Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 4, Number 1, May 2024
https://doi.org/10.47709/brilliance.v4i1.4105

Sentiment Analysis on Social Media X (Twitter) Against ChatGBT Using the K-Nearest Neighbors Algorithm

Asep Arwan Sulaeman[1*], Muhtajuddin Danny[2], Sufajar Butsianto[3], Suria Pratama[4]
[1,2,3,4]Informatics Engineering Study Program, Faculty of Engineering, Pelita Bangsa University, Indonesia
[1]aseparwan@pelitabangsa.ac.id , [2]utat@pelitabangsa.ac.id, [3]sufajar.rilvani@pelitabangsa.ac.id, [4]suriaprat@gmail.com

open access

***Corresponding Author**

Article History:
Submitted: 15-06-2024
Accepted: 16-06-2024
Published: 01-07-2024

Keywords:
ChatGPT; K-Nearest Neighbors (KNN); sentiment analysis; Twitter.

ABSTRACT

This research aims to analyze the public's response to ChatGPT through data obtained from Twitter. Apart from that, it is also to understand whether people's responses tend to be positive or negative towards ChatGPT, as well as to test the performance of the K-Nearest Neighbors (KNN) method in classifying sentiment patterns in tweet data. The sentiment analysis method is carried out by dividing public responses into positive and negative categories. Next, the performance of the K-Nearest Neighbors (KNN) method was tested with varying k values to classify sentiment patterns in tweet data. This testing includes dataset division, vectorization of text data using TF-IDF, initialization and training of the KNN model, and evaluation of model performance using metrics such as precision, recall, and f1-score. The results of sentiment analysis show that the majority of people's responses to ChatGPT are positive (74.3%), while 25.7% of responses are negative. Performance testing of the KNN model shows that the highest accuracy of 88% is achieved when the k value is 5. Evaluation of model performance also shows satisfactory levels of precision, recall and f1-score. Based on the research results, it was concluded that sentiment analysis and classification using KNN were effective in understanding people's responses to ChatGPT.

INTRODUCTION

In a situation of increasingly rapid advances in information and communication technology (Triyono & Febriani, 2018), ChatGPT has become an innovation that has attracted attention from the public. ChatGPT is a natural language processing technology or what is called (natural language processing / NLP) which is capable of receiving commands from users and can also answer user questions in the form of text or what can be called prompts that are entered into the application .

Sentiment analysis is a method of extracting information from text which aims to understand whether an opinion or assessment from internet users on social media is positive, neutral or negative. This technique is used to evaluate personal opinions expressed by users, providing insight into their views on something (Fransiska Vina Sari & Arief Wibowo, 2019). Sentiment analysis can provide valuable insight into how users evaluate ChatGPT's performance and help identify potential improvements (Lubis et al., 2024). By continuously monitoring user sentiment, the development team can respond to feedback more effectively, ensuring that ChatGPT continues to evolve according to user expectations and needs (Irwansyah Suwahyu et al., 2024).

This research uses the KNN classification method, namely K-Nearest Neighbors, this method was chosen because it has the advantage of classifying data based on the level of similarity with the nearest neighbors. KNN as a classification method is classified as a relatively easy approach without requiring complicated calculations (Endang Sholihatin et al., 2023). By using this method, it is hoped that this method can identify sentiment patterns that appear in tweets on Twitter (Novianti & Wibowo, 2022).

The existence of ChatGPT as a smart language processing technology that is being widely discussed in various places has given rise to various views, both positive and negative, from users. Therefore, this study aims to investigate what people think about ChatGPT, whether their views tend to be positive or negative. In this research, the author will use a simple approach with the K-Nearest Neighbors (KNN) Method to better understand how people see the advantages and disadvantages of ChatGPT. Thus, this research not only helps understand people's overall opinion of ChatGPT, but also contributes to further understanding of how language processing technology is received by society (Irsalinda et al., 2021).

LITERATURE REVIEW

Faiza Rizqi Irawan, Ahmad Jazuli, Tutik Khotimah (Rizqi Irawan, 2022), in Sentiment Analysis Of Gojek Users Using The K-Nearset Neighbors Method explained that the application of the K-Nearest Neighbor method in classifying Twitter user responses can be a basis for evaluating and assessing Gojek services for the company. Testing this method using a confusion matrix on a dataset of 1409 shows an accuracy level of 79.43% with a value of k=15.

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 4, **Number** 1, May 2024
https://doi.org/10.47709/brilliance.v4i1.4105

The research process begins with collecting data (crawling data), managing data (data preprocessing), labeling the data (labeling), classifying using the KNN algorithm, and finally, carrying out evaluation. The evaluation results show an accuracy level of 94.33% in classifying data. This opinion is the result of research by Muhamad Trian Diwandanu, Lu'lu Mawaddah Wisudawati (Diwandanu & Wisudawati, 2023), in his journal entitled Sentiment Analysis of Twit Maxim On Twitter Using R Programming And K Nearest Neighbors.

The Flouting Maxim on Twitter Influencers' Tweets, this research aims to determine the use of maxim principles in tweets made by certain social media influencers in Indonesia whose method was carried out qualitatively. This research is limited to whether users comply with cooperative principles, maxims, especially the maxim of relevance, what purpose users usually violate these maxims(Hassani, 2019). The results obtained vary: most of the conversations do not meet the principle of the maxim of relevance, or in other words do not imply the principle of the maxim of relevance. Moreover, the goal is to crack jokes, and to keep the conversation going smoothly while engaging in good manners (Syaifuddin et al., 2021).

Analysis Of Sentiment Towards The Community For New Banknote Using The K-Nearest Neighbor (KNN) Algorithm is research from Septi Hasanah, Intan Purwasih, Imam Santoso (Septi Hasanah et al., 2023). The research used K-Nearest Neighbor on 510 data for sentiment analysis, achieving an accuracy of 75.06%. The process involves data crawling, pre-processing, and classification with RapidMiner. Evaluation was carried out by taking True Positives and True Negatives, showing positive sentiment of 67.74% and negative 76.65%. In conclusion, the KNN method with RapidMiner is effective in sentiment analysis with quite high accuracy (Rinata & Dewi, 2019).

Sentiment Analysis Regarding the Sinovac Vaccine Using the Support Vector Machine (SVM) and K-Nearest Neighbor (KNN) Algorithms (Anna Baita et al., 2021), research analyzed netizens' opinions about the SiNovac vaccine from 1030 tweets on Twitter, with 1030 positive, 388 negative and 687 neutral. SVM and KNN were used for sentiment classification, with SVM achieving an average accuracy of 0.7, better than KNN (0.56). SVM performs best with a linear kernel (Sandova et al., 2024), while KNN achieves the best results with 76 neighbors. However, the accuracy value is low, thought to be due to automatic labeling using TextBlob. Future research is recommended to use manual labeling methods or other platforms such as VADER or spaCy for more accurate results. The results of this research are listed in a journal entitled Sentiment Analysis Regarding The Sinovac Vaccine Using The Support Vector Machine (SYM) And K-Nearest Neighbor (KNN) Algorithm (Anna Baita et al., 2021).

METHOD

Research methods

This research methodology uses the Knowledge Discovery Database (KDD) process as a guide for the steps that will be applied at each stage of the research which will be explained in this research method. The series of research processes will be described in detail to ensure the smoothness and clarity of each step taken. The following image from 3.1 shows the method used in this research, namely the Knowledge Discovery Database, which is as follows

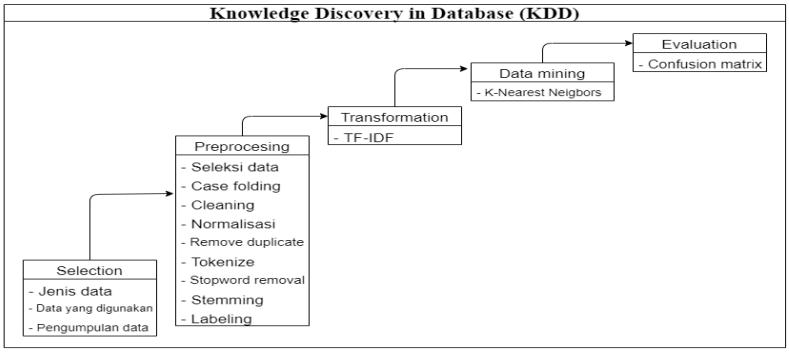

Figure 1. KNowledge Discovery in Database (KDD) Method

1. Selection

The Selection stage is the first step in the Knowledge Discovery in Databases (KDD) process. At this stage, you will determine the appropriate type of data, select the dataset that will be used, and outline the data collection process that will be applied. Each step in this selection is very important because it determines the foundation of the analysis that will be carried out.

a. Data Type

This type of data is qualitative in nature which is collected by data crawling and involves tweets posted by Twitter users. The data is in text form and includes a variety of opinions related to ChatGPT.

b. Data used

The data used in this research is tweet data from the social media Twitter. where the data is the result of crawling data searched based on the keyword "chatgpt" and searched randomly in 2023 in Indonesian and using an auth token.

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 4, **Number** 1, May 2024
https://doi.org/10.47709/brilliance.v4i1.4105

Of course, you must have a Twitter account, then the data will be saved in .csv file format and then ready to be processed to the data preprocessing stage.

c. Data collection

In the process of collecting data for sentiment analysis, the first thing is to take text data from Twitter social media using Google Colab. You definitely have to have a Twitter account to get the auth token code needed at the next stage. At the search data collection stage, Indonesian text was used and the keyword "chatgpt" was used. The data search process was carried out in 2023, and after crawling the data the results were obtained in the form of 10,722 data. Then the crawling results can be exported into a CSV file for further processing. The following is an example of the results of crawling Twitter data which can provide an overview of the data that has been collected. Below is an image of the results of collecting tweet data on Twitter, as follows:

created_at	id_str	full_text	lang	user_id_str	conversation_id_str	username	tweet_url
Fri Feb 10 02:36:13 +0000 2023	1623873440984989697	Lelaki Ini Ajar Cara Mudah Jana RM100 Sehari D...	in	2481818051	1623873440984989697	thevocket	https://twitter.com/thevocket/status/162387344...
Tue Feb 28 16:34:49 +0000 2023	1630607463807598592	average chatGPT moment fr. pertanyaan bodoh ap...	in	729629351902220288	1630607463807598592	arslechanneling	https://twitter.com/arslechanneling/status/163...
Thu Feb 23 09:30:00 +0000 2023	1628688616250830848	Keberadaan #ChatGPT menohok #Google sebagai me...	in	1192046460210008064	1628688616250830848	BNchanneltv	https://twitter.com/BNchanneltv/status/1628688...
Fri Feb 17 10:29:23 +0000 2023	1626529236092882944	Kekuatan utama chatGPT adalah kreatifitas user...	in	1043171583483817986	1626529236092882944	arsyazidane	https://twitter.com/arsyazidane/status/1626529...
Wed Mar 08 06:16:07 +0000 2023	1633350868761296902	@OfficialDGISPR #chatgpt ko nanga kero ghaddar...	in	69813417	1633350868761296902	farazkontakts	https://twitter.com/farazkontakts/status/16333...

Figure 2. Results of Tweet Data Collection on Twitter

2. Preprocessing

Data preprocessing includes a series of steps or methods that are applied to crawled data before it is used in the next stage. This process aims to clean, organize and prepare data so that it can be processed optimally by machine learning algorithms. There are several stages in data preprocessing that will be used for this research as follows.

a. Data Selection

In this stage, data selection is carried out to eliminate data that is not related to the dataset. Data selection focuses on the "full_Text" column as the main focus, where only text data is taken. The goal of this step is to identify and select data that makes a significant contribution to the dataset. This process ensures that the dataset used is more focused and only contains the necessary text, thereby simplifying analysis and further processing using the "full_Text" column. The following is an example of an image resulting from data selection which can be seen as follows:

full_text
@AbdulramonJemil @qhkmdev9 ChatGPT API
semoga dospim gua kaga tau ada teknologi yg namanya chatgpt
Barusan nyobain Chatgpt, bukan main loh ini..
Pak kalo ga bisa bikin alesan buat boong berbayar. Ada chatgpt kok kalo mau gratis
Chat GPT adalah topik yang sedang dibicarakan.Terutama, waktu saya menyeliki, saya mudah menemukan jawaban. #Bahasaindonesia
@hafiyakmal5 nnti semua hanya bisa chatgpt

Figure 3. Examples of Data Selection Results

b. Case folding

At this stage, letter processing is carried out by changing sentences or text containing capital letters to lowercase letters. This process aims to equate word forms uniformly. For example, steps for processing letters or case folding can be found in the following table.

Table 1. Example of Case Folding Processing Process

Before Process Case Folding	After Process Case Folding
Like an elementary school child finding a calculator. "Still taught to count, add, multiply using traditional methods." But also introduce that calculators are useful and can be used at certain times. They are not haram. ChatGPT and other AI applications are like that too."	Like an elementary school child finding a calculator. still taught to count, add, multiply using traditional methods". But also introduce that calculators are useful and can be used at certain times. They are not haram. Chatgpt and other AI applications are like that too.

c. Cleaning Text

Cleaning Text is a series of steps or processes carried out to remove unwanted elements from text data. The goal is to remove elements such as links, retweets, emoticons, hashtags and others that do not play a role in sentiment

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 4, **Number** 1, May 2024
https://doi.org/10.47709/brilliance.v4i1.4105

analysis. These elements are considered not to have a significant contribution in determining sentiment, so they need to be removed during the text cleaning stage. Below is an example of a table before going through the cleaning stage and after going through the cleaning stage.

Table 2. Examples of Cleaning Text Processing Process

Before the Text Cleaning Process	After the Text Cleaning Process
@wih Artificial intelligence (AI) technology is becoming increasingly widespread. Technological giants like Google don't even want to be left behind and are trying to master this technology too. cr: https://t.co/osfuovvonl #technews #google #artificialintelligence #chatgpt #openai https://t.co/sepgfihenr	wih tek*No*logi artificial intelligence ai semakin merajalela nih raksasa tek*No*logi seperti google bahkan tidak ingin ketinggalan dan berusaha untuk menguasai tek*No*logi ini juga.

d. Text Normalization

Text normalization in tweet data is very important for changing informal words into formal ones, as well as dealing with abbreviated words so that they can be explained in their actual word form. This Normalization action is very important because many social media users, especially on Twitter, often use informal language or abbreviated words. These steps help in the preparation of tweet data for further analysis or natural language processing, thereby ensuring consistent and accurate understanding and processing of the data. Below is an example before going through the Normalization stage and after going through the Normalization stage.

Table 3. Examples of Text Normalization

Before the Text Normalization Process	After the Text Normalization Process
Ntar	Nanti
Yg	Yang
Tp	Tapi
Gue	Aku
Begimana	bagaimana
Real	nyata
pinjem	pinjam

3. Transformations

After carrying out preprocessing, then in the transformation stage, the Text data will be divided. The division of this data is test data and training data, 70% of the training data will be taken while 30% of the test data will be taken. To maintain consistency in data division, random state is used as a parameter to control the randomness factor that appears in the division process.

Then there is Term Frequency-Inverse Document Frequency (TF-IDF) feature extraction which is used to measure the importance of words in the text. Tf-Idf takes into account the frequency of words in a particular text and the frequency with which they appear in all texts. Training data and test data are converted into Tf-Idf representation to simplify text analysis and classification processes. This helps simplify complex information so that it can be processed more efficiently in the study.

4. Classification Knn

At the classification process stage in this research, the K-Nearest Neighbors algorithm was used. The data that will be classified is tweet data which is divided into positive and negative sentiment. This process involves determining the nearest neighbors for each test data, followed by training a classifier using training data that has been converted into a Term Frequency-Inverse Document Frequency or Tf-Idf model. The training process aims for the model to learn patterns from each data point in the training data, including the class label. The trained model can be used to make predictions on the test set.

5. Evaluation

After the KNN model makes predictions on the test data set, the results will then be evaluated in this research. Researchers focus on several important metrics to measure the performance of the classification model. The evaluation results include accuracy, precision, recall and f1-score, all of which will be measured using the Confusion matrix and classification report.

Testing

In this test, we will apply the K-Nearest Neighbor method with different k values: 3, 5, 7, 9, and 11. The test will use the Euclidean Distance metric and divide the training data by 70% and testing data by 30%. Below is a table detailing the test details.

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 4, **Number** 1, May 2024
https://doi.org/10.47709/brilliance.v4i1.4105

Table 4. Tests

No.	K Value	Testing	Comparison ratio (%)
1	3	K-Nearest Neighbor	70% training data 30% test data
2	5		
3	7		
4	9		
5	11		

In each test, the data will be divided by the ratio mentioned above, and the K-Nearest Neighbor method will be applied with the appropriate k. The results will be evaluated to understand the performance of the model with each value of k used.

RESULT

Data Analysis

Response data analysis was carried out to obtain a comprehensive understanding of responses to ChatGPT after going through the Text preprocessing and automatic labeling process using the TextBlob Library. Response data collected in 2023 using the random selection method is divided into two category classes, namely positive responses and negative responses. The graph below provides an overall picture of the responses obtained by each category.

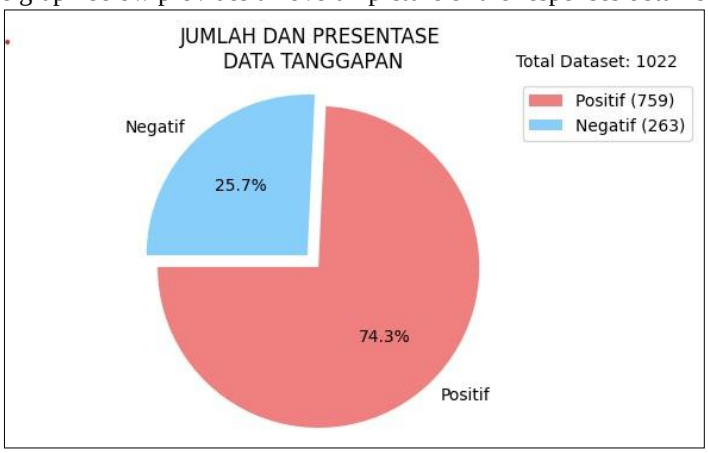

Figure 4. Graph of Data Analysis Results

Based on the graphic data above, it can be seen that the total number of responses obtained was 1022, with 759 responses (74.3%) being positive, indicating the public's view that they appreciate the existence and performance of ChatGPT. Meanwhile, there were 263 responses (25.7%) which were negative, reflecting the views of the public who did not like aspects of ChatGPT's existence and performance.

Looking at the percentage graph above, it can be concluded that the public's perception of the existence and performance of ChatGPT is more dominant on the positive side, with 759 responses (74.3%), compared to negative responses.

Testing Process

In the testing process, the dataset is divided equally, namely 70% for training data and 30% for test data. The next step involves calculating the Euclidean distance in the K-Nearest Neighbors algorithm using the Python programming language with Google Colab. Then, the test results by the system are compared with the actual classification, and the representation is given through the Confusion matrix table along with Performance Metrics, namely precision, recall, f1-score, and accuracy.

In implementing the K-Nearest Neighbors algorithm, Euclidean distance calculations are carried out using several Python libraries and modules from scikit-learn (sklearn). Details and explanation of Library usage can be found in the following table.

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 4, **Number** 1, May 2024
https://doi.org/10.47709/brilliance.v4i1.4105

Table 5. Library used

Libraries used	Explanation of library use
import pandas as pd	Read and work with data in the form of dataframes.
from sklearn.model_Selection import train_test_split	Used to divide data into training and testing sets
from sklearn.feature_extraction.Text import TfidfVectorizer	Used to vectorize text using the TF-IDF scheme. This converts text into a numerical representation that can be used by machine learning algorithms
from sklearn.neighbors import KNeighborsClassifier	Used to implement the K-Nearest Neighbors classification algorithm.
from sklearn.Metrics import classification_report, confusion_matrix	Contains several evaluation metrics such as classification report and Confusion matrix to evaluate model performance.

Below are the steps that will be explained by calculating the Euclidean distance in the testing process of the K-Nearest Neighbor method, as follows.

a. Reading data

At this stage, the process of reading the dataset which is saved in the .csv file format is carried out. The pandas library is used to read the .csv file and load the data into a data structure known as a dataframe. Below is an image of the source code for reading CSV files.

```
# Tentukan path ke file CSV
file_path = f"data.csv"
# Baca file CSV ke dalam DataFrame pandas
df = pd.read_csv(file_path)
```

Figure 5. Source Code Reading Files

b. Data sharing

Data division divides the data into 70% training data and 30% test data. The use of random_state=22 in this function aims to provide consistency in data distribution. By setting a certain random_state value, for example 22, data sharing will not occur randomly every time the code is run. Below is an image of the source code for dividing test data and training data.

```
# Bagi data menjadi set pelatihan dan pengujian
X_train, X_test, y_train, y_test = train_test_split(df['text'],
df['label'], test_size=0.3, random_state=22)
```

Figure 6. Source Code for Data Sharing

c. Vectorize text data using Tf-Idf

At this stage, the text from the training set and testing set will be converted into vector form using the Tf-Idf method. This process is carried out using TfidfVectorizer. After the vectorization process, examples of features and the Tf-Idf matrix will be displayed in array form. Below is an image of the source code for data vectorization using Tf-Idf and the output results from text data vectorization.

```
# Vektorisasi data teks menggunakan TF-IDF
vectorizer = TfidfVectorizer(max_features=1022)# max_features berdasarkan ukuran data
X_train_tfidf = vectorizer.fit_transform(X_train)
X_test_tfidf = vectorizer.transform(X_test)

# Menampilkan fitur (kata-kata atau n-gram) yang digunakan oleh vektorizer
features = vectorizer.get_feature_names_out()
print("\nContoh Fitur:")
print(features)

# Menampilkan matriks TF-IDF untuk data latih (hanya beberapa baris pertama dan 5 kolom)
print("\nMatriks TF-IDF untuk Data Latih:")
print(X_train_tfidf.toarray()[:5, :5])

# Menampilkan matriks TF-IDF untuk data uji (hanya beberapa baris pertama dan 5 kolom)
print("\nMatriks TF-IDF untuk Data Uji:")
print(X_test_tfidf.toarray()[:5, :5])
```

Figure 7. Data Vectorization Source Code using Tf-Idf

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 4, **Number** 1, May 2024
https://doi.org/10.47709/brilliance.v4i1.4105

```
Contoh Fitur:
['ada' 'adalah' 'adanya' ... 'youtube' 'zaman' 'zoom']

Matriks TF-IDF untuk Data Latih:
[[0.29676283 0.         0.         0.         0.        ]
 [0.         0.33033634 0.         0.         0.        ]
 [0.         0.         0.         0.         0.        ]
 [0.         0.         0.         0.         0.        ]
 [0.         0.         0.         0.         0.        ]]

Matriks TF-IDF untuk Data Uji:
[[0.         0.         0.         0.         0.        ]
 [0.23272792 0.         0.         0.         0.        ]
 [0.         0.         0.         0.         0.        ]
 [0.         0.         0.         0.         0.        ]
 [0.         0.         0.         0.         0.23340184]]
```

Figure 8. Output results of data vectorization using Tf-Idf

d. Initialization of the knn classification model and training of the knn model

In this section we will use the K-Nearest Neighbors algorithm with 5 nearest neighbors, and use Euclidean distance as a metric to measure the closeness between data. The model is then trained using training data that has been converted into Tf-Idf. The following is an image of the source code for knn classification and knn model training.

```
# Inisialisasi pengklasifikasi KNN
knn_classifier = KNeighborsClassifier(n_neighbors=5, metric='euclidean')

# Latih pengklasifikasi
knn_classifier.fit(X_train_tfidf, y_train)
```

Figure 9. Source Code for Initialization and Training of the KNN Model

e. Make predictions

Next, at this stage, classification results are predicted using the K-Nearest Neighbors method by calculating the Euclidean distance to the test data. The following is an image of the source code for making predictions on the knn test set and the output results from the knn model predictions.

```
# Lakukan prediksi pada set pengujian
y_pred = knn_classifier.predict(X_test_tfidf)
# Tampilkan sejumlah sampel hasil prediksi
print("Hasil Prediksi (sejumlah sampel):")
print(y_pred[:35])  # 35 jumlah sampel yang ingin ditampilkan
```

Figure 10. Source Code making predictions on the KNN Test Set

```
Hasil Prediksi (sejumlah sampel):
['Positif' 'Positif' 'Positif' 'Positif' 'Positif' 'Negatif' 'Positif'
 'Positif' 'Negatif' 'Negatif' 'Positif' 'Negatif' 'Positif' 'Negatif'
 'Negatif' 'Negatif' 'Positif' 'Positif' 'Positif' 'Positif' 'Positif'
 'Positif' 'Positif' 'Positif' 'Negatif' 'Positif' 'Positif' 'Positif'
 'Positif' 'Negatif' 'Positif' 'Positif' 'Positif' 'Positif' 'Negatif']
```

Figure 11. KNN Model Prediction Output Results

f. Model classification evaluation results

The classification results of the K-Nearest Neighbors model with Euclidean distance calculations are displayed via the Confusion matrix and classification report. The evaluation results of the model classification can be presented in detail, including precision, recall, f1-score, and other parameters. The following is an image of the source code for the KNN classification and confusion matrix along with the output results

```
# Laporan klasifikasi dan matriks konfusi
print("\nMatriks Konfusi:")
print(confusion_matrix(y_test, y_pred))

print("\nLaporan Klasifikasi:")
print(classification_report(y_test, y_pred))
```

Figure 12. Source Code Results of Classification and Confusion Matrix

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 4, **Number** 1, May 2024
https://doi.org/10.47709/brilliance.v4i1.4105

```
Confusion Matrix:
[[ 36  26]
 [ 10 235]]

Classification Report:
              precision    recall  f1-score   support

     Negatif       0.78      0.58      0.67        62
     Positif       0.90      0.96      0.93       245

    accuracy                           0.88       307
   macro avg       0.84      0.77      0.80       307
weighted avg       0.88      0.88      0.88       307
```

Figure 13. Output Results of Classification and Confusion Matrix Reports

Test result

The test results carried out by the system to evaluate the performance of the K-Nearest Neighbor method in measuring distance using Euclidean calculations have been achieved well. This test includes variations of parameter k with values 3, 5, 7, 9, and 11. The test results are represented in the form of a Confusion matrix and are accompanied by Performance Metrics such as precision, recall, f1 score and accuracy. The information available from this test is presented briefly in a table. Below are the results of the test table for each k:

a. Using the value k = 3

Test results with Confusion matrix:

Table 6. Confusion matrix results with k = 3

Actual	Prediction	
	Negative	Positif
Negative	34 (TN)	28 (FP)
Positif	19 (FN)	226 (TP)

Test results with Performance Metrics:

Table 7. Performance Metrics results with k = 3

Precision	226 / (226 + 28) = 0,89
Recall	226 / (226 + 19) = 0,92
F1 score	(2 * 0,89 * 0,92) / (0,92 + 0,89) = 0,91
Accuracy	(226 + 34) / (226 + 34 + 28 + 19) = 85%

b. Using the value k = 5

Test results with Confusion matrix:

Table 8. Confusion matrix results with k = 5

Actual	Prediction	
	Negative	Positive
Negative	36 (TN)	26 (FP)
Positive	10 (FN)	235 (TP)

Test results with Performance Metrics:

Table 9. Performance Metrics results with a value of k = 5

Precision	235 / (235 + 26) = 0,90
Recall	235 / (235 + 10) = 0,96
F1 score	(2 * 0,96 * 0,90) / (0,96 + 0,90) = 0,93
Accuracy	(235 + 36) / (235 + 36 + 26 + 10) = 88%

c. Using the value k = 7

Test results with Confusion matrix:

Table 10. Confusion matrix results with k = 7

Actual	Prediction	
	Negative	Positive
Negative	32 (TN)	30 (FP)
Positive	9 (FN)	236 (TP)

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 4, Number 1, May 2024
https://doi.org/10.47709/brilliance.v4i1.4105

Test results with Performance Metrics:

Table 11. Performance Metrics results, k = 7

Precision	236 / (236 + 30) = 0,89
Recall	236 / (236 + 9) = 0,96
F1 score	(2 * 0,96 * 0,89) / (0,96 + 0,89) = 0,92
Accuracy	(236 + 32) / (236 + 32 + 30 + 9) = 87%

d. Testing with a value of k = 9
Confusion matrix results:

Table 12. Confusion matrix results with k = 9

Actual	Prediction	
	Negative	Positive
Negative	30 (TN)	32 (FP)
Positive	8 (FN)	237 (TP)

Performance Metrics Results:

Table 13. Performance Metrics results with k = 9

Precision	237 / (237 + 32) = 0,88
Recall	237 / (237 + 8) = 0,97
F1 score	(2 * 0,97 * 0,88) / (0,97 + 0,88) = 0,92
Accuracy	(237 + 30) / (237 + 30 + 32 + 8) = 87%

e. Testing with a value of k = 11
Confusion matrix results:

Table 14. Confusion matrix results with a value of k = 11

Actual	Prediction	
	Negative	Positive
Negative	27 (TN)	35 (FP)
Positive	6 (FN)	239 (TP)

Performance Metrics Results:

Table 15. Performance Metrics results with a value of k = 11

Precision	239 / (239 + 35) = 0,87
Recall	239 / (239 + 6) = 0,98
F1 score	(2 * 0,98 * 0,87) / (0,98 + 0,87) = 0,92
Accuracy	(239 + 27) / (239 + 27 + 35 + 6) = 87%

DISCUSSION

In testing the performance of the K-Nearest Neighbors model in classifying responses to ChatGPT, a series of tests were carried out using a number of variations in k values, namely different numbers of neighbors. The assessment is carried out by paying attention to important metrics such as precision, recall, F1 score and accuracy. The following is a table of evaluation results for each k value tested.

Table 16. Evaluation Results for Each k value.

K	Precision	Recall	F1 score	Accuracy
3	0,89	0,92	0,91	85%
5	0,90	0,96	0,93	88%
7	0,89	0,96	0,92	87%
9	0,88	0,97	0,92	87%
11	0,87	0,98	0,92	87%

From the test results in the table above, the KNN model shows good ability in classifying responses. This can be seen from the accuracy of the model obtained, which ranges from 85% to 88%. It can be seen that the highest accuracy is achieved at a value of k = 5, namely 88%. Overall, the performance of the KNN model is relatively stable. The stability of KNN test results with Euclidean distance can be caused by several factors. Among them, the selection of methods and algorithms that are suitable for the dataset can contribute to the stability of the results. Choosing the right k v alue, good data quality, careful preprocessing steps, and determining data partitioning also play an important role. The following graph compares the accuracy of different k selections.

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 4, **Number** 1, May 2024
https://doi.org/10.47709/brilliance.v4i1.4105

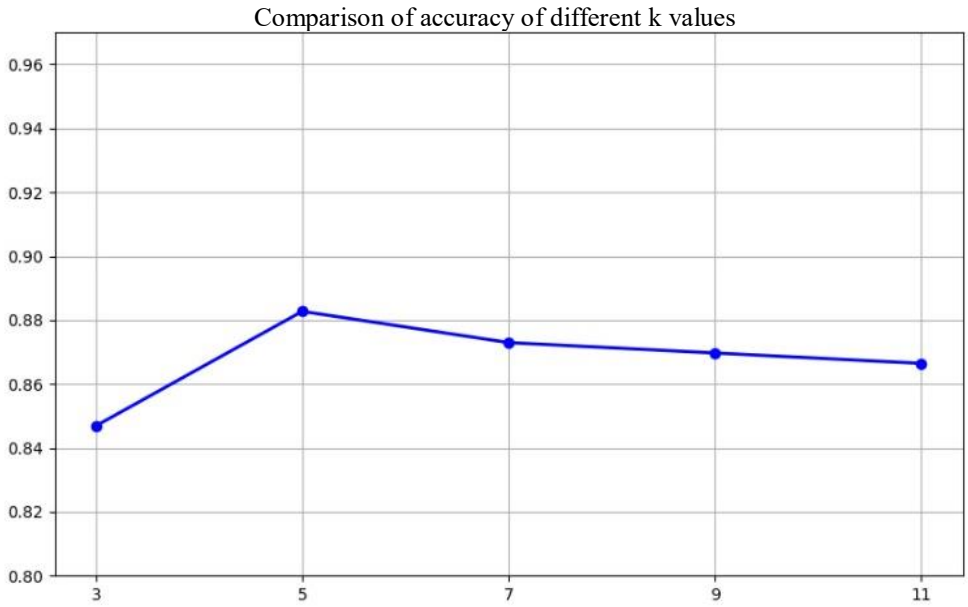

Figure 13. Graph of comparison results for different k values

CONCLUSION

Based on sentiment analysis on tweet data using textblob for the labeling process, the majority of public responses showed a positive response to the existence and performance of chatgpt. Data shows that 74.3% of the responses were positive, while the remaining 25.7% were negative. However, the labeling results using textblob still appear to be less accurate or unsatisfactory on this dataset. The K-Nearest Neighbors classification method has proven to be very successful in identifying and categorizing sentiment patterns in tweets on Twitter social media related to chatgpt. Proof of its success is that when using a k value of 5, the best accuracy obtained is 88%. The evaluation results also show that this classification has very satisfactory levels of precision, recall and f1-score.

REFERENCES

Anna Baita, Yoga Pristyanto, & Nuri Cahyono. (2021). Analisis Sentimen Mengenai Vaksin Sinovac Menggunakan Algoritma Support Vector Machine (SVM) dan K-Nearest Neighbor (KNN). *Information System Journal (INFOS)*, *4*(2), 42–46.

Diwandanu, M. T., & Wisudawati, L. M. (2023). ANALISIS SENTIMEN TERHADAP TWIT MAXIM PADA TWITTER MENGGUNAKAN R PROGRAMMING DAN K NEAREST NEIGHBORS. *Jurnal Ilmiah Informatika Komputer*, *28*(1), 1–16. https://doi.org/10.35760/ik.2023.v28i1.7909

Endang Sholihatin, Agatha Diani Putri Saka, Desta Rizky Andhika, Abdi Pranawa Satura Ardana, Chasetyo Ivan Yusaga, Rachmananta Ibnu Fajar, & Bagas Alif Virgano. (2023). Pemanfaatan Teknologi Chat GPT dalam Pembelajaran Bahasa Indonesia di Era Digital pada Mahasiswa Universitas Pembangunan Nasional Veteran Jawa Timur. *Jurnal TUAH*, *5*(1), 1–10.

Fransiska Vina Sari, & Arief Wibowo. (2019). ANALISIS SENTIMEN PELANGGAN TOKO ONLINE JD.ID MENGGUNAKAN METODE NAÏVE BAYES CLASSIFIER BERBASIS KONVERSI IKON EMOSI. *Jurnal SIMETRIS*, *10*(2), 681–686.

Hassani, N. (2019). The Flouting Maxim on Twitter Influencers' Tweets. *Journal of Pragmatics Research*, *1*(2), 139–155. https://doi.org/10.18326/jopr.v1i2.139-155

Irsalinda, N., Haswat, H., Sugiyarto, S., & Fitrianawati, M. (2021). Hidden Markov Model for Sentiment Analysis using Viterbi Algorithm. *EKSAKTA: Journal of Sciences and Data Analysis*, 18–23. https://doi.org/10.20885/EKSAKTA.vol2.iss1.art3

Irwansyah Suwahyu, Sofyan Taurid Ode Madi, Andi Ahmad Taufiq, & Andi Faiz Nabiel Rasyid. (2024). Analisis Pengaruh Kepercayaan dan Pemanfaatan Teknologi Terhadap Penggunaan Chat-GPT. *INTEC Journal: Information Technology Education Journal*, *3*(2), 105–109.

Lubis, R. K., Zein, A., & Salsabiela, I. (2024). Hubungan Empiris Chat GPT Pada Pembelajaran Mahasiaswa Bisnis Digital Di STMIK Pelita Nusantara Medan. *Jurnal Sains Dan Teknologi*, *5*(3), 900–903. https://doi.org/10.55338/saintek.v5i3.2857

Novianti, E. W., & Wibowo, W. (2022). Analisis Sentimen Pengguna Twitter terhadap Program Kartu Prakerja di Tengah Pandemi Covid-19 Menggunakan Metode Naïve Bayes Classifier. *Jurnal Sains Dan Seni ITS*, *11*(1).

E-ISSN : 2807-9035
Volume 4, **Number** 1, May 2024
https://doi.org/10.47709/brilliance.v4i1.4105

https://doi.org/10.12962/j23373520.v11i1.63552

Rinata, A. R., & Dewi, S. I. (2019). FANATISME PENGGEMAR KPOP DALAM BERMEDIA SOSIAL DI INSTAGRAM. *Interaksi: Jurnal Ilmu Komunikasi*, *8*(2), 13. https://doi.org/10.14710/interaksi.8.2.13-21

Rizqi Irawan, F. (2022). ANALISIS SENTIMEN TERHADAP PENGGUNA GOJEK MENGGUNAKAN METODE K-NEARSET NEIGHBORS. *JIKO (Jurnal Informatika Dan Komputer)*, *5*(1), 62–68. https://doi.org/10.33387/jiko.v5i1.4267

Sandova, F., Kurniawan, R., & Supratati, T. (2024). PENERAPAN DATA MINING MENGGUNAKAN METODE K-MEANS CLUSTERING PADA PENJUALAN TAS DI ASIA TOSERBA CIREBON. *JATI (Jurnal Mahasiswa Teknik Informatika)*, *8*(1), 245–251. https://doi.org/10.36040/jati.v8i1.8330

Septi Hasanah, Intan Purwasih, & Imam Santoso. (2023). Analisis Sentimen Terhadap Masyarakat Adanya Uang Kertas Baru Menggunakan Algoritma K-Nearest Neighbor(KNN). *Ikraith-Informatika*, *7*(2), 139–155.

Syaifuddin, A., Harianto, R. A., & Santoso, J. (2021). Analisis Trending Topik untuk Percakapan Media Sosial dengan Menggunakan Topic Modelling Berbasis Algoritme LDA. *Journal of Intelligent System and Computation*, *2*(1), 12–19. https://doi.org/10.52985/insyst.v2i1.150

Triyono, T., & Febriani, R. D. (2018). PENTINGNYA PEMANFAATAN TEKNOLOGI INFORMASI OLEH GURU BIMBINGAN DAN KONSELING. *Jurnal Wahana Konseling*, *1*(2), 74. https://doi.org/10.31851/juang.v1i2.2092