ILKOM Scientific Journal Vol. 15, No. 3, December 2023, pp. 473-487
Accredited 2nd by RISTEKBRIN No. 200/M/KPT/2020; E-ISSN 2548-7779 | P-ISSN 2087-1716

Research Article

Analysis of Twitter User Sentiment on Presidential Candidate *Anies Baswedan* Using Naïve Bayes Algorithm

Rudi Setiawan [a,1,*]; Fitria Dewi [a,2]
[a] *Trilogy University, Jl.TMP. Kalibata No.1 Pancoran, Jakarta Selatan 12760, Indonesia*
[1] *rudi@trilogi.ac.id;* [2]*fitria.dewi@trilogi.ac.id*
* *Corresponding author*

Article history: Received September 07, 2023; Revised September 09, 2023; Accepted November 29, 2023; Available online December 20, 2023

Abstract

Indonesian hold presidential election in 2024. One of the most discussed topics by public is the presidential candidates. The discussion about the presidential candidate certainly reaped various kinds of responses from public, ranging from support to statements of disapproval. This research was limited to the candidacy of *Anies Baswedan* as a presidential candidate before a vice president candidate as his pair was selected. The purpose of this study is to conduct a sentiment analysis of public responses regarding Indonesia 2024 presidential candidate *Anies Baswedan* using *tweets* data from October 2022 to January 2023 using the *naïve bayes classifier algorithm*. This is expected to provide an overview of the public opinions on Twitter. Three test models were carried out with differences in the division of the amount of training data and test data, respectively 60%:40%, 70%:30% and 80%:20%. The test results showed the highest accuracy level was obtained by the 3rd model using training and testing data of 80%:20% with an accuracy value of 76.21%. Further research is recommended to conduct sentiment analysis on the pairs of Presidential and Vice-Presidential candidates who have been officially registered with the General Election Commission using various other classification algorithms.

Keywords: Anies Baswedan; Naïve Bayes; Presidential Candidate; Sentiment Analysis.

Introduction

In Indonesia, the presidential election is one of the most important events for all levels of society. 2024 is the year of the return of the democratic party for all the people of the Republic of Indonesia, because in that year the term of the President and Vice President of the Republic of Indonesia for the 2019-2024 period end. There have been many surveys about the 2024 presidential candidates spread on social media since the pre-declaration period began. One of the earliest figures declared as a candidate for the 2024 presidency is *Anies Baswedan* who recently completed his administration as Governor of DKI Jakarta for the 2017-2022 with various policies and work programs that are quite effective in solving problems in the DKI Jakarta area [1].

The emergence of *Anies Baswedan* as a presidential candidate carried by the National Democratic party reaped pros and cons in the country. With the current situation, people can easily express their opinions to the public through social media. One of the social media that is widely used by the people of Indonesia is Twitter. Twitter is a social networking service that allows users to send and read a text-based message with other users using computer or mobile devices. Currently, Twitter is widely used to discuss hot issues such as entertainment, politics, social, to government [2]. Based on the results of the 2022 Internet User Penetration and Behavior Survey owned by the Indonesian Internet Service Providers Association or APJII, the number of people who have been connected to the internet in the 2021, 2022 period has touched 210,026,769 people out of a total population of 272,682,600 in 2021. Of the total, 98.02% are social media users, 92.21% of users use the internet to access information/news, and 90.21% are used to work and study from home [3].

In social media, it is not surprising that many users give negative or positive opinions on an issue or news, as well as news about the 2024 presidential candidate. The election of *Anies Baswedan* as a candidate for the 2024 presidency carried by the *Nasdem* party managed to reap various kinds of comments from Twitter users in Indonesia. The responses were ranging from support, feedback, to criticism. Opinions circulating on Twitter can be categorized based on their respective sentiment classes using classification algorithms. Several classification algorithms can be used in the sentiment analysis process are *random forest, decision tree, k-nearest neighbors, Naïve Bayes, and logistic regression* [4], [5]. Naïve Bayes algorithm is a classification method that applies probability theory and statistics, invented by Thomas Bayes who is a scientist from England [6]. Naïve Bayes can work with small amounts of data

http://dx.doi.org/10.33096/ilkom.v15i3.1775.473-487

with large class handling, has a simple structure that is fast in processing because training and classification can be done with one iteration of data [7].

Previous research on the opinion classification of the presidential and vice-presidential candidates for the 2019-2024 period showed that *Joko Widodo* and *Ma'aruf Amin* had the highest positive sentiment value, which was 25% compared to positive sentiment of Prabowo *Subianto* and *Sandiaga Uno* which only produced a value of 5.1% [8]. In 2021, research on sentiment analysis using the Naïve Bayes classifier showed a harmonic value between recall and precision of 0.92, which meant that the system worked well in detecting existing sentiments [9]. The 2023 research on sentiment analysis of the 2024 Indonesian Presidential candidates using the LSTM algorithm conducted by [10] on 3 figures of presidential candidates, namely *Ganjar Pranowo*, *Prabowo Subianto* and *Ridwan Kamil*, showed 82% accuracy results in the *Ganjar Pranowo* model, 82% in the *Prabowo Subianto* model and 89% accuracy in the *Ridwan Kamil* model. In the same year, a study entitled sentiment analysis of 2024 presidential candidates using the Support Vector Machine algorithm on Twitter with data collection between October 17-25, 2022, found that the positive sentiment obtained by presidential candidate *Ganjar Pranowo* was higher than other presidential candidates, namely 55%, *Prabowo* 30% and *Anies Baswedan* 15%, while *Anies Baswedan*'s negative sentiment was higher 89% than *Ganjar* 8% and *Prabowo* 3% [11].

In this study, the Naïve Bayes algorithm was chosen because of its advantages in classifying without requiring large amounts of data based on the positive and negative comments obtained from Twitter. The input used in this study is in the form of Indonesian tweets data classified using the Naïve Bayes Classifier to support the classification process and sentiment analysis of each tweet. The benefit of this study is to get an idea of how much the positive and negative opinions of the public towards the candidacy of *Anies Baswedan* as a presidential candidate in the 2024 presidential election.

Literature Review

A. Sentiment Analysis

Sentiment analysis aims to analyze an opinion, sentiment, evaluation, attitude, judgment and emotion conveyed by a person to find out whether the speaker or writer has an opinion or agrees with a particular topic, product, service, individual, or activity [12]. Sentiment analysis has several classification methods such as Naïve Bayes classifier, support vector machine, and maximum entropy. However, the classification method that is more often used by researchers in the sentiment analysis process is the Naïve Bayes classifier because it is considered easier to use, short processing time, and has a fairly high level of accuracy [13].

B. Text Mining

Text mining is a process of extracting or searching for information to extract knowledge from textual data sources [14]. This method is often used to analyze and understand large amounts of text automatically, allowing the identification of patterns, trends, and knowledge that is difficult or impractical if conducted manually [15]. Some of the main aspects of text mining include information retrieval, text parsing, entity recognition, sentiment analysis, text classification, text grouping, information extraction, topic modeling, text recommendation. Text mining has become an important part in a variety of applications, including social media analysis and natural language processing. The existence of text mining allows organizations and individuals to find valuable information from existing data sets so that they can be used for better decision processes for an organization.

C. Classification

Classification is the process of assessing a data that has similar characteristics in a document and entering these data into certain classes [16]. Some classification methods that are often used include random forest, decision tree, k-nearest neighbors, Naïve Bayes, and logistic regression.

In the classification process there are two main tasks performed [17]:

1. Development models are used as prototypes that will be stored in memory.
2. The use of models that have been created to help the classification process of a data object to find out the placement of the data object class.

D. Data Pre-processing

Pre-processing is a series of processes used to select data in text form into a more structured set of data by going through several stages, such as case folding, tokenizing, filtering, and stemming, *pre*-processing is important to make data more readable and freer from data errors [18]. The stages in pre-processing according to [19] are case folding, tokenizing, filtering, stemming, and stop-word removal.

1. Case folding is a stage where there will be a process of changing text that is originally uppercase to lowercase, and removing all punctuation marks contained in the sentence.
2. Tokenizing is a stage where every word in a sentence will be separated based on a predetermined space.
3. Filtering is a stage of removing unimportant words by utilizing Indonesian stop words.
4. Stemming is the stage of converting a word into a basic word form.
5. Stop-word removal is the stage of removing unimportant words by utilizing a bag-of-words approach.

E. Naïve Bayes

Naïve Bayes classifier or bayesian classification is an algorithm and one of the classification techniques that combines the use of statistics and probability theory to solve the problem of a supervised learning case [20]. The formula for calculating probability in the Naïve Bayes method is shown in Equation 1.

$$P(H|X) = \frac{P(X|H)\,P(H)}{P(X)} \tag{1}$$

Where,
X : Unspecified class data
H : Hypothesis data X is a specific class
$P(H|K)$: Probability of hypothesis H based on condition X (posterior probability)
$P(H)$: Probability of hypothesis H (prior probability)
$P(X|H)$: Probability X based on certain conditions
$P(X)$: Probability of X

Equation 1 can also be written in the form of Equation 2.

$$posterior \;=\; \frac{prior\ x\ likelihood}{evidence} \tag{2}$$

The posterior value that has been produced will be compared with the posterior value generated from other classes so that later the class of each sample classification can be determined.

F. Confusion Matrix

Confusion matrix is a method that is generally widely used by researchers in the process of calculating the level of accuracy of a classification process [21]. Confusion matrix is presented in the form of a table in which it contains various information in the form of values that are successfully predicted by the classification system [22]. An example of presenting values in the confusion matrix can be seen in **Table 1**.

Table 1. Confusion Matrix

	Prediction Class	
	True	***False***
True	True Positive (TP)	False Negative (FN)
False	False Positive (FP)	True Negative (TN)

The values produced by the confusion matrix are as follows.

1. Accuracy is the result of a comparison between the predicted value and the actual value obtained. The accuracy calculation formula is shown in Equation 3.
$$Accuracy \;=\; \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$
2. Precision is a calculation method used to find out how much comparison or accuracy is produced by the system in providing predictions on data sets from each class. The formula for calculating the precision value is shown in Equation 4.
$$Precision \;=\; \frac{TP}{FP + TP} \tag{4}$$
3. Recall is a calculation method in the confusion matrix used to determine the percentage of success of the system in providing predictions on the entire data. The formula for calculating the recall value is in Equation 5.
$$Recall \;=\; \frac{TP}{FN + TP} \tag{5}$$

G. Term Frequency-Inverse Document Frequency (TF-IDF)

Term Frequency-Inverse Document Frequency or commonly known as TF-IDF word weighting is one of the calculation methods used to determine the extent of the relationship of a word (term) to the document by giving a weight value to each word that appears [23]. The calculation of TF-IDF is shown in Equations 6, 7, and 8.

$$tf = 0.5 + 0.5 \times \frac{tf}{\max(tf)} \tag{6}$$

$$idf_t = log\left(\frac{D}{dft}\right) \tag{7}$$

$$W_{d.t} = tf_{d.t} \times idf_{d.t} \tag{8}$$

Where,

D : Document number – d
t : Term number – t of a document
W : Document weight number – d to term – t
Tf : Number of terms i in a document
idf : Inversed Document Frequency
df : Number of documents containing term i

Method

The research phase began with the process of collecting data from Twitter. The process of data collection or data crawling is carried out by querying the stubs "*anies baswedan* 2024", "*anies baswedan*" and "*anies presidential election*". The stages of research conducted in this study are shown in **Figure 1**.

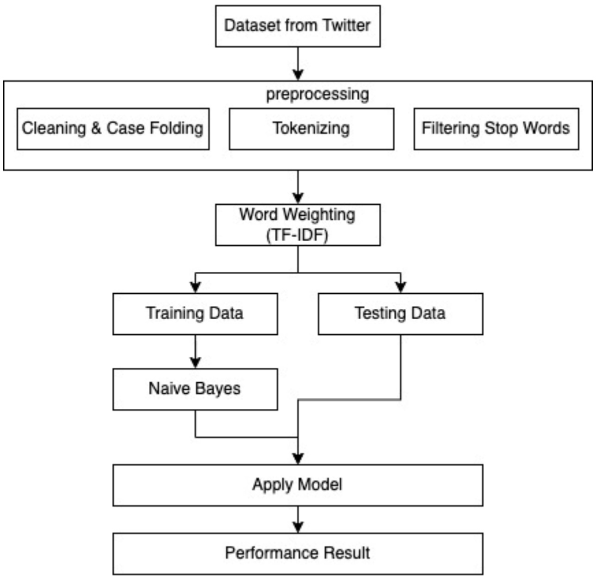


Figure 1. Stages of Research

After the data was obtained, the next stage was data pre-processing. At this stage there were several sub-processes, including cleaning data by removing special characters and punctuation marks that were not needed. Case folding converted all letters in a sentence into lowercase. Tokenizing is separating each word. Filtering is to remove unnecessary words using stop word lists of Indonesia. The next stage was grouping each tweets data or labeling data based on their respective labels which was done manually. The class or sentiment labels used in this study were classes labeled positive and classes labeled negative.

The next stage of the research was the weighting words or assigning values to each word in a document using the formula for calculating TF-IDF. The weighted data was divided into 2 parts, namely training data and test data. The application of the Naïve Bayes classifier method to the training data was used for the classification process in the testing process of the Naïve Bayes classifier model and a classification accuracy value would be obtained using the confusion matrix.

A. Data Collection

The data used in this study was a collection of tweets written or posted by Twitter users from October 2022 to January 2023 with the keywords "*anies baswedan* 2024", "*anies baswedan*" and "*anies presidential election*". The total data obtained was 7000 tweets written in Indonesian. The number of tweets that originally collected 7000 data was reduced to 1133 due to the process of removing duplicate data or data with two or more of the same sentences. **Table 2** is an example of the results of data collection.

Table 2. Data Collection Results

No	Samples of Tweets
1	*Cetak Sejarah, Anies Baswedan Diangkat Jadi Anggota Dewan Universitas Oxford* https://t.co/T4Bhs36mdN,"1614021608800399361"
2	*Dalam sejarahnya, ini pertama kali seorang berasal Indonesia diundang untuk menjadi anggota suatu dewan (board) di Universitas Oxford*. https://t.co/JQqW8o9FhI,"1614191858850336768"
3	*Anies Baswedan Diangkat jadi Anggota Dewan Universitas Oxford, Eks Gubernur DKI Jakarta Buat Ini di Instagram* https://t.co/Y6EaH5XxxZ,"1614036216730750977"
4	*RT @merahputihsuara: Komisi Pemberantasan Korupsi (KPK) akan mengusut dugaan korupsi Bantuan Sosial (Bansos) pandemi Covid-19 tahun 2020 sa…*,"1614692038733369344"
5	*RT @Muhammad2553: @alisyarief Untuk mendulang suara di Jawa Timur & Jawa Tengah, hanya pasangan; Anies R Baswedan + AHY yang kuat ...*,"1614691614638878721"
6	*@Uki23 https://t.co/puALApTe4A [Inilah 26 Prestasi/Penghargaan Gubernur DKI Jakarta Anies Baswedan - Fakta Kini]* https://t.co/HvxcrAySee,"1614691032884740096"
7	*RT @sutanmangara: Saatnya Indonesia Bangkit Bersama ANIES BASWEDAN !* https://t.co/11M8NOE0z9,"1614686795467292674"

B. Text Pre-processing

Text pre-processing is a series of processes of selecting data in text form to become a more structured set of data before the data enters the word weighting process using the TF-IDF calculation formula and the classification process using the Naïve Bayes classifier. The stages of text pre-processing that have been carried out are as follows.

1. Data Cleaning and Case Folding

   At this stage, all special characters in the text and punctuation marks in the text (**Table 3**) were removed and the text that originally used uppercase letters was changed to lowercase letters (**Table 4**).

Table 3. Punctuation Removal Results

No.	Before	After
1	*@AgusYudhoyono @PDemokrat Pak Anies Baswedan pilihan rakyat Indonesia pemimpin yang baik dan cerdas presiden di tahun 2024*	*Pak Anies Baswedan pilihan rakyat Indonesia pemimpin yang baik dan cerdas presiden di tahun 2024*
2	*@Relawananies Semoga presiden yang berikutnya pada tahun 2024 mengangkat Anies Baswedan sebagai menteri luar negeri.*	*Semoga presiden yang berikutnya pada tahun 2024 mengangkat Anies Baswedan sebagai menteri luar negeri*

Table 4. Case Folding Process Results

No.	Before	After
1	*Pak Anies Baswedan pilihan rakyat Indonesia pemimpin yang baik dan cerdas presiden di tahun 2024*	*pak anies baswedan pilihan rakyat indonesia pemimpin yang baik dan cerdas presiden di tahun 2024*
2	*Semoga presiden yang berikutnya pada tahun 2024 mengangkat Anies Baswedan sebagai menteri luar negeri*	*semoga presiden yang berikutnya pada tahun 2024 mengangkat anies baswedan sebagai menteri luar negeri*

2. Tokenizing

   Tokenizing is a stage where every word contained in a sentence will be separated based on a predetermined space. **Table 5** represents the results of the tokenizing process.

Table 5. Tokenizing Processing Results

No.	Before	After
1.	*pak anies baswedan pilihan rakyat indonesia pemimpin yang baik dan cerdas presiden di tahun 2024*	*|anies| |baswedan| |pilihan| |rakyat| |Indonesia| |pemimpin| |yang| |baik| |cerdas| |presiden| |tahun|*
2.	*semoga presiden yang berikutnya pada tahun 2024 mengangkat anies baswedan sebagai menteri luar negeri*	*|semoga| |presiden| |yang| |berikutnya| |pada| |tahun| |mengangkat| |anies| |baswedan| |sebagai| |menteri| |luar| |negeri|*

3. Filtering

This stage is carried out the removal of unnecessary words using the list stop words of Indonesia. **Table 6** is the processing results of filtering.

Table 6. Processing Results of *Filtering*

No.	Before	After
1.	*|anies| |baswedan| |pilihan| |rakyat| |Indonesia| |pemimpin| |yang| |baik| |cerdas| |presiden| |tahun|*	*|anies| |baswedan| |pilihan| |rakyat| |Indonesia| |pemimpin| |cerdas| |presiden|*
2.	*|semoga| |presiden| |yang| |berikutnya| |pada| |tahun| |mengangkat| |anies| |baswedan| |sebagai| |menteri| |luar| |negeri|*	*|semoga| |presiden| |mengangkat| |anies| |baswedan| |menteri| |negeri|*

C. Word Visualization

WordCloud is one of the methods in RapidMiner software aims to provide an overview or visualization of a data in the form of text interestingly and informatively by providing a display of a list of words whose size depends on the frequency of the appearance of the word in the tweets data [24].

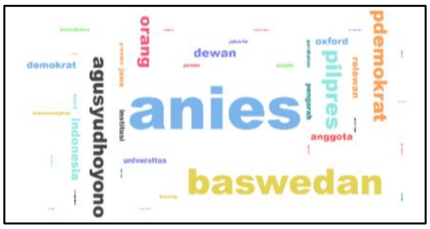


Figure 2. *WordCloud* Before *Pre-processing*

Figure 2 is the result of wordcloud visualization of the top 40 words that most often appear in a collection of tweets discussing *Anies Baswedan* as a 2024 presidential candidate before pre-processing text. In WordCloud, it shows that the most frequently appearing words are "*anies*" with a frequency of occurrence of 2159 times and "*baswedan*" with a occurrence frequency 1363 times. **Figure 3** is the result of WordCloud visualization of the top 40 words that appear most often after text pre-processing. In WordCloud, it can be seen that the words "*anies*" and "*baswedan*" are still the most frequently appearing words in the tweets with an occurrence frequency 675 and 364 times.

Figure 3. *WordCloud* After *Pre-processing*

D. Data Labeling

Data labeling is one of research stages where the data of tweets going through preprocessing would be grouped based on their respective labels manually. The data labeling stage was done manually based on personal opinions by looking at the meaning of words that have negative or positive connotations in tweets. Labeling data was conducted for the learning process and training the model to understand the pattern or characteristics of each data. An example of data labeling results is shown in **Table 7**.

Table 7. Example of Data Label

No	Tweets	Label
1	*pemimpin berprestasi kerjanya dihargai diakui nasional internasional anies baswedan presiden*	Positive
2	*pdip menang capres menghentikan langkahnya anies baswedan presiden megawati soekarnoputri maju puan ganjar berat*	Negative

E. Term Frequency Inverse Document Frequency

Term Frequency-Inverse Document Frequency (*TF-IDF*) is one stage that aims to determine the extent of a word's relationship to the document by giving a weight value to each word that appears in the document. **Table 8** is an example of four documents derived from previously obtained tweets data. These four documents would be assigned a weight value to each word that appeared. The TF-IDF calculation process was carried out in a Jupyter notebook using the Python programming language.

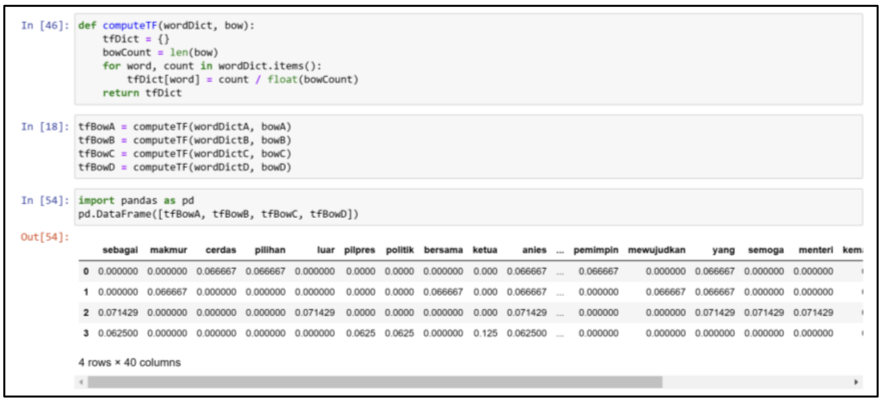
Table 8. Document Group

Document Group	Tweets
Doc A	*pak anies baswedan pilihan rakyat indonesia pemimpin yang baik dan cerdas presiden di tahun 2024*
Doc B	*mewujudkan masyarakat yang adil dalam kemakmuran makmur dalam keadilan bersama anies rasyid baswedan pemilu 2024*
Doc C	*semoga presiden yang berikutnya pada tahun 2024 mengangkat anies baswedan sebagai menteri luar negeri*
Doc D	*ketua dmi dan wakil ketua dmi manfaatkan dmi sebagai kendaraan politik anies baswedan di pilpres 2024*

The following are the steps of calculating word weight in four documents that have been determined using *TF-IDF*.

1. Term Frequency

The process of calculating word weight is carried out in *a* jupyter notebook using the python programming language. The display of the source code used in the process of calculating term frequency along with the results obtained is shown in **Figure 4**.



```
In [46]: def computeTF(wordDict, bow):
             tfDict = {}
             bowCount = len(bow)
             for word, count in wordDict.items():
                 tfDict[word] = count / float(bowCount)
             return tfDict

In [18]: tfBowA = computeTF(wordDictA, bowA)
         tfBowB = computeTF(wordDictB, bowB)
         tfBowC = computeTF(wordDictC, bowC)
         tfBowD = computeTF(wordDictD, bowD)

In [54]: import pandas as pd
         pd.DataFrame([tfBowA, tfBowB, tfBowC, tfBowD])

Out[54]:
```

	sebagai	makmur	cerdas	pilihan	luar	pilpres	politik	bersama	ketua	anies	...	pemimpin	mewujudkan	yang	semoga	menteri
0	0.000000	0.000000	0.066667	0.066667	0.000000	0.0000	0.0000	0.000000	0.000	0.066667	...	0.066667	0.000000	0.066667	0.000000	0.000000
1	0.000000	0.066667	0.000000	0.000000	0.000000	0.0000	0.0000	0.066667	0.000	0.066667	...	0.000000	0.066667	0.066667	0.000000	0.000000
2	0.071429	0.000000	0.000000	0.000000	0.071429	0.0000	0.0000	0.000000	0.000	0.071429	...	0.000000	0.000000	0.071429	0.071429	0.071429
3	0.062500	0.000000	0.000000	0.000000	0.000000	0.0625	0.0625	0.000000	0.125	0.062500	...	0.000000	0.000000	0.000000	0.000000	0.000000

4 rows × 40 columns

Figure 4. Term Frequency Calculation Results

2. Inverse Document Frequency

The inverse document frequency or IDF calculation stage aims to provide a description of the contribution of each word into a predetermined document. The display of the source code used in the process of calculating the inverse document frequency along with the results obtained is shown in **Figure 5**.

```
In [21]: def computeIDF(docList):
             import math
             idfDict = {}
             N = len(docList)

             idfDict = dict.fromkeys(docList[0].keys(), 0)
             for doc in docList:
                 for word, val in doc.items():
                     if val > 0:
                         idfDict[word] +=1

             for word, val in idfDict.items():
                 idfDict[word]=math.log(N / float(val))

             return idfDict

In [34]: idfs = computeIDF([wordDictA, wordDictB, wordDictC, wordDictD])

         idfs

Out[34]: {'sebagai': 0.6931471805599453,
          'makmur': 1.3862943611198906,
          'cerdas': 1.3862943611198906,
          'pilihan': 1.3862943611198906,
          'luar': 1.3862943611198906,
          'pilpres': 1.3862943611198906,
          'politik': 1.3862943611198906,
          'bersama': 1.3862943611198906,
          'ketua': 1.3862943611198906,
          'anies': 0.0,
          'baik': 1.3862943611198906,
          'presiden': 0.6931471805599453,
          'rakyat': 1.3862943611198906,
          'indonesia': 1.3862943611198906,
          'masyarakat': 1.3862943611198906,
          'tahun': 0.6931471805599453,
          'pak': 1.3862943611198906,
```

Figure 5. IDF Calculation Results

3. Weighting of TF-IDF Results

This word weighting stage is a combination of the results of calculating TF-IDF aiming to find out the overview of value from the data set. The display of the source code used in the process of calculating word weight along with the results obtained is shown in **Figure 6**.

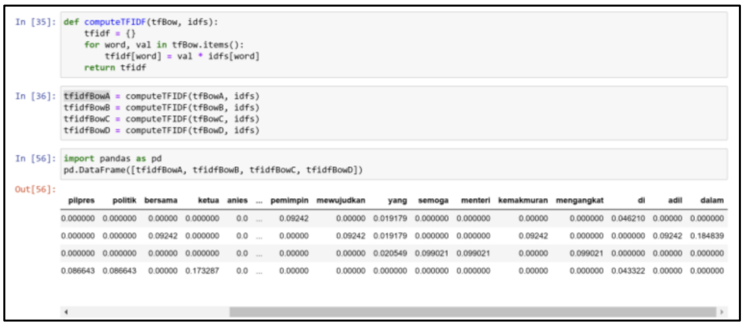


Figure 6. TF-IDF word weighting results

The calculation results obtained from the process of calculating TF, IDF, and word weights by integrating TF-IDF are presented in table form which can be seen in **Table 9**, **10**, **11** and **12**.

Table 9. Calculation of TF-IDF Value in Document "A"

Words	Word Appearance Frequency				DF	TF	IDF	Weight (W)
	A	*B*	*C*	*D*		*A*	*A*	*A*
anies	1	1	1	1	4	0.067	0.0	0.0
baswedan	1	1	1	1	4	0.067	0.0	0.0
presiden	1	0	1	0	2	0.067	0.693	0.046
ketua	0	0	0	2	1	0.0	1.386	0.0
adil	0	1	0	0	1	0.0	1.386	0.0
cerdas	1	0	0	0	1	0.067	1.386	0.092
indonesia	1	0	0	0	1	0.067	1.386	0.092
keadilan	0	1	0	0	1	0.0	1.386	0.0
kemakmuran	0	1	0	0	1	0.0	1.386	0.0
kendaraan	0	0	0	1	1	0.0	1.386	0.0
makmur	0	1	0	0	1	0.0	1.386	0.0
manfaatkan	0	0	0	1	1	0.0	1.386	0.0
masyarakat	0	1	0	0	1	0.0	1.386	0.0
mengangkat	0	0	1	0	1	0.0	1.386	0.0
menteri	0	0	1	0	1	0.0	1.386	0.0
mewujudkan	0	1	0	0	1	0.0	1.386	0.0
negeri	0	0	1	0	1	0.0	1.386	0.0
pemilu	0	1	0	0	1	0.0	1.386	0.0
pemimpin	1	0	0	0	1	0.067	1.386	0.092
pilihan	1	0	0	0	1	0.067	1.386	0.092
pilpres	0	0	0	1	1	0.0	1.386	0.0
politik	0	0	0	1	1	0.0	1.386	0.0
rakyat	1	0	0	0	1	0.067	1.386	0.092
rasyid	0	1	0	0	1	0.0	1.386	0.0
semoga	0	0	1	0	1	0.0	1.386	0.0
wakil	0	0	0	1	1	0.0	1.386	0.0

Table 10. Calculation of TF-IDF Value in Document "B"

Words	Word Appearance Frequency				DF	TF	IDF	Weight (W)
	A	*B*	*C*	*D*		*B*	*B*	*B*
anies	1	1	1	1	4	0.067	0.0	0.0
baswedan	1	1	1	1	4	0.067	0.0	0.0
presiden	1	0	1	0	2	0.0	0.693	0.0
ketua	0	0	0	2	1	0.0	1.386	0.0
adil	0	1	0	0	1	0.067	1.386	0.092
cerdas	1	0	0	0	1	0.0	1.386	0.0

Words	Word Appearance Frequency				DF	TF	IDF	Weight (W)
	A	*B*	*C*	*D*		*B*	*B*	*B*
indonesia	1	0	0	0	1	0.0	1.386	0.0
keadilan	0	1	0	0	1	0.067	1.386	0.092
kemakmuran	0	1	0	0	1	0.067	1.386	0.092
kendaraan	0	0	0	1	1	0.0	1.386	0.0
makmur	0	1	0	0	1	0.067	1.386	0.092
manfaatkan	0	0	0	1	1	0.0	1.386	0.0
masyarakat	0	1	0	0	1	0.067	1.386	0.092
mengangkat	0	0	1	0	1	0.0	1.386	0.0
menteri	0	0	1	0	1	0.0	1.386	0.0
mewujudkan	0	1	0	0	1	0.067	1.386	0.092
negeri	0	0	1	0	1	0.067	1.386	0.0
pemilu	0	1	0	0	1	0.067	1.386	0.092
pemimpin	1	0	0	0	1	0.0	1.386	0.0
pilihan	1	0	0	0	1	0.0	1.386	0.0
pilpres	0	0	0	1	1	0.0	1.386	0.0
politik	0	0	0	1	1	0.0	1.386	0.0
rakyat	1	0	0	0	1	0.0	1.386	0.0
rasyid	0	1	0	0	1	0.067	1.386	0.092
semoga	0	0	1	0	1	0.0	1.386	0.0
wakil	0	0	0	1	1	0.0	1.386	0.0

Table 11. Calculation of TF-IDF Value in Document "C"

Words	Word Appearance Frequency				DF	TF	IDF	Weight (W)
	A	*B*	*C*	*D*		*B*	*B*	*B*
anies	1	1	1	1	4	0.714	0.0	0.0
baswedan	1	1	1	1	4	0.714	0.0	0.0
presiden	1	0	1	0	2	0.714	0.693	0.049
ketua	0	0	0	2	1	0.0	1.386	0.0
adil	0	1	0	0	1	0.0	1.386	0.0
cerdas	1	0	0	0	1	0.0	1.386	0.0
indonesia	1	0	0	0	1	0.0	1.386	0.0
keadilan	0	1	0	0	1	0.0	1.386	0.0
kemakmuran	0	1	0	0	1	0.0	1.386	0.0
kendaraan	0	0	0	1	1	0.0	1.386	0.0
makmur	0	1	0	0	1	0.0	1.386	0.0
manfaatkan	0	0	0	1	1	0.0	1.386	0.0
masyarakat	0	1	0	0	1	0.0	1.386	0.0
mengangkat	0	0	1	0	1	0.714	1.386	0.099
menteri	0	0	1	0	1	0.714	1.386	0.099
mewujudkan	0	1	0	0	1	0.0	1.386	0.0
negeri	0	0	1	0	1	0.714	1.386	0.099
pemilu	0	1	0	0	1	0.0	1.386	0.0
pemimpin	1	0	0	0	1	0.0	1.386	0.0
pilihan	1	0	0	0	1	0.0	1.386	0.0
pilpres	0	0	0	1	1	0.0	1.386	0.0
politik	0	0	0	1	1	0.0	1.386	0.0
rakyat	1	0	0	0	1	0.0	1.386	0.0
rasyid	0	1	0	0	1	0.0	1.386	0.0
semoga	0	0	1	0	1	0.714	1.386	0.099
wakil	0	0	0	1	1	0.0	1.386	0.0

Table 12. Calculation of TF-IDF Value in Document "D"

Words	Word Appearance Frequency				DF	TF	IDF	Weight (W)
	A	*B*	*C*	*D*		*D*	*D*	*D*
anies	1	1	1	1	4	0.0625	0.0	0.0
baswedan	1	1	1	1	4	0.0625	0.0	0.0
presiden	1	0	1	0	2	0.0	0.693	0.0
ketua	0	0	0	2	1	0.125	1.386	0.173
adil	0	1	0	0	1	0.0	1.386	0.0
cerdas	1	0	0	0	1	0.0	1.386	0.0
indonesia	1	0	0	0	1	0.0	1.386	0.0
keadilan	0	1	0	0	1	0.0	1.386	0.0
kemakmuran	0	1	0	0	1	0.0	1.386	0.0
kendaraan	0	0	0	1	1	0.0625	1.386	0.086
makmur	0	1	0	0	1	0.0	1.386	0.0
manfaatkan	0	0	0	1	1	0.0625	1.386	0.086
masyarakat	0	1	0	0	1	0.0	1.386	0.0
mengangkat	0	0	1	0	1	0.0	1.386	0.0
menteri	0	0	1	0	1	0.0	1.386	0.0
mewujudkan	0	1	0	0	1	0.0	1.386	0.0
negeri	0	0	1	0	1	0.0	1.386	0.0
pemilu	0	1	0	0	1	0.0	1.386	0.0
pemimpin	1	0	0	0	1	0.0	1.386	0.0
pilihan	1	0	0	0	1	0.0	1.386	0.0
pilpres	0	0	0	1	1	0.0625	1.386	0.086
politik	0	0	0	1	1	0.0625	1.386	0.086
rakyat	1	0	0	0	1	0,0	1.386	0.0
rasyid	0	1	0	0	1	0.0	1.386	0.0
semoga	0	0	1	0	1	0.0	1.386	0.0
wakil	0	0	0	1	1	0.0625	1.386	0.086

F. Naïve Bayes

Multinomial Naïve Bayes is one of a series of text classification processes owned by Naïve Bayes by presenting a calculation method to determine the frequency of each word that appears in a document [25]. Table 13 is a collection of documents that have been assigned a weight value for each word that appears using the TF-IDF calculation method with the total weight in positive class documents is $W(+) = 1{,}242$; and the total number of Ws in negative class documents is $W(-) = 1{,}048$; and the total number of idf in all classes is $B' = 32{,}571$.

Table 13. Word Weight Table

Words	In	
	Positive (c1)	*Negative (c2)*
anies	0	0
baswedan	0	0
presiden	0.046	0.049
ketua	0	0.173
adil	0.092	0
cerdas	0.092	0
indonesia	0.092	0
keadilan	0.092	0
kemakmuran	0.092	0
kendaraan	0	0.086
makmur	0.092	0
manfaatkan	0	0.086
masyarakat	0.092	0
mengangkat	0	0.099
menteri	0	0.099
mewujudkan	0.092	0

Words	**In**	
	Positive (c1)	***Negative (c2)***
negeri	0	0.099
pemilu	0.092	0
pemimpin	0.092	0
pilihan	0.092	0
pilpres	0	0.086
politik	0	0.086
rakyat	0.092	0
rasyid	0.092	0
semoga	0	0.099
wakil	0	0.086

Based on the results of the weight calculation obtained in **Table 13**, the prior probability is calculated to obtain the probability of a class appearing using Equation 9.

$$P(c) = \frac{N_c}{N_{doc}} \tag{9}$$

Where,
c : Category or class of a document
doc : Document
N_c : Number of category c in the entire document
N_{doc} : Total number of documents

From Equation 9 found results as in **Table 14**.

Table 14. Prior Probability Calculation Results

P(c1)	**P(c2)**
$P(c1) = \frac{2}{4} = 0.5$	$P(c2) = \frac{2}{4} = 0.5$

After getting the results of the probability prior, proceed to calculate the likelihood of each word to find out whether a word belongs to a certain category or class. The probability calculation can be done using Equation 10.

$$P(w|c) = \frac{Wct + 1}{(\sum w' \in VW\, ct') + B'} \tag{10}$$

Where,
Wct : Weight of term t in category c
$\sum w' \in VW\, ct'$: Total number of W within category c
B' : Total number of idf values in all documents

From Equation 10, the results of the probability calculation are obtained as in **Table 15**.

Table 15. Likelihood Calculation Results

Words	$\boldsymbol{P(w\|c) = \frac{Wct + 1}{(\sum w' \in VW\, ct') + B'}}$	
	Positive	***Negative***
anies	0.030	0.030
baswedan	0.030	0.030
presiden	0.076	0.079
ketua	0.030	0.203
adil	0.122	0.030
cerdas	0.122	0.030
indonesia	0.122	0.030
keadilan	0.122	0.030
kemakmuran	0.122	0.030
kendaraan	0.030	0.116
makmur	0.122	0.030
manfaatkan	0.030	0.116
masyarakat	0.122	0.030
mengangkat	0.030	0.129
menteri	0.030	0.129

Words	$P(w\|c) = \frac{Wct + 1}{(\sum w' \in VW\, ct') + B'}$	
	Positive	***Negative***
mewujudkan	0.122	0.030
negeri	0.030	0.129
pemilu	0.122	0.030
pemimpin	0.122	0.030
pilihan	0.122	0.030
pilpres	0.030	0.116
politik	0.030	0.116
rakyat	0.122	0.030
rasyid	0.122	0.030
semoga	0.030	0.129
wakil	0.030	0.116

The next stage of calculation is to determine the document's chances of the sentiment class by comparing the values contained in the positive class and the values contained in the negative class with the results as shown in **Table 16**.

Table 16. Document Probability Calculation Results

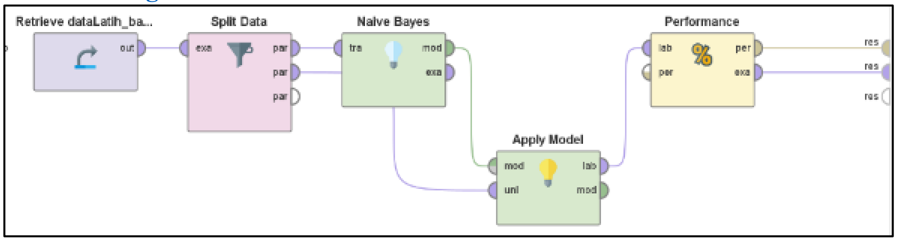
anies baswedan pilihan rakyat indonesia pemimpin cerdas presiden	
$P(positive\ (c1), doc\ A)$	$P(negative\ (c1), doc\ A)$
P(c1) × P(*anies*) × P(*baswedan*) × P(*pilihan*) × P(*rakyat*) × P(*indonesia*) × P(*pemimpin*) × P(*cerdas*) × P(*presiden*) $= 0.5 \times 0.030 \times 0.030 \times 0.122 \times 0.122 \times 0.122 \times 0.122 \times 0.122 \times 0.076 = 9.243 \times 10^{-10}$	P(c1) × P(*anies*) × P(*baswedan*) × P(*pilihan*) × P(*rakyat*) × P(*indonesia*) × P(*pemimpin*) × P(*cerdas*) ×P(*presiden*) $= 0.5 \times 0.030 \times 0.030 \times 0.030 \times 0.030 \times 0.030 \times 0.030 \times 0.030 \times 0.079 = 8.638 \times 10^{-13}$

From the calculation of document probability in **Table 13**, it is found that the probability of document A against the positive sentiment class have a greater value, which is 9.243×10^{-10} compared to the negative sentiment class document, which is 8.638×10^{-13}. Therefore, it can be concluded that the sample data of document A is classified as a positive sentiment.

Results and Discussion

A. Classification Algorithm Modeling

In this study, the RapidMiner application became one of the software used in making the Naïve Bayes Classifier model, because the RapidMiner application is able to provide an effective and efficient experience starting from the process of retrieving data or information needed to data sentiment analysis. The Naïve Bayes classifier model that has been created can be seen in **Figure 7**.



Gambar 7. *Cross Validation Operator*

Cross validation operator is a collection of RapidMiner operators, such as retrieve, split data, Naïve Bayes, apply model, and performance that are used with the aim of determining the accuracy results of a data where each data has been given a sentiment or label in the previous stage.

B. Model Testing

In this study, three data testing processes were carried out using the confuse matrix method contained in RapidMiner. The dataset used in this study has 585 tweets belonging to the positive label class and 548 tweets with the negative label class, therefore, the total tweets data as a whole is 1133 tweets which can be seen in **Figure 8**.

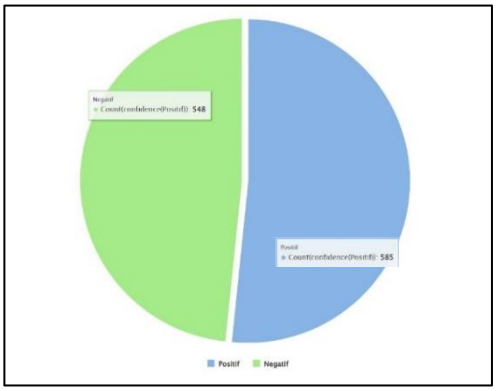


Figure 8. Proportion of Positive and Negative Sentiment Analysis

The process of testing the Naïve Bayes classifier algorithm was carried out three times where in the first test the ratio used was 0.6 or 60% for training data and 0.4 or 40% for test data. Then the second test using a ratio of 0.7 or 70% training data and 0.3 or 30% test data. In the third test using a ratio of 0.8 or 80% training data and 0.2 or 20% test data. The process of testing the Naïve Bayes classifier model produced sentiment accuracy values from each tweets data. This research is limited to the nomination of *Anies Baswedan* as a presidential candidate before having vice president. Data was taken from October 2022 to January 2023. Further research is recommended to conduct sentiment analysis on the pairs of presidential and vice-presidential candidates who have been officially registered with the General Election Commission using various other classification algorithms.

Model 1 Testing

The first Naïve Bayes classifier model test used a ratio of 60% for the training data and 40% for the test data. The accuracy value obtained was 73.95%. The classification in this first test succeeded in producing 2,209 attributes derived from the results of weighting 1.133 tweets with 585 labeled as positive and 548 labeled as negative. The overall accuracy results can be seen in **Table 17**.

Table 17. Naive Bayes 1st *Model Test Results*

	True Positive	**True Negative**	**Class Prediction**
Predicted Positive	156	40	79.59%
Predicted Negative	78	179	69.65%
Class Recall	66.67%	81.74%	

Model 2 Testing

The second *Naïve Bayes classifier* model test used a ratio of 70% for the trainer data and 30% for the test data. After the system applied the model to the test data, the overall accuracy level results can be seen in **Table 18**, where the accuracy value result was 74.63%. The attributes produced in this second test were the same as the previous test, which was 2,209 attributes derived from the results of weighting 1133 *tweets* with 585 positive and 548 negative labeled data.

Table 18. *Naive Bayes* 2nd Model Test Results

	True Positive	**True Negative**	**Class Prediction**
Predicted Positive	112	23	82.96%
Predicted Negative	63	141	69.12%
Class Recall	64%	85.98%	

Model 3 Testing

In testing the Naïve Bayes model, the third classifier used a ratio of 80% for the training data and 20% for the test data. The overall accuracy rate results can be seen in **Table 19**, where the accuracy value result was 76.21%.

Table 19. Naïve Bayes 3rd Model Test Results

	True Positive	**True Negative**	**Class Prediction**
Predicted Positive	75	12	86.21%
Predicted Negative	42	98	70%

	True Positive	True Negative	Class Prediction
Class Recall	64.10%	89.09%	

Based on the results of the three model testing processes that have been carried out, the obtained average accuracy results were shown in **Table 20**.

Table 20. Test Results

No.	Data Split Ratio	Accuracy
1.	60% : 40%	73.95%
2.	70% : 30%	74.63%
3.	80% : 20%	76.21%
Average		**74.93%**

From **Table 20**, it indicated that trials have been carried out on the Naïve Bayes classification model on 1133 data using a data sharing ratio of 80% for training data and 20% for test data resulted in a higher accuracy value among others. It was 76.21% with a precision class recall value produced by true positive of 64.10%, and class recall at true negative of 89.09%.

Conclusion

The sentiment analysis conducted in this study took tweets data from October 2022 to January 2023 totaling 7000 tweets in Indonesian and carried out a process of removing duplicate data or data that has two or more of the same sentences. It resulted in 1133 tweets to be used to conduct sentiment analysis. Three test models were carried out with differences in the division of the amount of training data and test data, respectively 60-40%, 70-30% and 80-20%. The highest level of accuracy was in the 3rd model test with the use of training data and test data of 80-20% with an accuracy value of 76.21% and an average test accuracy value of the three models was 74.93%.

For further research, it is recommended to increase the number of datasets with a longer time span and be able to conduct sentiment analysis on the pair of presidential candidates *Anies Baswedan* and *Muhaimin Iskandar* or other presidential candidate pairs using various types of classification algorithms such as Random Forest algorithms, Neural Networks, K-Nearest Neighbors and other classification algorithms.

References

[1] N. Saputra, K. Nurbagja, and T. Turiyan, “Sentiment Analysis of Presidential Candidates Anies Baswedan and Ganjar Pranowo Using Naïve Bayes Method,” J. Sisfotek. Glob, vol. 12, no. 2, pp. 114-119, 2022.

[2] A. Rivaldy, H.A.F. Wowor, S.R. Maisya, D. Safitri, "The Use of Twitter in Increasing Political Literacy of Communication Science Students of Universitas Negeri Jakarta," J. Ilmu. Com. Pol. and Kom. Bis, vol. 5, no. 1, pp. 41-48, 2021.

[3] APJII. Profil Internet Indonesia 2022 [online]. Available: https://apjii.or.id/survei/surveiprofilinternetindonesia2022-21072047.

[4] M. Siahaan, “An Analysis of Contract Employee Performance Assessment Using Machine Learning,” J. of Inf. and Tel. Eng, vol. 5, no. 1, pp. 121-131, 2021.

[5] I. Irwanto, C. Widodo, A. Hasanah, P.A.D. Kusumah, Kusrini, and Kusnawi, “Sentiment Analysis and Classification of Forest Fires in Indonesia,” ILKOM J. Ilmiah, vol. 15, no. 1, pp. 175-185, 2023.

[6] M. Idris, M, "Implementation of Data Mining with Naïve Bayes Algorithm to Predict Birth Rate," J. Pel. Info, vol. 7, no. 3, pp. 421-428, 2019.

[7] N. Kalcheva, M. Todorova, and G. Marinova, “Naive Bayes Classifier, Decision Tree - Advantages and Disadvantages,” 6th International Scientific Conference ERAZ 2020 Conference Proceedings, pp. 153-157, 2020.

[8] R. Ardiansyah, "Sentiment Analysis of Presidential and Vice Presidential Candidates for the 2019-2024 Period After the Presidential Election Debate on Twitter," J. ScientiCO: Comp. Sci and Inf, vol. 2, no. 1, pp. 21-28, 2019.

[9] P. Arsi, B.A Kusuma, and A. Nurhakim, "Capital Sentiment Analysis Based on Naive Bayes Classifier," J. Inf. Upgris, vol. 7, no. 1, 2021.

[10] M.H. Al-Areef, K. Saputra, "Analysis of Twitter User Sentiment Regarding Indonesia's 2024 Presidential Candidate Using LSTM Algorithm," J. Sains. Manaj. Info. and Komp, vol. 22, no. 2, pp. 270-279, 2023.

[11] A.P. Nardilasari, A.L. Hananto, S. Hilabi, Tukino, B. Priyatna, "Sentiment Analysis of 2024 Presidential Candidates Using SVM Algorithm on Twitter Social Media," J. of Info. Tech. and Comp Sci, vol. 8, no. 1, pp. 11-18, 2023.

[12] F. Al Isfahani, R. Mubarok, "Analysis of Twitter User Sentiment on the Policy of Large-Scale Social Restrictions (PSBB) with the Naïve Bayes Method," J. Sili. Series. Science and Tech, vol. 7, no. 1, pp. 19-24, 2021.

[13] R.S Putra, W. Agustin, M.K Anam, Lusiana, S. Yaakub, “The Application of Naïve Bayes Classifier Based Feature Selection on Analysis of Online Learning Sentiment in Online Media,” J. Transfor, vol. 20, no.1, pp. 44-56, 2022.

[14] H. Hassani, C. Beneki, S. Unger, M.T Mazinani, M.R. Yeganegi, “Text Mining in Big Data Analytics,” J. Big Data and Cog. Comp, vol. 4, no. 1, 2020.

[15] H. N. Alhazmi, “Text Mining in Online Social Networks: A Systematic Review,” Int. J. of Comp. Sci. and Net. Sec, vol. 22, no. 3, pp. 396-404, 2022.

[16] R. Cheng, X. Kong, M. Yu, N. Wang, “A Classification Algorithm: Data Mining and Mathematical Model,” J. Phys. Conf. Ser. 2021. doi:10.1088/1742-6596/2068/1/012012

[17] H. Sujadi, "Analysis of Twitter Social Media User Sentiment on the Covid-19 Outbreak with Naive Bayes Classifier and Support Vector Machine Methods," J. Infotech, vol. 8, no. 1, pp. 22–27, 2022.

[18] D. Priyanto, A.R. Iman, D. Jollyta, “Naïve Bayes and K-Nearest Neighbor Approaches in Data Mining Classification of Drugs Addictive Diseases,” ILKOM J. Ilmiah, vol. 15, no. 2, pp. 262-270, 2023.

[19] D. Indriani, "Sentiment Analysis on Tweets with the Hashtag# kpujangancheat Using the Naive Bayes Method," Ph.D. dissertation, Universitas Islam Riau, Riau., Indonesia, 2019.

[20] M.R.F. Sya'bani, U. Enri, and T. N. Padilah, "Sentiment Analysis of 2024 Presidential Candidates with Naïve Bayes Algorithm," J. Ris. Kom, vol. 9, no. 2, pp. 265-273, 2022.

[21] R.C. Chen, C. Dewi, S.W. Huang, and R.E. Caraka, “Selecting critical features for data classification based on machine learning methods,” J. Big Data, vol. 7, no. 52. 2020. doi.org/10.1186/s40537-020-00327-4

[22] B. Gunawan, H. Sastypratiwi, and E.E. Pratama, "Sentiment Analysis System in Product Reviews Using Naive Bayes Method," J. Edu. and Pen. Info, vol. 4, no. 2, pp. 113-118, 2018.

[23] R. Ramadhan, Y.A. Sari, P.P. Adikara, "Comparison of Term Frequency-Inverse Document Frequency and Term Frequency-Relevance Frequency Weighting to N-Gram Features in Sentiment Analysis," J. Peng. Tech. Info. and Science Comp, vol. 5, no. 11, pp. 5075-5079, 2021.

[24] M. G. Pradana, "Use of Wordcloud and Document Term Matrix Features in Text Mining," J. Scientific. Info, vol. 8, no. 1, pp. 38-43, 2020.

[25] A. H. Rahayu and A. Sudrajat. “Naïve Bayes Performance in Analysis of Public Opinion Sentiment Against COVID-19,” J. of Appl. Intell. Sys, vol. 7, no. 3, pp. 237-245, 2022.