



Ref : 156/UMA/JITE/X/2024

Medan, 14 October 2024

Subject : Letter of Acceptance

To :

Mr./Mrs. **Ridho Ahmad Fikri**

Assalamu'alaikum Wr. Wb

We would like to express our sincere gratitude for your participation in submitting an article to the Journal of Informatics and Telecommunication Engineering (JITE). We hereby inform you that the article listed below:

Paper : Sentimen Terhadap Duta Sopan Santun Di Media Sosial: Studi Kasus Dengan Metode Naive Bayes

Author : Ridho Ahmad Fikri & Rizki Muliono

Based on the review results, we are pleased to inform you that your submitted article has been ACCEPTED for publication in JITE Journal - **Special Issues 2024: Innovations in Predictive Analytics and Sentiment Analysis - Applications in Education, Healthcare, and Social Media**, ISSN: 2549-6247 (Print) ISSN: 2549-6255 (Online).

We would like to thank you for your attention and cooperation.

Wassalamu'alaikum, Wr.Wb.

Best Regards,



Muhathir, ST., M.Kom
Chief Editor

JITE (Journal of Informatics and Telecommunication Engineering)

Available online <http://ojs.uma.ac.id/index.php/jite> DOI: 10.31289/jite.v8i1.11639



Received: 25-March-2024

Accepted: 26-July-2024

Published: 31-July-2024

Sentimen Terhadap Duta Sopan Santun di Media Sosial: Studi Kasus dengan Metode Naive Bayes

Ridho Ahmad Fikri1)*

1) Program Studi Teknik Informatika, Universitas Medan Area, Indonesia

*Corresponding Email: rizkimuliono@gmail.com

Abstrak

Media sosial telah memberikan pengaruh besar terhadap kehidupan masyarakat modern, menjadi platform utama untuk berbagi informasi dan pendapat. Salah satu fenomena yang menarik perhatian adalah kasus viral seorang polisi wanita, Putri Cikita, yang mendapat julukan "Duta Sopan Santun" karena aksinya dalam sebuah video. Penelitian ini bertujuan untuk menganalisis sentimen publik terhadap kasus tersebut di media sosial Twitter menggunakan metode Naive Bayes Classifier (NBC). Metode penelitian ini adalah kuantitatif deskriptif dengan analisis sentimen berbasis Text Mining menggunakan Python dan Google Colab. Data yang digunakan adalah 2000 tweet berbahasa Indonesia yang diambil dari Agustus hingga November 2024 dengan kata kunci "Duta Sopan Santun" dan "Putri Cikita". Tahapan penelitian mencakup pengumpulan data, praproses data (case folding, tokenizing, filtering, stemming), dan pelabelan sentimen dengan kelas positif, negatif, dan netral. Hasil analisis menunjukkan 11,55% tweet bersentimen positif, 68,40% netral, dan 20,05% negatif. Metode Naive Bayes terbukti efektif dalam mengklasifikasikan sentimen data teks. Penelitian ini memberikan wawasan tentang persepsi masyarakat terhadap peristiwa viral dan menyarankan pentingnya pengelolaan citra publik di era digital.

Kata Kunci: analisis sentimen, media sosial, Naive Bayes Classifier, Duta Sopan Santun, Text Mining.

Abstract

Social media has had a significant impact on modern society, serving as a primary platform for sharing information and opinions. One intriguing phenomenon is the viral case of a female police officer, Putri Cikita, who earned the title "Ambassador of Courtesy" due to her actions in a video. This study aims to analyze public sentiment regarding this case on Twitter using the Naive Bayes Classifier (NBC) method. The research adopts a quantitative descriptive approach with sentiment analysis based on Text Mining, utilizing Python and Google Colab. The dataset consists of 2,000 Indonesian-language tweets collected from August to November 2024 using the keywords "Ambassador of Courtesy" and "Putri Cikita." The research stages include data collection, data preprocessing (case folding, tokenizing, filtering, stemming), and sentiment labeling into positive, negative, and neutral classes. The analysis results reveal that 11.55% of tweets express positive sentiment, 68.40% are neutral, and 20.05% are negative. The Naive Bayes method proves effective in classifying textual sentiment data. This research provides insights into public perceptions of viral events and underscores the importance of public image management in the digital era.

Keywords: sentiment analysis, social media, Naive Bayes Classifier, Ambassador of Courtesy, Text Mining..

How to Cite: Fikri, R.A. (2024). Sentimen Terhadap Duta Sopan Santun di Media Sosial: Studi Kasus dengan Metode Naive Bayes. *JITE (Journal of Informatics And Telecommunication Engineering)*, 8(1), 43-55.

I. PENDAHULUAN

Perkembangan terus menerus dari Teknologi Informasi, terutama salah satunya melalui Media Sosial, dapat di rasakan telah memberikan pengaruh yang sangat besar terhadap kehidupan dalam masyarakat modern (Putra, 2020). Banyak media social seperti Twitter, Facebook, Instagram dan sebagainya telah menjadi platform utama yang di gunakan masyarakat luas untuk berinteraksi dan berkomunikasi dalam dunia maya (Liah, et al., 2023). Di media sosial, setiap orang dapat berbagi informasi dan mengemukakan pendapat mereka dengan bebas.

Ditengah boomingnya aktivitas online dari banyak individu, tidak menutup kemungkinan untuk beberapa orang akan menggunakan dan memanfaatkan media social untuk mencari popularitas. Popularitas tersebut banyak bermunculan dari berbagai aspek, seperti adanya berbagai isu-isu social, politik dan budaya yang sering menjadi diskusi, dari media social pula banyak di temukan bakat-bakat luar biasa (Hikmawan, et al, 2020). Selain pengaruh positif, tentunya media social juga memiliki sisi negatifnya pula, seperti banyak isu-isu palsu dan postingan-postingan kurang baik yang berakhir dengan memicu banyak penilaian dari para pengguna social media atau yang sering di sebut dengan netizen (Wijaya & Agus, 2018).

Beberapa waktu lalu, tersebar sebuah video viral yang memunculkan banyak komentar dan tanggapan negative dari para Netizen (Aditiya & Nurdin, 2024). Dimana dalam video tersebut terlihat seorang polisi wanita atau polwan yang bernama Putri Cikita sedang memarahi dan mendorong seorang laki-laki yang sedang makan dan mengatakan bahwa laki-laki tersebut tidak sopan karena berbicara dengannya sambil mengunyah makanan, padahal polisi wanita tersebut yang duluan mengajak berbicara kepada si laki-laki tersebut. Lantaran aksinya yang di anggap berlebihan dalam video tersebut, Polwan itu kini menjadi bulan-bulanan netizen dan mendapat julukan “Duta Sopan Santun” (Lesmana, 2024).

Banyaknya opini yang bermunculan tentang “Duta Sopan Santun” menjadikan peneliti tertarik untuk menganalisis sentimen yang terjadi di Media social. Analisis sentimen adalah studi komputasional dari opini-opini orang, penilaian dan emosi melalui entitas, event dan atribut yang dimiliki (Krisdiyanto, 2021). Yang mana tugas dasar analisis sentimen adalah mengelompokkan polaritas dari teks yang ada dalam dokumen, kalimat, atau fitur/tingkat aspek apakah pendapat yang dikemukakan dalam teks tersebut bersifat positif, negatif atau netral (Dharmendra, et al., 2022).

Analisis sentimen opini publik di Media Sosial dapat berguna untuk memberikan wawasan yang berharga terkait bagaimana persepsi dan tanggapan masyarakat terhadap suatu peristiwa (Andika, et al, 2019). Dalam kasus viralnya video Polwan yang di juluki “Duta Sopan Santun” di Indonesia, penting untuk memahami bagaimana netizen atau pengguna Media Sosial dalam menilai Polisi Wanita dalam Video dan yang berujung dengan lembaga terkait. Hal ini dapat mencakup respons terhadap kinerja polisi dalam pandangan publik.

Dengan menggunakan *Text mining* yang merupakan salah satu teknik untuk melakukan klasifikasi dokumen, dimana text mining merupakan variasi dari data mining yang berusaha menemukan pola yang menarik dari sekumpulan data teks yang berjumlah besar (Rezki, et al., 2020). Salah satu metode klasifikasi yang dapat digunakan dalam melakukan text mining adalah metode Naïve Bayes Classifier (NBC).

Metode Naive Bayes, yang merupakan salah satu algoritma pembelajaran mesin, sering digunakan dalam analisis sentimen karena efisiensinya dalam menangani data teks (Pamungkas & Kharisudin, 2021). Beberapa penelitian terdahulu banyak menggunakan Metode Algoritma ini untuk menganalisis sentiment yang sering terjadi di media social akibat banyaknya isu dan popularitas yang meninggi. Seperti penelitian tentang analisis sentiment terhadap pemilihan capres yang marak di berbagai platform social media yang di analisis oleh (Fitriyyah, et al, 2019) dengan judul penelitian Analisis Sentimen Calon Presiden Indonesia 2019 dari Media Sosial Twitter Menggunakan Metode Naive Bayes. Dengan hasil penelitian di dapat pada pengujian 3 kelas paslon 01 dan paslon 02 didapat hasil akurasi berturut-turut 64, 6% dan 58%. Sedangkan pada pengujian 2 kelas paslon 01 dan paslon 02 didapat hasil akurasi berturut-turut 77, 7% dan 88%. Performansi tertinggi terdapat pada calon presiden nomor urut dua dengan nilai f-measure sebesar 0,88.

Ada pula penelitian terdahulu lainnya yang mengangkat isu sentiment corona-19 yang pernah menggemparkan dunia yang di teliti oleh (Marga, 2021) dengan judul penelitian Sentimen Analisis Tentang Kebijakan Pemerintah Terhadap Kasus Corona Menggunakan Metode Naive Bayes. Dengan hasil penelitian

nilai dari pengujian akurasi penggunaan metode Naive Bayes dengan seleksi fitur TF-IDF mendapat total akurasi sebesar 81% dengan nilai Precision 78%, Recall 91% dan f1-Score 84%, sementara hasil tertinggi didapatkan dari penggunaan parameter algoritma Naive Bayes dan N-Gram jenis Trigram yaitu sebesar 84% dengan nilai Precision 84%, Recall 86%, dan f1-Score 85%. Algoritma Naive Bayes dengan penggunaan ekstrasi fitur N-Gram jenis trigram menunjukkan performa yang cukup baik dalam proses pengklasifikasian data tweet masyarakat terhadap kebijakan pemerintah pada pemberlakuan sistem New Normal.

Penelitian terdahulu lainnya dari (Krisdiyanto, 2021) yang berjudul Analisis sentimen opini masyarakat Indonesia terhadap kebijakan PPKM pada media sosial Twitter menggunakan naïve bayes classifiers, dan penelitian dari (Nurdiansyah, et al, 2018) yang berjudul Sentiment analysis system for movie review in Bahasa Indonesia using naive bayes classifier method, juga menunjukkan bahwa metode ini dapat memberikan hasil yang akurat dalam mengklasifikasikan sentimen dari teks media social.

Sehingga dengan berdasarkan latar belakang tersebut, peneliti mengangkat penelitian ini dengan judul Sentimen Terhadap Duta Sopan Santun di Media Sosial: Studi Kasus dengan Metode Naive Bayes.

II. METODE PENELITIAN

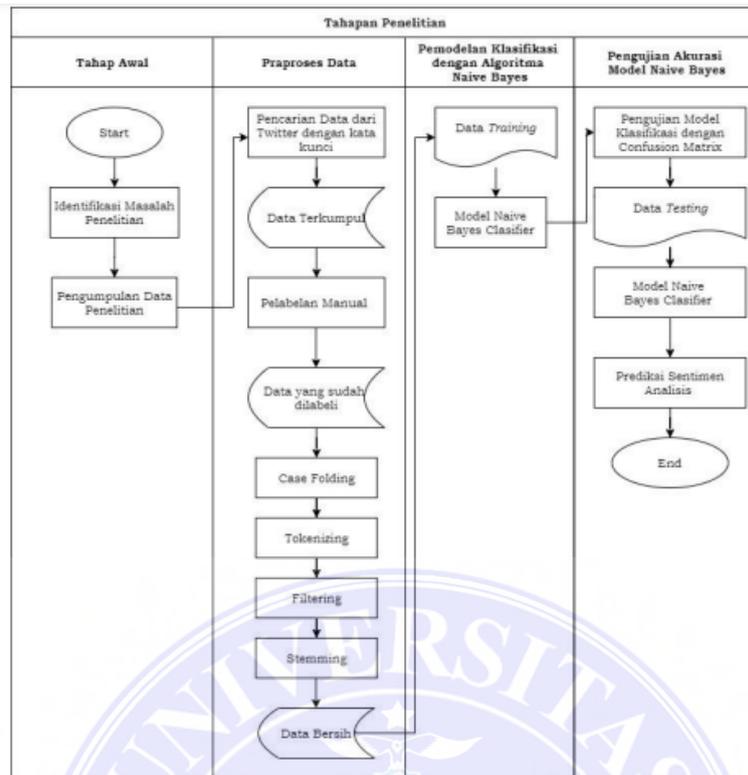
Metode yang di gunakan dalam penelitian ini adalah metode kuantitatif deskriptif dengan metode analisis sentimen menggunakan bantuan Python dan Google Colab. Data diambil dari media sosial Twitter, dengan kata kunci terkait “Duta Sopan Santun”. Data yang digunakan dalam penelitian ini berupa tanggapan masyarakat mengenai postingan video viral dari The Police, sosok polwan yang kini di beri gelar Duta sopan santun oleh netizen yang diambil dari bulan Agustus sampai dengan November 2024. Jumlah data tweet yang diambil sebanyak 2000 tweet. Data yang akan diambil adalah data tweet berbahasa Indonesia dengan kata kunci tertentu terkait dengan Duta Sopan Santun. Kata kunci yang digunakan dalam penelitian ini adalah “Duta sopan santun” dan “Putri Cikita”. Data yang didapatkan kemudian akan dibagi menjadi data latih sebanyak 80% dari jumlah data keseluruhan dan data uji sebanyak 20% dari data keseluruhan. Data latih merupakan data yang telah dibagi sesuai dengan kategori secara manual yang akan dibagi kedalam kelas sentimen positif, negatif dan netral, sedangkan data uji merupakan data yang belum diketahui kategorinya.

Bahan penelitian yang dibutuhkan dalam penelitian ini adalah data opini berbahasa Indonesia yang didapatkan dari sosial media twitter. Informasi lain juga diperoleh melalui studi literatur, internet dan bahan bacaan yang sesuai dengan topik penelitian

Praproses data merupakan tahap dimana teks akan diseragamkan bentuk dan format agar dapat dipersiapkan menjadi data yang dapat diolah pada tahap selanjutnya. Text preprocessing meliputi case folding, tokenizing, filtering dan stemming.



Gambar 1 Tahapan Text Processing



Gambar 2 Tahap Penelitian

III. HASIL DAN PEMBAHASAN

A. Hasil

1. Pengumpulan data

Tahap awal yang dilakukan adalah mengumpulkan data tweet berbahasa Indonesia dengan pencarian kata kunci “Duta sopan santun” dan “Putri Cikita” menggunakan Twitter API. Terlebih dahulu dilakukan Import Libraries yang disediakan oleh Python. Library yang digunakan dalam pengumpulan data yang digunakan adalah library tweepy yang dapat mengakses API twitter secara langsung di console ataupun script. Library sys adalah library yang digunakan untuk menyediakan akses ke beberapa variabel yang digunakan atau dikelola oleh penerjemah, library matplotlib.pyplot digunakan untuk membuat fungsi visualisasi kedalam bentuk grafik.

```

# Import Libraries
from textblob import TextBlob
import sys
import tweepy
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
import os
import nltk
import pycountry
import re
import string
    
```

Gambar 3 Import Library Python

Untuk dapat mengambil data dari twitter, terlebih dahulu mendaftar di twitter Developer, dilakukan pengisian semua form yang ditampilkan. Setelah selesai proses mendaftar akan diberikan consumer key, consumer secret, acces token dan acces token secret yang digunakan untuk mengakses data pada twitter. Setelah didapatkan Token dari Developer twitter selanjutnya dilakukan autentikasi token dengan code python, seperti berikut:

```
# Authentication
consumerKey = "ptvioriiTU1M3tB55wEVZkce1"
consumerSecret = "yBHQDANTueQv5nE2Rfpv8Wx92dEW6JWE1U01eVekT77nvgrRCX"
accessToken = "8158487597749376-hfdh3LULlqR5p1PwIFeG5WQFc45og5"
accessTokenSecret = "GrppnST61FGf10GGl0ZBTgrFGpKpQHmHmP5pFX8r4aP"

auth = tweepy.OAuthHandler(consumerKey, consumerSecret)
auth.set_access_token(accessToken, accessTokenSecret)
api = tweepy.API(auth,wait_on_rate_limit=True)
```

Gambar 4 Code Python Authentikasi Token

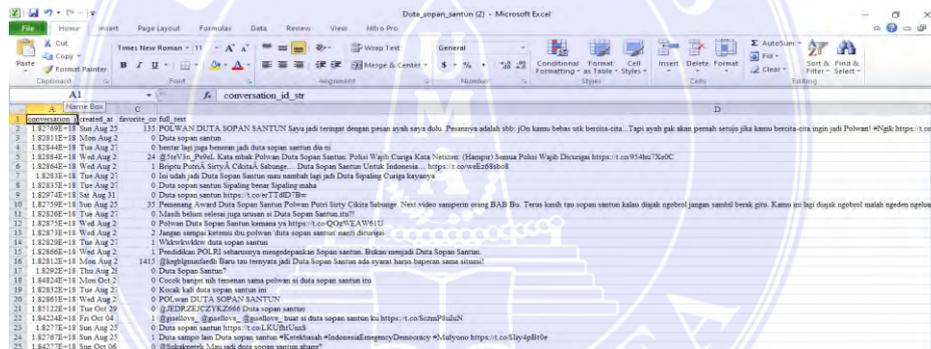
Setelah terkoneksi dengan twitter, program akan meminta data tentang kata kunci maupun hashtag yang dicari, kemudian dimasukkan kedalam variable keyword dan akan meminta jumlah data tweet yang akan dianalisis. Kemudian dimasukkan kedalam variabel noOfTweet yang berupa integer. Variabel tweet akan melakukan operasi pengambilan data yang telah disimpan pada variable searchTerm dan noOfSearchTerm. Kata kunci yang digunakan dalam pencarian data adalah "Duta sopan santun" dan "Putri Cikita".

```
keyword = input("Masukkan Hashtag atau Query yang dicari: ")
noOfTweet = int(input("Masukkan jumlah Tweet : "))

tweets = tweepy.Cursor(api.search, q=keyword, since='2021-01-01', until='2021-03-31').items(noOfTweet)
```

Gambar 5 Code Python Crawling data

Data yang terkumpul sebanyak 2000 tweet dikumpulkan dan disimpan kedalam file excel dengan format .xlsx. Tidak ada batasan dalam jumlah data yang diambil namun dengan 2000 data diharapkan dapat mewakili hasil opini masyarakat secara umum. Data yang tersimpan terdiri dari tanggal pembuatan tweet, user twitter yang memposting tweet mengenai kuliah daring dan kuliah online serta tweet yang diposting. Data yang terkumpul dalam format excel dapat dilihat sebagai berikut:



Gambar 6 Hasil Pengumpulan data dengan Twitter API

2. Praprocessing

Praprocessing data dilakukan dengan tahap Case Folding, Tokenizing, Filtering, Stemming sehingga menghasilkan data bersih dan siap untuk lanjut pada proses berikutnya

a. Case Folding

Case Folding adalah proses merubah data tweet menjadi lowercase.

```
# ----- Case Folding -----
# gunakan fungsi Series.str.lower() pada Pandas
tw_list['text'] = tw_list['text'].str.lower()

print('Case Folding Result : \n')
print(tw_list['text'].head(5))
print('\n\n')
```

Gambar 7 Kode Program tahap Case Folding

b. Tokenizing

Tahap tokenization dilakukan dengan menggunakan fungsi nltk_tokenize(), library pada bahasa pemrograman Python3 yang bernama NLTK. Dilakukan Import library terlebih dahulu

```
import string
import re #regex Library
```

Gambar 8 Import Library yang dibutuhkan

Library string digunakan untuk memuat satu karakter atau lebih yang ada pada data tweet. Terlebih dahulu diimport library re untuk melakukan tahapan Regular Expression (regex) atau deretan karakter yang digunakan untuk pencarian teks dengan menggunakan pola (pattern). Dengan menggunakan library regex dapat memudahkan dalam mencari string tertentu dari teks yang banyak. Selain itu pada tahap ini juga dilakukan proses removing number, whitespace dan punctuation (tanda baca).

```
def remove_tweet_special(text):
    # remove tab, new line, and back slice
    text = text.replace('\t', ' ').replace('\n', ' ').replace('\u', ' ').replace('\ ', ' ')
    # remove non ASCII (emojicon, chinese word, etc)
    text = text.encode('ascii', 'replace').decode('ascii')
    # remove mention, link, hashtag
    text = ' '.join(re.sub("([@#][A-Za-z0-9]+)|(\u+|\//\S+)", "", text).split())
    # remove incomplete URL
    return text.replace("http://", " ").replace("https://", " ")

tw_list['text'] = tw_list['text'].apply(remove_tweet_special)

#remove number
def remove_number(text):
    return re.sub("^\d+", "", text)

tw_list['text'] = tw_list['text'].apply(remove_number)

#remove punctuation
def remove_punctuation(text):
    return text.translate(str.maketrans("", "", string.punctuation))

tw_list['text'] = tw_list['text'].apply(remove_punctuation)

#remove whitespace leading & trailing
def remove_whitespace_LT(text):
    return text.strip()
```

Gambar 9 Code Python Tahap Tokenizing

Pada tahap ini juga dilakukan proses normalization, yaitu mengubah kata yang tidak lengkap, kesalahan dalam pengetikan (typo) kedalam kata yang normal dan dapat dipahami dengan baik.

```
normalized_word_dict = {}

for index, row in normalized_word.iterrows():
    if row[0] not in normalized_word_dict:
        normalized_word_dict[row[0]] = row[1]

def normalized_term(document):
    return [normalized_word_dict[term] if term in normalized_word_dict else term for term in document]

tw_list['tweet_normalized'] = tw_list['tweet_tokens_WSW'].apply(normalized_term)
```

Gambar 10 Code Python Tahap Normalization

c. Filtering

Proses filtering disebut dengan Stopword Removal. Pada tahap ini menggunakan nltk. NLTK (Natural Language Toolkit) adalah Library yang disediakan oleh Python untuk membangun program analisis teks. Terlebih dahulu dilakukan proses install library nltk pada anaconda prompt.

```
(base) C:\Users\Claudia>pip install nltk
```

Gambar 11 Code Python Install NLTK

Setelah library diinstall selanjutnya dilakukan proses download stopwords yang ada dan menggunakan stopwords bahasa indonesia. Hasilnya tweet yang didapatkan dan menggunakan kata yang ada dalam list stopwords indonesia akan dibersihkan

```
import nltk
nltk.download('stopwords')

[nltk_data] Downloading package stopwords to
[nltk_data] C:\Users\62813\AppData\Roaming\nltk_data...
[nltk_data] Package stopwords is already up-to-date!
```

Gambar 12 Code Python Import nltk

Selain list stopword Indonesia yang disediakan oleh library nltk, ditambahkan list kata yang tidak dibutuhkan dalam analisis sentimen dengan cara menambahkan secara langsung kata pada list_stopword.extends agar dapat dihapus oleh system.

```
from nltk.corpus import stopwords

# ----- get stopword from NLTK stopword -----
# get stopword Indonesia
list_stopwords = stopwords.words('indonesian')

# ----- manually add stopword -----
# append additional stopword
list_stopwords.extend(['yg', 'gpp', 'make', 'dg', 'gua', 'gw', 'nt', 'dgn', 'ny', 'd', 'klo',
                      'kalo', 'amp', 'biar', 'bikin', 'bilang',
                      'gak', 'ga', 'krn', 'nya', 'nih', 'sih',
                      'sl', 'tau', 'tdk', 'tuh', 'utk', 'ya',
                      'jd', 'jgn', 'sdh', 'aja', 'n', 't',
                      'nyg', 'hehe', 'pen', 'u', 'nan', 'loh', 'rt',
                      '&amp;', 'yah'])

# ----- add stopword from txt file -----
# read txt stopword using pandas
txt_stopword = pd.read_csv("coba\stopwords.txt", names= ["stopwords"], header = None)

# convert stopword string to list & append additional stopword
list_stopwords.extend(txt_stopword["stopwords"][0].split(' '))

# -----
# convert list to dictionary
list_stopwords = set(list_stopwords)

# remove stopword pada list token
def stopwords_removal(words):
    return [word for word in words if word not in list_stopwords]

tw_list['tweet_tokens_NSW'] = tw_list['tweet_tokens'].apply(stopwords_removal)
```

Gambar 13 Code Python Tahap Filtering

d. Stemming

Stemming adalah tahap mencari root (dasar) kata dari tiap kata hasil filtering dengan menghapus kata imbuhan di depan maupun imbuhan di belakang kata. Tahap stemming dilakukan dengan menggunakan bantuan library pada bahasa pemrograman Python3 yang bernama Sastrawi. Dalam hal ini, dilakukan instalasi library menggunakan perintah pip install sastrawi Praprocessing data dilakukan dengan tahap Case Folding, Tokenizing, Filtering, Stemming sehingga menghasilkan data bersih dan siap untuk lanjut pada proses berikutnya.

```
pip install sastrawi
```

Gambar 14 Perintah Instalasi Sastrawi

Setelah library telah berhasil diinstall, selanjutnya adalah mengimport kelas StemmerFactory dari library Sastrawi

```
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
```

Gambar 15 Import Kelas Sastrawi

```

# create stemmer
factory = StemmerFactory()
stemmer = factory.create_stemmer()

# stemmed
def stemmed_wrapper(term):
    return stemmer.stem(term)

term_dict = {}

for document in tw_list['tweet_normalized']:
    for term in document:
        if term not in term_dict:
            term_dict[term] = ''

print(len(term_dict))
print("-----")

for term in term_dict:
    term_dict[term] = stemmed_wrapper(term)
    print(term,":", term_dict[term])

print(term_dict)
print("-----")

# apply stemmed term to dataframe
def get_stemmed_term(document):
    return [term_dict[term] for term in document]

tw_list['tweet_tokens_stemmed'] = tw_list['tweet_normalized'].swifter.apply(get_stemmed_term)
print(tw_list['tweet_tokens_stemmed'])

```

Gambar 16 Code Python Tahapan Stemming

3. Pelabelan Data

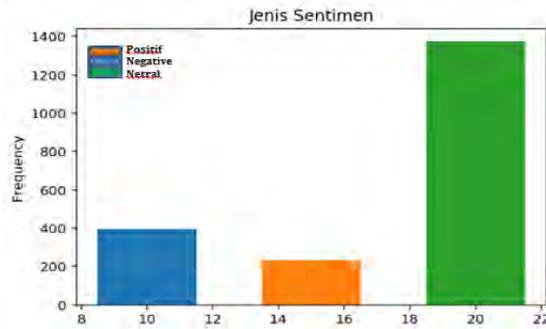
Pelabelan data hasil crawling dan telah melalui tahapan preprocessing dilakukan dengan menggunakan library python textblob dengan melihat polarity,subjectivity yang dimiliki oleh teks tweet yang telah dikumpulkan. Textblob adalah salah satu library yang disediakan oleh Python untuk pemrosesan dibidang Natural Language Processing yang dapat memberikan tag kata, ekstraksi kata, penerjemahan kata dan sentiment analysis. Hasil objek textblob dapat digunakan untuk melakukan proses pembelajaran bahasa alami namun saat ini textblob hanya tersedia dalam bahasa inggris oleh sebab itu pada penelitian ini yang menggunakan data bahasa Indonesia dilakukan translate ke dalam bahasa inggris terlebih dahulu. Penentuan kelas positif, netral dan negatif didasari oleh nilai polaritas. Nilai polaritas pada analisis sentimen berada pada rentang 1 sampai -1 yang menunjukkan kelas sentimen data. Teks tweet dengan nilai polaritas mengarah ke nilai 1 menunjukkan sentimen kelas positif, nilai polaritas mengarah ke nilai -1 menunjukkan kelas sentimen negatif dan nilai polaritas berkisar pada nilai 0 masuk kedalam kelas netral.

Selanjutnya data akan dibagi dengan 80% data latih dan 20% data uji. Sentimen pada data training sejumlah 1600 dibagi secara manual sesuai kelasnya. 400 data lainnya akan digunakan sebagai data uji. Hasil klasifikasi data dapat dilihat pada tabel berikut:

Tabel 1 Data Sentimen

Positif	Netral	Negative	Jumlah
231	1368	401	2000
11,55%	68,40%	20,05%	100%

Didapatkan hasil akhir dari pelabelan dengan menggunakan library textblob sebanyak 2000 data tweet adalah 401 tweet yang masuk dalam kelas positif, 1368 tweet kelas netral dan 231 tweet kelas negatif. Persentase labeling data dengan textblob sebagai berikut:



Gambar 17 Hasil persentase labelling data tweet dengan textblob

4. Ekstraksi Fitur

Dalam tahapan ekstraksi fitur, tahap awal yang dilakukan adalah mengubah dataset penelitian ke dalam representasi vector. Python memiliki Library yang bernama Scikit Learn yang dapat digunakan dalam Machine Learning. Pada library ini terdapat algoritma CountVectorizer yang dapat mengubah fitur teks menjadi sebuah representasi vector. Hasil representasi vector didapatkan 2000 angka yang mempunyai 4092 Kata.

```
#Applying CountVectorizer
countVectorizer = CountVectorizer(analyzer=clean_text)
countVector = countVectorizer.fit_transform(tw_list['text'])
print('{} Number of reviews has {} words'.format(countVector.shape[0], countVector.shape[1]))
#print(countVectorizer.get_feature_names())
```

Gambar 18 Code Python CountVectorizer

Setelah merubah data teks menjadi word vector, selanjutnya dilakukan penjelajahan beberapa kata yang sering digunakan atau kata mana yang muncul secara bersamaan. Digunakan model probabilistik N-gram yang digunakan untuk memprediksi kata berikutnya. Penerapan N-gram dapat dilakukan dalam membuat Bigram dan Trigram. Bigram adalah N-gram yang terdiri dari 2 item atau kata dari sebuah kalimat. Trigram adalah N-gram yang terdiri dari 3 item atau kata dari sebuah kalimat. Code Python untuk menjalankan perintah bigram.

```
#n2 bigram
n2_bigrams = get_top_n_gram(data['Text'],(2,2),20)
n2_bigrams
```

Gambar 19 Tahapan Bigram

```
#n3 trigram
n3_trigrams = get_top_n_gram(data['Text'],(3,3),20)
n3_trigrams
```

Gambar 20 Tahapan Triagram

5. Klasifikasi Naïve Bayes

Proses ini dilakukan dengan menggunakan bantuan library pada bahasa pemrograman Python3 yang bernama scikit-learn untuk proses klasifikasi. Pada proses klasifikasi digunakan data uji sebanyak 20% dari data keseluruhan.

```
from sklearn.model_selection import train_test_split
import pandas as pd
X_train, X_test, y_train, y_test = train_test_split(countVector, data['Label'],
                                                    test_size=0.2, random_state=21)
```

Gambar 21 Pembagian Data Tes dan Uji

Didapatkan hasil dari x_train_shape, x_test_shape, y_train_shape dan y_test_shape seperti berikut:

```
print("X_train_shape : ",X_train.shape)
print("X_test_shape : ",X_test.shape)
print("y_train_shape : ",y_train.shape)
print("y_test_shape : ",y_test.shape)

X_train_shape : (1600, 4092)
X_test_shape : (400, 4092)
y_train_shape : (1600,)
y_test_shape : (400,)
```

Gambar 22 Hasil dari pembagian data

Langkah selanjutnya adalah melakukan klasifikasi digunakan library Multinomial Naive Bayes, sebagai berikut:

```
from sklearn.naive_bayes import MultinomialNB # Naive Bayes Classifier

model_naive = MultinomialNB().fit(X_train, y_train)
predicted_naive = model_naive.predict(X_test)
```

Gambar 23 Script Python Klasifikasi Naive Bayes

6. Pengujian Model

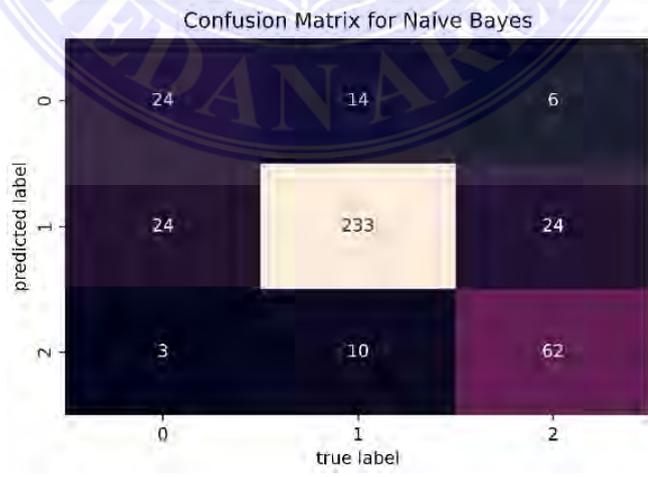
Untuk mengetahui performa dari Algoritma Naive Bayes, maka dilakukan pengujian terhadap model yang telah dibuat. Hasil klasifikasi akan divisualisasi dalam bentuk confusion matrix. Confusion matrix merupakan salah satu metode yang dapat digunakan untuk mengukur kinerja suatu metode klasifikasi. Pada dasarnya confusion matrix mengandung informasi yang membandingkan hasil klasifikasi yang dilakukan oleh sistem dengan hasil klasifikasi yang seharusnya.

```
from sklearn.metrics import confusion_matrix
import seaborn as sns

plt.figure(dpi=600)
mat = confusion_matrix(y_test, predicted_naive)
sns.heatmap(mat.T, annot=True, fmt='d', cbar=False)

plt.title('Confusion Matrix for Naive Bayes')
plt.xlabel('true label')
plt.ylabel('predicted label')
plt.savefig("confusion_matrix.png")
plt.show()
```

Gambar 24 Code Python Confusion Matrix



Gambar 25 Visualisasi Confusion Matrix

Setelah diketahui Confusion Matrix dari model yang dibuat selanjutnya dilakukan perhitungan nilai akurasi dari model klasifikasi Naive Bayes yang telah dibuat menggunakan sklearn.metrics dengan mengimport accuracy score yang disediakan oleh library scikit.learn. Didapatkan hasil dari perhitungan matrix dengan Code Python yaitu sebesar 0,7975

```

from sklearn.metrics import accuracy_score

score_naive = accuracy_score(predicted_naive, y_test)
print("Accuracy with Naive-bayes: ", score_naive)

Accuracy with Naive-bayes: 0.7975
    
```

Gambar 26 Code Python Perhitungan Akurasi Model

7. Evaluasi Model

Evaluasi model bertujuan untuk menghasilkan confusion matrix dengan ukuran 3 x 3. Confusion Matrix memberikan informasi perbandingan hasil klasifikasi yang dilakukan oleh model klasifikasi dengan hasil klasifikasi sebenarnya. Hasil evaluasi model dengan confusion matrix adalah sebagai berikut:

Tabel 2 Hasil Confusion Matrix

		Predict Class		
		Positif	Netral	Negatif
Actual class	Positif	24	14	6
	Netral	24	233	24
	Negative	3	10	62

Sumber: Olah data, 2024

Dari confusion matrix dapat dijelaskan bahwa model mengklasifikasikan secara benar sebesar 24 data sebagai positif, 233 data sebagai netral dan 62 sebagai data negatif. Selain itu model salah dalam memprediksi 6 data ke dalam kelas negative yang seharusnya positif (False Positif), serta salah dalam memprediksi 3 data ke dalam kelas positif yang seharusnya negatif (False Negatif). Berdasarkan hasil pada pengujian model klasifikasi Naive Bayes sebelumnya nilai akurasi pada keseluruhan sistem dapat dihitung sebesar 79,75 %.

```

from sklearn.metrics import accuracy_score

score_naive = accuracy_score(predicted_naive, y_test)
print("Accuracy with Naive-bayes: ", score_naive)
    
```

Gambar 27 Kode Python Perhitungan Akurasi

Perhitungan akurasi manual dari perhitungan matriks diatas adalah sebagai berikut.

$$\begin{aligned}
 \text{Akurasi} &= \frac{\text{True Positif} + \text{True Netral} + \text{True Negatif}}{\text{Total Data yang di Uji}} \times 100\% \\
 &= \frac{319}{400} \times 100\% \\
 &= 79,75\%
 \end{aligned}$$

Akurasi didapatkan dari perhitungan rasio prediksi benar dengan keseluruhan data. Dengan mengetahui besarnya nilai akurasi pada kinerja keseluruhan sistem dapat dinyatakan tingkat kemampuan sistem dalam mencari ketepatan antara informasi yang diinginkan pengguna dengan jawaban yang diberikan sistem. Tingkat keberhasilan system dalam menemukan sebuah informasi dalam penelitian ini sebesar 79,75 %.

Selanjutnya untuk melihat performa klasifikasi dari setiap kelas dapat diketahui melalui nilai presisi, recall dan f1 score pada setiap kelas klasifikasi. Presisi menggambarkan tingkat keakuratan data yang diminta dengan hasil yang diberikan oleh model. Presisi didapatkan dari perhitungan rasio prediksi benar dibandingkan dengan keseluruhan hasil yang diprediksi positif. Recall menggambarkan keberhasilan model dalam menemukan kembali informasi yang dimasukkan dalam pengujian. Recall didapatkan dari hasil perhitungan rasio prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif. F1-Score merupakan parameter tunggal ukuran keberhasilan retrieval yang menggabungkan recall dan precision.

Hasil nilai presisi, recall, dan f1-score memiliki nilai sebesar 0-1. Semakin tinggi nilai maka semakin baik hasil model yang dibuat. Nilai akurasi yang tinggi didapat ketika banyak data yang berhasil diklasifikasi dengan benar sesuai kelas sentimennya. Dapat diketahui juga nilai Precision dan Recall. Nilai

Precision mengikuti nilai akurasi, semakin tinggi nilai akurasi maka akan diikuti nilai Precision yang tinggi juga, begitu sebaliknya. Nilai Precision adalah jumlah data positif yang benar diklasifikasi sebagai data positif dibagi total data yang diklasifikasi sebagai data positif. Sedangkan Nilai recall adalah jumlah data positif yang benar diklasifikasi sebagai data positif dibagi jumlah data positif sebenarnya. Pada confusion matrix sebelumnya, dapat diketahui nilai true positif dan true negatif. True positif adalah nilai data positif yang diklasifikasi dengan benar sesuai kelas sentimennya, yaitu positif. True negatif adalah nilai data sentimen yang diklasifikasi dengan benar sesuai kelas sentimennya, yaitu negatif.

Tabel 3 Nilai Presisi, Recall, dan F1-Score Evaluasi Model

Jenis Klasifikasi	Presisi	Recall	F1-Score
Positif	0,55	0,47	0,51
Netral	0,83	0,91	0,87
Negative	0,83	0,67	0,74

Sumber: Olah data, 2024

Hasil dari evaluasi model dapat dilihat bahwa nilai presisi dan recall disetiap kelas dapat dikatakan memiliki tingkat kemampuan yang tinggi dalam mencari ketepatan antara informasi yang diminta oleh pengguna. Nilai presisi untuk kelas positif sebesar 55%, untuk kelas netral sebesar 83%, untuk kelas negatif sebesar 83%. Angka ini dapat diartikan bahwa proporsi label yang diprediksi dengan benar dari total prediksi cukup tinggi untuk kelas netral dan negatif. Sedangkan tingkat keberhasilan sistem dalam menemukan kembali sebuah informasi untuk kelas positif sebesar 47%, untuk kelas netral sebesar 91% dan kelas negatif sebesar 67%. Hal ini berarti kinerja keberhasilan sistem dalam menemukan kembali informasi yang bernilai positif dalam dokumen rendah dibandingkan dengan menemukan informasi kembali yang bernilai negatif dan netral. Didapatkan rata rata nilai precision sebesar 0,79%, nilai recall sebesar 80% dan nilai F1-Score sebesar 79%

IV. KESIMPULAN

Berdasarkan hasil pengujian algoritma Naive Bayes Classifier yang telah dilakukan ada beberapa hal yang dihasilkan, antara lain :

1. Pada penelitian ini, algoritma Naive Bayes Classifier terbukti algoritma yang akurat karena menghasilkan nilai akurasi sebesar 79,75%
2. Hasil analisis sentimen twitter dengan kata Duta sopan santun dan putri cikita pada penelitian ini memiliki nilai Precision sebesar 79%, nilai Recall sebesar 80% dan nilai F1-Score sebesar 79%.

DAFTAR PUSTAKA

- Aditiya, S., & Nurdin, S. (2024, Agustus 26). *Sosok Polwan yang Disebut Warganet 'Duta Sopan Santun'*. Retrieved November 29, 2024, from viva.co.id: <https://www.viva.co.id/trending/1745835-sosok-polwan-yang-disebut-warganet-duta-sopan-santun>
- Andika, L. A., Azizah, P. A., & Respatiwan, R. (2019). Analisis sentimen masyarakat terhadap hasil quick count pemilihan presiden indonesia 2019 pada media sosial twitter menggunakan metode naive bayes classifier. *Indonesian Journal of Applied Statistics*, 2(1), 34-41.
- Aris, H. (2024, Agustus 25). *Video Viral Polwan Brigadir Putri Cikita Dapat Julukan Duta Sopan Santun dari Netizen, Begini Kronologinya*. Retrieved November 29, 2024, from radarsemarang.jawapos.com: <https://radarsemarang.jawapos.com/berita/725015722/video-viral-polwan-brigadir-putri-cikita-dapat-julukan-duta-sopan-santun-dari-netizen-begini-kronologinya>
- Dharmendra, I. K., Januhari, N. N., Ramayasa, I. P., & Putra, I. M. (2022). Uji Komparasi Sentiment Analysis Pada Opini Alumni Terhadap Perguruan Tinggi. *Jurnal Teknik Informatika UNIKA Santo Thomas*, 7(1), 1-6.
- Fitriyyah, S. N., Safriadi, N., & Pratama, E. E. (2019). Analisis Sentimen Calon Presiden Indonesia 2019 dari Media Sosial Twitter Menggunakan Metode Naive Bayes. *JEPIN (Jurnal Edukasi dan Penelitian Informatika)*, 5(3), 279-285.
- Hikmawan, S., Pardamean, A., & Khasanah, S. N. (2020). Sentimen Analisis Publik Terhadap Joko Widodo terhadap wabah Covid-19 menggunakan Metode Machine Learning. *Jurnal Kajian Ilmiah*, 20(2), 167-176.

- Juniardi, T., & Sugianto, C. A. (2024). ANALISIS SENTIMEN TIM NASIONAL SEPAK BOLA INDONESIA DI TURNAMEN PIALA DUNIA U-17 INDONESIA PADA TWITTER (X) MENGGUNAKAN ALGORITMA NAÏVE BAYES. *Jurnal Informatika dan Teknik Elektro Terapan*, 12(3S1).
- Krisdiyanto, T. (2021). Analisis sentimen opini masyarakat Indonesia terhadap kebijakan PPKM pada media sosial Twitter menggunakan naïve bayes classifiers. *Jurnal CoreIT: Jurnal Hasil Penelitian Ilmu Komputer dan Teknologi Informasi*, 7(1), 32-37.
- Lesmana, A. S. (2024, Agustus 27). *Viral Lagi! Polwan Putri Cikita "Duta Sopan Santun" Tuduh Pria Ngobat: Polisi Wajib Curiga!* Retrieved 11 31, 2024, from Suara.com: <https://www.suara.com/news/2024/08/27/092959/viral-lagi-polwan-putri-cikita-duta-sopan-santun-tuduh-pria-ngobat-polisi-wajib-curiga>
- Liah, A. N., Maulana, F. S., Aulia, G. N., Syahira, S., Nurhaliza, S., Rozak, R. W., et al. (2023). Pengaruh media sosial terhadap degradasi moral generasi Z. *Nautical: Jurnal Ilmiah Multidisiplin Indonesia*, 2(1), 68-73.
- Marga, N. S. (2021). Sentimen Analisis Tentang Kebijakan Pemerintah Terhadap Kasus Corona Menggunakan Metode Naive Bayes. *Jurnal Informatika Dan Rekayasa Perangkat Lunak*, 2(4), 453-463.
- Nurdiansyah, Y., Bukhori, S., & Hidayat, R. (2018). Sentiment analysis system for movie review in Bahasa Indonesia using naïve bayes classifier method. *In Journal of Physics: Conference Series. IOP Publishing.*, Vol. 1008, No. 1, p. 012011.
- Pamungkas, F. S., & Kharisudin, I. (2021). Analisis Sentimen dengan SVM, NAIVE BAYES dan KNN untuk Studi Tanggapan Masyarakat Indonesia Terhadap Pandemi Covid-19 pada Media Sosial Twitter. *In PRISMA, Prosiding Seminar Nasional Matematika*, Vol. 4, pp. 628-634.
- Putra, M. W. (2020). Analisis Sentimen Dompot Elektronik pada Twitter Menggunakan Metode Naïve Bayes Classifier. *IT Journal Research and Development*, 5(1), 72-86.
- Rezki, M., Kholifah, D. N., Faisal, M., Priyono, P., & Suryadithia, R. (2020). Analisis Review Pengguna Google Meet dan Zoom Cloud Meeting Menggunakan Algoritma Naïve Bayes. *Jurnal Infortech*, 2(2), 264-270.
- Rezki, M., Kholifah, D. N., Faisal, M., Priyono, P., & Suryadithia, R. (2020). Analisis Review Pengguna Google Meet Dan Zoom Cloud Meeting Menggunakan Algoritma Naïve Bayes. *Jurnal Infortech*, 2(2), 264-270.
- Wijaya, P., & Agus, S. (2018). Naive Bayes Classification pada Klasifikasi Dokumen Untuk Identifikasi Konten E-Government. *Journal of Applied Intelligent System.*, 1(1). 48-55.