Journal of Applied Informatics and Computing (JAIC)
Vol.10, No.2, April 2026, pp. 1662~1673
e-ISSN: 2548-6861

Sentiment Classification of Indonesian E-Government Application Reviews Using Advanced Learning Models

Aulia Diaz Gustiavani [1], Muljono [2]*
[1,2] Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro
111202214446@mhs.dinus.ac.id [1], muljono@dsn.dinus.ac.id [2]

Article Info

Article history:

Received 2026-01-08
Revised 2026-02-20
Accepted 2026-02-27

Keyword:

E-Government Reviews,
Learning Models.
Natural Language,
Sentiment Analysis,
Text Classification.

ABSTRACT

The digital transformation of public services in Indonesia has led to the development of e-government applications such as Cek Bansos, aimed at improving transparency in social assistance distribution. However, user reviews indicate varying perceptions of service quality. This study conducts a comparative evaluation of machine learning and deep learning models for sentiment classification of Indonesian e-government application reviews. A total of 28,697 reviews were collected via web scraping, with 27,985 retained after preprocessing. Sentiment labels were assigned automatically based on rating scores (1–2 as negative, 4–5 as positive), while neutral reviews were excluded. To address class imbalance, SMOTE and Random Oversampling were applied to the training data for machine learning and deep learning models, respectively. TF-IDF features were used with Logistic Regression, Support Vector Machine, and Random Forest, while word embeddings were implemented with CNN, BiLSTM, and BiGRU. Results show that BiLSTM achieved the highest accuracy (85.71%), whereas Logistic Regression obtained the highest F1-score (0.7975). The small performance gap (<2%) indicates that traditional machine learning models remain competitive with deep learning approaches under statistically comparable performance. This study provides empirical evidence in the Indonesian e-government context and offers practical insights for monitoring public feedback to improve digital public services.

I. PENDAHULUAN

Adanya pandemi COVID-19 telah menyebabkan guncangan ekonomi yang cukup signifikan di Indonesia, yang memicu lonjakan angka kemiskinan serta penurunan kesejahteraan masyarakat secara luas [1]. Pandemi ini membawa perubahan besar pada hal-hal yang mempengaruhi tingkat kemiskinan, di mana pertumbuhan ekonomi yang melambat berdampak langsung pada peningkatan jumlah penduduk miskin di berbagai wilayah [2]. Angka kemiskinan nasional diperkirakan akan meningkat drastis hingga hampir 14% tanpa intervensi pemerintah melalui program bantuan sosial [3]. Namun, meskipun program bantuan sosial tunai berperan sebagai jaring pengaman, implementasi di lapangan masih menghadapi tantangan besar terkait ketepatan sasaran akibat ketidaksinkronan data [4].

Untuk mengatasi permasalahan tersebut, pemerintah Indonesia bersama Kementerian Sosial menginisiasi berbagai program perlindungan sosial, seperti Program Keluarga Harapan (PKH) dan Bantuan Sosial Tunai [3]. Sebagai usaha untuk mendorong keterbukaan dan akuntabilitas dalam penyaluran berbagai bantuan tersebut, pemerintah mengembangkan layanan digital melalui peluncuran aplikasi Cek Bansos pada platform Google Playstore. Aplikasi ini memiliki fungsi strategis yang memungkinkan masyarakat untuk memeriksa status penerima bantuan, mengajukan sanggahan terhadap penerima yang dianggap tidak layak, serta mengusulkan secara mandiri calon penerima bantuan sosial [5]. Implementasi *e-government* ini menjadi tahapan penting dalam modernisasi layanan sosial yang lebih responsif dan transparan.

Analisis opini publik terhadap Aplikasi Cek Bansos penting dilakukan karena keberhasilan layanan digital

bantuan sosial tidak hanya bergantung pada aspek teknis, namun juga oleh tingkat kepuasan dan kepercayaan masyarakat [6]. Ulasan pengguna memberikan gambaran kemampuan aplikasi dalam memenuhi kebutuhan atau justru menimbulkan hambatan. Oleh karena itu, diperlukan evaluasi berkelanjutan untuk meningkatkan kualitas layanan, kemudahan penggunaan, serta kepercayaan publik melalui sistem yang transparan dan responsif [7].

Berdasarkan pentingnya pemahaman terhadap opini publik tersebut, penelitian ini menggunakan metode analisis sentiment untuk menggali dan mengevaluasi pandangan pengguna berdasarkan ulasan yang diberikan. Analisis sentimen berfokus pada proses pengelompokan teks pada tingkat dokumen, kalimat, atau fitur tertentu untuk mengidentifikasi kecenderungan opini yang terkandung di dalamnya [8]. Setiap ulasan pengguna dapat merepresentasikan sentimen positif atau negative terhadap layanan yang disediakan oleh aplikasi [9]. Platform digital seperti Google Play Store menjadi sarana utama untuk masyarakat umum dalam menyampaikan pengalaman dan keluhan terkait penggunaan layanan publik berbasis aplikasi [10]. Dengan Memanfaatkan teknik *web scraping*, data ulasan pengguna Aplikasi Cek Bansos dapat dikumpulkan dan dianalisis untuk mengidentifikasi pola sentimen yang muncul [11].

Sebagai penguatan terhadap pendekatan tersebut, analisis sentimen yang merupakan bagian dari bidang Natural Language Processing (NLP) memiliki peran strategis dalam mengidentifikasi kecenderungan emosi dan opini publik yang tertuang dalam bentuk teks [12]. Pada domain aplikasi publik, penggunaan metode ini mempermudah pihak pengelola dalam membedakan antara apresiasi dan keluhan pengguna secara sistematis. Integrasi teknologi NLP ini memungkinkan pengolahan ribuan ulasan aplikasi Cek Bansos dilakukan secara cepat dan akurat [13]. Melalui sistem otomatisasi tersebut, pemerintah dapat memperoleh ringkasan persepsi masyarakat secara menyeluruh sebagai dasar perbaikan layanan yang lebih responsif.

Dalam mengkaji penelitian ini, terdapat sejumlah penelitian terdahulu yang relevan dengan pendekatan machine learning pada analisis sentimen. Jahan et al. [14] melakukan analisis komparatif beberapa algoritma machine learning, yaitu Logistic Regression, Support Vector Machine (SVM), dan Random Forest, dengan representasi fitur TF-IDF serta penanganan ketidakseimbangan data menggunakan SMOTE. Berdasarkan hasil pengujian, model Logistic Regression dan SVM memperlihatkan kinerja paling optimal dengan akurasi sebesar 86,22%, sementara Random Forest menunjukkan performa yang cukup kompetitif. Temuan serupa juga ditunjukkan dalam penelitian Putri et al. [15] yang menerapkan TF-IDF, SVM, dan SMOTE untuk analisis sentimen data Twitter. Penelitian tersebut menegaskan bahwa penerapan SMOTE mampu menyeimbangkan distribusi kelas, meskipun peningkatan performa tidak selalu tercermin pada nilai akurasi, sehingga metrik F1-score menjadi indikator evaluasi yang lebih representatif.

Selain pendekatan machine learning, perkembangan riset analisis sentimen juga menunjukkan peningkatan penggunaan pendekatan deep learning yang mampu menangkap representasi semantik teks secara lebih kompleks. Gao et al. [16] mengusulkan model hibrida CNN + BiGRU untuk analisis sentimen teks pendek dengan memanfaatkan word embedding sebagai representasi kata. Hasil pengujian memperlihatkan bahwa kombinasi antara CNN dan BiGRU terbukti mampu meningkatkan kinerja klasifikasi secara signifikan dibandingkan CNN dan LSTM tunggal, dengan peningkatan akurasi hingga lebih dari 4%. Temuan ini menegaskan bahwa CNN cukup efektif dalam mengekstraksi fitur lokal, sementara BiGRU mampu menangkap dependensi konteks dua arah secara lebih optimal. Selain itu, Talaat [17] dalam Journal of Big Data (2023) mengkaji model deep learning berbasis word embedding yang dikombinasikan dengan BiLSTM dan BiGRU, serta menerapkan oversampling pada kelas minoritas, yang terbukti meningkatkan performa klasifikasi dibandingkan algoritma machine learning konvensional.

Meskipun analisis sentimen berbasis machine learning dan deep learning telah banyak diteliti, sebagian besar studi sebelumnya berfokus pada domain umum seperti media sosial atau ulasan produk dan jarang melakukan perbandingan kedua pendekatan tersebut pada objek penelitian yang sama dengan desain eksperimen yang terkontrol. Penelitian yang secara khusus mengkaji ulasan aplikasi e-government di Indonesia masih terbatas, padahal layanan publik digital memiliki karakteristik linguistik yang berbeda, seperti bahasa yang informal, kontekstual, dan didominasi oleh keluhan administratif. Berdasarkan celah tersebut, penelitian ini menyajikan evaluasi komparatif antara model machine learning dan deep learning pada ulasan aplikasi e-government Indonesia dengan menerapkan teknik penyeimbangan data dan representasi fitur yang disesuaikan dengan karakteristik masing-masing pendekatan, guna memperoleh perbandingan kinerja yang lebih adil dan relevan.

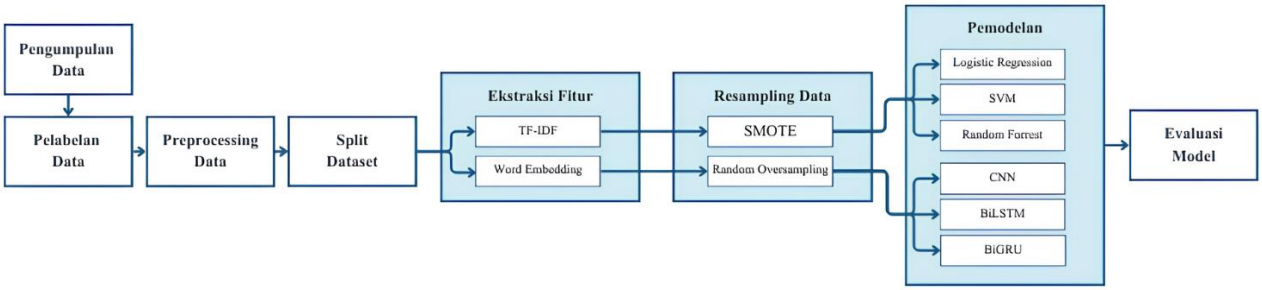
Berdasarkan temuan literatur tersebut, pendekatan *deep learning* diasumsikan memiliki potensi performa yang lebih baik dibandingkan *machine learning* tradisional dalam menangkap konteks semantik ulasan, sementara representasi *word embedding* diperkirakan lebih efektif dibandingkan TF-IDF dalam merepresentasikan variasi bahasa ulasan berbahasa Indonesia. Selain itu, penerapan teknik penyeimbangan data diharapkan dapat meningkatkan kemampuan model dalam mengenali kelas minoritas secara lebih konsisten. Oleh karena itu, penelitian ini berfokus pada evaluasi komparatif algoritma Logistic Regression, SVM, dan Random Forest dengan model *deep learning* CNN, BiLSTM, dan BiGRU dalam klasifikasi sentimen ulasan pengguna aplikasi e-government di Indonesia.

Tujuan penelitian ini adalah menilai kinerja serta efektivitas sejumlah model pembelajaran lanjutan (*advanced learning models*), yang mencakup pendekatan model machine learning dan deep learning, dalam melakukan klasifikasi sentimen terhadap ulasan pengguna aplikasi e-government di

Indonesia. Evaluasi dilakukan dengan membandingkan kinerja setiap model berdasarkan metrik evaluasi yang relevan, sehingga dapat diidentifikasi kelebihan dan keterbatasan masing-masing metode dalam mengolah data ulasan berbasis teks. Penelitian ini juga diharapkan dapat memberikan kontribusi bagi penelitian selanjutnya yang berfokus pada pengembangan dan penerapan analisis sentimen pada layanan e-government di Indonesia.

II. METODE

Metodologi penelitian ini mengikuti alur kerja sistematis yang diilustrasikan pada Gambar 1, mencakup tahap pengumpulan data, prapemrosesan teks, pembagian dataset, penyeimbangan data menggunakan SMOTE dan Random Over-sampling, ekstraksi fitur berbasis TF-IDF dan word embedding, pemodelan menggunakan algoritma machine learning dan deep learning, serta evaluasi performa model.



Gambar 1. Metodologi Penelitian

Secara lebih mendalam, rincian teknis dari setiap tahapan yang disajikan pada Gambar 1 akan dijelaskan pada sub-bab berikut, yang mencakup seluruh tahapan mulai dari proses pengumpulan data hingga prosedur evaluasi model.

A. *Pengumpulan Data*

Tahap pengumpulan data dilakukan dengan mengekstraksi uladan dari pengguna Aplikasi Cek Bansos yang tersedia pada Google Play Store menggunakan teknik *web scraping*, sehingga diperoleh data ulasan dalam jumlah besar yang mencerminkan pengalaman nyata pengguna terhadap aplikasi tersebut [11]. Dataset yang diperoleh terdiri atas teks ulasan pengguna dan nilai rating yang diberikan dalam rentang 1 hingga 5 bintang. Hasil proses *web scraping* menghasilkan 28.697 ulasan sebagai dataset mentah, yang kemudian disimpan dalam format CSV agar dapat diproses lebih lanjur pada tahap prapemrosesan dan pemodelan.

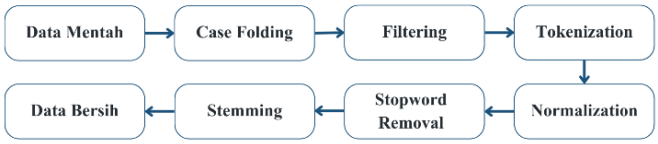
B. *Pelabelan Data*

Penelitian ini menerapkan proses pelabelan data dengan memanfaatkan bahasa pemrograman Python dengan memanfaatkan nilai rating yang diberikan pengguna pada setiap ulasan Aplikasi Cek Bansos. Rating dengan nilai 4 dan 5 digunakan untuk merepresentasikan sentimen positif, sedangkan rating 1 dan 2 merepresentasikan sentimen negatif. Adapun ulasan yang memiliki rating 3 dikecualikan dari dataset karena dianggap memiliki kecenderungan sentimen yang netral atau tidak jelas. Pelabelan dengan rating diterapkan karena efisiensinya dalam memproses dataset skala besar dan kemampuannya merepresentasikan opini pengguna secara umum, sehingga menghasilkan dataset biner yang valid untuk tahap analisis sentiment [18].

Setelah proses pelabelan dan penghapusan ulasan netral, jumlah data yang digunakan dalam penelitian ini menjadi 27.985 ulasan. Meskipun demikian, pendekatan ini berpotensi menimbulkan label noise akibat ketidaksesuaian antara nilai rating dan isi teks. Untuk menilai validitas pelabelan, dilakukan sampling manual terhadap 100 ulasan, yang menunjukkan adanya sebagian kecil label mismatch, seperti ulasan bernada kritik dengan rating tinggi atau sebaliknya. Oleh karena itu, hasil klasifikasi sentimen dalam penelitian ini perlu diinterpretasikan dengan mempertimbangkan keterbatasan pelabelan otomatis tersebut.

C. *Preprocessing Data*

Tahap prapemrosesan dilakukan dengan tujuan untuk mengonversi teks yang masih mentah menjadi teks yang bersih dan terstruktur [19]. Tahap prapemrosesan menggambarkan proses transformasi teks secara bertahap, dimulai dari data ulasan mentah hingga menghasilkan teks akhir yang siap digunakan pada tahap ekstraksi fitur dan pemodelan. Alur prapemrosesan yang diterapkan diilustrasikan secara detail pada Gambar 2.



Gambar 2. Alur Prapemrosesan Data

Tahapan prapemrosesan data yang diterapkan dalam penelitian ini dijelaskan di bawah ini.

1) *Case Folding*

Semua teks ulasan diubah menjadi *lowercase* guna menjaga konsistensi bentuk kata [20].

2) *Filtering*

Elemen teks yang tidak memiliki pengaruh berarti, seperti URL, angka, tanda baca, dan simbol khusus, dihapus untuk meminimalkan noise pada data [21].

3) *Tokenisasi*

Setelah proses pembersihan, teks dipecah menjadi token kata untuk memungkinkan pemrosesan lanjutan pada level kata sehingga memudahkan tahapan normalisasi, penghapusan stopword, dan proses ekstraksi fitur [22].

4) *Normalization*

Mengubah bentuk kata yang ditulis tidak sesuai kaidah, termasuk singkatan dan variasi ejaan, dikonversi menjadi bentuk baku dengan memanfaatkan kamus normalisasi untuk menyamakan kata-kata yang bermakna sama tetapi ditulis dengan variasi kata yang berbeda [23].

5) *Stopword Removal*

Menghilangkan berbagai kata yang sering digunakan namun memiliki nilai kontribusi rendah terhadap klasifikasi sentimen, contohnya adalah kata penghubung dan keterangan umum [24].

6) *Stemming*

Menghilangkan imbuhan untuk memperoleh bentuk dasar kata dan tidak mengubah makna inti kata. Proses stemming dilakukan menggunakan algoritma stemming Bahasa Indonesia sehingga variasi bentuk kata dapat direpresentasikan dalam satu bentuk dasar yang sama [25].

D. Split Dataset

Dataset dibagi menggunakan metode *hold-out* dengan komposisi 80% sebagai data latih dan 20% sebagai data uji secara *stratified* agar distribusi kelas sentimen tetap terjaga. Pada pendekatan machine learning, seluruh data latih digunakan langsung dalam proses pelatihan model. Sementara itu, pada pendekatan deep learning, data latih dibagi kembali menjadi 70% data latih dan 10% data validasi yang digunakan untuk memantau performa model selama proses pelatihan, dengan data uji yang sama digunakan pada seluruh model guna memastikan perbandingan performa yang objektif.

E. Ekstraksi Fitur

Tahap ekstraksi fitur digunakan untuk mentransformasikan teks menjadi representasi numerik [26]. Pada jalur machine learning, digunakan metode TF-IDF. Sementara itu, pada jalur deep learning, teks direpresentasikan menggunakan word embedding [27]. Metode ekstraksi fitur yang diterapkan dalam penelitian ini disesuaikan dengan pendekatan pemodelan yang diterapkan pada masing-masing jalur.

1) *TF-IDF*

Metode Term Frequency–Inverse Document Frequency (TF-IDF) diterapkan sebagai teknik ekstraksi fitur pada model machine learning, termasuk Logistic Regression (LR), Support Vector Machine (SVM), dan Random Forest (RF). Representasi teks dibangun dengan pendekatan unigram dan bigram (n-gram 1–2) guna menangkap makna kata tunggal serta konteks frasa dua kata yang berpotensi memengaruhi sentiment [28]. Proses ini menghasilkan matriks fitur berdimensi tinggi yang selanjutnya digunakan sebagai masukan bagi model machine learning.

2) *Word Embedding*

Pada model deep learning (CNN, BiLSTM, dan BiGRU), teks direpresentasikan menggunakan word embedding dengan mengonversi setiap kata menjadi vektor numerik yang disusun sebagai sekuens sesuai urutan kemunculannya. Proses ini melibatkan tokenizer dan padding untuk menyeragamkan panjang sekuens, sehingga data siap dimanfaatkan sebagai input pada model neural network dan memungkinkan pembelajaran hubungan semantik serta urutan kata dalam teks ulasan [29].

F. Resampling Data

Tahap resampling data dilakukan untuk menangani ketidakseimbangan distribusi jumlah data antar kelas sentimen pada dataset [30]. Ketidakseimbangan kelas dapat mengakibatkan kecenderungan model untuk memihak kelas mayoritas., sehingga diperlukan strategi penyeimbangan data yang diterapkan secara hati-hati tanpa memengaruhi data uji [31]. Berdasarkan pertimbangan tersebut, berikut dijelaskan metode penyeimbangan data yang digunakan dalam penelitian ini.

1) *SMOTE*

Pada model machine learning, penyeimbangan data dilakukan menggunakan metode SMOTE (Synthetic Minority Over-sampling Technique), yang diterapkan setelah proses ekstraksi fitur menggunakan TF-IDF sehingga bekerja pada representasi numerik vector teks. SMOTE hanya diterapkan pada data latih untuk menghasilkan sampel sintetis pada kelas minoritas berdasarkan kedekatan antar data dalam ruang fitur, sementara data uji tidak mengalami penyeimbangan guna mencegah data leakage dan menjaga validitas evaluasi model [32]. Meskipun SMOTE efektif dalam mengatasi ketidakseimbangan kelas, metode ini berpotensi meningkatkan risiko *overfitting* akibat kemiripan data sintetis dengan sampel asli, sehingga evaluasi kinerja model dilakukan menggunakan berbagai metrik klasifikasi, termasuk *precision*, *recall*, dan *F1-score*, untuk memastikan peningkatan performa yang lebih representatif.

2) *Random Oversampling*

Pada model deep learning, penyeimbangan data dilakukan menggunakan metode Random Oversampling pada data latih. Pendekatan ini digunakan untuk meningkatkan jumlah data pada kelas minoritas tanpa mengubah struktur sekuens teks, sehingga sesuai dengan kebutuhan input model neural network berbasis urutan kata [30].

G. Pemodelan

Eksperimen pada penelitian ini dirancang dengan mengombinasikan setiap metode representasi fitur dengan algoritma klasifikasi yang sesuai, sebagaimana dirangkum pada Tabel 1.

TABEL I
SKENARIO EKSPERIMEN REPRESENYASI FITUR DAN MODEL KLASIFIKASI

Representasi Fitur	Model Klasifikasi	Karakteristik
TF-IDF	Logistik Regression	Model baseline untuk klasifikasi sentimen berbasis fitur teks berdimensi tinggi
	SVM	Model linear yang efektif untuk data teks bersifat sparse
	Random Forrest	Model ensemble berbasis pohon keputusan sebagai pembanding
Word Embedding	CNN	Menangkap pola lokal dan frasa penting pada teks ulasan
	BiLSTM	Memodelkan dependensi urutan kata dua arah untuk memahami konteks
	BiGRU	Alternatif ringan BiLSTM dengan waktu pelatihan lebih efisien

Tabel I merangkum skenario eksperimen yang digunakan dalam penelitian ini, di mana setiap metode representasi fitur dipasangkan dengan algoritma klasifikasi yang sesuai dengan karakteristik data. Pada jalur machine learning, representasi fitur TF-IDF dikombinasikan dengan Logistic Regression, Support Vector Machine (SVM) linear, dan Random Forest. Logistic Regression digunakan sebagai model baseline karena kemampuannya menangani fitur teks berdimensi tinggi secara efisien dengan waktu pelatihan yang relatif cepat [33]. SVM linear dipilih karena sangat sesuai untuk data teks berbasis TF-IDF yang bersifat sparse dan mampu menghasilkan batas pemisah optimal antar kelas sentimen, khususnya ketika menggunakan fitur unigram dan bigram [34]. Random Forest digunakan sebagai pembanding untuk mengevaluasi kinerja pendekatan berbasis pohon keputusan pada data teks berdimensi tinggi, meskipun performanya cenderung lebih rendah dibandingkan model linear [35].

Pada jalur deep learning, representasi word embedding dipadukan dengan model CNN, BiLSTM, dan BiGRU. CNN digunakan untuk menangkap pola lokal dalam teks, seperti kombinasi kata atau frasa yang berpengaruh terhadap sentimen, sehingga sesuai untuk ulasan pendek [36]. BiLSTM diterapkan untuk memodelkan dependensi urutan kata secara dua arah guna memahami konteks kalimat secara lebih menyeluruh, termasuk pola negasi dan perubahan makna [37]. Sebagai alternatif yang lebih ringan, BiGRU digunakan karena arsitekturnya lebih sederhana dengan waktu pelatihan yang lebih efisien, namun tetap dapat menghasilkan performa yang sebanding dalam tugas klasifikasi sentimen [38].

H. Evaluasi Model

Untuk menjamin perbandingan yang objektif, performa klasifikasi sentimen pada seluruh model diukur menggunakan dataset uji yang seragam. Evaluasi ini dilakukan dengan mengandalkan *confusion matrix* serta beragam metrik lainnya untuk meninjau tingkat presisi dan kemampuan tiap model dalam mengidentifikasi setiap kategori sentimen.

Confusion matrix digunakan untuk membandingkan label aktual dan prediksi hasil klasifikasi. [39]. Matriks ini tersusun atas empat komponen utama, yaitu True Positive (TP), True Negative (TN), False Positive (FP), dan False Negative (FN), yang digunakan sebagai dasar perhitungan metrik evaluasi lainnya.

TABEL II
CONFUSION MATRIX

Actual	Predicted	
	Positif	**Negatif**
Positif	TP	FN
Negatif	FP	TN

Akurasi menunjukkan proporsi prediksi yang tepat, yaitu gabungan true positive dan true negative, dibandingkan dengan keseluruhan sampel. yang diklasifikasikan. [39], sebagaimana ditunjukkan pada Persamaan (1).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

Presisi pada pengklasifikasi biner dapat diartikan sebagai rasio jumlah sampel positif yang diprediksi dengan benar terhadap seluruh prediksi positif, yang dalam confusion matrix direpresentasikan oleh nilai true positive pada kolom prediksi positif. [39]. Secara matematis, metrik ini dirumuskan pada Persamaan (2).

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

Recall menunjukkan kemampuan model dalam mengenali seluruh data yang termasuk ke dalam suatu kelas tertentu [39], sebagaimana ditunjukkan pada Persamaan (3).

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

F1-score didefinisikan sebagai rata-rata harmonik antara precision dan recall, dan digunakan sebagai metrik utama dalam penelitian ini karena lebih seimbang pada data tidak seimbang [40]. Secara matematis, metrik ini dirumuskan pada Persamaan (4).

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

III. HASIL DAN PEMBAHASAN

A. Dataset

Data yang dikumpulkan dalam penelitian ini berasal dari ulasan pengguna Aplikasi Cek Bansos yang diperoleh dari Google Play Store melalui penerapan teknik *web scraping.* Dataset ini selanjutnya disimpan dalam bentuk CSV. Jumlah keseluruhan data yang digunakan adalah 28.697 data. Contoh data hasil pengumpulan ditampilkan pada Tabel III.

TABEL III
SAMPEL DATA MENTAH

Content	Score
Daftar akun tapi tidak bisa karena nama dan nik sudah digunakan,padahal belum pernah daftar	1
aplikasi GK guna sama sekali mau mau bikin akun udh kembali LGI ,,,,sama sekali GK berguna 😠😠	1
aplikasi gak jelas, tiap mau daftar gak bisa 🙏😭	2
baik sangat mudah untuk mendapat impormasi terkait sosial	5
bagus sangat mudah utk cek bantuan	4

B. Pelabelan Data

Pada tahap pelabelan, sentimen ulasan pengguna Aplikasi Cek Bansos ditentukan secara otomatis dengan memanfaatkan nilai rating yang tersedia di Google Play Store. Ulasan dengan rating 1 dan 2 diklasifikasikan sebagai sentimen negatif, sedangkan rating 4 dan 5 dikategorikan sebagai sentimen positif. Pendekatan ini digunakan untuk menghasilkan dataset sentimen biner yang konsisten dan siap digunakan pada tahap analisis selanjutnya. Hasil pelabelan data ditunjukkan pada Tabel IV.

TABEL IV
HASIL PELABELAN DATA

Content	Label
Daftar akun tapi tidak bisa karena nama dan nik sudah digunakan,padahal belum pernah daftar	Negatif
aplikasi GK guna sama sekali mau mau bikin akun udh kembali LGI ,,,,sama sekali GK berguna 😠😠	Negatif
aplikasi gak jelas, tiap mau daftar gak bisa 🙏😭	Negatif
baik sangat mudah untuk mendapat impormasi terkait sosial	Positif
bagus sangat mudah utk cek bantuan	Positif

C. Preprocessing Data

Mengingat data ulasan pengguna bersifat tidak terstruktur dan mengandung banyak variasi bahasa informal, preprocessing menjadi langkah penting untuk mengurangi gangguan pada data serta meningkatkan kualitas data sebelum dilakukan ekstraksi fitur dan pemodelan.

1) *Case Folding*

Semua teks ulasan diseragamkan ke dalam format huruf kecil melalui proses *case folding* untuk menghindari pengaruh variasi huruf kapital terhadap hasil analisis. Hasil dari standarisasi teks ini dapat dilihat pada Tabel V.

TABEL V
PERBANDINGAN SEBELUM DAN SESUDAH CASE FOLDING

Sebelum Case Folding	Setelah Case Folding
Daftar akun tapi tidak bisa karena nama dan nik sudah digunakan,padahal belum pernah daftar	daftar akun tapi tidak bisa karena nama dan nik sudah digunakan,padahal belum pernah daftar
aplikasi GK guna sama sekali mau mau bikin akun udh kembali LGI ,,,,sama sekali GK berguna 😠😠	aplikasi gk guna sama sekali mau mau bikin akun udh kembali lgi ,,,,sama sekali gk berguna 😠😠
aplikasi gak jelas, tiap mau daftar gak bisa 🙏😭	aplikasi gak jelas, tiap mau daftar gak bisa 🙏😭
baik sangat mudah untuk mendapat impormasi terkait sosial	baik sangat mudah untuk mendapat impormasi terkait sosial
bagus sangat mudah utk cek bantuan	bagus sangat mudah utk cek bantuan,bagus sangat mudah utk cek bantuan

2) *Filtering*

Tahap filtering menghasilkan teks yang lebih bersih dengan menghapus komponen teks yang tidak memiliki relevansi, termasuk URL, angka, tanda baca, dan simbol khusus. Hasil filtering menunjukkan berkurangnya noise pada data teks. Hasil *filtering* dapat dilihat dalam Tabel VI.

TABEL VI
PERBANDINGAN SEBELUM DAN SESUDAH FILTERING

Sebelum Filtering	Setelah Filtering
daftar akun tapi tidak bisa karena nama dan nik sudah digunakan,padahal belum pernah daftar	daftar akun tapi tidak bisa karena nama dan nik sudah digunakan padahal belum pernah daftar
aplikasi gk guna sama sekali mau mau bikin akun udh kembali lgi ,,,,sama sekali gk berguna 😠😠	aplikasi gk guna sama sekali mau mau bikin akun udh kembali lgi sama sekali gk berguna
aplikasi gak jelas, tiap mau daftar gak bisa 🙏😭	aplikasi gak jelas tiap mau daftar gak bisa
baik sangat mudah untuk mendapat impormasi terkait sosial	baik sangat mudah untuk mendapat impormasi terkait sosial
bagus sangat mudah utk cek bantuan,bagus sangat mudah utk cek bantuan	bagus sangat mudah utk cek bantuan

3) Tokenisasi

Di tahap tokenisasi, teks ulasan dipecah yang kemudian menjadi satuan kata atau token. Proses ini mempermudah

analisis lanjutan karena setiap kata dapat diproses secara terpisah, serta menjadi dasar untuk tahapan normalisasi dan stopword removal. Hasil tokenisasi disajikan dalam Tabel VII.

TABEL VII
PERBANDINGAN SEBELUM DAN SESUDAH TOKENISASI

Sebelum Tokenisasi	Setelah Tokenisasi
daftar akun tapi tidak bisa karena nama dan nik sudah digunakan padahal belum pernah daftar	['daftar', 'akun', 'tapi', 'tidak', 'bisa', 'karena', 'nama', 'dan', 'nik', 'sudah', 'digunakan', 'padahal', 'belum', 'pernah', 'daftar']
aplikasi gk guna sama sekali mau mau bikin akun udh kembali lgi sama sekali gk berguna	['aplikasi', 'gk', 'guna', 'sama', 'sekali', 'mau', 'mau', 'bikin', 'akun', 'udh', 'kembali', 'lgi', 'sama', 'sekali', 'gk', 'berguna']
aplikasi gak jelas tiap mau daftar gak bisa	['aplikasi', 'gak', 'jelas', 'tiap', 'mau', 'daftar', 'gak', 'bisa']
baik sangat mudah untuk mendapat impormasi terkait sosial	['baik', 'sangat', 'mudah', 'untuk', 'mendapat', 'impormasi', 'terkait', 'sosial']
bagus sangat mudah utk cek bantuan	['bagus', 'sangat', 'mudah', 'utk', 'cek', 'bantuan']

4) Normalisasi

Untuk meminimalkan variasi kata yang berlebihan, dilakukan normalisasi dengan mengubah singkatan dan ejaan tidak baku menjadi bentuk standar. Hal ini memastikan konsistensi representasi makna dalam data. Adapun hasil normalisasi tersebut dapat dilihat pada Tabel VIII.

TABEL VIII
PERBANDINGAN SEBELUM DAN SESUDAH NORMALISASI

Sebelum Normalisasi	Setelah Normalisasi
['daftar', 'akun', 'tapi', 'tidak', 'bisa', 'karena', 'nama', 'dan', 'nik', 'sudah', 'digunakan', 'padahal', 'belum', 'pernah', 'daftar']	['daftar', 'akun', 'tapi', 'tidak', 'bisa', 'karena', 'nama', 'dan', 'nik', 'sudah', 'digunakan', 'padahal', 'belum', 'pernah', 'daftar']
['aplikasi', 'gk', 'guna', 'sama', 'sekali', 'mau', 'mau', 'bikin', 'akun', 'udh', 'kembali', 'lgi', 'sama', 'sekali', 'gk', 'berguna']	['aplikasi', 'tidak', 'guna', 'sama', 'sekali', 'mau', 'mau', 'bikin', 'akun', 'udh', 'kembali', 'lgi', 'sama', 'sekali', 'tidak', 'berguna']
['aplikasi', 'gak', 'jelas', 'tiap', 'mau', 'daftar', 'gak', 'bisa']	['aplikasi', 'gak', 'jelas', 'tiap', 'mau', 'daftar', 'gak', 'bisa']
['baik', 'sangat', 'mudah', 'untuk', 'mendapat', 'impormasi', 'terkait', 'sosial']	['baik', 'sangat', 'mudah', 'untuk', 'mendapat', 'impormasi', 'terkait', 'sosial']
['bagus', 'sangat', 'mudah', 'utk', 'cek', 'bantuan']	['bagus', 'sangat', 'mudah', 'utk', 'cek', 'bantuan']

5) *Stopword Removal*

S*topword removal* dilakukan dengan menghilangkan berbagai kata yang banyak digunakan tetapi tidak mempunyai pengaruh yang siginifikan terhadap klasifikasi sentimen. Hasil dari *stopword removal* disajikan dalam Tabel IX.

TABEL IX
PERBANDINGAN SEBELUM DAN SESUDAH STOPWORD REMOVAL

Sebelum Stopword Removal	Setelah Stopword Removal
['daftar', 'akun', 'tapi', 'tidak', 'bisa', 'karena', 'nama', 'dan', 'nik', 'sudah', 'digunakan', 'padahal', 'belum', 'pernah', 'daftar']	['daftar', 'akun', 'nama', 'nik', 'daftar']
['aplikasi', 'tidak', 'guna', 'sama', 'sekali', 'mau', 'mau', 'bikin', 'akun', 'udh', 'kembali', 'lgi', 'sama', 'sekali', 'tidak', 'berguna']	['aplikasi', 'bikin', 'akun', 'udh', 'lgi', 'berguna']
['aplikasi', 'gak', 'jelas', 'tiap', 'mau', 'daftar', 'gak', 'bisa']	['aplikasi', 'gak', 'daftar', 'gak']
['baik', 'sangat', 'mudah', 'untuk', 'mendapat', 'impormasi', 'terkait', 'sosial']	['mudah', 'impormasi', 'terkait', 'sosial']
['bagus', 'sangat', 'mudah', 'utk', 'cek', 'bantuan']	['bagus', 'mudah', 'utk', 'cek', 'bantuan']

6) *Stemming*

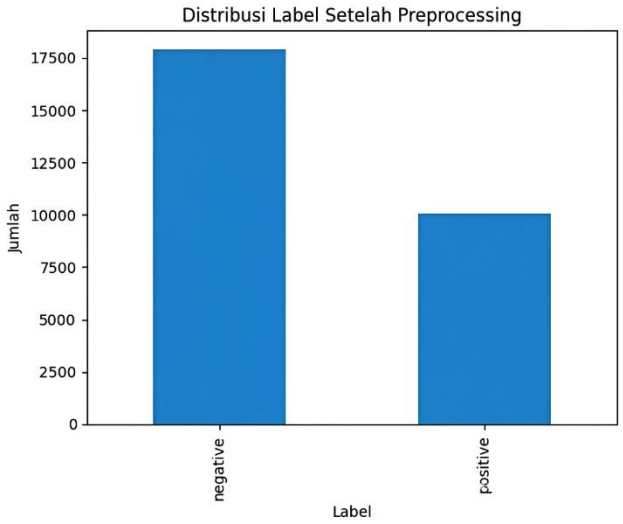
Stemming adalah tahap dimana kata dikembalikan ke dalam bentuk dasar melalui penghapusan imbuhan. Hasil stemming menunjukkan bahwa berbagai bentuk kata berimbuhan dapat disatukan ke dalam satu bentuk dasar, sehingga memperkaya representasi fitur dan meningkatkan konsistensi data teks. Tabel X menyajikan hasil proses *stemming.*

TABEL X
PERBANDINGAN SEBELUM DAN SESUDAH STEMMING

Sebelum Stemming	Setelah Stemming
['daftar', 'akun', 'nama', 'nik', 'daftar']	['daftar', 'akun', 'nama', 'nik', 'daftar']
['aplikasi', 'bikin', 'akun', 'udh', 'lgi', 'berguna']	['aplikasi', 'bikin', 'akun', 'udh', 'lgi', 'guna']
['aplikasi', 'gak', 'daftar', 'gak']	['aplikasi', 'gak', 'daftar', 'gak']
['mudah', 'impormasi', 'terkait', 'sosial']	['mudah', 'impormasi', 'kait', 'sosial']
['bagus', 'mudah', 'utk', 'cek', 'bantuan']	['bagus', 'mudah', 'utk', 'cek', 'bantu']

Setelah tahap preprocessing, kumpulan data yang digunakan pada penelitian ini berjumlah 27.985 data, yang

terdiri dari 17.909 data sentimen positif dan 10.076 data sentimen negatif. Distribusi data di setiap kelas sentimen setelah preprocessing ditampilkan pada Gambar 3, yang memperlihatkan adanya ketidakseimbangan kelas.



Gambar 3. Distribusi Label Setelah Preprocessing

D. Split Data

Dataset dibagi menggunakan metode *hold-out* dengan proporsi 80% sebagai data latih dan 20% sebagai data uji secara *stratified* untuk mempertahankan distribusi kelas sentimen. Pada model machine learning, seluruh data latih dimanfaatkan dalam proses pelatihan model. Sementara itu, pada model deep learning, data latih dibagi kembali menjadi 70% data latih dan 10% data validasi dimanfaatkan sebagai acuan untuk mengevaluasi performa model selama proses pelatihan berlangsung, dengan data uji yang sama digunakan pada seluruh model guna memastikan perbandingan performa yang objektif.

E. Ekstraksi Fitur

Pada model machine learning, teks hasil preprocessing diekstraksi menggunakan metode TF-IDF dengan pendekatan unigram dan bigram. Proses ini menghasilkan matriks fitur berdimensi tinggi yang merepresentasikan bobot kepentingan kata dan frasa dalam setiap ulasan. Bobot TF-IDF yang tinggi menunjukkan kata atau frasa yang lebih spesifik dan relevan terhadap suatu ulasan. Contoh representasi fitur TF-IDF ditunjukkan pada Tabel XI, yang menggambarkan bobot kata pada beberapa dokumen ulasan.

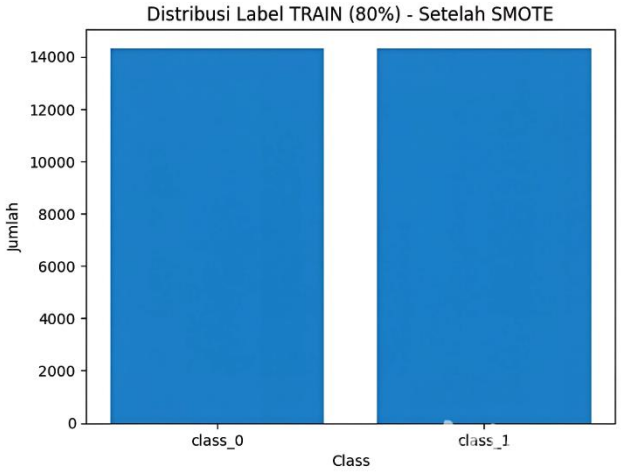
TABEL XI
SAMPEL SKOR TF-IDF

Kata	Skor
tiap mau	0.567826
tiap	0.463431
aplikasi jelas	0.371133
mau daftar	0.348054
jelas	0.302361
mau	0.237119
daftar	0.189343
aplikasi	0.142918

Hasil representasi teks menggunakan word embedding menunjukkan bahwa setiap ulasan yang telah melalui preprocessing diubah menjadi deret indeks kata (sequence) berdasarkan kamus kata yang dibangun pada data latih. Sebagai contoh, kalimat "aplikasi jelas tiap mau daftar" direpresentasikan sebagai urutan indeks [2, 30, 290, 7, 4], di mana setiap angka merepresentasikan satu kata dalam vocabulary. Selanjutnya, sequence tersebut diproses menggunakan padding sehingga memiliki panjang tetap, yaitu 100 token, menghasilkan bentuk data (1, 100). Secara keseluruhan, data hasil embedding memiliki dimensi (19.589, 100) untuk data latih, (2.799, 100) untuk data validasi, dan (5.597, 100) untuk data uji, yang memungkinkan model deep learning memproses input teks dengan panjang yang seragam.

F. Resampling Data

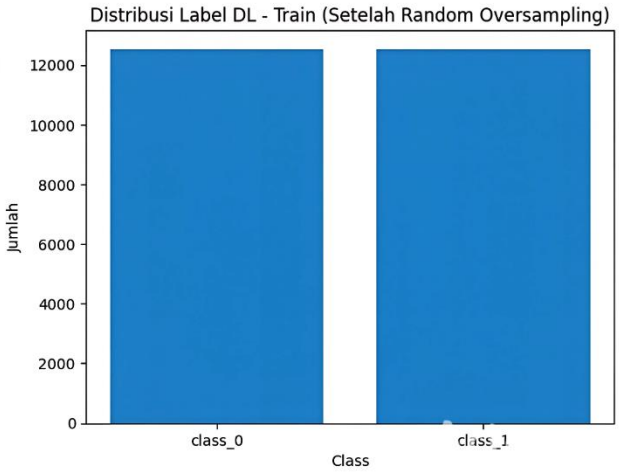
Distribusi label pada data latih sebelum dilakukan resampling menunjukkan ketidakseimbangan kelas, dengan 14.327 data kelas negatif (0) dan 8.061 data kelas positif (1). Untuk mengatasi permasalahan tersebut, dilakukan penyeimbangan data dengan menggunakan metode SMOTE yang diterapkan hanya pada data latih, sehingga jumlah data pada masing-masing kelas menjadi seimbang, yaitu 14.327 data untuk kelas negatif dan 14.327 data untuk kelas positif. Distribusi data latih setelah proses resampling ditunjukkan pada Gambar 5, yang memperlihatkan keseimbangan kelas sebagai dasar yang lebih optimal untuk proses pelatihan model.



Gambar 4. Distribusi Dataset Setelah SMOTE

Pada jalur deep learning, distribusi label pada data latih sebelum dilakukan resampling menunjukkan ketidakseimbangan kelas, dengan 12.536 data kelas negatif (0) dan 7.053 data kelas positif (1). Untuk mengatasi

ketidakseimbangan tersebut, diterapkan metode Random Oversampling (ROS) yang dilakukan hanya pada data latih, sehingga total data pada setiap kelas menjadi seimbang, yaitu 12.536 data untuk kelas negatif dan 12.536 data untuk kelas positif. Distribusi data latih setelah proses resampling ditunjukkan pada Gambar 6.



Gambar 5. Distribusi Dataset Setelah Random Oversampling

G. Modelling

Berdasarkan hasil evaluasi yang disajikan pada Tabel XII, seluruh algoritma yang diuji, baik *machine learning* maupun *deep learning*, menunjukkan performa klasifikasi yang relatif tinggi dengan nilai akurasi berada pada kisaran 84,03% hingga 85,71%. Temuan ini mengindikasikan bahwa pendekatan pembelajaran mesin secara umum efektif dalam mengklasifikasikan sentimen ulasan pengguna Aplikasi *Cek Bansos*. Meskipun demikian, perbedaan kinerja antar model pada metrik *precision*, *recall*, dan *F1-score* menunjukkan adanya variasi karakteristik prediksi yang perlu dianalisis lebih lanjut.

TABEL XII
METRIK EVALUASI DEEP LEARNING

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression	0.8546	0.7995	0.7955	0.7975
SVM	0.8403	0.7735	0.7866	0.7800
Random Forest	0.8526	0.8269	0.7469	0.7849
CNN	0.8551	0.8209	0.7643	0.7916
BiLSTM	0.8571	0.8369	0.7489	0.7905
BiGRU	0.8547	0.8042	0.7886	0.7963

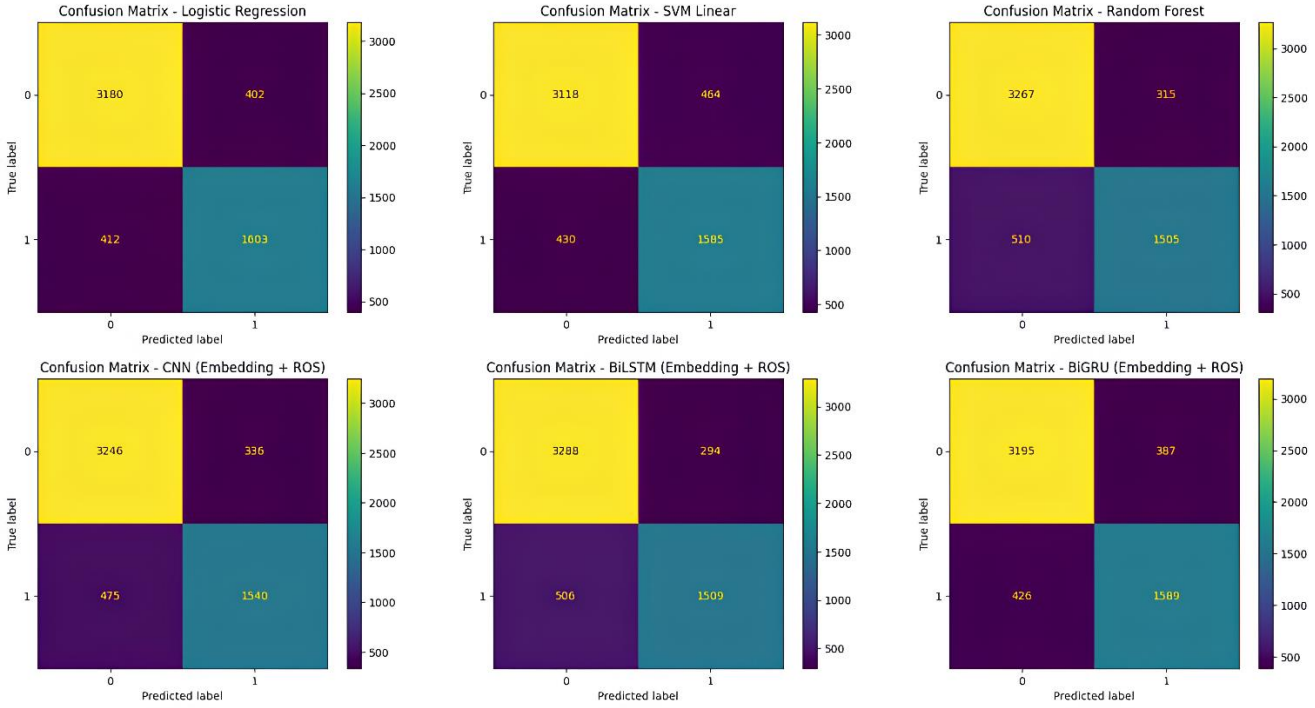
Pada kelompok *machine learning*, Logistic Regression menunjukkan performa paling konsisten dengan nilai akurasi sebesar 85,46% dan nilai F1-score tertinggi sebesar 0,7975 di antara seluruh model yang diuji. Nilai F1-score yang tinggi mencerminkan keseimbangan yang baik antara *precision* dan *recall*, sehingga model ini mampu menjaga proporsi prediksi antar kelas secara lebih stabil. Kombinasi representasi TF-IDF dan penyeimbangan data menggunakan SMOTE terbukti efektif dalam membantu Logistic Regression mengenali pola sentimen pada teks ulasan yang bersifat ringkas dan informal. Sementara itu, Random Forest memperoleh nilai *precision* tertinggi (0,8269) namun memiliki *recall* yang lebih rendah, yang menunjukkan kecenderungan model dalam mengurangi *false positive* dengan konsekuensi meningkatnya *false negative*. Model SVM Linear menunjukkan performa yang cukup stabil, meskipun berada sedikit di bawah Logistic Regression pada hampir seluruh metrik evaluasi.

Pada kelompok *deep learning*, BiLSTM mencapai nilai akurasi tertinggi sebesar 85,71%, yang menunjukkan kemampuannya dalam menangkap dependensi sekuensial dua arah pada teks ulasan. Meskipun akurasi BiLSTM meningkat, nilai *recall* yang lebih rendah dibandingkan model lainnya menunjukkan bahwa model ini masih kurang optimal dalam mengidentifikasi kelas minoritas. Hal ini membuktikan bahwa akurasi tinggi tidak menjamin kemampuan deteksi yang baik pada data yang langka. Sebaliknya, BiGRU menunjukkan nilai *recall* tertinggi (0,7886) dan F1-score tertinggi di antara model deep learning (0,7963), yang mengindikasikan sensitivitas yang lebih baik terhadap kelas positif, meskipun dengan nilai *precision* yang lebih rendah. Model CNN menunjukkan performa yang relatif seimbang pada seluruh metrik, mencerminkan kemampuannya dalam mengekstraksi fitur lokal dari teks, meskipun kurang optimal dalam menangkap konteks sekuensial yang lebih panjang.

Secara keseluruhan, hasil pada Tabel XII menunjukkan bahwa tidak terdapat satu model yang secara dominan unggul pada seluruh metrik evaluasi. Model *deep learning* cenderung unggul dari sisi akurasi, sementara model *machine learning*, khususnya Logistic Regression, menunjukkan keseimbangan prediksi yang lebih baik yang tercermin pada nilai *F1-score*. Selisih performa antar model yang relatif kecil mengindikasikan bahwa peningkatan kompleksitas model tidak selalu diikuti oleh peningkatan kinerja yang signifikan. Temuan ini menjadi dasar untuk analisis lebih lanjut pada subbab berikutnya, khususnya terkait uji signifikansi statistik dan pembahasan karakteristik kesalahan klasifikasi.

H. Evaluasi

Evaluasi *confusion matrix* dilakukan untuk menganalisis karakteristik kesalahan klasifikasi yang tidak sepenuhnya tercermin dari metrik agregat seperti akurasi dan F1-score, khususnya terkait kesalahan *false positive* dan *false negative* pada klasifikasi sentimen layanan publik. Gambar 6 menampilkan *confusion matrix* dari seluruh model *machine learning* dan *deep learning* yang digunakan dalam penelitian ini sebagai dasar untuk menganalisis pola kesalahan klasifikasi secara lebih mendalam.



Gambar 6. Confusion Matrix Semua Model

Pada kelompok *machine learning*, Logistic Regression menunjukkan distribusi kesalahan yang relatif seimbang antara *false positive* (402) dan *false negative* (412), yang mencerminkan stabilitas prediksi dan selaras dengan perolehan nilai F1-score tertinggi pada kelompok ini. SVM Linear menghasilkan jumlah kesalahan yang lebih tinggi, sedangkan Random Forest memiliki *false positive* terendah (315) namun *false negative* tertinggi (510), yang mengindikasikan kecenderungan model dalam mengorbankan sensitivitas kelas positif.

Pada kelompok *deep learning*, BiLSTM mencatat *false positive* terendah (294) dan *true negative* tertinggi, sehingga menghasilkan akurasi tertinggi. Namun, jumlah *false negative* yang relatif tinggi (506) menunjukkan keterbatasan model dalam mengenali sebagian ulasan positif, yang berdampak pada nilai *recall*. CNN menunjukkan performa yang cukup seimbang, sementara BiGRU memiliki distribusi kesalahan yang lebih stabil dengan *false negative* lebih rendah, sehingga menghasilkan nilai *recall* dan F1-score yang kompetitif.

Secara keseluruhan, hasil *confusion matrix* menunjukkan bahwa seluruh model mampu mengklasifikasikan kelas mayoritas dengan baik, namun masih menghadapi tantangan pada kelas minoritas. Model *deep learning* cenderung unggul dalam menekan *false positive*, sedangkan model *machine learning* menunjukkan keseimbangan kesalahan yang lebih konsisten. Hasil ini membuktikan bahwa performa sebuah model tidak semata-mata bergantung pada tingkat kerumitan arsitekturnya, melainkan juga sangat dipengaruhi oleh sifat data ulasan yang cenderung ringkas dan bermakna lugas.

I. Uji Statistik dan Komputasi

Uji signifikansi statistik dilakukan menggunakan McNemar test untuk membandingkan kinerja model terbaik dari pendekatan *machine learning* dan *deep learning*, yaitu Logistic Regression dan BiLSTM, berdasarkan hasil prediksi pada data uji yang sama. Hasil pengujian menunjukkan nilai *p-value* sebesar 0,4416, yang berada di atas tingkat signifikansi $\alpha = 0{,}05$, sehingga tidak terdapat perbedaan performa yang signifikan secara statistik antara kedua model. Temuan ini mengindikasikan bahwa selisih kinerja yang diamati pada metrik evaluasi tidak dapat dianggap sebagai keunggulan yang konsisten secara statistik.

Selain uji statistik, perbandingan efisiensi komputasi menunjukkan bahwa Logistic Regression hanya memerlukan waktu pelatihan sebesar 0,51 detik, sedangkan BiLSTM membutuhkan waktu pelatihan yang jauh lebih lama, yaitu 242,51 detik. Dengan mempertimbangkan kinerja yang sebanding secara statistik serta perbedaan waktu pelatihan yang signifikan, hasil ini menunjukkan bahwa model *machine learning* lebih efisien untuk diterapkan pada sistem pemantauan sentimen layanan publik berbasis e-government, sementara model *deep learning* tetap relevan untuk analisis yang membutuhkan pemodelan konteks teks yang lebih kompleks.

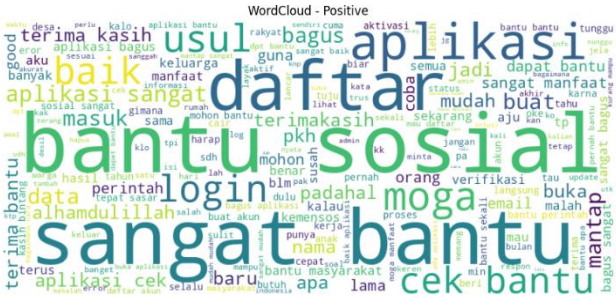
J. Analisis dan Diskusi

Analisis performa menunjukkan tidak adanya perbedaan yang mencolok antara metode *machine learning* dan *deep learning* untuk klasifikasi sentimen ini. Meski BiLSTM unggul dalam angka akurasi, uji McNemar mengonfirmasi bahwa keunggulan tersebut tidak signifikan dibandingkan dengan *Logistic Regression*, sehingga kedua model dianggap memiliki efektivitas yang sebanding. Temuan ini

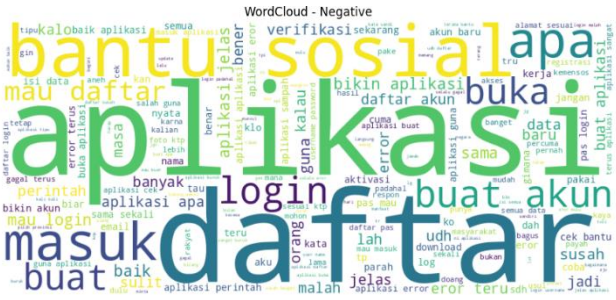
mengindikasikan bahwa peningkatan kompleksitas arsitektur deep learning tidak selalu menghasilkan peningkatan performa yang konsisten, khususnya pada data ulasan aplikasi e-government yang umumnya bersifat singkat, eksplisit, dan berfokus pada permasalahan layanan. Representasi TF-IDF yang dikombinasikan dengan Logistic Regression terbukti cukup efektif dalam menangkap pola sentimen polar, sehingga menghasilkan keseimbangan precision dan recall yang stabil.

Dari perspektif implementasi, hasil penelitian ini menunjukkan bahwa sistem analisis sentimen berbasis machine learning dapat dimanfaatkan oleh pengembang aplikasi dan instansi pemerintah sebagai alat pemantauan opini publik yang efisien dan mudah diintegrasikan. Model dengan kompleksitas rendah dan waktu pelatihan yang singkat memungkinkan pemantauan sentimen dilakukan secara berkala untuk mengidentifikasi pola keluhan dominan, mengevaluasi kualitas layanan digital, serta mendukung pengambilan keputusan dan perbaikan kebijakan layanan e-government secara lebih responsif dan berbasis data.

Untuk melengkapi evaluasi kuantitatif, dilakukan analisis kualitatif melalui visualisasi word cloud pada ulasan positif dan negative yang disajikan pada gambar 7 dan 8. Word cloud sentimen positif didominasi oleh istilah yang berkaitan dengan kemudahan penggunaan dan manfaat aplikasi, sementara sentimen negatif menampilkan kata-kata yang berasosiasi dengan kendala teknis, proses verifikasi, dan kesalahan data. Pola ini konsisten dengan hasil klasifikasi dan confusion matrix, yang menunjukkan bahwa kesalahan model umumnya muncul pada ulasan kontekstual dengan keluhan administratif, sehingga menegaskan kompleksitas tantangan dalam layanan e-government.



Gambar 7. WordCloud Positif



Gambar 8. WordCloud Negatif

IV. KESIMPULAN

Penelitian ini melakukan perbandingan kinerja antara algoritma *machine learning* dan *deep learning* dalam konteks analisis sentimen. Fokus utama evaluasi dilakukan pada ulasan pengguna Aplikasi Cek Bansos, yang merupakan salah satu representasi layanan *e-government* di Indonesia. Hasil eksperimen menunjukkan bahwa BiLSTM mencapai akurasi tertinggi (85,71%), sedangkan Logistic Regression menghasilkan nilai F1-score tertinggi (0,7975), yang mencerminkan keseimbangan precision dan recall yang lebih stabil. Namun, berdasarkan uji McNemar, perbedaan kinerja kedua model tersebut tidak signifikan secara statistik ($p > 0,05$), sehingga dapat disimpulkan bahwa pendekatan machine learning dan deep learning memiliki performa yang sebanding (comparable) pada konteks klasifikasi sentimen ulasan aplikasi e-government.

Dari sisi efisiensi, Logistic Regression menunjukkan waktu pelatihan yang jauh lebih singkat dibandingkan BiLSTM, yang mengindikasikan bahwa peningkatan kompleksitas model deep learning tidak selalu sebanding dengan peningkatan manfaat performa. Secara praktis, hasil klasifikasi sentimen ini dapat dimanfaatkan oleh pengembang aplikasi dan instansi pemerintah sebagai alat pemantauan opini publik secara berkala, untuk mengidentifikasi pola keluhan dominan, mengevaluasi aspek layanan yang paling sering dikeluhkan masyarakat, serta mendukung perumusan kebijakan dan perbaikan layanan digital yang lebih responsif dan berbasis data. Studi ini memiliki beberapa batasan, di antaranya ketergantungan pada satu sumber data aplikasi dan metode pelabelan otomatis yang berbasis nilai *rating*. Untuk meningkatkan kualitas penelitian selanjutnya, disarankan untuk mengeksplorasi lebih banyak objek kajian serta menggunakan teknik pelabelan dan klasifikasi multikelas yang lebih komprehensif.

DAFTAR PUSTAKA

[1] N. Desripa *et al.*, “Volume 2 ; Nomor 2 ; Februari 2024,” pp. 7–12, doi: 10.59435/gjmi.v2i2.275.

[2] M. H. Asqalani and B. Fitanto, “Analisis Determinan Tingkat Kemiskinan Pada Masa Pre-Post Pandemi Covid-19 Di Provinsi Jawa Timur,” *Journal of Development Economic and Social Studies*, vol. 3, no. 4, pp. 1098–1114, Oct. 2024, doi: 10.21776/jdess.2024.03.4.10.

[3] S. J. A. Samuda and E. Suprihartiningsih, “COVID-19 Social Assistance Program and Poverty: Evidence from Indonesia,” *Jurnal Ekonomi Pembangunan*, vol. 20, no. 2, pp. 125–134, Jan. 2023, doi: 10.29259/jep.v20i2.19088.

[4] A. Rahma *et al.*, “Menggali Akar Permasalahan: Kajian Mendalam Terhadap Data Kemiskinan Dan Mekanisme Bantuan Sosial,” *Jurnal Penelitian Nusantara*, vol. 1, pp. 192–198, 2025, doi: 10.59435/menulis.v1i3.93.

[5] Dwiarto, Raden. "Inovasi Penyaluran Jaminan Sosial Tepat Sasaran Melalui Kebijakan Pengelolaan Anggaran Data Terpadu Kesejahteraan Sosial (DTKS) Dan Pemanfaatan Aplikasi" Cek Bansos"." Prosiding Seminar Nasional Unimus. Vol. 6. 2023.

[6] M. Khanif Hidayatul Khasan, T. Riasih, and E. Gunawan Wibisono, “Persepsi Warga Masyarakat Terhadap Aplikasi Cek Bansos Dalam Penyaluran Bantuan Sosial Di Desa Kalikidang Kecamatan Sokaraja Kabupaten Banyumas Jawa Tengah,” 2024, doi: 10.31595/10.31595/lindayasos.v7i1.1604.

[7] D. Salsabila, D. Chusnulitta Jatnika, and F. P. Firsanty, "Focus : Jurnal Pekerjaan Sosial Pemanfaatan Teknologi Digital Dalam Layanan Sosial Di Indonesia: Tinjauan Sistematis," vol. 8, no. 1, pp. 50–59, 2025, doi: 10.24198/focus.v8i1.63672.

[8] T. Shaik, X. Tao, C. Dann, H. Xie, Y. Li, and L. Galligan, "Sentiment analysis and opinion mining on educational data: A survey," *Natural Language Processing Journal*, vol. 2, p. 100003, Mar. 2023, doi: 10.1016/j.nlp.2022.100003.

[9] Q. A. Xu, V. Chang, and C. Jayne, "A systematic review of social media-based sentiment analysis: Emerging trends and challenges," *Decision Analytics Journal*, vol. 3, p. 100073, Jun. 2022, doi: 10.1016/j.dajour.2022.100073.

[10] A. Noor, M. D. Mehmood, and T. Das, "End Users' Perspective of Performance Issues in Google Play Store Reviews," in *Product-Focused Software Process Improvement*, D. Taibi, M. Kuhrmann, T. Mikkonen, J. Klünder, and P. Abrahamsson, Eds., Cham: Springer International Publishing, 2022, pp. 603–609.

[11] A. Yasin, R. Fatima, A. N. Ghazi, and Z. Wei, "Python data odyssey: Mining user feedback from google play store," *Data Brief*, vol. 54, Jun. 2024, doi: 10.1016/j.dib.2024.110499.

[12] J. R. Jim, M. A. R. Talukder, P. Malakar, M. M. Kabir, K. Nur, and M. F. Mridha, "Recent advancements and challenges of NLP-based sentiment analysis: A state-of-the-art review," Mar. 01, 2024, *Elsevier Ltd*. doi: 10.1016/j.nlp.2024.100059.

[13] N. Malik and M. Bilal, "Natural language processing for analyzing online customer reviews: a survey, taxonomy, and open research challenges," *PeerJ Comput Sci*, vol. 10, 2024, doi: 10.7717/PEERJ-CS.2203.

[14] Israt Jahan, Md Nakibul Islam, Md Mahadi Hasan, and Md Rafiuddin Siddiky, "Comparative analysis of machine learning algorithms for sentiment classification in social media text," *World Journal of Advanced Research and Reviews*, vol. 23, no. 3, pp. 2842–2852, Sep. 2024, doi: 10.30574/wjarr.2024.23.3.2983.

[15] S. R. Putri *et al.*, "Analisis Sentimen Publik terhadap Nadiem Makarim sebagai Mendikbudrisktek menggunakan Support Vector Machine (SVM)." Available: http://sistemasi.ftik.unisi.ac.id

[16] Z. Gao, Z. Li, J. Luo, and X. Li, "Short Text Aspect-Based Sentiment Analysis Based on CNN + BiGRU," *Applied Sciences (Switzerland)*, vol. 12, no. 5, Mar. 2022, doi: 10.3390/app12052707.

[17] A. S. Talaat, "Sentiment analysis classification system using hybrid BERT models," *J Big Data*, vol. 10, no. 1, Dec. 2023, doi: 10.1186/s40537-023-00781-w.

[18] L. Ashbaugh and Y. Zhang, "A Comparative Study of Sentiment Analysis on Customer Reviews Using Machine Learning and Deep Learning," *Computers*, vol. 13, no. 12, Dec. 2024, doi: 10.3390/computers13120340.

[19] C. P. Chai, "Comparison of text preprocessing methods," *Nat Lang Eng*, vol. 29, no. 3, pp. 509–553, May 2023, doi: 10.1017/S1351324922000213.

[20] M. A. Palomino and F. Aider, "Evaluating the Effectiveness of Text Pre-Processing in Sentiment Analysis," *Applied Sciences (Switzerland)*, vol. 12, no. 17, Sep. 2022, doi: 10.3390/app12178765.

[21] M. Kumar, L. Khan, and H. T. Chang, "Evolving techniques in sentiment analysis: a comprehensive review," 2025, *PeerJ Inc*. doi: 10.7717/PEERJ-CS.2592.

[22] V. Gupta and P. Rattan, "Advancing Sentiment Analysis in Restaurant Reviews through Unsupervised Machine Learning Algorithms," *International Journal of Intelligent Engineering and Systems*, vol. 17, no. 4, pp. 1108–1121, 2024, doi: 10.22266/IJIES2024.0831.82.

[23] N. Jiang, C. Luo, V. Lakshman, Y. Dattatreya, and Y. Xue, "Massive Text Normalization via an Efficient Randomized Algorithm," in *WWW 2022 - Proceedings of the ACM Web Conference 2022*, Association for Computing Machinery, Inc, Apr. 2022, pp. 2946–2956. doi: 10.1145/3485447.3512015.

[24] K. S. Eljil, F. Nait-Abdesselam, E. Hamouda, and M. Hamdi, "Enhancing Sentiment Analysis on Social Media with Novel Preprocessing Techniques," *Journal of Advances in Information Technology*, vol. 14, no. 6, pp. 1206–1213, 2023, doi: 10.12720/jait.14.6.1206-1213.

[25] H. Dwiharyono and S. Suyanto, "Stemming for Better Indonesian Text-to-Phoneme," *Ampersand*, vol. 9, Jan. 2022, doi: 10.1016/j.amper.2022.100083.

[26] L. Zhang, "Features extraction based on Naive Bayes algorithm and TF-IDF for news classification," *PLoS One*, vol. 20, no. 7 July, Jul. 2025, doi: 10.1371/journal.pone.0327347.

[27] K. Yusupov, M. R. Islam, I. Muminov, M. Sahlabadi, and K. Yim, "Comparative Analysis of Machine Learning and Deep Learning Models for Email Spam Classification Using TF-IDF and Word Embedding Techniques," in *Advances on Broad-Band Wireless Computing, Communication and Applications*, L. Barolli, Ed., Cham: Springer Nature Switzerland, 2025, pp. 114–122.

[28] P. Sankar, N. Palanichamy, and K. W. Ng, "Sentiment Analysis on Twitter Data for Depression Detection," *Journal of Logistics, Informatics and Service Science*, vol. 11, no. 3, pp. 21–36, 2024, doi: 10.33168/JLISS.2024.0302.

[29] A. Rajesh and T. Hiwarkar, "Sentiment analysis from textual data using multiple channels deep learning models," *Journal of Electrical Systems and Information Technology*, vol. 10, no. 1, Nov. 2023, doi: 10.1186/s43067-023-00125-x.

[30] M. Hayaeian Shirvan, M. H. Moattar, and M. Hosseinzadeh, "Deep generative approaches for oversampling in imbalanced data classification problems: A comprehensive review and comparative analysis," Feb. 01, 2025, *Elsevier Ltd*. doi: 10.1016/j.asoc.2024.112677.

[31] C. Suhaeni and H. S. Yong, "Mitigating Class Imbalance in Sentiment Analysis through GPT-3-Generated Synthetic Sentences," *Applied Sciences (Switzerland)*, vol. 13, no. 17, Sep. 2023, doi: 10.3390/app13179766.

[32] N. Alturayeif and J. Hassine, "Data leakage detection in machine learning code: transfer learning, active learning, or low-shot prompting?," *PeerJ Comput Sci*, vol. 11, 2025, doi: 10.7717/peerj-cs.2730.

[33] P. Pramanik, S. Samanta, R. K. Mondal, J. Patra, U. Adhikari, and S. Gupta, "Enhancing Text Intelligence with Soft Voting and TF-IDF Logistic Learners," in *Proceedings of Data Analytics and Management*, A. Swaroop, B. Virdee, S. D. Correia, and Z. Polkowski, Eds., Cham: Springer Nature Switzerland, 2026, pp. 301–313.

[34] A. Alqurafi and T. Alsanoosy, "Measuring Customers' Satisfaction Using Sentiment Analysis: Model and Tool," *Journal of Computer Science*, vol. 20, no. 4, pp. 419–430, 2024, doi: 10.3844/jcssp.2024.419.430.

[35] S. U. Hassan, J. Ahamed, and K. Ahmad, "Analytics of machine learning-based algorithms for text classification," *Sustainable Operations and Computers*, vol. 3, pp. 238–248, Jan. 2022, doi: 10.1016/j.susoc.2022.03.001.

[36] S. E. Sorour, A. Alojail, A. El-Shora, A. E. Amin, and A. A. Abohany, "A Hybrid Deep Learning Approach for Enhanced Sentiment Classification and Consistency Analysis in Customer Reviews," *Mathematics*, vol. 12, no. 23, Dec. 2024, doi: 10.3390/math12233856.

[37] Y. Mao, Y. Zhang, L. Jiao, and H. Zhang, "Document-Level Sentiment Analysis Using Attention-Based Bi-Directional Long Short-Term Memory Network and Two-Dimensional Convolutional Neural Network," *Electronics (Switzerland)*, vol. 11, no. 12, Jun. 2022, doi: 10.3390/electronics11121906.

[38] D. Pandya and A. Thakkar, "Sentiment Analysis of Self Driving Car Dataset: A comparative study of Deep Learning approaches," in *Procedia Computer Science*, Elsevier B.V., 2024, pp. 12–21. doi: 10.1016/j.procs.2024.04.002.

[39] E. Altuncu, V. N. L. Franqueira, and S. Li, "Deepfake: definitions, performance metrics and standards, datasets, and a meta-review," 2024, *Frontiers Media SA*. doi: 10.3389/fdata.2024.1400024.

[40] M. C. Hinojosa Lee, J. Braet, and J. Springael, "Performance Metrics for Multilabel Emotion Classification: Comparing Micro, Macro, and Weighted F1-Scores," *Applied Sciences (Switzerland)*, vol. 14, no. 21, Nov. 2024, doi: 10.3390/app14219863.