Jurnal Informatika dan Tekonologi Komputer

E-ISSN: 2809-9230
P-ISSN: 2809-9249

Analisis Sentimen Ulasan Pengguna Aplikasi Threads di Google Play Menggunakan Algoritma XGBoost Dengan Penguatan SMOTE

Bayu Nur Setiyono[1*], Nadia Annisa Maori[2], dan Teguh Tamrin [3]

[1] Universitas Islam Nahdlatul Ulama Jepara, Jawa Tengah; e-mail : bayunursetiyono@gmail.com
[2] Universitas Islam Nahdlatul Ulama Jepara, Jawa Tengah; e-mail : nadia@unisnu.ac.id
[3] Universitas Islam Nahdlatul Ulama Jepara, Jawa Tengah; e-mail : Teguh@unisnu.ac.id
* Corresponding Author : Bayu Nur Setiyono

Abstract: Threads is a text-based social media application developed by Meta that has gained significant popularity since its launch. However, user reviews on the Google Play Store reveal an imbalanced sentiment distribution, with a dominance of positive sentiment, potentially reducing the accuracy of sentiment classification models. This study aims to evaluate the effectiveness of combining the Extreme Gradient Boosting (XGBoost) algorithm with the Synthetic Minority Over-sampling Technique (SMOTE) to address the data imbalance in user reviews of the Threads application. The dataset consists of 1,000 user reviews, which underwent preprocessing steps including case folding, cleaning, tokenization, stopword removal, and stemming. The data were then represented using the TF-IDF weighting method and analyzed using XGBoost, both before and after applying SMOTE. Results show that without SMOTE, the model achieved an accuracy of 87.60%, with a low recall for the negative class (0.69). After applying SMOTE, accuracy improved to 97.49%, and recall for the negative class reached 0.99, with balanced F1-scores for both positive and negative classes (0.98 and 0.97, respectively). These findings demonstrate that SMOTE is effective in handling class imbalance and enhancing model performance. In conclusion, the integration of XGBoost and SMOTE significantly improves fairness and accuracy in sentiment classification of app reviews, offering valuable insights for the application of machine learning in user opinion analysis. Future research is recommended to use larger datasets and consider deep learning models such as BERT.

Keywords: Sentiment Analysis; SMOTE; XGBoost

Received: May 23, 2025
Revised: May 30, 2025
Accepted: July 6, 2025
Published: July 10, 2025
Curr. Ver.: July 10, 2025

Copyright: © 2025 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY SA) license (https://creativecommons.org/licenses/by-sa/4.0/)

Abstrak: Threads merupakan aplikasi media sosial berbasis teks yang dikembangkan oleh Meta dan memperoleh popularitas tinggi sejak peluncurannya. Namun, ulasan pengguna terhadap aplikasi ini di Google Play Store menunjukkan distribusi sentimen yang tidak seimbang, dengan dominasi sentimen positif, yang berpotensi menurunkan akurasi model klasifikasi sentimen. Penelitian ini bertujuan untuk mengevaluasi efektivitas kombinasi algoritma Extreme Gradient Boosting (XGBoost) dan teknik Synthetic Minority Over-sampling Technique (SMOTE) dalam mengatasi ketidakseimbangan data ulasan aplikasi Threads. Dataset terdiri dari 1.000 ulasan pengguna, yang melalui tahapan preprocessing seperti case folding, cleaning, tokenization, stopword removal, dan stemming. Data kemudian direpresentasikan menggunakan metode pembobotan TF-IDF dan dianalisis menggunakan XGBoost, baik sebelum maupun sesudah penerapan SMOTE. Hasil penelitian menunjukkan bahwa tanpa SMOTE, model menghasilkan akurasi 87,60%, dengan recall rendah pada kelas negatif (0,69). Setelah penerapan SMOTE, akurasi meningkat menjadi 97,49% dan recall untuk kelas negatif mencapai 0,99, dengan F1-score yang seimbang antara kelas positif dan negatif (0,98 dan 0,97). Temuan ini membuktikan bahwa SMOTE efektif dalam menangani ketidakseimbangan kelas dan meningkatkan performa model klasifikasi. Kesimpulannya, integrasi XGBoost dengan SMOTE mampu meningkatkan keadilan dan akurasi dalam klasifikasi sentimen ulasan aplikasi, serta memberikan kontribusi penting dalam

penerapan machine learning untuk analisis opini pengguna. Penelitian lanjutan disarankan untuk menggunakan dataset yang lebih besar dan mempertimbangkan model berbasis deep learning seperti BERT.

Kata kunci: Sentimen Analisis; SMOTE; XGBoost

1. Pendahuluan

Di era kemajuan teknologi, transformasi digital telah masuk ke dalam kehidupan sehari-hari kita tanpa kita ketahui. Teknologi telah berkembang pesat dan kini melekat dalam berbagai aspek kehidupan manusia. Salah satu peran pentingnya adalah mendukung efisiensi pekerjaan dan menyederhanakan beragam aktivitas manusia.[1]. Masyarakat secara luas menggunakan platform media sosial sebagai sarana untuk membagikan pendapat, mengungkapkan emosi, menceritakan pengalaman pribadi, dan berbagai keperluan lainnya.[2]. Threads menjadi salah satu contoh aplikasi media sosial yang memungkinkan penggunanya berhubungan dengan orang-orang dari seluruh dunia dengan mudah. Pengguna dapat mengunggah dan membagikan konten dalam bentuk teks, gambar, dan video melalui fitur aplikasi ini. Mereka juga dapat berinteraksi dengan unggahan pengguna lainnya melalui komentar, membagikan ulang, dan menyukai postingan. Threads dapat diunduh di Google Play Store untuk perangkat berbasis Android dan App Store untuk perangkat Ios.[3]. Threads merupakan aplikasi media sosial berbasis teks yang dikembangkan oleh Meta dan berhasil mencatatkan lebih dari 100 juta pengguna hanya dalam waktu kurang dari seminggu sejak peluncurannya. Aplikasi ini memungkinkan pengguna berbagi konten teks, gambar, dan video serta berinteraksi melalui fitur komentar, repost, dan like.[4] Namun, seiring meningkatnya jumlah pengguna, ulasan yang diberikan di Google Play Store menjadi semakin banyak dan beragam, mencerminkan persepsi positif maupun negatif terhadap performa dan fitur aplikasi tersebut.[5]

Penelitian oleh [6] menganalisis sentimen ulasan aplikasi Dana di Google Play Store menggunakan algoritma *Naïve Bayes* dan pendekatan *KDD*. Data dikumpulkan melalui *web scraping* dan dianalisis dengan RapidMiner. Hasilnya menunjukkan akurasi sebesar 50,21%, dengan precision dan recall 0%, yang mengindikasikan performa model kurang optimal. Sementara itu, penelitian oleh [7] menggunakan algoritma *Naïve Bayes* untuk mengklasifikasikan sentimen pada data Twitter terkait BMKG. Dengan pendekatan *text mining*, model berhasil mencapai akurasi sebesar 69,97%, menunjukkan performa yang lebih baik dalam klasifikasi tiga kelas sentimen: positif, negatif, dan netral. Hasil kedua penelitian tersebut menunjukkan bahwa Naïve Bayes kurang efektif dalam menangani data teks yang kompleks sehingga Akurasi rendah. Sebagai solusi, digunakan algoritma yang lebih kuat dan adaptif, yaitu *Extreme Gradient Boosting* (XGBoost), yang terbukti memiliki performa klasifikasi yang lebih baik. XGBoost adalah kumpulan pohon keputusan yang terdiri dari CART, yaitu berbagai pohon sebelumnya.[8]

Penelitian oleh Habib Hakim [9] membandingkan algoritma Decision Tree dan XGBoost untuk klasifikasi sentimen vaksin Covid-19 di Twitter. Dengan 13.115 tweet berbahasa Indonesia yang telah melalui proses balancing dan preprocessing, XGBoost menunjukkan akurasi tertinggi sebesar 66%. Hasil ini menegaskan keunggulan XGBoost, meskipun disarankan penambahan data minoritas untuk hasil yang lebih optimal. Dalam konteks serupa, penelitian oleh [10]. menerapkan teknik SMOTE pada algoritma XGBoost untuk analisis sentimen aplikasi Muamalat DIN dan berhasil meningkatkan akurasi menjadi 78% dengan F1-Score 0.77. Temuan ini menegaskan efektivitas SMOTE dalam mengatasi ketidakseimbangan data dan meningkatkan performa klasifikasi.

Ulasan aplikasi threads mengandung berbagai sentimen, mulai dari apresiasi hingga kritikan, yang bila dianalisis secara mendalam dapat menyediakan wawasan berharga bagi tim pengembang aplikasi dan pihak-pihak yang berkepentingan. Permasalahan ada ketika terdapat ketidakseimbangan dalam distribusi sentimen pada ulasan pengguna, dengan 676 ulasan positif yang jauh melebihi 324 ulasan negatif. Kondisi ini mengakibatkan penurunan efektivitas model klasifikasi

Tujuan penelitian ini adalah untuk mengukur efektivitas *algoritma Extreme Gradient Boosting* (XGBoost) yang dipadukan dengan metode *Synthetic Minority Over-sampling Technique* (SMOTE) dalam mengatasi masalah ketidakseimbangan data yang berkaitan dengan analisis sentimen pengguna untuk aplikasi Threads di Google Play. Secara spesifik, penelitian ini ditujukan untuk mengembangkan model klasifikasi sentimen dengan tingkat akurasi optimal.

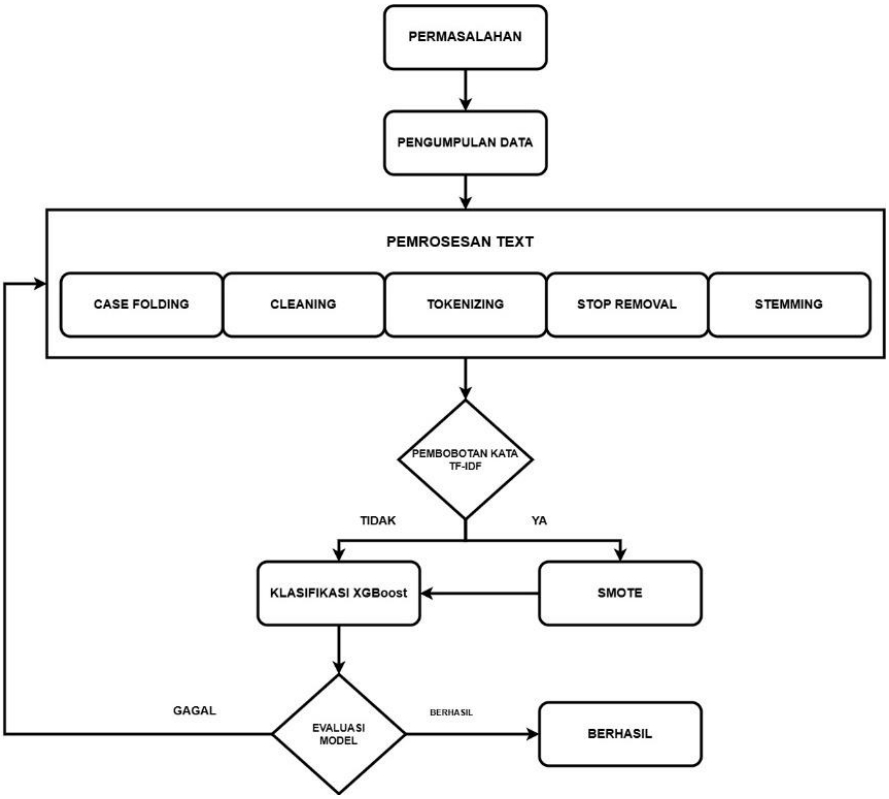
Dalam era digital yang terus berkembang, pemahaman yang akurat terhadap opini pengguna menjadi sangat penting bagi pengembang aplikasi dan perusahaan teknologi untuk meningkatkan kualitas layanan dan pengalaman pengguna. Meskipun berbagai metode klasifikasi sentimen telah dikembangkan, tantangan utama tetap terletak pada ketidakseimbangan data yang dapat menurunkan performa model, terutama ketika ulasan positif mendominasi secara signifikan. Penelitian ini menghadirkan inovasi dengan menggabungkan algoritma XGBoost yang terkenal dengan kecepatan dan akurasinya, bersama teknik SMOTE yang efektif dalam menangani ketidakseimbangan kelas. Kebaruan penelitian ini terletak pada penerapan gabungan kedua metode tersebut dalam konteks analisis sentimen ulasan aplikasi Threads, sebuah aplikasi baru yang tengah naik daun dan memiliki karakteristik data yang unik. Hal ini memberikan kontribusi signifikan dalam pengembangan model klasifikasi yang lebih adil dan akurat, serta menawarkan solusi praktis yang dapat diadaptasi pada analisis opini pengguna aplikasi lain yang menghadapi tantangan serupa. Dengan demikian, studi ini tidak hanya memperkaya literatur di bidang machine learning dan sentiment analysis, tetapi juga memberikan panduan strategis bagi pengembang aplikasi dan pemangku kepentingan dalam mengelola dan memahami opini pengguna secara lebih efektif dan objektif.

23. Metode yang Diusulkan

Pada penelitian ini, peneliti melakukan klasifikasi sentimen terhadap ulasan pengguna aplikasi Threads yang tersedia di Google Play Store dengan menggunakan algoritma Extreme Gradient Boosting (XGBoost). Untuk mengatasi permasalahan ketidakseimbangan kelas pada data sentimen, penelitian ini juga mengimplementasikan teknik Synthetic Minority Over-sampling Technique (SMOTE) untuk meningkatkan akurasi dan keseimbangan performa model klasifikasi dalam mengidentifikasi komentar positif dan negatif dari pengguna aplikasi Threads, serta memberikan wawasan yang berguna bagi pengembang aplikasi dalam memahami persepsi dan pengalaman pengguna.

2.1 Tahapan Metode Penelitian

Penelitian ini terdiri dari beberapa tahapan yang sistematis untuk membangun model klasifikasi sentimen, dapat dilihat pada gambar 1 berikut



Gambar 1. Alur Penelitian

Pada gambar 1 memperlihatkan tahapan penelitian yang diawali dengan pengumpulan data ulasan dari platform Kaggle. Selanjutnya, data melalui proses pra-pemrosesan yang mencakup pembersihan teks, tokenisasi, penghapusan stopword, dan stemming. Pada tahap pelatihan, teknik SMOTE digunakan untuk menyeimbangkan data training sebelum diaplikasikan algoritma XGBoost guna mengatasi ketidakseimbangan kelas sentimen. Setelah model dilatih, dilakukan evaluasi performa menggunakan metrik akurasi, precision, recall, dan F1-Score. Penelitian diakhiri dengan analisis dan kesimpulan berdasarkan perbandingan hasil klasifikasi dengan dan tanpa penerapan SMOTE.

2.2 Pengumpulan Data

Dataset yang digunakan berisi 1.000 ulasan dengan 6 kolom, diambil dari Kaggle yang berisi review pengguna Threads di Google Play Store Indonesia. Data ini menjadi dasar untuk proses pra-pemrosesan dan pelatihan model.

2.3 Pemrosesan Text

Penelitian ini menggunakan *Google Colaboratory* (Google Colab) sebagai lingkungan pengembangan dan eksekusi kode. Google Colab, yang merupakan kependekan dari Google Colaboratory, platform berbasis cloud milik Google yang memungkinkan pengguna menjalankan dan menulis kode Python menggunakan browser tanpa konfigurasi tambahan[11]. Google Colab sangat cocok digunakan dalam penelitian berbasis *machine learning* karena menyediakan akses ke pustaka populer seperti Scikit-learn, Pandas, NumPy, dan Matplotlib tanpa

perlu konfigurasi tambahan.[12] Setelah lingkungan pengembangan siap, langkah selanjutnya dalam penelitian ini adalah tahap *preprocessing.*

Preprocessing adalah tahap di mana data dibersihkan dan diperbaiki. Data yang dikumpulkan biasanya terdiri dari karakter yang tidak terstruktur dan tidak penting. Preprocessing dilakukan untuk menghilangkan suara atau gangguan sebelumnya.[13]. Proses preprocessing meliputi case folding, cleaning, removal of stopwords, stemming, dan tokenization.

1) Case Folding : Penulis melakukan transformasi seluruh kata dan kalimat ke dalam bentuk huruf kecil pada tahap Case Folding.contoh kata "Treads" diubah menjadi "treads".[14]
2) Cleaning : Yaitu melakukan pembersihan data dari berbagai bentuk noise seperti hashtag, username, tautan URL, dan tanda baca. [15]
3) Tokenizing : Tahapan ini berfungsi untuk mentransformasi kata ke dalam bentuk dasarnya. Kata-kata dengan imbuhan akan dibersihkan menjadi bentuk standarnya selama proses ini. Contoh [aplikasi treads] diubah menjadi ['aplikasi', 'treads'].[16]
4) Stop Removal : Menghapus kata-kata yang tidak penting atau tidak signifikan dari struktur kalimat. Contoh ['aplikasi','untuk','berbagi'] menjadi ['aplikasi','berbagi']. [17]
5) Stemming : Tahapan ini berfungsi untuk mengembalikan kata ke bentuk aslinya. Kata-kata berimbuhan akan dipisahkan menjadi bentuk standarnya selama proses ini. Contoh "membosankan" menjadi "bosan". [14]

2.4 Pembobotan Kata

Metodologi pemrosesan bahasa alami adalah *Term Frequency-Inverse Document Frequency* (TF-IDF) mempertimbangkan *Term Frequency* (TF) dan *Inverse Document Frequency* (IDF) adalah dua komponen utama. Kata yang muncul dalam dokumen tertentu diukur oleh TF dengan membandingkan jumlah kata yang muncul dalam dokumen lain. Sedangkan *Inverse Document Frequency* mengkuantifikasi tingkat keunikan atau kelangkaan sebuah istilah dalam seluruh kumpulan dokumen, dengan membandingkan logaritma total dokumen keseluruhan terhadap jumlah dokumen yang memuat kata-kata tersebut tersebut.[18]

2.5 *Synthetic Minority Over-sampling Technique* (SMOTE)

SMOTE (*Synthetic Minority Over-Sampling Technique*) adalah metode yang digunakan untuk memperbaiki ketidakseimbangan data, di mana kuantitas data dalam satu kelompok sangat melebihi kelompok lainnya. Ketidakseimbangan tersebut dapat mengakibatkan model yang dikembangkan menjadi bias terhadap kelompok data mayoritas, sehingga kemampuan prediksinya terhadap kelompok minoritas menjadi kurang optimal.[19]. . SMOTE menghasilkan data sintetis baru untuk populasi yang tidak terwakili sebelumnya, bukan hanya melakukan duplikasi terhadap data yang sudah ada.[20]. Adapun rumus teknik SMOTE dapat dilihat pada persamaan (1) berikut:

$$X_{syn} = x_i + (x_{knn} - x_i) x \delta \qquad (1)$$

Dimana:

$Xsyn$:Data Sintesis
Xi :Data yang akan disimpan
$Xknn$:Data yang memiliki jarak terdekat dengan data Xi
δ :Nilai random antara 0 dan 1
[21]

2.6 *Extreme Gradient Boosting* (XGBoost)

XGBoost merupakan evolusi dari algoritma gradient tree boosting yang berbasis ensemble, dan mampu mengelola kasus machine learning dalam skala besar secara efisien. Metode ini dipilih karena dilengkapi dengan fitur tambahan untuk mempercepat perhitungan dan mencegah overfitting.[8]. Pendekatan XGBoost sangat sesuai untuk mengatasi dataset yang tidak seimbang, di mana kelas minoritas sering kali kurang memiliki representasi yang memadai. Di samping keunggulannya dalam efisiensi komputasi, kapabilitas optimasi yang dimiliki XGBoost memungkinkan pemrosesan dataset berukuran besar dengan kecepatan tinggi.[22]. Model *Extreme Gradient Boosting* (XGBoost) dapat direpresentasikan pada Persamaan (2) berikut ini :

$$y_i = \sum_{k=1}^{k} f_{k\ (xi),}\ f_{k \in F} \qquad (2)$$

Keterangan:
K = Jumlah pohon keputusan yang ada
$fk(xi)$ = Input yang digunakan oleh pohon keputusan ke-k
$\hat{y}\,l$ = Nilai prediksi
F = mengumpulkan semua kemungkinan CART
[23]

2.7 Evaluasi Model

Evaluasi model dilakukan untuk mengukur performa model klasifikasi sentimen terhadap ulasan aplikasi Threads. Model dievaluasi menggunakan *confusion matrix*.

Confusion Matrix menunjukkan jumlah prediksi yang salah dan benar yang dibuat oleh model klasifikasi untuk mengevaluasi kinerja model. Empat komponen utama membentuk *Confusion Matrix*.[22] Melalui *confusion matrix*, Seberapa akurat model dalam memprediksi kelas positif dan negatif dapat dilihat, serta frekuensi kesalahan prediksi yang dilakukan oleh model.[24]

Table 1. *Confusion matrix*

		Kelas Sebenarnya	
		True	*False*
Kelas	*True*	TP	FP
Prediksi	*False*	FN	TN

Keterangan :

True Positive (TP)	:	Data kelas positif yang dianggap positif
True Negative (TN)	:	Data kelas negatif yang ddianggap sebagai negatif.
False Positive (FP)	:	Data kelas negatif yang salah dianggap sebagai positif.
False Negative (FN)	:	Data kelas positif yang salah dianggap sebagai negatif.

Rumus *confusion matrix* untuk menghitung akurasi, presisi, recall, dan f1-score. [25]

$$Accuracy = \frac{total\ prediksi\ benar}{Total\ data} \quad (3)$$

$$Presisi(per\ sentimen) = \frac{TP}{TP + FP} \quad (4)$$

$$Recall(per\ sentimen) = \frac{TP}{TP + FN} \quad (5)$$

$$F1 - score(per\ sentimen) = 2 \times \frac{Presisi\ \times\ Recall}{Presisi + Recall} \quad (6)$$

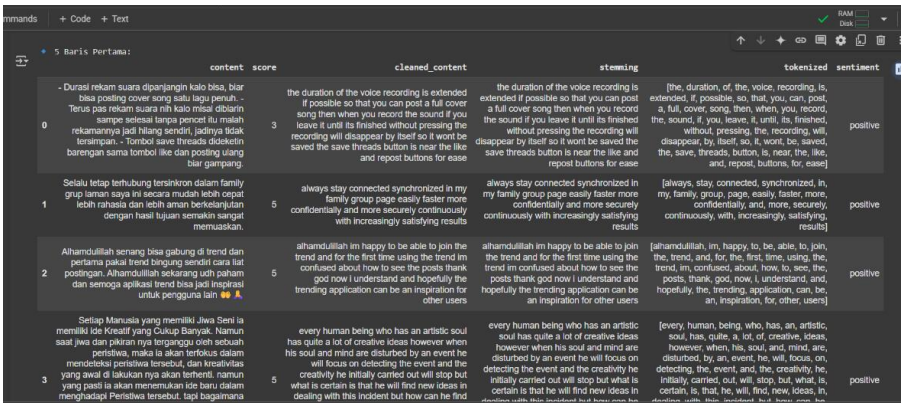
4. Hasil dan Pembahasan

Dalam penelitian ini, sebanyak 1000 data ulasan aplikasi diproses melalui tahap preprocessing teks untuk memastikan data bersih dan siap dianalisis. Proses preprocessing Bisa dilihat pada gambar 2 mencakup beberapa langkah, yaitu case folding untuk mengubah seluruh huruf menjadi huruf kecil; cleaning untuk menghapus karakter tidak relevan seperti tanda baca, angka, tautan, dan simbol; serta tokenization untuk memisahkan teks menjadi kata-kata individu; stop-word removal untuk menghilangkan kata-kata umum yang tidak memiliki makna signifikan seperti "yang", "di", dan "untuk"; dan stemming untuk mengubah kata ke bentuk dasarnya, seperti "menonton" menjadi "tonton";.

```
# ==============================
# 2. 🧹 Preprocessing
# ==============================
text_col = 'translated_content'
label_col = 'sentiment'

def cleaning_text(text):
    text = str(text).lower()
    text = re.sub(r'[^a-zA-Z\s]', '', text)
    text = re.sub(r'\s+', ' ', text).strip()
    return text

df['clean_text'] = df[text_col].apply(cleaning_text)
df['tokens'] = df['clean_text'].apply(lambda x: x.split())

stop_factory = StopWordRemoverFactory()
stopwords = set(stop_factory.get_stop_words())
df['tokens'] = df['tokens'].apply(lambda tokens: [t for t in tokens if t not in stopwords])

stem_factory = StemmerFactory()
stemmer = stem_factory.create_stemmer()
df['stemmed'] = df['tokens'].apply(lambda tokens: ' '.join([stemmer.stem(t) for t in tokens]))

print("\n📌 Contoh setelah preprocessing:")
print(df[[text_col, 'clean_text', 'stemmed']].head())
```

Gambar 2. Pemrosesan text

Setelah preprocessing, teks pada kolom translated_content dibersihkan, dihapus stopwords, dan distemming. Hasilnya ditampilkan dalam kolom baru gambar 3.

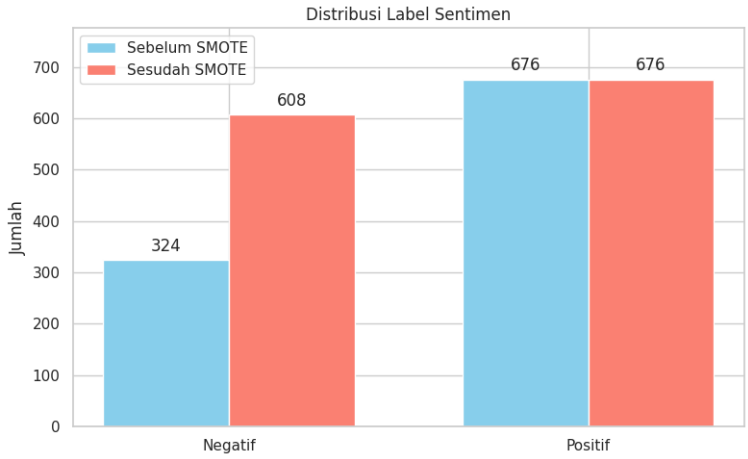


5 Baris Pertama:

	content	score	cleaned_content	stemming	tokenized	sentiment
0	- Durasi rekam suara dipanjangin kalo bisa, biar bisa posting cover song satu lagu penuh. - Terus pas rekam suara nih kalo misal diblarin sampe selesai tanpa pencet itu malah rekamannya jadi hilang sendiri, jadinya tidak tersimpan. - Tombol save threads dideketin barengan sama tombol like dan posting ulang biar gampang.	3	the duration of the voice recording is extended if possible so that you can post a full cover song then when you record the sound if you leave it until its finished without pressing the recording will disappear by itself so it wont be saved the save threads button is near the like and repost buttons for ease	the duration of the voice recording is extended if possible so that you can post a full cover song then when you record the sound if you leave it until its finished without pressing the recording will disappear by itself so it wont be saved the save threads button is near the like and repost buttons for ease	[the, duration, of, the, voice, recording, is, extended, if, possible, so, that, you, can, post, a, full, cover, song, then, when, you, record, the, sound, if, you, leave, it, until, its, finished, without, pressing, the, recording, will, disappear, by, itself, so, it, wont, be, saved, the, save, threads, button, is, near, the, like, and, repost, buttons, for, ease]	positive
1	Selalu tetap terhubung tersinkron dalam family grup laman saya ini secara mudah lebih cepat lebih rahasia dan lebih aman berkelanjutan dengan hasil tujuan semakin sangat memuaskan.	5	always stay connected synchronized in my family group page easily faster more confidentially and more securely continuously with increasingly satisfying results	always stay connected synchronized in my family group page easily faster more confidentially and more securely continuously with increasingly satisfying results	[always, stay, connected, synchronized, in, my, family, group, page, easily, faster, more, confidentially, and, more, securely, continuously, with, increasingly, satisfying, results]	positive
2	Alhamdulillah senang bisa gabung di trend dan pertama pakai trend bingung sendiri cara liat postingan. Alhamdulillah sekarang udh paham dan semoga aplikasi trend bisa jadi inspirasi untuk pengguna lain 🙌🙏	5	alhamdulillah im happy to be able to join the trend and for the first time using the trend im confused about how to see the posts thank god now i understand and hopefully the trending application can be an inspiration for other users	alhamdulillah im happy to be able to join the trend and for the first time using the trend im confused about how to see the posts thank god now i understand and hopefully the trending application can be an inspiration for other users	[alhamdulillah, im, happy, to, be, able, to, join, the, trend, and, for, the, first, time, using, the, trend, im, confused, about, how, to, see, the, posts, thank, god, now, i, understand, and, hopefully, the, trending, application, can, be, an, inspiration, for, other, users]	positive
3	Setiap Manusia yang memiliki Jiwa Seni ia memiliki Ide Kreatif yang Cukup Banyak. Namun saat jiwa dan pikiran nya terganggu oleh sebuah peristiwa, maka ia akan terfokus dalam mendeteksi peristiwa tersebut, dan kreativitas yang awal di lakukan nya akan terhenti. namun yang pasti ia akan menemukan ide baru dalam menghadapi Peristiwa tersebut. tapi bagaimana	5	every human being who has an artistic soul has quite a lot of creative ideas however when his soul and mind are disturbed by an event he will focus on detecting the event and the creativity he initially carried out will stop but what is certain is that he will find new ideas in dealing with this incident but how can he find	every human being who has an artistic soul has quite a lot of creative ideas however when his soul and mind are disturbed by an event he will focus on detecting the event and the creativity he initially carried out will stop but what is certain is that he will find new ideas in	[every, human, being, who, has, an, artistic, soul, has, quite, a, lot, of, creative, ideas, however, when, his, soul, and, mind, are, disturbed, by, an, event, he, will, focus, on, detecting, the, event, and, the, creativity, he, initially, carried, out, will, stop, but, what, is, certain, is, that, he, will, find, new, ideas, in,	positive

Gambar 3. Hasil pemrosesan text

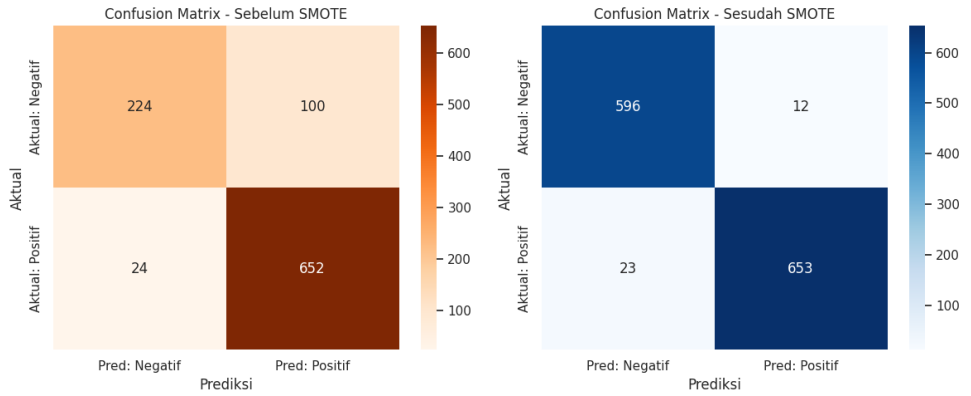
Setelah melalui tahap preprocessing teks dan pelabelan data, dilakukan perhitungan jumlah label untuk menganalisis distribusi sentimen pengguna terhadap aplikasi Threads di Google Play. Penghitungan ini membantu memberikan gambaran mengenai jumlah ulasan dengan sentimen positif dan negatif yang terkandung dalam data.



Gambar 4. Hasil perlabel data asli dan dengan smote

Gambar 4 memperlihatkan distribusi label sentimen sebelum dan sesudah penerapan teknik SMOTE (Synthetic Minority Over-sampling Technique). Sebelum SMOTE, jumlah data dengan sentimen positif adalah 676, sedangkan sentimen negatif hanya 324. Ketimpangan ini menunjukkan bahwa data awal tidak seimbang, yang dapat menyebabkan model cenderung bias terhadap kelas mayoritas (positif). Setelah dilakukan SMOTE, jumlah data pada kelas minoritas (negatif) ditingkatkan menjadi 608, mendekati jumlah data pada kelas mayoritas (676). Meskipun belum benar-benar seimbang, perbedaan ini sudah jauh lebih kecil dan membantu model untuk belajar secara lebih adil terhadap kedua kelas. Strategi ini diambil untuk mengurangi bias dan meningkatkan performa model, khususnya dalam mendeteksi sentimen negatif.

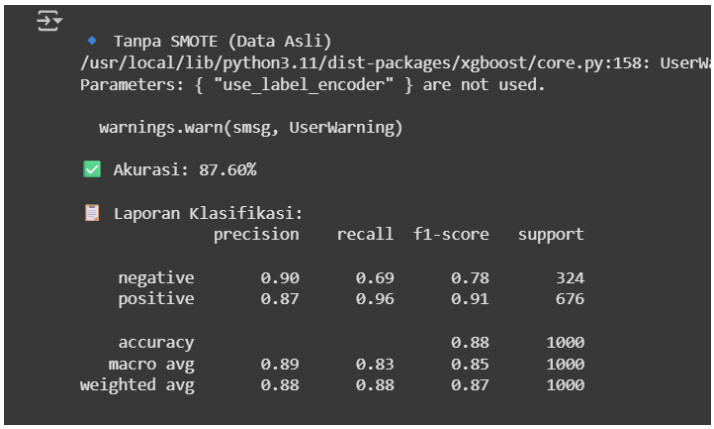
Proses selanjutnya yaitu mengimplementasikan data ke dalam algoritma XGBoost tanpa dan dengan penerapan teknik SMOTE untuk penyeimbangan kelas. Seluruh data digunakan untuk pelatihan dan evaluasi secara langsung. Proses klasifikasi ini menggunakan metode pembobotan kata TF-IDF sebagai representasi fitur teks.



Gambar 5. Confusion Matrix sebelum dan sesudah SMOTE

Gambar 5 pada matrix kiri (sebelum SMOTE), model memprediksi dengan akurasi tinggi pada kelas positif (652 True Positive), namun performa pada kelas negatif masih terbatas (224 True Negative), dan terjadi 100 False Positive serta 24 False Negative. Ini menunjukkan model cenderung bias terhadap kelas mayoritas (positif). Setelah penerapan SMOTE (matrix kanan), performa meningkat secara signifikan dan lebih seimbang: 596 True Negative dan 653 True Positive. False Positive dan False Negative masing-masing turun menjadi 12 dan 23. Ini menunjukkan bahwa SMOTE berhasil membantu model mengenali kedua kelas secara lebih adil.

Langkah selanjutnya dalam penelitian ini adalah melakukan evaluasi terhadap model untuk memastikan bahwa model yang dikembangkan mampu mencapai tujuan dan memenuhi kebutuhan yang telah ditetapkan.



```
🔹 Tanpa SMOTE (Data Asli)
/usr/local/lib/python3.11/dist-packages/xgboost/core.py:158: UserW
Parameters: { "use_label_encoder" } are not used.

  warnings.warn(smsg, UserWarning)

✅ Akurasi: 87.60%

📋 Laporan Klasifikasi:
              precision    recall  f1-score   support

    negative       0.90      0.69      0.78       324
    positive       0.87      0.96      0.91       676

    accuracy                           0.88      1000
   macro avg       0.89      0.83      0.85      1000
weighted avg       0.88      0.88      0.87      1000
```

Gambar 6. Implementasi model xgboost

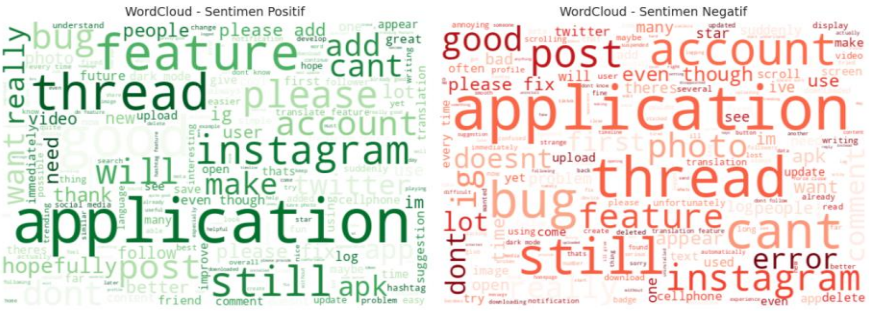
Pada Gambar 6 ditampilkan performa model XGBoost tanpa penerapan SMOTE, dengan akurasi sebesar 87,60%. Meski akurasinya cukup tinggi, performa terhadap kelas negatif masih rendah, terlihat dari nilai recall sebesar 0,69 dan f1-score hanya 0,78. Ini menunjukkan model cukup sering gagal mengenali data sentimen negatif (banyak False Negative). Sementara itu, untuk kelas positif, model menunjukkan performa yang jauh lebih baik dengan recall 0,96 dan f1-score 0,91, mengindikasikan adanya ketidakseimbangan dalam pengenalan kelas.

```
📊 Jumlah label setelah SMOTE Parsial: Counter({1: 676, 0: 608})

🔹 Dengan SMOTE Parsial (Mendekati Seimbang)
/usr/local/lib/python3.11/dist-packages/xgboost/core.py:158: UserWa
Parameters: { "use_label_encoder" } are not used.

  warnings.warn(smsg, UserWarning)

✅ Akurasi: 97.27%

📋 Laporan Klasifikasi:
              precision    recall  f1-score   support

    negative       0.96      0.98      0.97       608
    positive       0.98      0.97      0.97       676

    accuracy                           0.97      1284
   macro avg       0.97      0.97      0.97      1284
weighted avg       0.97      0.97      0.97      1284
```

Gambar 7. Implementasi xgboost dengan smote

Pada Gambar 6 ditampilkan hasil evaluasi model setelah dilakukan penyeimbangan data menggunakan metode SMOTE parsial. Model menunjukkan peningkatan signifikan dengan akurasi mencapai 97,27%. Performa model dalam mengenali kedua kelas menjadi sangat baik, dengan recall sebesar 0,98 untuk kelas negatif dan 0,97 untuk kelas positif. Nilai f1-score yang seimbang antara kelas negatif (0,97) dan positif (0,97), serta macro dan weighted average yang konsisten di angka 0,97, menunjukkan bahwa model telah berhasil mengatasi bias terhadap kelas tertentu dan mampu melakukan klasifikasi dengan sangat baik.

Setelah proses evaluasi model dilakukan, tahap terakhir adalah memahami karakteristik data melalui visualisasi teks. Visualisasi pada Gambar 7 berupa WordCloud menampilkan kata-kata yang paling sering muncul dalam data setelah melalui tahap preprocessing. Kata-kata ini dikelompokkan berdasarkan label sentimen, sehingga memudahkan dalam mengidentifikasi ciri khas dari masing-masing sentimen, baik positif maupun negatif.



Gambar 8. wordcloud kata positif dan negatif

Gambar 8 menunjukkan visualisasi WordCloud untuk sentimen positif (kiri) dan sentimen negatif (kanan). Pada WordCloud sentimen positif, kata-kata yang paling menonjol antara lain "application", "thread", "feature", dan "instagram", yang menunjukkan bahwa pengguna banyak memberikan komentar positif terhadap fitur-fitur tertentu dalam aplikasi. Sementara itu, pada WordCloud sentimen negatif, kata-kata seperti "application", "thread", "error", "bug", dan "cant" muncul lebih dominan, mencerminkan keluhan pengguna terhadap berbagai kendala teknis, kesalahan aplikasi, atau keterbatasan fungsi.

Secara kesuluruhan hasil pengujian menunjukkan bahwa penerapan teknik SMOTE secara efektif membantu mengatasi masalah ketidakseimbangan kelas pada dataset ulasan pengguna aplikasi Threads. Data awal menunjukkan dominasi ulasan positif, yang menyebabkan model cenderung bias terhadap kelas tersebut dan menurunkan performa dalam mengenali ulasan negatif. Setelah dilakukan oversampling dengan SMOTE, distribusi data menjadi lebih seimbang, yang tercermin dari peningkatan jumlah ulasan negatif dan peningkatan performa model dalam mengenali kedua kelas secara lebih adil. Perubahan ini terbukti dari peningkatan nilai recall untuk kelas negatif dari 0,69 menjadi 0,99, serta peningkatan akurasi secara keseluruhan. Visualisasi kata-kata yang paling sering muncul dalam

ulasan positif dan negatif melalui WordCloud juga memberikan wawasan tambahan mengenai aspek apa yang paling diperhatikan oleh pengguna dari segi sentimen mereka, seperti fitur aplikasi dan kendala teknis. Hasil ini memperlihatkan bahwa teknik oversampling tidak hanya meningkatkan performa statistik model, tetapi juga berkontribusi pada interpretasi yang lebih akurat terhadap opini pengguna, yang sangat penting dalam pengembangan dan peningkatan kualitas aplikasi berbasis feedback pengguna.

6. Kesimpulan

Penelitian ini bertujuan untuk mengevaluasi efektivitas algoritma Extreme Gradient Boosting (XGBoost) yang dikombinasikan dengan teknik Synthetic Minority Over-sampling Technique (SMOTE) dalam mengatasi ketidakseimbangan data pada analisis sentimen ulasan pengguna aplikasi Threads di Google Play Store. Berdasarkan hasil yang diperoleh, model XGBoost tanpa SMOTE menghasilkan akurasi sebesar **87,60%**, dengan performa yang kurang optimal dalam mengenali sentimen negatif recall **0,69**. Setelah penerapan SMOTE, akurasi meningkat signifikan menjadi **97,49%**, dengan peningkatan substansial pada kemampuan model mengenali kelas minoritas, ditunjukkan oleh recall sebesar **0,99** untuk kelas negatif dan F1-score yang seimbang antara kelas positif dan negatif.

Penelitian ini membuktikan bahwa SMOTE dapat memperbaiki bias model terhadap kelas mayoritas dan meningkatkan performa klasifikasi secara keseluruhan. Kontribusi utama penelitian ini adalah memperkuat penggunaan teknik balancing dalam machine learning untuk analisis opini pengguna. Namun, penelitian ini masih terbatas pada jumlah data dan dua kelas sentimen. Penelitian selanjutnya disarankan untuk menggunakan dataset lebih besar, menyertakan sentimen netral, dan mencoba algoritma lain yang lebih kontekstual seperti BERT.

Keterbatasan utama dari penelitian ini terletak pada penggunaan dataset yang relatif kecil dan terbatas pada dua kelas sentimen, yaitu positif dan negatif, sehingga belum dapat merepresentasikan seluruh spektrum opini pengguna secara lengkap, terutama yang bersifat netral atau ambigu. Selain itu, proses oversampling dengan SMOTE dapat meningkatkan risiko overfitting karena data sintetis yang dihasilkan mungkin tidak sepenuhnya merepresentasikan variasi asli dari opini pengguna. Selain itu, penggunaan algoritma XGBoost meskipun sangat efektif, masih memiliki keterbatasan dalam menangani konteks bahasa yang lebih kompleks dan berlapis, seperti yang dapat diatasi oleh model berbasis deep learning seperti BERT. Untuk penelitian selanjutnya, disarankan agar mengembangkan dataset yang lebih besar dan beragam, termasuk kelas netral, serta mengaplikasikan model berbasis transformer seperti BERT untuk meningkatkan akurasi dan kemampuan memahami konteks. Selain itu, eksperimen dengan teknik oversampling dan undersampling yang berbeda juga dapat dilakukan untuk mengurangi risiko overfitting dan memperoleh model yang lebih robust terhadap variasi data opini pengguna.

Daftar Pustaka

[1] F. Adelia Irawan, A. Rialdy Atmadja, A. Wahana, T. Informatika, and U. Sunan Gunung Djati Bandung, "Analisis Sentimen Ulasan Aplikasi Bank Digital Menggunakan Algoritma Naive Bayes," *J. Comput. Sci. Inf. Technol. E-ISSN*, vol. 4, no. 2, p. 60, 2024.

[2] Indah Tri Handayani, "19-27+Indah+Tri+Handayani,+Hafidzah,+Uppit+Yuliani," *J. Juit*, vol. Vol 3, no. 1, pp. 19–27, 2024.

[3] H. Zikra *et al.*, "THREADS USER DECLINE ANALYSIS USING THE UNFIELD THEORY OF ACCEPTANCE AND USE OF TECHNOLOGY METHOD ANALISA PENURUNAN PENGGUNA THREADS MENGGUNAKAN METODE UN-FIELD THEORY OF," pp. 927–939, 2024.

[4] A. Ajuj, R. M. G. S. Bintang, and D. Arya Bahytsani, "Analisis Bukti Digital Forensik pada Aplikasi Threads MenggunakanMetode Digital Forensic Research Workshop," *J. Inform. Komputer, Bisnis dan Manaj.*, vol. 22, no. 2, pp. 1–10, 2024, doi: 10.61805/fahma.v22i2.118.

[5] S. S. Tandiapa and G. C. Rorimpandey, "Analisis Sentimen Ulasan Pengguna Pada Aplikasi Threads Dengan Metode Lexicon Based dan Naive Bayes Classifier," *J. Cahaya Mandalika*, vol. 3, no. 1, pp. 339–353, 2024.

[6] Y. A. W. Windy Astuti, Rudi Kurniawan, "Analisis Data Sentimen Ulasan Aplikasi Dana di Google Play Store Menggunakan Algoritma Naïve Bayes," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 6, no. 1, pp. 158–163, 2024, doi: 10.36040/jati.v7i6.8178.

[7] D. Darwis, N. Siskawati, and Z. Abidin, "Penerapan Algoritma Naive Bayes Untuk Analisis Sentimen Review Data Twitter Bmkg Nasional," *J. Tekno Kompak*, vol. 15, no. 1, p. 131, 2021, doi: 10.33365/jtk.v15i1.744.

[8] S. E. Herni Yulianti, Oni Soesanto, and Yuana Sukmawaty, "Penerapan Metode Extreme Gradient Boosting (XGBOOST) pada Klasifikasi Nasabah Kartu Kredit," *J. Math. Theory Appl.*, vol. 4, no. 1, pp. 21–26, 2022, doi: 10.31605/jomta.v4i1.1792.

[9] H. H. Sinaga and S. Agustian, "Pebandingan Metode Decision Tree dan XGBoost untuk Klasifikasi Sentimen Vaksin Covid-19 di Twitter," *J. Nas. Teknol. dan Sist. Inf.*, vol. 8, no. 3, pp. 107–114, 2022, doi: 10.25077/teknosi.v8i3.2022.107-114.

[10] W. M. Baihaqi, I. R. Yunita, A. S. T. Damayanti, and L. Akhaerunnisa, "Analysis of Classification Algorithm Performance on User Review Sentiment of the Muamalat DIN Application," *CogITo Smart J.*, vol. 9, no. 2, pp. 241–251, 2023, doi: 10.31154/cogito.v9i2.511.241-251.

[11] R. Nazar, "Implementasi Pemrograman Python Menggunakan Google Colab," *J. Inform. dan Komput.* , vol. 15, no. 1, pp. 50–56, 2024.

[12] R. Gelar Guntara, "Pemanfaatan Google Colab Untuk Aplikasi Pendeteksian Masker Wajah Menggunakan Algoritma Deep Learning YOLOv7," *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 5, no. 1, pp. 55–60, 2023, doi: 10.47233/jteksis.v5i1.750.

[13] S. Adi *et al.*, "ANALISIS SENTIMEN APLIKASI HALO BCA DI GOOGLE PLAY STORE MENGGUNAKAN METODE NAIVE BAYES , SUPPORT VECTOR MACHINE DAN RANDOM FOREST PENDAHULUAN," vol. 15, no. c, pp. 69–79, 2024.

[14] I. F. Rahman, A. N. Hasanah, and N. Heryana, "Analisis Sentimen Ulasan Pengguna Aplikasi Samsat Digiital Nasional (Signal) Dengan Menggunakan Metode Naïve Bayes Classifier," *J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 2, pp. 963–969, 2024, doi: 10.23960/jitet.v12i2.4073.

[15] F. Matheos Sarimole and K. Kudrat, "Analisis Sentimen Terhadap Aplikasi Satu Sehat Pada Twitter Menggunakan Algoritma Naive Bayes Dan Support Vector Machine," *J. Sains dan Teknol.*, vol. 5, no. 3, pp. 783–790, 2024, doi: 10.55338/saintek.v5i3.2702.

[16] M. F. Y. Herjanto and C. Carudin, "Analisis Sentimen Ulasan Pengguna Aplikasi Sirekap Pada Play Store Menggunakan Algoritma Random Forest Classifer," *J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 2, pp. 1204–1210, 2024, doi: 10.23960/jitet.v12i2.4192.

[17] R. Q. Rohmansa, N. Pratiwi, and M. J. Palepa, "Analisis Sentimen Ulasan Pengguna Aplikasi Discord Menggunakan Metode K-Nearest Neighbor," *JIPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, vol. 9, no. 1, pp. 368–378, 2024, doi: 10.29100/jipi.v9i1.4943.

[18] N. P. Husain, S. Sukirman, and S. SAJIAH, "Analisis Sentimen Ulasan Pengguna Tiktok pada Google Play Store Berbasis TF-IDF dan Support Vector Machine," *J. Syst. Comput. Eng.*, vol. 5, no. 1, pp. 91–102, 2024, doi: 10.61628/jsce.v5i1.1105.

[19] H. Hidayatullah, P. Purwantoro, and Y. Umaidah, "PENERAPAN NAÏVE BAYES DENGAN OPTIMASI INFORMATION GAIN DAN SMOTE UNTUK ANALISIS SENTIMEN PENGGUNA APLIKASI CHATGPT," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 7, no. 3, pp. 1546–1553, Oct. 2023, doi: 10.36040/jati.v7i3.6887.

[20] F. S. Pratiwi *et al.*, "IMPLEMENTASI METODE SMOTE DAN RANDOM OVER- SAMPLING PADA ALGORITMA MACHINE LEARNING UNTUK," vol. 8, no. 1, pp. 87–98, 2025.

[21] M. P. Pulungan, A. Purnomo, and A. Kurniasih, "Penerapan SMOTE untuk Mengatasi Imbalance Class dalam Klasifikasi Kepribadian MBTI Menggunakan Naive Bayes Classifier," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 10, no. 7, pp. 1493–1502, 2023, doi: 10.25126/jtiik.1077989.

[22] L. N. Wakhidah, A. K. Zyen, and B. B. Wahono, "Evaluation of Telecommunication Customer Churn Classification with SMOTE Using Random Forest and XGBoost Algorithms," vol. 9, no. 1, pp. 89–95, 2025.

[23] D. Safitri, Susanti, Rahmaddeni, and T. A. Fitri, "Perbandingan Algoritma XGBoost dan SVM Dalam Analisis Opini Publik Pemilihan Presiden 2024," *Indones. J. Comput. Sci.*, vol. 13, no. 3, pp. 4763–4775, 2024, doi: 10.33022/ijcs.v13i3.4041.

[24] "Eskiyaturrofikoh" and R. R. 'Suryono, "Analisis Sentimen Aplikasi X Pada Google Play Store Menggunakan Algoritma Naïve Bayes Dan Support Vector Machine (Svm)," *JIPI(Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, vol. 9, no. 3, pp. 1408–1419, 2024, [Online]. Available: https://www.jurnal.stkippgritulungagung.ac.id/index.php/jipi/article/view/5392

[25] M. Febiansyah *et al.*, "Prediksi Retweet Berdasarkan Konten Dan Pengguna Dengan Metode Classifier Selection," vol. 12, no. 1, pp. 2105–2111, 2025.