International Journal Software Engineering and Computer Science (IJSECS)
4 (2), 2024, 822-835
Published Online August 2024 in IJSECS (http://www.journal.lembagakita.org/index.php/ijsecs)
P-ISSN : 2776-4869, E-ISSN : 2776-3242. DOI: https://doi.org/10.35870/ijsecs.v4i2.2881.

RESEARCH ARTICLE Open Access

Sentiment Analysis of BMKG Weather Information Service Using K-Nearest Neighbor Method

Muhamad Fajar Sodiq *
Informatics Engineering Study Program, Universitas Stikubank, Semarang City, Central Java Province, Indonesia.
Corresponding Email: muhamadfajarsodiq@mhs.unisbank.ac.id.

Fatkhul Amin
Informatics Engineering Study Program, Universitas Stikubank, Semarang City, Central Java Province, Indonesia.
Email: fatkhulamin@edu.unisbank.ac.id.

Received: JUly 9, 2024; Accepted: August 10, 2024; Published: August 30, 2024.

Abstract: The Meteorology, Climatology, and Geophysics Agency (BMKG) is a government institution that provides information related to air quality, climate, dry days, satellite imagery, wave forecasts, wind forecasts, and fire potential. This information is disseminated not only through BMKG's official website but also via the social media platform X, making it easier for the public to access up-to-date information. This study aims to classify user sentiment towards the weather information services provided by BMKG using the K-Nearest Neighbor (KNN) method. Data was collected through a web crawling technique, resulting in 1,031 data points analyzed in this research. The data processing stages included Pre-Processing and sentiment calculation using Vader's Sentiment and Random Forest. The classification results using the KNN algorithm showed an accuracy rate of 96%, with an average precision of 96%, an average recall of 96%, and an average f-measure of 96%. These findings indicate that the KNN model can effectively classify user sentiment towards BMKG's services.

Keywords: BMKG; Platform X; K-Nearest Neighbor (KNN).

1. Introduction

The Meteorology, Climatology, and Geophysics Agency (BMKG) is a government institution responsible for providing information related to meteorological, climatological, and geophysical conditions in Indonesia. The services offered by BMKG include data on air quality, climate, dry days, satellite imagery, wave forecasts, wind forecasts, and the potential for forest fires [1]. With the advancement of technology and the growing demand for fast and accurate information, BMKG has expanded its distribution channels beyond its official website at https://www.bmkg.go.id/ by also utilizing various social media platforms, including X. The dissemination of information through social media aims to make it easier for the public to access relevant weather and environmental data. For instance, in 2024, BMKG released the annual rainfall predictions through the Climate Outlook 2024, providing crucial information for communities and stakeholders in planning activities dependent on weather conditions [2]. Accuracy in weather forecasting is critical, as precise predictions can assist in making informed decisions and mitigating the risks of natural disasters. Therefore, achieving high accuracy in weather predictions is a primary goal in various research and technological developments in meteorology [3].

One approach to evaluating public perception of the information services provided by BMKG is through sentiment analysis of social media users. Sentiment analysis, often referred to as opinion mining, is a branch of text mining research that aims to uncover the opinions, attitudes, or feelings of individuals towards a particular subject based on the text they produce. In this study, sentiment analysis is used to identify whether the responses of users to the weather information provided by BMKG are positive, negative, or neutral. The method used in this study is the K-Nearest Neighbor (KNN) algorithm, a popular technique in sentiment analysis due to its ability to compare a single data point with a set of data that has already been collected and studied, allowing for accurate predictions [4]. KNN is a supervised learning method that uses labeled data to predict the class of new data. In this case, KNN is employed to classify the sentiment of social media users towards BMKG's information services. Social media usage in Indonesia in 2023 reached 240 million people, reflecting the high internet penetration and the popularity of digital platforms among the population. According to the Digital 2023 Indonesia report, the number of X users in Indonesia increased by 5.6 million, equivalent to a 30.1% growth compared to the previous year. Among X users in Indonesia, 45.3% are female, and 54.7% are male [5][6]. These figures indicate that X is an effective medium for reaching various segments of the population, including the dissemination of weather information.

This study aims to identify the perspectives and reactions of users towards BMKG's weather information services distributed via X. The findings from this study are expected to provide valuable insights for BMKG to enhance the quality of its services, particularly in the communication of information through social media. By understanding how users respond to the information provided, BMKG can make necessary adjustments to ensure that the information is not only accurate but also well-received by the public. Additionally, this study is anticipated to contribute to the development of more effective communication strategies for weather information dissemination. Through sentiment analysis, BMKG can pinpoint areas requiring improvement and optimize its interactions with social media users. As a result, this study focuses on both the technical aspects of weather prediction and the human aspect, examining how the information is received and understood by the public.

2. Research Method

This study collected data from the social media platform X related to the weather information provided by BMKG, covering the period from October 2023 to March 2024. The data was gathered using a web crawling technique, which is essential for building a search engine index. This technique allows researchers to collect large volumes of data and ensures that the index is regularly updated to maintain the accuracy of search results. In this study, the crawling technique successfully retrieved 1,031 data points, which were then used as test data. The data processing began with the Pre-Processing stage, which involves several critical steps to clean the dataset of irrelevant or inconsistent elements. According to Suradi, these steps include data cleaning, case folding, normalization, tokenization, and stemming. Data cleaning aims to remove redundant or irrelevant data, while case folding standardizes the case of the letters in the dataset to lowercase. Normalization restores words to their correct forms in Indonesian, tokenization breaks sentences into individual words, and stemming reduces words to their root forms. Once the Pre-Processing stage was completed, the data was analyzed using the K-Nearest Neighbor (KNN) algorithm. KNN is a supervised learning algorithm used for classification based on the proximity of the test data features to the training data. In this study, KNN was employed to classify

user sentiment towards the information provided by BMKG. Before applying KNN, the pre-processed data was further processed using Vader's Sentiment analysis and the Random Forest method to produce a structured and comprehensible output.

The research methodology involves several systematically ordered steps. The problem formulation stage identifies and clearly defines the research problem, which in this case is classifying user sentiment towards BMKG's weather information disseminated through X. Conceptual modeling involves developing a conceptual model to describe the relationships between the variables being analyzed. This stage also includes preparing and installing the necessary libraries for data processing, such as emoji, Sastrawi, tweet-preprocessor, Pandas, os, regular expressions, NLTK, NumPy, Seaborn, sklearn, and Matplotlib. In the data collection or output phase, data collected from X over six months is used as input for the research. The output of this stage is a dataset ready for further analysis. The modeling phase involves processing and modeling the collected data using the KNN method, which includes further data cleaning, removing duplicate entries, and handling missing data. The simulation phase involves determining the sentiment of each processed data point through simulation, which includes automatically labeling the data by translating it from Indonesian to English, followed by applying the VaderSentiment technique for sentiment analysis. In the verification, validation, and experimentation phase, the analysis results are verified and validated to ensure accuracy, and experiments are conducted to test the effectiveness of the methods used. Finally, in the output analysis phase, the study's findings are analyzed and presented in the form of a classification report, including performance metrics such as accuracy, precision, recall, and f-measure. Through these steps, this study successfully identified user sentiment towards the weather information provided by BMKG, providing a strong foundation for developing more effective communication strategies.

3. Result and Discussion

3.1 Results

3.1.1. Problem Formulation

The initial stage of this research involves identifying and formulating the specific problem to be addressed. The focus of this study is the sentiment analysis of the weather service provided by BMKG, using data collected through a web crawling method. The data was sourced from the social media platform X, with the collection process conducted using a Python program executed in Google Colabs. This approach ensured that a substantial amount of relevant data could be retrieved efficiently for analysis. The gathered data served as the foundation for understanding how users perceive and respond to the weather information provided by BMKG.

Figure 1. Crawling Data View

3.1.2. Conceptual Model

The conceptual model phase of the research involves several critical steps aimed at preparing the data for analysis. These steps include the installation of necessary libraries and the reading of the dataset, which are essential to ensure that the modeling process aligns with the objectives of the study. The installation of libraries is a fundamental process that must be completed before proceeding to the data Pre-Processing stage. This stage involves setting up the environment with all the necessary tools for data manipulation and analysis. The libraries required for this research include emoji, Sastrawi, tweet-preprocessor, pandas, os, regular

expressions, nltk, numpy, seaborn, sklearn, and matplotlib. Each of these libraries serves a specific purpose in handling and analyzing the data. For instance, Sastrawi is used for stemming in the Indonesian language, while nltk provides tools for text processing. The successful installation of these libraries ensures that the data can be processed effectively, facilitating accurate sentiment analysis.

```
import os
import pandas as pd
import re
import string
from textblob import TextBlob
import preprocessor as p
from preprocessor.api import clean, tokenize, parse
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
import datetime
from datetime import timedelta
import numpy as np
import emoji
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.feature_extraction.text import CountVectorizer
from wordcloud import WordCloud, STOPWORDS, ImageColorGenerator
from Sastrawi.StopWordRemover.StopWordRemoverFactory import StopWordRemoverFactory
```

Figure 2. Install Python Library

After installing the necessary libraries, the next step is to read the dataset. This involves loading the data into the environment so that it can be manipulated and analyzed. The dataset used in this research is in CSV format, containing posts related to BMKG that were sourced from the social media platform X. The process of reading the dataset is crucial as it allows the researcher to access the raw data, which will then undergo various stages of processing. The structure and content of the dataset are examined to ensure that it is suitable for the intended analysis.

```
# buka akses ke google drive
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
# buka folder pada G Drive
import sys
sys.path.append('/content/drive/My Drive/Colab Notebooks/unisbank/ML/')
```

```
df = pd.read_csv('/content/drive/MyDrive/Colab Notebooks/unisbank/ML/dataset_bmkg.csv', sep=";")
df
```

Figure 3. Data Reading View

3.1.3. Collection of Input/Output Data

In this phase, data was collected over a six-month period, amounting to 1,013 entries that serve as the input for this research. The data collection spanned from October 2023 to March 2024, ensuring that the data is both recent and relevant. The primary goal during this period was to gather a comprehensive dataset that could provide insights into user interactions and sentiments regarding BMKG's weather services. The use of the social media platform X as a data source was strategic, given its widespread use and the active engagement of users on this platform. The data collected during this period is expected to reflect the public's real-time reactions and sentiments towards the weather information disseminated by BMKG.

```
Welcome to the Twitter Crawler 🐥

This script uses Chromium Browser to crawl data from Twitter with *your* Twitter auth token.
Please enter your Twitter auth token when prompted.

Note: Keep your access token secret! Don't share it with anyone else.
Note: This script only runs on your local device.

? What's your Twitter auth token? › 8788? What's your Twitter auth token? › *8788? What's your Twitter auth token? › **8788? What's your Twitter auth token? › ***8788? What's your Twitter auth token? › ****8788? What's your Twitter auth token? ›

up to date, audited 4 packages in 719ms

found 0 vulnerabilities
Installing dependencies...
Hit:1 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2204/x86_64  InRelease
Hit:2 https://deb.nodesource.com/node_18.x nodistro InRelease
Hit:3 http://archive.ubuntu.com/ubuntu jammy InRelease
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:5 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Hit:6 https://cloud.r-project.org/bin/linux/ubuntu jammy-cran40/ InRelease
Hit:7 http://archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:8 https://ppa.launchpadcontent.net/c2d4u.team/c2d4u4.0+/ubuntu jammy InRelease
Hit:9 https://ppa.launchpadcontent.net/deadsnakes/ppa/ubuntu jammy InRelease
```

Figure 4. Post X Crawling Data Results Display

3.1.4. Modeling Phase

The modeling phase is a critical part of the research, as it involves the application of various data processing techniques to derive meaningful insights. The first step in this phase is the Pre-Processing of data,

which is essential for cleaning the dataset and preparing it for analysis. Pre-Processing ensures that the data is free from inconsistencies and is structured in a way that is suitable for the subsequent analytical methods. Data cleaning is an integral part of the Pre-Processing stage. This process involves the removal of duplicate entries and any empty posts that may exist within the dataset. By eliminating redundant data, the researcher ensures that the analysis is based on unique and relevant entries, thereby improving the accuracy and reliability of the results. The cleaned dataset is then ready for further processing, which will enhance the quality of the sentiment analysis.

```
[ ] df = df.dropna()

[ ] df.shape

(964, 3)

[ ] def clean_twitter_text(text):
      text = re.sub(r'@[A-Za-z0-9_]+', '', text)
      text = re.sub(r'#\w+', '', text)
      text = re.sub(r'RT[\s]+', '', text)
      text = re.sub(r'https?://\S+', '', text)

      text = re.sub(r'[^A-Za-z0-9 ]', '', text)
      text = re.sub(r'\s+', ' ', text).strip()

      return text

    df['full_text'] = df['full_text'].apply(clean_twitter_text)

[ ] df['full_text'] = df['full_text'].str.lower()

[ ] df
```

Figure 5. Cleaning Data View

Case folding is another vital step in the Pre-Processing phase. This process standardizes all the text within the dataset to lowercase. If the dataset contains any capital letters, they are automatically converted to lowercase using a specific formula. The purpose of case folding is to eliminate variations in text that could affect the analysis, such as treating "BMKG" and "bmkg" as different entities. By ensuring uniformity in the text case, the analysis becomes more consistent and accurate.

	full_text	username	created_at
0	citra radar cuaca wilayah jabodetabek dan seki...	BPBDJakarta	Wed Feb 28 23:54:47 +0000 2024
1	prakiraan cuaca jawa tengah berlaku pukul wib ...	stageof_bji	Wed Feb 28 23:50:47 +0000 2024
2	selamat pagi sobat bmkg berikut prakiraan cuac...	BMKG_KERTAJATI	Wed Feb 28 23:49:24 +0000 2024
3	kamis amp jumat bmkg menjelaskan cuaca ekstrem...	TradeInves	Wed Feb 28 23:47:03 +0000 2024
4	bmkg wilayah berpotensi hujan lebat dan angin ...	TradeInves	Wed Feb 28 23:47:02 +0000 2024
...	...	...	...
1008	halo mitra maritim kalbar berikut kami sampaik...	bmkgmaritimpnk	Sun Feb 25 02:34:38 +0000 2024
1009	halo mitra maritim kalbar berikut kami sampaik...	bmkgmaritimpnk	Sun Feb 25 02:33:57 +0000 2024
1010	citra radar cuaca wilayah jabodetabek dan seki...	BPBDJakarta	Sun Feb 25 02:17:07 +0000 2024
1011	update peringatan dini cuaca wilayah bangka be...	infoBMKG	Sun Feb 25 02:04:51 +0000 2024
1012	update peringatan dini cuaca wilayah kalimanta...	infoBMKG	Sun Feb 25 02:01:58 +0000 2024

1013 rows × 3 columns

Figure 6. Case Folding View

Normalization is the process of converting words in the dataset to their correct forms in the Indonesian language. This step is particularly important for handling abbreviations and informal language often found in social media posts. During normalization, abbreviated words are expanded into their full forms, and any non-standard language is corrected to ensure that the data is accurate and suitable for analysis. This process is crucial for enhancing the quality of the sentiment analysis, as it ensures that the text is standardized and interpretable.

```
#Normalisasi
norm = {" yg ":" yang "," twt ":" tweet "," tp ":" tapi "," tgl ":" tanggal "," nggak ":" tidak "," jd ":" jadi "," ga ":" tidak "," hbs ":" habis "}

def normalisasi(str_text):
  for i in norm:
    str_text = str_text.replace(i, norm[i])
  return str_text

df['full_text'] = df['full_text'].apply(lambda x: normalisasi(x))
df
```

Figure 7. Normalization Display and Results

Tokenization involves breaking down sentences into individual words or tokens. This step is essential for analyzing the text at a granular level, as it allows the researcher to examine each word independently. The process includes stemming, which reduces words to their root forms, using functions from the Sastrawi library or based on the KBBI dictionary. By tokenizing the text, the researcher can better understand the components of the sentences and how they contribute to the overall sentiment.

```
citra radar cuaca wilayah jabodetabek dan sekitar kamis februari pukul wib sumber bmkg
prakira cuaca jawa tengah laku pukul wib tanggal februari
selamat pagi sobat bmkg ikut prakira cuaca bandara kertajati tanggal februari moga manfaat ya
```

Figure 8. Example of Preprocessing Results

3.1.5. Simulation Phase

In the simulation phase, the research employs automatic labeling to determine the sentiment of each post on X, categorizing them as positive, negative, or neutral. Since the posts on X are originally in Indonesian, they are first translated into English using the df.replace(translations, inplace=True) function. This translation is necessary because the subsequent analysis, particularly using VaderSentiment, is conducted in English. VaderSentiment is a tool designed to analyze sentiment in English text, which is why the posts must be translated before this step. The use of automatic labeling streamlines the process, allowing the researcher to efficiently categorize a large volume of data.

```
df.replace(translations, inplace=True)
df.head(10)
```

	full_text
0	weather radar image of the Jabodetabek area an...
1	Central Java weather forecast for February
2	Good morning, BMKG friends, follow the Kertaja...
3	Thursday & Friday, it is clear that the extrem...
4	bmkg areas with the potential for heavy rain a...
5	Good morning, join me in updating the weather ...
6	East Java weather forecast for February, good ...
7	BMKG issued a warning about extreme weather su...
8	update information, remember early weather in ...
9	BMKG warned early on that the weather forecast...

```
df = df.to_csv('translate_bmkg.csv', encoding='utf8', index=False)
```

Figure 9. Automatic Labeling With Translations

After the automatic labeling and translation, the data is processed using VaderSentiment. This tool assesses the sentiment of the text and classifies it into positive, negative, or neutral categories. The application of VaderSentiment provides a more structured and objective analysis of the text, as it converts the qualitative content into quantifiable sentiment scores. The results of this phase are crucial for understanding how users perceive BMKG's weather services on social media.

```
!pip install VaderSentiment==3.3.2
```

```
Requirement already satisfied: VaderSentiment==3.3.2 in /usr/local/lib/python3.10/dist-packages (3.3.2)
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-packages (from VaderSentiment==3.3.2) (2.31.0)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests->VaderSentiment==3.3.2) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests->VaderSentiment==3.3.2) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests->VaderSentiment==3.3.2) (2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests->VaderSentiment==3.3.2) (2024.6.2)
```

```
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer
analyzer = SentimentIntensityAnalyzer()
```

```
scores = [analyzer.polarity_scores(x) for x in df['full_text']]
print(scores)
df['sentiment_score'] = [x['compound'] for x in scores]
```

```
[{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0}, {'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0}, {'neg': 0.0, 'neu': 0.504, 'pos': 0.496, 'compound': 0.8957}, {'neg': 0.0, 'neu': 0.943, 'pos': 0.057, 'compound': 0.3818}, {'neg': 0.044,
```

```
df
```

Figure 10. Results of the Stages Using VaderSentiment

The results of the automatic labeling are implemented through Python programming, ensuring that the process is efficient and scalable. The data is then further analyzed to develop distribution models, which visually represent the sentiment data.

	full_text	sentiment_score	Sentiments
0	weather radar image of the jabodetabek area an...	0.0000	Netral
1	central java weather forecast for february	0.0000	Netral
2	good morning bmkg friends follow the kertajati...	0.8957	Positif
3	thursday friday it is clear that the extreme w...	0.3818	Positif
4	bmkg areas with the potential for heavy rain a...	0.6369	Positif
...	...	...	...
995	good afternoon bmkg samarinda friends join me ...	0.9186	Positif
996	good afternoon weather friends come with us fo...	0.7964	Positif
997	early warning update for the weather in the ja...	-0.3400	Negatif
998	good afternoon weather friends please remember...	0.9201	Positif
999	update remember early weather for the bengkulu...	0.0000	Netral

1000 rows × 3 columns

Figure 11. English Automatic Labeling Results

With the data obtained, the distribution of sentiment is developed into visual representations. As seen in Figure 20, the sentiment distribution within the BMKG dataset shows that there are 665 neutral posts, 181 positive posts, and 167 negative posts. This distribution provides a clear overview of the general sentiment towards BMKG's weather services, indicating areas where the service is well-received and areas that may require improvement.

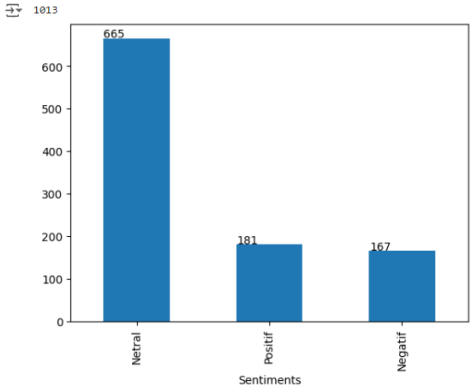

Figure 12. Distribution of Sentiment Spread Using Bar Chart

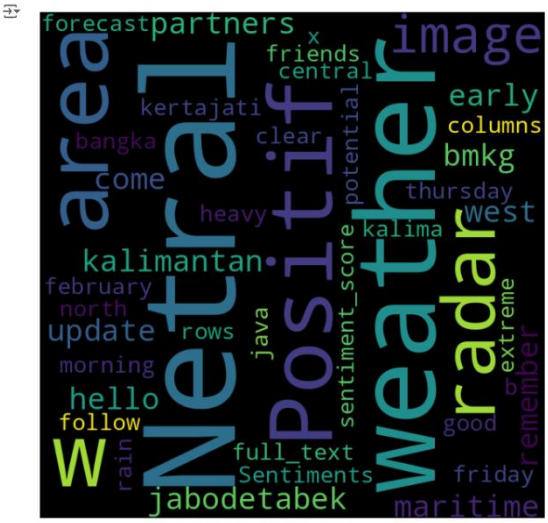

Figure 13. Sentiment Distribution Using Wordcloud

3.1.6. Conclusion Verification, Validation, and Experimentation

This section of the research involves several critical steps to verify and validate the conclusions drawn from the data analysis. The process includes feature extraction, data balancing, and the application of the K-Nearest Neighbor (KNN) algorithm to classify the data. Feature extraction in this research is conducted using the sklearn library and modules such as TfidfVectorizer and TfidfTransformer. The sklearn library is utilized for processing data in sentiment analysis, while feature_extraction.text is specifically used for extracting features from the text. TfidfVectorizer and TfidfTransformer calculate the frequency of words in the text and convert them into vector values. This process is essential for quantifying the text, allowing for more accurate sentiment analysis.

```
# Basic Operation
import pandas as pd
import numpy as np

from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.feature_extraction.text import TfidfTransformer
from nltk.corpus import stopwords
import re
from wordcloud import WordCloud,STOPWORDS
from nltk import SnowballStemmer


from sklearn.model_selection import train_test_split # Split Data


# Model Building
from sklearn.neighbors import KNeighborsClassifier

from sklearn.metrics import classification_report , confusion_matrix , accuracy_score # Performance Metrics


# Data Visualization
import matplotlib.pyplot as plt
from wordcloud import WordCloud
import seaborn as sns
import warnings


warnings.filterwarnings('ignore')
%matplotlib inline
```

Figure 14. TF-IDF stages

The explanation regarding Index 0 on the left side of the data, which has a count of 15, indicates that the first text of a document, while Index 1 with a count of 6, indicates the second text of a document, and Index 2 with a count of 4, indicates the third text of a document, and so on until Index 1012. The numbers in parentheses outside the data indicate the results of the TF-IDF weighting from the Python program.

```
'weather': 932, 'radar': 678, 'image': 404,
1013, 968)
  (0, 932)      0.017556732196616753
  (0, 678)      0.41107449095116017
  (0, 404)      0.46975086281362977
  (0, 596)      0.15058367948544282
  (0, 843)      0.04321343281256177
  (0, 442)      0.2698270076840857
  (0, 55)       0.18033470983263925
  (0, 41)       0.13134819583274085
  (0, 58)       0.34313581909528895
  (0, 862)      0.4474669693752854
  (0, 301)      0.02514683598959444
  (0, 65)       0.1431669448275974
  (0, 946)      0.07360282837077554
  (0, 779)      0.34313581909528895
  (0, 115)      0.025761449146715406
  (1, 932)      0.06819619728791149
  (1, 301)      0.09767868924056419
  (1, 156)      0.7167886087032642
  (1, 451)      0.5961347472798834
  (1, 319)      0.3282279863403273
  (1, 318)      0.09440336246911771
  (2, 932)      0.014434461882993079
  (2, 843)      0.03552840253987623
  (2, 301)      0.020674749805640217
  (2, 115)      0.021180060821944166
  :     :
```

```
  :     :
(1011, 115)   0.020587198607782478
(1011, 318)   0.019422179080300746
(1011, 264)   0.030215059932013674
(1011, 900)   0.030941336138902823
(1011, 708)   0.041496590371041205
(1011, 637)   0.062221434478966886
(1011, 698)   0.03461105273275665
(1011, 219)   0.06917806393317977
(1011, 320)   0.052126871890701296
(1011, 81)    0.6992071889248627
(1011, 103)   0.6992071889248627
(1012, 932)   0.025433124230019873
(1012, 301)   0.03642833965642742
(1012, 115)   0.037318683747993305
(1012, 318)   0.03520683763747432
(1012, 264)   0.05477123369292345
(1012, 900)   0.05608776405700817
(1012, 708)   0.07522141123617965
```

Figure 15. TF-IDF Calculation Results

The SMOTE (Synthetic Minority Over-sampling Technique) method is used in this research to address the issue of imbalanced data. SMOTE is a technique for generating synthetic samples from the minority class to create a more balanced dataset. By using SMOTE, the research ensures that the model is trained on a dataset that is more representative of all classes, which improves the accuracy of the sentiment classification. The imbalanced dataset is adjusted through the imblearn library, which facilitates the generation of synthetic samples and enhances the model's performance.


```
[ ]  # Handling imbalanced using SMOTE
     from imblearn.over_sampling import SMOTE # Handling Imbalanced
     smote = SMOTE()
     x_sm,y_sm = smote.fit_resample(x_tfidf,y)
```

Figure 16. SMOTE Method Display

In this stage, the dataset is split into training and test sets using a 90:10 ratio. This ratio is chosen because the dataset consists of 1,031 entries, and it is important that the model learns from a large portion of the data to achieve high accuracy. The splitting process ensures that the model is tested on data it has not seen during training, providing a realistic evaluation of its performance.

```
[ ]  from sklearn.model_selection import train_test_split
     x_train, x_test, y_train, y_test = train_test_split(x_sm, y_sm, test_size=0.1, random_state=0)
```

Figure 17. Display of Training and Test Data Sharing Script

The K-Nearest Neighbor (KNN) algorithm is applied in this stage to classify the sentiment data. The KNN architecture is imported through sklearn, and the dataset, which has been prepared and cleaned, is used to train the model. The command "classifier.fit(x_train, y_train)" is used to fit the model with the training data, allowing it to learn the patterns within the data.

```
[ ]  from sklearn.neighbors import KNeighborsClassifier
     classifier = KNeighborsClassifier(n_neighbors=5, metric='minkowski', p=2)
     classifier.fit(x_train, y_train)
```

```
▾ KNeighborsClassifier
KNeighborsClassifier()
```

Figure 18. K-Nearest Neighbor / KNN Classification Stage Display

The KNN algorithm then calculates the sentiment classification by comparing the test data with the training data, using different ratios such as 90:10, 80:20, 70:30, and 60:40. By testing the model with different ratios, the research identifies the best configuration for achieving accurate sentiment classification. The results are summarized in Table 2, which shows the accuracy of the model for each ratio. With the following formula:

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

Where
x: feature vector from test data
y: feature vector from training data
n: number of features

By comparing between trials at several ratios, the results are displayed in a table.

Table 1. Ratio Comparison Table

Training Data	Test Data	Accuracy
90%	10%	0.96
80%	20%	0.9448621553884712
70%	30%	0.9449081803005008
60%	40%	0.9360902255639098

After classification, the model is evaluated using a confusion matrix to ensure the effectiveness of the method applied. The confusion matrix provides a summary of the prediction results, including the number of correct and incorrect predictions for each class. The results are categorized as neutral, positive, or negative, and the matrix helps in identifying any areas where the model may need improvement.


```
[ ] from sklearn.metrics import confusion_matrix
    cm = confusion_matrix(y_test, y_pred)
    print(cm)

[[70  0  0]
 [ 2 63  6]
 [ 0  0 59]]
```

Figure 19. Confusion Matrix Results

3.1.7. Output Analysis Phase

The final phase of the research involves analyzing the output of the K-Nearest Neighbor (KNN) method. The analysis includes key performance metrics such as accuracy, precision, recall, and F1-score, which provide a comprehensive evaluation of the model's performance. These metrics help in determining how well the model has classified the sentiment data and provide insights into any areas that may require further refinement.

	precision	recall	f1-score	support
Negatif	0.97	1.00	0.99	70
Netral	1.00	0.89	0.94	71
Positif	0.91	1.00	0.95	59
accuracy			0.96	200
macro avg	0.96	0.96	0.96	200
weighted avg	0.96	0.96	0.96	200

Figure 20. K-Nearest Neighbor / KNN Classification Report Results

3.2 Discussion

The results of this study provide valuable insights into the sentiment expressed by users on the social media platform X regarding the weather services provided by BMKG. By using the K-Nearest Neighbor (KNN) algorithm and other data processing techniques, the research successfully categorized user sentiment into positive, negative, and neutral categories. This categorization allows for a more nuanced understanding of public perception, which is crucial for enhancing the effectiveness of BMKG's communication strategies. One of the key findings from the sentiment analysis is the distribution of sentiment among the dataset, as shown in Figure 12 and Figure 13. The data reveals that a significant portion of the sentiment is neutral, with 665 out of 1,013 posts falling into this category . This neutrality could indicate that while users are engaging with the weather information provided by BMKG, they may not have strong opinions or reactions to the content. Neutral sentiment could also reflect a level of satisfaction with the information, where users find it adequate but not remarkable. This suggests that while BMKG's weather services are being utilized, there may be opportunities to further engage the public and evoke more positive responses.

The presence of 181 positive posts is also noteworthy, as it indicates that a portion of the user base finds value in the weather services provided by BMKG. Positive sentiment is often associated with satisfaction, trust, and a high level of engagement with the content. For BMKG, this is an encouraging sign that their efforts in disseminating weather information are appreciated by the public. However, the relatively lower number of positive posts compared to neutral ones suggests that there is still room for improvement in making the content more appealing and engaging to users. This could involve enhancing the visual presentation of the information, making it more interactive, or providing more personalized weather updates. Conversely, the study also identified 167 negative posts within the dataset. Negative sentiment is a critical area of focus as it can indicate dissatisfaction, frustration, or confusion among users. The sources of negative sentiment can vary widely, ranging from the perceived inaccuracy of weather forecasts to the clarity and accessibility of the information provided. For BMKG, understanding the specific reasons behind the negative sentiment is essential for addressing user concerns and improving the overall quality of the service. By analyzing the content of these negative posts, BMKG can identify common themes or issues that need to be addressed, such as improving forecast accuracy or making the information more user-friendly.

The application of the K-Nearest Neighbor algorithm in this study has been proven to be effective in classifying sentiment with a high degree of accuracy. As shown in Table 1, the model achieved an accuracy rate of up to 96% when using a 90:10 ratio of training to test data. This high accuracy indicates that the KNN algorithm is well-suited for this type of sentiment analysis, providing reliable results that can be used to inform decision-making processes. The use of other performance metrics, such as precision, recall, and F1-score,

further supports the robustness of the model. These metrics ensure that the model not only classifies sentiment correctly but also minimizes errors, particularly in distinguishing between positive and negative sentiments.

The use of feature extraction techniques, including TfidfVectorizer and TfidfTransformer, played a significant role in enhancing the model's performance. By converting textual data into numerical vectors, these techniques allowed the algorithm to process and analyze the text more effectively. The success of this approach highlights the importance of proper data preparation in achieving accurate sentiment classification. The detailed explanation of the feature extraction process, as shown in Figure 14 and Figure 15, provides a clear understanding of how the textual data was transformed and used in the analysis. Moreover, the implementation of SMOTE (Synthetic Minority Over-sampling Technique) was crucial in addressing the issue of imbalanced data, as detailed in Figure 16. Imbalanced datasets are a common challenge in sentiment analysis, where one sentiment category may dominate the dataset, leading to biased results. By generating synthetic samples for the minority class, SMOTE helped to balance the dataset, ensuring that the model was trained on a more representative set of data. This, in turn, contributed to the overall accuracy and reliability of the sentiment classification. The confusion matrix, presented in Figure 19, provided further validation of the model's performance by showing the distribution of correct and incorrect classifications across the sentiment categories. The matrix revealed that the model was particularly effective at identifying neutral sentiment, which aligns with the overall distribution of the dataset. However, it also indicated areas where the model could be improved, particularly in differentiating between closely related sentiments, such as slightly positive versus neutral.

The final analysis, as summarized in Figure 20, underscores the effectiveness of the KNN algorithm in performing sentiment analysis on social media data. The high accuracy, precision, recall, and F1-score achieved by the model demonstrates its capability to provide meaningful insights into user sentiment. These findings are particularly valuable for BMKG as they seek to improve their weather services and better engage with the public. This study has demonstrated the importance of using advanced data analysis techniques, such as KNN and SMOTE, to gain a deeper understanding of public sentiment. The insights gained from this research can guide BMKG in refining their communication strategies, addressing user concerns, and ultimately enhancing the value of their weather services to the public. Future research could build on these findings by exploring other machine learning algorithms or incorporating additional data sources, such as user feedback surveys, to further enhance the accuracy and relevance of the sentiment analysis.

4. Related Work

The K-Nearest Neighbor (KNN) algorithm is an integral part of machine learning, particularly within the supervised learning category. Supervised learning utilizes labeled data, where both inputs and outputs are predefined, enabling the model to learn from existing data to make predictions about new, unseen data. The primary function of the KNN algorithm is to establish a relationship between input and output data, allowing it to predict outcomes for new inputs that were not part of the original dataset. In supervised learning, common examples include regression, where the model predicts the price of an item based on its features, and classification, where the model assigns a category to a given input [7]. KNN is a non-parametric algorithm, meaning it does not assume any specific distribution for the data, making it highly adaptable to various data types. Its instance-based nature, where predictions are made based on the nearest data points in the training set, further underscores KNN's utility in supervised learning tasks [4].

The application of the KNN method in this study follows a structured process designed to produce clear and interpretable results. The process begins with problem formulation, where the specific research problem—sentiment analysis of BMKG's weather services—is identified and defined. This study uses data collected through a web crawling technique to evaluate the efficiency and scalability of processing large volumes of web pages, ensuring that the search index is regularly updated for accuracy [8]. Following the problem formulation, the conceptual modeling phase primarily focuses on data pre-processing, organizing and preparing the data for subsequent analysis. The data collection phase involves gathering input data, which is then processed to generate output data that aligns with the conceptual model. This step is essential for ensuring that the data used in the analysis is both relevant and properly formatted. The modeling phase then determines the approach for analyzing sentiment data, with an emphasis on distinguishing between negative and positive sentiments.

The simulation phase involves testing and evaluating the data to classify the sentiment as negative or positive using the KNN algorithm. The objective of this phase is to simulate real-world conditions and assess the model's performance in classifying sentiments accurately. The final stage—conclusion (verification,

validation, and experimentation)—aims to validate the findings and ensure that the results are accurate and reliable. This stage includes thorough verification and validation processes to confirm that the KNN algorithm effectively analyzes and categorizes sentiment data. The output analysis phase provides the final results of the KNN calculations, offering a complete view of the sentiment analysis outcomes.

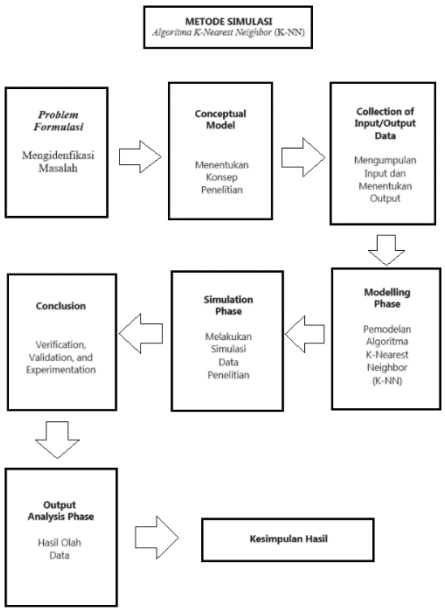

Figure 21. K-Nearest Neighbor (K-NN) Algorithm Research Flow View

The main objective of this study is to apply the KNN method for sentiment analysis of BMKG's weather services on the social media platform X, with the specific aim of identifying user opinions and reactions. This analysis is intended to assist BMKG in improving their services by making decisions informed by user feedback. The results of this research could enable BMKG to optimize their services, enhance interaction with social media users, and strengthen public engagement with weather information. This study builds upon previous research in the field of sentiment analysis using the KNN method, particularly within the context of weather forecasting. The first related study focused on predicting and calculating the accuracy of weather data in Indonesia. Although this study also dealt with weather data, it employed a different methodology. The findings were particularly useful for the community, helping them anticipate colder temperatures [9]. The second related study addressed a similar subject—weather forecast optimization—but utilized a different method. This research succeeded in providing optimal weather forecasts and improving classification accuracy [10]. The third related study differed in its subject matter but applied the same KNN method. It focused on classifying stroke disease and was able to predict test samples for the disease using the KNN algorithm [11].

The primary distinction between this study and previous research lies in the specific focus of the analysis. While earlier studies explored different methodologies or subjects, this research uniquely focuses on weather information specific to Central Java. Unlike prior studies that may have included multiple variables, this research narrows its scope to a single variable: weather information for Central Java. Despite these differences, all the studies, including this one, share a common approach—the application of the KNN algorithm for classification tasks. This study effectively demonstrates the use of the K-Nearest Neighbor (KNN) algorithm in sentiment analysis. By focusing on weather information for Central Java, it provides BMKG with practical insights that can enhance communication strategies and improve public engagement. The comparison with earlier studies highlights the adaptability and effectiveness of the KNN method across various domains, reaffirming its value in machine learning and sentiment analysis.

In this research, the implementation of the K-Nearest Neighbor (KNN) algorithm aligns with related works across multiple fields, including healthcare, finance, social media analysis, text categorization, and disaster management. KNN has consistently demonstrated reliable performance across these domains, highlighting its robustness and flexibility, making it an ideal choice for sentiment analysis of BMKG's weather services. For instance, KNN has been employed in research on credit risk classification, where the algorithm performed well in predicting credit risk [12]. Additionally, KNN has been applied to academic performance analysis, considering various cognitive and non-cognitive features, demonstrating its capacity to handle complex data [13]. The findings of this study indicate that KNN can achieve very high accuracy, with a confusion matrix score reaching

96%. This aligns with other studies that have shown KNN's effectiveness in managing classification and regression tasks, often outperforming other algorithms in areas such as cancer classification [14]. Furthermore, earlier research has noted that KNN can achieve strong results in data classification using various weighting and similarity measurement techniques . The average precision and recall scores of 96% each further demonstrate that the model is not only accurate but also capable of correctly identifying positive data, which is essential in sentiment analysis.

Consequently, this study offers valuable recommendations for BMKG to improve their interaction and communication with the public through social media. Future research could build upon these findings by exploring hybrid models or integrating additional data[15] sources to refine the analysis and expand its applicability. KNN, as a method that is both simple and effective, shows significant potential in a wide range of applications, including the sentiment analysis that is the focus of this research [16][17]. The related works reviewed in this study highlight the versatility and effectiveness of the K-Nearest Neighbor (KNN) algorithm across various domains. By successfully applying KNN to sentiment analysis of weather information for Central Java, this research not only aligns with but also builds upon previous studies, demonstrating the algorithm's capability to deliver accurate and reliable results in diverse applications. The findings of this study contribute to a broader understanding of KNN's utility, providing a solid foundation for future research and practical applications in sentiment analysis and beyond.

5. Conclusion

This study applied the K-Nearest Neighbor (KNN) algorithm to classify user sentiment towards the weather information services provided by BMKG. The algorithm was tested with various ratios of training and test data, yielding highly satisfactory performance. The analysis using the confusion matrix revealed that the KNN algorithm achieved an accuracy rate of 96%, indicating that the model possesses strong and accurate predictive capabilities. The average precision value of 96% suggests that the model successfully classified positive data with a high degree of correctness, with most predictions being accurate. Additionally, the average recall value of 96% demonstrates the model's effectiveness in identifying and classifying all positive data within the dataset. The average f-measure value of 96% reflects an optimal balance between precision and recall, indicating that the model provides consistent and balanced classification for both positive and negative data. These findings confirm that the KNN algorithm is an effective tool for sentiment analysis, particularly in evaluating the public's response to BMKG's weather services. The high level of accuracy, along with the balance between precision and recall, suggests that this model can be relied upon for accurate predictions. Therefore, the application of KNN in sentiment analysis offers a strong foundation for BMKG to enhance their interaction and communication with the public through social media, making them more responsive to user needs and perceptions. This study also paves the way for future research, such as exploring other algorithms or combining methods to further improve the accuracy and effectiveness of sentiment analysis.

References

[1] Hartanto, B., Astriawati, N., Supartini, D., & Yekti, D. K. (2022). *Pencarian dan Pemanfaatan Informasi Data Badan Meteorologi, Klimatologi, dan Geofisika (BMKG)*. Yogyakarta: Sekolah Tinggi Maritim Yogyakarta. https://doi.org/10.55123/insologi.v1i5.906

[2] Prasetyaningtyas, K. (2024). *Climate Outlook 2024*. Jakarta: BMKG.

[3] Mubarok, H., & Dipalokareswara, A. (2022). *Optimization of Weather Forecast Using Ensemble Method on Naïve Bayes and C4.5*. Tasikmalaya: Universitas Siliwangi. https://doi.org/10.28932/jutisi.v8i3.5455

[4] Daqiqi, I. (2021). *Machine Learning: Teori, Studi Kasus dan Implementasi Menggunakan Python*. Riau: UR Press.

[5] Kemp, S. (2023). *Digital 2023: Indonesia*. Retrieved from https://datareportal.com/reports/digital-2023-indonesia

[6] Suradi, S. (2019). *Pemodelan Sistem*. Makassar: Tohar Media.

[7] Dinata, R. K., & Hasdyna, N. (2020). *Machine Learning*. Sulawesi: Unimal Press.

[8] Alafandy, K. A., Omara, H., Lazaar, M., & Al Achhab, M. (2022). *Machine Learning*. Maroco.

[9] Rangkuti, M. Y. R., Alfansyuri, M. V., & Gunawan, W. (2021). *Penerapan Algoritma K-Nearest Neighbor (KNN) Dalam Memprediksi dan Menghitung Tingkat Akurasi Data Cuaca di Indonesia*. Sumbawa: Universitas Teknologi Sumbawa. https://doi.org/10.36761/hexagon.v2i2.1082

[10] Yani, V. I., & Mubarok, H. (2022). *Optimasi Prakiraan Cuaca Menggunakan Metode Ensemble Pada Naïve Bayes dan C4.5*. Tasikmalaya: Universitas Siliwangi. https://doi.org/10.28932/jutisi.v8i3.5455

[11] Zuriati, Z., & Qomariyah, N. (2023). *Klasifikasi Penyakit Stroke Menggunakan Algoritma K-Nearest Neighbor (KNN)*. Lampung: Politeknik Negeri Lampung. https://doi.org/10.25181/rt.v1i1.2665

[12] Jasmir, J., Sika, X., Mulyadi, M., & Amelia, R. (2022). Klasifikasi kelayakan pemberian kredit pada calon debitur menggunakan naïve bayes. *Jurikom (Jurnal Riset Komputer)*, 9(6), 1833. https://doi.org/10.30865/jurikom.v9i6.5131

[13] Owusu-Boadu, B., Nti, I., Nyarko-Boateng, O., Aning, J., & Boafo, V. (2021). Academic performance modelling with machine learning based on cognitive and non-cognitive features. *Applied Computer Systems*, 26(2), 122-131. https://doi.org/10.2478/acss-2021-0015

[14] Desiani, A., Lestari, A., Al-Ariq, M., Amran, A., & Andriani, Y. (2022). Comparison of support vector machine and k-nearest neighbors in breast cancer classification. *Pattimura International Journal of Mathematics (Pijmath)*, 1(1), 33-42. https://doi.org/10.30598/pijmathvol1iss1pp33-42

[15] Narulita, L. (2019). Analisa sentimen pada tinjauan buku dengan algoritma k-nearest neighbour. *Konvergensi*, 13(2). https://doi.org/10.30996/konv.v13i2.2758

[16] Triguero, I., García-Gil, D., Maillo, J., Luengo, J., García, S., & Herrera, F. (2018). Transforming big data into smart data: an insight on the use of the k-nearest neighbors algorithm to obtain quality data. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 9(2). https://doi.org/10.1002/widm.1289

[17] Sun, M., & Yang, R. (2020). An efficient secure k-nearest neighbor classification protocol with high-dimensional features. *International Journal of Intelligent Systems*, 35(11), 1791-1813. https://doi.org/10.1002/int.22272.

