work procedure been written clearly so that the experiment or the analogy of the research can be repeated with the same results? Avoid the use of imperative sentences when describing the procedure. All quantities are written in standard and consistent units. If using chemicals, it is stated explicitly with the purity and brand, written in its pure form or precursor, not in the form of a solution (example: H2 SO4 (99%, MERCK), not like this: H2 SO4 1 N). Each step is stated clearly, including the number of repetitions, and all techniques/procedures are stated (mention the name of the standard or description if the procedure is new or modified). Small and non-main tools (commonly found in the lab, such as scissors, measuring cups, and pencils) do not need to be written, but write the main equipment series or the main tools used for analysis and/or characterization, even down to the type and accuracy. Write the research location, number of respondents, how to process the results of observations or interviews or questionnaires, and how to measure performance benchmarks in full; standard methods do not need to be written in detail but

refer to reference books. For other types of qualitative research in special fields, adjust to the specifics of the field of science. The benchmark for success or performance needs to be written clearly, for example, in the form of an equation or formula or the form of criteria.

The stages in conducting this research begin with collecting and understanding data and preparing data with data preprocessing, which includes several processes such as data cleansing, data selection, and data balancing.

1.1 Data Collection and Data Understanding

The initial stage of data mining is the process of collecting critical information from a large amount of data. The information obtained includes the number of data records, data types, and data columns. Weather data was obtained from the website **www.kaggle.com**, which was uploaded and published in 2021. Five variables were used in this study.

1.2 Data Preprocessing

Data preprocessing is a technique for adding initial data to change the data structure, provide missing values, and correct value errors and inconsistencies. Preprocessing can affect the results of data mining, which produces good accuracy and clearer information; this process consists of data cleansing, data selection, and data balancing.

1.3 Split Data

The data division process is one of the methods used to assess performance by adding or reducing the percentage of training data and testing data. So that the model will later get maximum performance, in this model evaluation method, the dataset is divided into two parts: training data and test data. In this study, we will look for the highest percentage of accuracy results in several experiments conducted.

1.4 K-Nearest Neighbor

KNN is used to classify data based on training data obtained from the nearest neighbors, where k is the number of nearest neighbors (Advernesia, 2017). In the formation of the model, there are several stages of work, namely:

1. Determining the number of neighbors (k)

2. Calculating the distance of new data to the specified number of neighbors.

3. Taking the k closest neighbors with the highest accuracy to make a prediction decision based on the distance results.

The calculation of distance between new and old data objects can be measured by measuring the Euclidean Distance with the formula:

$$Euclidean\ Distance = \sqrt{(a1 - b1)^2 + \cdots + (an - bn)^2}$$

Information :
a = a1,a2,a3,… an up to the nth value
b = b1,b2,b3,…., bn up to the nth value

1.5 Model evaluation

At this model evaluation stage, a confusion matrix calculation is needed to determine the performance of the algorithm in determining the accuracy of the model. Confusion matrix is one of the techniques that can be used to measure the performance of a model, especially in the case of classification (supervised learning) in machine learning (Nugroho and Kuncahyo, 2019). The confusion matrix has four different combinations, namely predicted values and actual values; the model evaluation table is in the following table:

Table 1. Confusion Matrix for Model Evaluation

Confusion	*Matrix*	Actual Value	
		Positif	**Negatif**
Prediction Value	**Positif**	*True Positives*	*False Positives*
	Negatif	*False Negatives*	*True Negatives*

Description of the 4 values in the table, as follows:

a. True Positives (TP): the number of data that has a positive value and is expected to be positive

b. False Positives (FP): the number of data that has a negative value and is expected to be positive

c. False Negatives (FN): the number of data that has a positive value and is expected to be negative

d. True Negatives (TN): the number of data that has a negative value and is expected to be negative.

The primary function of the confusion matrix is to represent the prediction and accrual state of the data generated by the machine learning algorithm by determining the values of accuracy, precision, recall, and F1 score. These four measurements are very useful for measuring the performance of the classifier used to make a prediction. The formula for these four measurements is as follows:

$Accuracy$ $$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

Presisi : $$Precision = \frac{TP}{TP + FP}$$

Recall : $$Recall = \frac{TP}{TP + FN}$$

F1-Score : $$F1 - Score = \frac{2\ x\ Recall\ x\ Precision}{Recall + Precision}$$

RESULTS AND DISCUSSION

This study aims to create a model by applying the KNN algorithm. In this process, training data is used to store old cases that are already known. When new data is entered, or called test data, the weather can be predicted based on the proximity measure and the best accuracy results. Here is a weather dataset with 1460 data records.

	date	precipitation	temp_max	temp_min	wind	weather
0	2012-01-01	0.0	12.8	5.0	4.7	drizzle
1	2012-01-02	10.9	10.6	2.8	4.5	rain
2	2012-01-03	0.8	11.7	7.2	2.3	rain
3	2012-01-04	20.3	12.2	5.6	4.7	rain
4	2012-01-05	1.3	8.9	2.8	6.1	rain
...	...	...	...	...	...	...
1456	2015-12-27	8.6	4.4	1.7	2.9	rain
1457	2015-12-28	1.5	5.0	1.7	1.3	rain
1458	2015-12-29	0.0	7.2	0.6	2.6	fog
1459	2015-12-30	0.0	5.6	-1.0	3.4	sun
1460	2015-12-31	0.0	5.6	-2.1	3.5	sun

Figure 1. Weather Data

1. Data Processing Process

This stage checks the data to see if it has missing values or incomplete data.

a. Data Cleaning

```
1 cuaca.isna().sum()
```

```
date            0
precipitation   0
temp_max        0
temp_min        0
wind            0
weather         0
dtype: int64
```

Figure 2. Clean Dataset from Missing Values

In the research dataset, there are no missing values in each variable. If no missing value is detected, it can be continued to the next stage.

b. Data Selection

At the data selection stage, only useful data in the weather prediction category is selected. In this study, only five attributes were selected from 6 attributes.

```
# Menghapus baris NA
cuaca.dropna(inplace=True)
# Menghapus kolom 'island' dan 'sex'
cuaca.drop([ 'date'], axis=1, inplace=True)
cuaca
```

	precipitation	temp_max	temp_min	wind	weather
0	0.0	12.8	5.0	4.7	drizzle
1	10.9	10.6	2.8	4.5	rain
2	0.8	11.7	7.2	2.3	rain
3	20.3	12.2	5.6	4.7	rain
4	1.3	8.9	2.8	6.1	rain
...	...	...	...	...	...
1456	8.6	4.4	1.7	2.9	rain
1457	1.5	5.0	1.7	1.3	rain
1458	0.0	7.2	0.6	2.6	fog
1459	0.0	5.6	-1.0	3.4	sun
1460	0.0	5.6	-2.1	3.5	sun

Figure 3. New Data

2. Data Split Process

```
from sklearn.model_selection import train_test_split
# membuat data fitur/input
X = cuaca.drop('weather', axis=1)
# membuat data output
y = cuaca['weather']
# membagi data traning 80% dan data testing 20%
X_train, X_test, y_train, y_test = train_test_split(X, y,
                                   test_size=0.2, random_state=42)
```

Figure 4. Import Library, dividing data into testing and training data

Based on the image above, the data processing process is complete, continued with the data divided into 2, namely training data and test data to be used in modeling the K-Nearest Neighbor algorithm in making predictions; dividing the data into two is one of the valuable methods to find out whether the K-Nearest Neighbor algorithm model can predict the test data correctly by processing the training data and using it in the test data. Based on the image above, with a data division of 80% training data and 20% test data, the amount of training data of 1168 data records, and test data of 292 data records used in this study.

3. Creating KNN Model

```
from sklearn.preprocessing import StandardScaler

scaler = StandardScaler()

# features scaling : standardize
scaled_X_train = scaler.fit_transform(X_train)
scaled_X_test = scaler.fit_transform(X_test)
```

Figure 5. Data transformation

Before entering the stage of making the model, which can be seen in the image, it is necessary to scale the features. To avoid differences in data scale between features that will affect the performance of the resulting model. One of the scaling methods used is standardized; the standardized process will make each feature have an average value of 0 and a standard deviation of 1.

```python
#membuat model KNN
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import confusion_matrix, classification_report


# help(KNeighborsClassifier)

knn_model = KNeighborsClassifier(n_neighbors=8)

# Membuat model berdasarkan data training
knn_model.fit(scaled_X_train, y_train)

# Memprediksi/evaluasi output data testing
y_pred = knn_model.predict(scaled_X_test)

# Menampilkan confunssion matrix
print(confusion_matrix(y_test, y_pred))

print(classification_report(y_test, y_pred))
```

Gambar 1 Penerapan *Algoritma K-Nearest Neighbor*

Based on the image above, the k value used is 8. In n_neighbors, it can be replaced with the value to be tested. After getting the KNN model based on training data, it is necessary to evaluate the model to predict the output of the test data. Next, you will get the metrics used for the classification problem, namely accuracy, balanced accuracy, and f1 score. It can also be seen from the confusion matrix to compare the prediction results to the actual classification.

```
[[  1   0   1   0   7]
 [  0   2   5   0  18]
 [  0   1 101   0  18]
 [  0   1   5   1   1]
 [  0   2  15   0 114]]
              precision    recall  f1-score   support

     drizzle       1.00      0.11      0.20         9
         fog       0.33      0.08      0.13        25
        rain       0.80      0.84      0.82       120
        snow       1.00      0.12      0.22         8
         sun       0.72      0.87      0.79       131

    accuracy                           0.75       293
   macro avg       0.77      0.41      0.43       293
weighted avg       0.73      0.75      0.71       293
```

Figure 7. Model Performance Results Using Confusion Matrix

Based on the results above, an accurate model was obtained using eight nearest neighbors. Of the 9 Drizzle weather in the testing data, the model was able to predict 1 or 100% correctly. Of the 25 fog weather, the model was able to predict 2 or 33% correctly. Likewise, from the rain, snow, and sun weather, each predicted correctly by 80%, 100%, and 72%, with the highest recall value of 87%, namely sun weather. Overall, this model has an accuracy of 75%.

4. Find the best K value.

```
k =  1  => Akurasi =  0.7
k =  2  => Akurasi =  0.64
k =  3  => Akurasi =  0.74
k =  4  => Akurasi =  0.74
k =  5  => Akurasi =  0.73
k =  6  => Akurasi =  0.74
k =  7  => Akurasi =  0.74
k =  8  => Akurasi =  0.75
k =  9  => Akurasi =  0.74
k =  10  => Akurasi =  0.74
k =  11  => Akurasi =  0.74
k =  12  => Akurasi =  0.73
k =  13  => Akurasi =  0.73
k =  14  => Akurasi =  0.72
k =  15  => Akurasi =  0.72
k =  16  => Akurasi =  0.72
k =  17  => Akurasi =  0.73
k =  18  => Akurasi =  0.73
k =  19  => Akurasi =  0.72
k =  20  => Akurasi =  0.72
k =  21  => Akurasi =  0.72
k =  22  => Akurasi =  0.72
k =  23  => Akurasi =  0.72
k =  24  => Akurasi =  0.73
```

Figure 8. Comparison of the Influence of Performance on Each Value of k

Based on the results of the confusion matrix calculation above, it shows that the accuracy results obtained are 75% and can be grouped into good status with a value of k = 8 and a data split of 80% training data and 20% test data, so the k-nearest neighbor algorithm model is able to predict the weather by looking at signs of temperature and cloud shape.

CONCLUTION

The conclusion is only sufficient to answer the problem or research objective (do not be a discussion again) or produce a new theory. If the purpose is only one thing, then the conclusion is enough to refer to that objective. It is also a conclusion from the author logically and honestly: "must be based on the facts obtained."?. Implications or suggestions may be added (not mandatory). It is better to write it in paragraph form, not in the form of an item list/numbering. If there is a forced item list/numbering, but it is written in paragraph form. Based on the results of the application of KNN, it can be concluded that the applied KNN model is able to predict the weather well based on an accuracy value of 75%. With a reasonably high accuracy value, it is expected to help predict the weather so that it can help the community to do outdoor activities and maintain the body's immune system, which is at risk of causing illness.

REFERENCES

Harsani, P., & Qurania, A. (2018). Penerapan K-Nearest Neighbor (KNN) untuk Klasifikasi Anggrek Berdasarkan Karakter Morfologi Daun dan Bunga. *Komputasi: Jurnal Ilmiah Ilmu Komputer dan Matematika*, *15*(1), 118-125.

Said, H., Matondang, N. H., & Irmanda, H. N. (2022). Penerapan Algoritma K-Nearest Neighbor Untuk Memprediksi Kualitas Air Yang Dapat Dikonsumsi. *Techno. Com*, *21*(2), 256-267.

Yustanti, W. (2012). Algoritma K-Nearest Neighbour untuk Memprediksi Harga Jual Tanah. *Jurnal Matematika, Statistika dan Komputasi*, *9*(1), 57-68.

Zhang, J., & Wang, Y. (2013). Weather prediction using K-nearest neighbor model. International Journal of Computer Applications, 62(18), 1–5.

Bhat, M. A., & Quadri, S. M. K. (2016). A hybrid model using KNN for weather forecasting. Procedia Computer Science, 85, 623–630.

Kisku, D. R., & Raju, S. (2020). Application of KNN algorithm in prediction of rainfall and temperature. International Journal of Innovative Technology and Exploring Engineering (IJITEE), 9(3), 1022–1027.

Waghmare, A. R., & Patil, A. V. (2014). Prediction of rainfall using data mining technique over India. International Journal of Computer Applications, 95(16), 36–39.

Salah, A. M., & Ali, H. F. (2021). Comparative study of KNN and SVM for weather classification. Journal of Theoretical and Applied Information Technology, 99(2), 345–351.

Mohan, A., & Kumar, S. (2017). Rainfall prediction using data mining techniques. International Journal of Computer Science and Information Technologies, 8(2), 231–234.

Jain, A., & Jain, R. (2015). Weather forecasting using machine learning algorithms. International Journal of Computer Applications, 120(9), 44–49.

Adams, S., & McNish, R. (2014). Short term weather prediction using KNN and decision trees. Proceedings of the International Conference on Computer Science and Information Technology, 89–94.

Garg, A., & Garg, S. (2019). A study on weather forecasting models using KNN and Random Forest. International Journal of Computer Sciences and Engineering, 7(6), 1035–1040.

Nurhasanah, N., & Wijaya, H. (2020). Prediksi cuaca menggunakan metode K-Nearest Neighbor berbasis data historis BMKG. Jurnal Teknologi dan Sistem Komputer, 8(4), 345–351.