

PERBANDINGAN PERFORMA ALGORITMA MACHINE LEARNING UNTUK ANALISIS SENTIMEN PENGGUNA X MENGENAI TOPIK DEEPSEEK AI

Alif Suryo Nugroho¹, Widi Widayat²

¹Jalan Jend Ahmad Yani, Pabelan, Kartasura, Kab. Sukoharjo, Jawa Tengah, Indonesia

²Jalan Jend Ahmad Yani, Pabelan, Kartasura, Kab. Sukoharjo, Jawa Tengah, Indonesia

 Email korespondensi: l200200004@student.ums.ac.id

Abstrak. Penelitian ini menganalisis sentimen pengguna X (Twitter) terhadap peluncuran DeepSeek AI menggunakan lima algoritma klasifikasi: *Naïve Bayes*, *Random Forest*, *Logistic Regression*, *KNN*, dan *SVM*. Data dikumpulkan menggunakan Tweet Harvest dan dilabeli secara otomatis dengan TextBlob serta sebagian secara manual. Untuk memvalidasi konsistensi labeling manual, digunakan metode *Cohen's Kappa*. Teknik *SMOTE* diterapkan untuk menangani ketidakseimbangan kelas. Setiap algoritma diuji dalam lima versi tuning berbeda. Hasil evaluasi menunjukkan bahwa SVM versi V4 (*tunning agresif*) memberikan performa terbaik dengan *F1-Score* 85,86%, *akurasi* 85,98%, *precision* 86,06%, dan *recall* 85,98%. Temuan ini menunjukkan bahwa SVM sangat efektif dalam analisis sentimen pada data sosial media yang kompleks.

Kata kunci: Analisis Sentimen, Kecerdasan Buatan, Pembelajaran Mesin, Twitter

PENDAHULUAN

Perkembangan teknologi komunikasi dan informasi di era digital telah mengubah dinamika interaksi sosial. Media sosial, khususnya platform X (sebelumnya Twitter), menjadi wadah utama bagi pengguna untuk menyampaikan pendapat, opini, dan sentimen terhadap berbagai isu (Setiyawati & Cahyono, 2023). Dengan 611,3 juta pengguna global pada April 2024 (Stevany, 2024), platform X menghasilkan volume data yang besar, menjadikannya sumber kaya untuk analisis sentimen, terutama terkait teknologi kecerdasan buatan (AI) seperti DeepSeek AI. Teknologi AI kini telah



terintegrasi dalam kehidupan sehari-hari, dengan contoh seperti Claude AI, ChatGPT, dan Gemini AI. DeepSeek AI, sebagai model bahasa besar (LLM) terbaru, menonjol dengan kemampuan analisis kode, arsitektur multimodal, dan aplikasi seperti asisten penelitian ilmiah serta pengembangan perangkat lunak, yang hampir setara dengan model proprietary seperti OpenAI-o1, namun dengan biaya pelatihan lebih rendah dan bobot model yang dibagikan secara terbuka (Abraham, 2025). Sentimen pengguna X terhadap DeepSeek AI bervariasi, mulai dari pujian atas kecanggihan teknologi, pandangan netral yang membandingkannya dengan AI lain, hingga kritik terkait kemampuan menangkap maksud pengguna, terutama pada bahasa informal atau ambigu.

Analisis sentimen menjadi alat krusial untuk memahami opini publik di platform X, yang sering kali mengandung kebisingan data seperti bahasa informal, sarkasme, atau konteks budaya yang kompleks (Sjoradia et al., 2024). Penelitian terdahulu, seperti Darwis et al. (2021), menerapkan algoritma Naïve Bayes untuk mengklasifikasikan sentimen tweet dengan akurasi 68,97%, menunjukkan keterbatasan dalam menangani data yang kompleks. Penelitian lain oleh Saepudin et al. (2024) membandingkan Support Vector Machine (SVM) (91%), Random Forest (94%), dan Logistic Regression (86%), sementara Pratmanto et al. (2023) menunjukkan K-Nearest Neighbors (KNN) mencapai akurasi 82,6% untuk sentimen positif dan 83,1% untuk negatif. Herdiyani dan Zailani (2022) menggunakan Random Forest untuk menganalisis sentimen perpindahan ibu kota, dengan distribusi sentimen positif (55%), netral (32%), dan negatif (14%). Kurniawati et al. (2023) menunjukkan SVM unggul dengan akurasi 90,20% dibandingkan Decision Tree (89,80%) dalam analisis ulasan aplikasi transportasi online. Celah penelitian teridentifikasi pada belum adanya studi yang membandingkan performa Naïve Bayes, KNN, Logistic Regression, Random Forest, dan SVM pada data sentimen pengguna X terkait DeepSeek AI. Setiap algoritma memiliki kelebihan, seperti kemampuan SVM dalam data berdimensi tinggi atau ketahanan Random Forest terhadap overfitting, namun efektivitasnya pada konteks DeepSeek AI belum teruji. Penelitian ini relevan mengingat meningkatnya adopsi AI dan dampaknya pada opini publik di platform X. Urgensi penelitian terletak pada kebutuhan memahami persepsi masyarakat terhadap DeepSeek AI untuk mendukung pengembangan teknologi, strategi pemasaran, dan kebijakan terkait AI.

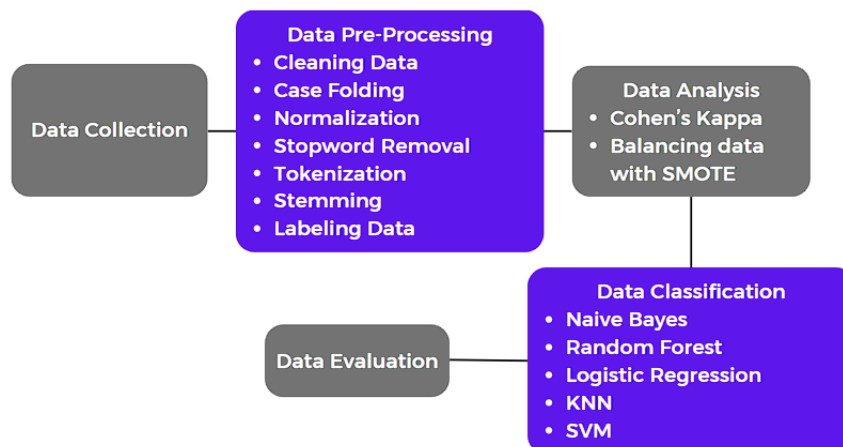
Tujuan Penelitian: Penelitian ini bertujuan untuk menganalisis sentimen pengguna X terhadap DeepSeek AI menggunakan algoritma klasifikasi Naïve Bayes, KNN, Logistic Regression, Random Forest, dan SVM, serta menentukan algoritma dengan akurasi tertinggi berdasarkan metrik Accuracy, Precision, Recall, dan F1-Score.



Manfaat Penelitian: Secara teoritis, penelitian ini memperkaya literatur analisis sentimen dengan konteks AI terbaru. Secara praktis, hasilnya memberikan wawasan bagi pengembang AI, pemasar, dan pemangku kepentingan untuk memahami persepsi publik, meningkatkan strategi komunikasi, dan mengoptimalkan pengembangan teknologi AI.

METODE

Pada studi kali ini, *Tweet Harvest* digunakan sebagai *tools* untuk proses *Crawling Data* pada platform X, yang nantinya akan diolah pada tahap selanjutnya. 5 model klasifikasi yang nantinya akan digunakan adalah *Naïve Bayes*, *Random Forest*, *Logistic Regression*, *KNN*, dan *SVM*. Langkah-langkah pada proses *pre-processing* adalah *Cleaning data*, *Case Folding*, *Normalization*, *Stopword Removal*, *Tokenization*, *Stemming*, dan *Labeling*. Lalu akan dilanjutkan dengan data analisis menggunakan *SMOTE* untuk melakukan *balancing data* dan *Cohen's Kappa* untuk mengukur tingkat kesepakatan antara prediksi model dengan label aktual. Hasil klasifikasi kemudian dievaluasi menggunakan beberapa metrik, yaitu *Accuracy*, *Precision*, *Recall*, *F1-Score*. Penjelasan setiap langkah yang akan dilakukan akan dituliskan di bawah.

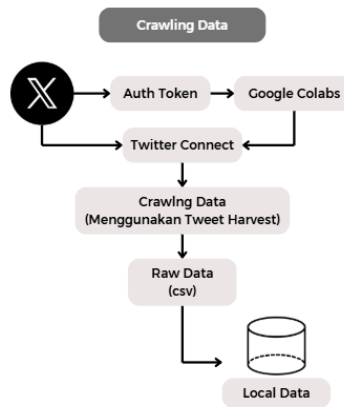


Gambar 1. Tahapan Metodologi Penelitian

Gambar 1 (Wang et al., 2023) merepresentasikan tahapan dalam melakukan penelitian yang digunakan untuk menyelidiki data yang diungkapkan oleh publik di platform media sosial Twitter mengenai *DeepSeek AI* dalam tweet berbahasa Inggris.



1. Pengumpulan Data



Gambar 2. Tahapan *Crawling Data*

Proses pengumpulan data menggunakan metode *crawling data* dengan aplikasi *google colabs* dan *tools tweet harvest*, akan menghasilkan data yang kemudian di eksekusi oleh kode *python* yang diarahkan untuk mendapatkan data *tweet* berupa *tweet* Bahasa Inggris, yang memiliki *keyword* “DeepSeek AI”. Dimulai dari tanggal 2 Desember 2023, sampai dengan 15 Maret 2025. Tahapan *crawling data* ditunjukkan oleh gambar 2 diatas.

2. Data Pre-Processing

Data yang telah didapatkan pada proses *crawling data* kemudian akan di olah pada proses ini. *Pre-processing* merupakan tahap yang digunakan untuk mempersiapkan data *review* sehingga akan menghasilkan data yang berkualitas dan nantinya akan siap digunakan (Putra & Ummah, 2024).

- a) *Cleaning Data* adalah proses untuk membersihkan data *text* dari kata yang tidak diperlukan untuk mengurangi noise (Khoiruddin et al., 2024). Kata yang dihilangkan antara lain adalah karakter HTML, *emoticon*, hastag (#), URL, dan lain sebagainya.
- b) *Case folding* merupakan proses yang dilakukan untuk menyeragamkan karakter pada data (Adityarini et al., 2021). Pada kasus ini, seluruh *text* akan di rubah kedalam huruf non-kapital (*lower case*).
- c) *Normalization* dilakukan untuk mengatasi munculnya istilah-istilah yang tidak standar dengan melakukan proses konversi terhadap kata tidak baku menjadi kata baku (Imaduddin et al., 2023)



- d) *Stopword Removal* digunakan untuk menghilangkan kata yang nilai informasinya rendah. Contohnya kata “to”, “in”, “and”, “with” (Sheviraa et al., 2022).
- e) *Tokenization* sangat penting pada saat melakukan analisis data, dikarenakan langkah ini dapat membantu model dalam memahami dan menganalisis struktur data teks (Alomari & Ahmad, 2024).
- f) *Stemming* adalah proses transformasi kata, untuk merubah kata yang sudah ada menjadi kata dasarnya, dan hanya bagian bagian yang penting saja yang tersisa (Munggaran et al., 2023).
- g) *Labeling Data*, pada tahap ini pelabelan dilakukan dengan menggunakan *library TextBlob*, dengan memberikan klasifikasi atau keterangan positif, netral, atau negative, sehingga setiap data akan memiliki bobot tersendiri, diharapkan dengan adanya klasifikasi data yang baik, nantinya akan diperoleh model klasifikasi yang baik pula (Chamid et al., 2024).

3. Analisis Data

- a) *Cohen’s Kappa*, digunakan untuk mengukur tingkat kesepakatan antar pelabel pada data kategori. Pada penelitian ini digunakan untuk mengevaluasi konsistensi hasil labeling manual terhadap sebagian data (Landis & Koch, 1977).
- b) *Balancing* data bertujuan untuk menyamakan distribusi antar kelas agar model tidak bias. Salah satu teknik yang digunakan adalah *SMOTE*, yang bekerja dengan membuat data sintesis dari kelas minoritas berdasarkan tetangga terdekatnya. Teknik ini terbukti mampu meningkatkan performa model, terutama dalam klasifikasi data tidak seimbang (Chawla et al., 2002).

4. Klasifikasi Data

Pada tahap ini, digunakan beberapa algoritma klasifikasi untuk melakukan pemodelan teks dengan tujuan mengklasifikasikan data sesuai dengan kategori yang telah ditentukan. Klasifikasi merupakan salah satu teknik dalam pembelajaran mesin (*machine learning*) yang digunakan untuk memprediksi kelas atau kategori dari suatu data berdasarkan pola yang ditemukan pada data pelatihan. Dalam penelitian ini, digunakan beberapa algoritma klasifikasi, antara



lain *Naïve Bayes*, *Random Forest*, *Logistic Regression*, *K-Nearest Neighbors (KNN)*, dan *Support Vector Machine (SVM)*.

a) Preprocessing Data dengan TF-IDF Vectorization

Pada tahap ini data teks akan di proses agar mampu digunakan oleh model klasifikasi menggunakan metode *TF-IDF (Term Frequency-Inverse Document Frequency) Vectorization*. TF biasanya dibagi dengan panjang kalimat. DF adalah penghitungan jumlah kalimat dimana suatu kata muncul, dengan demikian semakin sedikit frekuensi kalimatnya semakin kecil pula bobot kalimat tersebut (Yudhanata & Sudarmilah, n.d.). Maka perhitungan TF-IDF dapat menggunakan persamaan (1).

$$(1) \quad tf - idf_{t,d} = tf_{t,d} * idf_t$$

b) *Data Splitting*

Setelah vektorisasi dilakukan, data kemudian dibagi menjadi dua bagian: data latih (training data) dan data uji (test data). Proses ini dilakukan menggunakan fungsi "*train_test_split*" dari pustaka *sklearn.model_selection*. Dalam penelitian (Joseph, 2022) pembagian dengan komposisi 80% data untuk pelatihan dan 20% untuk pengujian lebih disarankan bila jumlah data yang dimiliki banyak.

c) Klasifikasi *Naïve Bayes*

Klasifikasi *Naïve Bayes* adalah klasifikasi probabilitas sederhana yang akan menghitung himpunan probabilitas dengan mengukur frekuensi dan kombinasi nilai dari kumpulan data yang diberikan (Pradana & Sugiharti, 2023). Model *Multinomial Naïve Bayes* menghitung probabilitas kata-kata dalam dokumen tertentu terkait dengan kategori sentimen tertentu. Rumus untuk menghitung probabilitas suatu kata t_k pada kategori sentimen p adalah (2)

$$(2) \quad P(t_k|p) = \frac{n(t_k, p)}{n(p)}$$

Dimana:



- $P(t_k | p)$ adalah probabilitas kemunculan kata t_k dalam dokumen dengan sentimen p .
- $n(t_k, p)$ adalah jumlah kemunculan kata t_k dalam dokumen dengan sentimen p .
- $n(p)$ adalah jumlah kata total dalam dokumen dengan sentimen p .

d) Klasifikasi *Random Forest*

Random Forest adalah algoritma yang menggunakan metode pemisahan biner rekursif untuk mencapai node akhir dalam struktur pohon berdasarkan pohon klasifikasi dan regresi (Pamuji & Ramadhan, 2021). *Random Forest* menggunakan banyak pohon keputusan T_1, T_2, \dots, T_n yang masing-masing mengklasifikasikan data dan memberikan voting untuk menentukan kategori akhir. Prediksi final adalah mayoritas voting dari semua pohon. (3)

$$(3) \quad \hat{y} = \text{majority_vote}(T_1(x), T_2(x), \dots, T_n(x))$$

Dimana :

- \hat{y} adalah prediksi akhir (sentimen).
- $T_1(x), T_2(x), \dots, T_n(x)$ adalah hasil klasifikasi dari setiap pohon keputusan terhadap data uji x .

e) Klasifikasi *Logistic Regression*

Logistic Regression adalah model klasifikasi yang memprediksi probabilitas bahwa suatu data termasuk dalam salah satu dari dua kelas (Madžar, 2022). Meskipun namanya mengandung kata "regresi", algoritma ini sebenarnya digunakan untuk klasifikasi, bukan prediksi nilai kontinu. Untuk menganalisis sentimen dengan *Logistic Regression*, kita menggunakan rumus berikut (4).

$$(4) \quad P(y = 1|X) = \frac{1}{1 + e^{-(\theta_0 + \theta_1 X_1 + \theta_2 X_2 + \dots + \theta_n X_n)}}$$



Dimana :

- X_1, X_2, \dots, X_n adalah fitur-fitur yang digunakan untuk prediksi (misalnya, kata-kata yang ada dalam tweet).
- $\theta_0, \theta_1, \dots, \theta_n$ adalah bobot model yang dipelajari selama pelatihan.
- $P(\hat{y} = 1|X)$ adalah probabilitas bahwa tweet tersebut memiliki sentimen positif.

f) Klasifikasi KNN (*K-Nearest Neighbors*)

KNN adalah algoritma non-parametrik yang mengklasifikasikan data berdasarkan k tetangga terdekat. Dalam manajemen data, KNN tidak sensitif terhadap outlier dan tidak memerlukan pelatihan ulang, cukup dengan memperbarui sampel tetangga saat data diperbarui (Tan et al., 2024). Proses KNN dapat dijelaskan dengan rumus jarak Euclidean yang sederhana (5)

$$(5) \quad d(x, y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_n - y_n)^2}$$

Dimana :

- $d(x, y)$ adalah jarak antara data baru x dan data pelatihan y .
- x_1, x_2, \dots, x_n adalah fitur dari data baru.
- y_1, y_2, \dots, y_n adalah fitur dari data pelatihan.

g) Klasifikasi *Support Vector Machine*

Support Vector Machine (SVM) merupakan satu metode machine learning yang bekerja atas prinsip *Structural Risk Minimization* (SRM) dengan tujuan menemukan *hyperplane* terbaik yang memisahkan dua buah class pada *input space* (Praghakusma & Charibaldi, 2021). Dengan menggunakan kernel *linier* pada penelitian ini, menunjukkan bahwa, kernel *linier* mengacu pada **aturan atau kondisi** yang harus dipenuhi agar model SVM dapat menemukan *hyperplane* terbaik yang memisahkan dua kelas. Rumus perhitungan dengan kernel *linier* adalah sebagai berikut (6) (7)



$$(6) \quad \text{maximize } \frac{1}{2} \|\mathbf{w}\|^2$$

Dengan Batasan :

$$(7) \quad y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 \quad \text{untuk setiap } i$$

Dimana :

- \mathbf{w} adalah vektor bobot yang menentukan posisi hyperplane.
- \mathbf{b} adalah bias yang memungkinkan hyperplane tidak selalu melewati titik asal (0,0).
- y_i adalah label kelas (misalnya, +1 atau -1 untuk dua kelas).
- x_i adalah data titik.
- Margin $\frac{1}{2}\|\mathbf{w}\|^2$ harus dimaksimalkan agar ada jarak terbesar antara dua kelas.

5. Evaluasi

Evaluasi model klasifikasi bertujuan untuk menilai kinerja model dalam mengklasifikasikan data ke dalam kategori yang benar. Salah satu alat utama untuk evaluasi ini adalah *confusion matrix*, yang memberikan gambaran detail mengenai hasil prediksi model, selain itu untuk memberikan visualisasi, pada penelitian ini, saya juga menambahkan *bar chart* untuk memberikan visualisasinya. Hasil evaluasi meliputi recall, accuracy, dan precision.

a) *Confusion Matix*

Confusion matrix adalah tabel yang digunakan untuk mengevaluasi kinerja model klasifikasi dengan menunjukkan jumlah prediksi yang benar dan salah untuk setiap kelas (Fatah, n.d.). Tabel ini terdiri dari empat komponen utama



Tabel 1. *Confusion Matrix*

Actual Class	Predicted Class	
	Positive	Negative
Positive	True Positive (TP)	False Negative (FN)
Negative	False Positive (FP)	True Negative (TN)

Berdasarkan tabel 1 diperoleh empat nilai keluaran :

- *Accuracy* adalah mengukur jumlah prediksi yang benar (positif maupun negatif) keseluruhan prediksi yang dibuat.

$$(8) \quad \frac{TP+TN}{TP+TN+FP+FN}$$

- *Recall* adalah mengukur jumlah dari sampel yang benar-benar terdeteksi (positif maupun negatif)

$$(9) \quad \frac{TP}{TP+FN}$$

- *Precision* adalah mengukur jumlah dari prediksi yang benar-benar terdeteksi (positif maupun negatif)

$$(10) \quad \frac{TP}{TP+FP}$$

- *F1-Score* adalah mengukur kinerja model klasifikasi

$$(11) \quad 2 * \frac{Precision * Recall}{Precision + Recall}$$

b) *Bar Chart*

Visualisasi *bar chart* ini digunakan untuk menggambarkan jumlah prediksi model pada masing-masing kategori sentimen, seperti Positif, Negatif, dan Netral. Grafik ini memberikan gambaran yang jelas tentang bagaimana model mengklasifikasikan data sentimen, memudahkan untuk melihat apakah model cenderung memprediksi salah satu kategori lebih banyak daripada yang lain. Setiap kategori sentimen diwakili oleh sebuah batang, dan tinggi batang menggambarkan jumlah data yang diklasifikasikan ke dalam kategori tersebut.



HASIL

3.1 Pengumpulan Data

Proses pengumpulan data dilakukan dengan Teknik *Crawling Data*, menggunakan platform *Google Colabs* dan *Tools Tweet Harvest*, dan nantinya akan dieksekusi oleh kode python terhadap *tweet* yang berbahasa inggris, serta menggunakan *keyword* "DeepSeek AI". Proses pengambilan data dilakukan mulai dari tanggal 29 Juli 2024, sampai dengan 11 Maret 2025, dan menghasilkan data sejumlah data. Hasil *crawling data* akan diperlihatkan pada tabel 2.

Tabel 2. Hasil *Crawling Data*

Kata Kunci	Jumlah Data Total	Jumlah Data Duplikat	Jumlah Data Tanpa Duplikat
Deepseek AI	3262	72	3190

3.2 Preprocessing

Preprocessing akan dilakukan dengan menggunakan jumlah dataset tanpa duplikasi, sebanyak 3190 dataset. Pada preprocessing ini, dilakukan dengan menggunakan bahasa pemrograman *python* dan di implementasikan pada aplikasi *google colab*. Tujuan dilakukan tahap preprocessing adalah memudahkan dalam pengolahan data, dan membuat hasil data memiliki akurasi yang baik.

1. *Cleaning Data*

Dalam tahap ini, akan dilakukan pembersihan kata kata seperti penyebutan nama pengguna, link penghubung, dan lain sebagainya. Hasilnya akan ditunjukkan pada tabel 3.

Tabel 3. Hasil *Cleaning Data*

Sebelum	Sesudah
@zephyr_z9 Nahh deepseek is one of a kind you will see that when they release something next time. Its gonna be deepseek moment all over again.	Nahh deepseek is one of a kind you will see that when they release something next time. Its gonna be deepseek moment all over again.



2. *Case Folding*

Dalam tahap ini, akan dilakukan pengkonversian huruf kapital, menjadi huruf kecil. Hasilnya akan ditunjukkan pada tabel 4.

Tabel 4. Hasil *Case Folding*

Sebelum	Sesudah
Nahh deepseek is one of a kind you will see that when they release something next time. Its gonna be deepseek moment all over again.	nahh deepseek is one of a kind you will see that when they release something next time. its gonna be deepseek moment all over again.

3. *Normalization*

Dalam tahap ini, akan dilakukan pengkonversian atau perubahan istilah tidak baku menjadi istilah baku. Seperti "Nahh" menjadi "No", lalu "Gonna" menjadi "Going To" Hasilnya akan ditunjukkan pada tabel 5.

Tabel 5. Hasil *Normalization*

Sebelum	Sesudah
nahh deepseek is one of a kind you will see that when they release something next time. its gonna be deepseek moment all over again.	no deepseek is one of a kind you will see that when they release something next time. its going to be deepseek moment all over again.

4. *Stopword Removal*

Dalam tahap ini, akan dilakukan penghapusan kata kata yang tidak memiliki arti, seperti "is", "of", "a", "you", "that", "when", "they", dll. Hasilnya akan ditunjukkan pada tabel 6.

Tabel 6. Hasil *Stopword Removal*

Sebelum	Sesudah
nahh deepseek is one of a kind you will see that when they release something next time. its gonna be deepseek moment all over again.	no deepseek kind see release something next time. its going to deepseek moment over again.

5. *Tokenization*

Dalam tahap ini, akan dilakukan pembagian kalimat menjadi kata atau unit kecil seperti token. Hasilnya akan ditunjukkan pada tabel 7.



Tabel 7. Hasil *Tokenization*

Sebelum	Sesudah
no deepseek kind see release something next time. its going to deepseek moment over again.	['no', 'deepseek', 'kind', 'see', 'release', 'something', 'next', 'time', 'its', 'going', 'to', 'deepseek', 'moment', 'over', 'again']

6. *Stemming*

Dalam tahap ini, akan dilakukan proses yang mengubah kata menjadi bentuk kata dasar. Hasilnya akan ditunjukkan pada tabel 8.

Tabel 8. Hasil *Tokenization*

Sebelum	Sesudah
deepseek kind see release something next time. going to deepseek moment all over again.	no deepseek kind see releas someth next time. its go to deepseek moment over again.

3.3 *Labeling*

Selanjutnya adalah proses *labeling*. Tujuannya adalah untuk menganalisis data Twitter untuk menentukan nada emosional dari setiap tweet dengan sesedikit mungkin *noise*. TextBlob, yang merupakan pustaka Python akan digunakan untuk memproses data tekstual, untuk mengklasifikasikan setiap tweet sebagai positif, negatif, atau netral. keterangan positif, negatif, dan netral (Islam L et al., 2023). Hasil analisis bisa dilihat pad tabel 9.

Tabel 9. Hasil *Labeling*

Klasifikasi	Jumlah Data
Positif	= 1569
Netral	= 758
Negatif	= 863
Total	= 3190

Tabel 10. Hasil Contoh *Labeling*

Tweet Contoh	Hasil Klasifikasi
deepseek greater than find me 5 potential bugs	Positif
breaking deepseek v2 will be released march 17th	Netral
it turns out chatgpt o1 and deepseekr1 cheat at chess if thello re losing which makes me	Negatif



wonder if i should i should trust artificial intelligence with anything

3.4 Cohen's Kappa Method

Selanjutnya proses validasi data dilakukan untuk mengidentifikasi sentimen emosional, 30 dari 1053 data teks yang dikumpulkan, dengan tujuan untuk menghasilkan klasifikasi sentimen yang akurat. Setiap teks dianalisis secara independen oleh dua panelis untuk menetapkan label Positif, Negatif, atau Netral berdasarkan nada dan konteks teks. Penilaian dilakukan dengan mempertimbangkan fokus teks terhadap subjek utama, yaitu DeepSeek AI. Untuk memastikan konsistensi, proses labeling divalidasi menggunakan metrik *Cohen's Kappa*, yang mengukur tingkat kesepakatan setelah memperhitungkan kemungkinan kesepakatan secara kebetulan. Hasil labeling dan analisis kesepakatan antar-rater dapat dilihat pada tabel 11.

Tabel 11. Contoh Hasil Validasi

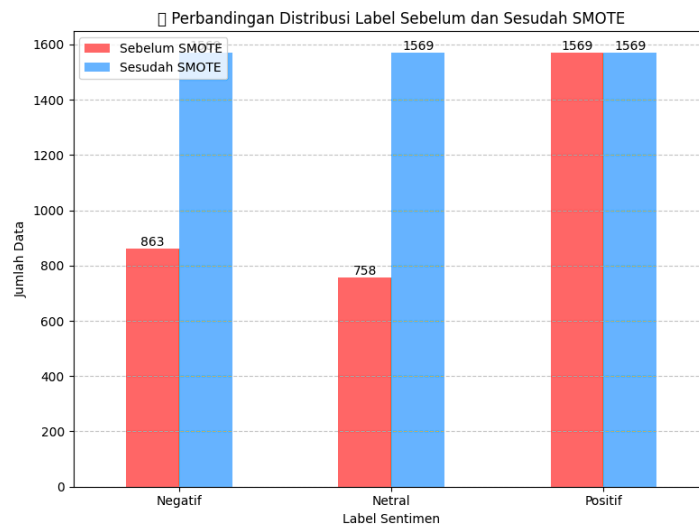
Teks	Panelis 1	Panelis 2	Hasil
european willingness to experiment with adds to pressure on artificial intelligence leaders to compete with the chinese upstart	Negatif	Netral	Tidak Setuju
a deepseek moment	Netral	Positif	Tidak Setuju
nahh deepseek is one of a kind you will see that when thello release something next time its going to be deepseek moment all over agartificial intelligencen	Positif	Positif	Setuju

Dengan menggunakan rumus *Cohen's Kappa*, yakni $k = \frac{P_0 - P_e}{1 - P_e}$, dengan P_0 adalah proporsi kesepakatan yang diamati, yakni $P_0 = \frac{12 + 9 + 6}{30} = 0.9$, dan P_e adalah proporsi kesepakatan yang diharapkan, yakni $P_e = 0.18 + 0.1 + 0.06 = 0.34$, maka $k = \frac{0.9 - 0.34}{1 - 0.34} = 0.84$. Maka dengan ini, bisa dikatakan bahwa validasi dari 1053 teks data yang di labeling manual adalah valid, karena menunjukkan *Score Cohen's Kappa* yang tergolong tinggi.



3.5 Balancing Data dengan SMOTE

Pada tahap ini, data teks yang memiliki ketidakseimbangan kelas diproses menggunakan *SMOTE* untuk menghasilkan data sintetis yang seimbang. *SMOTE* bekerja dengan mengidentifikasi sampel minoritas (dalam hal ini, kelas Negatif dan Netral yang memiliki jumlah lebih sedikit dibandingkan Positif) dan menghasilkan sampel sintetis melalui interpolasi antar tetangga terdekat. Distribusi awal menunjukkan 863 data Negatif, 758 data Netral, dan 1569 data Positif. Sehingga hasil sampel sintetis baru akan disesuaikan menjadi 1569 untuk menghindari bias pada model yang tidak begitu baik dalam menganalisa. Hasil dapat dilihat pada gambar 3 dibawah ini.



Gambar 3. Hasil Penyesuaian *SMOTE*

3.6 Klasifikasi

Proses klasifikasi 3190 data menggunakan algoritma *Naïve Bayes*, *Random Forest*, *Logistic Regression*, KNN, dan SVM dilakukan dengan menggunakan perbandingan 80:20 membantu memaksimalkan penggunaan data untuk pelatihan dan menyediakan subset data yang cukup besar untuk evaluasi kinerja model. Serta akan dilakukan eksperimen dalam lima skenario tuning hyperparameter yang berbeda. Hasil dari setiap *tunning* akan disajikan pada tabel 12, setiap versi *tunning* memiliki tingkat kompleksitas dan eksplorasi parameter yang meningkat, yaitu:



a) V1 – Default

Pengaturan standar dari setiap algoritma digunakan tanpa penyesuaian apa pun. Versi ini dimaksudkan sebagai dasar pembandingan (baseline) untuk melihat sejauh mana model bisa bekerja tanpa dilakukan tuning.

b) V2 – Tuning Ringan

Pada versi ini, beberapa parameter disesuaikan secara ringan untuk meningkatkan performa dasar. Contohnya seperti mengganti kernel SVM menjadi linear, menyesuaikan jumlah tetangga pada KNN, atau mengatur alpha pada Naïve Bayes.

c) V3 – Tuning Sedang

Pengaturan dibuat lebih optimal dan diperluas dibanding versi sebelumnya. Misalnya, jumlah pohon pada Random Forest ditambah, Logistic Regression diatur dengan solver dan iterasi yang lebih baik, serta KNN mulai menggunakan pembobotan jarak.

d) V4 – Tuning Agresif

Versi ini menggunakan konfigurasi yang lebih kompleks agar model mampu menangkap pola yang lebih rumit. SVM dikonfigurasi menggunakan kernel polinomial, Random Forest dibuat lebih dalam dan luas, dan Logistic Regression memiliki pengaturan regularisasi yang lebih ketat.

e) V5 – Eksploratif

Ini merupakan versi eksperimen yang mencoba berbagai kombinasi parameter yang belum digunakan sebelumnya. Beberapa contoh di antaranya adalah penggunaan kernel sigmoid pada SVM, metrik Manhattan pada KNN, serta solver berbeda untuk Logistic Regression. Tujuannya adalah untuk mencari kemungkinan performa terbaik dari konfigurasi yang tidak konvensional.

Tabel 12. Hasil Klasifikasi

Model	Versi	Accuracy	Precision	Recall	F1-Score
Naive Bayes	V1	78.77%	80.78%	78.77%	79.07%
Random Forest		81.53%	81.57%	81.53%	81.51%
Logistic Regression		83.23%	83.22%	83.22%	83.19%
KNN		63.13%	71.50%	63.13%	54.48%
SVM		82.48%	84.99%	82.48%	82.87%
Naive Bayes	V2	80.57%	81.93%	80.57%	80.84%



Random Forest		75.69%	75.83%	75.69%	75.53%
Logistic Regression		81.21%	81.21%	81.21%	81.19%
KNN		63.16%	71.50%	63.13%	54.48%
SVM		84.39%	84.34%	84.39%	84.35%
Naive Bayes		80.99%	82.12%	80.99%	81.25%
Random Forest	V3	78.76%	78.89%	78.79%	78.66%
Logistic Regression		80.04%	80.13%	80.04%	80.06%
KNN		65.71%	73.69%	65.71%	57.34%
SVM		80.25%	83.98%	80.25%	80.74%
Naive Bayes		81.95%	82.80%	81.95%	82.17%
Random Forest	V4	78.13%	78.20%	78.13%	78.03%
Logistic Regression		74.03%	74.93%	74.31%	74.36%
KNN		64.86%	72.77%	64.86%	56.02%
SVM		85.98%	86.06%	85.98%	85.86%
Naive Bayes		78.77%	80.78%	78.77%	79.07%
Random Forest	V5	80.04%	80.14%	80.04%	79.99%
Logistic Regression		84.07%	84.07%	84.07%	84.02%
KNN		41.40%	57.81%	41.40%	31.19%
SVM		82.80%	82.70%	82.80%	82.69%

PEMBAHASAN

Penelitian ini bertujuan untuk menganalisis sentimen pengguna platform X terhadap DeepSeek AI menggunakan lima algoritma klasifikasi (Naïve Bayes, Random Forest, Logistic Regression, K-Nearest Neighbor (KNN), dan Support Vector Machine (SVM)) serta menentukan algoritma paling efektif berdasarkan metrik F1-Score. Hasil menunjukkan bahwa SVM pada skenario V4 (tuning agresif) mencapai performa terbaik dengan F1-Score 85,86%, Accuracy 85,98%, Precision 86,06%, dan Recall 85,98%, menunjukkan keseimbangan optimal dalam mengklasifikasikan sentimen positif, netral, dan negatif. Sebaliknya, KNN pada V4 memiliki F1-Score terendah (56,02%), mengindikasikan performa yang kurang efektif. Distribusi sentimen dari 3.190 tweet (49,2% positif, 23,8% netral, 27,0% negatif) dan nilai Cohen's Kappa (0,84) menegaskan reliabilitas proses labeling.

Keunggulan SVM pada V4 dapat dijelaskan oleh prinsip Structural Risk Minimization (SRM), yang memungkinkan SVM menemukan hyperplane optimal menggunakan kernel polynomial untuk memisahkan kelas sentimen dalam ruang fitur TF-IDF (Praghakusma & Charibaldi, 2021). Kernel polynomial, yang mendukung pemetaan data ke dimensi non-



linear, memungkinkan SVM menangani pola kompleks pada tweet, seperti variasi bahasa informal (Tabel 5: “nahh” → “no”, “gonna” → “going to”). TF-IDF, yang menimbang kata berdasarkan frekuensi dan kekhasan (Yudhanata & Sudarmilah, n.d.), menghasilkan representasi vektor yang mendukung SVM dalam menangkap konteks sentimen. Tuning agresif pada V4, termasuk regularisasi ketat dan optimasi parameter kernel, meningkatkan ketahanan SVM terhadap noise, seperti sarkasme atau ambiguitas, serta cocok untuk dataset besar seperti 3.190 tweet. Sebaliknya, KNN pada V4 menunjukkan performa buruk karena ketergantungannya pada metrik jarak Euclidean, yang rentan terhadap kebisingan dan distribusi data tidak seragam (Tan et al., 2024). Tweet dengan bahasa informal atau konteks ambigu, seperti “european willingness to...” (Tabel 11), menyulitkan KNN untuk mengklasifikasikan dengan akurat berdasarkan tetangga terdekat.

Performa algoritma lain pada V4 menunjukkan Naïve Bayes (F1-Score 82,17%), Random Forest (78,03%), dan Logistic Regression (74,36%). Naïve Bayes, yang menghitung probabilitas berdasarkan frekuensi kata (Pradana & Sugiharti, 2023), memiliki performa lebih baik dari KNN karena sifatnya yang sederhana, tetapi terbatas oleh asumsi independensi fitur, yang kurang realistis pada data teks. Random Forest, yang mengandalkan ensemble pohon keputusan (Pamuji & Ramadhanata, 2021), menunjukkan performa kompetitif, tetapi keterbatasan kedalaman pohon pada V4 membuatnya kalah dari SVM. Logistic Regression, yang memprediksi probabilitas melalui kombinasi linier (Madžar, 2022), kurang efektif dalam menangani hubungan non-linear pada data sentimen, terutama dibandingkan dengan SVM yang menggunakan kernel trick.

Proses pre-processing (cleaning, case folding, normalization, stopword removal, tokenization, stemming) memainkan peran penting dalam meningkatkan kualitas data, mengurangi noise seperti emotikon, hashtag, dan kata tidak baku (Putra & Ummah, 2024; Khoiruddin et al., 2024; Adityarini et al., 2021; Shevira et al., 2022; Alomari & Ahmad, 2024; Munggaran et al., 2023). Penerapan SMOTE untuk menyeimbangkan distribusi kelas (1.569 positif, 758 netral, 863 negatif) terbukti efektif, meningkatkan Recall kelas minoritas (netral dan negatif) tanpa mengorbankan Precision, sebagaimana dilaporkan oleh Chawla et al. (2002). Labeling dengan TextBlob (Chamid et al., 2024) menghasilkan klasifikasi awal yang andal, divalidasi oleh Cohen’s Kappa (0,84), menunjukkan kesepakatan tinggi antar pelabel (Landis & Koch, 1977). Namun, TextBlob menunjukkan keterbatasan dalam menangkap nuansa seperti sarkasme, sebagaimana terlihat pada ketidaksesuaian validasi (Tabel 11: “european willingness to...” dianggap negatif oleh satu panelis dan netral oleh panelis lain), mencerminkan tantangan dalam analisis konteks.



F1-Score dipilih sebagai metrik utama karena menyeimbangkan Precision dan Recall, terutama pada dataset tidak seimbang (Fatah, n.d.). Berbeda dengan Accuracy, yang dapat bias terhadap kelas mayoritas (positif), F1-Score memberikan evaluasi yang lebih adil untuk semua kelas. Tuning hyperparameter (V1–V5) menunjukkan bahwa konfigurasi agresif (V4) secara konsisten meningkatkan performa SVM dan Naïve Bayes, tetapi tuning eksploratif (V5), seperti kernel sigmoid pada SVM atau metrik Manhattan pada KNN, sering kali menurunkan performa, kemungkinan karena ketidaksesuaian parameter dengan karakteristik data tweet.

Kontribusi Penelitian: Penelitian ini memperkuat bukti bahwa SVM dengan kernel polynomial dan tuning agresif adalah algoritma optimal untuk analisis sentimen pada data media sosial, dengan F1-Score 85,86% pada V4. Dibandingkan dengan penelitian Joseph (2022), yang menyarankan pembagian data 80:20 untuk dataset besar, penelitian ini menunjukkan efektivitas pendekatan serupa pada 3.190 tweet, didukung oleh SMOTE dan pre-processing yang ketat. Kontribusi lainnya adalah validasi penggunaan SMOTE untuk meningkatkan performa kelas minoritas (Chawla et al., 2002) dan penerapan TextBlob untuk labeling otomatis (Chamid et al., 2024), menambah literatur tentang analisis sentimen di platform X.

Keterbatasan Penelitian:

1. **Labeling dengan TextBlob:** TextBlob kurang efektif menangkap nuansa seperti sarkasme atau konteks implisit, seperti pada ketidaksesuaian validasi (Tabel 11), membatasi akurasi labeling (Chamid et al., 2024).
2. **Ukuran Dataset:** Dataset 3.190 tweet relatif kecil dibandingkan studi seperti Joseph (2022), yang menggunakan dataset lebih besar, berpotensi mengurangi generalisasi.
3. **Fokus Bahasa Inggris:** Analisis terbatas pada tweet berbahasa Inggris, membatasi relevansi untuk audiens multibahasa.

Implikasi dan Saran: Hasil ini menegaskan bahwa SVM dengan kernel polynomial dan tuning agresif adalah pilihan terbaik untuk analisis sentimen pada data media sosial yang bising, memberikan wawasan bagi pengembang DeepSeek AI untuk memahami persepsi pengguna dan mendukung strategi pemasaran atau pengembangan produk. Keunggulan SVM dalam menangani noise dan data besar, sebagaimana disebutkan di **Penutup**, menjadikannya solusi robust. Untuk penelitian masa depan, disarankan untuk: (1) mengadopsi model deep learning seperti BERT untuk menangkap konteks lebih baik, (2) memperluas dataset dengan tweet multibahasa, (3) meningkatkan validasi manual untuk mengurangi ambiguitas labeling, dan (4) menggunakan grid search untuk optimasi



hyperparameter yang lebih sistematis. Penelitian ini berkontribusi pada literatur analisis sentimen dengan menyoroti pentingnya pre-processing, balancing data, dan tuning hyperparameter dalam konteks platform X.

UCAPAN TERIMA KASIH

Puji syukur penulis panjatkan kepada Allah SWT atas rahmat dan hidayah-Nya, sehingga skripsi ini dapat diselesaikan dengan penuh perjuangan. Proses penyusunan skripsi ini penuh dengan tantangan, namun berkat bantuan dan dukungan dari berbagai pihak, penulis mampu menyelesaikannya dengan tekad kuat. Ucapan terimakasih penulis ucapkan kepada Bapak Widi Widayat, S.Kom., M.Eng. selaku dosen pembimbing, bapak Dedi Gunawan, S.T., M.Sc., Ph.D. serta bapak Dr.Eng. Yusuf Sulisty Nugroho, S.T., M.Eng. selaku dosen penguji bagi penulis. Serta kepada kedua orang tua tercinta, Bapak Didik Nugroho dan Ibu Sri Mulyani, terimakasih atas doa, kasih sayang, dan dukungan tanpa henti. Terimakasih kepada saudara serta teman – teman Teknik Informatika angkatan 2020 yang selalu membantu penulis dalam menulis penelitian ini. Serta terimakasih atas dukungan dari Sekar Arum Nityasa, S.Pd. terhadap penulis. Penulis menyadari skripsi ini masih memiliki kekurangan. Kritik dan saran yang membangun sangat diharapkan. Semoga skripsi ini bermanfaat bagi dunia akademik dan masyarakat luas. Dengan tekad baja, penulis mengejar mimpi menyelesaikan skripsi ini, terinspirasi oleh Roronoa Zoro (*One Piece*): "*When I decided to follow my dream, I already threw my life away*" yang berarti rela mengorbankan segalanya demi tujuan.



DAFTAR PUSTAKA (format IEEE)

- [1] R. Abraham, "Democratizing AI's frontiers: A critical review of DeepSeek AI's open-source ecosystem," 2025, doi: 10.31219/osf.io/9r7e3_v1.
- [2] E. Adityarini, S. Nur Ayuni, and R. Aminatas Sa'adiah, "Analisis sentimen terhadap ulasan produk pada sistem penjualan toko Putra Elektronik," *Journal of Islamic Business Management Studies*, vol. 2, no. 2, pp. 84–98, 2021, doi: 10.51875/jibms.v2i2.184.
- [3] D. Alomari and I. Ahmad, "Exploring character trigrams for robust Arabic text classification: A comparative analysis in the face of vocabulary expansion and misspelled words," *IEEE Access*, vol. 12, pp. 57103–57116, 2024, doi: 10.1109/ACCESS.2024.3390048.
- [4] A. A. Chamid, Widowati, and R. Kusumaningrum, "Labeling consistency test of multi-label data for aspect and sentiment classification using the Cohen Kappa method," *Ingénierie des Systèmes d'Information*, vol. 29, no. 1, pp. 161–167, 2024, doi: 10.18280/isi.290118.
- [5] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002, doi: 10.1613/jair.953.
- [6] D. Darwis, N. Siskawati, and Z. Abidin, "Penerapan algoritma Naive Bayes untuk analisis sentimen review data Twitter BMKG Nasional," *Jurnal Tekno Kompak*, vol. 15, no. 1, pp. 131, 2021, doi: 10.33365/jtk.v15i1.744.
- [7] M. L. Islam, M. Mustufa, S. Shaikh, U. Shaikh, and Z. Kazi, "Twitter sentiment analysis," *International Journal of Advanced Research in Science, Communication and Technology*, pp. 476–481, 2023, doi: 10.48175/IJARSCT-9448.
- [8] Z. Fatah, "Klasifikasi jenis kendaraan menggunakan decision tree dan evaluasi akurasi melalui confusion matrix," n.d.
- [9] T. C. Herdiyani and A. U. Zailani, "Sentiment analysis terkait pemindahan ibu kota Indonesia menggunakan metode Random Forest berdasarkan tweet warga negara Indonesia," *Jurnal Teknologi Sistem Informasi*, vol. 3, no. 2, pp. 154–165, 2022, doi: 10.35957/jtsi.v3i2.2920.



- [10] H. Imaduddin, F. Y. A'la, and Y. S. Nugroho, "Sentiment analysis in Indonesian healthcare applications using IndoBERT approach," vol. 14, no. 8, 2023, doi: 10.14569/IJACSA.2023.0140813.
- [11] V. R. Joseph, "Optimal ratio for data splitting," *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 15, no. 4, pp. 531–538, 2022, doi: 10.1002/sam.11583.
- [12] L. Khoiruddin, A. Sidauruk, Y. Pristyanto, M. R. A. Yudiyanto, and H. Kurniawan, "Otomatisasi penerusan laporan pengaduan menggunakan neural network," *Smart Comp: Jurnalnya Orang Pintar Komputer*, vol. 13, no. 2, 2024, doi: 10.30591/smartcomp.v13i2.6662.
- [13] P. Kurniawati, R. Y. Fa'rifah, and D. Witarsyah, "Sentiment analysis of Maxim online transportation app reviews using Support Vector Machine (SVM) algorithm," *Building of Informatics, Technology and Science (BITS)*, vol. 5, no. 2, 2023, doi: 10.47065/bits.v5i2.4265.
- [14] J. R. Landis and G. G. Koch, "The measurement of observer agreement for categorical data," *Biometrics*, vol. 33, no. 1, pp. 159, 1977, doi: 10.2307/2529310.
- [15] L. Madžar, "Motives for the introduction of agricultural innovations in Serbia with particular accent on beekeepers: The application of logistic regression," *Ekonomika Poljoprivrede*, vol. 69, no. 1, pp. 27–41, 2022, doi: 10.5937/ekoPolj2201027M.
- [16] J. P. Munggaran, A. A. Alhafidz, M. Taqy, D. A. R. Agustini, and M. Munawir, "Sentiment analysis of Twitter users' opinion data regarding the use of ChatGPT in education," *Journal of Computer Engineering, Electronics and Information Technology*, vol. 2, no. 2, pp. 75–88, 2023, doi: 10.17509/coelite.v2i2.59645.
- [17] F. Y. Pamuji and V. P. Ramadhan, "Komparasi algoritma Random Forest dan Decision Tree untuk memprediksi keberhasilan immunotherapy," *Jurnal Teknologi dan Manajemen Informatika*, vol. 7, no. 1, pp. 46–50, 2021, doi: 10.26905/jtmi.v7i1.5982.
- [18] D. Pradana and E. Sugiharti, "Implementation data mining with Naive Bayes classifier method and Laplace smoothing to predict students learning results," *Recursive Journal of Informatics*, vol. 1, no. 1, pp. 1–8, 2023, doi: 10.15294/rji.v1i1.63964.
- [19] A. Z. Praghakusma and N. Charibaldi, "Komparasi fungsi kernel metode Support Vector Machine untuk analisis sentimen Instagram dan Twitter (studi kasus: Komisi Pemberantasan Korupsi)," *JSTIE (Jurnal Sarjana Teknik Informatika) (E-Journal)*, vol. 9, no. 2, pp. 88, 2021, doi: 10.12928/jstie.v9i2.20181.



- [20] D. Pratmanto, F. F. D. Imaniawan, and V. Maarif, "Analisis sentimen pada ulasan pengguna aplikasi identitas kependudukan digital dengan metode Naive Bayes dan K-Nearest," *Computatio: Journal of Computer Science and Information Systems*, vol. 7, no. 2, pp. 155–166, 2023, doi: 10.24912/computatio.v7i2.26322.
- [21] R. Z. B. Putra and K. R. Ummah, "Analisis sentimen komentar Twitter terhadap peristiwa kebakaran Bromo menggunakan metode Naive Bayes dan Support Vector Machine," 2024, <http://eprints.ums.ac.id/id/eprint/124836>.
- [22] A. Saepudin, A. Faqih, and G. Dwilestari, "Perbandingan algoritma klasifikasi Support Vector Machine, Random Forest dan Logistic Regression pada ulasan Shopee," *Jurnal Tekno Kompak*, vol. 18, no. 1, pp. 178, 2024, doi: 10.33365/jtk.v18i1.3764.
- [23] D. Setiyawati and N. Cahyono, "Analisa sentimen pengguna sosial media Twitter terhadap perokok di Indonesia," 2023, doi: 10.33022/ijcs.v12i1.3154.
- [24] S. Shevira, I. M. A. D. Suarjaa, and P. W. Buana, "Pengaruh kombinasi dan urutan pre-processing pada tweets bahasa Indonesia," *JITTER- Jurnal Ilmiah Teknologi dan Komputer*, vol. 3, no. 2, 2022.
- [25] D. F. Sjoradia, B. W. K. Guna, and D. Yudhakusuma, "Analisis sentimen film Dirty Vote menggunakan BERT (Bidirectional Encoder Representations from Transformers)," vol. 8, no. 2, pp. 393–404, 2024, doi: 10.35870/jtik.v8i2.1580.
- [26] R. Stevany, "Indonesia pengguna X atau Twitter terbanyak keempat di dunia," 2024, <https://rri.co.id/lain-lain/859350/indonesia-pengguna-x-atau-twitter-terbanyak-keempat-di-dunia>.
- [27] L. Tan, J. Yi, and F. Yang, "Improving performance of massive text real-time classification for document confidentiality management," *Applied Sciences*, vol. 14, no. 4, pp. 1565, 2024, doi: 10.3390/app14041565.
- [28] Y. Wang, E. Chukwusa, J. Koffman, and V. Curcin, "Public opinions about palliative and end-of-life care during the COVID-19 pandemic: Twitter-based content analysis," *JMIR Formative Research*, vol. 7, pp. e44774, 2023, doi: 10.2196/44774.
- [29] I. D. Yudhanata and E. Sudarmilah, "Analisis sentimen terhadap isu Islamofobia pada platform Twitter menggunakan metode klasifikasi Naive Bayes," n.d.

