

Journal Artificial Intelligence, Multimedia, and Mobile Technology (AI2MTech)

Homepage jurnal: <https://journal.darmajaya.ac.id/index.php/AI2MTech>

Penerapan Metode K-Nearest Neighbor Untuk Mendeteksi Ujaran Kebencian Dalam Postingan Twitter

Yuda Septiawan^a, Chirani^b

^aInstitut Informatika dan Bisnis Darmajaya, Jl. Za Pagar Alam No.93 Labuan Ratu, Bandar Lampung, Indonesia

^bInstitut Informatika dan Bisnis Darmajaya, Jl. Za Pagar Alam No.93 Labuan Ratu, Bandar Lampung, Indonesia

Corresponding author: Chairani@darmajaya.ac.id

ARTICLE INFO

Article history:

Submitted 14 August 2024

Received 15 August 2024

Received in revised form 09 September 2024

Accepted 15 September 2024

Available online on 27 September 2024

Keywords:

Sentiment analysis, K-Nearest Neighbor

Kata kunci:

Analisis Sentimen. K-Nearest Neighbor

ABSTRACT

This research focuses on evaluating the effectiveness of the K-Nearest Neighbor (KNN) method in detecting hate speech on social media platforms, particularly Twitter. The tweet data was collected using the Twitter API and labelled using the Sentistrength method to determine sentiment polarity. Next, the KNN method is applied to classify tweets based on sentiment using two dataset sharing ratios, namely 90:10 and 80:20 for training and test data. The test results show that the performance of KNN in detecting hate speech is less than optimal. At a dataset division ratio of 90:10, this algorithm produces an accuracy of 60.94% with an F-measure value of 62%. Meanwhile, at a ratio of 80:20, the accuracy increases to 63.02% with an F-measure of 63%. Based on these results, it can be concluded that the KNN method has a relatively low accuracy rate and inconsistent ability to classify hate speech data on Twitter, especially in terms of the balance between precision and recall.

ABSTRACT

Penelitian ini berfokus pada evaluasi efektivitas metode K-Nearest Neighbor (KNN) dalam mendeteksi ujaran kebencian pada platform media sosial, khususnya Twitter. Data tweet dikumpulkan menggunakan API Twitter dan diberi label menggunakan metode Sentistrength untuk menentukan polaritas sentimen. Selanjutnya, metode KNN diterapkan untuk mengklasifikasikan tweet berdasarkan sentimen dengan menggunakan dua rasio pembagian dataset, yaitu 90:10 dan 80:20 untuk data latih dan data uji. Hasil pengujian menunjukkan bahwa performa KNN dalam mendeteksi ujaran kebencian kurang optimal. Pada rasio pembagian dataset 90:10, algoritma ini menghasilkan akurasi sebesar 60,94% dengan nilai F-measure 62%. Sedangkan pada rasio 80:20, akurasi meningkat menjadi 63,02% dengan F-measure sebesar 63%. Berdasarkan hasil ini, dapat disimpulkan bahwa metode KNN memiliki tingkat akurasi yang relatif rendah serta kemampuan yang kurang konsisten dalam mengklasifikasikan data ujaran kebencian di Twitter, khususnya dalam hal keseimbangan antara presisi dan recall.

1. PENDAHULUAN

Perkembangan teknologi saat ini telah membawa kemudahan bagi masyarakat dalam mengakses informasi dan bertransaksi secara online. Di Indonesia, jumlah pengguna internet terus meningkat, mencapai 77,02% pada tahun 2021-2022, menurut APJII. Alasan utama penggunaan internet adalah untuk komunikasi melalui pesan instan dan media sosial [1]. Media sosial menjadi platform yang efektif dalam menyebarkan informasi kepada khalayak luas melalui berbagai bentuk konten seperti suara, gambar, dan video. Meskipun demikian, media sosial juga dapat disalahgunakan untuk menyebarkan konten negatif oleh pihak-pihak tertentu. Hal ini telah menjadi perhatian serius, seperti yang diungkapkan dalam konferensi pers Kementerian Komunikasi dan Informatika Republik Indonesia mengenai penggunaan media sosial untuk menyebarkan konten negative [2].

Konten negatif, khususnya ujaran kebencian, merupakan masalah serius yang muncul di media sosial. Menurut UU ITE dan Komisi Nasional Hak Asasi Manusia, ujaran kebencian adalah tindakan yang disengaja dan melanggar hukum, bertujuan untuk menyebarkan kebencian dan permusuhan terhadap individu atau kelompok tertentu berdasarkan suku, agama, dan ras [3]. Kebebasan berpendapat yang berlebihan di media sosial berpotensi disalahgunakan untuk menyebarkan kebencian dan memicu permusuhan. Aktivitas kebencian dapat berupa upaya membungkam orang lain, menindas, atau menimbulkan perasaan negatif seperti benci, malu, atau bersalah. Kebebasan untuk membenci juga mendorong berkembangnya narasi sektarian dan rasis.

Banyak peneliti yang fokus mendeteksi ujaran kebencian di media sosial, khususnya di platform seperti Twitter dan Instagram, untuk menekan penyebaran konten negatif. Penelitian ini melibatkan pembuatan data, klasifikasi ujaran kebencian, dan analisis sentimen. Namun, model pendekripsi ujaran kebencian masih memiliki kelemahan, yaitu kesulitan dalam mendekripsi kode-kode kebencian baru yang digunakan oleh masyarakat. Penelitian ini akan menguji deteksi teks ujaran kebencian pada postingan Twitter #indonesia menggunakan algoritma K-Nearest Neighbor. Data akan dikumpulkan menggunakan API Twitter dan diimplementasikan dalam bahasa pemrograman Python. Data tweet yang diperoleh kemudian akan dilabeli menggunakan SentiStrength, yang juga diimplementasikan dalam Python, untuk menentukan sentimen dari setiap tweet. Setelah data diberi label, klasifikasi menggunakan metode K-Nearest Neighbor akan diterapkan untuk mengklasifikasikan data dan menentukan sentimenya.

Hasil klasifikasi ini akan menjadi dasar untuk menentukan apakah postingan Twitter tersebut mengandung ujaran kebencian atau tidak. Metode ini diharapkan dapat membantu dalam mendekripsi dan mengurangi penyebaran ujaran kebencian di platform Twitter.

2. METODE PENELITIAN

Penelitian ini menggunakan metode eksperimen untuk menguji deteksi ujaran kebencian pada postingan Twitter. Tahapan penelitiannya tergambar dalam diagram alur penelitian (Gambar 1), yang dapat diringkas sebagai berikut:



Gambar 1 Alur Penelitian

Dengan mengikuti tahapan ini, penelitian ini bertujuan untuk mengembangkan model deteksi ujaran kebencian yang efektif dan memiliki tingkat akurasi tinggi.

2.1. Ujaran Kebencian

Ujaran kebencian (hate speech) adalah bentuk komunikasi yang mengandung kebencian, menyerang, dan bernada provokatif, yang bertujuan untuk memicu dampak tertentu, baik secara langsung (aktual) maupun tidak langsung (hanya pada niat). Tujuannya adalah untuk menginspirasi orang lain untuk melakukan kekerasan atau menyakiti orang atau kelompok lain, berdasarkan Buku saku penanganan ujaran kebencian[3].

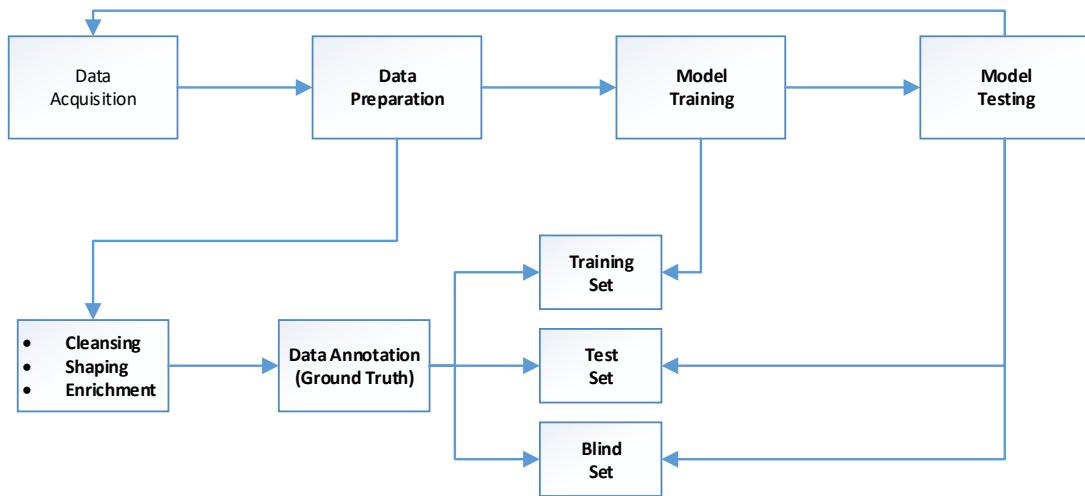
Dalam konteks hukum, hate speech merujuk pada ucapan, perilaku, tulisan, atau pertunjukan yang dilarang karena berpotensi memicu tindakan kekerasan dan prasangka, baik dari pihak pelaku maupun korban. Situs internet yang menggunakan atau menyebarkan hate speech disebut hate site. Situs ini sering menggunakan forum internet dan berita untuk mempromosikan sudut pandang tertentu yang penuh kebencian.

2.2. Machine Learning

Machine learning merupakan teknik atau ilmu yang menggunakan algoritma dan statistik komputasi untuk belajar dari data tanpa perlu diprogram secara eksplisit. Teknik ini memungkinkan computer untuk menemukan pola dalam data yang kompleks, bahkan yang tersembunyi, dan kemudian menggunakan untuk memprediksi peristiwa di masa depan atau membuat keputusan yang kompleks [4].

Machine learning dikategorikan menjadi tiga jenis. Supervised learning yang menggunakan data berlabel untuk melatih model yang kemudian dapat digunakan untuk memprediksi data baru. Jenis masalahnya bisa berupa klasifikasi (menentukan kategori) atau regresi (memprediksi nilai numerik). Unsupervised learning yang menggunakan data yang tidak berlabel untuk menemukan pola dan struktur tersembunyi dalam data. Dan reinforcement learning yang menggunakan sistem yang belajar melalui

trial and error, di mana sistem diberi hadiah atau hukuman berdasarkan tindakannya. Gambar 2 menunjukkan diagram alur umum dalam machine learning.



Gambar 2 Typical Machine Learning Flow Diagram

2.3. Text Mining

Menurut Mailoa[5], Text mining adalah salah satu teknik yang dapat digunakan untuk melakukan klasifikasi dokumen, clustering, information extraction, analisis sentimen dan information retrieval dimana text mining merupakan variasi dari data mining yang berusaha menemukan pola yang menarik dari sekumpulan data tekstual yang berjumlah besar. Text mining dapat memberikan solusi dari permasalahan seperti pemrosesan, pengorganisasian atau pengelompokan dan menganalisa unstructured text dalam jumlah besar. Dalam memberikan solusi, text mining mengadopsi dan mengembangkan banyak teknik dari bidang lain, seperti Data mining, Information Retrieval, Statistik dan Matematik, Machine Learning, Linguistic, Natural Languange Processing (NLP), dan Visualization.

2.4. Analisis Sentimen

Analisis Sentimen (opinion mining) merupakan suatu bidang yang luas dari pengolahan bahasa alami, komputasi linguistik dan text mining dimana memiliki tujuan dalam menganalisa pendapat, sentimen, evaluasi, sikap, penilaian dan emosi seseorang apakah pembicara atau penulis berkenaan dengan suatu topik, produk, layanan, organisasi, individu, ataupun kegiatan tertentu, Kolkata dkk[6]. Analisis Sentimen atau bisa disebut dengan opinion mining juga dapat memahami proses, mengekstrak dan mengolah data tekstual secara otomatis untuk mendapatkan informasi sentimen yang terkandung dalam suatu kalimat opini.

2.5. K-Nearest Neighbor

Menurut Farokhah[7], *K-Nearest Neighbor* (KNN) merupakan langkah pengelompokan yang dapat dikatakan sederhana untuk memisahkan suatu citra dengan melihat kedekatan dengan citra tetangganya.

Algoritma K-Nearest Neighbor sangat cocok untuk memperkirakan peluang apa yang akan terjadi selanjutnya menggunakan kasus-kasus yang sudah ada. Dengan metode K-Nearest Neighbor maka akan sangat cocok dalam pengambilan keputusan berdasarkan kemiripan dengan kasus-kasus terdahulunya.

Tahapan Metode K-Nearest Neighbor, Langkah yang digunakan dalam metode *K-Nearest Neighbor*:

- Tentukan parameter K (jumlah tetangga paling dekat).
 - Hitung kuadrat jarak euclid masing – masing objek terhadap data sampel yang diberikan.
 - Urutkan objek – objek kedalam kelompok yang memiliki jarak terkecil.
 - Kumpulkan kategori Y (Klasifikasi nearest neighbor).
 - Dengan kategori nearest neighbor yang paling banyak, maka dapat diprediksikan nilai query

instance yang telah dihitung.

K-Nearest Neighbor dirumuskan sebagai berikut:

$$\cos(\theta_{ij}) = \frac{\sum_k(dik \cdot dij)}{\sqrt{\sum_k d^2 ik} \sqrt{\sum_k d^2 jk}} \quad (1)$$

Keterangan:

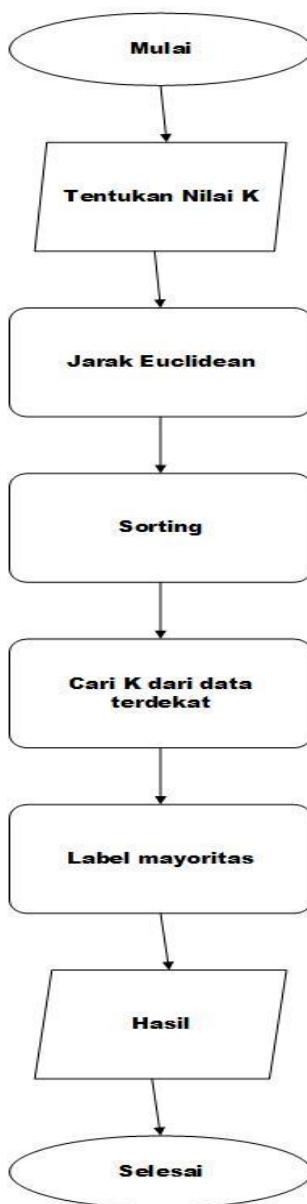
$\text{Cos}(\theta_{ij})$: Similarity <i>K-Nearest Neighbor</i>
dik	: bobot Data dokumen <i>Testing</i>
dij	: bobot Data dokumen <i>Training</i>
d^2	: Panjang Vektor dokumen

Bobot Label pada Dataset

1 = Positif

2 = Netral

3 = Negatif



Gambar 3 Flowchart *K-Nearest Neighbor*

2.6. Twitter

Twitter merupakan sosial media masif yang berubah menjadi situs berbagi informasi dan berkomunikasi secara cepat. Kecepatan dan kemudahan *Twitter* dalam hal publikasi, membuat *Twitter* menjadi sebuah medium pilihan bagi pengguna untuk berkomunikasi setiap hari. *Twitter* mempunyai peran dan andil penting dalam pergerakan sosial-politik seperti *Arab Spring* dan *The Occupy Wall Street movement*. *Twitter* juga dapat digunakan untuk melakukan laporan kerusakan dan persiapan informasi terkait bencana pada saat bencana alam akan dan sedang terjadi, Suryono[8].

a) *Glosarium Twitter*

Glosarium Twitter berisi kosakata dan istilah yang sering digunakan untuk membahas fitur dan aspek dari *Twitter*. Berikut ini merupakan kosakata pada *Twitter* berdasarkan *Support*.

b) *Streaming API*

Streaming API merupakan fitur pada *Twitter* yang membantu *developer* untuk melakukan akses secara langsung ke dalam *stream global Twitter* dengan *latency* yang rendah, sehingga memudahkan *developer* untuk melakukan pengambilan data. Beberapa tipe *endpoint* dalam *Streaming API* berdasarkan *Support Twitter*.

2.7. Klasifikasi

Klasifikasi adalah urutan yang sangat penting dalam data komunitas pertambangan. Klasifikasi adalah salah satu prediksi teknik data mining yang membuat prediksi tentang data nilai menggunakan hasil yang diketahui yang ditemukan dari kumpulan data yang berbeda. Masalah akurasi dari banyak algoritma klasifikasi adalah diketahui mengalami penurunan informasi saat dihadapi dengan data yang tidak seimbang, misalnya ketika distribusi sampel lintas kelas sangat miring. Dalam klasifikasi, ada variabel kategoris target, seperti braket pendapatan, yang, misalnya, dapat dipartisi menjaditiga kelas atau kategori: berpenghasilan tinggi, menengah pendapatan, dan pendapatan rendah. Menurut Apostolakis[9], model data mining memeriksa satu set besar catatan, masing-masing catatan yang berisi informasi tentang variabel target serta satu set input atau prediktor variable. Contoh tugas klasifikasi dalam bisnis dan penelitian meliputi:

- a) Menentukan apakah transaksi kartu kredit tertentu adalah penipuan
- b) Menempatkan siswa baru pada jalur tertentu yang berkaitan dengan kebutuhan khusus
- c) Menilai apakah aplikasi hipotek adalah risiko kredit yang baik atau buruk
- d) Mendiagnosis apakah ada penyakit tertentu
- e) Menentukan apakah surat wasiat ditulis oleh almarhum yang sebenarnya, atau curang oleh orang lain
- f) Mengidentifikasi apakah perilaku keuangan atau pribadi tertentu menunjukkan kemungkinan ancaman teroris

2.8. Lexicon

Menurut Al-Shabi[10], lexicon adalah kumpulan fitur seperti kata-kata dan klasifikasi emosi untuk setiap kata. Metode analisis sentimen ini didasarkan pada perbandingan kata-kata yang digunakan dalam teks dengan salah satu lexicon yang telah dipersiapkan sebelumnya. Lexicon juga dapat berupa kumpulan kata-kata yang diberi label atau skor sentimen, seperti positif, negatif, atau netral. Fungsi utama lexicon adalah sebagai panduan untuk mengidentifikasi sentimen atau emosi yang terkandung dalam teks yang dianalisis. Sentimen analisis teks bertujuan untuk menentukan perasaan atau pendapat yang diungkapkan oleh penulis teks, dan lexicon menjadi salah satu komponen penting dalam memahami dan mengklasifikasikan sentimen.

Lexicon dalam sentimen analisis teks adalah kumpulan kata-kata yang diberi label sentimen (misalnya positif, negatif, atau netral). Lexicon ini berfungsi sebagai panduan untuk mengidentifikasi dan mengekstraksi sentimen atau emosi yang terkandung dalam teks yang dianalisis. Penerapan lexicon dalam sentimen analisis teks bergantung pada bahasa yang digunakan, sehingga untuk bahasa Indonesia, kita menggunakan lexicon bahasa Indonesia.

2.10. Akurasi

Menurut Wibawa[11], akurasi adalah salah satu metrik untuk mengevaluasi model klasifikasi. Secara informal, akurasi adalah sebagian kecil dari prediksi model kami yang benar. Secara formal, akurasi memiliki definisi sebagai berikut:

$$Akurasi = \frac{\text{Number of Correct Prediction}}{\text{Total Number of Prediction}} \quad (1)$$

Untuk klasifikasi biner, akurasi juga dapat dihitung dalam hal positif dan negatif sebagai berikut :

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \quad (2)$$

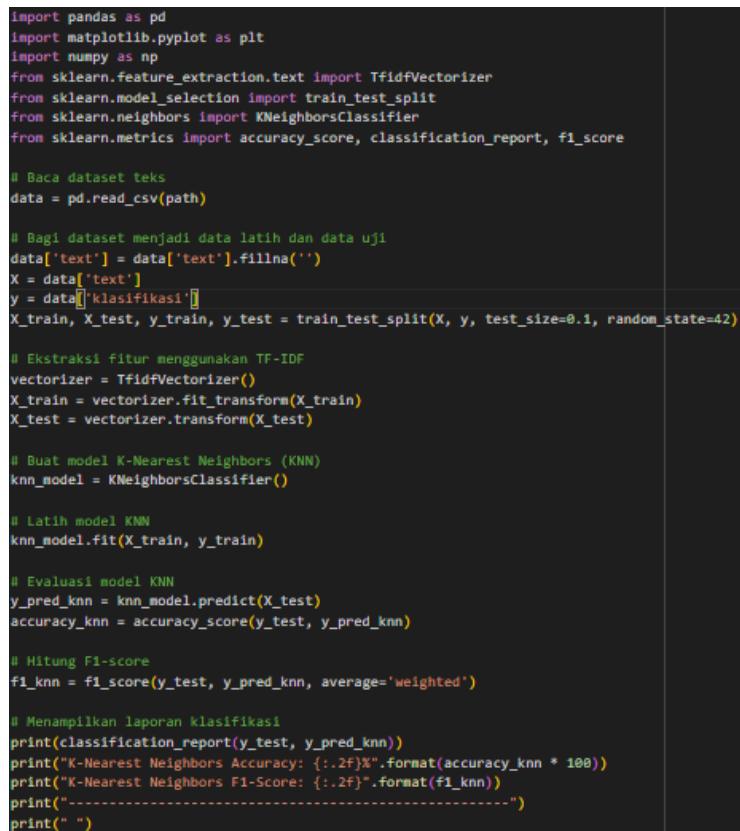
Dimana

- TP = True Positif
- TN = True Negatif
- FP = False Positif
- FN = False Negatif

3. HASIL PENELITIAN DAN PEMBAHASAN

3.1. Hasil Penelitian

Pada tahap ini, dataset yang digunakan sudah di import kedalam GoogleColeb dan akan dilakukan klasifikasi dengan menggunakan model algoritma *K-Nearest Neighbor*.



```

import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score, classification_report, f1_score

# Baca dataset teks
data = pd.read_csv(path)

# Bagi dataset menjadi data latih dan data uji
data['text'] = data['text'].fillna('')
X = data['text']
y = data['klasifikasi']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.1, random_state=42)

# Ekstraksi fitur menggunakan TF-IDF
vectorizer = TfidfVectorizer()
X_train = vectorizer.fit_transform(X_train)
X_test = vectorizer.transform(X_test)

# Buat model K-Nearest Neighbors (KNN)
knn_model = KNeighborsClassifier()

# Latih model KNN
knn_model.fit(X_train, y_train)

# Evaluasi model KNN
y_pred_knn = knn_model.predict(X_test)
accuracy_knn = accuracy_score(y_test, y_pred_knn)

# Hitung F1-score
f1_knn = f1_score(y_test, y_pred_knn, average='weighted')

# Menampilkan laporan klasifikasi
print(classification_report(y_test, y_pred_knn))
print("K-Nearest Neighbors Accuracy: {:.2f}%".format(accuracy_knn * 100))
print("K-Nearest Neighbors F1-Score: {:.2f}%".format(f1_knn))
print("-----")
print(" ")

```

Gambar 4 Source Code Program Klasifikasi Metode K-Nearest Neighbor

Pada gambar 4.23, menunjukkan impor pustaka-pustaka yang diperlukan untuk melakukan pemrosesan data, pembuatan model, dan evaluasi. Pustaka-pustaka yang diimpor antara lain: pandas untuk manipulasi data, matplotlib.pyplot untuk visualisasi, numpy untuk operasi matematika, *TfidfVectorizer*

dari `sklearn.feature_extraction.text` untuk ekstraksi fitur teks, `train_test_split` dari `sklearn.model_selection` untuk membagi data menjadi data latih dan data uji, `KNeighborsClassifier` dari `sklearn.neighbors` untuk membuat model *K-Nearest Neighbors* (KNN), serta `accuracy_score`, `classification_report`, dan `f1_score` dari `sklearn.metrics` untuk evaluasi model.

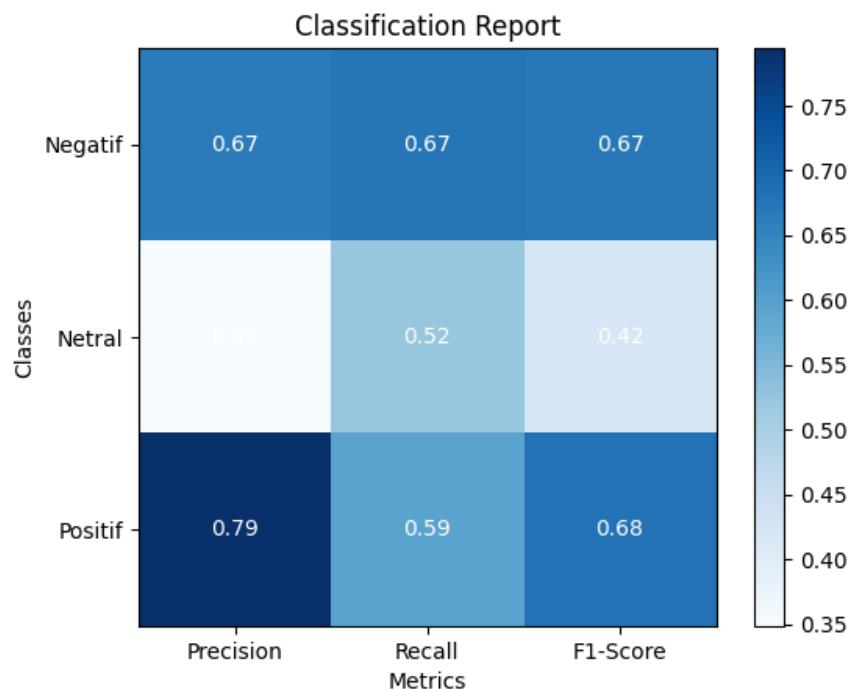
Data teks dibaca dari file CSV yang disimpan di path menggunakan pustaka *pandas*. Data yang dibaca terdiri dari kolom 'text' yang berisi teks yang akan dianalisis dan kolom 'klasifikasi' yang berisi label klasifikasi. Dataset dibagi menjadi data latih dan data uji menggunakan fungsi `train_test_split`. Data latih akan digunakan untuk melatih model, sedangkan data uji akan digunakan untuk menguji performa model.

Fitur-fitur teks diekstraksi menggunakan metode TF-IDF (*Term Frequency-Inverse Document Frequency*). Ini adalah teknik yang mengukur pentingnya suatu kata dalam suatu dokumen relatif terhadap seluruh korpus dokumen. Fitur-fitur ini akan digunakan sebagai input untuk model klasifikasi. Model *K-Nearest Neighbors* (KNN) dibuat menggunakan pustaka `KNeighborsClassifier` dari `sklearn.neighbors`. Model KNN dilatih menggunakan data latih (`X_train` dan `y_train`) dengan menggunakan metode `fit` pada objek `knn_model`. Model KNN dievaluasi menggunakan data uji (`X_test` dan `y_test`). Prediksi dari model (`y_pred_knn`) dibandingkan dengan label sebenarnya (`y_test`) untuk menghitung akurasi dan *F1-score*. *F1-score* dihitung menggunakan fungsi `f1_score` dari pustaka `sklearn.metrics`. *F1-score* adalah ukuran yang menggabungkan akurasi dan presisi untuk mengukur kinerja model pada kelas-kelas yang tidak seimbang.

Adapun hasil yang didapat setelah dilakukannya uji klasifikasi dataset menggunakan metode KNN adalah sebagai berikut:

- Hasil uji klasifikasi dengan perbandingan data 90:10

Penerapan metode KNN dengan menggunakan split validation dengan nilai akurasi dan nilai F1-Score dengan pembagian data training dan data testing sebesar 90:10 dapat dilihat *Classification Report* yang diperoleh pada gambar 4.24.



Gambar 5 Classification Report Metode KNN (90:10)

Tabel 1 Matrik Konfusi Metode KNN (90:10)

	Positif	Netral	Negatif
TRUE	1073	454	1069
FALSE	278	848	538

Berdasarkan gambar diatas, pada pembagian data sampling 90% dan data testing 10% menggunakan metode KNN maka didapatkan akurasi sebesar 60,94%. Dimana hasil ini didapatkan dari perhitungan seperti berikut:

Rumus: $Akurasi = \frac{TP+TN}{TP+TN+FP+FN}$

Dimana:

- TP (True Positive): Jumlah data dengan kelas yang benar diprediksi sebagai positif.
- TN (True Negative): Jumlah data dengan kelas yang benar diprediksi sebagai negatif.
- FP (False Positive): Jumlah data dengan kelas yang salah diprediksi sebagai positif.
- FN (False Negative): Jumlah data dengan kelas yang salah diprediksi sebagai negatif.

Dari laporan klasifikasi di atas, kita memiliki:

- TP (Positif) = 1073
- TN (Netral) = 454
- TN (Negatif) = 1069
- FP (Positif) = 278
- FNT (Netral) = 848
- FN (Negatif) = 538

Maka dapat dihitung akurasinya:

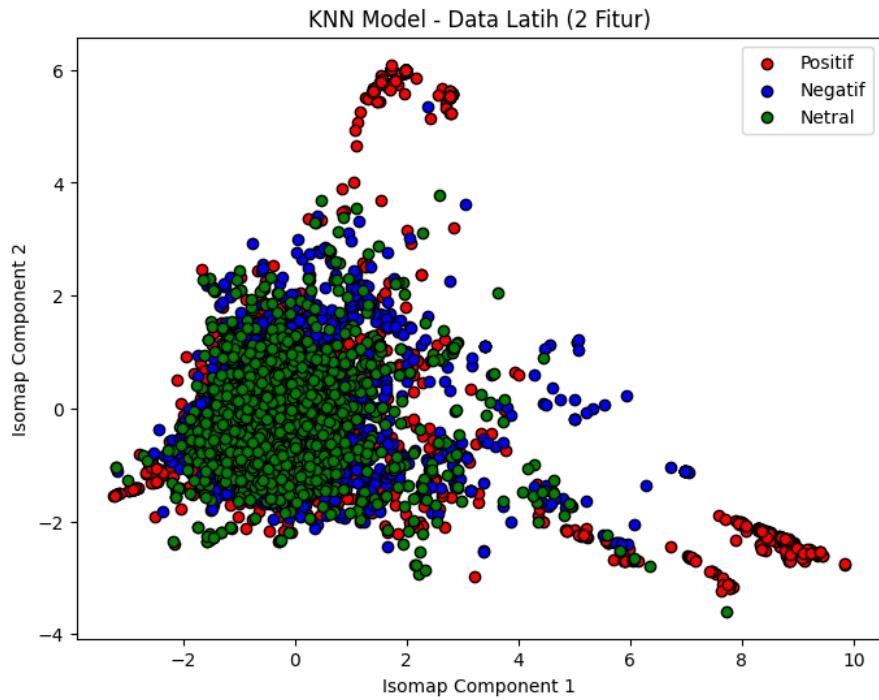
$$\begin{aligned} Akurasi &= \frac{1073 + 454 + 1069}{1073 + 454 + 1069 + 278 + 848 + 538} \\ &= \frac{2596}{4260} \\ &= 60,94\% \end{aligned}$$

Sedangkan untuk nilai F1-score yang diperoleh dari data tersebut sebesar 62%. Nilai F1-score ini, didapatkan berdasarkan kinerja model dengan memperhitungkan seberapa besar setiap kelas berkontribusi terhadap hasil keseluruhan. Dan berikut merupakan perhitungannya:

Rumus F1-score:

$$F1 - score = \frac{\frac{F1 - score Negatif \times Support Negatif + F1 - score Netral \times Support Netral + F1 - score Positif \times Support Positif}{Total Support}}{4260} = 62\%$$

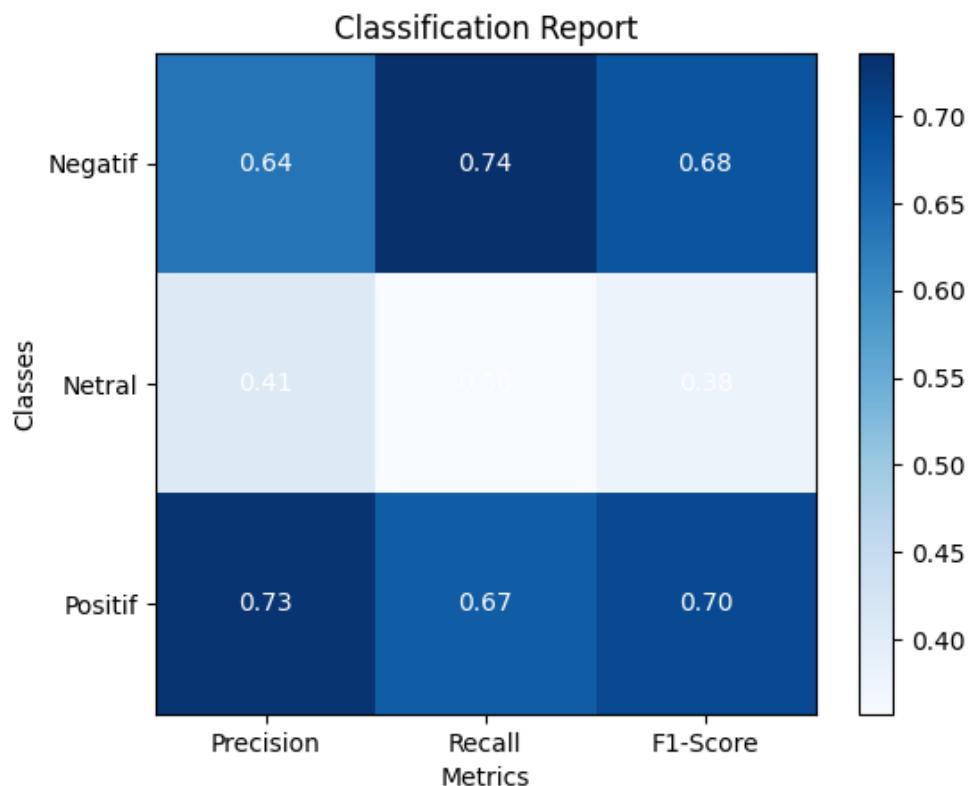
Adapun penyebaran klasifikasi data latih dari penerapan metode KNN pada pembagian data sampling 90% dan data testing 10% dapat dilihat pada gambar 4.25.



Gambar 6 Penyebaran Klasifikasi Data Latih Metode KNN (90:10)

- b. Hasil uji klasifikasi dengan perbandingan data 80:2

Penerapan metode KNN dengan menggunakan split validation dengan nilai akurasi dan nilai F1-Score dengan pembagian data training dan data testing sebesar 80:20 dapat dilihat *Classification Report* yang diperoleh pada gambar 4.26.



Gambar 7 Classification Report Metode KNN (80:20)

Tabel 2 Matrik Konfusi Metode KNN (80:20)

	Positif	Netral	Negatif
TRUE	2402	626	2341
FALSE	893	913	1344

Berdasarkan gambar diatas, pada pembagian data sampling 80% dan data testing 20% menggunakan metode KNN maka didapatkan akurasi sebesar 63,02%. Dimana hasil ini didapatkan dari perhitungan seperti berikut:

Rumus: $Akurasi = \frac{TP+TN}{TP+TN+FP+FN}$

Dimana:

- TP (True Positive): Jumlah data dengan kelas yang benar diprediksi sebagai positif.
- TN (True Negative): Jumlah data dengan kelas yang benar diprediksi sebagai negatif.
- FP (False Positive): Jumlah data dengan kelas yang salah diprediksi sebagai positif.
- FN (False Negative): Jumlah data dengan kelas yang salah diprediksi sebagai negatif.

Dari laporan klasifikasi di atas, kita memiliki:

- TP (Positif) = 2402
- TN (Netral) = 626
- TN (Negatif) = 2341
- FP (Positif) = 893
- FNT (Netral) = 913
- FN (Negatif) = 1344

Maka dapat dihitung akurasinya:

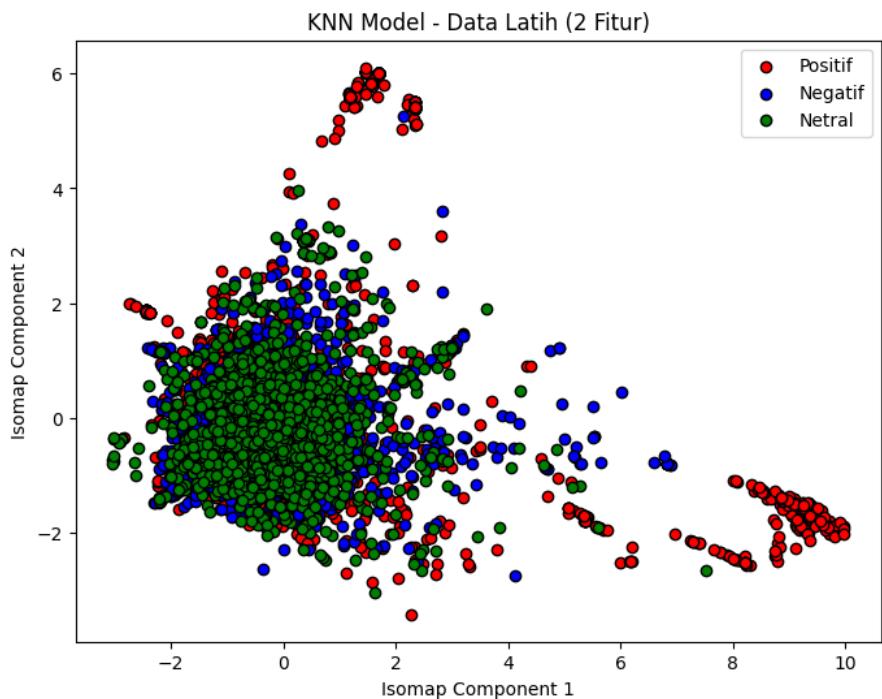
$$Akurasi = \frac{2402 + 626 + 2341}{2402 + 626 + 2341 + 893 + 913 + 1344} = \frac{5369}{8519} = 63,02\%$$

Sedangkan untuk nilai F1-score yang diperoleh dari data tersebut sebesar 63%. Nilai F1-score ini, didapatkan berdasarkan kinerja model dengan memperhitungkan seberapa besar setiap kelas berkontribusi terhadap hasil keseluruhan. Dan berikut merupakan perhitungannya:

Rumus F1-score:

$$F1 - score = \frac{\frac{F1 - score Negatif \times Support Negatif + F1 - score Netral \times Support Netral + F1 - score Positif \times Support Positif}{Total Support}}{0.68 \times 3182 + 0.38 \times 1752 + 0.70 \times 3585} = 63\%$$

Adapun penyebaran klasifikasi data latih dari penerapan metode KNN pada pembagian data sampling 80% dan data testing 20% dapat dilihat pada gambar 4.27.



Gambar 7 Penyebaran Klasifikasi Data Latih Metode KNN (80:20)

3.2. Pembahasan Hasil Penelitian

Pembahasan hasil penelitian ini menguraikan hasil analisis yang diperoleh dari pengujian terhadap data uji. Penelitian dilakukan dengan mengklasifikasikan data teks dari tweet di Twitter menggunakan algoritma tertentu, yaitu K-Nearest Neighbor (KNN). Algoritma tersebut digunakan untuk melakukan pengujian klasifikasi data, kemudian hasilnya dibandingkan berdasarkan akurasi yang diperoleh. Dataset yang digunakan menerapkan metode split validation dengan pembagian data latih (training) dan data uji (testing) sebesar 90:10 dan 80:20. Perbandingan akurasi yang diperoleh dari kedua skenario pembagian data ini dianalisis untuk menilai performa algoritma. Penjelasan rinci mengenai hasil pengujian dapat dilihat pada pemaparan berikut.

Tabel 3 Hasil Metode

No	Metode	Akurasi (90:10)	Akurasi (80:20)	F-measure (90:10)	F-measure (80:20)
1	K-Nearest Neighbor	60,94 %	63,02 %	0,62	0,63

4. KESIMPULAN

Dalam penelitian ini, metode klasifikasi K-Nearest Neighbor telah dievaluasi untuk mengukur kinerja yang diperoleh dalam memprediksi dan mengklasifikasikan data dalam konteks sistem informasi. Evaluasi yang dilakukan menggunakan dua skenario pembagian dataset, yaitu rasio 90:10 dan 80:20 untuk data latih dan data uji. Maka dapat disimpulkan seperti dibawah ini:

1. Akurasi pada skenario pembagian data (90:10 dan 80:20)
 - a) Pada pembagian data 90:10 (90% data latih, 10% data uji), akurasi yang dicapai adalah 60,94%.
 - b) Pada pembagian data 80:20 (80% data latih, 20% data uji), akurasi meningkat menjadi 63,02%.

Hal ini menunjukkan bahwa pembagian data dengan lebih banyak data latih (90:10) tidak selalu memberikan hasil akurasi yang lebih tinggi dibandingkan pembagian 80:20, yang mungkin disebabkan oleh distribusi data atau jumlah data uji yang memengaruhi performa model.

2. F-measure pada skenario pembagian data (90:10 dan 80:20)
 - a) Pada pembagian data 90:10, nilai F-measure tercatat sebesar 0,62.
 - b) Pada pembagian data 80:20, nilai F-measure meningkat menjadi 0,63.

F-measure, yang merupakan metrik keseimbangan antara presisi dan recall, menunjukkan performa yang konsisten dengan akurasi. Peningkatan kecil ini menandakan bahwa pembagian 80:20 memberikan model KNN performa yang lebih baik dalam mendekripsi sentimen secara keseluruhan.

Dalam konteks penggunaan metode klasifikasi dengan metode K-Nearest Neighbor menunjukkan hasil akurasi dan F-measure yang cukup baik untuk klasifikasi sentiment analysis, dengan nilai rata-rata akurasi sekitar 62% dan F-measure sekitar 0,63. Namun, performa ini juga menunjukkan bahwa metode KNN memiliki keterbatasan dalam menangani data sentimen, yang dapat ditingkatkan dengan pemrosesan data yang lebih baik, optimasi parameter, atau penggunaan metode klasifikasi lainnya.

REFERENCES

- [1] APJII, *Profil Internet Indonesia 2022*. Jakarta: Pustaka Pelajar Asosiasi Penyelenggara Jasa Internet Indonesia, 2022. [Online]. Available: <https://apjii.or.id/survei2022x>
- [2] Ferdinandus Setu, “Warganet Paling Banyak Laporkan Akun Twitter SIARAN PERS NO. 08/HM/KOMINFO/01/2019,” *Kementerian Komunikasi dan Informatika Republik Indonesia*, 2019. [Online]. Available: https://www.kominfo.go.id/index.php/content/detail/15852/siaran-pers-no-08hmkominfo012019-tentang-warganet-paling-banyak-laporkan-akun-twitter/0/siaran_pers
- [3] *Buku saku penanganan ujaran kebencian (hate speech)*. Komisi Nasional Hak Asasi Manusia Republik Indonesia, 2015. [Online]. Available: <https://books.google.co.id/books?id=F6U0xQEACAAJ>
- [4] D. Elisabeth, I. Budi, and M. O. Ibrohim, “Hate Code Detection in Indonesian Tweets using Machine Learning Approach: A Dataset and Preliminary Study,” in *2020 8th International Conference on Information and Communication Technology (ICoICT)*, Yogyakarta, Indonesia: IEEE, Jun. 2020, pp. 1–6. doi: 10.1109/ICoICT49345.2020.9166251.
- [5] F. F. Mailoa, “Analisis sentimen data twitter menggunakan metode text mining tentang masalah obesitas di indonesia,” *JISPH*, vol. 6, no. 1, p. 44, Oct. 2021, doi: 10.22146/jisph.44455.
- [6] Department of Computer Science & Engineering, Heritage Institute of Technology, Kolkata, India, L. Dey, S. Chakraborty, A. Biswas, B. Bose, and S. Tiwari, “Sentiment Analysis of Review Datasets Using Naïve Bayes’ and K-NN Classifier,” *IJIEEB*, vol. 8, no. 4, pp. 54–62, Jul. 2016, doi: 10.5815/ijieeb.2016.04.07.
- [7] L. Farokhah, “Implementasi K-Nearest Neighbor untuk Klasifikasi Bunga Dengan Ekstraksi Fitur Warna RGB,” *JTIK*, vol. 7, no. 6, p. 1129, Dec. 2020, doi: 10.25126/jtiik.2020722608.
- [8] S. Suryono *et al.*, “Analisis Sentiment Pada Twitter Dengan Menggunakan,” pp. 9–15, 2018.
- [9] J. Apostolakis, “An Introduction to Data Mining,” in *Data Mining in Crystallography*, D. W. M. Hofmann and L. N. Kuleshova, Eds., in *Structure and Bonding*, vol. 134. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 1–35. doi: 10.1007/978-3-642-04759-6_1.
- [10] M. A. Al-Shabi, “Evaluating the performance of the most important Lexicons used to Sentiment analysis and opinions Mining,” 2020.
- [11] A. P. Wibawa, M. G. A. Purnama, M. F. Akbar, and F. A. Dwiyanto, “Metode-metode Klasifikasi,” vol. 3, no. 1, 2018.