

# OPTIMIZATION OF SENTIMENT ANALYSIS ON TOKOPEDIA USER REVIEWS USING GRIDSEARCHCV AND SMOTE WITH MACHINE LEARNING ALGORITHMS

## OPTIMASI ANALISIS SENTIMEN ULASAN TOKOPEDIA DENGAN GRIDSEARCHCV, SMOTE DAN ALGORITMA MACHINE LEARNING

Athallah Yasyfi Imran<sup>1\*</sup>, M. Rudi Sanjaya<sup>2\*</sup>, Bayu Wijaya Putra<sup>3</sup>,  
Gabriel Ekoputra Hartono Cahyadi<sup>4</sup>

<sup>1,2,3,4</sup>Universitas Sriwijaya, Sistem Informasi, Jl. Palembang – Prabumulih KM. 32, Indralaya, Kabupaten Ogan Ilir, Sumatera Selatan 30662

Email : athallahyasyfi@gmail.com<sup>1</sup>, m.rudi.sjy@ilkom.unsri.ac.id<sup>2\*</sup>

**Abstract** - Understanding user sentiment from e-commerce reviews is essential for platform improvement and business strategy. This study compares three machine learning algorithms Logistic Regression, Random Forest, and XGBoost for sentiment classification of Indonesian-language Tokopedia reviews. A dataset of 6,822 user reviews was preprocessed through tokenization, stopword removal, and TF-IDF vectorization. To address class imbalance, the Synthetic Minority Oversampling Technique (SMOTE) was applied to the training set. Models were evaluated using accuracy, precision, recall, and F1-score. Results demonstrate that Random Forest achieved the highest accuracy at 86.86%, followed by Logistic Regression at 84.86%, and XGBoost at 82.60%. The application of SMOTE significantly improved classification performance across all models, particularly for minority sentiment classes. These findings indicate that tree-based ensemble methods, especially Random Forest, are effective for sentiment analysis in imbalanced e-commerce datasets. This research provides practical insights for e-commerce platforms to implement automated sentiment monitoring systems, enabling faster response to customer feedback and targeted service improvements. However, the study is limited to Tokopedia reviews and may not generalize to other platforms or languages. Future work should explore deep learning approaches and cross-platform validation to enhance model robustness.

**Keywords** – Sentiment Analysis, Machine Learning, SMOTE, E-commerce, TF-IDF

**Abstrak** - Analisis sentimen ulasan pengguna e-commerce sangat penting untuk perbaikan platform dan strategi bisnis. Penelitian ini membandingkan kinerja tiga algoritma machine learning Logistic Regression, Random Forest, dan XGBoost dalam klasifikasi sentimen ulasan pengguna Tokopedia berbahasa Indonesia. Dataset berisi 6.822 ulasan yang diproses melalui tokenisasi, penghapusan stopword, dan vektorisasi TF-IDF. Untuk mengatasi ketidakseimbangan kelas, diterapkan Synthetic Minority Oversampling Technique (SMOTE) pada data latih. Model dievaluasi menggunakan metrik akurasi, presisi, recall, dan F1-score. Hasil penelitian menunjukkan bahwa Random Forest mencapai akurasi tertinggi sebesar 86,86%, diikuti Logistic Regression 84,86%, dan XGBoost 82,60%. Penerapan SMOTE terbukti meningkatkan kinerja klasifikasi pada seluruh model, khususnya pada kelas sentimen minoritas. Temuan ini menunjukkan bahwa metode ensemble berbasis pohon keputusan, terutama Random Forest, efektif untuk analisis sentimen pada dataset e-commerce yang tidak seimbang. Penelitian ini memberikan wawasan praktis bagi platform e-commerce untuk mengimplementasikan sistem monitoring sentimen otomatis, memungkinkan respons lebih cepat terhadap umpan balik pelanggan dan perbaikan layanan yang lebih terarah. Namun, penelitian terbatas pada ulasan Tokopedia dan belum tentu dapat digeneralisasi ke platform atau bahasa lain. Riset mendatang perlu mengeksplorasi pendekatan deep learning dan validasi lintas platform.

**Kata Kunci** - Sentiment Analysis, Machine Learning, SMOTE, E-commerce, TF-IDF.

## I. PENDAHULUAN

Transformasi digital sektor ritel Indonesia telah mengakselerasi adopsi platform *e-commerce* secara masif. Tokopedia, sebagai salah satu marketplace dominan dengan pangsa pasar terbesar tahun 2023, mengalami lonjakan signifikan volume transaksi dan interaksi pengguna[1]. Fenomena ini menghasilkan akumulasi data ulasan pengguna dalam skala petabyte yang mengandung informasi berharga mengenai persepsi konsumen terhadap produk dan layanan. Kompleksitas analisis manual terhadap volume ulasan yang eksponensial menciptakan bottleneck dalam pengambilan keputusan bisnis berbasis *customer feedback*. Subjektivitas interpretasi manusia dan keterbatasan kapasitas pemrosesan menghasilkan inkonsistensi dalam ekstraksi insight, sehingga organisasi kehilangan peluang optimalisasi berdasarkan *voice of customer* yang autentik.

Analisis sentimen sebagai subdisiplin *natural language processing* menawarkan solusi sistematis untuk mengotomatisasi klasifikasi opini pengguna ke dalam kategori sentimen diskrit [2]. Pendekatan komputasional ini memungkinkan transformasi data tekstual kualitatif menjadi representasi kuantitatif yang dapat dianalisis secara statistik untuk mengidentifikasi pola sentiment dan trend kepuasan pelanggan. Seleksi algoritma klasifikasi menjadi determinan kritical dalam keberhasilan implementasi sistem analisis sentimen. Literature menunjukkan variabilitas performa algoritma machine learning bergantung pada karakteristik domain, bahasa, dan distribusi data [3]. Logistic Regression menawarkan interpretabilitas tinggi dan efisiensi komputasi sebagai baseline classifier. Random Forest dengan mekanisme ensemble bagging menunjukkan robustness terhadap overfitting dan kemampuan menangani data berdimensi tinggi. XGBoost mengimplementasikan gradient boosting dengan optimisasi yang canggih untuk menangkap hubungan non-linear secara lebih efektif.

Meskipun analisis sentimen berbahasa Indonesia telah banyak dilakukan, tantangan ketidakseimbangan kelas sentimen masih menjadi kendala utama, terutama pada kelas netral yang lebih sulit dikenali dibandingkan kelas positif dan negatif. Untuk mengatasi hal ini, beberapa penelitian telah menerapkan Synthetic Minority Oversampling Technique (SMOTE). Syaputra et al. [4] menunjukkan bahwa kombinasi TF-IDF dan SMOTE meningkatkan akurasi klasifikasi ulasan e-wallet hingga 90%. Hasil serupa juga ditunjukkan Cahyaningtyas *et al.* [5] pada ulasan pengguna Shopee, di mana SMOTE berpengaruh signifikan terhadap peningkatan akurasi dan AUC. Namun, kajian yang secara khusus mengevaluasi kestabilan berbagai algoritma pada ulasan Tokopedia dengan fokus pada distribusi kelas sentimen masih terbatas.

Berdasarkan permasalahan tersebut, penelitian ini berupaya menjawab pertanyaan riset berikut:

*“Algoritma machine learning mana yang paling stabil dan akurat untuk klasifikasi tiga kelas sentimen pada ulasan pengguna Tokopedia berbahasa Indonesia setelah penerapan SMOTE?”*

Penelitian ini bertujuan untuk melakukan evaluasi komparatif terhadap tiga algoritma Logistic Regression, Random Forest, dan XGBoost—pada dataset ulasan Tokopedia menggunakan pendekatan TF-IDF sebagai ekstraksi fitur dan GridSearchCV untuk optimisasi hiperparameter. Adapun kontribusi utama penelitian ini meliputi: (i) Penyusunan dataset ulasan berbahasa Indonesia yang representatif dari domain e-commerce; (ii) Evaluasi performa tiga algoritma machine learning menggunakan representasi TF-IDF; (iii) Analisis dampak penerapan SMOTE terhadap kestabilan performa model per kelas sentimen; dan (iv) Rilis hasil evaluasi sebagai baseline serta rekomendasi praktik terbaik untuk penelitian lanjutan di bidang ID-NLP.

## II. SIGNIFIKASI STUDI

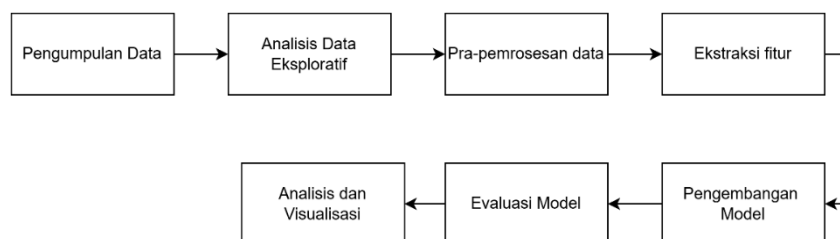
### A. Studi Literatur

Beberapa penelitian terkini menunjukkan efektivitas SMOTE dalam meningkatkan performa klasifikasi teks tidak seimbang, khususnya pada Logistic Regression, Random Forest, dan XGBoost. Budaya & Suniantara (2024) mencatat akurasi Logistic Regression meningkat dari 82% menjadi 89%, dan Random Forest dari 83% menjadi 88% setelah penerapan SMOTE pada ulasan Google Reviews [6]. Siagian et al. (2025) melaporkan akurasi XGBoost naik dari 80% menjadi 91% pada data ulasan produk [7], sementara Siregar & Arifin (2024) mencatat peningkatan dari 79% ke 90% dengan SVM-SMOTE [8]. Studi lokal lain juga menunjukkan tren serupa, dengan Logistic Regression meningkat dari 78% ke 86% dan Random Forest dari 81% ke 88% pada opini publik Indonesia [9]. Namun, sebagian besar penelitian tersebut masih fokus pada klasifikasi biner dan belum menyentuh multi-kelas pada domain e-commerce Indonesia, khususnya Tokopedia. Penelitian ini berkontribusi (i) teoretis dengan memberikan bukti empiris dampak SMOTE pada klasifikasi multi-kelas teks Indonesia, dan (ii) praktis dengan menjadi acuan pemilihan model yang efisien dari segi akurasi dan sumber daya komputasi.

### B. Metode Penelitian

#### 1. Desain Penelitian

Penelitian ini mengadopsi paradigma kuantitatif dengan desain eksperimental komparatif. Pendekatan *comparative experimental design* dipilih untuk mengevaluasi treatment effect tiga algoritma machine learning (Random Forest, Logistic Regression, XGBoost) terhadap variable dependen berupa akurasi klasifikasi sentimen. *Framework* penelitian mengintegrasikan *Research and Development (R&D) methodology* yang memungkinkan siklus iteratif development, testing, dan refinement sistem klasifikasi [10]. Pendekatan R&D memfasilitasi tidak hanya model generation tetapi juga *continuous evaluation* dan *optimization* untuk menghasilkan sistem yang *robust* dan *reliable*.



**Gambar 1.** Desain Penelitian

#### 2. Dataset dan Sumber Data

Dataset penelitian terdiri dari ulasan pengguna aplikasi Tokopedia yang diekstrak dari Google Play Store menggunakan *automated web scraping technique*. Kriteria inklusi dataset meliputi: (1) ulasan berbahasa Indonesia, (2) mengandung content textual substantive (minimum 10 karakter), (3) bebas dari spam atau duplicate content, dan (4) memiliki timestamp dalam periode terbaru untuk memastikan agar data relevan. Data *preprocessing* melibatkan *quality assurance procedure* untuk menghilangkan noise dan memastikan integritas data. Proses *cleaning* mencakup penghapusan entri duplikat, penyaringan konten spam, dan normalisasi *encoding* teks untuk mempertahankan konsistensi di seluruh dataset.

#### 3. Preprocessing dan Feature Engineering

Pipeline preprocessing teks mengimplementasikan tahapan terstandar untuk Bahasa Indonesia [11]. Tahapan ini mencakup case folding (konversi huruf ke bentuk kecil) [12], normalisasi slang menggunakan kamus slang internal (mis. “gk” → “tidak”, “bgt” → “banget”, “tokped” → “tokopedia”), serta tokenisasi berbasis spasi dan tanda baca. Karakter non-alfabetik dihapus untuk menjaga fokus semantik [11] sedangkan stopword removal menggunakan daftar Bahasa Indonesia dari Sastrawi dan NLTK, ditambah stopword domain seperti “tokopedia” dan “barang”. Stemming

dilakukan dengan pustaka Sastrawi Stemming diterapkan sebagai normalisasi morfologis dengan pustaka Sastrawi untuk mengurangi variasi infleksi ke bentuk dasar [12]. Ekstraksi fitur menggunakan TF-IDF (Term Frequency–Inverse Document Frequency) dengan parameter dasar  $n\text{-gram\_range} = (1, 2)$ ,  $\text{min\_df} = 2$ ,  $\text{max\_df} = 0.95$ ,  $\text{sublinear\_tf} = \text{True}$ . Selain itu, dilakukan validasi  $n\text{-gram}$  pada (1,1), (1,2), dan (1,3) untuk memperoleh konfigurasi optimal. TF-IDF dipilih karena kemampuannya dalam pembobotan kata dan pengelolaan data berdimensi tinggi [13].

#### 4. Handling Class Imbalance dengan SMOTE

SMOTE (Synthetic Minority Over-sampling Technique) sering digunakan dalam tahapan pra-pemrosesan untuk mengatasi ketidakseimbangan kelas pada data teks, yaitu dengan menciptakan contoh sintetik pada kelas minoritas sehingga distribusi kelas menjadi lebih seimbang [14]. Ketidakseimbangan kelas ditangani menggunakan pustaka *imbalanced-learn*, dengan Parameter SMOTE yang digunakan: (1)  $\text{sampling\_strategy} = \text{'auto'}$  (2)  $k\text{-neighbors} = 5$  (3)  $\text{random\_state} = 42$ . Penerapan SMOTE dilakukan hanya pada data latih dalam setiap fold melalui *ImbPipeline* untuk mencegah data leakage. Dengan cara ini, data uji tidak ikut dalam proses pembentukan sampel sintesis sehingga evaluasi model lebih valid.

#### 5. Model Development dan Hyperparameter Optimization

Algorithm Implementation menggunakan tiga algoritma target yang diimplementasikan menggunakan *scikit-learn* dan *XGBoost* libraries dengan konfigurasi optimal.

- a) *Logistic Regression* (LR) adalah model statistik untuk klasifikasi biner, di mana variabel respons hanya memiliki dua kemungkinan nilai (misalnya 0 dan 1); model ini memodelkan peluang (probabilitas) terjadinya kelas 1 sebagai kombinasi linier variabel prediktor dengan menerapkan fungsi logit (sigmoid) sehingga outputnya berada dalam rentang 0 hingga 1 [15]. Parameter yang digunakan:  $C$  (*regularization strength*): [0.1, 1, 10].
- b) *Random Forest* memecah ruang data secara rekursif, di mana setiap pohon keputusan tumbuh dengan mempartisi ruang fitur menjadi node dan cabang yang merepresentasikan hubungan nonlinier di antara fitur [16]. *Random Forest* menggunakan *ensemble* dari *decision trees* dengan *voting mechanism*. Parameter yang digunakan:  $n\text{-estimators}$ : [50, 100, 200], dan  $\text{max\_depth}$ : [None, 10, 20].
- c) *XGBoost* adalah varian *gradient boosting* yang dioptimalkan untuk kinerja tinggi dan mampu menangani dataset besar. Literatur menyatakan *XGBoost* “didesain untuk performa tinggi dan kecepatan” dengan optimasi paralel/terdistribusi sehingga sangat cocok untuk data berukuran besar [17]. Parameter yang digunakan:  $n\text{-estimators}$ : [50, 100, 200],  $\text{max\_depth}$ : [3, 5, 7], dan  $\text{learning\_rate}$ : [0.01, 0.1, 0.3]

#### 6. Hyperparameter Tuning dan Evaluation

Selama pelatihan dilakukan *hyperparameter tuning* (misalnya menentukan banyak pohon  $n\text{-estimators}$ , kedalaman pohon  $\text{max\_depth}$ , parameter regularisasi, *learning rate*, dll.) menggunakan *GridSearchCV* untuk mendapatkan kombinasi parameter terbaik. Performance Metrics menggunakan evaluasi model dengan metrik komprehensif untuk menangkap berbagai aspek performa klasifikasi yaitu *Accuracy*, *Precision*, *Recall* dan *F1-Score* [18]. Strategi validasi mengimplementasikan *stratified k-fold cross-validation* ( $k=5$ ) untuk memastikan estimasi performa yang *robust*. Stratifikasi mempertahankan representasi proporsional setiap kelas sentimen dalam setiap fold, meminimalkan bias dari ketidakaturan distribusi kelas. *Holdout test* set tambahan (20% dari total dataset) dialokasikan untuk evaluasi model final, memastikan penilaian performa yang tidak bias pada data yang belum terlihat.

### III. HASIL DAN PEMBAHASAN

#### A. Pengumpulan dan Karakteristik Dataset

Penelitian ini menggunakan dataset ulasan aplikasi Tokopedia yang dikumpulkan melalui Google Play Store API dengan total 10.000 ulasan awal. Tabel 2 menunjukkan sampel dataset yang berhasil dikumpulkan, mencakup informasi username, skor rating, timestamp, dan konten ulasan. Dataset mentah kemudian disimpan dalam format CSV untuk tahap preprocessing selanjutnya.

Tabel 1. Sampel Dataset Pengumpulan Data

Username	Score	At	Content
Tenebris Skandavia	1	2025-09-13 14:15:30	banyak penipuan

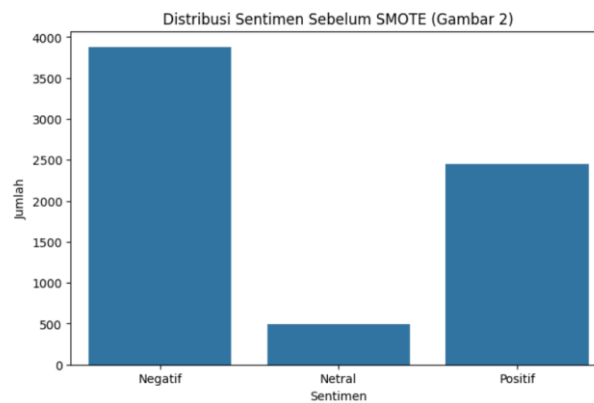
Tahap pembersihan data menghasilkan 6.822 ulasan valid setelah melalui proses eliminasi duplikasi, filtering panjang teks minimum 5 karakter, penghapusan spam, dan deteksi bahasa Indonesia. Tabel 3 menampilkan hasil pelabelan sentimen berdasarkan skor rating, dimana rating 1-2 dikategorikan sebagai sentimen negatif, rating 3 sebagai netral, dan rating 4-5 sebagai positif.

Tabel 2. Sampel Dataset Pelabelan Data

Content	Score	Label
banyak penipuan	1	Negatif

#### B. Distribusi dan Karakteristik Sentimen

Analisis distribusi sentimen pada **Gambar 2** menunjukkan ketidakseimbangan data yang signifikan, dengan dominasi sentimen negatif sebesar 56,75% (3.884 ulasan), diikuti sentimen positif 35,99% (2.463 ulasan), dan sentimen netral hanya 7,26% (497 ulasan). Ketidakseimbangan ini mencerminkan kecenderungan pengguna untuk lebih vokal dalam menyampaikan keluhan dibandingkan pujian.



Gambar 2. Distribusi Sentimen Sebelum SMOTE

#### B. Preprocessing Teks dan Transformasi Data

Proses *preprocessing* teks dilakukan melalui serangkaian tahapan yang sistematis. Tabel 4 hingga Tabel 9 menyajikan hasil setiap tahapan, yang meliputi *case folding*, normalisasi slang, *cleaning* (penghapusan tanda baca dan angka), *stopword removal*, *tokenizing*, serta *stemming* dengan menggunakan algoritma Sastrawi.

Tabel 3. Case Folding

Original Text	Case Folding
ribet amat ni apk verifikasi data diri , harus ini lah itu lah, ga bisa lebih simpel lagi ?? misal selfie dgn ktp aja tanpa harus scan qr	ribet amat ni apk verifikasi data diri , harus ini lah itu lah, ga bisa lebih simpel lagi ?? misal selfie dgn ktp aja tanpa harus scan qr

Tabel 4. Normalisasi Slang

Original Text	Normalized Text
ribet amat ni apk verifikasi data diri , harus ini lah itu lah, ga bisa lebih simpel lagi ?? misal selfie dgn ktp aja tanpa harus scan qr	ribet amat ni apk verifikasi data diri , harus ini lah itu lah, tidak bisa lebih simpel lagi ?? misal selfie dengan ktp saja tanpa harus scan qr

Tabel 5. Cleaning

Original Text	Cleaned Text
ribet amat ni apk verifikasi data diri , harus ini lah itu lah, ga bisa lebih simpel lagi ?? misal selfie dgn ktp aja tanpa harus scan qr	ribet amat ni apk verifikasi data diri harus ini lah itu lah tidak bisa lebih simpel lagi misal selfie dengan ktp saja tanpa harus scan qr

Tabel 6. Stopword Removal

Original Text	After Stopword Removal
ribet amat ni apk verifikasi data diri , harus ini lah itu lah, ga bisa lebih simpel lagi ?? misal selfie dgn ktp aja tanpa harus scan qr	ribet ni apk verifikasi data simpel selfie ktp scan qr

Tabel 7. Tokenizing

Original Text	Tokens
ribet amat ni apk verifikasi data diri , harus ini lah itu lah, ga bisa lebih simpel lagi ?? misal selfie dgn ktp aja tanpa harus scan qr	['ribet', 'ni', 'apk', 'verifikasi', 'data', 'simpel', 'selfie', 'ktp', 'scan', 'qr']

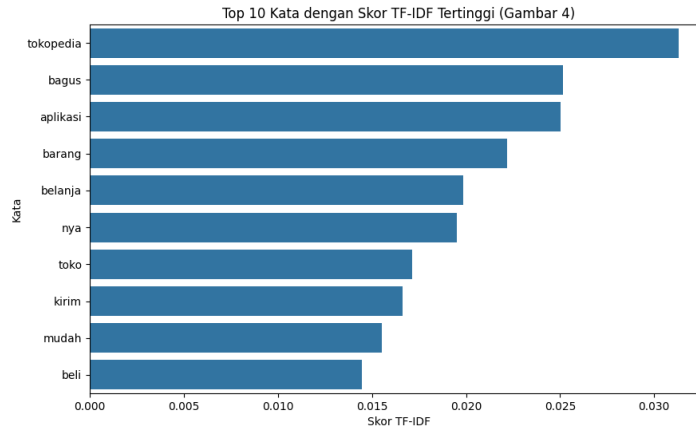
Tabel 8. Stemming

Original Text	Stemmed Text
ribet amat ni apk verifikasi data diri , harus ini lah itu lah, ga bisa lebih simpel lagi ?? misal selfie dgn ktp aja tanpa harus scan qr	ribet ni apk verifikasi data simpel selfie ktp scan qr

Tabel 6 menunjukkan hasil *cleaning* yang berfokus pada penghapusan tanda baca serta karakter non-alfabet sehingga teks menjadi lebih bersih dan terstandarisasi. Tabel 7 menggambarkan efektivitas *stopword removal* dalam mengeliminasi kata-kata umum yang tidak memiliki makna signifikan, seperti “di”, “ke”, dan “dari”. Tabel 8 menyajikan hasil *tokenizing* yang memecah teks menjadi unit-unit kata individual. Selanjutnya, Tabel 9 menampilkan hasil *stemming* yang mereduksi kata ke bentuk dasarnya menggunakan algoritma Sastrawi.

### C. Ekstraksi Fitur dengan TF-IDF

Implementasi TF-IDF dengan parameter `max_features=10.000` dan `ngram_range=(1,2)` berhasil mengekstrak fitur tekstual yang relevan. Tabel 9 dan Gambar 4 menunjukkan 10 kata dengan skor TF-IDF tertinggi, dimana "tokopedia" (0,0303), "aplikasi" (0,0245), dan "bagus" (0,0235) menjadi term paling dominan.



Gambar 3. 10 Kata dengan Skor TF-IDF Tertinggi

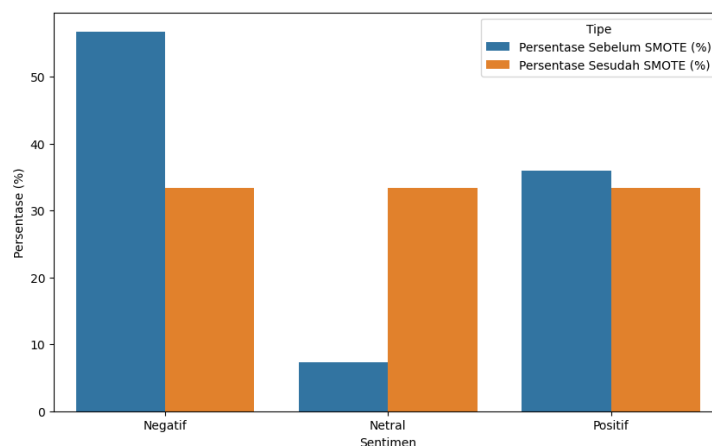
Tingginya skor TF-IDF untuk kata "tokopedia" dan "aplikasi" mengindikasikan bahwa pengguna sering menyebut nama platform secara eksplisit dalam ulasan mereka. Kehadiran kata "bagus" dan "barang" di peringkat teratas menunjukkan fokus pengguna pada kualitas produk dan layanan.

**D. Distribusi Sentimen Sebelum dan Sesudah SMOTE**

Distribusi awal sentimen pada dataset menunjukkan ketidakseimbangan kelas yang cukup signifikan. Sentimen negatif mendominasi dengan persentase 56,75%, diikuti sentimen positif sebesar 35,99%, sementara sentimen netral hanya 7,26%. Ketimpangan ini berpotensi menimbulkan bias pada model klasifikasi, terutama dalam mendeteksi kelas minoritas. Untuk mengatasi permasalahan tersebut, digunakan teknik *Synthetic Minority Oversampling Technique* (SMOTE). Hasil penerapan SMOTE berhasil menyeimbangkan distribusi kelas sehingga masing-masing kelas (positif, negatif, netral) memiliki proporsi yang sama yaitu 33,33%.

Tabel 9. Distribusi Sentimen Sebelum dan Sesudah SMOTE

Sentimen	Jumlah Sebelum SMOTE	Persentase Sebelum SMOTE (%)	Jumlah Sesudah SMOTE	Persentase Sesudah SMOTE (%)
Negatif	3884	56.7504	3884	33.3333
Netral	497	7.2618	3884	33.3333
Positif	2463	35.9877	3884	33.3333



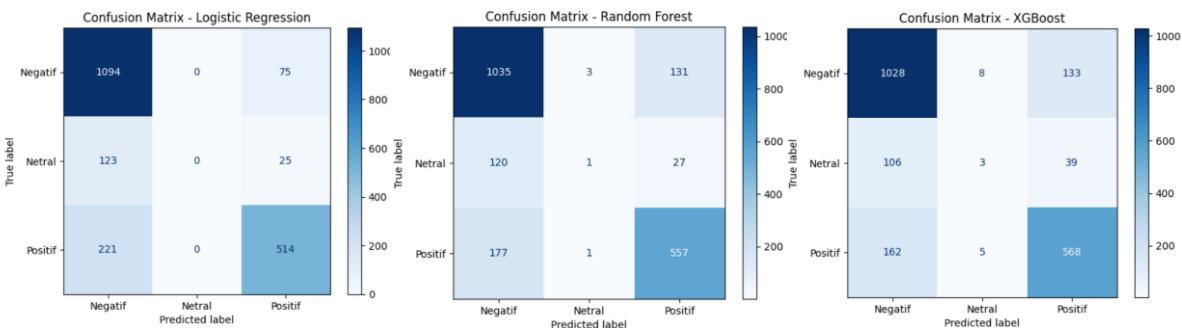
Gambar 4. Visualisasi Distribusi Sentimen

**E. Hasil Pelatihan dan Evaluasi Model**

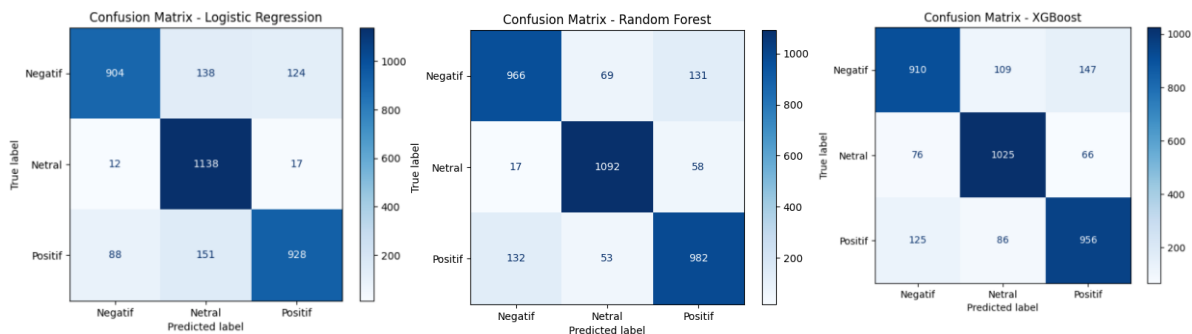
Setelah melalui tahap *preprocessing* dan *balancing* data, langkah berikutnya adalah menguji performa model klasifikasi. Pada penelitian ini digunakan tiga algoritma pembandingan, yaitu *Logistic Regression (LR)*, *Random Forest (RF)*, dan *XGBoost (XGB)*. Proses pengujian dilakukan dengan membagi dataset menjadi 70% data latih dan 30% data uji, di mana data latih digunakan untuk membangun model sedangkan data uji digunakan untuk mengukur kemampuan generalisasi model terhadap data baru. Hasil pengujian model ditampilkan pada Tabel 10, yang memuat perbandingan performa algoritma sebelum dan sesudah penerapan SMOTE.

Tabel 10. Hasil Evaluasi Model Sebelum dan Sesudah SMOTE

Metrik / Sentimen	<i>Logistic Regression</i>	<i>Logistic Regression + SMOTE</i>	<i>Random Forest</i>	<i>Random Forest + SMOTE</i>	<i>XGBoost</i>	<i>XGBoost + SMOTE</i>
<i>Accuracy</i>	0.7836	0.8486	0.7763	0.8686	0.7792	0.8260
<i>Macro-F1</i>	0.5338	0.8469	0.5364	0.8681	0.5469	0.8255
<i>Negatif – Precision</i>	0.7608	0.9004	0.7770	0.8664	0.7932	0.8191
<i>Negatif – Recall</i>	0.9358	0.7753	0.8854	0.8285	0.8794	0.7804
<i>Negatif – F1</i>	0.8393	0.8332	0.8277	0.8470	0.8341	0.7993
<i>Netral – Precision</i>	0.0000	0.7975	0.2000	0.8995	0.1875	0.8402
<i>Netral – Recall</i>	0.0000	0.9751	0.0068	0.9357	0.0203	0.8783
<i>Netral – F1</i>	0.0000	0.8774	0.0131	0.9173	0.0366	0.8588
<i>Positif – Precision</i>	0.8371	0.8681	0.7790	0.8386	0.7676	0.8178
<i>Positif – Recall</i>	0.6993	0.7952	0.7578	0.8415	0.7728	0.8192
<i>Positif – F1</i>	0.7620	0.8301	0.7683	0.8400	0.7702	0.8185



Gambar 5. Confusion Matrix Sebelum SMOTE



Gambar 6. Confusion Matrix Sesudah SMOTE

Analisis terhadap confusion matrix memberikan gambaran detail mengenai distribusi prediksi model. Sebelum penerapan SMOTE, Logistic Regression mengalami bias kuat terhadap kelas mayoritas. Hal ini terlihat dari nilai precision dan recall pada kelas netral yang bernilai 0.0000, menandakan model sama sekali tidak mampu mengenali sentimen netral. Setelah penerapan SMOTE, kinerja

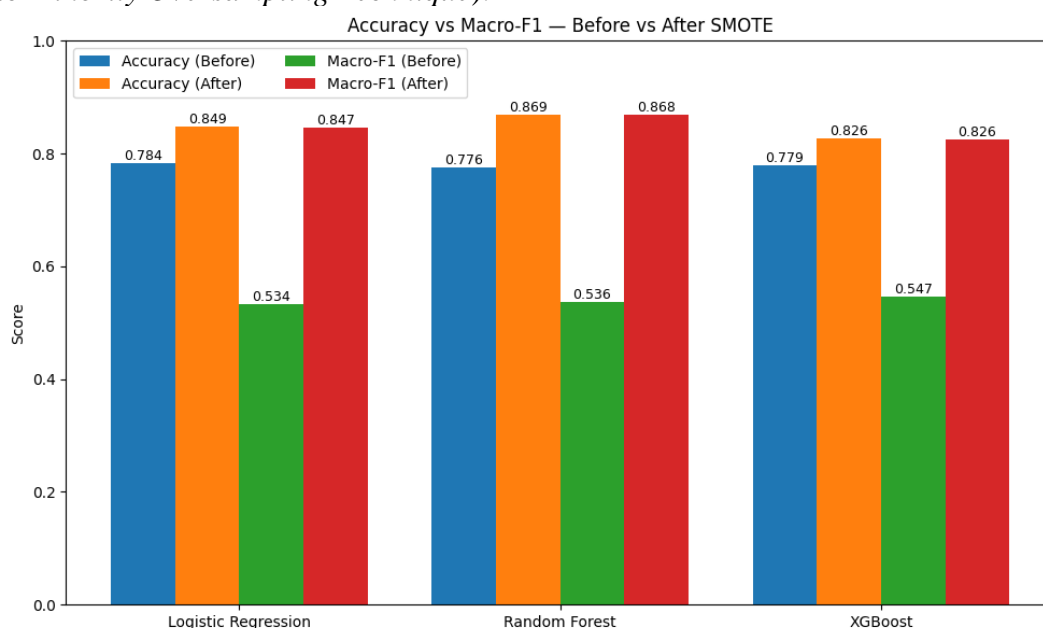


meningkat drastis dengan recall netral naik menjadi 0.9751 dengan F1 mencapai 0.8774. Meski demikian, terjadi penurunan recall pada kelas negatif (dari 0.9358 menjadi 0.7753), menandakan adanya trade-off ketika sensitivitas terhadap kelas minoritas diperbaiki. Secara keseluruhan, akurasi meningkat dari 0.7836 menjadi 0.8486, dan macro-F1 naik signifikan dari 0.5338 menjadi 0.8469, yang mencerminkan peningkatan keseimbangan performa pada seluruh kelas sentimen.

Random Forest pada kondisi awal sudah relatif stabil, dengan performa cukup baik pada kelas positif dan negatif. Namun, kemampuan mengenali sentimen netral masih rendah (recall hanya 0.0068, F1 0.0131). Setelah SMOTE diterapkan, model mampu mengenali netral jauh lebih baik (recall naik menjadi 0.9357, F1 0.9173). Selain itu, akurasi keseluruhan meningkat dari 0.7763 menjadi 0.8686 dan macro-F1 meningkat dari 0.5364 menjadi 0.8681, yang merupakan nilai tertinggi di antara semua model. Hal ini menunjukkan bahwa Random Forest mendapatkan manfaat paling signifikan dari SMOTE. XGBoost sebelum balancing juga lemah dalam mengenali sentimen netral (recall hanya 0.0203, F1 0.0366). Setelah penerapan SMOTE, terjadi peningkatan besar dengan recall netral mencapai 0.8783 dan F1 0.8588. Meski demikian, performa kelas negatif sedikit menurun (recall dari 0.8794 ke 0.7804), dan presisi pada kelas positif meningkat moderat. Akurasi keseluruhan juga naik dari 0.7792 menjadi 0.8260, sementara macro-F1 meningkat dari 0.5469 menjadi 0.8255, meskipun masih berada di bawah performa Random Forest + SMOTE.

#### F. Visualisasi

Untuk memberikan gambaran yang komprehensif mengenai performa model klasifikasi sentimen, dilakukan visualisasi perbandingan akurasi ketiga model machine learning yang telah diuji. Visualisasi ini menampilkan perbandingan akurasi sebelum dan sesudah penerapan teknik SMOTE (*Synthetic Minority Oversampling Technique*).



Gambar 7. Perbandingan Akurasi dan Macro-F1 Sebelum dan Sesudah SMOTE

Dari visualisasi perbandingan akurasi yang ditampilkan pada Gambar 7, dapat diamati bahwa penerapan teknik SMOTE memberikan dampak positif yang signifikan terhadap performa semua model. Random Forest mencapai akurasi tertinggi setelah penerapan SMOTE dengan nilai 86,86%, sekaligus mengalami peningkatan akurasi terbesar (+9,23%). Selain itu, macro-F1 juga meningkat signifikan dari 0,536 menjadi 0,868 atau naik sekitar 61,94%, menunjukkan peningkatan kinerja model yang lebih merata di semua kelas sentimen. Peningkatan akurasi dan Macro-F1 yang konsisten pada semua model membuktikan efektivitas teknik SMOTE dalam mengatasi masalah ketidakseimbangan data pada dataset ulasan Tokopedia.



- [3] G. Airlangga, "Comparative Study of XGBoost, Random Forest, and Logistic Regression Models for Predicting Customer Interest in Vehicle Insurance," *sinkron*, vol. 8, no. 4, pp. 2542–2549, Oct. 2024, doi: 10.33395/sinkron.v8i4.14194.
- [4] M. R. Syaputra, M. Arifin, and D. L. Fithri, "Edumaac: Jurnal Pendidikan Informaaka Klasifikasi Sentimen Ulasan E-Wallet menggunakan TF-IDF dan Random Forest dengan Penyeimbangan Data SMOTE," vol. 9, no. 2, pp. 532–541, 2025, doi: 10.29408/eduma.
- [5] C. Cahyaningtyas, Y. Nataliani, and I. R. Widiyasa, "Analisis sentimen pada rating aplikasi Shopee menggunakan metode Decision Tree berbasis SMOTE," *AITI: Jurnal Teknologi Informasi*, vol. 18, no. Agustus, pp. 173–184, 2021.
- [6] I. G. B. A. Budaya and I. K. P. Suniantara, "Comparison of Sentiment Analysis Algorithms with SMOTE Oversampling and TF-IDF Implementation on Google Reviews for Public Health Centers," *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 4, no. 3, pp. 1077–1086, Jul. 2024, doi: 10.57152/malcom.v4i3.1459.
- [7] N. A. Siagian, S. P. Sipayung, A. Rikki, and N. Marbun, "Integrating SMOTE with XGBoost for Robust Classification on Imbalanced Datasets: A Dual-Domain Evaluation," *sinkron*, vol. 9, no. 3, pp. 1094–1107, Jul. 2025, doi: 10.33395/sinkron.v9i3.15029.
- [8] A. Yusran Siregar and A. Setyo Arifin, "Enhancing XGBoost Classification with SVM-SMOTE & EasyEnsemble for Imbalanced Telemedicine Sentiment Data," *Indonesian Journal of Social Technology*, vol. 5, no. 10, p. 3893, 2024, [Online]. Available: <http://jst.publikasiindonesia.id/>
- [9] W. O. Simanjuntak, A. Bijaksana, P. Negara, and R. Septriana, "Perbandingan Algoritma Logistic Regression dan Random Forest (Studi Kasus: Klasifikasi Emosi Tweet) Comparison Of Logistic Regression And Random Forest Algorithms (Case Study: Tweet Emotion Classification)," 2023, doi: 10.26418/juara.v2i1.69682.
- [10] Deden Moh Alfiansyah, Wiilys, Lila Setiyani, Devi Fajar Wati, and Dedih, "Pengembangan Chatbot Berbasis Web untuk Layanan Informasi di Horizon University," *bit-Tech*, vol. 7, no. 3, pp. 1068–1077, Apr. 2025, doi: 10.32877/bt.v7i3.2318.
- [11] R. Budiarto, P. Siber, and S. Negara, "Text Preprocessing for Optimal Accuracy in Indonesian Sentiment Analysis Using a Deep Learning Model with Word Embedding," 2021. [Online]. Available: <https://www.researchgate.net/publication/365799401>
- [12] Rianto, A. B. Mutiara, E. P. Wibowo, and P. I. Santosa, "Improving the accuracy of text classification using stemming method, a case of non-formal Indonesian conversation," *J Big Data*, vol. 8, no. 1, Dec. 2021, doi: 10.1186/s40537-021-00413-1.
- [13] O. I. Gifari, M. Adha, I. Rifky Hendrawan, F. Freddy, and S. Durrand, "Analisis Sentimen Review Film Menggunakan TF-IDF dan Support Vector Machine," *JIFOTECH (JOURNAL OF INFORMATION TECHNOLOGY)*, vol. 2, no. 1, 2022.
- [14] F. Suandi *et al.*, "Enhancing Sentiment Analysis Performance Using SMOTE and Majority Voting in Machine Learning Algorithms," 2024, pp. 126–138. doi: 10.2991/978-94-6463-620-8\_10.
- [15] D. Dey *et al.*, "The proper application of logistic regression model in complex survey data: a systematic review," *BMC Med Res Methodol*, vol. 25, no. 1, Dec. 2025, doi: 10.1186/s12874-024-02454-5.
- [16] T. C. Attanasi Emil D. and Coburn, "Random Forest," in *Encyclopedia of Mathematical Geosciences*, Q. and M. J. and A. F. Daya Sagar B. S. and Cheng, Ed., Cham: Springer International Publishing, 2023, pp. 1182–1185. doi: 10.1007/978-3-030-85040-1\_265.
- [17] T. Greyling and S. Rossouw, "Development and Validation of a Real-Time Happiness Index Using Google Trends™," *J Happiness Stud*, vol. 26, no. 3, Mar. 2025, doi: 10.1007/s10902-025-00881-9.
- [18] B. Tri Putra, E. Yulianingsih, S. Informasi, S. Teknologi, and U. Bina Darma, "Analisis Tingkat Akurasi Prediksi Gejala COVID-19 Dengan Menggunakan Metode Logistic Regression dan Support Vector Machine," 2023, [Online]. Available: <http://github.com/nshomron/covidpred>.