COMPARISON OF K-NEAREST NEIGHBORS AND NAIVE BAYES ALGORITHM FOR WEATHER FORECAST

[1]Teguh Iman Santoso, [2]Hironimus Leong
[1,2]Program Studi Teknik Informatika Fakultas Ilmu Komputer,
Universitas Katolik Soegijapranata
[2]marlon.leong@unika.ac.id

ABSTRACT

Weather is an atmospheric condition that occurs at a certain time and place. Weather is very influential in daily life. For example for agriculture, farmers usually plant corn during the rainy season. That's why accurate weather predictions are needed. Weather prediction is a way to know the future weather using science and technology.

To be able to know the weather in the future, certain datasets and algorithms are needed. The algorithm used is Naïve Bayes and K-Nearest Neighbors Both are included in the classification of data mining which aims to group data into a certain category by reading the previous data.

The result of this research is to find out which algorithm is better for weather forecast, between Naïve Bayes and K-Nearest Neighbors. Confusion Matrix is a method to determine the performance results of an algorithm. So that it can be known which algorithm is better in terms of accuracy, precision, recall, and F1 score

Keywords: Data mining, Classification, Weather Forecasting, Naïve Bayes, K-NN

INTRODUCTION

Weather is the state of the atmosphere at a certain time and place. Weather is very important for human life. Weather conditions can affect human activities, for example when the weather is rainy this affects human decisions to move outdoors or not, for farmers, this affects their decision to work on their paddy fields or not, and so on. Therefore it is important to know what the weather will be in the future. So a weather forecast or weather prediction is made, which is a process for estimating atmospheric conditions by utilizing the use of technology and science.

These changing weather conditions are caused by temperature, rainfall, wind speed, humidity, pressure, and so on. These factors will later be used as attributes in the dataset. The dataset will later be used in the classification algorithm. While the classification algorithm will be used K-Nearest Neighbors and Naive Bayes.By using K-Nearest Neighbors and Naive Bayes algorithm, the weather ahead can be predicted. The difference between the two algorithms is that The K-Nearest Neighbors algorithm

predicts the weather by counting the number of nearest neighbors. While the Naïve Bayes algorithm predicts by calculating the possibility of an event based on events that have occurred.

The results of the two algorithms will be compared using the Confusion Matrix. The Confusion Matrix

LITERATURE STUDY

In this article, [1] Andri created a weather forecast website. The dataset used in this article is obtained from a microcontroller that functions to measure wind speed, humidity, and temperature. The method used in this article is that the data obtained from the microcontroller will be divided into training and testing data. then will be used in the process of predicting the weather on the Naive Bayes algorithm. After the prediction is complete, the results will be evaluated using the Confusion Matrix. It was found that the accuracy of the Naive Bayes algorithm is quite high. Whereas Naive Bayes got 82.67%. And this proves that the Naive Bayes algorithm is good enough to be used in weather classification. This article is related to my research because in this article it explains how the Naive Bayes algorithm works. whose algorithm became part of my research.

This research, [2] Alvi made a weather prediction application for the city of Palembang using the KNN algorithm. The dataset used in this study was obtained from the Meteorology, Climatology, and Geophysics Agency (BMKG) in Palembang. The attributes contained in this dataset are minimum temperature, maximum temperature, average temperature, air humidity, irradiation duration, wind direction, wind speed, greatest wind speed, and rainfall. The research methods used include. The dataset used will be pre-processed first. Then the processes dataset uses K-NN and LSH. Meanwhile, the test will use MSE (Men Square Error). After testing the MSE value obtained is 0.301 and the accuracy value is 70%. With a fairly high accuracy value, it proves that the KNN algorithm can be used in predicting the weather.

Arif, Muhamman, Yuli [3] from Caltex Riau Polytechnic. Creating a system to detect weather conditions based on digital image processing using the K-NN algorithm. The dataset from this study was obtained in real-time using an IP camera. The research method used is, the cloud images obtained will be extracted to look for RGB features which will later be used as datasets. The dataset will be processed using the KNN algorithm to find the RGB feature equation. As for the testing method using a confusion matrix which produces an accuracy of 84.21%. With this, it can be concluded that the KNN algorithm is a classification algorithm that is good enough to be used in weather forecasting.

This article Trya, Muhammad, Rahmad, Surya, and Benny. [4] Discusses the classification of forest fires using Naive Bayes. The data selection in this article uses 5 attributes, namely temperature, rainfall, wind speed, humidity, and class. Preprocessing is done to clean up missing values, data inconsistencies, and data discrepancies. After these steps are carried out, the data is processed into the Naive Bayes algorithm. As for the evaluation method using the Confusion Matrix. After all the above processes have been carried out, it can be seen that of the four temperature training attributes, the attribute that has the highest probability is 0.978260870. and has an accuracy of 81.03% in the 2017 dataset. While the 2019 dataset has an accuracy of 82%.

It can be concluded that the Naive Bayes algorithm can be used in predicting the possibility of forest fires.

Mohammad [5] from Informatics Engineering, University of Surabaya. Create a program to compare the SVM, KNN, and CNN algorithms in classifying the weather. The dataset used in this study is the Mendeley repository (Ajayi, 2018). The method of this research is that the dataset that has been collected will be resized to a size of 64 x 64. Then the dataset is separated into training data and testing data with a ratio of 80:20. After that the data will be processed into each algorithm, namely KNN, SVM, and CNN. For testing methods using cross-validation and confusion matrix. With an average performance value of 76.6% for KNN, 86% for SVM, and 94.2% for CNN. With this, it can be concluded that the CNN algorithm has the best performance in making predictions in this case.

Indrayuni [6] created a system to predict cosmetic product reviews for Indonesian using the Naive Bayes algorithm. The data used is obtained from the site https://femaledaily.com/ which consists of 100 positive reviews and 100 negative reviews. The data obtained will be preprocessed by performing the Tokenization and Stopword Filter processes and then feature selection with Generate N-grams.n The validation method used is cross-validation. While the evaluation method used a confusion matrix to measure the accuracy and value of the training cycle. Then to measure the quality of the results the Area Under Curve (AUC) value on the ROC Curve will be measured. After doing all these methods using cross-validation = 5. The accuracy value is 90.50% while the AUC value is 0.715. With this, it can be concluded that the Naive Bayes algorithm can be used for cosmetic product reviews. [6]

In this article, [7] Rofiq, Kartika, and Yuliyantu from Computer Science at Unisa Gorontalo. Make a research on the application of data mining to determine the potential for daily rain using the KNN algorithm. In this study, the dataset was obtained using the direct interview method with the BMKG at Jalaludin Gorontalo Station. The research method is data preprocessing by accumulating daily data for one month and then calculating the average of the data. Data validation was carried out using the K-fold cross-validation technique. While the evaluation method uses the MSE Main Squer Error. By using this method it is found that the average rainfall value is 37.3333.

This research, [8] Amat, Umbar, Hengki, and Muhammad from STMIK Al Muslim, Bekasi created a program designed to clarify the eligibility of recipients of basic food assistance. The algorithm used in this research is Naive Bayes. The Train Data used has 135 data records, while the Testing Data has 40 data records. The first step is to eliminate noise data (inconsistent or irrelevant data), then group the data in which there are 7 inner attributes consisting of name, layoff status, number of dependents, head of household, condition of the house, amount of income, and status of the homeowner. At the implementation stage, the system is developed into a website using the PHP and Mysql programming languages. After testing it is known that the program has an accuracy value of 86%, Recall 85%, and Precision 88%. it can be concluded that the Naive Bayes algorithm can be used in this case with a fairly high accuracy value.

This article [9] Tia, Tia, Eka, Agus, and Anjar from STIKOM makes a classification for the development of water sources using the Naive Bayes algorithm. The data used was obtained from PDAM Tirta Lihou, the data has 9 attributes and 26 data records. The data will later be trained into the Naive Bayes algorithm and will be tested using RapidMiner tools. After testing using 19 test data. Resulting in 8 locations that are classified as feasible while 11 locations are classified as not feasible. With these results, it is known that this method has an accuracy value of 78.95%, while the precision value is 76.92%. It can be concluded that the Naive Bayes algorithm is capable of classifying suitable locations for developing water sources.

Yusuf, Valensyah, and Wawan, [10] research the application of the KNN algorithm in predicting and calculating the accuracy of weather data in Indonesia. The data used in this study were obtained from the website of the National Centers for Environmental Information. Preprocessing methods used include grouping data into several parts and cleaning data. After that, a check is carried out to find out whether the data is dependent or independent. then the data will be divided into 2, namely, training data and testing data with a ratio of 80:20. Then the data is implemented into the KNN algorithm. after calculating the accuracy value obtained using this method is 89.93%. Which is quite a large value and can prove that the KNN algorithm is suitable for use in this case.

RESEARCH METHODOLOGY

In this study, several methods used were data collection, data preprocessing, implementation and evaluation.

Data collection

The dataset used in this study is weather data obtained from the Kaggle website. Figure 1 shows an example of the dataset obtained.

Date/Time	Temp_C	Dew Poin	Rel Hum_%	Wind Spe	Visibility_	Press_kPa	Weather
01/01/2012 00:00	-1,8	-3,9	86	4	8	101,24	Fog
01/01/2012 01:00	-1,8	-3,7	87	4	8	101,24	Fog
01/01/2012 02:00	-1,8	-3,4	89	7	4	101,26	Freezing Drizzle,Fog
01/01/2012 03:00	-1,5	-3,2	88	6	4	101,27	Freezing Drizzle,Fog
01/01/2012 04:00	-1,5	-3,3	88	7	4,8	101,23	Fog
01/01/2012 05:00	-1,4	-3,3	87	9	6,4	101,27	Fog
01/01/2012 06:00	-1,5	-3,1	89	7	6,4	101,29	Fog
01/01/2012 07:00	-1,4	-3,6	85	7	8	101,26	Fog
01/01/2012 08:00	-1,4	-3,6	85	9	8	101,23	Fog
01/01/2012 09:00	-1,3	-3,1	88	15	4	101,2	Fog
01/01/2012 10:00	-1	-2,3	91	9	1,2	101,15	Fog
01/01/2012 11:00	-0,5	-2,1	89	7	4	100,98	Fog
01/01/2012 12:00	-0,2	-2	88	9	4,8	100,79	Fog
01/01/2012 13:00	0,2	-1,7	87	13	4,8	100,58	Fog
01/01/2012 14:00	0,8	-1,1	87	20	4,8	100,31	Fog
01/01/2012 15:00	1,8	-0,4	85	22	6,4	100,07	Fog
01/01/2012 16:00	2,6	-0,2	82	13	12,9	99,93	Mostly Cloudy
01/01/2012 17:00	3	0	81	13	16,1	99,81	Cloudy
01/01/2012 18:00	3,8	1	82	15	12,9	99,74	Rain
01/01/2012 19:00	3,1	1,3	88	15	12,9	99,68	Rain
01/01/2012 20:00	3,2	1,3	87	19	25	99,5	Cloudy

Figure 1. Dataset

Data Preprocessing

Date/Time	Temp_C	ew Point Temp_	Rel Hum_%	d Speed_	Visibility_km	Press_kPa	Weather
1/21/12 20:00	0	0	0,057142857	0,114285714	0,492374728	0,906417112	Clear
1/21/12 21:00	0,106382979	0,128571429	0,228571429	0,257142857	0,492374728	0,935828877	Clear
1/21/12 22:00	0,038297872	0,039285714	0,085714286	0	0,492374728	0,949197861	Clear
1/22/12 0:00	0,008510638	0,067857143	0,342857143	0	0,492374728	0,962566845	Clear
1/22/12 1:00	0,042553191	0,110714286	0,4	0,114285714	0,492374728	0,981283423	Clear
1/22/12 2:00	0	0,064285714	0,342857143	0	0,492374728	1	Clear
1/22/12 21:00	0,238297872	0,257142857	0,342857143	0,428571429	0,298474946	0,911764706	Clear
1/23/12 14:00	0,90212766	0,832142857	0,514285714	0,742857143	1	0,545454546	Rain
1/23/12 15:00	0,90212766	0,839285714	0,542857143	0,857142857	0,472766885	0,505347594	Rain
1/23/12 17:00	0,919148936	0,857142857	0,571428571	0,8	0,298474946	0,438502674	Rain
1/24/12 0:00	0,936170213	0,953571429	0,971428571	0,485714286	0,22875817	0,286096257	Rain
1/24/12 2:00	0,919148936	0,946428571	1	0,257142857	0,159041394	0,299465241	Rain
1/24/12 3:00	1	1	0,942857143	0,542857143	0,159041394	0,326203209	Rain
1/24/12 14:00	0,889361702	0,785714286	0,371428571	1	0,472766885	0,475935829	Rain
1/24/12 15:00	0,868085106	0,792857143	0,457142857	0,742857143	0,368191721	0,489304813	Snow
1/26/12 1:00	0,489361702	0,439285714	0,257142857	0,114285714	0,492374728	0,796791444	Clear
1/26/12 2:00	0,438297872	0,421428571	0,371428571	0,171428571	0,492374728	0,807486631	Clear
1/26/12 3:00	0,442553192	0,425	0,371428571	0	0,492374728	0,802139037	Clear
1/26/12 4:00	0,425531915	0,389285714	0,257142857	0	0,492374728	0,796791444	Clear
1/26/12 5:00	0,404255319	0,375	0,285714286	0,171428571	0,492374728	0,815508021	Clear
1/26/12 6:00	0,310638298	0,335714286	0,428571429	0,171428571	0,492374728	0,826203209	Clear
1/26/12 8:00	0,395744681	0,396428571	0,4	0,171428571	0,472766885	0,831550802	Clear
1/26/12 21:00	0,565957447	0,439285714	0	0,571428571	0,492374728	0,540106952	Snow
1/26/12 22:00	0,561702128	0,507142857	0,285714286	0,628571429	0,298474946	0,521390374	Snow
1/26/12 23:00	0,540425532	0,546428571	0,542857143	0,628571429	0,08714597	0,49197861	Snow
1/27/12 0:00	0,544680851	0,557142857	0,6	0,8	0,052287582	0,457219251	Snow
1/27/12 1:00	0,553191489	0,575	0,657142857	0,685714286	0	0,430481283	Snow
1/27/12 2:00	0,553191489	0,578571429	0,657142857	0,742857143	0,017429194	0,401069519	Snow
1/27/12 3:00	0,561702128	0,582142857	0,657142857	0,8	0,034858388	0,363636364	Snow
1/27/12 4:00	0,578723404	0,6	0,657142857	0,742857143	0	0,310160428	Snow
1/27/12 5:00	0,617021277	0,635714286	0,685714286	0,628571429	0,298474946	0,270053476	Snow
1/27/12 6:00	0,69787234	0,721428571	0,8	0,542857143	0,368191721	0,221925134	Snow
1/27/12 9:00	0,817021277	0,821428571	0,8	0,485714286	0,22875817	0,098930481	Rain
1/27/12 17:00	0,761702128	0,789285714	0,857142857	0,628571429	0,008714597	0	Snow
1/27/12 18:00	0,774468085	0,807142857	0,914285714	0,314285714	0	0,042780749	Snow
1/27/12 19:00	0,791489362	0,814285714	0,857142857	0,857142857	0,22875817	0,090909091	Snow
1/28/12 19:00	0,880851064	0,778571429	0,371428571	0,542857143	0,368191721	0,299465241	Snow
1/28/12 20:00	0,859574468	0,828571429	0,657142857	0,685714286	0,368191721	0,288770054	Snow
1/29/12 23:00	0,642553192	0,6	0,428571429	0,428571429	0,492374728	0,502673797	Clear
1/30/12 0:00	0,629787234	0,557142857	0,285714286	0,314285714	0,492374728	0,513368984	Clear

Figure 1. Data After Preprocessing

The data that has been obtained is preprocessed first. Preprocessing is done by deleting data if the weather attribute contains data other than Rain, Sunny, and Snow. This is done because there are many types of data on weather attributes. The next preprocessing stage is data normalization using the MinMax normalization method. MinMax is a normalization method that is liner with the original data. Normalization using minmax is done so that the data used is equal or not one-sided. With this method the data will be converted into a 0-1 scale. Figure 2 shows the data that has been carried out by the two preprocessing methods.

Then the data is divided into 2, namely Data Train and Data Test. This division is done with a ratio of 80:20. Which is 80% for train data and 20% for Test data. can be seen in figure 3 and figure 4.

1/21/12 20:00	0	0	0,057142857	0,114285714	0,492374728	0,906417112	Clear
1/21/12 21:00	0,106382979	0,128571429	0,228571429	0,257142857	0,492374728	0,935828877	Clear
1/21/12 22:00	0,038297872	0,039285714	0,085714286	0	0,492374728	0,949197861	Clear
1/22/12 0:00	0,008510638	0,067857143	0,342857143	0	0,492374728	0,962566845	Clear
1/22/12 1:00	0,042553191	0,110714286	0,4	0,114285714	0,492374728	0,981283423	Clear
1/22/12 2:00	0	0,064285714	0,342857143	0	0,492374728	1	Clear
1/22/12 21:00	0,238297872	0,257142857	0,342857143	0,428571429	0,298474946	0,911764706	Clear
1/23/12 14:00	0,90212766	0,832142857	0,514285714	0,742857143	1	0,545454546	Rain
1/23/12 15:00	0,90212766	0,839285714	0,542857143	0,857142857	0,472766885	0,505347594	Rain
1/23/12 17:00	0,919148936	0,857142857	0,571428571	0,8	0,298474946	0,438502674	Rain
1/24/12 0:00	0,936170213	0,953571429	0,971428571	0,485714286	0,22875817	0,286096257	Rain
1/24/12 2:00	0,919148936	0,946428571	1	0,257142857	0,159041394	0,299465241	Rain
1/24/12 3:00	1	1	0,942857143	0,542857143	0,159041394	0,326203209	Rain
1/24/12 14:00	0,889361702	0,785714286	0,371428571	1	0,472766885	0,475935829	Rain
1/24/12 15:00	0,868085106	0,792857143	0,457142857	0,742857143	0,368191721	0,489304813	Snow
1/26/12 1:00	0,489361702	0,439285714	0,257142857	0,114285714	0,492374728	0,796791444	Clear
1/26/12 2:00	0,438297872	0,421428571	0,371428571	0,171428571	0,492374728	0,807486631	Clear
1/26/12 3:00	0,442553192	0,425	0,371428571	0	0,492374728	0,802139037	Clear
1/26/12 4:00	0,425531915	0,389285714	0,257142857	0	0,492374728	0,796791444	Clear
1/26/12 5:00	0,404255319	0,375	0,285714286	0,171428571	0,492374728	0,815508021	Clear
1/26/12 6:00	0,310638298	0,335714286	0,428571429	0,171428571	0,492374728	0,826203209	Clear
1/26/12 8:00	0,395744681	0,396428571	0,4	0,171428571	0,472766885	0,831550802	Clear
1/26/12 21:00	0,565957447	0,439285714	0	0,571428571	0,492374728	0,540106952	Snow
1/26/12 22:00	0,561702128	0,507142857	0,285714286	0,628571429	0,298474946	0,521390374	Snow
1/26/12 23:00	0,540425532	0,546428571	0,542857143	0,628571429	0,08714597	0,49197861	Snow
1/27/12 0:00	0,544680851	0,557142857	0,6	0,8	0,052287582	0,457219251	Snow
1/27/12 1:00	0,553191489	0,575	0,657142857	0,685714286	0	0,430481283	Snow
1/27/12 2:00	0,553191489	0,578571429	0,657142857	0,742857143	0,017429194	0,401069519	Snow
1/27/12 3:00	0,561702128	0,582142857	0,657142857	0,8	0,034858388	0,363636364	Snow
1/27/12 4:00	0,578723404	0,6	0,657142857	0,742857143	0	0,310160428	Snow
1/27/12 5:00	0,617021277	0,635714286	0,685714286	0,628571429	0,298474946	0,270053476	Snow
1/27/12 6:00	0,69787234	0,721428571	0,8	0,542857143	0,368191721	0,221925134	Snow

Figure 3. Data Train

1/27/12 9:00	0,817021277	0,821428571	0,8	0,485714286	0,22875817	0,098930481	Rain
1/27/12 17:00	0,761702128	0,789285714	0,857142857	0,628571429	0,008714597	0	Snow
1/27/12 18:00	0,774468085	0,807142857	0,914285714	0,314285714	0	0,042780749	Snow
1/27/12 19:00	0,791489362	0,814285714	0,857142857	0,857142857	0,22875817	0,090909091	Snow
1/28/12 19:00	0,880851064	0,778571429	0,371428571	0,542857143	0,368191721	0,299465241	Snow
1/28/12 20:00	0,859574468	0,828571429	0,657142857	0,685714286	0,368191721	0,288770054	Snow
1/29/12 23:00	0,642553192	0,6	0,428571429	0,428571429	0,492374728	0,502673797	Clear
1/30/12 0:00	0,629787234	0,557142857	0,285714286	0,314285714	0,492374728	0,513368984	Clear

Figure 4. Data Test

Implementation

At this stage the two data that have been preprocessed will be processed using each algorithm.

Naïve Bayes Algorithm

Naive Bayes is a simple probabilistic classification algorithm that sums the frequencies and combinations of values from a given data set and then calculates the probabilities. Because the data used in this research is mostly numeric data. Then the Gaussian method will be used, by

calculating the normal distribution of data. The advantage of using the Naive Bayes algorithm is that it requires very little training data for classification.

1. First, calculate the probability value for each target attribute. First calculate the probability value for each target artibute.
2. After that, calculate the average value of each attribute for each class.
3. Then calculate the standard deviation value of each attribute for each class using the formula below.

$$\sigma = \sqrt{\frac{\sum(x_i - \mu)^2}{N - 1}} \quad (1)$$

4. Then calculate the probability of each item to the existing class using normal distribution with the formula below.

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (2)$$

5. Calculate the overall probability by multiplying each possibility that has been calculated in step 2.
6. Then compare which probability is greater that will be classified on the data.

K-Nearest Neighbors

The K-Nearest Neighbor (KNN) algorithm is a simple and easy-to-implement machine learning algorithm that can be used to solve classification problems. The K-Nearest Neighbor Algorithm (KNN) is an algorithm that functions to classify data based on learning data, taken from its k nearest neighbors. Where k is the number of nearest neighbors. This machine learning algorithm is lazy learning. Such machine learning waits for the test data to appear and then stores the training data. Classification is carried out only after obtaining the test data. They spend less time training but more time predicting. The flow of how the KNN algorithm works is as follows:

1. Choose the value for the number of K neighbors, the K value here take from the square root of the total train data. If after doing this the result is a decimal number then the number will be rounded up.
2. Calculate the distance from the number of neighbors K. The distance calculation here uses Euclidean Distance. Berikut adalah bagaimana cara menghitung Euclidean Distance.

$$dist(x,y) = \sqrt{\sum_{i=1}^{n}(xi - yi)^2} \quad (3)$$

3. Take the closest neighbor K according to the calculated distance.
4. Among these k neighbors, count the number of data points in each category.
5. Assign a new data point to the category with the most neighbors.

Evaluation

At this stage, the performance of each algorithm will be assessed using the Confusion Matrix method. This method is commonly used to assess the performance of classification methods. The Confusion Matrix will calculate accuracy, precision, recall, and F score. After these values are obtained, it can be concluded that the algorithm that has the highest value is more capablex of being used for weather forecasting cases in this dataset.

RESULTS

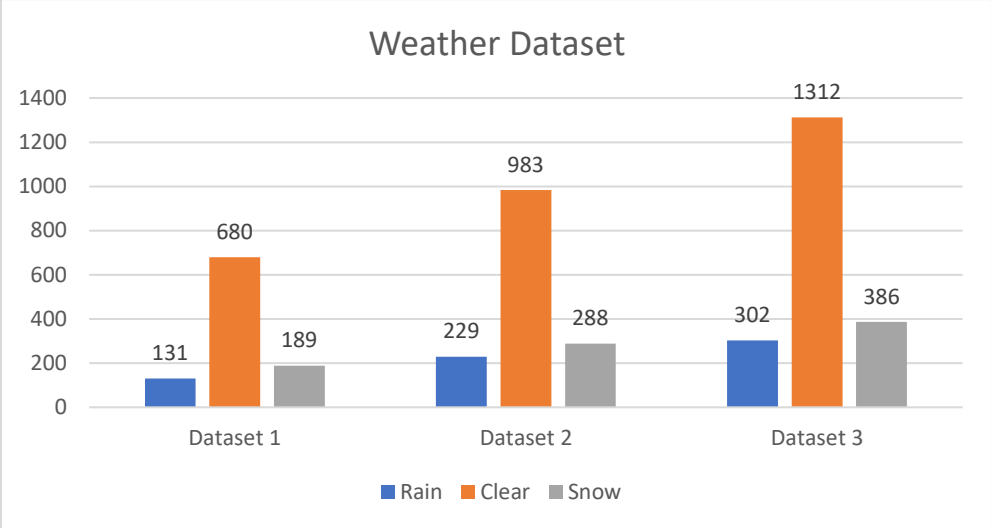

Figure 5. Weather Dataset

This research uses data taken from the Kaggel website. This data has 8784 data records. However, after preprocessing, there are only 2022 data that can be used. In this research, 3 experiments were carried out using the following data as shown in figure 5, Dataset 1 using 1000 data, Dataset 2 using 1500 data, and Dataset 3 using 2000 data. Each of these datasets has different target data. It can be seen in the figure above that Dataset 1 has 131 Rain data, 680 Clear data, and 189 Snow data, Dataset 2 has 229 Rain data, 983 Clear data, and 288 Snow data, and Dataset 3 has 302 Rain data, 1312 Clear data, and 386 Snow data. With this, it can be seen that the amount

of data that shows no rain is more than the amount of data that shows rain. Of course, this will be how the algorithm makes predictions later.

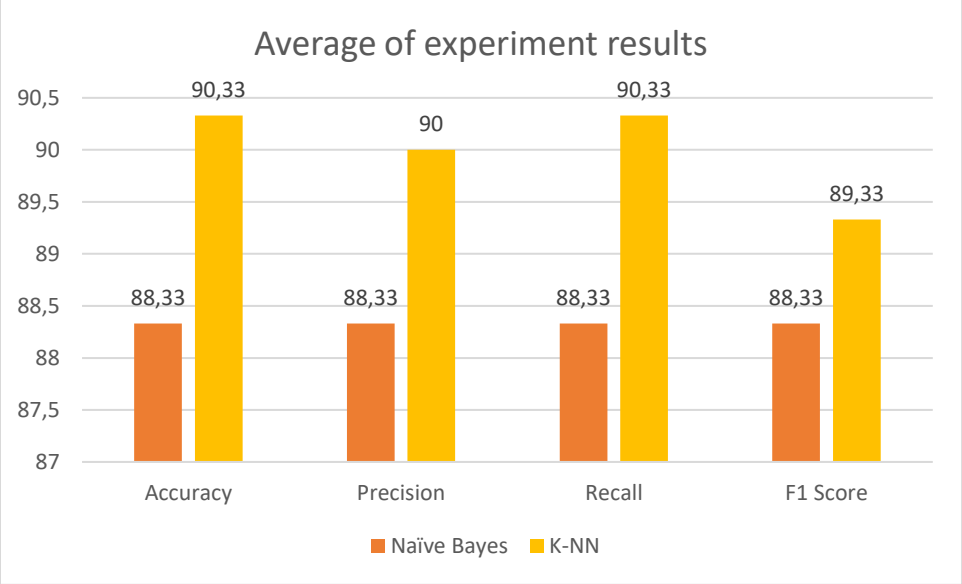

Figure 6. Average of Experiment Result

Figure 6 menunjukan the average value of the 3 experiments for the Naïve Bayes algorithm is 88,33% Accuracy, 88,33% Precision, 88,33% Recall, and 88,33% F1 Score. While the KNN algorithm is 90,33% accuracy, 90% Precision, 90,33% Recall, and 89,33% F1 Score. With this, it can be concluded that the KNN algorithm performs slightly better than the Naïve Bayes algorithm, in predicting the weather for this case. With a 2% higher accuracy value, 1,67% higher precision value, 2% higher recall value, and 1% higher f1-score value.

CONCLUSION

In this research, a system was created to compare the Naïve Bayes algorithm with KNN in the case of weather forecasting. It can be seen that the Naïve Bayes algorithm gets an average accuracy value of from 3 trials of 88,33%. Meanwhile, the KNN algorithm gets an average accuracy value of 90,33%. With this, it can be concluded that the KNN algorithm is better than the Naïve Bayes algorithm with a difference in the average value of accuracy of 2%.

Suggestions for those who want to improvise for further research can try using the Cross Validation method to improve the model to be used. The data used in this study amounted to 2022 data. So if you want to improve performance, you can do so by increasing the number of datasets to be used. This will help the algorithm in determining the value.

REFEERENCES

[1] R. Andri, "PROGRAM STUDI TEKNIK INFORMATIKA FAKULTAS TEKNOLOGI INFORMASI DAN ELEKTRO UNIVERSITAS TEKNOLOGI YOGYAKARTA 2019," p. 12. http://eprints.uty.ac.id/2916/

[2] A. S. Utami, D. P. Rini, and E. Lestari, "Prediksi Cuaca di Kota Palembang Berbasis Supervised Learning Menggunakan Algoritma K-Nearest Neighbour," vol. 13, no. 1, p. 10, 2021. https://jurnal.polsri.ac.id/index.php/jupiter/article/view/3243

[3] A. Handhoko, M. I. Zul, and Y. Fitrisia, "SISTEM DETEKSI KONDISI CUACA BERDASARKAN PENCITRAAN AWAN BERBASIS PENGOLAHAN CITRA DIGITAL MENGGUNAKAN ALGORITMA k-NEAREST NEIGHBOR (k-NN)," p. 5. https://d1wqtxts1xzle7.cloudfront.net/56471651/SISTEM_DETEKSI_KONDISI_CUACA_BERBASIS_PENGOLAHAN_CITRA_DIGITAL_MENGGUNAKAN_ALGORITMA_k-NEAREST_NEIGHBOR_k-NN-libre.pdf

[4] T. A. Pratiwi, M. Irsyad, R. Kurniawan, S. Agustian, and B. S. Negara, "Klasifikasi Kebakaran Hutan Dan Lahan Menggunakan Algoritma Naïve Bayes Di Kabupaten Pelalawan," Com, Engine, Sys, Sci, vol. 6, no. 1, p. 139, Feb. 2021, doi: 10.24114/cess.v6i1.22555.

[5] M. F. Naufal, "Analisis Perbandingan Algoritma SVM, KNN, dan CNN untuk Klasifikasi Citra Cuaca," JTIIK, vol. 8, no. 2, p. 311, Mar. 2021, doi: 10.25126/jtiik.2021824553. http://repository.ubaya.ac.id/39091/

[6] E. Indrayuni, "Klasifikasi Text Mining Review Produk Kosmetik Untuk Teks Bahasa Indonesia Menggunakan Algoritma Naive Bayes," no. 1, p. 8, 2019. https://ejournal.bsi.ac.id/ejurnal/index.php/khatulistiwa/article/view/5740

[7] R. Harun, K. C. Pelangi, and Y. Lasena, "PENERAPAN DATA MINING UNTUK MENENTUKAN POTENSI HUJAN HARIAN DENGAN MENGGUNAKAN ALGORITMA K NEAREST NEIGHBOR (KNN)," vol. 3, no. 1, 2020. http://e-journal.stmiklombok.ac.id/index.php/misi/article/view/125

[8] A. Damuri, U. Riyanto, H. Rusdianto, and M. Aminudin, "Implementasi Data Mining dengan Algoritma Naïve Bayes Untuk Klasifikasi Kelayakan Penerima Bantuan Sembako," vol. 8, no. 6, p. 7, 2021. http://www.ejurnal.stmik-budidarma.ac.id/index.php/jurikom/article/view/3655

[9] T. Imandasari, E. Irawan, A. P. Windarto, and A. Wanto, "Algoritma Naive Bayes Dalam Klasifikasi Lokasi Pembangunan Sumber Air," SENARIS, vol. 1, p. 750, Sep. 2019, doi: 10.30645/senaris.v1i0.81 http://tunasbangsa.ac.id/seminar/index.php/senaris/article/view/81.

[10] M. Y. R. Rangkuti, M. V. Alfansyuri, and W. Gunawan, "PENERAPAN ALGORITMA K-NEAREST NEIGHBOR (KNN) DALAM MEMPREDIKSI DAN MENGHITUNG TINGKAT AKURASI DATA CUACA DI INDONESIA," Hxg, vol. 2, no. 2, pp. 11–16, Jul. 2021, doi: 10.36761/hexagon.v2i2.1082. http://www.jurnal.uts.ac.id/index.php/hexagon/article/view/1082