

Perbandingan Kinerja Model *Probabilistic*, *Linear Model*, *Instance-Based*, dan *Ensemble Learning* (Studi Kasus: Ulasan *Google Playstore* Aplikasi *Threads*)

Auriel Diya Ulhaque Muntaz Waris¹⁾, Novera Kristianti²⁾, Felicia Sylviana³⁾

¹⁾²⁾³⁾Jurusan Teknik Informatika, Fakultas Teknik, Universitas Palangka Raya
Kampus Tanjung Nyaho, Jl. Yos Sudarso, Palangka Raya

¹⁾aurielhaquedhea@gmail.com

²⁾noverakristianti@eng.upr.ac.id

³⁾felicia.upr@it.upr.ac.id

Abstrak

Pertumbuhan *Threads* sebagai media sosial baru memunculkan kebutuhan akan pemahaman sentimen pengguna. Ulasan di *Google Play Store* menjadi sumber penting untuk menangkap persepsi publik terhadap fitur dan performa aplikasi. Sebagai *platform* yang masih berkembang, *Threads* menyediakan peluang strategis bagi pengembang untuk merespons masukan secara tepat. Namun, besarnya volume dan format ulasan yang tidak seragam menyulitkan proses analisis secara manual. Penelitian ini membandingkan kinerja empat algoritma *Naïve Bayes* (*probabilistic*), *SVM* (*linear*), *K-NN* (*instance-based*), dan *Random Forest* (*ensemble*) dalam klasifikasi sentimen ulasan *Threads*. Penelitian ini dimulai dengan studi literatur dan pengumpulan data ulasan *Threads* melalui *scraping*. Dilabeli menggunakan *IndoBERT*, proses *preprocessing* mencakup *remove duplicate*, *cleaning*, *normalization*, *tokenization*, *stopwords* dan *lemmatization*, diekstraksi dengan *TF-IDF*, diseimbangkan memakai *SMOTE*, lalu dibagi 80:20 untuk latih dan uji. Evaluasi mencakup akurasi, presisi, *recall*, *F1-score*, serta waktu pelatihan dan prediksi. Hasil menunjukkan *Random Forest* menjadi algoritma terbaik (akurasi 81,8%; *F1-score* 76,6%), disusul *SVM* (81,4%; 75,9%), *Naïve Bayes* (79,1%; 72,4%), dan *K-NN* (61,2%; 57,6%). *Random Forest* unggul hampir di semua metrik, *SVM* efisien dengan performa seimbang, *Naïve Bayes* menonjol pada kecepatan, sementara *K-NN* lambat pada prediksi. Untuk implementasi, dibangun *dashboard* interaktif berbasis *Streamlit* guna memvisualisasikan perkembangan dan prediksi sentimen pengguna *Threads*.

Kata kunci: *Threads*, *Machine Learning*, Sentimen

Abstract

The growth of Threads as a new social media platform has created a need to understand user sentiment. Reviews on Google Play Store are an important source for capturing public perception of app features and performance. As a growing platform, Threads provides a strategic opportunity for developers to respond appropriately to feedback. However, the large volume and inconsistent format of reviews make manual analysis difficult. This study compares the performance of four algorithms: Naïve Bayes (probabilistic), SVM (linear), K-NN (instance-based), and Random Forest (ensemble) in classifying the sentiment of Threads reviews. This study begins with a literature review and the collection of Threads review data through scraping. Labeled using IndoBERT, the preprocessing process included removing duplicates, cleaning, normalization, tokenization, stopwords, and lemmatization, extracted with TF-IDF, balanced using SMOTE, then divided 80:20 for training and testing. The evaluation included accuracy, precision, recall, F1-score, as well as training and prediction time. The results show that Random Forest is the best algorithm (81.8% accuracy; 76.6% F1-score), followed by SVM (81.4%; 75.9%), Naïve Bayes (79.1%; 72.4%), and K-NN (61.2%; 57.6%). Random Forest excelled in almost all metrics, SVM was efficient with balanced performance, Naïve Bayes stood out in terms

of speed, while *K-NN* was slow in prediction. For implementation, an interactive dashboard based on *Streamlit* was built to visualize the development and prediction of *Threads* user sentiment.

Keywords: *Threads, Machine Learning, Sentiment*

1. PENDAHULUAN

Perkembangan teknologi digital telah mengubah cara manusia berkomunikasi, berinteraksi, dan memperoleh informasi, dengan media sosial menjadi wadah utama dalam berbagi opini. Salah satu *platform* terbaru yang menarik perhatian adalah *Threads*, aplikasi besutan *Meta* yang dirancang sebagai pesaing *Twitter* (kini *X*). Hanya dalam lima hari setelah peluncuran, *Threads* mencatat lebih dari 100 juta pengguna [1] dan pada awal 2025 mencapai 320 juta pengguna aktif bulanan [2]. Meski pertumbuhannya pesat, kepuasan pengguna tetap menjadi faktor kunci keberhasilan di tengah persaingan ketat media sosial.

Threads dipilih sebagai objek penelitian karena masih berada pada tahap pengembangan awal, di mana sejumlah fitur dasar belum tersedia sejak peluncuran [3]. Hal ini sejalan dengan laporan yang menyebutkan bahwa *Threads* dirilis dalam kondisi belum lengkap sehingga masukan pengguna menjadi penting untuk penyempurnaan aplikasi [4]. Ulasan di *Google Play Store* menjadi sumber utama kepuasan dan kritik, yang tidak hanya memengaruhi calon pengguna tetapi juga menyediakan data strategis bagi pengembang [5]. Namun, *rating* sering tidak konsisten dengan isi ulasan, misalnya komentar positif disertai *rating* rendah [6]. Selain itu, volume ulasan yang besar dan tidak terstruktur membuat analisis manual tidak *efisien*, sehingga diperlukan analisis sentimen berbasis *machine learning* untuk mengelompokkan opini secara otomatis.

Sejumlah penelitian sebelumnya telah dilakukan. Java et al. [7] membandingkan *Multinomial Naïve Bayes* dan *SVM* pada 4.352 ulasan *Threads* dengan akurasi 81%. Wulandari et al. [8] menguji *Naïve Bayes* dan *K-NN* pada komentar terkait *IKN*, dengan *Naïve Bayes* unggul (70% akurasi) dibanding *K-NN* (60%). Madyatmadja et al. [5] meneliti 2.583 ulasan *Threads* menggunakan *Naïve Bayes*, *SVM*, dan *Random Forest*, dengan hasil terbaik pada *Naïve Bayes* (81%). Sementara itu, Muliawan et al. [9] membandingkan *Naïve Bayes*, *K-NN*, dan *Random Forest* pada 1.639 *tweet*, dengan *Naïve Bayes* tetap unggul (65,26%). Dari berbagai penelitian tersebut terlihat bahwa metode *balancing data*, khususnya *SMOTE*, masih jarang digunakan, padahal teknik ini penting untuk mengatasi ketidakseimbangan kelas.

Berdasarkan hal tersebut, penelitian ini membandingkan empat algoritma dari kategori berbeda, yaitu *Naïve Bayes (probabilistic)*, *SVM (linear)*, *K-NN (instance-based)*, dan *Random Forest (ensemble)*. *TF-IDF* digunakan untuk ekstraksi fitur, sedangkan *SMOTE* diterapkan untuk menyeimbangkan data, dengan tujuan memperoleh model terbaik dalam klasifikasi sentimen *Threads* serta memberi kontribusi bagi analisis opini di media sosial.

2. TINJAUAN PUSTAKA

2.1 Analisis Sentimen

Analisis sentimen atau *opinion mining* merupakan bidang penelitian yang fokus pada pemahaman opini, penilaian, dan reaksi emosional terhadap produk, layanan, organisasi, maupun isu tertentu [10]. Menurut Nufairi et al. [11], dapat dipahami sebagai teknik untuk identifikasi pandangan positif atau negatif dari suatu teks melalui proses ekstraksi otomatis, sehingga mampu memberikan informasi penting yang berguna dalam mendukung pengambilan keputusan.

2.2 SMOTE (Synthetic Minority Over-sampling Technique)

SMOTE (Synthetic Minority Over-sampling Technique) adalah teknik *oversampling* yang digunakan untuk mengatasi ketidakseimbangan kelas dalam *dataset*. Metode ini menghasilkan sampel sintesis pada kelas minoritas melalui interpolasi (contoh data baru) antar data sehingga mengurangi risiko *overfitting* dan meningkatkan performa klasifikasi [12].

2.3 Probabilistic (Naïve Bayes)

Model *probabilistic* dalam *machine learning* memanfaatkan prinsip probabilitas untuk memprediksi hasil. Contoh yang umum digunakan meliputi *Linear Regression*, *Logistic Regression*, *Generalized Linear Models (GLM)*, *Linear Discriminant Analysis*, dan *Naïve Bayes* [13]. Dalam penelitian ini dipilih algoritma *Naïve Bayes* karena kesederhanaannya, efisiensi

komputasi, serta kemampuannya menangani data teks berdimensi tinggi. *Naïve Bayes* bekerja dengan menghitung probabilitas kategori berdasarkan data sebelumnya [14]. Versi yang banyak digunakan untuk klasifikasi teks adalah *Multinomial Naïve Bayes*, yang mengandalkan frekuensi kata untuk memprediksi kelas dokumen [15].

2.4 Linear (SVM)

Linear model menggambarkan hubungan langsung antar variabel dalam bentuk garis lurus atau *hyperplane* [16]. Contoh *linear model* meliputi *linear regression* untuk prediksi nilai kontinu, *logistic regression* untuk klasifikasi biner dengan fungsi sigmoid, serta *Support Vector Machine (SVM)* linear yang mencari *hyperplane* pemisah dengan margin maksimal [17]. Dibanding *linear regression* yang hanya cocok untuk prediksi nilai kontinu, serta *logistic regression* yang efektif untuk klasifikasi namun kurang optimal pada data berdimensi tinggi, *SVM* linear lebih unggul dalam menangani data teks dengan fitur besar dan memberikan margin pemisahan yang jelas [17]. Oleh karena itu, penelitian ini memilih *SVM* linear karena stabil dan *robust* dalam klasifikasi teks.

2.5 Ensemble Learning (Random Forest)

Ensemble learning menggabungkan beberapa model untuk meningkatkan akurasi dan mengurangi kesalahan. Metode utamanya meliputi *bagging (Random Forest)*, *boosting (AdaBoost, XGBoost)*, dan *stacking (meta-model)*. Pemilihan *Random Forest* dalam penelitian ini didasarkan pada temuan Iman et al. [18] yang menunjukkan bahwa algoritma ini memiliki akurasi lebih tinggi (68%) dibandingkan *XGBoost*. Selain itu, *Random Forest* memiliki akurasi tinggi, cepat dalam pemrosesan, tahan terhadap *overfitting*, mampu menangani data hilang, serta tidak memerlukan normalisasi, meski membutuhkan komputasi besar dan sulit diinterpretasikan [19].

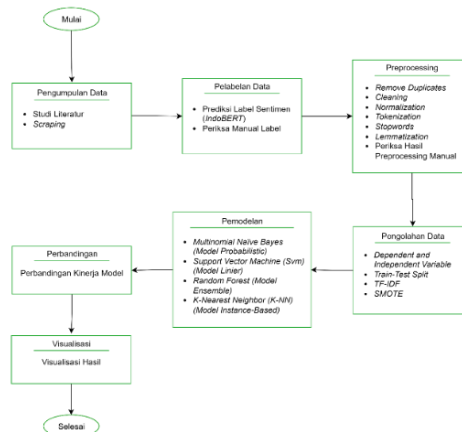
2.6 Instance Based (K-NN)

Instance-based learning adalah metode *lazy learning* yang menyimpan data pelatihan dan menggunakan contoh tersebut langsung untuk prediksi tanpa membentuk model generalisasi. Contoh metodenya meliputi *K-Nearest Neighbor (KNN)*, *Locally Weighted Regression (LWR)*, *Radial Basis Functions (RBF)*, dan *Case-Based Reasoning (CBR)* [22]. Pada penelitian ini dipilih *KNN* karena sederhana, mudah diimplementasikan, dan tidak memerlukan pemodelan kompleks. Cara kerjanya adalah dengan mencari sejumlah tetangga terdekat dari data uji, kemudian menentukan kelas berdasarkan mayoritas kelas tetangga tersebut.

2.7 Preprocessing

Tahap ini bertujuan mengatasi permasalahan seperti *missing value*, duplikasi, atau format data yang tidak sesuai [23]. Langkah-langkah yang diterapkan dalam penelitian ini meliputi: *remove duplicates* untuk menghapus entri ganda agar data unik [21], *cleaning* menghilangkan karakter selain huruf a–z seperti angka atau tanda baca [7], *normalization* penyesuaian kata sesuai standar KBBI [11], *tokenization* memecah kalimat menjadi unit kata untuk menghitung frekuensi distribusinya [7], *stopword removal* penghapusan kata umum yang tidak bermakna signifikan [9], serta *lemmatization* mengubah kata ke bentuk dasar sesuai konteks, berbeda dengan *stemming* yang hanya memotong akhiran kata [20].

3. METODE PENELITIAN



Gambar 1. Metodologi Penelitian

Penelitian ini diawali dengan studi literatur dan *scraping* data, lalu dilakukan pelabelan menggunakan *IndoBERT* yang diverifikasi manual. Data diproses melalui *pre-processing* (*remove duplicates, cleaning, normalization, tokenization, stopwords removal, lemmatization*) dan diperiksa ulang. Selanjutnya, dilakukan *train-test split*, ekstraksi fitur dengan *TF-IDF*, serta penyeimbangan data menggunakan *SMOTE*. Pemodelan menggunakan *Multinomial Naïve Bayes, SVM, Random Forest*, dan *KNN*, kemudian dibandingkan kinerjanya melalui evaluasi dan visualisasi.

4. PEMBAHASAN

4.1 Pengumpulan Data

Data penelitian dikumpulkan melalui proses *scraping* menggunakan library *google-play-scraper*. Jumlah data terkumpul sebanyak 16.717 ulasan berbahasa Indonesia dengan rentang waktu dari Juli 2023 hingga Juni 2025.

Table 1. Hasil Scraping

Text	Date	Rating	Label Rating
Cukup elegan dan nyaman seharian. Aplikasi enteng dan mudah dipahami	06/07/2023	1	Negatif
gbs donlot	06/07/2023	1	Negatif
Hasilnya sangat baik dan cepat	10/04/2024	5	Positif

4.2 Labeling Data

Pelabelan data dilakukan dengan model pralatih *mdhugol/indonesia-BERT-sentiment-classification* berbasis *IndoBERT*. Hasil prediksi disertai *confidence score* sebagai tingkat keyakinan model, namun tetap diperiksa manual untuk memastikan akurasi.

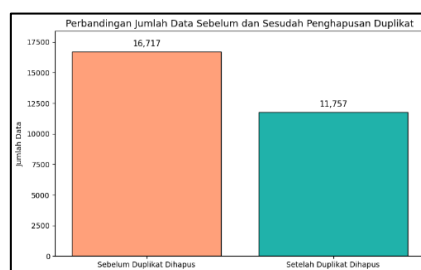
Table 2. Hasil Labeling Data

Text	Date	Label Rating	Label Model	Confidence Score
Cukup elegan dan nyaman seharian. Aplikasi enteng dan mudah dipahami	06/07/2023	Negatif	Positif	0.990835
gbs donlot	06/07/2023	Negatif	Negatif	0.997644
Hasilnya sangat baik dan cepat	10/04/2024	Positif	Positif	0.990468

4.3 Preprocessing Data

4.3.1 Remove duplicate

Dari 16.717 data awal, sebanyak 4.960 terdeteksi duplikat sehingga tersisa 11.757 data unik. Langkah ini penting untuk menjaga keberagaman data dan meningkatkan kualitas prediksi.



Gambar 2. Sebelum dan Sesudah *Remove Duplicates*

4.3.2 Cleaning

Teks diubah menjadi huruf kecil, dibersihkan dari *URL, tag HTML*, tanda baca, dan karakter *non-ASCII*.

Table 3. Hasil *Cleaning*

Sebelum <i>Cleaning</i>	Sesudah <i>Cleaning</i>
Cukup elegan dan nyaman seharian. Aplikasi enteng dan mudah dipahami	cukup elegan dan nyaman seharian aplikasi enteng dan mudah dipahami
gbs donlot	gbs donlot
Hasilnya sangat baik dan cepat	hasilnya sangat baik dan cepat

4.3.3 Normalization

Proses normalisasi dilakukan untuk mengubah kata-kata tidak baku atau slang menjadi bentuk baku menggunakan kamus bahasa yang telah disiapkan.

Table 4. Hasil *Normalization*

Sebelum <i>Normalization</i>	Sesudah <i>Normalization</i>
cukup elegan dan nyaman seharian aplikasi	cukup elegan dan nyaman seharian aplikasi
enteng dan mudah dipahami	enteng dan mudah dipahami
gbs donlot	tidak bisa download
hasilnya sangat baik dan cepat	hasilnya sangat baik dan cepat

4.3.4 Tokenization

Teks yang dinormalisasi dipisahkan menjadi kata-kata (*token*) menggunakan *RegexTokenizer* *NLTK* dengan pola $|w+$, lalu disimpan di kolom *tokens* sebagai persiapan ekstraksi fitur.

Table 5. Hasil *Tokenization*

Sebelum <i>Tokenization</i>	Sesudah <i>Tokenization</i>
cukup elegan dan nyaman seharian aplikasi	['cukup', 'elegan', 'dan', 'nyaman', 'sehariannya', 'aplikasi', 'enteng', 'dan', 'mudah', 'dipahami']
enteng dan mudah dipahami	
tidak bisa download	['tidak', 'bisa', 'download']
hasilnya sangat baik dan cepat	['hasilnya', 'sangat', 'baik', 'dan', 'cepat']

4.3.5 Stopwords

Kata tidak bermakna dihapus untuk meningkatkan kualitas fitur teks, menggunakan daftar *NLTK*, namun kata-kata negasi tetap dipertahankan karena penting untuk analisis sentimen.

Table 6. Hasil *Stopwords*

Sebelum <i>Stopwords</i>	Sesudah <i>Stopwords</i>
['cukup', 'elegan', 'dan', 'nyaman', 'sehariannya', 'aplikasi', 'enteng', 'dan', 'mudah', 'dipahami']	['elegan', 'nyaman', 'sehariannya', 'aplikasi', 'enteng', 'mudah', 'dipahami']
['tidak', 'bisa', 'download']	['tidak', 'download']
['hasilnya', 'sangat', 'baik', 'dan', 'cepat']	['hasil', 'sangat', 'baik', 'cepat']

4.3.6 Lemmatization

Kata-kata diubah ke bentuk dasar menggunakan *Stanza*, hasil *lemmatization* disimpan di *lemmatized* dan digabung menjadi *final_text* sebagai representasi akhir untuk vektorisasi dan klasifikasi sentimen.

Table 7. Hasil *lemmatized*

Sebelum <i>lemmatized</i>	Sesudah <i>lemmatized</i>
['elegan', 'nyaman', 'sehariannya', 'aplikasi', 'enteng', 'mudah', 'dipahami']	elegan nyaman seharian aplikasi enteng mudah paham
['tidak', 'download']	tidak download
['hasil', 'sangat', 'baik', 'cepat']	hasil sangat baik cepat

4.4 Pengolahan Data

4.4.1 Penentuan *Dependent* dan *Independent Variable*

Data teks yang telah diproses dan dilabeli dipisahkan menjadi variabel independen (X : *final_text*) dan dependen (y : label sentimen) agar model *machine learning* dapat mempelajari hubungan antara isi ulasan dan kelas sentimennya.

4.4.2 Split data

Data dibagi menggunakan *train-test split* 80:20, menghasilkan 9.072 data latih dan 2.268 data uji. Pembagian dilakukan sebelum ekstraksi fitur *TF-IDF* untuk memastikan pelatihan dan pengujian model berlangsung adil dan bebas dari kebocoran data.

4.4.3 Ekstraksi Fitur

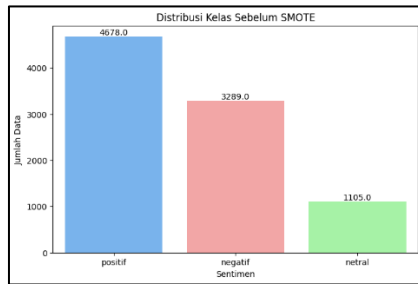
Setelah data dibagi menjadi latih dan uji, teks diubah menjadi bentuk numerik menggunakan *TF-IDF* dengan *TfidfVectorizer* dari *scikit-learn*. Hasil vektorisasi disimpan dalam *data frame* dan objek vektorisasi disimpan dengan *joblib*. Penelitian ini menggunakan unigram karena efektif menangkap kata kunci penting untuk klasifikasi sentimen tanpa menimbulkan pertumbuhan fitur berlebihan [24].

word	tfidf_score
bagus	573.023273
aplikasi	419.312158
tidak	365.183805
sangat	253.647607
bisa	240.020625
keren	211.958601
threads	208.969259
oke	204.915531
mantap	191.293133
suka	167.117406

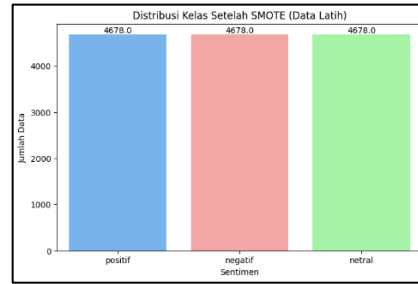
Gambar 3. Hasil TF-IDF

4.4.4 Teknik SMOTE

SMOTE diterapkan pada data latih untuk menyeimbangkan distribusi kelas, sehingga meningkatkan *performa* model pada data yang tidak seimbang.



Gambar 4. Sebelum SMOTE



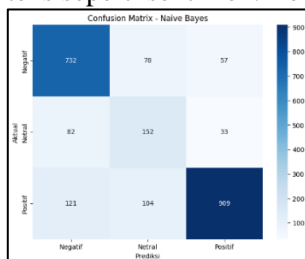
Gambar 5. Sesudah SMOTE

Setelah *SMOTE*, distribusi kelas lebih seimbang, memungkinkan model belajar tiap kelas secara adil, meningkatkan akurasi, dan mengurangi bias terhadap kelas mayoritas.

4.5 Pemodelan Data

4.5.1 Multinomial Naïve Bayes (Model Probabilistic)

Naïve Bayes, model berbasis probabilitas dengan asumsi independensi fitur, cocok untuk klasifikasi teks seperti sentimen. Berikut evaluasi kinerjanya:



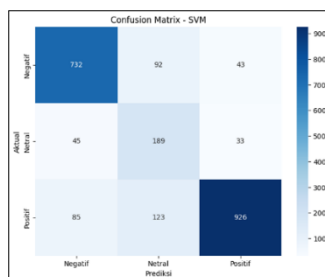
Gambar 6. Matrix Naïve Bayes

Laporan Klasifikasi:				
	precision	recall	f1-score	support
Negatif	0.783	0.844	0.812	867
Netral	0.455	0.569	0.506	267
Positif	0.910	0.802	0.852	1134
accuracy			0.791	2268
macro avg	0.716	0.738	0.724	2268
weighted avg	0.808	0.791	0.796	2268

Gambar 7. Laporan Klasifikasi Naïve Bayes

4.5.2 Support Vector Machine (SVM) (Model Linier)

Support Vector Machine digunakan untuk mengklasifikasikan data sentimen dengan membentuk *hyperplane* pemisah antar kelas. Berikut evaluasi kinerjanya:



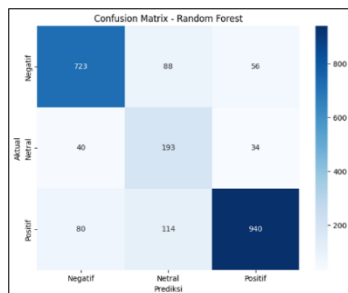
Gambar 8. Matrix SVM

Laporan Klasifikasi:				
	precision	recall	f1-score	support
Negatif	0.849	0.844	0.847	867
Netral	0.468	0.708	0.563	267
Positif	0.924	0.817	0.867	1134
accuracy			0.814	2268
macro avg	0.747	0.790	0.759	2268
weighted avg	0.842	0.814	0.824	2268

Gambar 9. Laporan Klasifikasi SVM

4.5.3 Random Forest (Model Ensemble)

Random Forest merupakan model berbasis pohon keputusan yang memanfaatkan voting dari banyak pohon untuk melakukan klasifikasi. Hasil evaluasinya sebagai berikut:

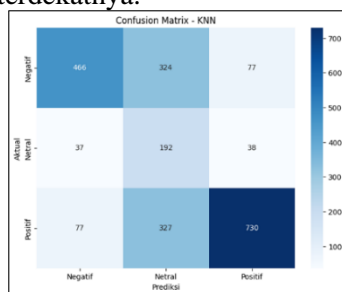


	precision	recall	f1-score	support
Negatif	0.858	0.834	0.846	867
Netral	0.489	0.723	0.583	267
Positif	0.913	0.829	0.869	1134
accuracy			0.818	2268
macro avg	0.753	0.795	0.766	2268
weighted avg	0.842	0.818	0.826	2268

Gambar 10. Matrix Random Forest Gambar 11. Laporan Klasifikasi Random Forest

4.5.4 K-Nearest Neighbor (K-NN) (Model Instance-Based)

Model K-Nearest Neighbor melakukan klasifikasi berdasarkan kedekatan data baru terhadap data latih terdekatnya.



	precision	recall	f1-score	support
Negatif	0.803	0.537	0.644	867
Netral	0.228	0.719	0.346	267
Positif	0.864	0.644	0.738	1134
accuracy			0.612	2268
macro avg	0.632	0.633	0.576	2268
weighted avg	0.766	0.612	0.656	2268

Gambar 12. Matrix K-NN Gambar 13. Laporan Klasifikasi K-NN

4.6 Perbandingan Data

Perbandingan performa model klasifikasi sentimen dilakukan untuk menilai kemampuan model mengelompokkan opini menjadi Positif, Netral, dan Negatif. Model yang diuji meliputi Naïve Bayes, SVM, K-Nearest Neighbors, dan Random Forest dengan pembagian data 80:20, dievaluasi menggunakan accuracy, precision, recall, F1-score, serta waktu pelatihan dan prediksi. Hasil lengkap disajikan pada tabel berikut.

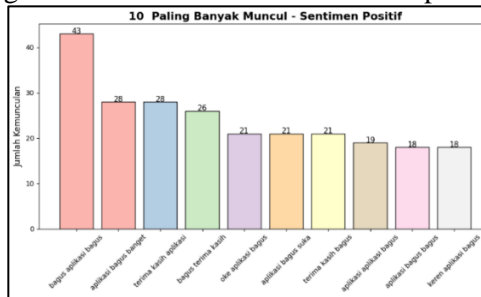
Algoritma	Akurasi	Presisi	Recall	F1-Score	Waktu Pelatihan	Waktu Prediksi
Naïve Bayes	79,1%	71,6%	73,8%	72,4%	0.0404	0.0007 s
SVM	81,4%	74,7%	79%	75,9%	0.0645	0.0007 s
Random Forest	81,8%	75,3%	79,5%	76,6%	20.621	0.2517 s
KNN	61,2%	63,2%	63,3%	57,6%	0.0204	0.7204 s

Berdasarkan hasil pengujian, Random Forest menempati posisi terbaik. Keunggulan ini karena Random Forest menggunakan ensemble learning dengan banyak decision tree, teknik bagging, dan pemilihan fitur acak, sehingga prediksi lebih stabil meski waktu pelatihan lebih lama [19][25]. SVM berada di posisi kedua dengan akurasi 81,4% dan F1-score 75,9%, mampu memisahkan kelas secara optimal melalui hyperplane dan bekerja efektif pada data berdimensi tinggi [26]. Naïve Bayes menempati posisi ketiga dengan akurasi 79,1% dan F1-score 72,4%, unggul dalam menangani data berdimensi tinggi dan sangat cepat dalam pelatihan serta prediksi [8]. Kinerja terendah dicatat K-Nearest Neighbors dengan akurasi 61,2% dan F1-score 57,6%, karena merupakan lazy learner yang membandingkan data uji dengan seluruh data latih sehingga kurang optimal untuk data TF-IDF berdimensi tinggi.

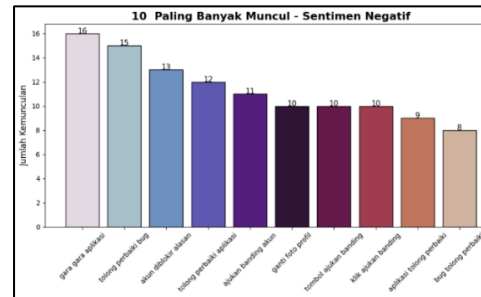
4.7 Visualisasi Data

4.7.1 Visualisasi Frekuensi Frasa Dominan per Sentimen

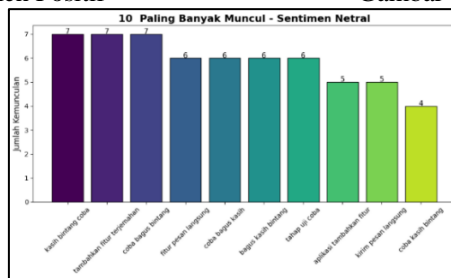
Tahap selanjutnya adalah visualisasi frasa dominan berdasarkan kategori sentimen Positif, Negatif, dan Netral untuk memahami pola bahasa yang sering muncul dalam ulasan pengguna *Threads*. Pada visualisasi digunakan *trigram* agar konteks bahasa terlihat lebih jelas, sehingga frasa populer seperti “aplikasi bagus suka” atau “tolong perbaiki bug” muncul utuh. Pendekatan ini sejalan dengan penelitian Purbaya et al. [24], yang menunjukkan bahwa visualisasi berbasis *trigram* lebih informatif untuk menampilkan ekspresi sentimen.



Gambar 14. Sentimen Positif



Gambar 15. Sentimen Negatif



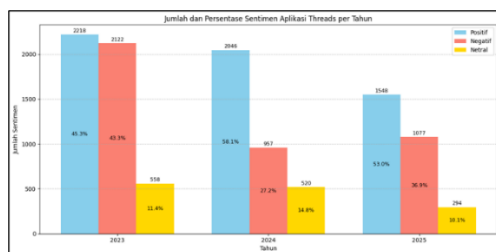
Gambar 16. Sentimen Netral

Pada sentimen Positif muncul frasa “bagus”, “terima kasih aplikasi”, dan “oke aplikasi bagus” yang menunjukkan kepuasan serta *User Experience and Satisfaction* [27], sehingga pengembang disarankan mempertahankan kualitas dan kenyamanan aplikasi. Sentimen Negatif menampilkan frasa “gara gara aplikasi”, “tolong perbaiki bug”, dan “aku diblokir alasan” yang berkaitan dengan *bug*, keterbatasan fitur, dan masalah akun, sesuai penelitian Hanif et al. [28] serta Pidada & Dewi [29], sehingga pengembang perlu memprioritaskan perbaikan teknis, transparansi mekanisme banding akun, dan peningkatan fungsi dasar. Adapun sentimen Netral yang memunculkan frasa “kasih bintang coba”, “fitur pesan langsung”, dan “tahap uji coba” lebih bersifat deskriptif dan memberi masukan fitur, yang dapat dijadikan acuan untuk meningkatkan pengalaman awal pengguna [30].

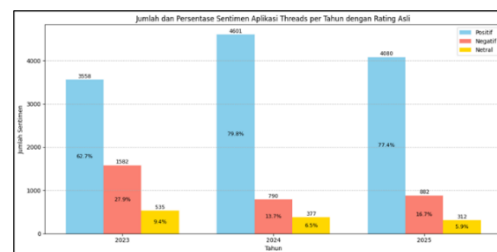
Secara keseluruhan, temuan ini menunjukkan bahwa pengembang *Threads* perlu menyeimbangkan fokus antara menjaga kepuasan, memperbaiki kelemahan teknis, serta merespons kebutuhan fitur baru.

4.7.2 Visualisasi Persebaran Sentimen

Setelah menganalisis kata-kata pada tiap sentimen, tahap berikutnya adalah melihat perkembangan sentimen aplikasi *Threads* dari 2023 hingga 2025.



Gambar 17. Sentimen dengan label IndoBERT



Gambar 18. Sentimen dengan Rating Asli

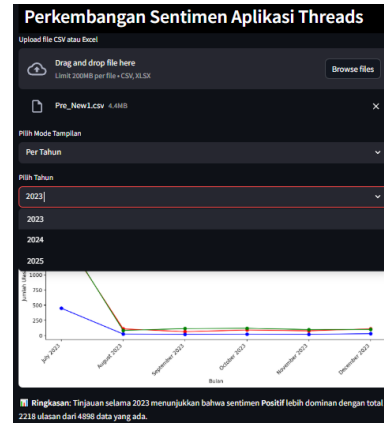
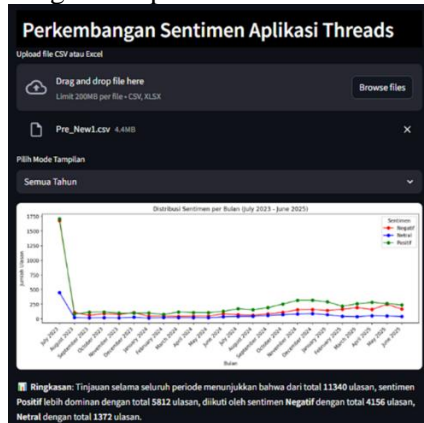
Kedua gambar tersebut menampilkan visualisasi sentimen pertahun menggunakan *IndoBERT* dan *rating* asli secara berdampingan. Hasil *IndoBERT* menunjukkan bahwa pada 2023 Positif

dan Negatif hampir seimbang (45,3% dan 43,3%), Netral 11,4%, mencerminkan pro dan kontra awal peluncuran. Tahun 2024 Positif meningkat menjadi 58,1% dan Negatif menurun menjadi 27,2%, sementara paruh 2025 Positif turun menjadi 53,0% dan Negatif naik ke 36,9%. Sebaliknya, rating asli menunjukkan dominasi Positif sejak 2023 hingga 2025, dengan Negatif dan Netral tetap rendah.

Perbedaan ini menunjukkan bahwa *rating* saja tidak cukup merepresentasikan persepsi pengguna, sedangkan analisis berbasis *NLP* seperti *IndoBERT* mampu menangkap nuansa kepuasan dan ketidakpuasan pengguna secara lebih detail.

4.7.3 Prediksi Menggunakan Model Terbaik

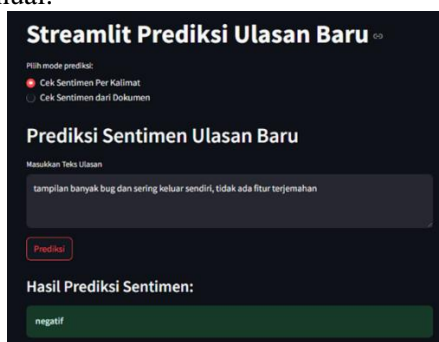
Penelitian ini menghadirkan dashboard interaktif berbasis *Streamlit* untuk menampilkan perkembangan dan prediksi sentimen ulasan aplikasi *Threads*.



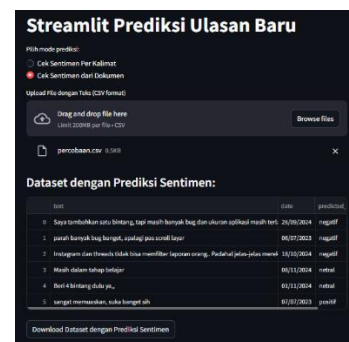
Gambar 19. *Dashboard* Perkembangan Sentimen

Gambar 20. Perkembangan Per-Tahun

Dashboard ini memungkinkan visualisasi distribusi sentimen Positif, Negatif, dan Netral dari Juli 2023 hingga Juni 2025. Pengguna dapat memilih cakupan periode analisis, baik “Semua Tahun” maupun “Per Tahun”, sehingga grafik garis yang ditampilkan dapat mencerminkan tren sentimen secara keseluruhan atau spesifik tiap tahun. Ringkasan otomatis di bawah grafik menunjukkan jumlah ulasan tiap kategori sentimen, memudahkan pemahaman tanpa perhitungan manual.



Gambar 21. *Dashboard* Prediksi Ulasan



Gambar 22. *Dashboard* Prediksi Ulasan Dokumen

Selain itu, *dashboard* menyediakan fitur prediksi ulasan menggunakan model *Random Forest*. Pada mode prediksi per kalimat, pengguna dapat mengetik ulasan baru dan sistem langsung menampilkan hasil prediksi. Sedangkan pada mode prediksi dari dokumen, pengguna dapat mengunggah file *CSV* berisi ulasan, dan sistem akan memproses seluruh dataset, menampilkan tabel hasil prediksi, serta memungkinkan pengunduhan dataset lengkap dengan prediksi sentimennya. Dengan demikian, *dashboard* ini mempermudah analisis tren sentimen serta evaluasi ulasan baru maupun dataset skala besar secara cepat dan interaktif.

5. KESIMPULAN

Hasil penelitian menunjukkan bahwa sentimen positif mendominasi ulasan pengguna aplikasi *Threads*, sementara sentimen negatif muncul terkait *bug* dan performa, serta sentimen netral lebih bersifat informatif. Dari sisi performa model, *Random Forest* menjadi algoritma terbaik (akurasi 81,8%; *F1-score* 76,6%), dengan waktu pelatihan lebih lama karena banyaknya pohon keputusan, tetapi prediksinya tetap cepat. Diikuti oleh *SVM* (81,4%; 75,9%), *Naïve Bayes* (79,1%; 72,4%), dan *KNN* (61,2%; 57,6%)., dimana pelatihannya cepat, tetapi prediksinya lambat karena harus menghitung jarak ke seluruh data latih. Implementasi *dashboard* berbasis *Streamlit* berfungsi untuk menampilkan perkembangan sekaligus memprediksi sentimen pengguna terhadap aplikasi *Threads*.

6. DAFTAR PUSTAKA

- [1] T. Ongun, "Pengguna *Threads* capai 100 juta dalam 5 hari setelah diluncurkan," Anadolu Agency, Jul. 11, 2023. [Online]. Available: <https://www.aa.com.tr/id/dunia/pengguna-Threads-capai-100-juta-dalam-5-hari-setelah-diluncurkan/2942356>. [Accessed: Apr. 25, 2025].
- [2] Mashable Indonesia, "Threads raih 20 juta pengguna bulanan baru, total mencapai 320 juta," Mashable Indonesia, Feb. 26, 2025. [Online]. Available: <https://id.mashable.com/tech/6632/Threads-raih-20-juta-pengguna-bulanan-baru-total-mencapai-320-juta>. [Accessed: Feb. 5, 2025].
- [3] Influencer Marketing Hub, "Comparing X (Twitter) and *Threads* to understand the difference," Nov. 29, 2024. [Online]. Available: <https://influencermarketinghub.com/x-twitter-Threads/>. [Accessed: May 16, 2025].
- [4] R. Montti, "Threads: What is the new social app from Instagram?" Search Engine Journal, Oct. 8, 2024. [Online]. Available: <https://www.searchenginejournal.com/Threads/493069/>. [Accessed: Aug. 8, 2025].
- [5] E. D. Madyatmadja, H. Candra, J. Nathaniel, and M. R. Jonathan, "Sentiment Analysis on User Reviews of *Threads* Applications in Indonesia," Journal Européen des Systèmes Automatisés, vol. 57, no. 4, p. 1165, 2024. [Online]. Available: <https://www.proquest.com/openview/7ca13f2eb65a75ef65a2ffca2c340b76/1?cb1=2069456&pq-origsite=gscholar>. [Accessed: Apr. 25, 2025].
- [6] A. Nadira, N. Y. Setiawan, and W. Purnomo, "Analisis Sentimen Pada Ulasan Aplikasi Mobile Banking Menggunakan Metode Naïve Bayes Dengan Kamus Inset," Indexia, vol. 5, no. 1, pp. 35–47, 2023. [Online]. Available: —. [Accessed: July. 5, 2025].
- [7] M. A. Java, M. Syafrullah, W. Windarto, and P. Painem, "Analisis Sentimen Ulasan Pengguna Aplikasi *Threads* pada Google Play Store Menggunakan Multinomial Naïve Bayes dan Support Vector Machine," Jurnal Ticom: Technology of Information and Communication, vol. 12, no. 2, pp. 75–80, 2024. [Online]. Available: <https://jurnal-ticom.jakarta.aptikom.org/index.php/Ticom/article/view/112/91>. [Accessed: Apr. 25, 2025].
- [8] N. Wulandari, Y. Cahyana, R. Rahmat, and H. Hikmayanti, "Sentiment Analysis on the Relocation of the National Capital (IKN) on Social Media X Using Naive Bayes and K-Nearest Neighbor (KNN) Methods," Journal of Applied Informatics and Computing, vol. 9, no. 3, pp. 724–731, 2025. [Online]. Available: <https://jurnal.polibatam.ac.id/index.php/JAIC/article/view/9552/2768>. [Accessed: Jul. 9, 2025].
- [9] J. Muliawan and E. Dazki, "Sentiment Analysis of Indonesia's Capital City Relocation using Three Algorithms: Naïve Bayes, KNN, and Random Forest," Jurnal Teknik Informatika (JUTIF), vol. 4, no. 5, pp. 1227–1236, 2023. [Online]. Available: <https://pdfs.semanticscholar.org/71d2/6332c2d30b7cb91a356e7cda9e1327e0590d.pdf>. [Accessed: Jul. 5, 2025].
- [10] R. Khairunnas, J. A. Pagua, G. Fitriya, and Y. Ruldeviyani, "User sentiment dynamics in social media: a comparative analysis of X and *Threads*," IAES International Journal of Artificial Intelligence, vol. 14, no. 1, pp. 447–456, 2025. [Online]. Available: —.

-
- https://www.researchgate.net/publication/388598598_User_sentiment_dynamics_in_social_media_a_comparative_analysis_of_X_and_Threads. [Accessed: May 1, 2025].
- [11] F. Nufairi, N. Pratiwi, and F. Herlando, "Analisis Sentimen Pada Ulasan Aplikasi Threads Di Google Playstore Menggunakan Algoritma Support Vector Machine," *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 9, no. 1, pp. 339–348, 2024. [Online]. Available: <https://jurnal.stkipgritulungagung.ac.id/index.php/jipi/article/view/4929>. [Accessed: Feb. 5, 2025].
- [12] M. P. Pulungan, A. Purnomo, and A. Kurniasih, "Penerapan SMOTE untuk Mengatasi Imbalance Class dalam Klasifikasi Kepribadian MBTI Menggunakan Naïve Bayes Classifier," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 11, no. 5, pp. 1033–1042, 2024. [Online]. Available: https://www.researchgate.net/publication/376959015_Penerapan_SMOTE_untuk_Mengatasi_Imbalance_Class_dalam_Klasifikasi_Kepribadian_MBTI_Menggunakan_Naive_Bayes_Classifier. [Accessed: Feb. 11, 2025].
- [13] ProjectPro, "Probabilistic Models in Machine Learning," ProjectPro, 2024. [Online]. Available: <https://www.projectpro.io/article/probabilistic-models-in-machine-learning/784>. [Accessed: Apr. 25, 2025].
- [14] F. T. Berton, D. E. Ratnawati, and M. A. Rahman, "Perbandingan Naïve Bayes Dan K-Nearest Neighbor Untuk Analisis Sentimen Terhadap Ulasan Aplikasi Threads," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 8, no. 9, 2023. [Online]. Available: <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/14109/6301>. [Accessed: Apr. 25, 2025].
- [15] A. S. Nabila, "Perbandingan Model Machine Learning Dan Deep Learning Terhadap Analisis Sentimen Pelanggan Shopee," Doctoral dissertation, UIN Ar-Raniry Banda Aceh, 2024. [Online]. Available: <https://repository.ar-raniry.ac.id/eprint/37469/1/Alissa%20Sausan%20Nabila,%20190705057,%20FST,%20TI.pdf>. [Accessed: Jul. 5, 2025].
- [16] BINUS School of Information Systems, "Description and application of Linear and non-Linear in data mining," BINUS School of Information Systems, Jan. 28, 2022. [Online]. Available: <https://sis.binus.ac.id/2022/01/28/description-and-application-of-linear-and-non-linear-in-data-mining-2/>. [Accessed: May 1, 2025].
- [17] Kirim.ai, "Model Linear: Regresi Linear, Regresi Logistik, dan SVM," Kirim.ai, 2025. [Online]. Available: <https://hub.kirim.ai/model-linear-regresi-linear-logistik-SVM/>. [Accessed: Aug. 13, 2025].
- [18] A. H. Iman, F. R. Permana, G. P. Wardana, R. K. Rachmansyah, and M. M. Santoni, "Perbandingan Algoritma Klasifikasi Random Forest dan Extreme Gradient Boosting pada Dataset Cuaca Provinsi DKI Jakarta Tahun 2018," in *Prosiding Seminar Nasional Mahasiswa Bidang Ilmu Komputer dan Aplikasinya*, vol. 3, no. 2, pp. 793–803, Aug. 2022. [Online]. Available: <https://conference.upnvj.ac.id/index.php/senamika/article/view/2218/1691>. [Accessed: Aug. 13, 2025].
- [19] GeeksforGeeks, "Random Forest Algorithm in Machine Learning," GeeksforGeeks, Jan. 16, 2025. [Online]. Available: <https://www.geeksforgeeks.org/machine-learning/random-forest-algorithm-in-machine-learning/>. [Accessed: May 1, 2025].
- [20] GeeksforGeeks, "Lemmatization vs Stemming: A deep dive into NLP's text Normalization techniques," GeeksforGeeks, n.d. [Online]. Available: <https://www.geeksforgeeks.org/lemmatization-vs-stemming-a-deep-dive-into-NLPs-text-Normalization-techniques/#what-is-lemmatization>. [Accessed: Apr. 30, 2025].
- [21] GeeksforGeeks, "What is data Cleaning?" GeeksforGeeks, n.d. [Online]. Available: <https://www.geeksforgeeks.org/what-is-data-Cleaning/#2-removing-duplicates>. [Accessed: Apr. 30, 2025].
- [22] T. M. Mitchell, *Machine Learning*. McGraw-Hill, 1997. [Online]. Available: https://www.cs.swarthmore.edu/~meeden/cs63/s16/reading/Mitchell_ch8.pdf. [Accessed: Aug. 13, 2025].
-

-
- [23] M. D. Maulidan, S. Sumarlinda, and S. Sopingi, "Development of Sentimen Analysis System of Simple Pol Application on Google Playstore Using Naïve Bayes Classifier Method and BERT Prediction," *Journal of Dinda: Data Science, Information Technology, and Data Analytics*, vol. 4, no. 2, pp. 115–122, 2024. [Online]. Available: <https://journal.itelkom-pwt.ac.id/index.php/dinda/article/view/1577/507>. [Accessed: Feb. 5, 2025].
- [24] M. E. Purbaya, D. P. Rakhmadani, M. P. Arum, and L. Z. Nasifah, "Implementation of N-Gram Methodology to Analyze Sentiment Reviews for Indonesian Chips Purchases in Shopee E-Marketplace," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 7, no. 3, pp. 609–617, 2023. [Online]. Available: https://www.researchgate.net/publication/371281311_Implementation_of_N-Gram_Methodology_to_Analyze_Sentiment_Reviews_for_Indonesian_Chips_Purchases_in_Shopee_E-Marketplace. [Accessed: Aug. 19, 2025].
- [25] H. Basri, M. B. S. Junianto, and I. Kusyadi, "Enhancing usability testing through sentiment analysis: A comparative study using SVM, Naïve Bayes, Decision Trees and Random Forest," *Jurnal Teknik Informatika dan Sistem Informasi, Univ. Pamulang*, 2024. [Online]. Available: <https://openjournal.unpam.ac.id/index.php/JTSI/article/view/45117>. [Accessed: Aug. 16, 2025].
- [26] G. K. Putri, H. Sujaini, and D. I. Ulumi, "Perbandingan Algoritma Support Vector Machine (SVM) dan Naive Bayes pada Analisis Sentimen Bahasa Jawa dan Sunda," *JUSTIN (Jurnal Sistem dan Teknologi Informasi)*, vol. 13, no. 2, pp. 299–306, 2025. [Online]. [Accessed: Aug. 16, 2025].
- [27] A. N. K. Kambayo, S. S. Berutu, J. Jatmika, and A. Nshimiyimana, "Aspect-Based Sentiment Analysis Using Latent Dirichlet Allocation (LDA) and DistilBERT on Threads App Reviews," *Infact: International Journal of Computers*, vol. 9, no. 1, pp. 25–34, 2025. [Online]. Available: <https://journal.ukrim.ac.id/index.php/JIF/article/view/707/527>. [Accessed: Aug. 20, 2025].
- [28] M. F. Hanif, S. H. Wijoyo, and W. H. N. Putra, "Klasifikasi Sentimen Ulasan Aplikasi Threads Berbasis Algoritma Naive Bayes dan Metode Root Caus Analysis," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 8, no. 6, 2024. [Online]. Available: <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/13786/6131>. [Accessed: Aug. 20, 2025].
- [29] I. A. I. Pidada and N. P. N. C. Dewi, "The Effect of User Interface, User Experience, and Perceived Ease of Use on Interest in Using the SOCO by Sociolla Application with Perceived Usefulness as a Moderator," *Golden Ratio of Marketing and Applied Psychology of Business*, vol. 5, no. 1, pp. 234–245, 2025. [Online]. [Accessed: Aug. 20, 2025].
- [30] N. M. Damayanti, "Analisis Sentimen Publik Pada Tagar #Btscomeback Di Platform X Menggunakan IndoBERTweet," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 13, no. 3, 2025. [Online]. [Accessed: Aug. 20, 2025].
-