

**LAPORAN ANALISIS  
KLASIFIKASI SENTIMEN KOMENTAR DEBAT CALON  
PRESIDEN 2024 DI YOUTUBE DENGAN ALGORITMA  
K-NEAREST NEIGHBORS**

**Guswita Helmi<sup>1</sup>**

<sup>1</sup>Universitas Adzkia

[guswita helmi0211@gmail.com](mailto:guswita helmi0211@gmail.com)



**OLEH :**

**GUSWITA HELMI : 2170005**

**DISUSUN GUNA MEMENUHI UJIAN AKHIR SEMESTER**

**MATA KULIAH :**

**PENGANTAR DATA SCIENCE**

**DOSEN PENGAMPU :**

**YOMEI HENDRA, M. KOM**

**PROGRAM STUDI INFORMATIKA**

**UNIVERSITAS ADZKIA**

**PADANG**

**2024**

## **1. Pendahuluan**

Salah satu peristiwa penting dalam kehidupan sebuah negara adalah pemilihan umum, di mana warga memiliki kesempatan untuk memilih pemimpin yang mereka anggap dapat mewakili kepentingan dan keinginan mereka. Media sosial telah berkembang menjadi sarana penting untuk berbagai diskusi politik dan perdebatan dalam era internet saat ini. YouTube adalah platform populer di mana orang sering berkomentar pada video debat calon presiden dan wakil presiden untuk mengungkapkan pendapat dan pandangan mereka tentang calon presiden dan wakil presiden serta masalah politik umum.

Dalam situasi ini, elemen yang signifikan adalah analisis sentimen yang ditemukan dalam komentar yang dibuat oleh para calon presiden dan wakil presiden di YouTube. Tujuan dari analisis ini adalah untuk menemukan dan mengklasifikasikan sentimen yang terkandung dalam komentar tersebut, sehingga dapat memberikan pemahaman lebih mendalam tentang respons dan perspektif masyarakat terhadap para calon wakil presiden serta masalah yang dibahas dalam debat.

Algoritma K-Nearest Neighbors (KNN) adalah salah satu metode yang dapat digunakan untuk melakukan klasifikasi sentimen. Algoritma ini adalah metode pembelajaran mesin yang sederhana dan efektif untuk mengklasifikasikan data berdasarkan kedekatan tetangga terdekatnya. Algoritma KNN diharapkan dapat menghasilkan pola sentimen yang membantu memahami dinamika opini publik melalui komentar debat calon presiden dan wakil presiden di YouTube. Studi ini tidak hanya meningkatkan pemahaman masyarakat tentang sentimen masyarakat terhadap calon presiden dan wakil presiden dan masalah politik, tetapi juga membangun dasar untuk strategi komunikasi politik yang lebih baik. Laporan ini bertujuan untuk mendukung proses demokrasi dengan memberikan pemahaman yang lebih baik tentang dinamika opini publik selama kampanye pemilihan umum dan diskusi politik.

Berdasarkan latar belakang diatas, maka penulis mengangkat tema tentang Klasifikasi Sentimen Komentar Debat Calon Presiden 2024 Di Youtube Dengan Algoritma K-Nearest Neighbors.

## **2. Landasan Teori**

### **2.1 Analisis Sentimen**

Analisis sentimen adalah proses untuk menentukan atau mengukur nilai sentimen atau pendapat seseorang terhadap suatu objek atau peristiwa dalam teks. Sentimen dapat positif, negatif, atau netral. Banyak pengguna internet saat ini menuliskan pendapat, opini, pengalaman, dan berbagai hal lain yang mereka anggap menarik. Analisis sentimen dianggap sebagai masalah pengelompokan (Wandani, 2021). Jika komentar youtube mengandung kata-kata polar atau kata-kata berlawanan, itu dianggap positif atau negatif. Jika tidak, itu dianggap netral.

### **2.2 Youtube**

YouTube adalah platform berbagi video daring yang memungkinkan pengguna mengunggah, menonton, dan berbagi video. Didirikan pada tahun 2005, YouTube menjadi salah satu situs web paling populer di dunia dengan berbagai jenis video, termasuk video pribadi, tutorial, hingga konten hiburan dan berita.

### **2.3 Python**

Python, yang pertama kali dikembangkan oleh Guido van Rossum pada awal tahun 1990, telah menjadi salah satu bahasa pemrograman yang paling populer dan banyak digunakan dalam berbagai bidang

pengembangan perangkat lunak. Ini adalah bahasa pemrograman umum yang difokuskan pada keterbacaan kode dan sintaksis yang bersih. Dalam laporan ini menggunakan library sebagai berikut:

#### 2.4 Text Preprocessing

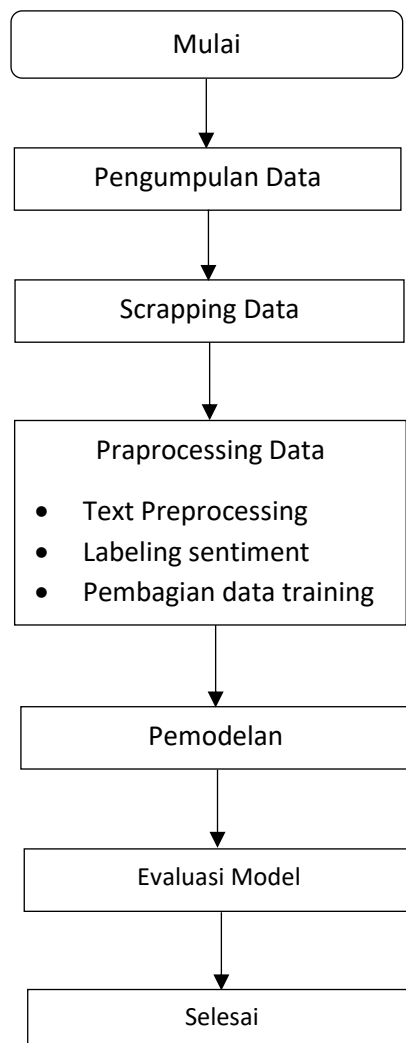
Text preprocessing adalah langkah atau serangkaian langkah yang dilakukan untuk membersihkan dan mengubah teks mentah menjadi bentuk yang lebih mudah diolah dan ditafsirkan oleh algoritma pemrosesan bahasa alami (Natural Language Processing, NLP), atau model pembelajaran mesin lainnya.

#### 2.5 Algoritma K-Nearest Neighbors

Algoritma K-Nearest Neighbors (KNN) adalah salah satu algoritma dalam bidang pembelajaran mesin yang digunakan untuk tugas klasifikasi dan regresi. Algoritma ini bekerja dengan mengukur kedekatan antara suatu data dengan data lain dalam ruang fitur. Dalam konteks klasifikasi, KNN menentukan kategori atau kelas suatu data berdasarkan mayoritas kelas dari k tetangga terdekatnya.

### 3. Metodologi Penelitian

Tahapan penelitian pada Klasifikasi Sentimen Komentar Debat Calon Wakil Presiden 2024 Di Youtube Dengan Algoritma K-Nearest Neighbors adalah sebagai berikut:



**Gambar 1.** Kerangka Penelitian

### 3.1 Scarping Data

Pertama, melakukan scrapping data komentar YouTube menggunakan YouTube API. Request dibuat ke API YouTube untuk mendapatkan data komentar dari video tertentu. Jenis data yang diambil termasuk waktu komentar, nama pengguna, isi komentar, dan jumlah like. Hasilnya, sebanyak 7517 data komentar dikumpulkan dan disimpan dalam dataframe pandas. Tahap ini sangat penting untuk mendapatkan data komentar YouTube untuk analisis sentiment.

### 3.2 Filtering dan Cleaning

Pada langkah kedua, data komentar difilter dan dibersihkan. Beberapa proses yang dilakukan termasuk menghapus mention, hashtag, dan URL link, mengubah ke huruf kecil, menghapus tanda baca, menghapus angka dan digit, melakukan tokenisasi, stemming, dan menghapus stopword. Pada tahap ini, penting untuk membersihkan data agar lebih fokus pada kata-kata penting dalam komentar yang dapat menunjukkan perasaan. Dengan data yang lebih bersih, diharapkan analisis perasaan dapat dilakukan dengan lebih baik.

### 3.3 Pembagian Data

Pada tahap ketiga, data komentar dievaluasi berdasarkan perasaan mereka: apakah itu positif, netral, atau negatif. Jumlah like pada komentar menentukan sentiment. Jika lebih dari lima like, sentiment itu positif; jika antara 3 dan 5 like, sentiment itu netral; dan jika kurang dari 3 like, sentiment itu negatif. Tujuan pemberian label ini adalah untuk melatih model untuk memprediksi perasaan pada data baru. Dalam supervised learning, labeling perasaan sangat penting karena memungkinkan model untuk belajar melalui contoh.

### 3.4 Konversi Data

Pada tahap kelima, metode TF-IDF, yang merupakan metode pembobotan kata yang didasarkan pada frekuensi kata yang muncul dalam dokumen atau komentar, digunakan untuk mengubah data teks komentar menjadi vektor. Vektorisasi data teks sangat penting agar model dapat memahami data teks, dan TF-IDF adalah salah satu metode vektorisasi yang paling populer.

### 3.5 Membuat model KNN

Tahap keenam adalah pembuatan model KNN ( $K=3$ ) dan pelatihannya menggunakan data pelatihan yang sudah di-vector-kan. Model KNN dipilih karena mudah difahami dan mudah digunakan. Saat memprediksi kelas, nilai  $K$  menunjukkan berapa banyak tetangga terdekat yang dipertimbangkan. Untuk membuat model classifier, pemodelan ini merupakan langkah penting dari analisis sentiment.

### 3.6 Evaluasi Performa Model

Tahap ketujuh adalah melakukan evaluasi kinerja model dengan data pengujian. Beberapa metrik yang digunakan untuk evaluasi adalah akurasi, presisi, recall, f1-score, dan confusion matrix. Evaluasi performa model sangat penting untuk mengetahui seberapa baik model memprediksi perasaan pada data baru. Hasil evaluasi dapat digunakan untuk memperbaiki model.

## 4. Hasil dan Pembahasan

### 1. Pengumpulan Data

Data yang digunakan adalah komentar yang didapat dari video youtube tvOneNews yang berjudul "[LIVE] Debat Cawapres 2024 (22/12/2023) | tvOne". Dalam proses crawling data menggunakan API

key sebagai kunci untuk mengakses data youtube, dan mengambil jumlah data tidak terbatas. Pada proses ini, pengumpulan data komentar dilakukan dengan teknik scrapping, yaitu pengambilan data komentar youtube yang dilakukan dengan pemrograman bahasa Python. Berikut tahap crawling data dengan menggunakan API Key

```
# api key
api_key = 'AIzaSyCx5pWDM6VDEHYiAZIRh35qhDW9gEwAr3s'
video_id = "UQtvZgKmGJo" #isikan dengan kode / ID video

# Call function
comments = video_comments(video_id)

comments
```

Data yang diambil adalah data komentar dari bulan Desember 2023 sampai dengan bulan Januari 2024.

Berikut adalah codingan untuk menampilkan tabel data Komentar youtube yang digunakan terdiri dari 7517 data, sebagai berikut :

Hasil Data	Total Data 7517
Negatif	6609
Netral	300
Positif	608

```
df = pd.DataFrame(comments, columns=['Waktu', 'Penulis', 'Komentar', 'Suka'])
df
```

	Waktu	Penulis	Komentar	Suka	
0	2024-01-22T10:55:47Z	@Caliystafdr	semangat pak prabowo gibram	0	il.
1	2024-01-22T07:45:03Z	@PutrajayaSelangor	Yg namanya buah tak kan pernah jauh jatuh dr po...	0	
2	2024-01-22T06:56:05Z	@borugultom63	Sekali Pak Jokowi tetap pak Jokowi beliau apa ...	0	
3	2024-01-21T22:59:42Z	@user-fk5js3mb4l	Tetap 02 ♥ Bapak Prabowo dan mas Gibran  La...	0	
4	2024-01-21T22:21:32Z	@Iwanyana8282	Ini bohong semua hilirisasi bukti semua proven...	0	
...	...	...	...	...	...
7512	2023-12-22T14:48:39Z	@pandiahmadjawara5983		1	4
7513	2023-12-22T14:48:39Z	@Anna-GRX	Gibran MVP	10	
7514	2023-12-22T14:48:32Z	@MarGono-hh1xf	Mahfud MD jitu	0	
7515	2023-12-22T14:48:22Z	@mr_kesai	SAMSUL blunder <a href="about:invalid#zCSafez"...	3	
7516	2023-12-22T14:48:02Z	@KusnulChotimah-fg4ev		2	0

7517 rows x 4 columns

## 2. Data Preprocessing

Data preprocessing adalah proses pembersihan data untuk mengurangi noise agar lebih mudah dianalisis. Prosesnya adalah sebagai berikut :

2.1 Pada tahap awal pengolahan data, proses scraping data komentar YouTube menggunakan YouTube API dilakukan. Untuk melakukan ini, kode Python menggunakan ID video tertentu untuk mengakses YouTube API dan mengumpulkan data komentar video. Proses scraping ini sangat penting untuk

mendapatkan kumpulan data komentar YouTube untuk analisis sentiment karena mencakup 7517 komentar, yang mencakup kolom waktu, nama pengguna, teks, dan jumlah like.

- 2.2 Pada tahap ketiga, data komentar dilabeli berdasarkan jumlah like dalam kolom untuk mendapatkan label sentiment positif, netral, atau negatif. Aturan untuk labeling adalah sebagai berikut, jika jumlah like lebih dari lima, komentar dilabeli positif, jika antara 3 dan 5 dilabeli netral, dan jika kurang dari 3 dilabeli negatif. Labeling sentiment ini sangat penting karena ini merupakan target variable yang akan diprediksi oleh model. Setelah labeling, proporsi datanya adalah 6609 negatif, 608 positif, dan 300 netral. Saat modeling dilakukan, proposi yang tidak seimbang ini harus diperbaiki.

	Waktu	Penulis	Komentar	Suka	Sentiment	
0	2023-12-22T14:48:02Z	@KusnuCholimah-fg4ev		2	0	Negatif
1	2023-12-22T14:48:22Z	@mr_kesal	SAMSUL blunder <a href="about:invalid#zCSafez"...	3		Netral
2	2023-12-22T14:48:32Z	@MarGono-hh1xf	Mahfud MD jitu	0		Negatif
3	2023-12-22T14:48:39Z	@Anna-GRX	Gibran MVP	10		Positif
4	2023-12-22T14:48:39Z	@pandiahmadjawara5983		1	4	Netral

- 2.3 Tahap ketiga, menghapus variabel yang tidak dipakai yaitu menghilangkan variabel waktu dan dan penulisan dalam data komentar.

	Komentar	Suka	Sentiment
0	[]	0	Negatif
1	[samsul, blunder, a, hrefaboutinvalid]	3	Netral
2	[mahfud, md, jitu]	0	Negatif
3	[gibran, mvp]	10	Positif
4	[]	4	Netral

- 2.4 Dalam tahap keempat, data akan digunakan untuk membangun model, sedangkan data pemeriksaan akan digunakan untuk mengevaluasi model. Pembagian data menjadi training dan testing dilakukan dengan proporsi 70% hingga 30%. Ini penting untuk menghindari overfitting saat modeling.

- 2.5 Filtering, dalam konteks umum, "filtering" merujuk pada proses memilih atau menyaring elemen tertentu dari suatu kumpulan berdasarkan kriteria tertentu. Dalam pemrograman, termasuk dalam Python, "filtering" sering mengacu pada penggunaan fungsi atau metode yang memungkinkan Anda memilih elemen tertentu dari suatu kumpulan data.

```
!pip install Sastrawi
Collecting Sastrawi
  Downloading Sastrawi-1.0.1-py2.py3-none-any.whl (209 kB)
    209.7/209.7 kB 3.4 MB/s eta 0:00:00
Installing collected packages: Sastrawi
Successfully installed Sastrawi-1.0.1
```

- 2.6 Pada tahap kedua, persiapan teks dilakukan dengan beberapa metode. Pertama, menggunakan regex untuk menghapus link URL dari teks komentar karena URL dianggap tidak relevan. Kedua, untuk menggabungkan format huruf, ubah semua karakter komentar menjadi lowercase. Ketiga, mention dari akun lain yang diawali dengan tanda @ dihapus karena dianggap tidak relevan. Keempat, hashtag yang diawali dengan tanda # dihapus karena dianggap tidak relevan. Kelima, gunakan regex untuk menghapus tanda baca seperti titik, koma, dan seru. Keenam, jika komentar dianggap tidak relevan, hapus angka atau digit. Ketujuh, tokenisasi, yang membagi komentar menjadi satu kata per kata. Kedelapan, stemming, yang menghilangkan awalan dan akhiran kata untuk mendapatkan kata dasar.

Kesembilan, menghapus stopwords, seperti kata hubung, kata depan, dan kata sambung. Tujuan dari tahap preprocessing teks ini adalah untuk membersihkan data sehingga lebih banyak perhatian dapat diberikan pada kata-kata komentar penting yang menggambarkan perasaan.

```
#clearing
#remove url
df01_data['Komentar'] = df01_data['Komentar'].str.replace('https\S', "", case=False) #tokenizing

#ubah teks jadi huruf kecil
df01_data['Komentar'] = df01_data['Komentar'].str.lower() #case folding

#remove mention
df01_data['Komentar'] = df01_data['Komentar'].str.replace('@\S+', "", case=False) #tokenizing

#remove hastag
df01_data['Komentar'] = df01_data['Komentar'].str.replace('#\S+', "", case=False) #tokenizing

#remove next character
df01_data['Komentar'] = df01_data['Komentar'].str.replace("\W+", "", case=False) #tokenizing

#remove punctuation
df01_data['Komentar'] = df01_data['Komentar'].str.replace('[^\w\s]', "", case=False) #tokenizing

#remove number
df01_data['Komentar'] = df01_data['Komentar'].str.replace(r'\d+', "", case=False) #tokenizing

#remove spasi berlebih
df01_data['Komentar'] = df01_data['Komentar'].str.replace("\s(2)", "", case=False) #tokenizing
```

### 3. Analisis Sentimen

3.1 Pertama, analisis komentar adalah sebagai berikut.

	Komentar	Suka	Sentiment
0		0	Negatif
1	[samsul, blunder, a, hrefaboutinvalid]	3	Netral
2	[mahfud, md, jitu]	0	Negatif
3	[gibran, mvp]	10	Positif
4		4	Netral

3.2 Tahap kedua, metode TF-IDF (Term Frequency Inverse Document Frequency) adalah teknik pembobotan kata yang populer digunakan untuk representasi teks untuk keperluan pembelajaran mesin. TF-IDF memberikan bobot yang lebih tinggi pada kata yang sering muncul dalam satu dokumen atau komentar tetapi jarang muncul di dokumen atau komentar lainnya. Proses vektorisasi menghasilkan matri yang sparse. Tahap ini dilakukan untuk memastikan bahwa model dapat memahami data teks komentar.

#### TD-IDF Vectorizer

```

      00 000 01 01h10m20s 01semua 01yes      02 02h30m29s 03 04 \
0      0.0 0.0 0.0      0.0 0.0 0.0 0.0 0.00000 0.0 0.0 0.0
1      0.0 0.0 0.0      0.0 0.0 0.0 0.0 0.00000 0.0 0.0 0.0
2      0.0 0.0 0.0      0.0 0.0 0.0 0.0 0.00000 0.0 0.0 0.0
3      0.0 0.0 0.0      0.0 0.0 0.0 0.0 0.00000 0.0 0.0 0.0
4      0.0 0.0 0.0      0.0 0.0 0.0 0.0 0.00000 0.0 0.0 0.0
...    ...    ...    ...    ...    ...    ...    ...    ...    ...
7512   0.0 0.0 0.0      0.0 0.0 0.0 0.0 0.00000 0.0 0.0 0.0
7513   0.0 0.0 0.0      0.0 0.0 0.0 0.0 0.28626 0.0 0.0 0.0
7514   0.0 0.0 0.0      0.0 0.0 0.0 0.0 0.00000 0.0 0.0 0.0
7515   0.0 0.0 0.0      0.0 0.0 0.0 0.0 0.00000 0.0 0.0 0.0
7516   0.0 0.0 0.0      0.0 0.0 0.0 0.0 0.00000 0.0 0.0 0.0

```

```

      ... zm4ng4zj9runtung zong zongk *gibran Ya deh nya senior \
0      ...      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
1      ...      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
2      ...      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
3      ...      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
4      ...      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
...    ...    ...    ...    ...    ...    ...    ...    ...
7512   ...      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
7513   ...      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
7514   ...      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
7515   ...      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
7516   ...      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

```

```

      tumbang Prabowo_Gibran
0      0.0      0.0
1      0.0      0.0
2      0.0      0.0
3      0.0      0.0
4      0.0      0.0
...    ...    ...
7512   0.0      0.0
7513   0.0      0.0
7514   0.0      0.0
7515   0.0      0.0
7516   0.0      0.0

```

[7517 rows x 12559 columns]

### 3.3 Matriks TF-IDF untuk data latih

Matriks TF-IDF data pelatihan:

```

[[0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 ...
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]]

```

### 3.4 Matriks TF-IDF untuk data uji

Matriks TF-IDF data uji:

```

[[0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 ...
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]]

```



3.5 Pada langkah kedua, klasifikasi sentiment dilakukan dengan menggunakan model K-Nearest Neighbors (KNN) dengan nilai K=3. Model KNN bekerja dengan mencari jarak terdekat antara data input dan K tetangga terdekat pada data latih, lalu menggunakan voting dari K tetangga tersebut untuk memprediksi kelas data input. Nilai K=3 memastikan bahwa model tidak terlalu kompleks namun tetap mempe Model KNN ini dilatih dengan data latih TF-IDF yang sudah divektorisasi.

```
# Pembuatan model
classifier = KNeighborsClassifier(n_neighbors=3)
classifier.fit(X_train, y_train)
```

3.6 Pada tahap ketiga, performa model KNN yang dilatih dievaluasi menggunakan metrik seperti akurasi, presisi, recall, skor F1, dan matrix confusion. Hasilnya menunjukkan akurasi yang cukup baik, yaitu 86,84%, tetapi nilai akurasi, recall, dan skor F1 masih rendah. Ini menunjukkan bahwa model belum berfungsi dengan baik, terutama dalam memprediksi kelas minoritas (netral dan positif). Selain itu, matrix kekacauan menunjukkan bahwa ada banyak kesalahan prediksi yang dilakukan oleh kelas minoritas tersebut.

a. Akurasi

Akurasi adalah ukuran sejauh mana suatu hasil atau pengukuran mendekati nilai yang sebenarnya atau benar. Dalam hal umum, akurasi digunakan untuk mengevaluasi sejauh mana suatu prediksi atau hasil sesuai dengan keadaan atau nilai yang benar. Sebuah formula dapat digunakan untuk menghitung akurasi:

$$\text{Akurasi} = \frac{\text{Jumlah prediksi benar}}{\text{Total Jumlah prediksi}} \times 100\%$$

```
# Hitung akurasi
accuracy_score = metrics.accuracy_score(y_test, y_pred)
accuracy_score = round(accuracy_score * 100, 2)
print('Accuracy: ' + str(accuracy_score) + '%')
```

```
Accuracy: 86.84%
```

b. Presisi

Dalam statistika dan pembelajaran mesin, presisi adalah metrik evaluasi yang digunakan untuk mengukur sejauh mana suatu model memberikan prediksi positif yang benar dibandingkan dengan total prediksi positif yang diberikan oleh model. Sebuah formula dapat digunakan untuk menghitung presisi:

$$\text{Presisi} = \frac{\text{Jumlah true positive}}{\text{Jumlah true positive} + \text{jumlah false positive}} \times 100\%$$

```
#menghitung presisi
macro_precision = (metrics.precision_score(y_test, y_pred, average='macro'))
macro_precision = round(macro_precision * 100, 2)
print('Precision : ' + str(macro_precision) + '%')
```

```
Precision : 45.23%
```

c. Recall

Dalam statistika dan pembelajaran mesin, recall, juga disebut sensitivitas atau rate positif asli, adalah metrik evaluasi yang mengukur sejauh mana suatu model dapat mengidentifikasi atau

mendeteksi semua instance positif yang sebenarnya dalam dataset. Recall dapat dihitung dengan menggunakan formula berikut:

$$\text{Presisi} = \frac{\text{Jumlah true positive}}{\text{jumlah true positive} + \text{jumlah false positive}} \times 100\%$$

```
#menghitung recall
macro_recall = (metrics.recall_score(y_test, y_pred, average='macro'))
macro_recall = round(macro_recall * 100, 2)
print('Recall : '+str(macro_recall) + '%')

Recall : 33.85%
```

d. F-1 Score

F-1 score adalah metrik evaluasi yang menyatukan presisi (precision) dan recall dalam satu nilai. Ini berguna terutama ketika ada trade-off antara presisi dan recall dan kita ingin memperoleh keseimbangan antara keduanya. Untuk menghitung skor F1, gunakan rumus berikut:

$$F1 - Score = 2 \times \left( \frac{\text{Presisi} \times \text{Recall}}{\text{Presisi} + \text{Recall}} \right)$$

```
#menghitung f1-score
macro_f1 = metrics.f1_score(y_test, y_pred, average='macro')
macro_f1_percentage = round(macro_f1 * 100, 2)
print('F1 : ' + str(macro_f1_percentage) + '%')

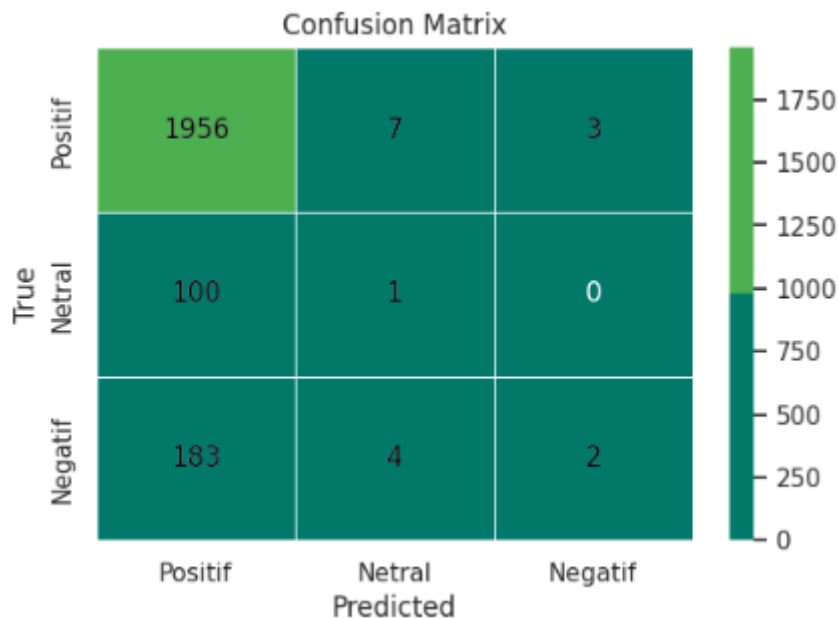
F1 : 32.29%
```

e. Tampilan hasil confusion Matrix

```
Confusion Matrix:
[[1956   7    3]
 [ 100    1    0]
 [ 183    4    2]]
```

```
Classification Report:
```

	precision	recall	f1-score	support
Negatif	0.87	0.99	0.93	1966
Netral	0.08	0.01	0.02	101
Positif	0.40	0.01	0.02	189
accuracy			0.87	2256
macro avg	0.45	0.34	0.32	2256
weighted avg	0.80	0.87	0.81	2256



## 5. Kesimpulan

Berdasarkan hasil dari pembahasan tentangnng klasifikasi sentimen komentar debat calon wakil presiden 2024 di youtube dengan algoritma K-nearest neighbors tersebut, maka dapat disimpulkan sebagai berikut:

1. Penelitian ini menggunakan machine learning untuk mengklasifikasikan perasaan (positif, netral, atau negatif) dalam komentar YouTube. Dataset yang digunakan adalah 7517 komentar YouTube yang diambil dari video tertentu menggunakan YouTube API.
2. Tahapan analisis yang dilakukan meliputi pemrosesan data, pengolahan teks sebelum pengolahan, labeling perasaan, pengujian pembagian data instruksional, vektorisasi TF-IDF, klasifikasi KNN, dan evaluasi model menggunakan ketepatan, ketepatan, recall, skor F1, dan matriks kekacauan.
3. Hasil penelitian menunjukkan bahwa model KNN memiliki akurasi yang cukup tinggi, 86,84%, tetapi nilai ketepatan, recall, dan F1 masih rendah, menunjukkan bahwa model belum bekerja dengan baik dalam memprediksi kelas minoritas. Untuk kelas positif dan netral, terlihat banyak kesalahan prediksi, tetapi untuk kelas negatif, prediksinya akurat. Ketidakseimbangan data didominasi kelas negatif.
4. Secara keseluruhan, model KNN yang dibangun cukup baik sebagai dasar untuk klasifikasi sentimen komentar YouTube; namun, masih ada beberapa hal yang perlu diperbaiki, terutama dalam hal menangani kelas yang tidak seimbang agar analisis sentimennya lebih efisien.

## DAFTAR RUJUKAN

- [1] Ahluna, F., Tutuarima, C. J., & Santoso, I. (2023). Metode K-Nearest Neighbor Untuk Analisis Sentimen Tentang Penghapusan Ujian Nasional. *IKRA-ITH Informatika: Jurnal Komputer dan Informatika*, 7(2), 170-175.
- [2] Hasanah, S., Purwasih, I., & Santoso, I. (2023). Analisis Sentimen Terhadap Masyarakat Adanya Uang Kertas Baru Menggunakan Algoritma K-Nearest Neighbor (Knn). *IKRA-ITH Informatika: Jurnal Komputer dan Informatika*, 7(2), 105-114.

- [3] Kurnianto, D. D., & Waluyo, S. (2023, October). Analisis Sentimen Masyarakat Terhadap Eks Pejabat Pajak Diperiksa Kpk Pada Youtube Menggunakan Metode K-Nearest Neighbor. In *Prosiding Seminar Nasional Mahasiswa Fakultas Teknologi Informasi (SENAFTI)* (Vol. 2, No. 2, pp. 632-641).
- [4] Larasakti, D. N., Aziz, A., & Aditya, D. (2023). Analisis Sentimen Komentar Video Youtube Dengan Metode K-Nearest Neighbor. *Jurnal Ilmiah Wahana Pendidikan*, 9(5), 132-142.
- [5] Pasaribu, F. B., Abulide, T., & Santoso, I. (2023). Analisis Sentimen Tentang Rencana Penghapusan Daya Listrik 450 Va Menjadi 900 Va Oleh Pemerintah Menggunakan Algoritma K-Nearest Neighbors. *IKRA-ITH Informatika: Jurnal Komputer dan Informatika*, 7(2), 90-97.
- [6] Rohmansyah, F. A., Bintoro, B., & Santoso, I. (2023). Analisis Sentimen Terhadap Penerapan Sistem Ganjil Genap Menggunakan Metode K-Nearest Neighbor (KNN). *IKRA-ITH Informatika: Jurnal Komputer dan Informatika*, 7(2), 165-169.
- [7] Septian, J. A., Fachrudin, T. M., & Nugroho, A. (2019). Analisis Sentimen Pengguna Twitter Terhadap Polemik Persepakbolaan Indonesia Menggunakan Pembobotan TF-IDF dan K-Nearest Neighbor. *INSYST: Journal of Intelligent System and Computation*, 1(1), 43-49.
- [8] Septiansyah, A., Yuliawati, A., & Santoso, I. (2023). Analisis Sentimen Terhadap Layanan Internet Di Indonesia Menggunakan Metode K-Nearest Neighbor. *IKRA-ITH Informatika: Jurnal Komputer dan Informatika*, 7(2), 98-104.
- [9] Tursina, M. J. (2019). *Sentimen analisis sistem zonasi sekolah pada media sosial youtube menggunakan metode k-nearest neighbor dengan algoritma levenshtein distance* (Bachelor's thesis, Fakultas Sains dan Teknologi Universitas Islam Negeri Syarif Hidayatullah Jakarta).
- [10] Tursina, M. J. (2019). *Sentimen analisis sistem zonasi sekolah pada media sosial youtube menggunakan metode k-nearest neighbor dengan algoritma levenshtein distance* (Bachelor's thesis, Fakultas Sains dan Teknologi Universitas Islam Negeri Syarif Hidayatullah Jakarta).