SENTIMEN ANALISIS TERHADAP KOMENTAR YOUTUBE MENGGUNAKAN ALGORITMA KLASIFIKASI NAIVE BAYES

DISUSUN OLEH:

NAMA	**:**
NIM	**: A11.2022.14741**
MATA KULIAH	**: DATA MINNING**
KELOMPOK	**: A11.4519**
DOSEN	**:**

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS ILMU KOMPUTER

UNIVERSITAS DIAN NUSWANTORO

2024

DAFTAR ISI

RINGKASAN DAN PERMASALAHAN

A. RINGKASAN

Penelitian ini bertujuan untuk menerapkan metode Naive Bayes dalam analisis sentimen komentar YouTube yang berkaitan dengan politik Indonesia masa kini. Dengan melakukan analisis ini, diharapkan dapat memberikan wawasan yang lebih mendalam mengenai bagaimana masyarakat Indonesia merespon dan menilai berbagai isu politik yang sedang berkembang. Hasil dari penelitian ini diharapkan dapat bermanfaat bagi para peneliti, pembuat kebijakan, serta para content creator dan media dalam memahami dinamika politik dan opini publik di Indonesia.

B. PERMASALAHAN

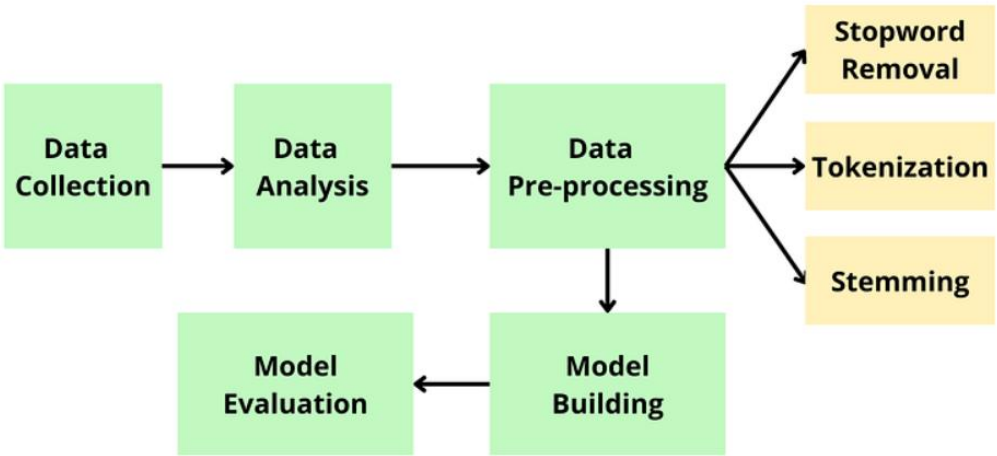
Penelitian ini memiliki rumusan masalah sebagai berikut:

1. Bagaimana cara mengumpulkan data komentar YouTube yang relevan dengan politik Indonesia masa kini?
2. Bagaimana cara memproses dan mempersiapkan data komentar untuk analisis sentimen?
3. Bagaimana penerapan metode Naive Bayes untuk analisis sentimen komentar YouTube?

C. TUJUAN

Penelitian ini bertujuan untuk mengumpulkan dan memproses data komentar YouTube yang relevan dengan politik Indonesia masa kini, serta menerapkan metode Naive Bayes untuk analisis sentimen. Melalui penerapan model ini, penelitian ini bertujuan untuk mengidentifikasi dan mengklasifikasikan sentimen komentar menjadi kategori positif, negatif, atau netral, serta mengungkap tren sentimen masyarakat terhadap isu-isu politik terkini. Hasil analisis ini diharapkan memberikan wawasan yang berguna bagi peneliti, pembuat kebijakan, content creator, dan media dalam memahami opini publik dan menyusun strategi komunikasi yang lebih efektif.

D. ALUR PENYELESAIAN ATAU PENGERJAAN



PENJELASAN DATASET DAN FEATURE DATASET

A. PENJELASAN DATASET

- Sumber data
  Dataset bisa didapatkan dengan cara Crawling Komentar Youtube melalu API atau Scrapping, disini saya menggunakan **CrawlYT.ipynb**

- Fitur atau Atribut data
  Dataset terdiri dari atribut berikut (username,comment,date)

youtube_video_comments.csv

1 to 10 of 4452 entries Filter

username	comment	date
@HendrikWirjana	2	2024-12-31T10:02:47Z
@NaniPancing	Suka liat video pasca pilpres...02🎉	2024-11-10T22:54:19Z
@BulanMerindu-r1f	🏛️	2024-08-08T13:28:03Z
@BulanMerindu-r1f	🎊	2024-08-08T13:27:23Z
@BulanMerindu-r1f	Bismillah Ya Allah semoga Allah swt 03 1menang AAamiinn Ya robbal Alamiinn 🎁🇲🇨📍	2024-08-08T13:26:11Z
@MuhammadSalehBangko548	Daerah Aceh laut biru 25.00	2024-06-25T14:46:41Z
@MuhammadSalehBangko548	110 laut biru pak	2024-06-25T14:19:03Z
@MuhammadSalehBangko548	Dari balikpapan 21.00 laut biru	2024-06-25T14:15:02Z
@MuhammadSalehBangko548	Masuk lagi laut biru 27.00 jumlahnya	2024-06-25T14:14:28Z
@MuhammadSalehBangko548	Kalimantan Timur daerah bontang 57.00	2024-06-25T14:04:06Z

Show 10 per page 1 2 10 100 400 440 446

B. EDA

Proses Eksplorasi dataset untuk memahami struktur dataset seperti distribusi data, panjang text

```
import pandas as pd

data = pd.read_csv('../content/drive/MyDrive/Colab Notebooks/data_debat/youtube_video_comments.csv')
print(data.head(2))
data.info()
data.describe()
```

```
          username                                comment                 date
0  @HendrikWirjana                                      2  2024-12-31T10:02:47Z
1     @NaniPancing  Suka liat video pasca pilpres...02🎉  2024-11-10T22:54:19Z
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4452 entries, 0 to 4451
Data columns (total 3 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   username  4452 non-null   object
 1   comment   4452 non-null   object
 2   date      4452 non-null   object
dtypes: object(3)
memory usage: 104.5+ KB
```

	username	comment	date
count	4452	4452	4452
unique	3679	4359	4340
top	@rachelayomi1275	Kalau ternyata rakyat masih ingin dipimpin jok...	2024-02-15T01:54:21Z
freq	15	6	3

print(data.head(2))
data.info()
data.describe()

C. PROSES FEATURE DATASET

Setelah proses EDA atau Eksplorasi data, maka proses feature dijalankan bertujuan untuk membersihkan dan menyiapkan data teks untuk dianalisis sentiment

- Melihat dataset (username, comment) dan menghapus data duplicate

```
import re
import string
import html
import nltk

df = pd.DataFrame(data[['username', 'comment']])
df.head()
```

	username	comment
0	@HendrikWirjana	2
1	@NaniPancing	Suka liat video pasca pilpres...02 🎉
2	@BulanMerindu-r1f	🏢
3	@BulanMerindu-r1f	🎎
4	@BulanMerindu-r1f	Bismillah Ya Allah semoga Allah swt 03 1m...

```
[63] df = df.drop_duplicates(subset=['username', 'comment'])
     df = df.dropna()
     df.shape
```

```
(4399, 2)
```

- Cleaning dataset

```
[165]     text = re.sub(r'\.', ' ', text)

          # Menghapus tag HTML seperti <br>
          text = re.sub(r'<[^>]+>', '', text)

          # Menghapus mention, hashtag, retweet, dan URL
          text = re.sub(r'@[A-Za-z0-9_]+', '', text)
          text = re.sub(r'#\w+', '', text)
          text = re.sub(r'RT[\s]+', '', text)
          text = re.sub(r'https?://\S+', '', text)

          # Menghapus angka
          text = re.sub(r'\b\d+\b', '', text)

          # Menghapus karakter non-alfanumerik kecuali spasi
          text = re.sub(r'[^A-Za-z0-9 ]', '', text)

          # Menghapus spasi berlebih
          text = re.sub(r'\s+', ' ', text).strip()

          return text

      # Terapkan fungsi cleaning pada kolom komentar
      df['cleaning'] = df['comment'].fillna('').str.lower().apply(clean_comment)

      # Pilih kolom yang diinginkan
      df_cleaned = df[['username', 'comment', 'cleaning']]

      # Simpan DataFrame yang telah dibersihkan ke file CSV
      df_cleaned.to_csv('aftercleaned_comments.csv', index=False)
```

```
df_cleaned.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 4132 entries, 0 to 4398
Data columns (total 3 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   username  4132 non-null   object
 1   comment   4132 non-null   object
 2   cleaning  4132 non-null   object
dtypes: object(3)
memory usage: 129.1+ KB
```

- Normalisasi dataset (membakukan kata yang tidak baku menggunakan **kamuskatabaku.xlsx**)

```
[68] #Baca kamus kata tidak baku
     kamus_data = pd.read_excel('../content/drive/MyDrive/Colab Notebooks/data_debat/kamuskatabaku.xlsx')
     kata_tidak_baku = dict(zip(kamus_data['tidak_baku'], kamus_data['kata_baku']))
```

```
kamus_data.head(10)
```

	tidak_baku	kata_baku
0	woww	wow
1	aminn	amin
2	met	selamat
3	netaas	menetas
4	keberpa	keberapa
5	eeeehhhh	eh
6	kata2nyaaa	kata-katanya
7	hallo	halo
8	kaka	kakak
9	ka	kak

Langkah berikutnya: Buat kode dengan kamus_data | Lihat plot yang direkomendasikan | New interactive sheet

```
#Terapkan fungsi penggantian kata tidak baku
data['normalisasi'], data['kata_baku'], data['kata_tidak_baku'], data['kata_tidak_baku_hash'] = zip(*data['cleaning'].apply(lambda x: replace_taboo_words(x, kata_tidak_baku)))

df = pd.DataFrame(data[['username', 'comment', 'normalisasi']])
df.head(10)
```

	username	comment	normalisasi
0	@HendrikWirjana	2	
1	@NaniPancing	Suka liat video pasca pilpres...02🎉	suka lihat video pasca pilpres
2	@BulanMerindu-r1f		
3	@BulanMerindu-r1f		
4	@BulanMerindu-r1f	Bismillah Ya Allah semoga Allah swt 03 1m...	bismillah ya allah semoga allah swt 1menang aa...
5	@MuhammadSalehBangko548	Daerah Aceh laut biru 25.00	daerah aceh laut biru
6	@MuhammadSalehBangko548	110 laut biru pak	laut biru pak
7	@MuhammadSalehBangko548	Dari balikpapan 21.00 laut biru	dari balikpapan laut biru
8	@MuhammadSalehBangko548	Masuk lagi laut biru 27.00 jumlahnya	masuk lagi laut biru jumlahnya
9	@MuhammadSalehBangko548	Kalimantan Timur daerah bontang 57.00	kalimantan timur daerah bontang

Langkah berikutnya: Buat kode dengan df | Lihat plot yang direkomendasikan | New interactive sheet

- Tokenization dan Stopword

```
import nltk
nltk.download('punkt_tab')
df['tokenization'] = df['normalisasi'].apply(nltk.word_tokenize)
df
```

```
[nltk_data] Downloading package punkt_tab to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt_tab.zip.
```

	username	comment	normalisasi	tokenization
0	@HendrikWirjana	2		[]
1	@NaniPancing	Suka liat video pasca pilpres...02🎉	suka lihat video pasca pilpres	[suka, lihat, video, pasca, pilpres]
2	@BulanMerindu-r1f			[]
3	@BulanMerindu-r1f			[]
4	@BulanMerindu-r1f	Bismillah Ya Allah semoga Allah swt 03 1m...	bismillah ya allah semoga allah swt 1menang aa...	[bismillah, ya, allah, semoga, allah, swt, 1me...
...	...	...	...	...
4394	@srinurhidayati3323	Alhamdulillah prabowo gibran ttp terbaik power...	alhamdulillah prabowo gibran tetap terbaik pow...	[alhamdulillah, prabowo, gibran, tetap, terbai...
4395	@DadiRohadi-gz7dp	Waw 03 minta d ulang cape aah udh jls 02 mnang	wow meminta di ulang capek aah sudah jls mnang	[wow, meminta, di, ulang, capek, aah, sudah, j...
4396	@widoyotop702	Alhamdulillah 02 masih Unggul ✌✌✌🙏🙏🙏	alhamdulillah masih unggul	[alhamdulillah, masih, unggul]
4397	@manikdee-b6k	03 minta di ulang pemilunya 🥰	meminta di ulang pemilunya	[meminta, di, ulang, pemilunya]
4398	@OziRemaja	Prabowo menang mna suara nya✌✌✌✌✌✌	prabowo menangmna suara ya	[prabowo, menangmna, suara, ya]

4399 rows × 4 columns

Langkah berikutnya: Buat kode dengan df | Lihat plot yang direkomendasikan | New interactive sheet

```
import nltk
from nltk.corpus import stopwords

# Download stopwords if not already downloaded
nltk.download('stopwords')

# Get the Indonesian stopwords
stop_words = stopwords.words('indonesian')

# Add the word 'ya' to the stopwords list
stop_words.append('ya')

# Optional: Convert the stopwords list to a set for faster lookup
stop_words = set(stop_words)
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
```

3.5.Stopword

```
def remove_stopwords(tokens):
  return [token for token in tokens if token not in stop_words]

df['stopwords'] = df['tokenization'].apply(remove_stopwords)
df
```

	username	comment	normalisasi	tokenization	stopwords
0	@HendrikWirjana	2		[]	[]
1	@NaniPancing	Suka liat video pasca pilpres...02	suka lihat video pasca pilpres	[suka, lihat, video, pasca, pilpres]	[suka, lihat, video, pasca, pilpres]
2	@BulanMerindu-r1f			[]	[]
3	@BulanMerindu-r1f			[]	[]
4	@BulanMerindu-r1f	Bismillah Ya Allah semoga Allah swt 03 1m...	bismillah ya allah semoga allah swt 1menang aa...	[bismillah, ya, allah, semoga, allah, swt, 1me...	[bismillah, allah, semoga, allah, swt, 1menang...
...	...	...	...	...	...
4394	@srinurhidayati3323	Alhamdulillah prabowo gibran ttp terbaik power...	alhamdulillah prabowo gibran tetap terbaik pow...	[alhamdulillah, prabowo, gibran, tetap, terbai...	[alhamdulillah, prabowo, gibran, terbaik, powe...
4395	@DadiRohadi-gz7dp	Waw 03 minta d ulang cape aah udh jls 02 mnang	wow meminta di ulang capek aah sudah jls mnang	[wow, meminta, di, ulang, capek, aah, sudah, j...	[wow, ulang, capek, aah, jls, mnang]
4396	@widoyotop702	Alhamdulillah 02 masih Unggul	alhamdulillah masih unggul	[alhamdulillah, masih, unggul]	[alhamdulillah, unggul]
4397	@manikdee-b6k	03 minta di ulang pemilunya	meminta di ulang pemilunya	[meminta, di, ulang, pemilunya]	[ulang, pemilunya]
4398	@OziRemaja	Prabowo menang mna suara nya	prabowo menangmna suara ya	[prabowo, menangmna, suara, ya]	[prabowo, menangmna, suara]

4399 rows × 5 columns

- Stemming

```
[79] !pip install Sastrawi

from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
from nltk.stem import PorterStemmer
from nltk.stem.snowball import SnowballStemmer
```

```
Collecting Sastrawi
  Downloading Sastrawi-1.0.1-py2.py3-none-any.whl.metadata (909 bytes)
Downloading Sastrawi-1.0.1-py2.py3-none-any.whl (209 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 209.7/209.7 kB 3.8 MB/s eta 0:00:00
Installing collected packages: Sastrawi
Successfully installed Sastrawi-1.0.1
```

```
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory

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

# Fungsi stemming
def stem_text(text):
    if isinstance(text, list):
        text = ' '.join(text)
    print("Text sebelum stemming:", text)  # Cetak teks sebelum stemming
    stemmed_text = stemmer.stem(text)
    print("Text setelah stemming:", stemmed_text)  # Cetak teks setelah stemming
    return stemmed_text

# Menerapkan stemming pada kolom 'stopwords'
df['stemming'] = df['stopwords'].apply(stem_text)
df
```

```
Text setelah stemming: detik omong jelek allah sogtunya
Text sebelum stemming: akui bilang curanglah mcm2 akui kelar repot
Text setelah stemming: aku bilang curang mcm2 aku kelar repot
Text sebelum stemming: selamat pdip blunder menjelekkan jokowi nyungsep tertinggal paslon selamat kesombonganmu
Text setelah stemming: selamat pdip blunder jelek jokowi nyungsep tinggal paslon selamat sombong
Text sebelum stemming: kalah mencari celah jiwa petarung
Text setelah stemming: kalah cari celah jiwa tarung
Text sebelum stemming: suara rakyat suara tuhan wajib menghormati pilihan rakyat suka suka bukti mayoritas rakyat memilih
Text setelah stemming: suara rakyat suara tuhan wajib hormat pilih rakyat suka suka bukti mayoritas rakyat pilih
Text sebelum stemming: perbuatan manusia setitik zarah balasannya
Text setelah stemming: buat manusia titik zarah balas
Text sebelum stemming: managawal pemungutan suara damai alhasil prabowo unggul
Text setelah stemming: managawal mungut suara damai alhasil prabowo unggul
```

- Labelling karena dataset saya belum punya Label, hanya commentar saja

```python
from textblob import TextBlob
import numpy as np
import pandas as pd

# Membaca file CSV ke dalam DataFrame
data = pd.read_csv('resulttranslated_data.csv')

sentiment = []

for text in data['translated']:
    if isinstance(text, str):
        blob = TextBlob(text)
        polarity = blob.sentiment.polarity
        sentiment.append(polarity)
    else:
        sentiment.append(np.nan)

data['sentiment_polarity'] = sentiment
data['sentiment'] = data['sentiment_polarity'].apply(lambda x: 'positive' if x > 0 else ('neutral' if x == 0 else 'negative'))

data = data[['username', 'comment', 'stemming', 'translated', 'sentiment_polarity', 'sentiment']]

# Menyimpan DataFrame ke file CSV
data.to_csv('labeled_data.csv', index=False)
```

```python
[100] import pandas as pd

# Assuming 'data' DataFrame has a 'Sentiment' column
sentiment_counts = data['sentiment'].value_counts()

# Menampilkan hasil dalam format tabel
sentiment_counts_df = sentiment_counts.reset_index()
sentiment_counts_df.columns = ['sentiment', 'count']
# Menghapus baris dengan Sentiment 'unknown'
sentiment_counts_df = sentiment_counts_df[sentiment_counts_df['sentiment'] != 'unknown']
print(sentiment_counts_df)

# Menyimpan hasil ke dalam file CSV
sentiment_counts_df.to_csv('sentiment_distribution.csv', index=False)
```

```
  sentiment  count
0  positive   1826
1   neutral   1655
2  negative    651
```

PROSES LEARNING ATAU MODELLING

A. Membagi data training dan data testing

```
[178] !pip install scikit-learn
```

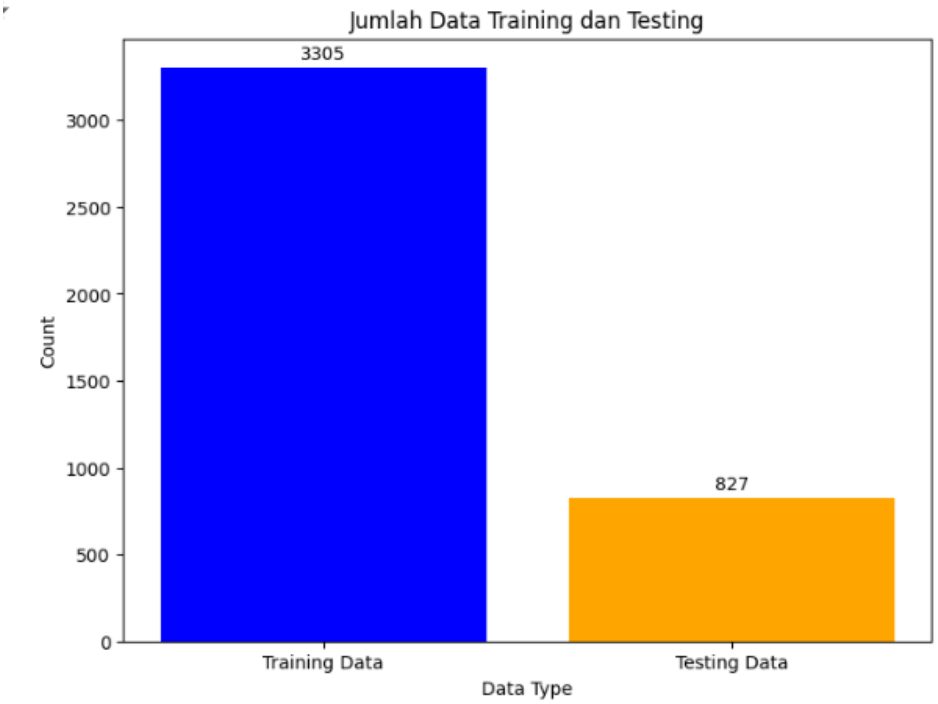
```
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.10/dist-packages (1.6.0)
Requirement already satisfied: numpy>=1.19.5 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (1.26.4)
Requirement already satisfied: scipy>=1.6.0 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (1.13.1)
Requirement already satisfied: joblib>=1.2.0 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (1.4.2)
Requirement already satisfied: threadpoolctl>=3.1.0 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (3.5.0)
```

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
# Import GridSearchCV
from sklearn.model_selection import GridSearchCV
```

```
[203] X_train, X_test, y_train, y_test = train_test_split(data['translated'], data['sentiment'], test_size=0.2, random_state=42)
```

```
[204] print(f'Jumlah data training: {len(X_train)}')
      print(f'Jumlah data testing: {len(X_test)}')
```

```
Jumlah data training: 3305
Jumlah data testing: 827
```



B. Ekstraksi Fitur untuk data training dan data testing

```
[206] # Handle NaN values in X_train and X_test
      X_train = X_train.fillna('')  # Replace NaN with empty strings in X_train
      X_test = X_test.fillna('')    # Replace NaN with empty strings in X_test

      # Update vectorizer with different parameters
      vectorizer = TfidfVectorizer(stop_words='english', max_features=5000, ngram_range=(1, 2), min_df=5, max_df=0.7)

      # Fit and transform X_train, transform X_test
      X_train_vec = vectorizer.fit_transform(X_train)
      X_test_vec = vectorizer.transform(X_test)
```

C. Hyperparameter tunning dan Trainning Model

```
[199] # Hyperparameter tuning
      param_grid = {'alpha': [0.1, 0.5, 1.0, 2.0]}
      grid_search = GridSearchCV(MultinomialNB(), param_grid, cv=5, scoring='accuracy')
      grid_search.fit(X_train_vec, y_train)
      best_model = grid_search.best_estimator_
```

```
from sklearn.naive_bayes import MultinomialNB

model = MultinomialNB()
model.fit(X_train_vec, y_train)
```

```
▸ MultinomialNB
```

D. Performa Model

```
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix

# Prediction and evaluation
predictions = best_model.predict(X_test_vec)
accuracy = accuracy_score(y_test, predictions)
print(f'Akurasi: {accuracy:.2f}')
print('\nClassification Report:\n', classification_report(y_test, predictions))
print('\nConfusion Matrix:\n', confusion_matrix(y_test, predictions))
```

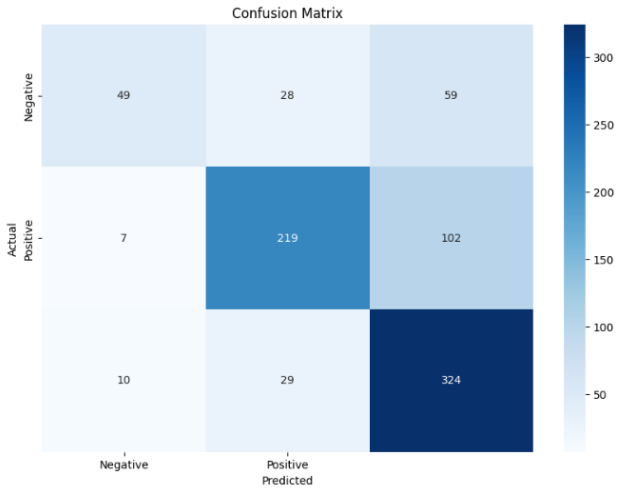
```
Akurasi: 0.72

Classification Report:
              precision    recall  f1-score   support

    negative       0.74      0.36      0.49       136
     neutral       0.79      0.67      0.73       328
    positive       0.67      0.89      0.76       363

    accuracy                           0.72       827
   macro avg       0.73      0.64      0.66       827
weighted avg       0.73      0.72      0.70       827


Confusion Matrix:
 [[ 49  28  59]
 [  7 219 102]
 [ 10  29 324]]
```



DISKUSI HASIL DAN KESIMPULAN

.