ANALISIS SENTIMEN PENGGUNA ROBLOX PADA GOOGLE PLAY STORE MENGGUNAKAN ALGORITMA RANDOM FOREST

Resti Haren Absari, Setyawan Wibisono
Teknik Informatika, Universitas Stikubank Semarang
Jalan Tri Lomba Juang Semarang, Indonesia
restiabsari3@gmail.com

ABSTRAK

Pertumbuhan pesat aplikasi Roblox di Google Play Store mendorong meningkatnya jumlah ulasan pengguna yang merepresentasikan beragam pengalaman, kepuasan, serta permasalahan teknis yang dihadapi. Volume ulasan yang besar dan bersifat tidak terstruktur menyulitkan pengembang dalam mengekstraksi informasi secara manual. Penelitian ini bertujuan untuk menganalisis sentimen ulasan pengguna Roblox secara otomatis dengan mengklasifikasikan opini ke dalam kategori positif, negatif, dan netral menggunakan algoritma *random forest*. Dataset terdiri dari 3.000 ulasan berbahasa Indonesia yang dikumpulkan melalui teknik *web scraping*. Tahapan penelitian meliputi *preprocessing* teks (*cleaning, case folding, tokenization, stopword removal, dan stemming*), ekstraksi fitur menggunakan *term frequency–inverse document frequency* (TF-IDF), serta pemodelan klasifikasi *random forest*. Evaluasi model dilakukan menggunakan metrik *accuracy, precision, recall,* dan Hasil penelitian menunjukkan bahwa algoritma *random forest* mampu memberikan performa klasifikasi yang sangat baik dengan akurasi sebesar 94% dan nilai *f1-score* yang seimbang pada seluruh kelas sentimen. Temuan ini menunjukkan bahwa *random forest* efektif digunakan untuk analisis sentimen ulasan aplikasi berbahasa Indonesia dan dapat dimanfaatkan sebagai dasar pengambilan keputusan bagi pengembang dalam meningkatkan kualitas dan pengalaman pengguna aplikasi Roblox.

Kata kunci : *Analisis Sentimen, Random Forest, TF-IDF, Google Play Store, Roblox.*

1. PENDAHULUAN

Perkembangan aplikasi digital berbasis komunitas telah mendorong meningkatnya interaksi pengguna pada platform distribusi seperti Google Play Store. Salah satu aplikasi yang menunjukkan dinamika interaksi pengguna yang sangat tinggi adalah Roblox, yang tidak hanya berfungsi sebagai permainan, tetapi juga sebagai ruang sosial dan kreatif. Tingginya jumlah pengguna aktif menghasilkan volume ulasan yang besar dengan karakteristik bahasa yang beragam, mulai dari apresiasi hingga keluhan teknis. Ulasan tersebut merepresentasikan pengalaman dan persepsi pengguna yang bernilai penting bagi pengembang, namun sulit dianalisis secara manual karena jumlah dan kompleksitasnya.

Untuk mengatasi permasalahan tersebut, analisis sentimen menjadi pendekatan yang relevan dalam mengekstraksi opini pengguna dari data teks tidak terstruktur. Analisis sentimen memungkinkan pengelompokan opini ke dalam kategori positif, negatif, dan netral secara otomatis dengan memanfaatkan teknik *text mining* dan *machine learning*. Beberapa penelitian menunjukkan bahwa metode ini efektif digunakan pada data ulasan digital, baik pada media sosial maupun aplikasi, karena mampu menggambarkan kecenderungan persepsi pengguna secara objektif dan sistematis[1], [2].

Sejumlah penelitian terdahulu telah mengkaji analisis sentimen pada ulasan Roblox dan aplikasi serupa menggunakan berbagai algoritma klasifikasi. Alkindi dan Nasution menerapkan *Support Vector Machine* dan *Naive Bayes* pada ulasan Roblox dan memperoleh akurasi hingga 90%[3]. Pendekatan berbasis *deep learning* seperti IndoBERT juga menunjukkan performa sangat baik dengan akurasi di atas 93%, namun membutuhkan sumber daya komputasi yang lebih besar [2]. Di sisi lain, algoritma *random forest* dilaporkan mampu memberikan performa yang stabil dan kompetitif pada berbagai domain ulasan aplikasi, termasuk layanan hiburan dan aplikasi kreatif[4], [5], [6].

Meskipun demikian, kajian yang secara khusus mengevaluasi kinerja algoritma *random forest* pada ulasan game Roblox berbahasa Indonesia masih relatif terbatas. Padahal, ulasan game memiliki karakteristik bahasa yang khas, seperti penggunaan istilah informal, singkatan, serta ekspresi emosional yang beragam. Kondisi ini menimbulkan tantangan tersendiri dalam proses klasifikasi sentimen dan memerlukan pendekatan yang mampu menangani data teks berdimensi tinggi serta tidak seimbang secara distribusi kelas[7], [8].

Berdasarkan kondisi tersebut, penelitian ini bertujuan untuk menganalisis sentimen pengguna terhadap aplikasi Roblox pada Google Play Store dengan menerapkan algoritma *random forest*. Penelitian dilakukan melalui tahapan pengumpulan data ulasan, *preprocessing* teks, ekstraksi fitur menggunakan TF-IDF, pemodelan klasifikasi, serta evaluasi kinerja model menggunakan metrik akurasi, presisi, *recall*, dan *f1-score*. Hasil penelitian diharapkan mampu memberikan gambaran yang komprehensif mengenai persepsi pengguna Roblox serta menjadi dasar rekomendasi bagi pengembang dalam meningkatkan kualitas dan pengalaman pengguna aplikasi.

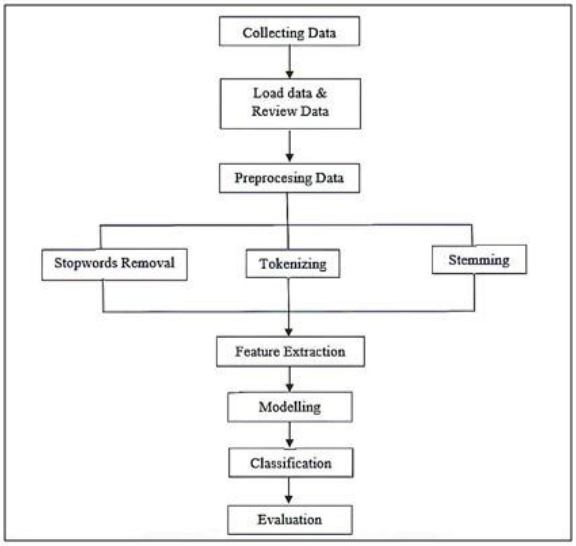
2. TINJAUAN PUSTAKA

2.1. Analisis Sentimen

Analisis sentimen merupakan bagian dari pengolahan bahasa alami (*Natural Language Processing*/NLP) yang berfokus pada identifikasi kecenderungan opini atau emosi dalam suatu teks, yang umumnya diklasifikasikan ke dalam sentimen positif, negatif, dan netral. Pendekatan ini banyak dimanfaatkan untuk menganalisis opini pengguna pada platform digital karena mampu merepresentasikan persepsi publik secara objektif dan terukur. Penelitian sebelumnya menunjukkan bahwa analisis sentimen efektif digunakan untuk menggali pola kepuasan dan ketidakpuasan pengguna pada berbagai layanan digital, termasuk *marketplace* dan aplikasi berbasis komunitas[2].

Dalam konteks aplikasi dan game, analisis sentimen berperan penting sebagai alat evaluasi pengalaman pengguna. Alkindi dan Nasution [3] membuktikan bahwa metode *machine learning* seperti *Support Vector Machine* (SVM) dan *naive bayes* mampu mengklasifikasikan ulasan pengguna Roblox dengan proporsi sentimen positif lebih dari 65%, di mana SVM menghasilkan performa terbaik. Temuan tersebut menunjukkan bahwa ulasan pengguna mengandung informasi emosional yang dapat dipetakan secara sistematis melalui pendekatan komputasional. Secara umum, proses analisis sentimen dilakukan melalui beberapa tahapan yang saling berkaitan dan memengaruhi hasil akhir klasifikasi.

a. Pengumpulan Data (*Data Collection*)
   Proses memperoleh data teks dari sumber yang relevan, dalam kasus ini adalah ulasan pengguna pada Google Play Store. Data ini menjadi dasar utama dalam analisis karena merepresentasikan pengalaman nyata pengguna terhadap suatu aplikasi.
b. Pra-pemrosesan Teks (*Text Preprocessing*)
   bertujuan untuk membersihkan data mentah dari unsur-unsur yang tidak relevan, seperti tanda baca, angka, emotikon, dan karakter khusus. Proses ini umumnya meliputi *case folding*, *tokenizing*, *stopword removal*, dan *stemming*. Penelitian Sukma dan Pribadi [4] menunjukkan bahwa *pre-processing* yang sistematis berpengaruh signifikan terhadap peningkatan kinerja model klasifikasi sentimen, khususnya pada data teks yang bersifat tidak terstruktur.
c. Ekstraksi Fitur (*Feature Extraction*)
   Teks yang sudah diproses selanjutnya diubah menjadi bentuk numerik supaya dipahami oleh algoritma machine learning. *TF-IDF* (*Term Frequency–Inverse Document Frequency*) atau *Word Embedding* merupakan dua metode yang paling umum digunakan[5].
d. Klasifikasi Sentimen (*Sentiment Classification*)
   Algoritma *machine learning* kemudian memahami teks yang telah diproses dalam format numerik. Dua metode yang paling umum digunakan adalah TF-IDF (*Term Frequency–Inverse Document Frequency*) dan *Word Embedding*[3].
e. Evaluasi Model (Model Evaluation)
   Pada langkah terakhir, kinerja model dinilai menggunakan metrik seperti *accuracy*, *precision*, *recall*, dan *f1-score*. Ini dilakukan untuk mengetahui kemampuan model untuk mengklasifikasikan sentimen dengan tepat.[9].



Gambar 1. *Flowchart* analisis sentimen

2.2. Text Mining

Text mining merupakan proses sistematis untuk mengekstraksi informasi dan pola pengetahuan dari data teks yang bersifat tidak terstruktur agar dapat diolah lebih lanjut dalam bentuk terstruktur. Dalam konteks analisis sentimen, *text mining* berperan penting sebagai fondasi awal karena data ulasan pengguna umumnya mengandung noise seperti simbol, singkatan, variasi bahasa informal, dan pengulangan kata. Penelitian sebelumnya menunjukkan bahwa penerapan *text mining* yang terstruktur mampu meningkatkan kualitas data dan akurasi model klasifikasi, khususnya pada data ulasan digital berskala besar [1], [7]. Tahapan utama dalam *text mining* meliputi pengumpulan data, *text preprocessing*, ekstraksi fitur, analisis, serta interpretasi hasil, di mana setiap tahapan saling berkaitan dan memengaruhi performa akhir sistem.

Beberapa penelitian terdahulu membuktikan bahwa integrasi text mining dengan algoritma *machine learning* memberikan hasil yang efektif dalam analisis sentimen. Agustina *et al.* [2] menunjukkan bahwa penerapan *text mining* yang dikombinasikan dengan *Support Vector Machine* mampu merepresentasikan sentimen pengguna secara akurat pada data Twitter. Studi lain oleh Rohim *et al.* [1] menegaskan bahwa *text mining* berperan penting dalam mengurangi ketidakteraturan data teks sebelum proses klasifikasi. Setelah teks diproses melalui tahapan *text mining*, data kemudian direpresentasikan dalam bentuk numerik menggunakan metode seperti TF-IDF untuk

selanjutnya diklasifikasikan menggunakan algoritma *machine learning*, seperti *random forest*, yang dikenal stabil dan efektif dalam menangani data teks berdimensi tinggi [4], [8]. Dengan demikian, *text mining* menjadi penghubung utama antara data ulasan mentah dan proses analisis sentimen berbasis *random forest.*

2.3. Data Collection

Dalam proses *text mining*, pengumpulan data merupakan tahap fundamental yang menentukan kualitas analisis pada tahapan selanjutnya. Data teks dapat diperoleh dari berbagai sumber digital seperti media sosial, forum daring, ulasan aplikasi, maupun platform distribusi aplikasi. Teknik yang umum digunakan dalam pengumpulan data meliputi pemanfaatan *Application Programming Interface* (API), pengambilan dataset publik, serta *web scraping*. Pada penelitian yang berfokus pada ulasan aplikasi, Google Play Store menjadi salah satu sumber data yang relevan karena menyediakan opini pengguna secara terbuka dalam jumlah besar, mencakup teks ulasan, nilai rating, serta metadata pendukung lainnya[1].

Beberapa penelitian terdahulu menunjukkan bahwa metode *web scraping* efektif digunakan untuk mengumpulkan data ulasan aplikasi dalam skala besar dan bersifat representatif. Rohim *et al.* [1] menegaskan bahwa akurasi dan kelengkapan data hasil *scraping* berpengaruh langsung terhadap performa analisis *text mining*. Hal serupa juga diterapkan pada penelitian analisis sentimen ulasan aplikasi digital yang memanfaatkan data Google Play Store sebagai sumber utama, karena data tersebut mencerminkan pengalaman pengguna secara aktual. Oleh karena itu, pemilihan metode pengumpulan data yang tepat menjadi faktor penting dalam menghasilkan dataset yang valid, relevan, dan mendukung proses analisis sentimen secara optimal.

2.4. Pre-Pocessing

Text pre-processing merupakan tahapan krusial dalam *text mining* karena kualitas data teks sangat menentukan performa model klasifikasi. Data ulasan pengguna umumnya bersifat tidak terstruktur dan mengandung berbagai bentuk *noise* seperti simbol, emotikon, singkatan, serta variasi penulisan yang tidak baku. Menurut Azis [7], *pre-processing* yang sistematis mampu meningkatkan akurasi model *machine learning* dengan mengurangi ketidakteraturan data dan mempertahankan informasi linguistik yang relevan. Penelitian Fauzi *et al.* [8] juga menunjukkan bahwa optimalisasi tahap *pre-processing* berpengaruh signifikan terhadap peningkatan kinerja *random forest* pada data sentimen berbahasa Indonesia. Berikut ini adalah tahapan *pre-processing* yang umum digunakan dalam penelitian analisis sentimen:

a. *Cleaning*
   Tahap *cleaning* bertujuan untuk menghilangkan komponen teks yang tidak relevan terhadap analisis sentimen, seperti angka, simbol, URL, tanda baca berlebihan, dan karakter khusus. Proses ini membantu mengurangi gangguan (*noise*) yang dapat memengaruhi hasil ekstraksi fitur. Rohim *et al.* [1] menyatakan bahwa pembersihan data yang tepat dapat meningkatkan konsistensi teks dan kualitas input sebelum diproses lebih lanjut oleh algoritma klasifikasi.
b. *Case Folding*
   Case folding adalah proses mengubah seluruh karakter teks menjadi huruf kecil (*lowercase*). Langkah ini dilakukan untuk menghindari perbedaan makna akibat variasi kapitalisasi huruf. Agustina *et al.* [2] menjelaskan bahwa *case folding* mampu mengurangi redundansi fitur kata dan membantu model dalam mempelajari distribusi kata secara lebih akurat pada proses analisis sentimen.
c. *Tokenization*
   Tokenisasi merupakan proses pemecahan teks menjadi unit-unit kata yang disebut token. Tahap ini penting karena sebagian besar metode NLP bekerja pada level kata. Menurut Agustina *et al.* [2], tokenisasi memungkinkan representasi teks yang lebih terstruktur sehingga hubungan antar kata dalam kalimat dapat dianalisis dengan lebih baik oleh algoritma *machine learning*.
d. *Stopword Removal*
   Stopword removal adalah proses menghapus kata-kata umum yang sering muncul namun tidak memiliki kontribusi signifikan terhadap penentuan sentimen, seperti “dan”, “yang”, atau “di”. Fauzi *et al.* [8] menyatakan bahwa penghapusan *stopword* dapat mengurangi jumlah fitur yang tidak informatif sehingga meningkatkan efisiensi komputasi dan kinerja model klasifikasi sentimen.
e. *Stemming*
   Stemming bertujuan untuk mengubah kata berimbuhan menjadi bentuk dasarnya agar variasi morfologi dianggap sebagai satu entitas makna. Adhan *et al.* [6] menunjukkan bahwa proses *stemming* dapat menurunkan dimensi fitur dan meningkatkan kemampuan model dalam mendeteksi pola sentimen, khususnya pada teks berbahasa Indonesia yang memiliki struktur morfologis kompleks.

2.5. Analysis and Information Extraction

Tahap *analysis and information extraction* merupakan proses penerapan algoritma *machine learning* untuk menganalisis pola linguistik dan mengekstraksi informasi dari teks ulasan yang telah dipra-pemrosesan. Pada tahap ini, representasi fitur menjadi komponen krusial karena algoritma klasifikasi hanya dapat bekerja pada data numerik. Salah satu metode yang paling umum digunakan adalah *Term Frequency–Inverse Document Frequency*

(TF-IDF), yang mengombinasikan frekuensi kemunculan kata dalam dokumen (TF) dan tingkat keunikan kata dalam keseluruhan korpus (IDF) sehingga mampu menonjolkan kata-kata yang relevan terhadap sentimen tertentu.

Pendekatan TF-IDF terbukti efektif dalam berbagai penelitian analisis sentimen, baik pada ulasan aplikasi maupun media sosial, karena mampu meningkatkan pemisahan antar kelas sentimen dan kinerja model klasifikasi [2], [4], [5]. Penelitian oleh Sukma dan Pribadi [4] serta Raff dan Ratnawati [5] menunjukkan bahwa penggunaan TF-IDF yang dikombinasikan dengan algoritma *random forest* menghasilkan performa klasifikasi yang stabil dan akurat pada data ulasan Play Store berbahasa Indonesia, sehingga metode ini dinilai relevan dan layak diterapkan dalam penelitian ini, yang dirumuskan sebagai berikut:

a. *Term Frequency* (TF)
Untuk mengukur frekuensi kemunculan kata(*t)* dalam dokumen(*d).*

$$TF(t,d) = \sum_{k=1}^{n} \frac{f_{k,d}}{f_{t,d}}$$

b. *Inverse Document Frequency* (IDF)
Mengukur tingkat keunikan kata dalam seluruh dokumen.

$$IDF(t) = \log\left(\frac{N}{df_t}\right)$$

c. TF-IDF
Bobot akhir kata dihitung sebagai hasil perkalian keduanya:

$$TF - IDF(t,d) = TF(t,d) \times IDF(t)$$

2.6. Interpretatioon and Evaluation

Tahap akhir dalam proses *text mining* adalah interpretasi hasil dan evaluasi kinerja model klasifikasi sentimen. Evaluasi ini bertujuan untuk menilai sejauh mana model mampu mengklasifikasikan sentimen secara akurat dan seimbang pada setiap kelas.

Sejumlah penelitian terdahulu menegaskan bahwa penggunaan metrik evaluasi kuantitatif merupakan pendekatan standar dalam analisis sentimen berbasis *machine learning*. Agustina *et al.* [2] serta Alkindi dan Nasution [3] menggunakan metrik evaluasi untuk memastikan model tidak hanya unggul dari sisi akurasi, tetapi juga mampu menangani distribusi kelas yang tidak seimbang. Selain itu, penelitian oleh Sukma dan Pribadi [4] serta Raff dan Ratnawati [5] menunjukkan bahwa evaluasi yang komprehensif menggunakan beberapa metrik memberikan gambaran performa model yang lebih objektif dibandingkan hanya mengandalkan satu indikator saja. Model klasifikasi sentimen dinilai menggunakan beberapa metrik kuantitatif berikut:

a. *Accuracy*
Accuracy digunakan untuk menghitung persentase prediksi yang benar dari keseluruhan data uji, sebagaimana umum diterapkan dalam evaluasi klasifikasi sentimen berbasis *machine learning* [4], [9], yang dirumuskan sebagai berikut:

$$Accuracy = \frac{TP + TN}{TP + TNFP + FN}$$

b. *Precision*
Precision menggambarkan proporsi prediksi positif yang benar terhadap seluruh prediksi positif, dengan menunjukkan tingkat ketepatan model dalam memberikan label sentimen dan banyak digunakan pada penelitian analisis sentimen berbasis *random forest* [4]. Dirumuskan sebagai berikut:

$$Precision = \frac{TP}{TP + FP}$$

c. *Recall*
Recall digunakan untuk mengukur kemampuan model dalam mengidentifikasi seluruh data yang benar-benar termasuk ke dalam suatu kelas sentimen, sehingga menunjukkan tingkat sensitivitas model terhadap kelas tertentu [8]. Yang dirumuskan sebagai berikut:

$$Recall = \frac{TP}{TP + FN}$$

d. *F1-Score*
Untuk menyeimbangkan nilai *precision* dan *recall*, digunakan metrik *f1-score* sebagai rata-rata harmonik keduanya, yang secara luas diterapkan dalam evaluasi klasifikasi sentimen untuk memastikan performa model tetap seimbang antar kelas sentimen [9]. Yang dapat dirumuskan dengan persamaan:

$$F1 = 2 \times \frac{Precision \times Recall}{Precision \times Recall}$$

e. *Confusion Matrix*
Confusion matrix digunakan untuk menggambarkan keterkaitan antara hasil prediksi model dan label aktual, sehingga kesalahan klasifikasi pada setiap kelas sentimen dapat dianalisis secara lebih rinci [10]. Metrik-metrik evaluasi tersebut dihitung menggunakan rumus matematis sebagai berikut:

$$CM = \begin{bmatrix} TPpos & FPpos \rightarrow net & FPpos \rightarrow neg \\ FPnet \rightarrow pos & TPnet & FPnet \rightarrow neg \\ FPneg \rightarrow pos & FPneg \rightarrow net & TPneg \end{bmatrix}$$

Di mana:
TP: *True Positive*
TN: *True Negative*
FP:*False Postive*
FN:*False Negative*

2.7. Algoritma Random Forest

Random forest merupakan algoritma *ensemble learning* yang membangun sejumlah *decision tree* dari subset data dan fitur yang dipilih secara acak, kemudian menggabungkan hasil prediksi setiap pohon melalui mekanisme *majority voting*. Pendekatan ini mampu meningkatkan stabilitas dan akurasi model dibandingkan penggunaan satu pohon keputusan tunggal, serta efektif dalam menangani data berdimensi tinggi dan bersifat *noisy*. Dalam konteks analisis teks dan sentimen, *random forest* terbukti mampu mengklasifikasikan data dengan baik pada berbagai domain, termasuk deteksi email spam dan

ulasan aplikasi digital [10]. Selain itu, algoritma ini relatif efisien secara komputasi dan tidak memerlukan asumsi distribusi data tertentu, sehingga banyak digunakan dalam penelitian *machine learning* berbasis teks [7].

Beberapa penelitian terdahulu menunjukkan bahwa Random Forest memiliki performa yang kompetitif dalam analisis sentimen ulasan aplikasi di Google Play Store. Sukma dan Pribadi [4] serta Raff dan Ratnawati [5] melaporkan bahwa Random Forest mampu menghasilkan akurasi di atas 85% pada klasifikasi sentimen ulasan aplikasi hiburan dan video editing. Optimalisasi lebih lanjut dengan teknik penyeimbangan data seperti *Synthetic Minority Over-sampling Technique* (SMOTE) dan pencarian parameter menggunakan *GridSearch* juga terbukti meningkatkan kinerja *random forest* pada dataset dengan distribusi kelas tidak seimbang [8]. Meskipun demikian, penerapan *random forest* secara khusus pada ulasan game Roblox berbahasa Indonesia masih relatif terbatas, sehingga penelitian ini dilakukan untuk mengisi celah tersebut dan mengevaluasi efektivitas algoritma *random forest* dalam konteks ulasan game berbasis komunitas.

2.8. Dataset Ulasan Roblox

Roblox merupakan platform permainan interaktif yang mengintegrasikan aspek hiburan, sosial, dan kreativitas dengan basis pengguna aktif yang sangat besar. Tingginya intensitas interaksi pengguna tersebut berdampak langsung pada meningkatnya jumlah ulasan di Google Play Store, yang berisi beragam opini terkait kualitas aplikasi, performa sistem, serta pengalaman bermain. Ulasan pengguna ini menjadi sumber data tekstual yang bernilai karena mampu merepresentasikan persepsi dan tingkat kepuasan pengguna secara langsung terhadap layanan yang diberikan aplikasi [3], [9]. Oleh karena itu, pemanfaatan data ulasan dalam skala besar melalui pendekatan analisis sentimen menjadi langkah strategis untuk memahami kecenderungan opini pengguna secara objektif dan sistematis.

Sejumlah penelitian terdahulu telah memanfaatkan ulasan aplikasi sebagai objek analisis sentimen dengan hasil yang menunjukkan efektivitas pendekatan *text mining* dan *machine learning*. Alkindi dan Nasution [3] menunjukkan bahwa ulasan Roblox didominasi sentimen positif dengan akurasi klasifikasi mencapai 90% menggunakan SVM, sementara Ansyah dan Suryono [9] berhasil meningkatkan performa klasifikasi melalui pendekatan IndoBERT. Pada konteks algoritma *random forest*, penelitian oleh Sukma dan Pribadi [4] serta Raff dan Ratnawati [5] membuktikan bahwa metode ini mampu menghasilkan performa yang stabil dan akurat dalam klasifikasi sentimen ulasan aplikasi Play Store. Namun, kajian yang secara khusus membahas penerapan *random forest* pada ulasan Roblox berbahasa Indonesia masih terbatas, sehingga penelitian ini berupaya mengisi celah tersebut dengan menganalisis sentimen pengguna Roblox menggunakan pendekatan *random forest* berbasis TF-IDF.

3. METODE PENELITIAN

Metodologi penelitian ini dirancang untuk melakukan analisis sentimen terhadap ulasan pengguna aplikasi Roblox pada Google Play Store menggunakan algoritma *random forest*. Seluruh tahapan disusun secara sistematis mulai dari pengumpulan data, pemrosesan teks, ekstraksi fitur, pemodelan, hingga evaluasi performa.

Gambar 2. Alur penelitian

3.1. Pengumpulan Data

Data penelitian diperoleh melalui *web scraping* terhadap ulasan aplikasi Roblox di Google Play Store. Platform ini dipilih karena menyediakan ulasan pengguna yang autentik dan mudah diakses. Proses pengambilan data dilakukan menggunakan library *google-play-scraper* pada Google Colab dengan *Application ID "com.roblox.client*" dan konfigurasi pengambilan 3.000 ulasan berbahasa Indonesia. Atribut yang diunduh meliputi teks ulasan, skor rating, identitas ulasan, waktu unggah, serta respons pengembang. Seluruh data disimpan dalam format CSV untuk mempermudah tahap analisis. Data bersifat sekunder dan digunakan hanya untuk tujuan penelitian tanpa mengubah isi ulasan asli.

3.2. Pre-Processing

Pre-processing dilakukan untuk menstandarkan ulasan yang bersifat tidak terstruktur dan penuh *noise*. Tahap ini terdiri dari lima langkah utama yang dijalankan berurutan. Pertama, *text cleaning* dilakukan untuk menghapus duplikasi, nilai kosong, angka tanpa konteks, tanda baca, emotikon, karakter khusus, URL, serta menormalisasi spasi. Kedua, *case folding* menyamakan seluruh huruf menjadi *lowercase* agar tidak terjadi perbedaan interpretasi kata. Ketiga, *tokenisasi* memecah kalimat menjadi kata-kata sebagai unit analisis. Keempat, *stopword removal* menghilangkan kata umum yang tidak berkontribusi pada pembentukan sentimen menggunakan daftar *stopword Sastrawi*. Terakhir, algoritma Nazief–Adriani digunakan untuk mengubah kata berimbuhan menjadi bentuk dasarnya melalui *stemming.* Teks yang telah dibersihkan dan siap untuk tahap ekstraksi fitur berikutnya dihasilkan melalui rangkaian prosedur ini.

3.3. Pelabelan Sentimen

Pelabelan sentimen dilakukan dengan memanfaatkan rating pengguna sebagai dasar penentuan kelas sentimen (*distant supervision*). Setiap ulasan diberi label berdasarkan skema tiga kategori, yaitu: positif untuk rating 4–5, netral untuk rating 3, serta negatif untuk rating 1–2. Pendekatan ini dipilih

karena rating merepresentasikan tingkat kepuasan pengguna secara langsung, memungkinkan proses anotasi dilakukan secara otomatis dalam jumlah besar, dan terbukti memiliki hubungan yang konsisten dengan ekspresi sentimen pada teks ulasan.

3.4. Ekstraksi Fitur

Pada langkah ini, teks yang telah diproses diubah menjadi format numerik yang dapat diolah oleh algoritma klasifikasi. Sebagai representasi, TF-IDF (*Term Frequency–Inverse Document Frequency*) digunakan untuk menggambarkan tingkat kepentingan setiap kata dalam korpus. Proses ekstraksi dijalankan dengan menerapkan *TfidfVectorizer* dari *scikit-learn* dengan pengaturan *max_features* 5.000, *ngram_range* (1,1) untuk *unigram*, dan *min_df* 2 agar hanya kata yang muncul minimal dua kali yang dipertahankan. Hasilnya adalah matriks berdimensi $n \times m$, di mana n merupakan jumlah dokumen dan m adalah fitur unik, dengan setiap nilai TF-IDF menunjukkan bobot suatu kata relatif terhadap dokumen dan keseluruhan dataset.

3.5. Pemodelan Random Forest

Dataset yang telah diseimbangkan dibagi menjadi data latih 80% dan data uji 20% menggunakan *stratified splitting* untuk menjaga proporsi kelas. Model dibangun dengan *RandomForestClassifier* menggunakan 200 pohon keputusan, *random_state* 42, dan pemrosesan paralel melalui *n_jobs*=-1. Model mempelajari hubungan antara fitur TF-IDF dan label sentimen pada tahap pelatihan. Setelah tahap ini selesai, model digunakan untuk membuat prediksi pada data uji.

3.6. Evaluasi Model

Kemudian kinerja model dinilai menggunakan metrik akurasi, presisi, *recall*, dan *f1-score*. Akurasi mengukur proporsi prediksi yang benar, presisi mengukur ketepatan prediksi yang positif, dan *recall* menunjukkan kemampuan model untuk mengumpulkan data yang positif secara keseluruhan. *F1-score* adalah rata-rata harmonis antara presisi dan *recall*. Untuk melihat distribusi kesalahan prediksi pada setiap kelas, evaluasi dilakukan dengan menggunakan *classification_report* dan *confusion matrix*.

4. HASIL DAN PEMBAHASAN

4.1. Deskripsi Data

Sebanyak 3.000 ulasan Roblox berhasil dikumpulkan dari Google Play Store dan seluruh entri dinyatakan valid setelah proses pembersihan. Ulasan memiliki variasi bahasa yang luas, mulai dari penggunaan bahasa informal, singkatan, hingga repetisi huruf yang menandakan ekspresi emosional. Variasi tersebut menunjukkan perlunya *pre-processing* yang menyeluruh agar teks lebih sesuai untuk proses analisis.

4.2. Pre-Processing

Pre-processing tahap ini adalah menormalisasi data teks yang tidak terstruktur (*noise*) menjadi format yang efisien dan bermakna untuk diolah oleh algoritma *machine learning*.

4.3. Cleaning

Tahap *cleaning* dapat menghilangkan elemen yang tidak relevan dari dataset yang telah dikumpulkan, seperti tanda baca, angka, simbol, serta menangani repetisi karakter yang berlebihan.

Tabel 1. Hasil *cleaning*

Index	***Cleaned***
1	Seru game nya walaupun ada beberapa bug dan kadang suka ngelag tpi gpp bikin game kayak gini tidak mudah jadi maklum game nya banyak bisa multiplayer banyak fitur kadang ada event event
2	Tolong ya developer ini device ku udah ga kentang grafik juga bisa rata kanan tapi kenapa ga bisa ngerender map nya bikin susah kalo main tolong banget di perbaiki lagi di bagian rendering susah amat render nya
3	Permainannya lumayan keren tapi gk ada bahasa lokal cuma ada bahasa inggris agak bingung sih mainya soalnya kurang paham bahasa inggris coba beri bahasa lokal deh biar bisa dimengerti saat bermain
4	Aplikasinya seru tetapi banyak oknum yg menyalahgunakan aplikasi ini tapi its okay saya tetap suka dengan gamenya mapnya bayak temanya seru apalagi map vd sama map gunung gunung seru banget
5	Akhir akhir ini jadi sering error baju classic jadi ngeblur tiba tiba disconnect padahal jaringan bagus masuk aplikasi tiba tiba log out sendiri lemot parah semenjak update

4.4. Case Folding

Pada tahap ini, seluruh huruf diubah menjadi huruf kecil untuk mengurangi duplikasi kata yang berbeda hanya karena kapitalisasi.

Tabel 2. Hasil *case folding*

Index	***Case Folding***
1	seru game nya walaupun ada beberapa bug dan kadang suka ngelag tpi gpp bikin game kayak gini tidak mudah jadi maklum game nya banyak bisa multiplayer banyak fitur kadang ada event event
2	tolong ya developer ini device ku udah ga kentang grafik juga bisa rata kanan tapi kenapa ga bisa ngerender map nya bikin susah kalo main tolong banget di perbaiki lagi di bagian rendering susah amat render nya

Index	***Case Folding***
3	permainannya lumayan keren tapi gk ada bahasa lokal cuma ada bahasa inggris agak bingung sih mainya soalnya kurang paham bahasa inggris coba beri bahasa lokal deh biar bisa dimengerti saat bermain
4	aplikasinya seru tetapi banyak oknum yg menyalahgunakan aplikasi ini tapi its okay saya tetap suka dengan gamenya mapnya bayak temanya seru apalagi map vd sama map gunung gunung seru banget
5	akhir akhir ini jadi sering error baju classic jadi ngeblur tiba tiba disconnect padahal jaringan bagus masuk aplikasi tiba tiba log out sendiri lemot parah semenjak update

4.5. Tokenization

Pada tahap ini tokenisasi telah berhasil memecah setiap kalimat ulasan menjadi satuan kata individual (*token*).

Tabel 3. Hasil *tokenization*

Index	***Tokenization***
1	['seru', 'game', 'nya', 'walaupun', 'ada', 'beberapa', 'bug', 'dan', 'kadang', 'suka', 'ngelag', 'tpi', 'gpp', 'bikin', 'game', 'kayak', 'gini', 'tidak', 'mudah', 'jadi', 'maklum', 'game', 'nya', 'banyak', 'bisa', 'multiplayer', 'banyak', 'fitur', 'kadang', 'ada', 'event', 'event']
2	['tolong', 'ya', 'developer', 'ini', 'device', 'ku', 'udah', 'ga', 'kentang', 'grafik', 'juga', 'bisa', 'rata', 'kanan', 'tapi', 'kenapa', 'ga', 'bisa', 'ngerender', 'map', 'nya', 'bikin', 'susah', 'kalo', 'main', 'tolong', 'banget', 'di', 'perbaiki', 'lagi', 'di', 'bagian', 'rendering', 'susah', 'amat', 'render', 'nya']
3	['permainannya', 'lumayan', 'keren', 'tapi', 'gk', 'ada', 'bahasa', 'lokal', 'cuma', 'ada', 'bahasa', 'inggris', 'agak', 'bingung', 'sih', 'mainya', 'soalnya', 'kurang', 'paham', 'bahasa', 'inggris', 'coba', 'beri', 'bahasa', 'lokal', 'deh', 'biar', 'bisa', 'dimengerti', 'saat', 'bermain']
4	['aplikasinya', 'seru', 'tetapi', 'banyak', 'oknum', 'yg', 'menyalahgunakan', 'aplikasi', 'ini', 'tapi', 'its', 'okay', 'saya', 'tetap', 'suka', 'dengan', 'gamenya', 'mapnya', 'bayak', 'temanya', 'seru', 'apalagi', 'map', 'vd', 'sama', 'map', 'gunung', 'gunung', 'seru', 'banget']
5	['akhir', 'akhir', 'ini', 'jadi', 'sering', 'error', 'baju', 'classic', 'jadi', 'ngeblur', 'tiba', 'tiba', 'disconnect', 'padahal', 'jaringan', 'bagus', 'masuk', 'aplikasi', 'tiba', 'tiba', 'log', 'out', 'sendiri', 'lemot', 'parah', 'semenjak', 'update']

4.6. Stopword Removal

Stopword removal mengeliminasi kata-kata umum yang memiliki frekuensi tinggi namun tidak memiliki kontribusi signifikan terhadap polaritas sentiment.

Tabel 4. Hasil *stopword removal*

Index	***Stopworld Removal***
1	['seru','game','beberapa','bug','kadang','suka','ngelag','gpp','bikin','game','kayak','gini','mudah','maklum','game','banyak','multiplayer','banyak','fitur','kadang','event','event']
2	['tolong','developer','device','udah','kentang','grafik','bisa','rata','kanan','kenapa','ga','bisa','ngerender','map','bikin','susah','main','tolong','banget','perbaiki','bagian','rendering','susah','amat','render']
3	['permainannya','lumayan','keren','gk','bahasa','lokal','bahasa','inggris','agak','bingung','mainya','kurang','paham','bahasa','inggris','coba','beri','bahasa','lokal','dimengerti','bermain']
4	['aplikasinya','seru','banyak','oknum','menyalahgunakan','aplikasi','okay','tetap','suka','game','map','bayak','temanya','seru','apalagi','map','vd','map','gunung','gunung','seru','banget']
5	['akhir','akhir','sering','error','baju','classic','ngeblur','disconnect','jaringan','bagus','masuk','aplikasi','log','out','sendiri','lemot','parah','semenjak','update']

4.7. Stemming

Stemming mengonsolidasikan semua bentuk morfologis (misalnya, *bermain*, *dimainkan* menjadi ‘main’) menjadi satu fitur tunggal, yang krusial untuk memadatkan ruang fitur.

Tabel 5. Hasil *stemming*

Index	***Stemming***
1	['seru','game','berapa','bug','kadang','suka','lag','gpp','bikin','game','kayak','gini','mudah','maklum','game','banyak','multiplayer','banyak','fitur','kadang','event','event']
2	['tolong','developer','device','udah','kentang','grafik','bisa','rata','kanan','kenapa','ga','bisa','render','map','bikin','susah','main','tolong','banget','baik','bagian','render','susah','amat','render']
3	['main','lumayan','keren','gk','bahasa','lokal','bahasa','inggris','agak','bingung','main','kurang','paham','bahasa','inggris','coba','beri','bahasa','lokal','erti','main']
4	['aplikasi','seru','banyak','oknum','salahguna','aplikasi','okay','tetap','suka','game','map','bayak','tema','seru','apalagi','map','vd','map','gunung','gunung','seru','banget']
5	['akhir','akhir','sering','error','baju','classic','blur','disconnect','jaring','bagus','masuk','aplikasi','log','out','sendiri','lemot','parah','semenjak','update']

4.8. Diatribusi Sentimen

Hasil pelabelan berbasis rating menghasilkan tiga kategori sentimen dengan distribusi sebagai berikut: positif sebesar 68,13%, negatif 25,80%, dan netral 6,07%. Dominasi ulasan positif menunjukkan bahwa mayoritas pengguna memberikan pengalaman yang memuaskan terhadap aplikasi. Sementara itu, proporsi ulasan negatif yang cukup besar menandakan bahwa berbagai keluhan teknis masih sering dialami pengguna. Ketidakseimbangan distribusi ini menjadi alasan diterapkannya teknik oversampling pada tahap pelatihan model agar performa klasifikasi tidak bias terhadap kelas mayoritas.

Tabel 6. Distribusi Rating

Sentimen	Rentang Rating	Jumlah Ulasan	Persentase
Positif	4 – 5	2.044	68,13%
Negatif	1 – 2	774	25,80%
Netral	3	182	6,07%
Total	–	3.000	100%

4.9. Ektraksi Fitur

Proses ekstraksi fitur dengan TF-IDF menghasilkan representasi numerik yang mampu menyoroti kata-kata yang berperan penting dalam membedakan sentimen. Bobot TF-IDF yang tinggi menunjukkan bahwa kata tersebut muncul secara signifikan dalam dokumen tertentu tetapi tidak umum di seluruh korpus, sehingga relevan sebagai fitur klasifikasi. Tabel berikut memberikan contoh nilai TF-IDF untuk beberapa kata yang sering muncul dalam ulasan:

Tabel 7. Contoh Bobot TF-IDF pada Beberapa Dokumen

Fitur	D1	D2	D3	D4	D5	DF	IDF
bagus	0.5073	–	–	–	–	802	2.3183
seru	0.5401	–	–	–	1.0000	690	2.4686
game	–	–	–	–	–	935	2.1651
roblox	–	1.0000	–	–	–	393	3.0303
masuk	–	–	0.8788	–	–	84	4.5640

Berdasarkan tabel tersebut, kata bagus dan seru memiliki bobot tinggi pada dokumen positif sehingga menjadi indikator kuat sentimen positif. Sebaliknya, kata seperti masuk cenderung muncul dalam keluhan login dan memiliki IDF tinggi, mencerminkan kontribusinya terhadap sentimen negatif. Sementara itu, kata umum seperti game dan roblox memiliki DF tinggi karena sering muncul di semua ulasan, sehingga bobotnya lebih rendah dan tidak bersifat diskriminatif.

4.10. Pemodelan Random Forest

Model *Random Forest* diuji menggunakan pembagian data 80% untuk *training* dan 20% untuk *testing*. Evaluasi dilakukan untuk menilai kemampuan model dalam mengenali pola sentimen dari teks ulasan yang telah melalui tahap preprocessing dan ekstraksi fitur.

Secara umum, model menunjukkan performa yang sangat baik dengan akurasi 94% pada data *testing*. Nilai *precision*, *recall*, dan *f1-score* pada seluruh kategori sentimen berada pada rentang 0,88–0,99, menandakan bahwa model mampu bekerja stabil untuk ketiga kelas.

Kategori netral memperoleh skor terbaik (*f1-score* 0,99), menunjukkan bahwa ulasan netral memiliki pola linguistik yang lebih seragam dan mudah dipisahkan. Kategori negatif juga menunjukkan performa kuat (*f1-score* 0,92), terutama karena model mampu mendeteksi keluhan teknis dengan baik. Sebagian besar kesalahan klasifikasi terjadi pada ulasan positif yang disertai keluhan ringan sehingga cenderung diprediksi sebagai netral.

Tabel 8. Evaluasi Model Random Forest

Sentimen	Precision	Recall	F1-Score	Support
Negatif	0.89	0.95	0.92	382
Netral	0.99	0.98	0.99	431
Positif	0.93	0.88	0.91	414
Akurasi	–	–	0.94	1227
Macro Avg	0.94	0.94	0.94	1227
Weighted Avg	0.94	0.94	0.94	1227

Hasil ini menunjukkan bahwa *Random Forest* mampu mempelajari pola linguistik ulasan Roblox dengan sangat baik, dan distribusi performa antar-kelas yang seimbang mengindikasikan bahwa teknik penyeimbangan data (*oversampling*) pada tahap training berjalan efektif.

4.11. Pembahasan

Hasil penelitian menunjukkan bahwa *Random Forest* mampu memberikan performa klasifikasi yang sangat baik pada data ulasan berbahasa Indonesia. Tingginya akurasi yang diperoleh tidak terlepas dari rangkaian preprocessing yang sistematis, pemanfaatan TF-IDF sebagai representasi fitur, serta penyeimbangan kelas menggunakan oversampling sehingga model dapat belajar secara lebih proporsional. Jika dibandingkan dengan beberapa penelitian sebelumnya, performa Random Forest pada penelitian ini berada pada tingkat yang kompetitif bahkan lebih unggul pada konteks dan dataset yang sama.

Distribusi sentimen memperlihatkan bahwa sebagian besar pengguna memberikan pengalaman positif dalam menggunakan Roblox. Meskipun demikian, ulasan bernada negatif tetap muncul dengan proporsi yang cukup berarti dan umumnya berkaitan dengan isu teknis seperti bug, lag, dan gangguan akses. Temuan ini memberikan gambaran mengenai aspek-aspek aplikasi yang perlu menjadi perhatian pengembang untuk peningkatan kualitas layanan.

Secara metodologis, penelitian ini menunjukkan bahwa kombinasi preprocessing teks bahasa Indonesia, ekstraksi fitur menggunakan TF-IDF, dan algoritma Random Forest merupakan pendekatan yang efektif untuk menganalisis ulasan aplikasi dalam skala besar. Pendekatan ini juga dinilai cukup efisien secara

komputasi dan dapat direplikasi pada objek penelitian serupa.

5. KESIMPULAN DAN SARAN

Penelitian ini menyimpulkan bahwa kombinasi preprocessing teks berbahasa Indonesia, representasi fitur menggunakan TF-IDF, serta algoritma *Random Forest* mampu menghasilkan model analisis sentimen dengan performa tinggi pada ulasan pengguna Roblox di Google Play Store. Model mencapai akurasi 94% dengan nilai *precision, recall,* dan *f1-score* yang stabil di seluruh kategori, menunjukkan kemampuan yang baik dalam mengenali pola linguistik pada data ulasan yang beragam. Distribusi sentimen mengindikasikan bahwa mayoritas pengguna memberikan respons positif, meskipun masih terdapat keluhan teknis yang cukup sering muncul. Untuk penelitian selanjutnya, disarankan menggunakan metode embedding yang mampu menangkap konteks semantik yang lebih dalam, seperti *Word2Vec, FastText,* atau BERT, serta memperluas sumber data dari berbagai platform agar hasil analisis lebih komprehensif dan generalis.

DAFTAR PUSTAKA

[1] Rohim, A., Irfani, M. H., Ramadhan, M., & Ubaidillah, U. (2023). Penerapan Metode Text Mining dengan Chatbot Questions And Answer pada PT PLN (Persero) Sumatera Selatan. *Klik-Jurnal Ilmu Komputer*, *4*(2), 59-67.

[2] Agustina, D. A., Subanti, S., & Zukhronah, E. (2021). Implementasi Text Mining Pada Analisis Sentimen Pengguna Twitter Terhadap Marketplace di Indonesia Menggunakan Algoritma Support Vector Machine. *Indonesian Journal of Applied Statistics*, *3*(2), 109-122.

[3] Alkindi, A. F., & Nasution, N. (2024). Analisis Sentimen Ulasan Pengguna Pada Game Roblox Dengan Metode Support Vector Machine Dan Naive Bayes. *J-Com (Journal of Computer)*, *4*(2), 164-177.

[4] Sukma, T. P. W., & Pribadi, M. R. (2024). Analisis Sentimen Review Pengguna Viu Pada Play Store Dengan Algoritma Random Forest. *Jurnal Software Engineering and Computational Intelligence*, *2*(01), 9-16.

[5] Raff, H. W., & Ratnawati, F. (2025). Klasifikasi Sentimen Ulasan Pengguna terhadap Aplikasi Video Editing Play Store Menggunakan Random Forest. *Techno. com*, *24*(3).

[6] Adhan, S. N., Wibawa, G. N. A., Arisona, D. C., Yahya, I., & Ruslan, R. (2024). Analisis Sentimen Ulasan Aplikasi Wattpad Di Google Play Store Dengan Metode Random Forest. *AnoaTIK: Jurnal Teknologi Informasi dan Komputer*, *2*(1), 6-15.

[7] Azis, A. R. (2024). Analisis Komparasi Algoritma Machine Learning dalam Prediksi Performa Akademik Mahasiswa: Literature Review. *Jurnal Ilmu Komputer dan Informatika*, *4*(2), 143-148.

[8] Fauzi, A., Yunial, A. H., Saputro, D. E., & Saputra, R. (2025). Optimalisasi Random Forest untuk Sentimen Bahasa Indonesia dengan GridSearch dan SMOTE. *Jurnal Ilmu Komputer dan Sistem Informasi*, *4*(2), 202-217.

[9] Ansyah, F., & Suryono, R. R. (2025). Sentiment Classification of Indonesian-Language Roblox Reviews Using IndoBERT with SMOTE Optimization. *Journal of Applied Informatics and Computing*, *9*(4), 1868-1877.

[10] Rustam, M., Brotokuncoro, A., & Roestam, R. (2024). Deteksi Email Spam dengan Continuous Bag-Of-Words dan Random Forest. *Ranah Research: Journal of Multidisciplinary Research and Development*, *6*(4), 758-765.