International Journal of Information Engineering and Science

E-ISSN: 3048-1953
P-ISSN: 3048-1902

Research Article

Sentiment Analysis of the Performance of the Legal System in Indonesia Based on Twitter Comments Using the Naïve Bayes Algorithm

Rasiban [1], Dadang Iskandar Mulyana [2], Muhammad Joko Umbaran Kharis Bahrudin [3], Nicola Marthy [4]

[1] Sekolah Tinggi Ilmu Komputer Cipta Karya Informatika Jakarta, Indonesia; email: rasiban@stikomcki.ac.id
[2] Sekolah Tinggi Ilmu Komputer Cipta Karya Informatika Jakarta, Indonesia; email: dadang@stikomcki.ac.id
[3] Sekolah Tinggi Ilmu Komputer Cipta Karya Informatika Jakarta, Indonesia; email: jokoumbaran@stikomcki.ac.id
[4] Sekolah Tinggi Ilmu Komputer Cipta Karya Informatika Jakarta, Indonesia; email: nicolamarthysia2@gmail.com

* Corresponding Author: rasiban@stikomcki.ac.id

Abstract: The development of social media, especially TWITTER, has become one of the main means for people to express opinions and criticism on various issues, including the performance of law in Indonesia. This study aims to analyze public sentiment towards the performance of law based on TWITTER user comments using the Naïve Bayes algorithm. The research data consists of 1004 comments collected from several videos related to legal topics. The analysis process includes the stages of data crawling, pre- processing (text cleaning, normalization, and tokenization), labeling sentiment into positive, negative, and neutral, and testing the Naïve Bayes model. The results show that the Naïve Bayes algorithm is able to classify sentiment with an accuracy level of 93.73%. The distribution of sentiment from 1004 comments shows that the majority of public opinion is (negative/positive/neutral), which indicates that public perception of the performance of law is still (critical/positive). These findings are expected to be input for related parties to understand public opinion and improve the quality of legal performance in Indonesia

Keywords: Legal Performance; Naïve Bayes; Python; Sentiment Analysis; Twitter.

Received: April, 30 2025
Revised: May, 15 2025
Accepted: May, 29 2025
Published: May, 31 2025
Curr. Ver.:May, 31 2025

1. Introduction

The development of information and communication technology has brought significant changes in the way people interact and express their opinions [1]. Social media, particularly Twitter [2], has become one of the most widely used platforms for expressing opinions, criticism, and support regarding various issues, including those related to legal performance in Indonesia. The rapid dissemination of information and ease of access have made Twitter a dynamic public space where public perceptions can be formed and spread quickly on a large scale [3].

On the one hand, social media provides opportunities for individuals to freely express their views [4]. On the other hand, the overwhelming volume of opinions that emerge on these platforms requires proper analysis in order to be understood objectively. Public perception of legal performance is important to examine because the opinions that develop can influence the level of public trust in legal institutions and the enforcement of justice [5].

Sentiment analysis is one of the most effective methods for understanding trends in public opinion on social media [6]. By utilizing machine learning algorithms such as Naïve Bayes, comment data can be classified into Pro and Contra sentiments. This method enables researchers to identify patterns of public perception quantitatively and accurately [7].

Considering the important role of public opinion in shaping the image and legitimacy of legal performance, research on sentiment analysis on the Twitter platform is highly relevant [8]. Through this study, it is expected that relevant stakeholders will gain a clearer understanding of public perceptions and use the findings as evaluation material in efforts to improve the quality of legal performance in Indonesia [9].

2. Literature Review

Sentiment Analysis

Sentiment analysis is a method used to extract opinion data, understand, and automatically process textual data in order to identify positive and negative sentiments contained within it [10]. Sentiment analysis is also defined as a field of study that analyzes attitudes, opinions, and emotions toward entities and their attributes as expressed in written text [11]. Sentiment analysis focuses on opinions that express or imply positive or negative sentiments, commonly referred to as positive opinions and negative opinions in everyday language. Sentiment analysis is one of the research topics that can detect issues currently circulating on social media, such as the issue of church dissolution [12].

Sentiment analysis is very important for businesses and organizations because they seek to understand consumer or public opinions regarding their services [13]. In addition, sentiment analysis can also be utilized by governments to determine public opinions about policies [14]. Sentiment analysis, also known as opinion mining, is used to analyze or classify users' opinions based on words, sentences, or documents [15]. In this context, sentiment analysis can be considered an important subfield of semantic analysis because it aims to identify the topics being discussed and the sentiments expressed toward those topics [16].

Twitter

Twitter is a social networking service, or more specifically an online microblogging platform, that allows users to send, read, and reply to text messages of up to 280 characters, known as tweets [17]. Initially, Twitter only allowed users to post tweets with a maximum of 140 characters. However, on November 7, 2017, Twitter increased the limit to 280 characters.

On Twitter, unregistered users can only read other users' tweets, while registered users can write, share, and like tweets through the website user interface and smartphone applications on Android and iOS (iPhone) devices [18]. A tweet refers to a post written and shared by a user. Meanwhile, a retweet refers to sharing another user's post on one's own profile timeline.

Machine Learning

Machine Learning, also known as machine-based learning, is a branch of computer science that enables systems to operate without being explicitly programmed. Machine learning is a field of artificial intelligence that studies how to create knowledge from data [19].

Machine Learning is a fast and powerful technique for discovering new problems and insights in research [20]. By definition, Machine Learning is a field of study that focuses on algorithms and statistical models used by computers to perform specific tasks without explicit instructions [21]. More generally, it refers to how a computer can learn from its surrounding environment, thereby developing and expanding its own "knowledge."

Stemming Algorithm – Sastrawi Library

The Sastrawi Stemmer is a stemming algorithm specifically designed for the Indonesian language. Its function is to transform affixed words into their root forms. For example, the word "membersihkan" will be converted into "bersih." The purpose is to reduce word variations and facilitate text analysis. The algorithm consists of the following stages: **a. Affix Identification:** The algorithm searches for and identifies affixes (prefixes, infixes, and suffixes) attached to a word; **b. Affix Removal:** The identified affixes are removed from the word; **c. Dictionary Check:** After the affixes are removed, the resulting word is checked against a root-word dictionary; **d. Return Root Word:** If the word exists in the dictionary, it is considered the root word. If not, the algorithm may apply additional affix-removal rules or return the original word.

The Sastrawi Stemmer utilizes a series of complex rules to handle various types of affixes in the Indonesian language while also considering exceptions and unique morphological characteristics of words.

Random Undersampling

Random Undersampling is a technique used to address class imbalance in a dataset. This situation commonly occurs when the number of instances in one class is significantly larger than in another class.

The main idea is to randomly reduce the number of samples from the majority class (the class with the largest amount of data) until it becomes balanced with the minority class (the class with the fewest data points), or balanced across all classes if more than two classes are involved.

One advantage of random undersampling is that it is simple and easy to implement. However, its drawback is that some data from the majority class are discarded, which may result in the loss of important information.

In your notebook code, this technique is implemented by randomly selecting half of the data from each sentiment class (Positive, Neutral, and Negative) to be used as the training dataset (train_set).

3. Materials and Method

Literature Review

The author conducted a literature review by collecting theoretical references related to this research. The sources used in this study include journals, reference books, undergraduate theses, and relevant websites related to Natural Language Processing (NLP), Data Mining, Machine Learning, TextBlob, Naïve Bayes, and Naïve Bayes Classifier.

Observation

At this stage, the author conducted data observation by collecting data using the tweet-harvest tool.

a. Prepare a Twitter authentication token so that tweet-harvest can access the data.
b. Ensure that all requirements for tweet-harvest (such as Node.js) have been installed.
c. Once everything is ready, run tweet-harvest by specifying the search keyword, namely "Indonesia_gelap" in Indonesian, for the period from January to August 2025.
d. Tweet-harvest then collects the relevant tweets and stores them in CSV format in the /content/tweet-data/ folder.
e. Load the CSV file into a pandas DataFrame in Python for further viewing and processing, and check the total number of tweets successfully collected.

Preprocessing

At this stage, the author performs five preprocessing steps before proceeding to the next stage in order to facilitate data analysis. The five stages are:

a. Cleaning Data
b. Normalization
c. Stopword Removal
d. Stemming
e. Translation

Naïve Bayes Classifier

The Naïve Bayes Classifier process is used to classify the sentiment of tweets that have been translated into English. After being trained on a portion of the data that already contains sentiment labels (Positive, Negative, Neutral), the classifier is used to predict the sentiment of other tweets. The classification results are then stored in a new column called classification_bayes.

TextBlob

TextBlob is a Python library that provides a simple API for common Natural Language Processing (NLP) tasks. Built on top of NLTK and Pattern, TextBlob makes it easier to perform various text-processing operations, including:

a. Sentiment Analysis
b. Tokenization
c. Part-of-Speech Tagging
d. Phrase Extraction
e. Classification
f. Translation and Language Detection
g. Stemming and Lemmatization

Naïve Bayes Evaluation

The results obtained from the previous Naïve Bayes classification model require a deeper understanding of its performance. Several evaluation aspects include:

a. Evaluation Metrics: In addition to accuracy, it is important to examine other evaluation metrics.
b. Confusion Matrix: A visualization that shows the number of correct and incorrect predictions for each sentiment class.
c. Comparison with Other Algorithms: Comparing the performance of Naïve Bayes with other classification algorithms, such as Support Vector Machine (SVM), Logistic Regression, or deep learning-based models.
d. Cross-Validation: Used to obtain a more robust estimate of model performance.

Research Flow

This research flow illustrates the sequence of research stages from beginning to end, developed based on the author's conceptual framework and implemented into a structured process.

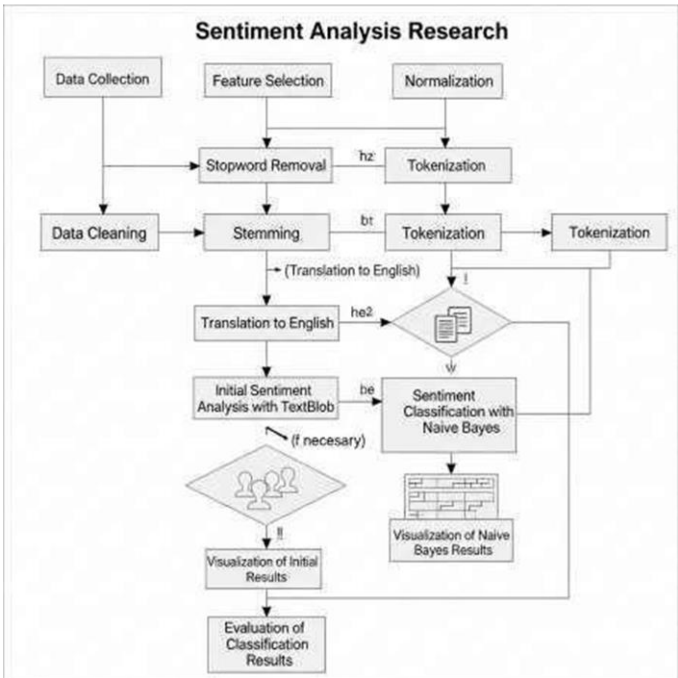

Figure 1. Sentiment Analysis Research

4. Results and Discussion

Data Collection

The initial stage of data collection in this study was conducted by loading Twitter data using a Twitter authentication token.

a. The code sets the author's Twitter authentication token, as some crawling tools require this token to access Twitter data.
b. Dependency Installation: The code then installs the required packages, including the Python package Pandas for data analysis and Node.js. Node.js is required because the tweet-harvest crawling tool is built using Node.js.
c. Tweet Crawling with tweet-harvest: The main data collection process is carried out using the command "npx -y tweet-harvest@2.6.1". This command runs the tweet-harvest tool to search for tweets based on specified criteria.
    1) Output File Name: The collected tweet data is stored in a CSV file named Indonesia_gelap.csv.
    2) Search Keyword: Tweets are searched using the keyword "Indonesia gelap since:2025-01-01 until:2025-08-30 lang". This means searching for tweets containing the phrase "Indonesia gelap" in Indonesian language (lang) from January 1, 2025, to August 30, 2025.
    3) Latest Tab: The search focuses on the most recent tweets (--tab "LATEST").
    4) Tweet Limit: The tool attempts to collect up to 1,000 tweets (-l 1000).
    5) Authentication Token: The provided Twitter authentication token is used during the crawling process.
d. Data Storage: tweet-harvest stores the successfully collected tweets in the specified CSV file within the /content/tweets-data/ directory. The output from code cell 0OXbiLB-sfBX indicates the storage location and the total number of collected tweets (970 tweets).
e. Loading Data into a Pandas DataFrame: After collection, the code reads the CSV file using the pandas library and stores it in a DataFrame. This allows the tweet data to be easily viewed and processed in Python.
f. Data Verification: Finally, the code prints the number of rows in the DataFrame to confirm how many tweets were successfully loaded.

The following figure shows the data crawling process:

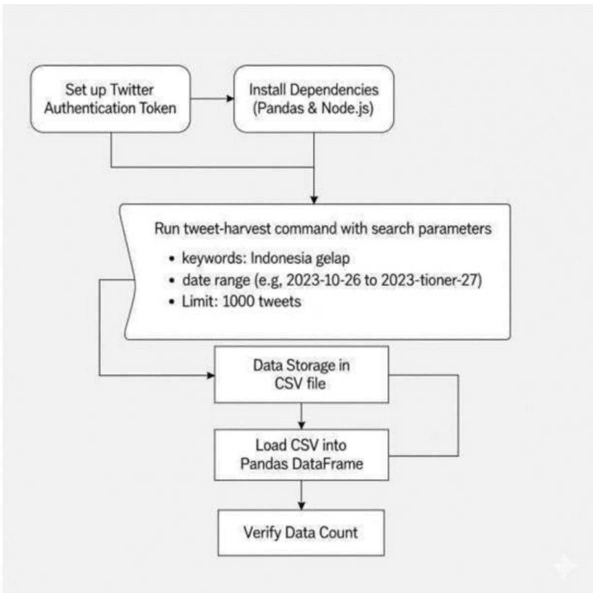

Figure 2. Prosess Crawling

Preprocessing

The data preprocessing stage involves several steps to clean and prepare text data for analysis. Based on the code, these steps include:

a. Selecting relevant columns: Only the full_text, username, and created_at columns are selected.
b. Removing duplicates: Rows containing identical tweet texts are removed.
c. Handling missing values: Rows containing empty values (NaN) are removed.
d. Text cleaning: Unwanted characters such as mentions (@), hashtags (#), RT markers, links (http or https), non-alphanumeric characters, and excessive spaces are removed.
e. Lowercasing: All text is converted to lowercase for standardization.
f. Normalization: Informal words or abbreviations such as "yg", "tp", "sdh", "nggak", "gak", "banget", "jg", "tuk", and "msh" are replaced with their standard forms.
g. Stopword Removal: Common Indonesian words that do not contribute significant meaning to sentiment analysis (including the word "tidak" which was manually added) are removed.
h. Stemming: Words are converted into their root forms using Sastrawi.

All of these steps aim to reduce noise in the text data and make it easier to process using sentiment analysis algorithms.
Data Collection, Load tweet data from a CSV file.
Load data from the CSV file /content/prabowo.csv using pandas. [DONE]

Data Preprocessing, Perform a series of steps to clean and prepare text data, including:

a. Selecting relevant columns.
b. Removing duplicates.
c. Handling missing values.
d. Cleaning text from unwanted characters (mentions, hashtags, links, etc.).
e. Converting text to lowercase (lowercasing).
f. Normalizing informal words.
g. Removing stopwords.
h. Performing stemming to obtain root words. [DONE]

Data Translation: Translate tweet texts from Indonesian into English.

Initial Sentiment Analysis (Using TextBlob): Perform initial sentiment analysis on the translated text using TextBlob to obtain preliminary sentiment labels (Positive, Negative, Neutral).

Data Labeling: Add sentiment labels generated from the initial analysis into the dataset.

Data Visualization: Create visualizations such as word clouds and bar charts to understand the distribution of words and sentiments within the data.

a. Sentiment Classification (Using Naïve Bayes)
b. Prepare the dataset for classification.
c. Split the data into training and testing sets (randomly selecting a portion of data from each class).
d. Train a Naïve Bayes Classifier using the training set.
e. Evaluate classifier accuracy.
f. Reclassify sentiment across the entire dataset using the trained Naïve Bayes Classifier.
g. Add the Naïve Bayes classification results to the dataset.
h. Create visualizations showing sentiment distribution based on Naïve Bayes classification results.

Result Evaluation: Compare the classification results obtained from the initial analysis (TextBlob) with those generated by the Naïve Bayes Classifier. Finish Task: Present the overall research workflow and analysis results.

The preprocessing stages are illustrated in:

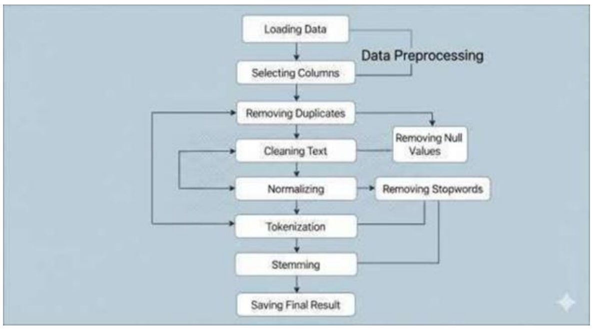

Figure 3. Preprocessing Stages

The following are examples of each process performed during the preprocessing stage.

Cleaning Data

Data cleaning is an important process in data analysis that aims to clean, improve, and correct inaccurate, incomplete, or irrelevant data within a dataset.

The goal is to ensure that the data being analyzed has good quality and consistency. Clean data produces more accurate and reliable analytical models or conclusions. The following script is used for data cleaning: import pandas as pd import re, import seaborn as sns import matplotlib.pyplot as plt

This cell imports the required libraries:

a. pandas: For working with tabular data (DataFrames).
b. re: For regular expression operations used in text cleaning.
c. seaborn and matplotlib.pyplot: For data visualization (although they are primarily used in the visualization section, they are imported at the beginning).

Cell ouiUtnP4UKJv:
df = pd.read_csv("/content/prabowo.csv")df.head().
This cell reads data from a CSV file named prabowo.csv located in /content/ and stores it in the DataFrame df. The command df.head() displays the first five rows to examine the initial data structure.

Cell CzMOpz7kUVC0:
df = df[['full_text','username','created_at']] df
This cell selects only three columns from the original DataFrame: full_text, username, and created_at. Other columns are ignored. The result is reassigned to df.

Cell RwuKGEdYUyNi:
df.shape
This cell displays the number of rows and columns in the DataFrame after column selection. This is useful for understanding the dataset dimensions.

Cell HpZAMD52U6Dg:
df = df.drop_duplicates(subset=['full_text'])
This cell removes duplicate rows based on the full_text column. The parameter subset=['full_text'] ensures that duplication is checked only within that column.

Cell s0VvTbrXVCM4:
df.duplicated().sum()
This cell checks whether duplicate rows still exist after the previous step. The .sum() function counts the number of duplicate rows remaining.

Cell BtiJt8zCVIn6:
df = df.dropna()
This cell removes rows containing null values (NaN) in any column.

Cell -tiblpDCVPbr:
df.isnull().sum()
This cell calculates the number of null values in each column after the removal process.

Cell zQo1NVY_VVfW:
df.shape
This cell displays the dataset dimensions again after removing duplicates and null values.
The following function is used for tweet text cleaning: 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)
This function uses regular expressions (re.sub) to remove specific elements from tweet text. The function is then applied to every value in the full_text column using the .apply() method.

Normalization

Normalization in text preprocessing is the process of converting non-standard words or phrases into their standard or commonly accepted forms. The purpose is to standardize different writing variations that share the same meaning, allowing text analysis models to recognize them as the same entity regardless of how they are written.

In your script (cell c7ixU0EtocUS), normalization is performed by replacing Indonesian abbreviations or non-standard forms with their standard equivalents. Examples include:

"yg" → "yang"
"tp" → "tetapi"
"sdh" → "sudah"
"nggak" and "gak" → "tidak"
"banget" → "sangat"
"jg" → "juga"
"tuk" → "untuk"
"msh" → "masih"

By performing normalization, words such as "yg" and "yang" are treated as the same term in subsequent analyses, which helps improve the accuracy of sentiment analysis results and other natural language processing tasks.

As explained previously, two methods were used for sentiment analysis:

TextBlob

TextBlob is a Python library that provides a simple API for common Natural Language Processing (NLP) tasks, including sentiment analysis. By default, TextBlob uses a lexicon-based approach to determine the polarity (positive/negative) and subjectivity (objective/subjective) of a text.

Naïve Bayes Classifier

The Naïve Bayes Classifier is a machine learning classification algorithm based on Bayes' Theorem with the assumption of independence among features. In this study, the model was trained using a portion of the data that had already been assigned sentiment labels by TextBlob and was subsequently used to classify the entire dataset.

Sentiment Analysis Results Using TextBlob (Cell DfG6JNwCZy66)

a. total_positive = 887: Indicates that 887 tweets were classified as having positive sentiment by TextBlob.
b. total_neutral = 4: Indicates that 4 tweets were classified as having neutral sentiment by TextBlob.
c. total_negative = 113: Indicates that 113 tweets were classified as having negative sentiment by TextBlob.
d. Total Data = 1004: Indicates the total number of tweets analyzed.

These results represent the initial sentiment analysis using TextBlob's lexicon-based approach.

Sentiment Analysis Results Using Naïve Bayes Classifier (Cell DfG6JNwCZy66 After Reclassification)

After training and applying the Naïve Bayes Classifier, the entire dataset was reclassified. The results are as follows:

a. total_positive = 828: Indicates that 828 tweets were classified as having positive sentiment by the Naïve Bayes Classifier.
b. total_neutral = 0: Indicates that no tweets were classified as having neutral sentiment by the Naïve Bayes Classifier.
c. total_negative = 176: Indicates that 176 tweets were classified as having negative sentiment by the Naïve Bayes Classifier.
d. Total Data = 1004: Remains the total number of tweets analyzed.

The differences between the initial TextBlob results and the Naïve Bayes Classifier results demonstrate how a trained machine learning model can produce classifications that differ slightly from a purely lexicon-based approach.

Visualization of Naïve Bayes Classification Results (Cell sCWAy9RTaIfu)

The bar chart displayed in cell sCWAy9RTaIfu visualizes the results of the Naïve Bayes classification:

a. The blue bar represents the number of Positive tweets (828).
b. The red bar represents the number of Negative tweets (176).
c. The yellow bar represents the number of Neutral tweets (0).

This visualization makes it easier to observe the overall sentiment distribution after applying the Naïve Bayes Classifier.

Summary of Results

Based on the sentiment analysis conducted using the Naïve Bayes Classifier, the majority of tweets in the dataset were classified as Positive (828 tweets), followed by Negative (176 tweets), while no tweets were classified as Neutral.

A comparison between the initial TextBlob classification and the Naïve Bayes Classifier results can also be observed in the data DataFrame (displayed in cell 77Al2bNZaOgQ) through the classification (TextBlob) and classification_bayes (Naïve Bayes) columns. Cell cMEZKrC4af_J displays examples of tweets for which both classifiers produced the same sentiment classification result.

5. Conclusion

Based on the sentiment analysis that has been conducted, several conclusions can be drawn. From the 1,004 tweets analyzed, the majority of tweets (828 tweets based on the Naïve Bayes classification results) expressed positive sentiment regarding the topic "Indonesia Gelap". This may indicate that although the topic being discussed is described as "dark," which may imply negative circumstances, many users expressed positive views, expectations, or comments regarding the situation. At the same time, a significant number of tweets (176 tweets based on the Naïve Bayes classification results) expressed negative sentiment. These tweets likely contained criticism, complaints, disappointment, or other negative perspectives related to the issue being discussed.

Another important finding is the absence of neutral sentiment in the final Naïve Bayes classification results, where no tweets were classified as neutral. This suggests that the tweets in the dataset tended to have a clear sentiment polarity, either positive or negative, after undergoing preprocessing and classification using the developed model. Differences were also observed between the initial TextBlob classification results (887 Positive, 4 Neutral, and 113 Negative) and the Naïve Bayes Classifier results (828 Positive, 0 Neutral, and 176 Negative). These differences indicate that the two methods evaluate sentiment differently, and the trained Naïve Bayes model may be more sensitive to negative expressions in the dataset than the standard TextBlob approach. Furthermore, the Naïve Bayes model achieved an accuracy of 93.7%, demonstrating strong performance in classifying sentiment based on the provided training data.

The generated word cloud revealed several prominent keywords, including "Indonesia," "gelap," "rakyat," "negara," "pemerintah," "demo," "banyak," and "sangat." These keywords indicate that the discussion primarily focused on the condition of Indonesia described as "dark," the roles of the people and the government, and the occurrence of demonstrations. The planned TF-IDF analysis is expected to further assist in identifying the keywords that most strongly distinguish positive sentiment from negative sentiment.

Overall, this study demonstrates the presence of diverse sentiments regarding the topic "Indonesia Gelap" on Twitter, with positive sentiment dominating the discussion while negative sentiment also appearing in a significant proportion. The main keywords identified reflect discussions centered on national conditions, the public, the government, and protest activities.

Recommendations

Based on the results of this study, the author recognizes that there are still several limitations that did not fully meet the expected outcomes. Therefore, several recommendations are proposed for future research to achieve more comprehensive and improved results. Future studies are encouraged to employ other classification algorithms, such as unsupervised learning methods, and compare their performance with undersampling techniques. In addition, future research should consider incorporating TikTok comments, including emojis that represent emotions such as anger, happiness, sadness, and other emotional expressions. The inclusion of such features may provide richer contextual information and contribute to more accurate sentiment analysis results.

References

[1] P. Y. Saputra, D. H. Subhi, F. Zain, and A. Winatama, "IMPLEMENTASI SENTIMEN ANALISIS KOMENTAR CHANNEL VIDEO PELAYANAN PEMERINTAH DI YOUTUBE MENGGUNAKAN ALGORITMA NAÏVE BAYES".

[2] D. Darwis, N. Siskawati, and Z. Abidin, "Penerapan Algoritma Naive Bayes untuk Analisis Sentimen Review Data Twitter BMKG Nasional," vol. 15, no. 1.

[3] R. Fajar, S. Program, P. Rekayasa, N. Lunak, and R. Bengkalis, "Implementasi Algoritma Naive Bayes Terhadap Analisis Sentimen Opini Film Pada Twitter," vol. 3, no. 1.

[4] "Volume 3 Nomor 2 Juli 2021", [Online]. Available: https://t.co/TIk5mK5bwS

[5] A. N. Badri, N. Noviandi, F. Anastya, and M. Roland, "SENTIMENT ANALISIS UNTUK IDENTIFIKASI KEPUASAN MASYARAKAT TERHADAP KENAIKAN BBM MENGGUNAKAN ALGORITMA NAÏVE BAYES," JIKO (Jurnal Informatika dan Komputer), vol. 7, no. 2, p. 287, Sep. 2023, doi: 10.26798/jiko.v7i2.873.

[6] A. V. Sudiantoro and E. Zuliarso, "Analisis Sentimen Twitter Menggunakan Text Mining Dengan Algoritma Naïve Bayes Classifier ANALISIS SENTIMEN TWITTER MENGGUNAKAN TEXT MINING DENGAN ALGORITMA NAÏVE BAYES CLASSIFIER," vol. 10, no. 2, pp. 69–73, 2018.

[7] B. Gunawan, H. Sasty, P. #2, E. Esyudha, and P. #3, "JEPIN (Jurnal Edukasi dan Penelitian Informatika) Sistem Analisis Sentimen pada Ulasan Produk Menggunakan Metode Naive Bayes," vol. 4, no. 2, pp. 17–29, 2018, [Online]. Available: www.femaledaily.com

[8] F. Amaliah, I. Kadek, and D. Nuryana, "Perbandingan Akurasi Metode Lexicon Based Dan Naive Bayes Classifier Pada Analisis Sentimen Pendapat Masyarakat Terhadap Aplikasi Investasi Pada Media Twitter," Journal of Informatics and Computer Science, vol. 03, 2022.

[9] F. Khoirunnisa and S. Topiq, "ANALISIS SENTIMEN TERHDAP KEPERCAYAAN MASYARAKAT PADA PROSES PENEGAK HUKUM DI INDONESIA DENGAN MENGGUNAKAN ALGORITMA NAÏVE BAYES," Jurnal Informatika dan Teknik Elektro Terapan, vol. 12, no. 3, Aug. 2024, doi: 10.23960/jitet.v12i3.4683.

[10] A. Safira, A. S. Masyarakat… , and F. N. Hasan, "ANALISIS SENTIMEN MASYARAKAT TERHADAP PAYLATER MENGGUNAKAN METODE NAIVE BAYES CLASSIFIER," Jurnal Sistem Informasi, vol. 5, no. 1, 2023.

[11] G. A. Buntoro, "ANALISIS SENTIMEN HATESPEECH PADA TWITTER DENGAN METODE NAÏVE BAYES CLASSIFIER DAN SUPPORT VECTOR MACHINE," 2016. [Online]. Available: https://www.researchgate.net/publication/309322787
[12] P. Catur et al., "Moderasi Beragama Dalam Wacana Digital: Analisis Sentimen Twitter Terhadap Kasus Penolakan Rumah Ibadah Di Cilegon."
[13] A. Fauzi, R. Y. Fa'rifah, and E. N. Alam, "Analisis Sentimen Trend Makanan Dan Minuman dengan Support Vector Machine Sebagai Rekomendasi Peluang Bisnis Bagi UMKM."
[14] F. R. B. Kahi, A. Talakua, and R. Reynaldi, "Analisis Sentimen Masyarakat Di Twitter Terhadap Pemerintahan Anies Baswedan Menggunakan Metode Naive Bayes Classifier," Jurnal Minfo Polgan, vol. 13, no. 1, pp. 324–336, Apr. 2024, doi: 10.33395/jmp.v13i1.13636.
[15] P. A. Permatasari, L. Linawati, and L. Jasa, "Survei Tentang Analisis Sentimen Pada Media Sosial," Majalah Ilmiah Teknologi Elektro, vol. 20, no. 2, p. 177, Dec. 2021, doi: 10.24843/mite.2021.v20i02.p01.
[16] S. Suryono, D. Emha, and T. Luthfi, "Analisis sentimen pada twitter dengan menggunakan metode Naïve Bayes Classifier."
[17] A. Deviyanto, M. R. Didik Wahyudi, and T. Informatika UIN Sunan Kalijaga Yogyakarta Jl Marsda Adi Sucipto No, "PENERAPAN ANALISIS SENTIMEN PADA PENGGUNA TWITTER MENGGUNAKAN METODE K-NEAREST NEIGHBOR," Jurnal Informatika Sunan Kalijaga), vol. 3, no. 1, pp. 1–13, 2018, [Online]. Available: https://twitter.com/search?l=id&q=AHY%20since%3A2017-01- 01%20until%3A2017-01-
[18] D. Duei Putri, G. F. Nama, and W. E. Sulistiono, "Analisis Sentimen Kinerja Dewan Perwakilan Rakyat (DPR) Pada Twitter Menggunakan Metode Naive Bayes Classifier," Jurnal Informatika dan Teknik Elektro Terapan, vol. 10, no. 1, Jan. 2022, doi: 10.23960/jitet.v10i1.2262.
[19] A. A. Soebroto, "Buku Ajar AI, Machine Learning & Deep Learning," 2019. [Online]. Available: https://www.researchgate.net/publication/348003841
[20] MACHINE LEARNING ALGORITHMS and their Use Cases. [Online]. Available: www.freepik.com
[21] F. Adiwidya Pradana, "Penggunaan Artificial Intelligence dan Machine Learning dalam Pembelajaran Ilmu Eksakta", doi: 10.13140/RG.2.2.21555.95527.