OPEN ACCESS
Jurnal Ilmiah Ilmu Terapan Universitas Jambi
Vol. 8 No. 2. December 2024, pp. 452-466 DOI. 10.22437/jiituj.v8i2.35254

Research Article

SENTIMENT ANALYSIS OF PUBLIC OPINION ON PRESIDENTIAL ADVISORY APPOINTMENTS USING NAIVE BAYES CLASSIFICATION

Edi Surya Negara[1,*], Rezki Syaputra[1], Deni Erlansyah[1], Ria Andryani[1], Prihambodo Hendro Saksono[1], Ferdi Aditya[1], Padel Mohammad Agam[1]

[1] Data Science Interdisciplinaty Research Center, Universitas Bina Darma, Sumatera Selatan, Indonesia
Corresponding author email: e.s.negara@binadarma.ac.id

Article Info
Recieved: Jun 26, 2024
Revised: Sep 15, 2024
Accepted: Nov 03, 2024
OnlineVersion: Nov 10, 2024

Abstract
Social media platforms such as Twitter, Facebook, and YouTube have become significant channels for public discourse, where users freely express opinions, including negative sentiments and hate speech. To better understand public opinion, particularly in politically charged contexts, sentiment analysis can classify user comments as either positive or negative. This study aims to analyze public sentiment regarding the formation of a special advisory team for President Jokowi, using a sentiment classification approach. The study employed a Naïve Bayes classifier to analyze sentiment from 3,000 comments gathered from Twitter, Facebook, and YouTube. The dataset was divided into 80% training data (used to train the model with known sentiment) and 20% test data. The Naïve Bayes algorithm was chosen for its simplicity and effectiveness in handling large datasets in text classification tasks. The Naive Bayes classification on sentiment analysis of public opinion regarding the appointment of presidential advisors achieved an overall accuracy of 71% in classifying the test data. Negative sentiment was classified with an accuracy of 71%, while positive sentiment was classified with an accuracy of 70%. The results demonstrate that the Naïve Bayes classifier is a viable method for sentiment analysis in political discourse, although the model's performance indicates room for improvement. The novelty of this research lies in its focus on sentiment analysis of public opinion specifically related to presidential advisory appointments, an area not yet extensively explored in sentiment analysis studies. This study contributes to the field by providing insights into the public's perception of political decisions using machine learning techniques. The implications for future research include refining classification methods for better accuracy and applying the model to other political or governmental topics.

Keywords: Classification, Naïve Bayes, Sentiment Analysis, Twitter, Youtube

INTRODUCTION

As social media rapidly evolves into a primary communication medium for users worldwide, its significance in Indonesia continues to grow (Masood et al., 2022; Park & Oh, 2023). Platforms such as YouTube, Twitter, and Facebook have emerged as vital channels for expressing and sharing opinions. According to Hootsuite, Indonesia had 160 million social media users as of January 2020, reflecting a notable increase of 12 million (8.1%) between April 2019 and January 2020, with a penetration rate of 59% (Negara et al., 2021). These platforms serve various purposes, including promotion and information dissemination, effectively connecting users across generations (Kusuma, 2020; Ida et al., 2020; Nurhachita & Negara, 2021; Suwarni, 2021; Ali et al., 2023; Utami, Haryanto, & Subagyo, 2024).

In the political sphere, social media facilitates government communication and public engagement (Sari, Omeiza, & Mwakifuna, 2023; Utami, Johari, & Anggereini, 2023; Yohanie et al., 2023; Fitriana & Waswa, 2024). However, contentious issues often arise, such as the mixed public reactions to the appointment of special presidential staff by Joko Widodo, which sparked significant debate across social media platforms. This discourse presents a critical opportunity to analyze public sentiment, categorizing opinions as positive, negative, or neutral. Despite the increasing presence of social media in political communication, effective methods for systematically analyzing public sentiment from the vast amount of generated data remain insufficient (Su et al., 2017). Traditional manual sentiment analysis is impractical given the sheer volume of data, which emphasizes the urgent need for automated techniques capable of handling this complexity. Existing methods often lack the scalability and accuracy required for real-time sentiment analysis, creating a gap that this research aims to address (Hajiali, 2020; Asrial et al., 2024; Zakiyah, Boonma, & Collado, 2024).

The urgency of this research is underscored by the rapid growth of social media data and the necessity for governments and policymakers to understand public sentiment in real time. Social media serves as a critical space for public discourse, accurately identifying sentiments can significantly influence decision-making processes and enhance government responsiveness (Munir, 2023; Belkahla Driss et al., 2019; Weng et al., 2021; Asrial et al., 2023). Timely insights derived from sentiment analysis can empower policymakers to adapt their strategies in accordance with public opinion, thereby fostering greater democratic engagement. Machine learning is a data-based method, namely building models based on data. The more data is learned, the model or algorithm is able to recognize the system well and has high accuracy (Respati et al., 2022; Habibi, Jiyane, & Ozsen, 2023; Saputro et al., 2024).

The use of the right algorithm is very important to see the speed of data processing and more accurate results, such as the use of the Floyd Warshall algorithm which is suitable for finding the shortest route that is simple and easy to implement (Lintang et al., 2021), or the C4.5 algorithm has advantages in processing data with nominal, orthogonal, and continuous values. Another algorithm that is often used for classification is support vector machine, such as in the study of Cause of Death Classification (Widya Utami & Nyoman Saptiari STMIK Primakara, 2020). To tackle the challenges of sentiment classification efficiently, this study will employ the Naive Bayes method, which is recognized for its speed and accuracy when applied to large and diverse datasets (Drus & Khalid, 2019; Ravinder et al., 2024). By processing a dataset of 3,000 comments collected from social media platforms, this research seeks to automate sentiment classification, thereby providing meaningful insights into public opinion and enhancing the understanding of political sentiment.

The primary objectives of this study are to analyze public sentiment regarding the appointment of presidential staff using advanced sentiment analysis techniques, implement the Naive Bayes method to classify sentiments into positive, negative, and neutral categories, and evaluate the effectiveness of the Naive Bayes classifier in processing social media data, focusing on accuracy and reliability. By addressing these objectives, this research aims to fill the identified gap in sentiment analysis methods, offering valuable insights into public sentiment and enhancing the understanding of political discourse within Indonesia's vibrant social media landscape. This contribution not only advances the field of sentiment analysis but also aids policymakers in navigating the complexities of public opinion in a digital age.

RESEARCH METHOD

Research Design

Development and improvement of algorithmic capabilities and approaches in conducting social networking. The analysis, especially on the detection of communities on social networks, also continues. This research employs a quantitative approach to analyze public sentiment regarding the appointment of special presidential staff. By utilizing sentiment analysis techniques, the study aims to classify textual data into distinct sentiment categories-positive, negative, and neutral. The research flow for sentiment analysis can be seen in Figure 1. which is processed using the Python tool. The primary instrument used in this research is the Naive Bayes classifier, a statistical method for classification based on Bayes' theorem. This method is particularly well-suited for sentiment analysis due to its efficiency and effectiveness in handling large datasets. The research also employs text preprocessing techniques to prepare the data for analysis, including tokenization, stop-word removal, and stemming, which are essential for improving classification accuracy.

Research Target/Subject

Data collection involved extracting public comments related to the appointment of special presidential staff from the selected social media platforms. Comments were gathered using web scraping techniques, which allow for the automated extraction of data from online sources. The data collection process was conducted over a specific period to ensure the relevance and timeliness of the information gathered. This study uses a dataset with a CSV file type with an excel file extension. CSV (Comma Separated Values) is a data format in a database where each record is separated by a comma (,) or a semicolon (;). The data used is data taken from the social media Youtube, Facebook, and Twitter, where for YouTube social media the researchers took data from the official CNN Indonesia account with the title of the special presidential staff, the data is 21 thousand, for social media Facebook the researchers took data from the page. official Kompas TV with the title of Joko Widodo presisden's special staff with 1400 data and for Twitter social media the researchers took data with keywords (stafsus, special staff, and stafsus_presiden) with a total of 1010 data. The number of datasets used by researchers from three social media is 3000 data. with 80% training data and 20% test data.

Dataset

The data analysis process includes several key steps. First, the collected comments undergo preprocessing to clean and prepare the text for analysis. Next, the Naive Bayes classifier is implemented to classify the sentiments expressed in the comments into positive, negative, or neutral categories. The model's performance is evaluated using accuracy metrics, with a focus on the accuracy of each sentiment class. Additionally, confusion matrices are generated to analyze the classifier's performance in distinguishing between the sentiment categories. This systematic approach ensures that the analysis provides reliable insights into public sentiment regarding the appointment of special presidential staff.

The data labeling process will be carried out to determine the classification of opinions or views from the results of the crawled comment data earlier (He & Zhou, 2011; Li et al., 2023). In this labeling process, it is divided into 3 classes. They are positive class, negative class and neutral class. An example of the data labeling process is shown in Table 1 below. In this case, class negative label -1 states that the comments are words that contain hate speech or hate speech, while positive class labeled 1 are words that do not contain hate speech elements and neutral class labeled 0 are neutral words. does not contain hate speech or contains hate speech. Each data will go through the preprocessing process by changing the form of unstructured data into structured data according to needs such as overcoming repetitive words, standard words, foreign words, and characters that have no meaning.

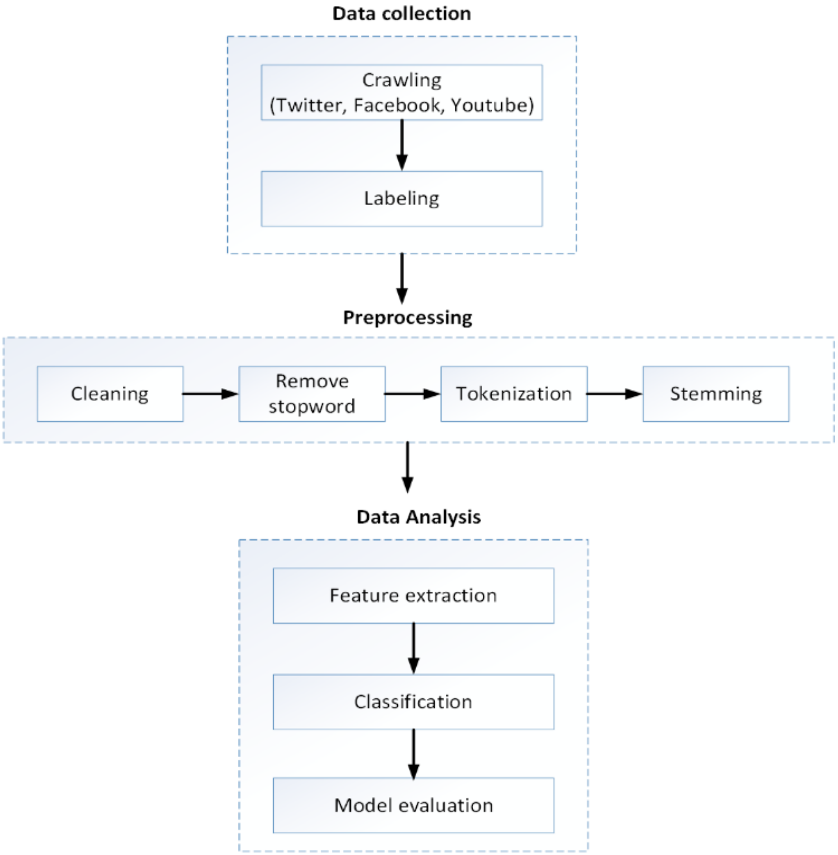

Figure 1. Research Framework

Table 1. Data Labelling Result

Comments	Source	Label
Dodol....wamen, stafsus, stafsusmilenial, komisaris a,b,c...yg sifatnya bagi2 kue kekuasaan dan.tidak perlu,itu yg perlu dirampingkan..	Twitter	-1
Demokrasi butuh oposisi milenial. Oposisi partai politik.	Youtube	0
Pa jokowi pasti sdh mempertimbangkan & aq sangat menghargai nya. Krn org muda yg berdedikasi lebih baik dr yg tua tp bebal & bengal	Facebook	1

Labeling Data

The types of data, how the data is collected, with which instruments the data is collected, and the techniques for collecting it, need to be explained clearly in this section. In the data analysis, several tasks were carried out, namely preprocessing. Preprocessing consists of several stages, namely: clearning, remove stopword, tokenization, stemming. Preprocessing is the stage of the process for cleaning data from unnecessary words or comments and words that have no meaning. This process is carried out in accordance with the contents of the data from the data collection process or data crawling from social media Youtube, Facebook, Twitter. The steps of the preprocessing process have the following sequence.

Cleaning is the process of removing symbols, punctuation, capital letters and numbers that often appear in comments from Twitter, Facebook and Youtube users (Krouska et al., 2016; Shahade et al., 2023), so that the data becomes ineffective and has no meaning such as: (# \\ S + "," ", x,!, ()). This process is carried out using a program, so that this cleaning runs automatically before saving the decoded results in the form of an excel file, the researcher creates the dataset in excel form because the Python programming language has a library that provides read and write services for csv or excel file types.

Remove Stopword is the process of removing meaningless words or words that have no meaning such as the word and, or, you, me (Chai, 2023; Saif et al., 2014). Tokenization is the process of breaking a sentence into pieces which are called tokens (Bakshi et al., 2016). A token can be thought of as a form of a word, phrase, or a meaningful element. Stemming is the process of changing a word into its basic form by removing the affixes before and after the word (Bakshi et al., 2016).

Data analysis technique

Data that has gone through the text processing stage can then proceed to the next stage, namely classification with the Naïve Bayes Classifier algorithm. Data in text form will appear two text classification results containing positive, neutral, and negative. The following is a calculation of the Naïve Bayes Classifier algorithm. The initial stage in the Naïve Bayes Classifier process is to calculate the probability of each class from the overall training data (Perdana & Pinandito, 2018). Testing process. This process is to determine the accuracy of the model built in the training process, generally using data called a test set to predict labels. The Naïve Bayes Classifier method consists of two stages in the text classification process, the training stage and the classification stage. At the training stage, an analysis process is carried out on a sample document in the form of vocabulary selection, namely words that may appear in the sample document collection which represent the document (Seref & Bostanci, 2018). The next step is to determine the probability for each category based on a sample of documents. The following below is an example of calculating the Naïve Bayes Classifier.

Table 2. Calculating the Naïve Bayes Classifier

Set	Document	Word	Label
1	Dokumen 1	President Jokowi must have considered this carefully, and I greatly appreciate it. Dedicated young people are better than the old but stubborn and unruly	1
2	Dokumen 2	Congratulations to the young individuals selected by Pakde... hopefully, you can contribute to building our nation NKRI	1
3	Dokumen 3	Democracy needs a millennial opposition. A political party opposition.	0
4	Dokumen 4	Wkwkwk, can't accept it, can you, sir, that President Jokowi chose millennial special staff? Are you jealous because, as a senior, you weren't selected?	-1
5	Dokumen 5	The Special Staff for Democracy needs full support from both the government and the public	1

In the example above, the negative class label -1 states that the comments are words that contain hate speech or hate speech, while the positive class labeled 1 are words that do not contain hate speech elements and the neutral class labeled 0 are words that contain hate speech elements. neutral does not contain hate speech or contain hate speech. The following below are the calculation results for testing data.

$$P(H|X) = \frac{P(X|H)P(H)}{P(H)} \quad \text{... (1)}$$

Calculate the prior probability of the positive, neutral, and negative classes.

y(pos)=2/4
y(net)=1/4
y(neg)=1/4

Then calculate the maximum likelihood value using the formula.

$$posterior = \frac{Perior\ x\ likelhood}{evidence} \qquad ... (2)$$

y(baik |pos) = (2+1)/(12+23) = 3/35
y(iri|pos) = (0+1) (12+23) = 1/35
y(demokrasi |pos) = (0+1)(12+23)= 1/35

y(baik |net) = (0+1)/(12+23) = 1/35
y(iri|net) = (0+1)(12+23)= 1/35
y(demokrasi |net) = (1+1)(12+23)= 2/35

y(baik |neg) = (0+1)/(12+23) = 1/35
y(iri|neg) = (1+1)(12+23)= 2/35
y(demokrasi|neg) = (0+1)(12+23)= 1/25
y(pos|d5) = 2/4*3/35*1/35*1/35= 3,49854E-05
y(net|d5) = 1/4*1/35*1/35*2/35 = 1,16618E-05
y(neg|d5) = 1/4*1/35*2/35*1/25 = 1,63265E-05
y(pos|d5) > y(neg|d5) dan y(net|d5)

The result of the above calculation is that the positive class on d5 has the highest value, so class d5 has a POSITIVE class. In the feature extraction process, the first process carried out by the system after tokenization is to convert the dataset into a vector representation using the library provided by Python called the Count Vectorizer library. For example, research uses 3 comments, including:

- (D1) “Demokrasi butuh oposisi milenial. Oposisi partai politik”
- (D2) “Selamat kpd para anak muda pilihan Pak Dhe...semoga bisa berkontribusi utk membangun NKRI”
- (D3) “Pa jokowi pasti sdh mempertimbangkan & aq sangat menghargai nya. Krn org muda yg berdedikasi lebih baik dr yg tua tp bebal & bengal”

After the system preprocesses, there are 4 standard words from the 3 sentences above, namely “Democracy”, “Congratulations”, “Youth”, and “Dedication”. After the above steps, each document is displayed as a vector with elements, when the word is present. in the document, the value is given 1, if not, then it is given a value of 0. For example, it is shown in Table 3.

Table 3. Making Word Vector

	Demokrasi	Selamat	Muda	Dedikasi
D1	1	1	0	0
D2	0	0	2	1
D3	1	0	0	1

Documents that have been converted into word vectors will then be calculated using the TF-IDF formula, using this formula will produce a word vector that has a weighted value. TF or Term Frequency itself is the number of times the words appear from a term in the document concerned, while IDF or Inverse Document Frequency is a calculation of how terms are spread or widely distributed in the collection of documents concerned. The process of calculating word weight is done by first calculating the TF or Term Frequency. You can see an example in Table 4.

Table 4. TF (Term Frequency) Calculating Process

	D1	D2	D3
Demokrasi	1	0	1
Selamat	1	0	0
Muda	0	2	0
Dedikasi	0	1	1

After the TF weight calculation process is complete, then the process of determining the DF or Document Frequency is carried out, namely the number of terms (t) appearing. You can see an example in Table 5.

Table 5. DF (Document Frequency) Calculating Process

T (Term)	DF (Document Frequency)
Demokrasi	2
Selamat	1
Muda	2
Dedikasi	2

Then after the TF and DF processes then proceed to calculate the IDF (Inverse Document Frequency) value by calculating the value from the log of D results or the number of documents in this case there are 3 tweets, of the 3 documents divided by the value of the DF (Document Frequency). Then it will produce a calculation value like Table 6.

Table 6. IDF (Inverse Document Frequency) Process

T (Term)	DF (Document Frequency)	D/DF	IDF (Inverse Document Frequency)
Demokrasi	2	1.5	log 1,5 = 0,176
Selamat	1	3	log 3 = 0,477
Muda	2	1.5	log 1,5 = 0,176
Dedikasi	2	1.5	log 1,5 = 0,176

After getting the IDF (Inverse Document Frequency) value, then proceed with calculating the TF-IDF. As in Table 7 Weighted Word Vector Examples below.

Table 7. Example of TF-IDF Calculating Process

Q	TF									
	D1	D2	D3	DF	D/DF	IDF	IDF+1	D1	D2	D3
Demokrasi	1	0	1	2	1.5	0.176	1.176	1.176	0	1.176
Selamat	1	0	0	1	3	0.477	1.477	1.477	0	0
Muda	0	2	0	2	1.5	0.176	1.176	0	2.35	0
Dedikasi	0	1	1	2	1.5	0.176	1.176	0	1.176	1.176
								2.653	3.528	2.352

The results of the word vectors that have been weighted can be seen in Table 8.

Table 8. Weighted Word Vector Example

	Demokrasi	Selamat	Muda	Dedikasi
D1	1.176	1.477	0	0
D2	0	0	2.352	1.176
D3	1.176	0	0	1.176

RESULTS AND DISCUSSION

Feature extraction process and the Naïve Bayes classification process which will later be compressed into one pipeline vectorizer class => transfomer => classifier. The classification process runs with the help of a library in the Python3 programming language which has the name scikit-learn library for the classification process, besides that there are numpy and pandas libraries as data reading. For the scikit-learn library used here are Pipeline, Count Vectorizer, Naïve Bayes, Multinomial NB, Confusion Matrix, Tfidf Transformer, and f1 Score. The first step in working on the feature extraction and classification process is to install the necessary libraries. Furthermore, after all libraries are installed, it is continued with the process of declaring all libraries that will be used. The program code for the declaration is in Figure 2.

```
import pandas as pd
import numpy as np
from sklearn.naive_bayes import MultinomialNB
from sklearn.svm import LinearSVC, SVC
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfTransformer
from sklearn.pipeline import Pipeline
from sklearn.model_selection import train_test_split
from sklearn.model_selection import cross_val_score
from sklearn.metrics import classification_report,confusion_matrix, accuracy_score, f1_score, precision_score, recall_score
```

Figure 2. The Library declaration function used

After completing the declaration of the library, it is continued with the process of taking a dataset that will be used as training data using the Pandas library. For the program code in Figure 3.

```
data = pd.read_excel
len(data)
```

Figure 3. Calls Up The Data Set

Furthermore, the process of making a pipeline class in which there are 3 steps, namely changing the dataset from which Twitter data is crawled into a vector representation (converting letters to numbers) using the Count Vectorizer library with weighting using word vectors in the Tfidf Transformer library, the last stage is classification using the Multinominal Naive Bayes library. The process of implementing the three pipeline class creation processes is in Figure 4.

```
#Multinomial Naive Bayes
pipeline_mnb = Pipeline([
    ('vect', CountVectorizer()),
    ('tfidf', TfidfTransformer(use_idf=True, smooth_idf=True)),
    ('clf', MultinomialNB(alpha=1))
])

txt = data['cleantext'].values.astype('U')
#X_train, X_test, y_train, y_test = train_test_split(data['cleantext'], data['label'],  test_size=0.33, random_state = 0)
X_train, X_test, y_train, y_test = train_test_split(txt, data['label'],  test_size=0.33, random_state = 0)
pipeline_mnb.fit(X_train, y_train)
```

Figure 4. The Pipeline Class Implementation Process

In the process of classifying this data, the researcher used randomized test data from 20% or 0.2 of the training data. The process of classifying this data is carried out using probability calculations from each class, so that new researchers can get clear results from the predicted data input. The final stage after carrying out all the classification processes, then it can be calculated from the performance of the algorithm used.

Feature extraction process and the Naïve Bayes classification process which will later be compressed into one pipeline vectorizer class => transfomer => classifier. The classification process runs with the help of a library in the Python3 programming language which has the name scikit-learn library for the classification process, besides that there are numpy and pandas libraries as data reading. For the scikit-learn library used here are Pipeline, Count Vectorizer, Naïve Bayes, Multinomial NB, Confusion Matrix, Tfidf Transformer, and f1 Score. The first step in working on the feature extraction and classification process is to install the necessary libraries. Furthermore, after all libraries are installed, it is continued with the

Test the Twitter Data Model

To determine the level of performance of the Naïve Bayes Algorithm, the researchers tested the model. The results of the classification will later be displayed in the form of confusion matrix. The table displayed in this confusion matrix consists of predicted class and actual class. The model of confusion matrix can be seen in Table 9.

Table 9. Confusion Matrix Model

		Predict Class	
		Class A	Class B
Actual Class	Positive	TP	FP
	Negative	FN	TN

To find out the value of the accuracy of the model, it is obtained from the right amount of data the clarification results are divided by the total of the data. In the test process this model will produce a confusion matrix with a size of 2x2 which can be seen in Table 10.

Table 10. Confusion Matrix Results

		Predict Class	
		Positive	Negative
Actual Class	Positive	358	96
	Negative	143	228

As in Table 10 above, the confused matrix matrix with a size of 2 x 2 each column represents the value of each class, namely the positive class and the negative class. To calculate the process of calculating the value of precision, recall and f-1 score in this system can be seen in Figure 5.

```
txt = data['cleantext'].values.astype('U')
#X_train, X_test, y_train, y_test = train_test_split(data['cleantext'], data['label'],  test_size=0.33, random_state = 0)
X_train, X_test, y_train, y_test = train_test_split(txt, data['label'],  test_size=0.33, random_state = 0)
pipeline_mnb.fit(X_train, y_train)
predictions = pipeline_mnb.predict(X_test)

print("Accuracy: {}".format(accuracy_score(y_test, predictions)))
print("F1 Score: {}".format(f1_score(y_test, predictions)))
print("Precision score: {}".format(precision_score(y_test, predictions)))
print("Recall score: {}".format(recall_score(y_test, predictions)))
print("Confusion matrix: {}".format(confusion_matrix(y_test, predictions)))
```

Figure 5. The process of calculating the value of precision, recall and F-1 score

The results of the precision, recall, and f-1 score have an assessment measure of 0-1. The higher the value, the better, in the sense that the closer to the number 1 value from 0, the better the system. The results of the process of evaluating the overall model of this system are shown in Figure 6 below.

```
Accuracy: 0.7103
F1 Score: 0.6561
Precision score: 0.7037
Recall score: 0.6146
Confusion matrix:
[[358  96]
 [143 228]]
             precision    recall  f1-score   support

          0       0.71      0.79      0.75       454
          1       0.70      0.61      0.66       371

avg / total       0.71      0.71      0.71       825
```

Figure 6. Model Test Result

By knowing the value of precision, recall, and f-1 Score in the performance of the entire system, we can find out the ability of the system to find the accuracy or truth of the information requested by the user with the answer results issued by the system and inform the success rate of a system. in determining back information or the accuracy value is 71%. The results of the precision, recall, and f-1 score values in each class are shown in Table 11 below.

Table 11. Results of The Value Precision, Recall, And F-1 Score

Classification	Precission	Recall	F-1 Score
Positive	0.71	0.79	0.75
Negative	0.70	0.61	0.66

It can be seen from the model test results in Table 11 that the precision and recall values of each class can be seen the level of system processing ability in finding the level of accuracy between the information desired by the user as a positive class is "71%", and for negative class is "70. % ". The success rate of system processing in retrieving positive class information was “79%”, for negative class it was “61%”. With these values it can be said that the system performance from the success of the system to find back information that is positive and negative. The result shown in Figure 7.

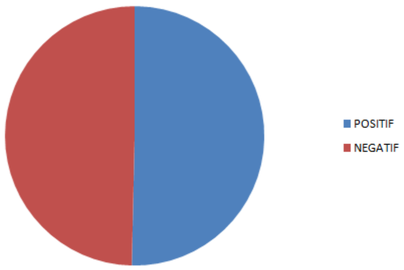

Figure 7. Graph of Sentiment Analysis Model Test Results on Twitter

Test the Facebook Data Model

In the test process this model will produce a confusion matrix with a size of 2x2 which can be seen in Table 12.

Table 12. Confusion Matrix Result

		Predict Class	
		Positive	Negative
Actual Class	Positive	558	105
	Negative	261	334

Get the results from the accuracy value and 2x2 confusion matrix in Figure 8.

```
Accuracy: 0.7100
F1 Score: 0.7496
Precision score: 0.7668
Recall score: 0.7331
Confusion matrix:
[[138  66]
 [ 79 217]]
             precision    recall  f1-score   support

          0       0.64      0.68      0.66       204
          1       0.77      0.73      0.75       296

avg / total       0.71      0.71      0.71       500
```

Figure 8. Model Test Results

The accuracy value obtained from testing the model is 71.0% whose calculation process is based on the number of values of the diagonal confusion matrix divided by the entire amount of data. Because the amount of data in each training data class is not balanced, the amount of accuracy value is not the most important. The results of the precision, recall, and f-1 score values in each class are shown in Table 13.

Table 13. Result Of The Value Precision, Recall, And F-1 Score

Classification	Precission	Recall	F-1 Score
Positive	0.64	0.68	0.66
Negative	0.77	0.73	0.75

It can be seen from the results of the evaluation of the model in Table 16, it can be seen that the precision and recall values of each class can be seen the level of system processing ability in finding the level of accuracy between the information desired by the user as a positive class is "64%", and for negative class is "77 % ". The success rate of system processing in retrieving positive class information is “68%”, for negative class is “73%”. With these values it can be said that the system performance from the success of the system to find back information that is positive and negative. The result shown in Figure 9.

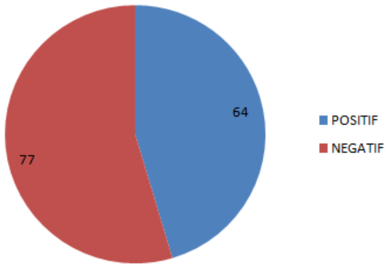

Figure 9. Graph of Sentiment Analysis Model Test Results on Facebook

Test the Youtube Data Model

In the test process this model will produce a confusion matrix with a size of 2x2 which can be seen in Table 14.

Table 14. Confusion Matrix Result Youtube

		Predict Class	
		Positive	Negative
Actual Class	Positive	440	144
	Negative	356	268

The accuracy value obtained from testing the model is 70.0% whose calculation process is based on the number of values of the diagonal confusion matrix divided by the entire amount of data. Because the amount of data in each training data class is not balanced, the amount of accuracy value is not the most important. The results of the precision, recall, and f-1 score values in each class are shown in Table 15 below.

Table 15. Results Of The Value Precision, Recall, And F-1 Score

Classification	Precission	Recall	F-1 Score
Positive	0.71	0.59	0.64
Negative	0.69	0.73	0.75

It can be seen from the results of the evaluation of the model in Table 15, it can be seen that the precision and recall values of each class can be seen the level of system processing ability in finding the level of accuracy between the information desired by the user as a positive class is "71%", and for negative class is "69 % ". The success rate of system processing in retrieving positive class information was “59%”, for negative class was “73%”. With these values it can be said that the system performance from the success of the system to find back information that is positive and negative. The result shown in Figure 10.

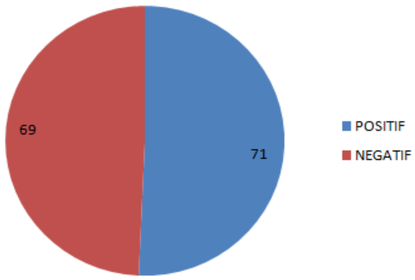

Figure 10. Graph of Sentiment Analysis Model Test Results on Youtube

The data results from this study highlight that the Naive Bayes classifier achieved an accuracy of 71% in categorizing public sentiment related to the appointment of special presidential staff. This accuracy aligns with findings from previous research, such as Samuel et al., (2020); Gaye et al., (2021); and Landeghem et al., (2022), which also demonstrated the robustness of Naive Bayes for sentiment classification on large, diverse datasets. However, while comparable to prior studies, the 71% accuracy indicates room for improvement, particularly in handling informal language and imbalanced data. The 71% accuracy suggests that while Naive Bayes is effective, factors such as non-standard language, including slang common on social media, and dataset imbalance affect the overall performance. Studies by (Junior et al., 2021) found similar limitations when working with informal, unstructured data. This research further confirms the need for enhanced preprocessing techniques to address these challenges and improve model accuracy. These findings can be generalized to broader political discussions on social media, offering valuable insights into how public opinion is shaped and expressed online.

The implications are significant for both political analysts and policymakers, as real-time sentiment analysis can offer actionable insights to guide decision-making and public communication strategies. Policymakers could use this data to better understand the public's response to political events and decisions, allowing them to respond more effectively. The novelty of this research lies in its application of Naive Bayes to sentiment analysis within Indonesia's political landscape. While Naive Bayes has been widely used in other domains, its specific application to understanding public sentiment surrounding presidential appointments in Indonesia offers fresh perspectives and adds to the growing field of political sentiment analysis. However, there are limitations to this study. The dataset, though substantial, was limited to 3,000 data points, and the imbalance in sentiment classes (positive, negative, neutral) affected the overall classification accuracy. Additionally, informal language on social media posed a challenge for classification models, underscoring the need for better handling of non-standard language.

Future research should focus on expanding the dataset and employing more advanced language preprocessing techniques, such as word embedding models or transformers, which have shown promise in handling unstructured social media data. Additionally, exploring ensemble methods that combine Naive Bayes with other classifiers may lead to improved accuracy in sentiment classification. This could pave the way for more robust sentiment analysis systems in the political domain and beyond

CONCLUSION

The Naive Bayes method proved effective for classifying sentiment in social media data, particularly from platforms like Twitter, Facebook, and YouTube. However, the accuracy of classification is significantly influenced by the balance of word distribution across training classes and the presence of non-standard language, such as slang. This underscores the need for balanced datasets and careful preprocessing of informal language to optimize classification performance. The analyze public sentiment regarding the formation of a special staff for President Jokowi, using the Naive Bayes Classifier algorithm for data classification. The classification results on the test data showed that the algorithm achieved an accuracy rate of 71%. The accuracy for each sentiment was 71% for negative sentiment and 70% for positive sentiment, indicating that the method performs reasonably well in sentiment classification, although there is still room for improvement.

The findings from this study demonstrate the potential of machine learning, specifically Naive Bayes, as a powerful tool for analyzing public sentiment in political discourse. In a political context, understanding public opinion such as reactions to government appointments can provide valuable insights for policymakers and political strategists. This type of analysis can be used to gauge public support or dissatisfaction with political decisions in real time, offering governments and political actors a data-driven approach to crafting more responsive and informed policies. Future sentiment analysis research should focus on enhancing preprocessing techniques to better manage non-standard language and achieve more balanced datasets. Moreover, the political world could benefit from further development of advanced sentiment analysis algorithms tailored to the nuances of political communication on social media, helping to identify public trends and opinions with greater accuracy.

ACKNOWLEDGMENTS

We would like to express our deepest gratitude to the Data Science Interdisciplinary Research Center and Universitas Bina Darma for the support and facilities provided, so that this research can be completed properly. We also thank all the colleagues involved for their contributions to the success of this research.

AUTHOR CONTRIBUTIONS

Author 1-2 creates articles and creates instruments and is responsible for research, author 3-4 Analyzes research data that has been collected, author 5-6-7 assist in research data analysis, instrument validation, and input research data

CONFLICTS OF INTEREST

The author(s) declare no conflict of interest.

REFERENCES

Ali, I., Balta, M., & Papadopoulos, T. (2023). Social media platforms and social enterprise: Bibliometric analysis and systematic review. *International Journal of Information Management*, *69*, 102510. https://doi.org/10.1016/J.IJINFOMGT.2022.102510.

Asrial, A., Syahrial, S., Kurniawan, D. A., Aldila, F. T., & Iqbal, M. (2023). Implementation of web-based character assessment on students' character outcomes: A review on perception and gender. *JOTSE: Journal of Technology and Science Education*, *13*(1), 301-328. https://doi.org/10.3926/jotse.1564.

Asrial, A., Syahrial, S., Kurniawan, D. A., Putri, F. I., Perdana, R., Rahmi, R., Susbiyanto, S., & Aldila, F. T. (2024). E-Assessment for character evaluation in elementary schools. *Qubahan Academic Journal*, *4*(3), 806-822. https://doi.org/10.48161/qaj.v4n3a595.

Bakshi, R. K., Kaur, N., Kaur, R., & Kaur, G. (2016). Opinion mining and sentiment analysis. *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, 452–455.

Belkahla Driss, O., Mellouli, S., & Trabelsi, Z. (2019). From citizens to government policy-makers: Social media data analysis. *Government Information Quarterly*, *36*(3), 560–570. https://doi.org/10.1016/J.GIQ.2019.05.002.

Chai, C. P. (2023). Comparison of text preprocessing methods. *Natural Language Engineering*, *29*(3), 509–553. https://doi.org/10.1017/S1351324922000213.

Drus, Z., & Khalid, H. (2019). Sentiment analysis in social media and its application: Systematic literature review. *Procedia Computer Science*, *161*, 707–714. https://doi.org/10.1016/J.PROCS.2019.11.174.

Fitriana, H., & Waswa, A. N. (2024). The influence of a realistic mathematics education approach on students' mathematical problem solving ability. *Interval: Indonesian Journal of Mathematical Education*, *2*(1), 29-35. https://doi.org/10.37251/ijome.v2i1.979.

Gaye, B., Zhang, D., & Wulamu, A. (2021). A tweet sentiment classification approach using a hybrid stacked ensemble technique. *Information (Switzerland)*, *12*(9). https://doi.org/10.3390/info12090374.

Habibi, M. W., Jiyane, L., & Ozsen, Z. (2024). Learning revolution: The positive impact of computer simulations on science achievement in madrasah ibtidaiyah. *Journal of Educational Technology and Learning Creativity*, *2*(1), 13-19. https://doi.org/10.37251/jetlc.v2i1.976.

Hajiali, M. (2020). Big data and sentiment analysis: A comprehensive and systematic literature review. *Concurrency and Computation: Practice and Experience*, *32*(14), e5671. https://doi.org/https://doi.org/10.1002/cpe.5671.

He, Y., & Zhou, D. (2011). Self-training from labeled features for sentiment analysis. *Information Processing & Management*, *47*(4), 606–616. https://doi.org/10.1016/J.IPM.2010.11.003.

Ida, R., Saud, M., & Mashud, M. (2020). An empirical analysis of social media usage, political learning and participation among youth: a comparative study of Indonesia and Pakistan. *Quality & Quantity*, *54*(4), 1285–1297. https://doi.org/10.1007/s11135-020-00985-9.

Junior, A. B., da Silva, N. F. F., Rosa, T. C., & Junior, C. G. C. (2021). Sentiment analysis with genetic programming. *Information Sciences*, *562*, 116–135. https://doi.org/10.1016/J.INS.2021.01.025.

Krouska, A., Troussas, C., & Virvou, M. (2016). The effect of preprocessing techniques on Twitter sentiment analysis. *2016 7th International Conference on Information, Intelligence, Systems & Applications (IISA)*, 1–5. https://doi.org/10.1109/IISA.2016.7785373.

Kusuma, R. S. (2020). Improving students' basic asking skills by using the discovery learning model. *Tekno - Pedagogi : Jurnal Teknologi Pendidikan*, *10*(2), 8-13. https://doi.org/10.22437/teknopedagogi.v10i2.32743.

Landeghem, J. Van, Blaschko, M., Anckaert, B., & Moens, M.-F. (2022). Benchmarking scalable predictive uncertainty in text classification. *IEEE Access*, *10*, 43703–43737. https://doi.org/10.1109/ACCESS.2022.3168734.

Li, Q., Yang, Y., Li, C., & Zhao, G. (2023). Energy vehicle user demand mining method based on fusion of online reviews and complaint information. *Energy Reports*, *9*, 3120–3130. https://doi.org/10.1016/J.EGYR.2023.02.004.

Cahyaningati, K. L., & Vikaliana, R. (2021). Implementasi floyd warshall algorithm untuk optimasi distribusi J&T Express: Studi kasus pickup distribution center J&T Express pasar minggu [Implementation of Floyd Warshall Algorithm for J&T Express distribution optimization: Case study of J&T Express pickup distribution center pasar minggu]. *Jurnal Ilmiah Ilmu Terapan Universitas Jambi*, *5*(1), 93-109. https://doi.org/10.22437/jiituj.v5i1.15416.

Munir, A. (2023). Leveraging Social Media Geographic Information for Smart Governance and Policy Making: Opportunities and Challenges. *Global Perspectives on Social Media Usage Within Governments*.

Masood, K., Khan, M. A., Saeed, U., Al Ghamdi, M. A., Asif, M., & Arfan, M. (2022). Semantic Analysis to Identify Students' Feedback. *The Computer Journal*, *65*(4), 918–925. https://doi.org/10.1093/comjnl/bxaa130.

Negara, E. S., Andryani, R., & Amanda, R. (2021). Network analysis of YouTube videos based on keyword search with graph centrality approach. *Indonesian Journal of Electrical Engineering and Computer Science*, *22*(2), 780–786. https://doi.org/10.11591/ijeecs.v22.i2.pp780-786.

Nurhachita, & Negara, E. S. (2021). A comparison between deep learning, naïve bayes and random forest for the application of data mining on the admission of new students. *IAES International Journal of Artificial Intelligence*, *10*(2), 324–331. https://doi.org/10.11591/ijai.v10.i2.pp324-331.

Park, J., & Oh, H. (2023). Dynamic Automated Labeling System for Real-Time User Intention Analysis. *IEEE Access*, *11*, 139882–139902. https://doi.org/10.1109/ACCESS.2023.3339773.

Perdana, R. S., & Pinandito, A. (2018). *Combining Likes-Retweet Analysis and Naive Bayes Classifier within Twitter for Sentiment Analysis*.

Ravinder, B., Seeni, S. K., Prabhu, V. S., Asha, P., Maniraj, S. P., & Srinivasan, C. (2024). Web data mining with organized contents using naive bayes algorithm. *2024 2nd International Conference on Computer, Communication and Control (IC4)*, 1–6. https://doi.org/10.1109/IC457434.2024.10486403.

Respati, S., Isram, M., & Kusrini, S. (2022). *Estimation of Queue Length at Signalized Intersections Using Artificial Neural Network*, *6*(2). https://doi.org/10.22437/jiituj.v6i2.22958.

Saif, H., Fernandez, M., & Alani, H. (2014). Automatic stopword generation using contextual semantics for sentiment analysis of twitter. In *Language Resources and Evaluation*. https://doi.org/10.13140/2.1.3523.8088.

Samuel, J., Ali, G. G. M. N., Rahman, M. M., Esawi, E., & Samuel, Y. (2020). COVID-19 public sentiment insights and machine learning for tweets classification. *Information (Switzerland)*, *11*(6). https://doi.org/10.3390/info11060314.

Saputro, H. D., Rustaminezhad, M. A., Amosa, A. A., & Jamebozorg, Z. (2023). Development of e-learning media using adobe flash program in a contextual learning model to improve students' learning outcomes in junior high school geographical research steps materials. *Journal of*

Educational Technology and Learning Creativity, *1*(1), 25-32. https://doi.org/10.37251/jetlc.v1i1.621.

Sari, R., Omeiza, I. I., & Mwakifuna, M. A. (2023). The influence of number dice games in improving early childhood mathematical logic in early childhood education. *Interval: Indonesian Journal of Mathematical Education*, *1*(2), 61-66. https://doi.org/10.37251/ijome.v1i2.776.

Seref, B., & Bostanci, E. (2018). Sentiment analysis using naive bayes and complement naive bayes classifier algorithms on hadoop framework. *2018 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, 1–7. https://doi.org/10.1109/ISMSIT.2018.8567243.

Shahade, A. K., Walse, K. H., Thakare, V. M., & Atique, M. (2023). Multi-lingual opinion mining for social media discourses: an approach using deep learning based hybrid fine-tuned smith algorithm with adam optimizer. *International Journal of Information Management Data Insights*, *3*(2), 100182. https://doi.org/10.1016/J.JJIMEI.2023.100182.

Su, L. Y.-F., Cacciatore, M. A., Liang, X., Brossard, D., Scheufele, D. A., & Xenos, M. A. (2017). Analyzing public sentiments online: combining human- and computer-based content analysis. *Information, Communication & Society*, *20*(3), 406–427. https://doi.org/10.1080/1369118X.2016.1182197.

Suwarni, R. (2021). Analysis the process of observing class iv students in thematic learning in primary schools. *Tekno - Pedagogi : Jurnal Teknologi Pendidikan*, *11*(1), 26-32. https://doi.org/10.22437/teknopedagogi.v11i1.32717.

Utami, N. W., & Saptiari, N. N. (2020). Penerapan data mining untuk klasifikasi penyebab kematian menggunakan algoritma support vector machine [Application of data mining for classification of causes of death using support vector machine algorithm]. *Jurnal Ilmiah Ilmu Terapan Universitas Jambi*, *4*(2), 234-240. https://doi.org/10.22437/jiituj.v4i2.13268.

Utami, R. E., Johari, A., & Anggereini, E. (2023). Learning outcomes for students junior high school: Entrepreneurship-Based learning video. *Integrated Science Education Journal*, *4*(2), 69-76. https://doi.org/10.37251/isej.v4i2.328.

Utami, S. M., Haryanto, H., & Subagyo, A. (2024). The development of electronic students' worksheets (e-lkpd) based on argument driven inquiry learning model to improve scientific argumentation skills. *Integrated Science Education Journal*, *5*(2), 65-73. https://doi.org/10.37251/isej.v5i2.810.

Weng, S., Schwarz, G., Schwarz, S., & Hardy, B. (2021). A Framework for Government Response to Social Media Participation in Public Policy Making: Evidence from China. *International Journal of Public Administration*, *44*(16), 1424–1434. https://doi.org/10.1080/01900692.2020.1852569.

Yohanie, D. D., Botchway, G. A., Nkhwalume, A. A., & Arrazaki, M. (2023). Thinking process of mathematics education students in problem solving proof. *Interval: Indonesian Journal of Mathematical Education*, *1*(1), 24-29. https://doi.org/10.37251/ijome.v1i1.611.

Zakiyah, Z., Boonma, K., & Collado, R. (2024). Physics learning innovation: Song and animation-based media as a learning solution for mirrors and lenses for junior high school students. *Journal of Educational Technology and Learning Creativity*, *2*(2), 54-62. https://doi.org/10.37251/jetlc.v2i2.1062.