Idealis: Indonesia Journal Information System
Volume 9, Nomor 2, Juli 2026
ISSN 2684-7280 (*online*)
Halaman 407-416
available online at http://jom.fti.budiluhur.ac.id/index.php/IDEALIS/index

Sentiment Analysis of Indonesian Mobile Legends: Bang Bang Reviews Using Random Forest, KNN, TF-IDF, and SMOTE

Ahmad Alif Candra Selamet[1*], Pratomo Setiaji[2], Wiwit Agus Triyanto[3]

[1,2,3]Sistem Informasi, Fakultas Teknik, Universitas Muria Kudus, Kudus, Indonesia
Email: [1*]alifcandra097@gmail.com, [2]pratomo.setiaji@umk.ac.id, [3]at.wiwit@umk.ac.id
(*: *corresponding author*)

Abstract - *Mobile Legends: Bang Bang (MLBB) is one of the most popular mobile games, generating a large number of user reviews on the Google Play Store. The large volume of reviews makes manual sentiment analysis impractical, requiring an automated machine learning approach. This study compares the performance of Random Forest and K-Nearest Neighbors (KNN) for classifying sentiment in Indonesian-language MLBB reviews. A total of 6,994 reviews were obtained through web scraping and preprocessing. The proposed framework includes text preprocessing, rating-based sentiment labeling, TF-IDF feature extraction, SMOTE-based class balancing, model training, and evaluation using Accuracy, Precision, Recall, F1-score, Macro-F1, Weighted-F1, and 5-fold cross-validation. Random Forest with SMOTE achieved the best performance, with an Accuracy of 84.0% and a Macro-F1 score of 80.8%, outperforming KNN with SMOTE, which achieved 73.8% Accuracy and 71.9% Macro-F1. The ablation study demonstrates that the effectiveness of SMOTE is model-dependent, improving Random Forest but degrading KNN performance. This study provides empirical evidence of SMOTE impact on different classifiers and employs cross-validation-based K selection to prevent test data leakage.*

Keywords*: K-Nearest Neighbors, Mobile Legends:Bang Bang, Random Forest, Sentiment Analysis, SMOTE.*

Abstrak - Mobile Legends: Bang Bang (MLBB) merupakan salah satu permainan mobile yang menghasilkan volume ulasan pengguna sangat besar di Google Play Store. Banyaknya ulasan tersebut menyulitkan proses identifikasi sentimen apabila dilakukan secara manual, sehingga diperlukan pendekatan otomatis berbasis *machine learning*. Penelitian ini bertujuan membandingkan kinerja algoritma Random Forest dan K-Nearest Neighbors (KNN) dalam mengklasifikasikan sentimen ulasan MLBB berbahasa Indonesia. Data penelitian diperoleh melalui proses web scraping dari Google Play Store dan, setelah melalui tahap filtrasi, menghasilkan 6.994 ulasan yang layak dianalisis. Tahapan penelitian meliputi text preprocessing, pelabelan sentimen berbasis rating, ekstraksi fitur menggunakan TF-IDF, penyeimbangan kelas dengan SMOTE, pembangunan model klasifikasi, serta evaluasi menggunakan metrik *Accuracy*, *Precision*, *Recall*, *F1-score*, *Macro-F1*, *Weighted-F1*, dan validasi silang 5-*fold*. Hasil pengujian menunjukkan bahwa Random Forest dengan SMOTE memberikan kinerja terbaik pada skenario penelitian ini dengan *Accuracy* sebesar 84,0% dan *Macro*-F1 sebesar 80,8%, sedangkan KNN dengan SMOTE memperoleh *Accuracy* 73,8% dan *Macro-F1* 71,9%. Eksperimen ablasi mengindikasikan bahwa dampak penerapan SMOTE bergantung pada karakteristik algoritma yang digunakan, yaitu mampu meningkatkan performa Random Forest, namun menurunkan performa KNN. Kontribusi penelitian ini terletak pada penyajian bukti empiris mengenai pengaruh SMOTE terhadap kedua algoritma serta penerapan pemilihan nilai K berbasis *cross-validation* untuk meminimalkan risiko kebocoran data uji.

Kata Kunci: Analisis Sentimen, K-Nearest Neighbors, Mobile Legends: Bang Bang, Random Forest, SMOTE.

1. PENDAHULUAN

Industri *game mobile* di Indonesia terus menunjukkan pertumbuhan yang masif, di mana permainan *Multiplayer Online Battle Arena* (MOBA) tetap mendominasi pasar digital. Berdasarkan laporan Peta Ekosistem Industri Gim Indonesia 2024 yang dikutip oleh Kompas, Indonesia memiliki sekitar 154,9 juta pemain *game* [1]. Selanjutnya, berdasarkan survei APJII 2025 yang melibatkan 8.700 responden, 48,99% responden yang bermain *game online* menyatakan bahwa Mobile Legends: Bang Bang (MLBB) merupakan *game* yang paling sering dimainkan. [2]. Skala global *game* ini juga terlihat pada tahun 2026, di mana tercatat sebanyak 107 negara dan wilayah mendaftarkan tim mereka untuk mengikuti kompetisi internasional MLBB [3]. Tingginya interaksi pengguna ini mengakumulasi jumlah ulasan yang sangat masif di Google Play Store, yang mencerminkan pengalaman langsung pengguna terhadap fitur dan performa aplikasi [4], [5].

Terlepas dari popularitasnya, MLBB tidak lepas dari berbagai keluhan pengguna, salah satunya ketidakseimbangan sistem *matchmaking* yang menempatkan pemain dalam tim dengan kapasitas yang tidak setara, sehingga berpotensi memicu perilaku negatif antarpemain [5], [6]. Keluhan-keluhan tersebut terakumulasi dalam ribuan ulasan teks tidak terstruktur di Google Play Store, sehingga timbul permasalahan bagaimana mengidentifikasi dan mengklasifikasikan sentimen pengguna secara akurat dan konsisten dari volume data yang besar tersebut, mengingat pengkajian secara manual tidak efisien untuk diterapkan. Oleh karena itu, dibutuhkan pendekatan otomatis berbasis *Natural Language Processing* (NLP) dan *Machine learning* untuk menjawab permasalahan tersebut [4], [7].

Beberapa penelitian sebelumnya telah mengevaluasi algoritma ini dalam berbagai konteks. Sanjaya dkk. (2026) membandingkan Naïve Bayes, KNN, dan SVM pada ulasan MLBB dengan hasil SVM sebagai model terbaik pada akurasi 79,88% [4]. Handoko dkk. (2024) menggunakan Random Forest dan KNN untuk ulasan

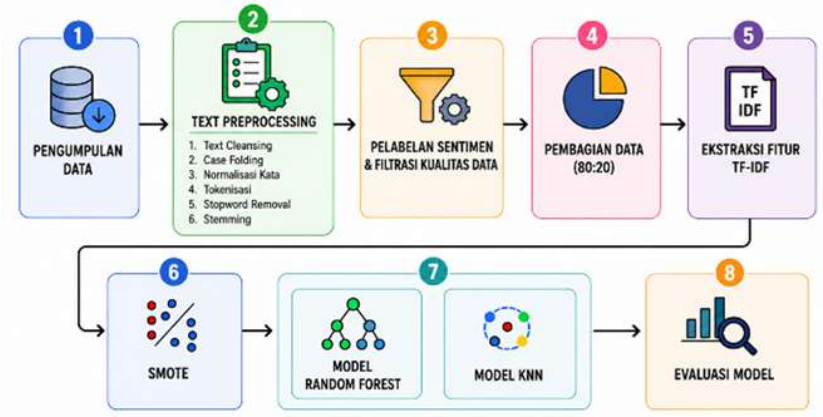
Idealis: Indonesia Journal Information System
Volume 9, Nomor 2, Juli 2026
ISSN 2684-7280 (*online*)
Halaman 407-416
available online at http://jom.fti.budiluhur.ac.id/index.php/IDEALIS/index

maskapai penerbangan dengan hasil Random Forest unggul pada akurasi 83% [8]. Penelitian lain oleh Ulfa dkk. (2024) menerapkan KNN pada aplikasi identitas digital (IKD) dan mencapai akurasi 83% dengan nilai K=13 [9]. Ma'arif dkk. (2026) membandingkan beberapa algoritma klasifikasi pada ulasan MLBB dengan Logistic Regression sebagai model terbaik pada akurasi 79,00% [5]. Adi dkk. 2025 menerapkan Naïve Bayes dengan pelabelan berbasis leksikon InSet pada ulasan MLBB dan mencapai akurasi 71,05% [6]. Meskipun demikian, penelitian-penelitian tersebut berfokus pada perbandingan algoritma atau pemilihan model terbaik tanpa mengevaluasi secara sistematis pengaruh teknik penyeimbangan data terhadap masing-masing algoritma. Selain itu, studi pada ulasan MLBB belum menunjukkan bagaimana penerapan SMOTE memengaruhi performa Random Forest dan KNN melalui eksperimen yang membandingkan kondisi sebelum dan sesudah *oversampling*.

Pemilihan Random Forest dan KNN didasarkan pada karakteristik kedua algoritma yang merepresentasikan dua pendekatan berbeda dalam klasifikasi teks, Random Forest sebagai model *ensemble* berbasis pohon keputusan yang *robust* terhadap fitur berdimensi tinggi [10], dan KNN sebagai model berbasis kemiripan *instance* yang sensitif terhadap struktur ruang fitur [11]. Perbandingan kedua pendekatan ini memberikan perspektif komplementer mengenai karakteristik data ulasan MLBB, sementara algoritma lain seperti SVM dan Naïve Bayes yang telah banyak dievaluasi pada studi-studi sebelumnya tidak menjadi fokus utama penelitian ini. Berdasarkan hal tersebut, kajian ini berfokus untuk menganalisis perbandingan Random Forest dan K-Nearest Neighbors dalam mengelompokkan polaritas ulasan MLBB menjadi dua kelas yakni positif dan negatif [4], [8]. Kontribusi utama penelitian ini terletak pada bukti empiris melalui eksperimen ablasi bahwa efektivitas SMOTE bersifat *model-dependent*, serta penerapan koreksi metodologis pada pemilihan *hyperparameter* K melalui *cross-validation* untuk mencegah kebocoran data uji. Hasil penelitian ini diharapkan dapat memberikan gambaran yang lebih objektif mengenai persepsi pengguna terhadap kualitas layanan MLBB, sekaligus menjadi referensi bagi pengembang dalam memprioritaskan aspek yang perlu diperbaiki berdasarkan data sentimen pengguna secara sistematis.

2. METODE PENELITIAN

Studi ini menerapkan metode kuantitatif yang berlandaskan *machine learning* guna mengklasifikasikan sentimen ulasan pengguna Mobile Legends: Bang Bang (MLBB). Tahapan penelitian disusun secara berurutan mulai dari pengumpulan data hingga evaluasi model untuk menghasilkan proses klasifikasi yang sistematis.



Gambar 1. Metodologi Penelitian

Gambar 1 mengilustrasikan alur penelitian secara keseluruhan. Proses penelitian diawali dengan pengumpulan data, kemudian dilanjutkan dengan pra-pemrosesan teks, pelabelan dan filtrasi kualitas data, pembagian data latih dan data uji, ekstraksi fitur menggunakan TF-IDF, penyeimbangan data menggunakan SMOTE pada data latih, pelatihan model Random Forest dan K-Nearest Neighbors (KNN), serta diakhiri dengan evaluasi performa model. Setiap tahapan tersebut dijelaskan lebih rinci pada subbab berikutnya.

2.1. Sumber dan Pengumpulan Data

Dataset didapatkan dengan teknik *web scraping* melalui Google Play Store dari kolom ulasan aplikasi MLBB [12], [13]. Langkah ini menghasilkan 9.189 data ulasan mentah terbaru, dengan periode publikasi ulasan mulai 19 Mei 2026 sampai 27 Mei 2026. Dataset mencakup empat atribut utama yaitu identitas pengguna (*user*), peringkat bintang (rating), konten ulasan (ulasan), dan waktu pengunggahan (tanggal). Seluruh data yang dikumpulkan bersifat publik dan dapat diakses oleh siapa pun melalui platform Google Play Store. Atribut identitas pengguna hanya dimanfaatkan sebagai penanda unik untuk keperluan pengelolaan data selama proses penelitian, tidak diikutsertakan sebagai fitur dalam proses klasifikasi sentimen, dan tidak disertakan dalam data maupun hasil yang

Idealis: Indonesia Journal Information System
Volume 9, Nomor 2, Juli 2026
ISSN 2684-7280 (*online*)
Halaman 407-416
available online at http://jom.fti.budiluhur.ac.id/index.php/IDEALIS/index

dipublikasikan. Penggunaan ulasan dari platform distribusi aplikasi merupakan sumber data primer yang krusial karena merefleksikan pengalaman pengguna secara langsung dan spontan.

2.2. Pra-Pemrosesan Teks (*Text Preprocessing*)

Text Preprocessing merupakan langkah fundamental dalam Natural Language Processing (NLP) untuk mengubah dokumen mentah yang acak menjadi data yang layak diproses [14], [15]. Tahapan ini sangat berdampak bagi performa model klasifikasi, terutama untuk menangani data ulasan yang memiliki banyak *noise* [14].

a. *Text Cleansing*
Text cleansing ialah fase penyaringan teks dari elemen yang kurang relevan dan berpotensi menjadi *noise* dalam analisis sentimen [16]. Pada penelitian ini, proses *cleansing* meliputi penghapusan URL, tanda baca, angka, karakter non-ASCII, karakter berulang, dan spasi berlebih, sebagaimana ditunjukkan pada baris "*Text Cleansing*" pada Tabel 1.

b. *Case folding*
Case folding merupakan proses standardisasi penulisan huruf dengan mengonversi semua karakter menjadi huruf kecil (lowercase). Ini bertujuan mengurangi ketidakseragaman representasi kata akibat penggunaan huruf kapital [17]. Hasil ditunjukkan pada baris "*Case Folding*" pada Tabel 1.

c. Normalisasi Kata
Normalisasi kata dilakukan guna mengonversi diksi tidak baku, singkatan, dan kata alay ke dalam format yang lebih standar. Proses ini bertujuan menyeragamkan variasi penulisan kata yang memiliki makna sama [18]. *Output* ditunjukkan pada baris "Normalisasi Kata" pada Tabel 1.

d. Tokenisasi
Tokenisasi merupakan tahapan memisahkan kalimat menjadi satuan kata (token) yang akan digunakan pada tahap analisis berikutnya [19]. Pada penelitian ini, tokenisasi dilakukan menggunakan NLTK RegexpTokenizer.Hasil ditunjukkan pada baris "Tokenisasi" pada Tabel 1.

e. *Stopword removal*
Eliminasi stopword dilakukan guna menyaring kosakata yang memiliki nilai informasi rendah dalam analisis sentimen [20]. Namun, kata negasi serta *intensifier* tetap dipertahankan karena dapat memengaruhi polaritas sentimen. *Output* ditunjukkan pada baris "*Stopword Removal*" pada Tabel 1.

f. *Stemming*
Stemming ialah fase mengubah kata berimbuhan menjadi kata dasarnya [21]. Dalam kajian ini, *stemming* diimplementasikan menggunakan algoritma Enhanced Confix Stripping pada pustaka Sastrawi. Hasil akhir ditunjukkan pada baris "*Stemming*" pada Tabel 1.

Tabel 1. *Text Preprocessing*

Tahapan	Sebelum	Sesudah
Text Cleansing	Moonton Gak sehat saya dikasih tim *dark system* kalau musuhnya jago jago. jadi saya gak bisa tahan dan memutuskan untuk pensiun dari Ml karena bikin saya mancing emosi gara gara dikasih kalah minta ampun 😭😭😭😭	Moonton Gak sehat saya dikasih tim *dark system* kalau musuhnya jago jago jadi saya gak bisa tahan dan memutuskan untuk pensiun dari Ml karena bikin saya mancing emosi gara gara dikasih kalah minta ampun
Case folding	Moonton Gak sehat saya dikasih tim *dark system* kalau musuhnya jago jago jadi saya gak bisa tahan dan memutuskan untuk pensiun dari Ml karena bikin saya mancing emosi gara gara dikasih kalah minta ampun	moonton gak sehat saya dikasih tim *dark system* kalau musuhnya jago jago jadi saya gak bisa tahan dan memutuskan untuk pensiun dari ml karena bikin saya mancing emosi gara gara dikasih kalah minta ampun
Normalisasi Kata	moonton gak sehat saya dikasih tim *dark system* kalau musuhnya jago jago jadi saya gak bisa tahan dan memutuskan untuk pensiun dari ml karena bikin saya mancing emosi gara gara dikasih kalah minta ampun	moonton tidak sehat saya dikasih tim *dark system* kalau musuhnya jago jago jadi saya tidak bisa tahan dan memutuskan untuk pensiun dari *mobile* legends karena membuat saya mancing emosi gara gara dikasih kalah minta ampun
Tokenisasi	moonton tidak sehat saya dikasih tim *dark system* kalau musuhnya jago jago jadi saya tidak bisa tahan dan memutuskan untuk pensiun dari mobile legends karena membuat saya mancing emosi gara gara dikasih kalah minta ampun	['moonton', 'tidak', 'sehat', 'saya', 'dikasih', 'tim', '*dark*', '*system*', 'kalau', 'musuhnya', 'jago', 'jago', 'jadi', 'saya', 'tidak', 'bisa', 'tahan', 'dan', 'memutuskan', 'untuk', 'pensiun', 'dari', '*mobile*', *'legends'*, 'karena', 'membuat', 'saya', 'mancing', 'emosi', 'gara', 'gara', 'dikasih', 'kalah', 'minta', 'ampun']

Idealis: Indonesia Journal Information System
Volume 9, Nomor 2, Juli 2026
ISSN 2684-7280 (*online*)
Halaman 407-416
available online at http://jom.fti.budiluhur.ac.id/index.php/IDEALIS/index

Tahapan	Sebelum	Sesudah
Stopword removal	['moonton', 'tidak', 'sehat', 'saya', 'dikasih', 'tim', *'dark'*, *'system*', 'kalau', 'musuhnya', 'jago', 'jago', 'jadi', 'saya', 'tidak', 'bisa', 'tahan', 'dan', 'memutuskan', 'untuk', 'pensiun', 'dari', '*mobile*', '*legends*', 'karena', 'membuat', 'saya', 'mancing', 'emosi', 'gara', 'gara', 'dikasih', 'kalah', 'minta', 'ampun']	['tidak', 'sehat', 'dikasih', 'tim', '*dark*', '*system*', 'musuhnya', 'jago', 'jago', 'tidak', 'tahan', 'memutuskan', 'pensiun', 'mancing', 'emosi', 'gara', 'gara', 'dikasih', 'kalah', 'ampun']
Stemming	['tidak', 'sehat', 'dikasih', 'tim', '*dark*', '*system*', 'musuhnya', 'jago', 'jago', 'tidak', 'tahan', 'memutuskan', 'pensiun', 'mancing', 'emosi', 'gara', 'gara', 'dikasih', 'kalah', 'ampun']	tidak sehat kasih tim *dark system* musuh jago jago tidak tahan putus pensiun mancing emosi gara gara kasih kalah ampun

2.3. Pelabelan dan Filtrasi Kualitas

Pelabelan menggunakan pendekatan berbasis rating (*lexicon-free labeling*). Rating 4–5 bintang dikategorikan ke dalam kategori positif dan rating 1–2 bintang ke dalam kategori negatif [9], [14]. Ulasan dengan rating 3 bintang dieliminasi karena bersifat ambigu [9], [14]. Metode ini sejalan dengan kebiasaan yang umum pada kajian analisis sentimen berbasis ulasan platform daring [9], [12].

Dua tahap filtrasi tambahan diterapkan setelah pelabelan. Pertama, ulasan kosong atau bernilai NaN pasca *preprocessing* dihapus. Kedua, filter rasio bahasa Indonesia diterapkan menggunakan pustaka WordFreq. Ulasan dengan rasio kata Indonesia di bawah 40% dari total token dieliminasi. Kosakata acuan mencakup 50.000 kata Indonesia paling umum yang diperkaya dengan 30 istilah domain gaming (*lag, rank, matchmaking, hero, mythic*, dan sebagainya).

2.4. Pembagian data

Dataset dipartisi dengan rasio 80:20 (subset latih : subset uji) menerapkan train_test_split dengan parameter stratify=y guna menjaga proporsi pada kedua subset, serta random_state=42 untuk reproduktibilitas eksperimen. Rasio itu dipilih karena umum digunakan pada penelitian klasifikasi sentimen berbasis *machine learning* untuk menjaga keseimbangan antara ukuran data latih yang memadai bagi pembelajaran model dan data uji yang representatif bagi evaluasi [10].

2.5. Ekstraksi Fitur TF-IDF

Pembobotan numerik teks dilakukan memakai *Term Frequency–Inverse Document Frequency* (TF-IDF), menjadi salah satu metode pembobotan fitur yang umum diterapkan pada analisis sentimen dan klasifikasi teks karena mampu merepresentasikan tingkat kepentingan suatu *term* dalam dokumen dan korpus secara efektif. Konfigurasi yang digunakan meliputi max_features = 10.000, ngram_range = (1,2) untuk menangkap konteks *unigram* dan *bigram*, sublinear_tf = True untuk menerapkan transformasi logaritmik pada frekuensi *term*, serta min_df = 2 untuk menghilangkan *term* yang hanya muncul pada satu dokumen. Proses *fitting* TF-IDF dilakukan khusus pada subset latih, selanjutnya model yang terbentuk diterapkan untuk mengonversi subset uji guna mencegah terjadinya *data leakage* yang dapat menyebabkan estimasi performa model menjadi terlalu optimistis [22], [23].

2.6. SMOTE

SMOTE menyeimbangkan dataset melalui penciptaan sampel sintetis berbasis interpolasi antar data minoritas, bukan sekadar duplikasi [7]. Konfigurasi menggunakan random_state = 42 untuk menjamin konsistensi dan reproduktibilitas hasil sintesis data. Fungsi fit_resample diterapkan secara eksklusif hanya pada data latih guna menghindari risiko kebocoran informasi dari data uji. Pada tahap validasi silang (*cross-validation*), SMOTE turut diintegrasikan ke dalam satu kesatuan pipeline bersama model klasifikasi, sehingga proses *oversampling* dilakukan ulang secara independen pada setiap *fold* pelatihan, guna mencegah kebocoran informasi antar *fold*. Tahapan ini bertujuan meningkatkan sensitivitas model dalam mengenali pola kategori sentimen yang berjumlah sedikit [16]. Untuk mengevaluasi pengaruh SMOTE secara objektif, dilakukan pula eksperimen ablasi yang membandingkan performa model dengan dan tanpa penerapan SMOTE.

2.7. Pemodelan

a. Random Forest
Random Forest adalah algoritma *ensemble* yang menciptakan sekumpulan pohon keputusan secara paralel pada segmen data *bootstrap* [10]. Prediksi akhir ditentukan melalui voting mayoritas seluruh pohon.

Idealis: Indonesia Journal Information System
Volume 9, Nomor 2, Juli 2026
ISSN 2684-7280 (*online*)
Halaman 407-416
available online at http://jom.fti.budiluhur.ac.id/index.php/IDEALIS/index

Konfigurasi yang digunakan: n_estimators = 200 pohon, min_samples_leaf = 3, random_state = 42, dan n_jobs = -1 untuk komputasi paralel.

b. K-Nearest Neighbors
KNN mengkategorikan *instance* baru berlandaskan label dominan dari K tetangga terdekat pada *space* fitur [11]. Konfigurasi yang digunakan: weights = 'distance' (tetangga lebih dekat diberi bobot lebih besar) dan metric = 'cosine' (lebih sesuai untuk ruang TF-IDF berdimensi tinggi dibanding *euclidean distance*). Nilai K optimal ditentukan melalui 5-fold *stratified cross-validation* pada subset data latih menggunakan metrik *Macro-F1*, dengan proses SMOTE diterapkan ulang pada setiap *fold* pelatihan (melalui *pipeline*) untuk mencegah kebocoran data. Kandidat nilai K yang diuji meliputi 3, 5, 7, 9, 11, dan 13.

2.8. Evaluasi Model

Evaluasi model dilakukan dengan memanfaatkan *confusion matrix* yang terbagi atas *True Positive* (TP), *True Negative* (TN), *False positive* (FP), dan *False negative* (FN), sebagai dasar penghitungan metrik evaluasi [24], yaitu *Accuracy*, *Precision*, *Recall*, dan *F1-score* per kelas, serta *Macro-F1* dan *Weighted-F1*. Mengingat dataset penelitian ini memiliki distribusi kelas yang tidak seimbang, tiga metrik utama digunakan untuk menilai dan membandingkan performa model, yaitu *Macro-F1* (menilai keseimbangan performa antar kelas), *Accuracy* (mengukur ketepatan klasifikasi secara keseluruhan), dan *Weighted-F1* (mempertimbangkan proporsi distribusi kelas) [16]. Sementara itu, *Precision*, *Recall*, dan *F1-score* per kelas digunakan sebagai metrik pendukung untuk menganalisis karakteristik prediksi masing-masing model, termasuk *trade-off* antara *false positive* dan *false negative* yang selanjutnya dianalisis melalui *confusion matrix*[15], [18]. Selain evaluasi pada data uji, validasi silang (*5-fold cross-validation*) juga dilakukan pada konfigurasi model final untuk menguji stabilitas performa [23].

3. HASIL DAN PEMBAHASAN

Hasil penelitian dijabarkan melalui delapan sub-bab berikut, mencakup karakteristik dataset, proses ekstraksi fitur, hingga evaluasi dan analisis performa model klasifikasi.

3.1. Statistik Dataset dan Distribusi Kelas

Dataset mentah sebanyak 9.189 ulasan mengalami reduksi bertahap hingga menghasilkan dataset final sebanyak 6.994 ulasan yang siap digunakan dalam proses pemodelan. Tabel 2 merangkum perubahan jumlah data pada setiap tahap filtrasi.

Tabel 2. Jumlah Data pada Setiap Tahap Filtrasi

Tahap Filtrasi	Jumlah Data	Selisih
Data mentah awal	9.189	0
Setelah eliminasi rating 3 (ambigu)	8.796	393
Setelah hapus ulasan kosong pasca *preprocessing*	8.738	58
Setelah filter bahasa Indonesia (≥40%)	6.994	1.744

Reduksi terbesar terjadi pada tahap filter bahasa Indonesia, di mana 1.744 ulasan (19,0% dari data) dieliminasi. Angka ini mencerminkan karakteristik nyata ulasan *game mobile* yang sering mengandung campuran bahasa, tidak hanya istilah gaming berbahasa Inggris seperti "gg", "*noob*", dan "*afk*", tetapi juga ekspresi berbahasa daerah seperti bahasa jawa dan sunda yang umumnya digunakan pengguna untuk mengungkapkan kekecewaan atau kemarahan. Filter ini menjaga konsistensi linguistik dataset, namun ulasan berbahasa daerah yang bermuatan sentimen negatif kuat ikut tereliminasi, ini berpotensi memengaruhi representasi kelas negatif dalam dataset final.

3.2. Hasil Ekstraksi Fitur TF-IDF

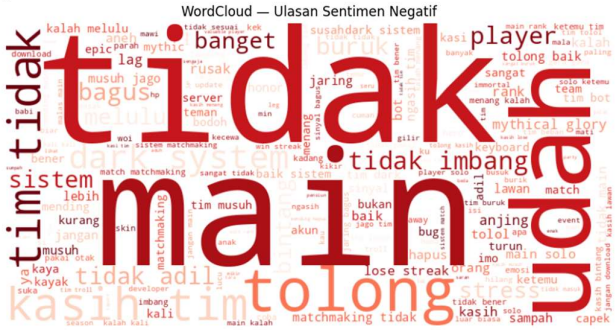
Proses TF-IDF menghasilkan 8.106 fitur aktif dari batas maksimum 10.000 yang dialokasikan. Angka ini mengindikasikan bahwa *vocabulary* efektif dataset pasca *stemming* memang terbatas pada 8.106 *term* setelah diterapkan filter min_df = 2, kondisi yang positif karena menunjukkan tidak ada fitur *noise* yang dipaksakan masuk ke ruang representasi. Pembagian data menghasilkan 5.595 ulasan latih dan 1.399 ulasan uji dengan proporsi kelas yang terjaga melalui *stratified splitting*. Ruang fitur sebesar 8.106 dimensi ini selanjutnya menjadi representasi vektor yang digunakan oleh kedua model dalam proses klasifikasi.

3.3. Analisis WordCloud

Penggambaran WordCloud dimanfaatkan untuk menyajikan ilustrasi deskriptif tentang kosakata yang paling dominan di setiap kategori sentimen. *Term* dominan pada ulasan negatif meliputi "tidak", "main", "kasih", "tim", dan "tolong". Kemunculan kata "tolong" yang mencolok mengindikasikan banyaknya ulasan berisi permintaan

Idealis: Indonesia Journal Information System
Volume 9, Nomor 2, Juli 2026
ISSN 2684-7280 (*online*)
Halaman 407-416
available online at http://jom.fti.budiluhur.ac.id/index.php/IDEALIS/index

perbaikan kepada *developer*, berbeda secara semantis dari sekadar ekspresi kekecewaan. Frasa "tidak imbang" dan "sistem *matchmaking*" yang menonjol mengarah pada keluhan spesifik mengenai sistem penjodohan tim, sebagaimana ditunjukkan pada Gambar 2.



Gambar 2. WordCloud Negatif

Term dominan pada ulasan positif meliputi "bagus", "suka", "seru", "banget", dan "tidak". Kemunculan "tidak" pada kelas positif disebabkan oleh frasa yang mengandung negasi pembentuk sentimen positif seperti "tidak membosankan", "tidak kecewa", dan "tidak ada masalah". Temuan ini menegaskan keputusan mempertahankan kata negasi dalam daftar stopword, sebagaimana ditunjukkan pada Gambar 3.

Gambar 3. WordCloud Positif

3.4. Penentuan Nilai K Optimal untuk KNN

Sebelum membandingkan performa Random Forest dan KNN, dilakukan terlebih dahulu proses tuning *hyperparameter* untuk menentukan nilai K optimal pada algoritma KNN. Ringkasan hasil 5-*fold cross-validation* ditampilkan pada Tabel 3.

Tabel 3. Hasil *Cross-validation* Pemilihan Nilai K

K	Mean *Macro-F1* (CV)	Std *Macro-F1* (CV)
3	69,62%	1,47%
5	70,19%	1,30%
7	71,08%	1,54%
9	71,51%	1,15%
11	**72,09%**	**1,00%**
13	71,98%	1,18%

Tabel 3 menunjukkan bahwa nilai K=11 memperoleh nilai rata-rata Macro-F1 tertinggi, yaitu 72,09%, dengan standar deviasi 1%. Hal ini mengindikasikan bahwa K=11 memberikan performa yang baik dan relatif stabil pada setiap *fold* validasi. Berdasarkan hasil tersebut, nilai K=11 dipilih sebagai *hyperparameter* KNN dan digunakan pada seluruh eksperimen selanjutnya.

3.5. Perbandingan Performa Random Forest dan KNN (*Ablation Study* SMOTE)

Perbandingan performa kedua model dilakukan melalui eksperimen ablasi yang membandingkan hasil klasifikasi dengan dan tanpa penerapan SMOTE, sebagaimana disajikan pada Tabel 4. Konfigurasi KNN pada evaluasi ini menggunakan K=11, sesuai hasil *cross-validation*. Nilai K yang sama diterapkan

Idealis: Indonesia Journal Information System
Volume 9, Nomor 2, Juli 2026
ISSN 2684-7280 (*online*)
Halaman 407-416
available online at http://jom.fti.budiluhur.ac.id/index.php/IDEALIS/index

pada kondisi dengan maupun tanpa SMOTE agar perbandingan hanya mencerminkan efek penyeimbangan data, bukan efek perbedaan *hyperparameter*.

Tabel 4. Performa Model dengan dan tanpa SMOTE (*Ablation Study*)

Model	*accuracy*	*Precision* (positif)	*Recall* (positif)	F1-*Score* (positif)	*Precision* (negatif)	*Recall* (negatif)	F1-*Score* (negatif)	Macro-F1	*Weighted-F1*
RF	84,5%	86,6%	58,7%	70,0%	83,9%	96,0%	89,5%	79,8%	83,5%
RF - SMOTE	**84,0%**	**75,9%**	**70,3%**	**73,0%**	**87,2%**	**90,1%**	**88,6%**	**80,8%**	**83,8%**
KNN	83,1%	77,9%	63,1%	69,7%	84,9%	92,0%	88,3%	79,0%	82,6%
KNN - SMOTE	73,8%	55,3%	78,0%	64,7%	88,0%	71,9%	79,1%	71,9%	74,7%

Merujuk Tabel 4, pada konfigurasi final penelitian ini dengan penerapan SMOTE, Random Forest (RF) secara umum menunjukkan performa yang lebih unggul dibandingkan KNN, dengan perbedaan terbesar terlihat pada *Precision* kelas positif (75,9% dibandingkan 55,3%) dan *Macro-F1* (80,8% dibandingkan 71,9%). KNN unggul tipis pada *Recall* kelas positif (78,0% dibandingkan 70,3%) dan *Precision* kelas negatif (88,0% dibandingkan 87,2%), yang menunjukkan bahwa KNN cenderung lebih agresif dalam mengklasifikasikan ulasan sebagai positif, namun menghasilkan jumlah *false positive* yang lebih tinggi.

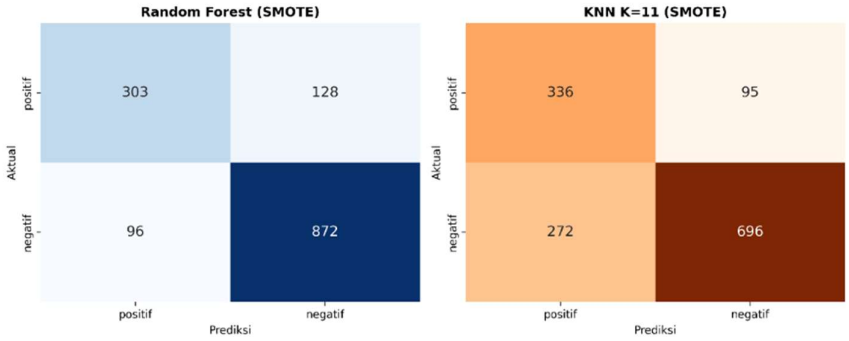
Pengaruh SMOTE juga terlihat berbeda pada masing-masing algoritma (*model-dependent*), sehingga SMOTE tidak dapat dianggap sebagai solusi universal untuk mengatasi ketidakseimbangan kelas. Pada Random Forest, penerapan SMOTE mampu meningkatkan keseimbangan *Recall* antar kelas. *Recall* kelas positif meningkat dari 58,7% menjadi 70,3%, sementara *Recall* kelas negatif mengalami penurunan yang relatif kecil dari 96,0% menjadi 90,1%. Kondisi tersebut menyebabkan nilai *Macro-F1* meningkat dari 79,8% menjadi 80,8%, meskipun *Accuracy* sedikit menurun dari 84,5% menjadi 84,0% akibat adanya *trade-off* antara *Precision* dan *Recall*. Sebaliknya, pada KNN, penerapan SMOTE justru menyebabkan penurunan performa pada sebagian besar metrik. *Accuracy* menurun dari 83,1% menjadi 73,8%, *Recall* kelas negatif turun dari 92,0% menjadi 71,9%, dan *Macro-F1* mengalami penurunan dari 79,0% menjadi 71,9%.

Berdasarkan hasil eksperimen ablasi tersebut, Random Forest dengan SMOTE ditetapkan sebagai konfigurasi model utama untuk tahap evaluasi lanjutan. Konfigurasi ini dipilih karena merupakan satu-satunya kombinasi yang menunjukkan peningkatan *Macro-F1* setelah penerapan SMOTE serta menghasilkan nilai *Macro-F1* dan *Weighted-F1* tertinggi dibandingkan konfigurasi lainnya.

3.6. Analisis *Confusion Matrix*

Evaluasi lebih lanjut terhadap kesalahan klasifikasi kedua model dilakukan melalui analisis *confusion matrix* pada konfigurasi akhir, yaitu Random Forest dan KNN (K=11) yang masing-masing telah menerapkan SMOTE. Kedua model menunjukkan kecenderungan kesalahan klasifikasi yang berlawanan arah. Random Forest menghasilkan lebih banyak *false negative* (128) dibandingkan *false positive* (96), yang mengindikasikan model cenderung lebih konservatif dalam mengklasifikasikan ulasan sebagai positif, sehingga sebagian ulasan positif justru terklasifikasi sebagai negatif. Sebaliknya, KNN menghasilkan *false positive* (272) yang jauh lebih besar dibandingkan *false negative* (95), yang mengindikasikan model lebih agresif dalam memprediksi kelas positif. Pola ini konsisten dengan nilai *Recall* kelas positif KNN yang tinggi (78,0%) namun *Precision* kelas positif yang rendah (55,3%) sebagaimana dilaporkan pada Tabel 4, karena model cenderung memprediksi kelas positif secara berlebihan sehingga menghasilkan banyak kesalahan pada kelas negatif.

Secara keseluruhan, total kesalahan klasifikasi Random Forest lebih rendah dibandingkan KNN, yang menegaskan superioritas Random Forest dalam meminimalkan kesalahan klasifikasi pada dataset ulasan MLBB yang digunakan pada penelitian ini. Visualisasi *confusion matrix* kedua model ditunjukkan pada Gambar 4.



Gambar 4. *Confusion matrix* RF dan KNN menggunakan SMOTE

Idealis: Indonesia Journal Information System
Volume 9, Nomor 2, Juli 2026
ISSN 2684-7280 (*online*)
Halaman 407-416
available online at http://jom.fti.budiluhur.ac.id/index.php/IDEALIS/index

3.7. Analisis Kesalahan Klasifikasi

Dari 1.399 data uji, tercatat 224 ulasan (16,01%) yang salah diklasifikasikan oleh Random Forest. Tabel 5 menyajikan tiga contoh representatif beserta pola kesalahannya.

Tabel 5. Contoh Ulasan Yang Salah Klasifikasi

Ulasan	Aktual	Prediksi	Pola Kesalahan
game nya bagus. tapi montton nya aja tolol ngasih tim kayak anjingg mana server nya suka ngelag lagi tapi tetap kasih Bintang 5 karna gw baik	Positif	Negatif	Bias label berbasis rating
kalah 10 kali tapi meneng 1 kali hebat bgt anjir tim gw lain kali cariin tim yang oon ya aku cintah kamuh	Positif	Negatif	Sarkasme
batalin hayaxhanabi, HAYAxKAGURA HRUS TETAP BERTAHAN SMPAI SKIN VALENTINE NYA #justice_for_hayagura	Negatif	Positif	Konten di luar topik kualitas *game*

Contoh 1 menunjukkan rating tinggi meski isi ulasan berupa komplain, menegaskan keterbatasan pelabelan berbasis rating. Contoh 2 (sarkasme) tidak tertangkap TF-IDF karena representasinya bersifat *bag of words*, tanpa mempertimbangkan urutan kata. Contoh 3 membahas isu komunitas, bukan kualitas aplikasi, sehingga sulit diklasifikasikan berdasarkan sentimen produk. Ketiganya bersifat ilustratif dari pemeriksaan kualitatif, bukan kuantifikasi menyeluruh terhadap seluruh kesalahan. Temuan ini memperkuat perlunya validasi manual label pada penelitian lanjutan.

3.8. Uji Robustness (*Cross Validation*)

Untuk menguji stabilitas performa model final terhadap variasi pembagian data, dilakukan validasi silang 5-*fold* pada data latih untuk kedua model dengan konfigurasi final, yaitu Random Forest dan KNN (K=11) yang sama-sama menggunakan SMOTE.

Tabel 6. Cross Validation

Model	*Accuracy* (mean ± std)	*Macro-F1* (mean ± std)	*Weighted-F1* (mean ± std)
Random Forest (SMOTE)	83,84% ± 0,76%	80,34% ± 0,90%	83,52% ± 0,75%
KNN K=11 (SMOTE)	74,10% ± 0,91%	72,09% ± 1,00%	74,97% ± 0,89%

Hasil pengujian pada Tabel 6 menunjukkan bahwa Random Forest memperoleh *Accuracy* rata-rata sebesar 83,84% dengan standar deviasi 0,76%, serta *Macro-F1* rata-rata sebesar 80,34% dengan standar deviasi 0,90%. Sementara itu, KNN memperoleh *Accuracy* rata-rata sebesar 74,10% dengan standar deviasi 0,91%, serta *Macro-F1* rata-rata sebesar 72,09% dengan standar deviasi 1,00%. Standar deviasi yang kecil pada seluruh metrik, yaitu di bawah 1%, menunjukkan bahwa performa kedua model relatif stabil dan tidak berfluktuasi signifikan antar *fold* pengujian.

Nilai rata-rata *Accuracy* dan *Macro-F1* hasil validasi silang ini juga konsisten dengan hasil evaluasi pada data uji yang telah dilaporkan pada Tabel 4. Konsistensi ini mengindikasikan bahwa performa Random Forest maupun KNN tidak bergantung pada satu skema pembagian data tertentu, sehingga memperkuat validitas temuan bahwa Random Forest secara konsisten unggul dibandingkan KNN pada dataset ulasan MLBB yang digunakan pada penelitian ini.

3.9. Diskusi

Berdasarkan hasil evaluasi dan pengujian robustness pada sub-bab sebelumnya, bagian ini membahas temuan tersebut secara lebih kritis dalam konteks penelitian terdahulu, penelitian ini menunjukkan beberapa perbedaan metodologis yang menjelaskan variasi hasil yang diperoleh. Sanjaya dkk. menggunakan SVM pada 6.000 ulasan dengan tiga kelas sentimen dan mencapai akurasi 79,88% [4], Ma'arif dkk. menerapkan Logistic Regression pada 1.000 ulasan tiga kelas dengan akurasi 79,00% [5], sementara Adi dkk. menggunakan Naïve Bayes dengan pelabelan berbasis leksikon InSet pada 1.000 ulasan dua kelas dengan akurasi 71,05% [6]. Random Forest dengan SMOTE pada penelitian ini mencapai akurasi 84,0% pada 6.994 ulasan.

Perbedaan skema klasifikasi turut menjelaskan variasi performa antar penelitian. Ma'arif dkk. mencatatkan *Precision*, *Recall*, dan *F1-score* sebesar 0,00 pada kelas netral akibat proporsi data netral yang sangat kecil (37 dari 1.000 ulasan) [5] , menunjukkan bahwa skema tiga kelas rentan gagal total pada kelas minoritas ekstrem tanpa penanganan ketidakseimbangan data yang memadai. Penelitian ini menggunakan skema dua kelas serta menerapkan SMOTE secara eksplisit untuk mengantisipasi risiko tersebut.

Temuan yang membedakan penelitian ini dari ketiga studi pembanding adalah eksperimen ablasi SMOTE. Sanjaya dkk. mencantumkan pustaka *imbalance* sebagai *tools* pendukung namun tidak menerapkannya pada hasil

akhir [4], sementara Ma'arif dkk. dan Adi dkk. merekomendasikan SMOTE sebagai arah penelitian mendatang tanpa menguji efeknya secara langsung [5], [6]. Penelitian ini mengisi celah tersebut dengan membuktikan secara empiris bahwa efek SMOTE bersifat berbeda antar algoritma, bisa dilihat pada Tabel 4, mengindikasikan bahwa manfaat SMOTE bersifat spesifik terhadap algoritma, bukan solusi generik seperti tersirat pada rekomendasi kedua penelitian tersebut.

Keunggulan akurasi penelitian ini terhadap ketiga studi pembanding perlu dimaknai secara proporsional, mengingat perbedaan skema klasifikasi dan skala data turut berkontribusi terhadap hasil, bukan semata-mata karena superioritas algoritma Random Forest. Kontribusi utama penelitian ini terletak pada bukti empiris bahwa efektivitas SMOTE bersifat *model-dependent*, serta koreksi metodologis pada pemilihan *hyperparameter* K melalui *cross-validation* untuk mencegah kebocoran data uji.

4. KESIMPULAN

Kajian ini menjawab permasalahan mengenai cara mengidentifikasi dan mengklasifikasikan sentimen pengguna dari volume ulasan yang besar, melalui pendekatan otomatis berbasis *machine learning* yang membandingkan Random Forest dan K-Nearest Neighbors pada 6.994 ulasan Mobile Legends: Bang Bang. Pada skenario eksperimen ini, Random Forest dengan SMOTE menunjukkan performa terbaik, mencapai *Accuracy* 84,0% dan *Macro-F1* 80,8%, dibandingkan KNN (K=11) dengan SMOTE yang memperoleh *Accuracy* 73,8% dan *Macro-F1* 71,9%, dengan selisih *Precision* kelas positif paling signifikan (75,9% berbanding 55,3%). Hasil ablasi turut menunjukkan bahwa efektivitas SMOTE bersifat *model-dependent*, meningkatkan *Macro-F1* Random Forest namun menurunkan *Macro-F1* KNN, sehingga SMOTE tidak dapat diperlakukan sebagai solusi universal untuk ketidakseimbangan kelas. Kontribusi metodologis lain penelitian ini adalah penentuan nilai K melalui *cross-validation* yang mengintegrasikan SMOTE ke dalam pipeline pelatihan guna mencegah kebocoran data uji.

Klaim keunggulan Random Forest perlu dimaknai secara proporsional, mengingat eksperimen hanya dilakukan pada satu dataset dan satu periode pengambilan data, serta pelabelan berbasis rating yang berpotensi kurang akurat pada ulasan bermuatan sarkasme. Penelitian ini juga belum menangani ulasan berbahasa daerah yang tereliminasi selama filtrasi, berpotensi memengaruhi representasi kelas negatif. Secara praktis, pipeline klasifikasi ini dapat dimanfaatkan pengembang untuk memetakan pola keluhan pengguna secara sistematis dari volume ulasan yang besar, sebagaimana teridentifikasi pada analisis WordCloud, sehingga prioritas perbaikan dapat ditentukan berdasarkan data sentimen alih-alih asumsi semata. Penelitian selanjutnya disarankan mengeksplorasi penanganan ulasan multibahasa dan pendekatan berbasis *deep learning* seperti IndoBERT.

DAFTAR PUSTAKA

[1] Kompas, "Transaksi 'Game' Rp 6,4 Miliar Per Hari, Siapa yang Menikmati?" Accessed: Jul. 14, 2026. [Online]. Available: https://www.kompas.id/artikel/transaksi-game-rp-64-miliar-per-hari-siapa-saja-yang-menikmati

[2] GoodStats, "Game Online Paling Sering Diakses Publik Indonesia 2025." Accessed: Jun. 12, 2026. [Online]. Available: https://data.goodstats.id/statistic/game-online-paling-sering-diakses-publik-indonesia-2025-e3fGx

[3] Indotelko, "Lima pemain wakili Indonesia di ENC 2026 Mobile Legends." Accessed: Jun. 12, 2026. [Online]. Available: https://www.indotelko.com/read/1781057622/lima-pemain-wakili-indonesia-di-enc-2026-mobile-legends

[4] S. S. Sanjaya, A. K. Jaya, R. Candra, and S. Zang, "Sentiment Analysis of Mobile Legends Game using Naïve Bayes, K-Nearest Neighbors and Support Vector Machine Algorithm," *Journal of Artificial Intelligence and Engineering Applications*, vol. 5, no. 2, pp. 2252–2260, Feb. 2026, doi: 10.59934/jaiea.v5i2.1840.

[5] W. A. Ma'arif, Sarwido, and T. Tamrin, "Analisis Sentimen Terhadap Ulasan Game Mobile Legend di Playstore menggunakan Algoritma Logistic Regression," *JUKI : Jurnal Komputer dan Informatika*, vol. 8, no. 1, pp. 43–49, May 2026, doi: 10.53842/juki.v8i1.2319.

[6] A. O. K. Adi, F. P. Gusti, and F. Wijaya, "Analisis Sentimen Ulasan Pengguna Aplikasi Mobile Legends Pada Google Playstore Menggunakan Naïve Bayes," in *Prosiding Seminar Nasional Teknologi dan Sains*, Kediri, Indonesia, 2025, pp. 641–646, doi: 10.29407/avw8b190.

[7] A. P. Zai and S. Saleh, "Perbandingan Naive Bayes Dan Support Vector Machine Untuk Analisis Sentimen Pengguna Jobstreet," *IDEALIS*, vol. 9, no. 1, pp. 30–38, Jan. 2026, doi: 10.36080/idealis.v9i1.3663.

[8] Handoko, D. Ramadhansyah, A. Asrofiq, Rahmaddeni, and Y. Yunefri, "Analisis Sentimen Ulasan Penumpang Maskapai Penerbangan di Indonesia dengan Algoritma Random Forest dan KNN," *ZONAsi: Jurnal Sistem Informasi*, vol. 6, no. 2, pp. 287–297, May 2024, doi: 10.31849/zn.v6i2.19177.

[9] M. Ulfa, R. H. Kusumodestoni, and A. Sucipto, "Analisis Sentimen Review Aplikasi Identitas Kependudukan Digital di Google Play Store Menggunakan KNN," *Jurnal Informatika Teknologi dan Sains*, vol. 6, no. 4, pp. 1155–1165, Nov. 2024, doi: 10.51401/jinteks.v6i4.4963.

[10] M. D. Rizkiyanto, M. D. Purbolaksono, and W. Astuti, "Sentiment Analysis Classification on PLN Mobile Application Reviews using Random Forest Method and TF-IDF Feature Extraction," *INTEK Jurnal Penelitian*, vol. 11, no. 1, pp. 37–43, Apr. 2024, doi: 10.31963/intek.v11i1.4774.

[11] J. Kalyzta, M. A. Willdan, S. Halfiani, and Indra, "Penerapan Analisis Sentimen Ujaran Kebencian Terhadap Vaksinasi Covid-19 pada Tweet Berbahasa Indonesia Menggunakan Algoritme K-Nearest Neighbor," *IDEALIS*, vol. 5, no. 2, pp. 87–97, Jul. 2022, doi: 10.36080/idealis.v5i2.2959.

[12] R. Maulana, A. Voutama, and T. Ridwan, "Analisis Sentimen Ulasan Aplikasi MyPertamina Pada Google Play Store Menggunakan Algoritma NBC," *Jurnal Teknologi Terpadu*, vol. 9, no. 1, pp. 42–48, Jul. 2023, doi: 10.54914/jtt.v9i1.609.

[13] A. B. Muzayyanah, R. E. Pawening, and Z. Arifin, "Analisis Sentimen Pada Ulasan Aplikasi Ehadrah Di Google Playstore Menggunakan Support Vector Machine," *IDEALIS*, vol. 7, no. 2, pp. 258–266, Jul. 2024, doi: 10.36080/idealis.v7i2.3250.

[14] R. Y. Hayuningtyas and R. Sari, "Analisis Sentimen Terhadap Pengguna Aplikasi Instagram di Google Play Store Menggunakan Support Vector Machine," *Journal of Accounting Information System*, vol. 5, no. 1, pp. 33–43, Jun. 2025, doi: 10.31294/jais.v5i01.8823.

[15] M. R. F. Rahmatullah, P. N. Andono, Affandy, and M. A. Soeleman, "Improving Random Forest Performance for Sentiment Analysis on Unbalanced Data Using SMOTE and BoW Integration: PLN Mobile Application Case Study," *Scientific Journal of Informatics*, vol. 12, no. 1, pp. 1–10, Apr. 2025, doi: 10.15294/sji.v12i1.19295.

[16] M. Fahrezi, Y. B. Pratama, and A. Pramudiyantoro, "Analisis Sentimen Debat Publik Pilpres 2024 Menggunakan Metode Algoritma LSTM dan IndoBERT Pada Platform Youtube," *JPIM: Jurnal Penelitian Ilmiah Multidisipliner*, vol. 2, no. 3, pp. 1936–1961, Oct. 2025.

[17] A. Wijaya, M. A. Dwiyanto, Z. Muttaqin, and M. A. F. Haq, "Sentiment Analysis of Netflix Review on Goggle Play with Machine Learning," *Jurnal Kecerdasan Buatan, Komputasi dan Teknologi Informasi*, vol. 6, no. 1, pp. 59–66, Jun. 2025, doi: 10.33650/coreai.v6i1.11731.

[18] A. I. Kamil, O. N. Pratiwi, and D. Witarsyah, "Analisis Sentimen dan Pemodelan Topik terhadap Aplikasi Pembelajaran Online pada Platform Google Play," *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 10, no. 2, pp. 836–849, Mar. 2025, doi: 10.29100/jipi.v10i2.6023.

[19] M. Z. Siregar, A. M. Elhanafi, and D. Irwan, "Analisis Sentimen Terhadap Ulasan Aplikasi Media Sosial Di Google Play Menggunakan Algoritma Naive Bayes," *Jurnal Mahasiswa Teknik Informatika*, vol. 9, no. 2, pp. 3373–3381, Apr. 2025, doi: 10.36040/jati.v9i2.12841.

[20] I. Amal and Jayanta, "Perbandingan Pelabelan Otomatis Dan Manual Untuk Analisis Sentimen Terhadap Kenaikan Harga BBM Pertamina Pada Twitter Menggunakan Algoritma Support Vector Machine," in *Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)*, Jakarta, Indonesia, Aug. 2023, pp. 473–487.

[21] J. Pardede and D. Darmawan, "Perbandingan Algoritma Stemming Porter, Sastrawi, Idris, dan Arifin & Setiono pada Dokumen Teks Bahasa Indonesia," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, vol. 12, no. 1, pp. 69–76, Feb. 2025, doi: 10.25126/jtiik.2025128860.

[22] K. P. Harmandini and K. Muslim, "Analysis of TF-IDF and TF-RF Feature Extraction on Product Review Sentiment," *Sinkron : Jurnal dan Penelitian Teknik Informatika*, vol. 8, no. 2, pp. 929–937, Mar. 2024, doi: 10.33395/v8i2.13376.

[23] A. Ichwani, R. I. Kesuma, A. Setiawan, E. I. Wicaksono, and R. Hanifah, "Preventing Data Leakage in Classification via Integrated Machine Learning Pipelines: Preprocessing, Feature Transformation, and Hyperparameter Tuning," *Jurnal Teknik Informatika (JUTIF)*, vol. 7, no. 1, pp. 391–410, Feb. 2026, doi: 10.52436/1.jutif.2026.7.1.5490.

[24] L. Hakim, A. Sobri, L. Sunardi, and D. Nurdiansyah, "Prediksi penyakit jantung berbasis mesin learning dengan menggunakan metode k-nn," *Jurnal Digital Teknologi Informasi*, vol. 7, no. 2, pp. 14–20, Feb. 2025, doi: 10.32502/digital.v7i2.9429.