Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 3, Maret 2026
e-ISSN 2685-998X
DOI 10.30865/json.v7i3.9390

Analisis Sentimen Ulasan Pengguna pada Aplikasi Gojek Menggunakan Metode *Deep Learning* Berbasis Algoritma LSTM

Raphael Hasiando Sihotang[1*], Suhirman[2]
[1]Informatika, Fakultas Sains & Teknologi, Universitas Teknologi Yogyakarta, Yogyakarta, Indonesia
[2]Teknologi Informasi, Program Pascasarjana, Universitas Teknologi Yogyakarta, Yogyakarta, Indonesia
Email: [1,*]raphael.hasiando1@gmail.com, [2]suhirman@uty.ac.id
Email Penulis Korespondensi: raphael.hasiando1@gmail.com[*]
Submitted: **29/11/2025**; Accepted: **17/03/2026**; Published: **31/03/2026**

Abstrak−Aplikasi layanan transportasi seperti Gojek sudah menjadi bagian dari kehidupan sehari-hari masyarakat. Seiring bertambahnya jumlah pengguna, makin banyak juga ulasan yang masuk terkait pengalaman mereka menggunakan aplikasi ini. Namun, sering ditemukan ketidaksesuaian antara skor bintang yang diberikan dengan isi ulasan, seperti pemberian skor tinggi yang disertai keluhan atau pengalaman buruk. Hal ini dapat menyulitkan pihak Gojek dalam mengidentifikasi kendala layanan atau sentimen pengguna apabila hanya mengandalkan skor bintang. Salah satu solusi untuk membantu pihak Gojek dalam mengidentifikasi kendala layanan secara otomatis adalah menerapkan analisis sentimen berbasis algoritma *Long Short-Term Memory* (LSTM) terhadap ulasan pengguna di *Google Play Store*. Penelitian ini bertujuan untuk menerapkan algoritma LSTM dalam melakukan klasifikasi sentimen ulasan pengguna ke dalam tiga kategori, yaitu positif, netral, dan negatif. Kontribusi penelitian ini terletak pada pengembangan model klasifikasi tiga kelas berbasis LSTM dengan pelabelan manual berbasis konteks, penerapan *Random Oversampling* untuk menangani kelas minoritas, serta *preprocessing* teks mendalam dan *embedding FastText* untuk meningkatkan performa model. Hasil pengujian menunjukkan model LSTM mencapai akurasi 88.15%, dengan F1 *score* positif 73%, netral 7%, dan negatif 94%. Model *Bidirectional* LSTM sebagai pembanding mencapai akurasi 90.32%, dengan F1 *score* positif 78%, netral 12%, dan negatif 95%. Kelas netral dapat diprediksi oleh kedua model, namun masih sulit diprediksi secara akurat.

Kata Kunci: Analisis Sentimen; LSTM; Deep Learning; Gojek; Ulasan Pengguna

Abstract−Transportation service applications such as Gojek have become a part of people's daily lives. As the number of users increases, more reviews are submitted regarding their experiences with the application. However, inconsistencies are often found between the star ratings given and the content of the reviews, such as high ratings accompanied by complaints or negative experiences. This situation makes it difficult for Gojek to identify service issues or user sentiment if relying only on star ratings. One of the solutions to automatically identify service issues is to apply sentiment analysis using the Long Short-Term Memory (LSTM) algorithm on user reviews from the Google Play Store. The purpose of this research is to implement the LSTM algorithm to classify user review sentiments into three categories, namely positive, neutral, and negative. The contribution of this research is reflected in the development of a three-class sentiment classification model based on LSTM with context-based manual labeling, implementing Random Oversampling to handle minority classes, as well as comprehensive text preprocessing and FastText embedding to improve model performance. The experimental results show that the LSTM model achieves an accuracy of 88.15%, with F1 scores of 73% for positive, 7% for neutral, and 94% for negative. The Bidirectional LSTM model used for comparison achieves an accuracy of 90.32%, with F1 scores of 78% for positive, 12% for neutral, and 95% for negative. Although both models are able to predict the neutral class, it remains difficult to predict accurately.

Keywords: Sentiment Analysis; LSTM; Deep Learning; Gojek; User Reviews

1. PENDAHULUAN

Teknologi telah menjadi bagian yang tidak terpisahkan dari kehidupan sehari-hari, termasuk dalam bidang transportasi. Masyarakat kini lebih memilih layanan yang praktis dan cepat melalui aplikasi transportasi online [1]. Salah satu aplikasi yang paling populer di Indonesia adalah Gojek, yang menyediakan berbagai layanan seperti transportasi, pembayaran digital melalui GoPay, serta pemesanan makanan melalui GoFood [2]. Gojek adalah layanan transportasi online terbesar dan digunakan oleh jutaan orang setiap harinya.

Di sisi lain, penggunaan aplikasi ini juga menghadapi berbagai kendala seperti gangguan sistem, keterlambatan layanan, dan masalah pembayaran. Pengguna sering menyampaikan pengalaman mereka melalui ulasan di *Google Play Store*, namun sering ditemukan ketidaksesuaian antara skor bintang dan isi komentar. Misalnya, ulasan dengan skor tinggi tetap berisi keluhan terhadap layanan. Saat ini, proses identifikasi dan respons terhadap ulasan pengguna masih dilakukan secara manual oleh tim evaluasi layanan. Kondisi tersebut menyulitkan proses identifikasi keluhan karena sentimen negatif tidak hanya muncul pada ulasan dengan skor rendah. Oleh karena itu, diperlukan pendekatan otomatis berupa analisis sentimen untuk mengklasifikasikan ulasan pengguna Gojek di *Google Play Store* sehingga proses identifikasi keluhan dapat dilakukan secara lebih sistematis.

Analisis sentimen merupakan pendekatan dalam *text mining* yang memanfaatkan *Natural Language Processing* (NLP) untuk mengidentifikasi dan mengklasifikasikan opini atau sentimen dalam teks ke dalam kategori seperti positif, netral, atau negatif. Salah satu metode yang dapat digunakan adalah algoritma *Long Short-Term Memory* (LSTM) berbasis *deep learning*, yang mampu memproses data teks dalam jumlah besar dan meningkatkan akurasi klasifikasi [3]. Penelitian ini melakukan analisis sentimen komentar pengguna

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 3, Maret 2026
e-ISSN 2685-998X
DOI 10.30865/json.v7i3.9390

menggunakan algoritma LSTM yang merupakan bagian dari metode *deep learning*, yaitu pendekatan yang mempelajari pola dalam data melalui jaringan berlapis untuk menghasilkan representasi yang lebih kompleks [4]. Algoritma LSTM cocok digunakan untuk menganalisis data berbentuk teks karena mampu memahami urutan kata dalam kalimat yang kompleks secara lebih efektif.

Beberapa penelitian sebelumnya telah dilakukan dengan pendekatan serupa. Pipin dan Kurniawan menerapkan algoritma LSTM dengan pembobotan TF-IDF untuk analisis sentimen berbasis emosi terhadap kebijakan Merdeka Belajar Kampus Merdeka (MBKM). Model dilatih menggunakan 658 tweet berbahasa Indonesia dan memperoleh akurasi pelatihan sebesar 80.42% [5]. Penelitian oleh Alghifari, Edi, dan Firmansyah melakukan analisis sentimen terhadap ulasan aplikasi Grab di *Google Play Store* menggunakan algoritma *Bidirectional Long Short-Term Memory* (BiLSTM) dengan pelabelan otomatis berdasarkan rating bintang. Model BiLSTM dibandingkan dengan LSTM serta beberapa metode *machine learning* seperti *Multinomial Naïve Bayes*, *Logistic Regression*, dan *Support Vector Classifier*. Hasil pengujian menunjukkan bahwa BiLSTM memperoleh akurasi tertinggi sebesar 91%, namun penelitian tersebut menggunakan dataset yang relatif terbatas sehingga berpotensi menimbulkan *overfitting* [6]. Penelitian oleh Isnain dkk membandingkan kinerja algoritma LSTM dan *Naïve Bayes* dalam analisis sentimen terhadap kebijakan *New Normal*. Penelitian ini menggunakan 1590 data hasil penyeimbangan sentimen positif dan negatif dari total 1823 data. Hasil pengujian menunjukkan bahwa LSTM memperoleh akurasi 83.33%, lebih tinggi dibandingkan *Naïve Bayes* sebesar 82%, sehingga menunjukkan keunggulan LSTM dalam analisis sentimen [7]. Penelitian oleh Musfiroh, Tholib, dan Arifin melakukan analisis sentimen positif, netral, dan negatif terhadap opini aplikasi Shopee di *Google Play Store* menggunakan metode LSTM, TF-IDF, dan *labeling* otomatis berdasarkan rating. Hasil pengujian pada model LSTM menunjukkan akurasi uji sebesar 83% yang di mana model tersebut cukup baik dalam memprediksi kelas positif dan negatif, namun kurang efektif pada kelas netral, dikarenakan data pada kelas tersebut kurang dan tidak seimbang, sehingga menghasilkan *overfitting* [8]. Penelitian Budaya, Pratiwi, dan Agustino menggunakan LSTM untuk klasifikasi sentimen positif, netral, dan negatif pada 694 opini pasien di Puskesmas, dengan akurasi 76.26%. Model efektif untuk sentimen positif dan negatif, namun performa netral rendah akibat ketidakseimbangan data [9].

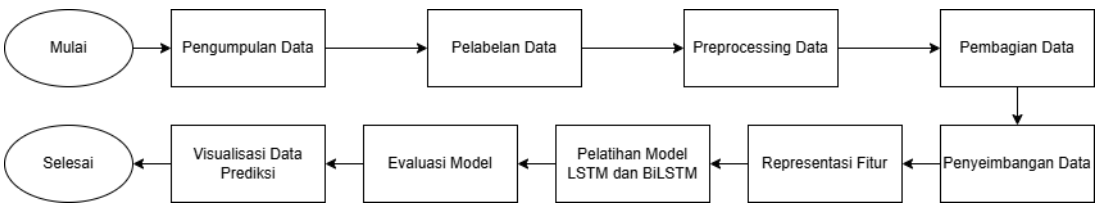
Meskipun berbagai penelitian telah menunjukkan efektivitas LSTM dalam analisis sentimen, beberapa studi masih menggunakan pelabelan otomatis berdasarkan skor bintang, sehingga berpotensi mengabaikan konteks isi ulasan yang tidak selalu selaras dengan skor. Beberapa penelitian juga memanfaatkan TF-IDF sebagai representasi fitur berbasis frekuensi, yang belum secara eksplisit menangkap hubungan makna dan konteks antar kata dalam teks [5], [10]. Adapun juga penelitian lain menggunakan dataset yang relatif terbatas dan distribusi kelas yang tidak seimbang, sehingga kelas minoritas sulit diprediksi secara akurat. Selain itu, sebagian penelitian belum mengoptimalkan tahapan *text preprocessing* secara menyeluruh dan sistematis.

Berdasarkan gap tersebut, penelitian ini mengusulkan pengembangan model klasifikasi sentimen tiga kelas yang mempertimbangkan konteks makna teks serta optimalisasi tahapan pengolahan data, menerapkan teknik penyeimbangan kelas, serta ekstraksi fitur untuk meningkatkan kualitas klasifikasi sentimen. Penelitian ini berkontribusi melalui penerapan pelabelan manual berbasis konteks, optimalisasi tahapan *text preprocessing*, penerapan *Random Oversampling* untuk menangani kelas yang minoritas, serta penggunaan *FastText* sebagai representasi fitur dalam pengembangan model untuk meningkatkan performa dibandingkan pendekatan yang digunakan pada penelitian sebelumnya. Tujuan penelitian ini adalah mengembangkan dan mengevaluasi model klasifikasi sentimen tiga kelas pada ulasan pengguna aplikasi Gojek di *Google Play Store* menggunakan algoritma LSTM. Harapannya, hasil penelitian ini dapat membantu pihak pengembang aplikasi dalam memahami persepsi pengguna secara lebih objektif berdasarkan analisis teks ulasan, serta menjadi bahan evaluasi dalam upaya peningkatan kualitas layanan.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Pada Gambar 1 berikut adalah gambaran alur pengembangan sistem analisis sentimen ulasan aplikasi Gojek.



Gambar 1. Tahapan Penelitian

Berdasarkan Gambar 1, penelitian ini diawali dengan proses pengumpulan data berupa ulasan pengguna aplikasi Gojek. Setiap ulasan kemudian diberi label sentimen yang terdiri dari positif, netral, dan negatif. Data yang telah

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 3, Maret 2026
e-ISSN 2685-998X
DOI 10.30865/json.v7i3.9390

dilabeli selanjutnya melalui tahap *preprocessing* untuk membersihkan dan menormalisasi teks. Setelah *preprocessing*, dataset dibagi menjadi data latih, data validasi, dan data uji. Untuk mengatasi ketidakseimbangan kelas, teknik *Random Oversampling* diterapkan pada data latih. Data latih kemudian direpresentasikan ke dalam bentuk numerik dan digunakan untuk melatih dua model *deep learning*, yaitu *Long Short-Term Memory* (LSTM) dan *Bidirectional Long Short-Term Memory* (BiLSTM). Selanjutnya, kedua model dievaluasi menggunakan data uji untuk mengukur performa klasifikasi sentimen. Hasil evaluasi kemudian dibandingkan untuk mengetahui model dengan kinerja yang lebih baik. Tahap terakhir adalah visualisasi hasil prediksi menggunakan *word cloud* untuk menampilkan kata-kata dominan pada setiap kategori sentimen.

2.2 Data Penelitian

Pada Gambar 2 berikut adalah data sampel yang telah dikumpulkan dari ulasan pengguna Gojek di *Google Play Store*.

	A	B
1	snippet	sentiment
2	setiap mengirim barang sudah dicantumkan semua nomor penerima dan pengir	Negatif
3	Untuk layanan goride, bila pengantarannya seperti di titik bandara, terminal, sta	Negatif
4	Sangat puas dengan layanan go jek ngampang dan praktis dan aman sampai tuj	Negatif
5	gojek ini perlu diperbaiki terkait fitur lokasi otomatis. selalu saja ga tepat sehin	Negatif
6	PESAN TERBUKA UNTUK PARA PEJUANG!! Terimakasih ya, dengan bantuan dar	Positif
7	Parah sekali. Untuk yg kesekian kalinya paylater saya tidak bisa digunakan, pad	Negatif
8	Saya baru nyoba gojek, biasanya gr dan max. saya mesan gocar harga 18 ribu ti	Negatif
9	makin ga karuan sekarang pakai gojeg,,ada biaya aplikasi,,biaya ini,,sudah gitu d	Negatif
10	Mikirnya order food pake yg express biar diantarnya duluan atau driver tidak da	Negatif
11	Pengalaman yang kurang baik.orderan masuk ke driver. tapi drivernya gak mau	Negatif
12	Lokasi aplikasi tidak jelas. Saya sudah biasa mesan go food si satu lokasi tanpa	Negatif

Gambar 2. Data Sampel Ulasan Pengguna Aplikasi Gojek

Gambar 2 menampilkan contoh data sampel yang digunakan dalam penelitian ini. Data yang digunakan dalam penelitian ini merupakan ulasan pengguna aplikasi Gojek yang diperoleh dari *Google Play Store*. Dataset terdiri dari 8159 ulasan dengan rentang waktu 2018 hingga 2025. Setiap data memiliki dua atribut utama, yaitu *snippet* yang berisi teks ulasan pengguna dan *sentiment* yang merupakan label hasil pelabelan manual ke dalam tiga kelas, yaitu positif, netral, dan negatif.

2.2.1 Pengumpulan Data

Pengumpulan data dilakukan menggunakan teknik *web scraping* pada *Google Play Store* dengan bantuan *SerpApi* melalui bahasa pemrograman *Python* untuk mengumpulkan data ulasan secara otomatis dari situs tersebut [11]. Data yang diperoleh kemudian disimpan dalam format CSV dan digunakan sebagai dataset dalam proses pelatihan dan pengujian model klasifikasi sentimen. Tahap ini bertujuan untuk menyediakan dataset ulasan pengguna yang digunakan dalam pengembangan dan evaluasi model klasifikasi sentimen tiga kelas menggunakan algoritma LSTM.

2.2.2 Pelabelan Data

Pelabelan data dilakukan secara manual, yaitu dengan memberikan label sentimen pada setiap data ulasan yang terdapat dalam dataset. Dataset ini memiliki tiga nilai kategorikal, yaitu positif, netral, dan negatif. Proses pelabelan dilakukan berdasarkan makna yang terkandung dalam teks ulasan. Tabel 1 berikut adalah 3 sampel data dengan sentimen positif, negatif, dan netral yang diambil dari dataset.

Tabel 1. Data Sampel Ulasan Pengguna Aplikasi Gojek

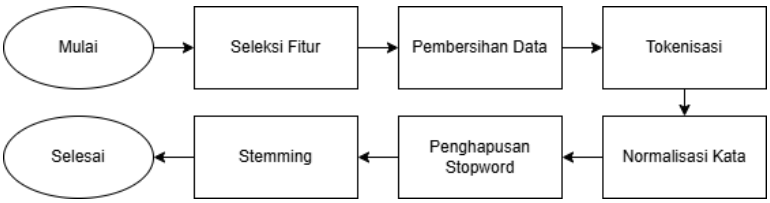
Ulasan	Sentimen
aplikasi yang sangat membantu ketika mau berpergian. dan aplikasi yang sangat mudah di gunakan tanpa ribet atau pun kendala lain nya. mantul lah pokok nya....	Positif
Saran Untuk GoTransit. Mungkin Bisa Diperluas Untuk MRT, LRT, Transjakarta Dan Transportasi Umum Lainnya Berbasis Scan QR ðŸ™•	Netral
Sengaja saya kasih bintang 5 , biar cepet ditanggapi.. Kronologi saya transfer dari gopay ke rekening bank, tapi status nya diproses mulu.. Mana lagi butuh banget duitnya, udah hubungi CS nya malah ga ada respon sama sekali, estimasi transfer gopay berapa hari si? Lama banget, no respon.. Nyesel !!	Negatif

Berdasarkan Tabel 1, ulasan yang berisi keluhan atau permasalahan terkait layanan Gojek, seperti antarmuka yang lambat atau kesalahan pada fitur peta, dikategorikan sebagai sentimen negatif. Ulasan yang menyampaikan pengalaman atau tanggapan positif terhadap layanan dikategorikan sebagai sentimen positif, sedangkan ulasan yang bersifat saran atau tidak mengandung keluhan maupun tanggapan positif dikategorikan sebagai sentimen netral [12]. Tahap ini bertujuan meningkatkan validitas data sentimen dengan mempertimbangkan konteks teks sehingga mendukung pembangunan model klasifikasi sentimen tiga kelas.

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 3, Maret 2026
e-ISSN 2685-998X
DOI 10.30865/json.v7i3.9390

2.3 *Preprocessing* Data

Preprocessing merupakan tahap awal dalam pengolahan data yang bertujuan untuk membersihkan dan menyiapkan data agar layak digunakan dalam proses pelatihan dan pengujian model LSTM dan BiLSTM [13]. Pada penelitian ini, *preprocessing* dilakukan pada data teks ulasan, serta pengurangan fitur untuk memastikan kualitas dan kesiapan data sebelum digunakan dalam proses pemodelan. Proses ini membantu mengurangi fitur yang tidak relevan serta mendukung peningkatan performa model dalam klasifikasi sentimen tiga kelas [14]. Gambar 3 berikut menunjukkan *flowchart* tahapan proses *preprocessing* data yang dilakukan sebelum data digunakan dalam pelatihan model.



Gambar 3. Tahapan *Preprocessing* Data

Pada Gambar 3, ditampilkan beberapa tahapan *preprocessing* data yang dilakukan sebelum data digunakan untuk melatih model LSTM. *Preprocessing* data dilakukan menggunakan bahasa pemrograman *Python* dengan bantuan beberapa *library* seperti *Pandas* untuk pengelolaan dataset, *regression expression* untuk pembersihan teks berbasis ekspresi reguler, Sastrawi untuk proses *stemming* Bahasa Indonesia. Pemilihan alat-alat ini dilakukan karena mendukung pengolahan data teks dalam Bahasa Indonesia secara efisien dan terintegrasi langsung dengan model *deep learning* yang digunakan.

a. Seleksi Fitur
   Tahap pertama adalah seleksi fitur, yaitu mengurangi kolom yang tidak relevan sehingga hanya tersisa dua kolom utama, yaitu *snippet* (ulasan teks) dan *sentiment* (label sentimen), yang berpengaruh terhadap akurasi model [15].
b. Pembersihan Teks
   Setelah seleksi fitur, seluruh data pada kolom ulasan teks dibersihkan dengan menghapus angka, simbol, *emoji*, spasi berlebih, tanda baca, dan karakter non-alfabet, serta diubah menjadi huruf kecil untuk menjaga konsistensi analisis.
c. Tokenisasi
   Setelah teks dibersihkan, setiap ulasan teks dipecah menjadi unit kata atau token. Proses ini dilakukan menggunakan fungsi *split* pada *Python*, yang memisahkan teks berdasarkan spasi atau *whitespace*.
d. Normalisasi Kata
   Normalisasi kata merupakan proses mengoreksi kata tidak baku, singkatan, atau variasi penulisan agar sesuai dengan bentuk yang benar menurut KBBI. Tahapan ini bertujuan menyamakan bentuk kata dalam teks yang memiliki makna serupa agar model dapat mengenali konteks dengan lebih baik [16]. Misalnya, kata seperti “psn”, “mesan”, atau “mesen” diubah menjadi “pesan”, serta “sya” menjadi “saya”. Selain itu, kata yang memiliki arti sama seperti “ponsel” dan “*handphone*” dikonversi menjadi satu bentuk kata yang konsisten. Dalam beberapa kasus, frasa seperti “tidak ribet” atau “tidak sulit” disederhanakan menjadi “mudah” untuk menjaga makna kalimat setelah penghapusan kata negatif seperti “tidak”.
e. Penghapusan *Stopword*
   Teknik ini digunakan untuk menghapus kata-kata umum yang tidak memiliki makna signifikan dalam analisis, seperti “dan”, “yang”, “di”, “ke”, dan sebagainya. Proses penghapusan *stopword* dilakukan menggunakan daftar kata umum bahasa Indonesia agar hanya kata bermakna penting yang digunakan sebagai masukan model [5].
f. *Stemming*
   Stemming merupakan proses mengubah kata menjadi bentuk dasarnya agar memiliki makna yang konsisten dalam analisis teks. Pada penelitian ini digunakan *library* Sastrawi untuk melakukan *stemming* pada bahasa Indonesia, misalnya kata “memesan”, “terpesan”, dan “dipesan” menjadi “pesan” [17]. Namun, Sastrawi memiliki kelemahan dalam memproses kata non-formal yang sering muncul pada media sosial, seperti “pesen” atau “mesen”, sehingga diperlukan proses normalisasi untuk menyamakan kata tidak baku menjadi bentuk formal [17], [18].

2.4 Pembagian Data

Setelah *preprocessing*, dataset dibagi menjadi data latih, data validasi, dan data uji. Data latih digunakan untuk melatih model, data validasi untuk memantau kinerja selama pelatihan dan mencegah *overfitting*, sedangkan data uji untuk mengevaluasi performa akhir model terhadap data yang tidak dilibatkan sebelumnya sehingga diperoleh nilai akurasi dan metrik lainnya [13]. Pembagian dilakukan dengan 70% data latih dan 30% data uji, di mana 30%

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 3, Maret 2026
e-ISSN 2685-998X
DOI 10.30865/json.v7i3.9390

dari data latih dijadikan data validasi. Pembagian ini bertujuan agar evaluasi model klasifikasi sentimen tiga kelas dilakukan secara terukur dan ilmiah.

2.5 Penyeimbangan Data

Tahap ini bertujuan untuk mengatasi ketidakseimbangan jumlah data antar kelas yang dapat menyebabkan model cenderung bias terhadap kelas mayoritas. Untuk mengatasi masalah tersebut, digunakan teknik *Random Oversampling* (ROS), yaitu metode yang memperbanyak sampel pada kelas minoritas secara acak hingga jumlahnya seimbang dengan kelas mayoritas. Dengan penyeimbangan data ini, model diharapkan dapat mempelajari setiap kelas secara lebih seimbang dan meningkatkan performa klasifikasi sentimen [19].

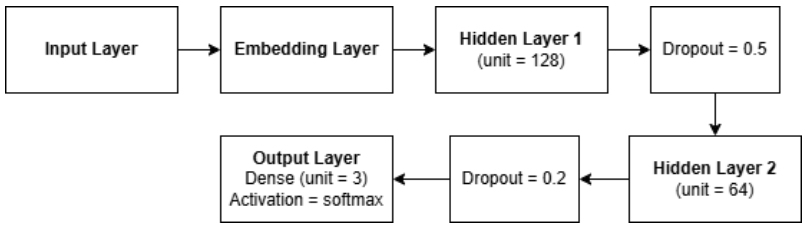
2.6 Representasi Fitur

Pada tahap ini, teks yang telah melalui proses *preprocessing* diubah menjadi representasi numerik agar dapat diproses oleh model LSTM dan BiLSTM. Proses ini dilakukan dengan mengonversi setiap token menjadi urutan bilangan bulat menggunakan *Tokenizer* dari *library bernama Keras* (*text to sequence*). Karena model memerlukan panjang input yang seragam, setiap urutan token kemudian disamakan panjangnya menggunakan teknik *padding*. Selanjutnya, kata direpresentasikan dalam bentuk vektor melalui *embedding matrix* yang digunakan sebagai input pada model. Karena kolom ulasan teks dalam dataset berbahasa Indonesia, penelitian ini menggunakan *pre-trained word embedding FastText* (*Indonesian Word Vector*) agar model dapat memahami hubungan semantik antar kata dalam bahasa Indonesia [10].

2.7 Pelatihan Model LSTM dan BiLSTM

Tahap ini bertujuan untuk melatih model *deep learning* agar mampu mempelajari pola dan hubungan dalam representasi numerik teks sehingga dapat mengklasifikasikan sentimen ke dalam tiga kelas. Pada penelitian ini digunakan dua model, yaitu LSTM sebagai model utama dan BiLSTM sebagai model pembanding. Kedua model dikembangkan menggunakan bahasa pemrograman *Python* dengan *library* bernama *TensorFlow* dan *Keras*.

a. Model LSTM
LSTM merupakan pengembangan dari *Recurent Neural Network* (RNN) yang dirancang untuk mengatasi masalah *vanishing gradient*. LSTM memiliki struktur khusus berupa unit memori (*cell state*) dan tiga gerbang utama, yaitu *input gate*, *forget gate*, dan *output gate*. Ketiga gerbang ini mengatur aliran informasi yang masuk, disimpan, dan dikeluarkan dari unit memori, sehingga jaringan dapat menyimpan informasi dalam jangka waktu yang panjang [5]. Arsitektur Model LSTM pada penelitian ini dapat dilihat pada Gambar 4 berikut.



Gambar 4. Arsitektur Model

Model LSTM pada penelitian ini terdiri dari 4 bagian utama yaitu *input layer* yang diawali dengan *embedding layer*, dua *hidden layer* bertipe LSTM, dan *output layer*. Proses dimulai dari *Input Layer* yang menerima data yang siap diolah. Selanjutnya, *Embedding Layer* mengubah setiap kata menjadi vektor berdimensi tetap agar model dapat memahami makna antar kata. Model memiliki dua *Hidden Layer* bertipe LSTM, masing-masing dengan 128 dan 64 unit, yang berfungsi menangkap konteks urutan kata. Untuk mencegah *overfitting*, digunakan *Dropout* sebesar 0.5 dan 0.2. Terakhir, *Output Layer* berupa *Dense layer* dengan 3 unit *neuron* dan fungsi aktivasi *Softmax*, yang menghasilkan nilai probabilitas dari sentimen positif, netral, dan negatif.

b. Model BiLSTM
BiLSTM adalah pengembangan LSTM yang memproses data sekuensial dalam dua arah (*forward* dan *backward*), sehingga dapat menangkap konteks sebelum dan sesudah kata dalam urutan teks, menghasilkan representasi fitur yang lebih lengkap [20]. Arsitektur BiLSTM pada penelitian ini sama dengan LSTM yang ditampilkan pada Gambar 4, hanya *hidden layer* diganti dengan BiLSTM. Model ini digunakan sebagai pembanding karena kemampuannya memahami konteks dua arah dan potensi performa lebih baik dalam analisis sentimen [6].

c. *Hyperparameter* Model
Hyperparameter yang digunakan pada penelitian ini dapat dilihat pada Tabel 2 berikut.

Tabel 2. *Hyperparameter* Model

Parameter	Nilai Parameter	Keterangan

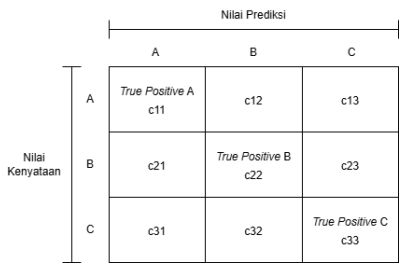
Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 3, Maret 2026
e-ISSN 2685-998X
DOI 10.30865/json.v7i3.9390

Learning Rate	0.0001	Mengatur kecepatan pembaruan bobot selama pelatihan. Nilai kecil dipilih untuk menjaga stabilitas pembelajaran dan menghindari pembaruan bobot yang terlalu besar sehingga proses konvergensi lebih stabil.
Epoch	50	Jumlah perulangan pelatihan terhadap seluruh data latih. Nilai tersebut digunakan untuk memberikan waktu pembelajaran yang cukup bagi model tanpa meningkatkan risiko *overfitting* [13].
Batch Size	64	Jumlah sampel yang diproses setiap iterasi atau sebelum model memperbarui bobotnya. Nilai tersebut dipilih karena merupakan konfigurasi umum dalam pelatihan *deep learning* yang mampu menjaga keseimbangan antara efisiensi komputasi dan stabilitas gradien.
Optimizer	*Adam*	Algoritma optimasi untuk mengoptimalkan pembaruan bobot jaringan. Nilai tersebut dipilih karena mampu mempercepat konvergensi dan efektif dalam menangani *sparse gradient* pada model berbasis LSTM dan BiLSTM.
Loss Function	*Categorical Crossentropy*	Metode yang ditugaskan untuk menghitung selisih antara output prediksi dan label sebenarnya pada masalah klasifikasi multi-kelas. Nilai *loss* menunjukkan seberapa besar kesalahan yang dihasilkan model dalam melakukan prediksi [13]. Fungsi tersebut dipilih karena sesuai dengan skema klasifikasi tiga kelas.
Callbacks	ReduceLROnPlateau(monitor='val_loss', factor=0.3, patience=4, min_lr=0.000001)	Karena nilai parameter *patience* adalah 4, *learning rate* akan diturunkan ketika nilai *validation loss* tidak menunjukkan perbaikan selama 4 *epoch* berturut-turut, sehingga membantu model mencapai titik optimal dalam proses pelatihan [21].

Hyperparameter merupakan parameter yang ditentukan sebelum proses pelatihan dimulai dan berperan dalam mengatur proses pembelajaran model. Pemilihan *hyperparameter* didasarkan pada praktik umum dalam pelatihan model LSTM dan BiLSTM, serta referensi dari penelitian sebelumnya untuk memperoleh konfigurasi yang mampu menghasilkan performa optimal tanpa meningkatkan risiko *overfitting*. Perubahan nilai *hyperparameter* dapat memberikan pengaruh yang signifikan terhadap kinerja model [13].

2.8 Evaluasi Model

Metode untuk mengukur kinerja suatu model pada penelitian ini adalah *Confusion Matrix* (Matriks kebingungan), yang menghasilkan *binary classification* atau hasil prediksi model. Matriks yang dievaluasi memiliki 4 nilai, meliputi *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN) [15]. Dalam penelitian ini akan melakukan klasifikasi dengan jumlah 3 kelas, yaitu positif, netral, dan negatif, sehingga matrik yang akan dievaluasi menggunakan matrik tiga baris dan tiga kolom. Gambaran *Confusion Matrix* pada penelitian ini dapat dilihat pada Gambar 5 berikut.



Gambar 5. *Confusion Matrix* untuk tiga kelas

Confusion Matrix pada Gambar 5 merepresentasikan evaluasi model untuk masing-masing kelas [22]. Nilai yang ditampil pada Gambar 5 hanya nilai *True Positive* untuk masing-masing kelas. Nilai *True Negative*, *False Positive*, dan *False Negative* harus dihitung melalui kombinasi beberapa sel lain. Keterangan dari keempat nilai tersebut dapat dilihat pada Tabel 3 berikut.

Tabel 3. Keterangan *Confusion Matrix*

Nama Nilai	Keterangan
TP (*True Positive*)	Model berhasil memprediksi positif, karena kenyataannya positif [15]. Misalnya, TP pada kelas A seperti pada Gambar 5 menunjukkan jumlah data kelas A yang diprediksi benar sebagai kelas A [22].

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 3, Maret 2026
e-ISSN 2685-998X
DOI 10.30865/json.v7i3.9390

TN (*True Negative*)	Model berhasil memprediksi negatif, karena kenyataannya negatif [15]. Pada Gambar 5, TN kelas A adalah penjumlahan sel c22, c23, c32, dan c33. TN kelas B adalah penjumlahan sel c11, c13, c31, dan c33. TN kelas C adalah penjumlahan sel c11, c12, c21, dan c22 [22].
FP (*False Positive*)	Model memprediksi negatif, tetapi salah karena kenyataannya positif [15]. Pada Gambar 5, FP kelas A adalah penjumlahan sel c21 dan c31. TN kelas B adalah penjumlahan sel c12 dan c32. TN kelas C adalah penjumlahan sel c13 dan c23 [22].
FN (*False Negative*)	Model memprediksi positif, tetapi salah karena kenyataannya negatif [15]. Pada Gambar 5, FN kelas A adalah penjumlahan sel c12 dan c13. TN kelas B adalah penjumlahan sel c21 dan c23. TN kelas C adalah penjumlahan sel c31 dan c32 [22].

Nilai-nilai pada *confusion matrix* digunakan sebagai dasar perhitungan berbagai metrik evaluasi, seperti akurasi, presisi, *recall*, dan F1 *score*. Perhitungan masing-masing metrik dilakukan dengan menggunakan rumus sebagai berikut.

$$Accuracy \text{ (Akurasi)} = \frac{TP_A+TP_B+TP_C}{c11+c12+c13+c21+c22+c23+c31+c32+c33} \quad (1)$$

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

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

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

Keterangan:

a. Akurasi, Rumus ini digunakan untuk mengukur seberapa tepat model dalam memprediksi seluruh data untuk tiga kelas dengan benar. Rumus ini dapat dihitung dengan cara dapat dilihat pada persamaan (1). Untuk mengukur akurasi dengan tiga kelas dapat dihitung dengan cara mengambil dan menjumlahkan data yang diprediksi benar, yaitu 'TP_A', 'TP_B', 'TP_C' (*True Positive* kelas A, B, dan C). Nilai tersebut kemudian dibagi dengan jumlah seluruh data [22].
b. Presisi, Rumus ini digunakan untuk mengukur tingkat ketepatan model dalam memprediksi kelas positif [15]. Rumus ini dapat dihitung dengan cara dapat dilihat pada persamaan (2).
c. *Recall*, Rumus ini digunakan untuk mengukur sejauh mana model mampu mengenali seluruh data positif dengan benar [15][15]. Rumus ini dapat dihitung dengan cara dapat dilihat pada persamaan (3).
d. F1 *Score*, Rumus ini merupakan perhitungan rata-rata dari kombinasi presisi dan *recall* [22]. Rumus ini dapat dihitung dengan cara dapat dilihat pada persamaan (4).

2.9 Visualisasi Data Prediksi

Setelah proses evaluasi model LSTM dan BiLSTM, hasil prediksi divisualisasikan menggunakan *word cloud* untuk menampilkan kata-kata yang paling sering muncul pada ulasan pengguna. *Word cloud* menampilkan kata dengan ukuran lebih besar berdasarkan frekuensi kemunculannya [23]. Pada penelitian ini ditampilkan tiga *word cloud* yang merepresentasikan kata dominan pada sentimen positif, netral, dan negatif. Visualisasi ini bertujuan memberikan gambaran konteks kata pada setiap kategori sentimen. Proses visualisasi dilakukan menggunakan *library* bernama *WordCloud* dan *Matplotlib* pada *Python*.

Secara keseluruhan, penelitian ini diawali dengan pengumpulan data ulasan pengguna aplikasi Gojek dari *Google Play Store* menggunakan teknik *web scraping* melalui *SerpApi*. Data kemudian diberi label secara manual menjadi tiga kelas sentimen, yaitu positif, netral, dan negatif, serta melalui tahapan *preprocessing* untuk memastikan kualitas dan konsistensi data. Dataset selanjutnya dibagi menjadi 70% data latih dan 30% data uji, di mana 30% dari data latih digunakan sebagai data validasi untuk memantau proses pelatihan dan membantu mencegah *overfitting*. Untuk mengatasi ketidakseimbangan distribusi kelas, dilakukan penyeimbangan data menggunakan metode *Random Oversampling* pada data latih. Data kemudian direpresentasikan dalam bentuk numerik menggunakan *FastText word embedding* sebelum digunakan dalam pelatihan model LSTM dan BiLSTM untuk melakukan klasifikasi sentimen tiga kelas. Kinerja model selanjutnya dievaluasi menggunakan *confusion matrix* dan metrik evaluasi lainnya, serta didukung visualisasi *word cloud* untuk memberikan gambaran kata dominan pada setiap kategori sentimen.

3. HASIL DAN PEMBAHASAN

3.1 Persiapan Data

Setelah melakukan pengumpulan data, dilakukan proses persiapan data yang terdiri dari *labeling* data, *preprocessing* data, pembagian data, dan ekstraksi fitur. Setelah seluruh proses ini selesai, data yang telah dipersiapkan akan digunakan untuk tahap pelatihan dan pengujian model LSTM dan BiLSTM.

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 3, Maret 2026
e-ISSN 2685-998X
DOI 10.30865/json.v7i3.9390

3.1.1 Hasil *Labeling* Data

Setelah melakukan *labeling* pada seluruh data hasil *web scraping*, jumlah data untuk kelas positif, netral, dan negatif masing-masing adalah 1145, 114, dan 6870. Data menunjukkan bahwa sentimen negatif merupakan kelas yang paling dominan dan sebagian besar berisi keluhan pengguna terhadap layanan Gojek di *Google Play Store*. Sebaliknya, kelas netral memiliki jumlah data paling sedikit sehingga dapat menyebabkan model kesulitan memahami konteks ulasan.

3.1.2 Hasil *Preprocessing* Data

Tabel 4 berikut menampilkan tahapan-tahapan *preprocessing* yang diterapkan pada salah satu sampel data dengan sentimen positif dari dataset.

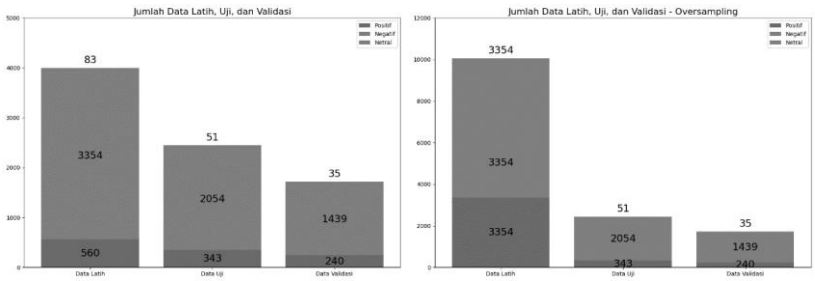
Tabel 4. Tahapan *Preprocessing* Data

Tahapan	Hasil	Keterangan
Data Awal	pelayanannya ramah, gak mahal sama sekali! 😊😊	Teks tersebut merupakan sampel data berlabel positif, yang berisi tanggapan dengan sentimen positif serta mengandung *emoji* yang menunjukkan ekspresi kepuasan.
Pembersihan Teks	pelayanannya ramah gak mahal sama sekali	Tahap ini melakukan pembersihan teks, yaitu menghapus *emoji*, tanda baca, angka, dan spasi berlebih, serta mengubah seluruh teks menjadi huruf kecil.
Tokenisasi	['pelayanannya', 'ramah', 'gak', 'mahal', 'sama', 'sekali']	Setiap teks akan dipecah menjadi unit-unit atau kata, dan disimpan dalam *list*. Hasil tersebut dibutuhkan tahapan-tahapan selanjutnya, supaya setiap kata dapat diproses.
Normalisasi Kata	['pelayanannya', 'ramah', 'murah', 'sama', 'sekali']	Kata tidak baku seperti “gak” dinormalisasi menjadi “tidak mahal”, kemudian disederhanakan menjadi “murah” karena memiliki makna yang sama (sinonim) dan lebih sesuai untuk sentimen positif.
Penghapusan Stopword	['pelayanannya', 'ramah', 'murah']	Tahap ini menghapus kata-kata yang tidak memiliki makna penting, Contoh kata dari hasil teks tersebut adalah “sama” dan “sekali”.
Stemming	['layan', 'ramah', 'murah']	Mengubah kata berimbuhan menjadi kata dasar, misalnya “pelayanannya” disederhanakan menjadi “layan”.

Setiap data yang telah diberikan label sentimen diproses melalui beberapa tahapan *preprocessing*, yang meliputi seleksi fitur, pembersihan teks, normalisasi kata, tokenisasi, penghapusan *stopword*, dan *stemming*. Sebelum dilakukan pembersihan teks, dilakukan seleksi fitur agar hanya tersisa dua kolom utama, yaitu teks ulasan pengguna sebagai fitur dan label sentimen sebagai target.

3.1.3 Hasil Pembagian dan Penyeimbangan Data

Gambar 6 menampilkan dua grafik yang menunjukkan perbandingan jumlah data pada setiap kategori sentimen untuk masing-masing jenis dataset. Grafik tersebut merepresentasikan hasil pembagian data serta hasil penerapan teknik penyeimbangan data menggunakan *Random Oversampling* pada data latih.



Gambar 6. Grafik Hasil Pembagian dan Penyeimbangan Data

Grafik di sebelah kiri pada Gambar 6 menunjukkan distribusi data sebelum proses penyeimbangan. Data latih terdiri dari 83 data netral, 3354 data negatif, dan 560 data positif, dengan total 3997 data. Data uji terdiri dari 51 data netral, 2054 data negatif, dan 343 data positif, dengan total 2448 data. Sementara itu, data validasi terdiri dari 35 data netral, 1439 data negatif, dan 240 data positif, dengan total 1714 data. Grafik di sebelah kanan pada Gambar 6 menunjukkan hasil penerapan teknik *Random Oversampling* pada data latih. Setelah proses penyeimbangan

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 3, Maret 2026
e-ISSN 2685-998X
DOI 10.30865/json.v7i3.9390

dilakukan, ketiga kelas pada data latih memiliki jumlah data yang sama, yaitu 3354 data untuk setiap kelas, sehingga total data latih menjadi 10062 data.

3.1.4 Representasi Fitur

Tabel 5 berikut menunjukkan proses *text to sequence* dan *padding* pada data sampel yang digunakan seperti yang ditampilkan pada Tabel 4.

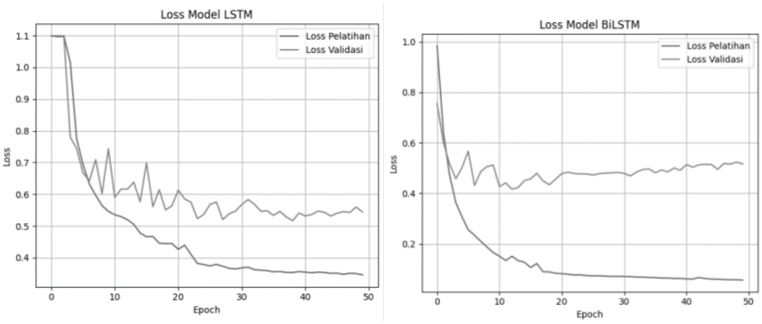
Tabel 5. Tahapan Representasi Fitur

Tahapan	Hasil	Keterangan
Text to Sequence	[285, 182, 144]	Tahap ini mengubah setiap kata menjadi numerik unik. Hasil tersebut merepresentasikan hasil tahapan *preprocessing stemming* pada Tabel 4. angka 285, 182, dan 144 representasi kata “layan”, “ramah” dan “murah”.
Padding	[285, 182, 144, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]	Setiap data memiliki panjang yang berbeda-beda diberikan panjang yang sama dengan memberikan panjang sebanyak 100. Kata-kata yang tidak terisi atau kosong diberikan angka 0.

Setelah melakukan pembagian data, setiap data direpresentasikan dengan cara mengubah setiap token menjadi nilai numerik dengan panjang urutan yang sama. Setelah proses *padding* selesai, dilakukan pembentukan *embedding matrix* dengan menggunakan *word embedding FastText*, yang merepresentasikan setiap kata ke dalam vektor numerik berdimensi 300.

3.2 Hasil Pelatihan Model LSTM dan BiLSTM

Setelah representasi fitur dalam bentuk numerik, model dilatih menggunakan data latih dengan pemantauan kinerja melalui data validasi selama 50 *epoch*. Hasil pelatihan diamati melalui nilai loss pada data latih dan data validasi yang ditampilkan pada Gambar 7.



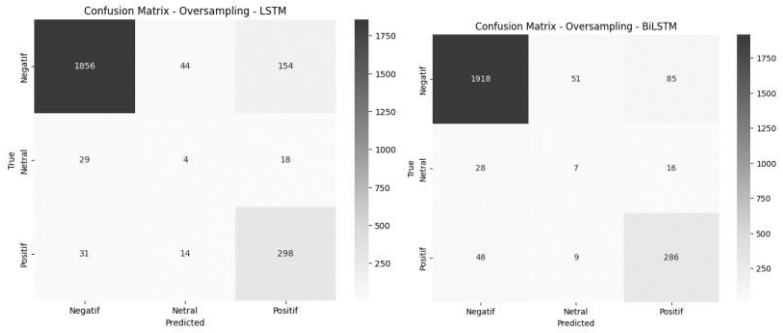
Gambar 7. Grafik Hasil Pelatihan Model LSTM dan BiLSTM

Grafik di sebelah kiri pada Gambar 7 menunjukkan proses pelatihan model LSTM, di mana nilai *loss* pelatihan menurun hingga 0.3377 dan *validation loss* mencapai 0.5443 pada akhir pelatihan. Sementara itu, grafik di sebelah kanan menunjukkan proses pelatihan model BiLSTM dengan nilai *loss* pelatihan menurun hingga 0.058 dan *validation loss* mencapai 0.517. Kedua model menunjukkan indikasi *overfitting*, karena nilai loss pelatihan jauh lebih rendah dibandingkan *validation loss*. Selain itu, akurasi pelatihan dan validasi pada model LSTM masing-masing mencapai 88%. Sementara itu, pada model BiLSTM akurasi pelatihan mencapai 98% dan akurasi validasi mencapai 89%. Berdasarkan hasil tersebut, model BiLSTM menunjukkan performa yang lebih baik dibandingkan model LSTM.

3.3 Hasil Evaluasi Model LSTM dan BiLSTM

Setelah proses pelatihan model LSTM dan BiLSTM selesai, kedua model kemudian diuji menggunakan data uji untuk mengevaluasi performa masing-masing model. Hasil evaluasi ditampilkan dalam bentuk *Confusion Matrix* yang menggambarkan jumlah klasifikasi benar dan salah untuk masing-masing kelas. Visualisasi *confusion matrix* untuk kedua model ditampilkan pada Gambar 8.

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 3, Maret 2026
e-ISSN 2685-998X
DOI 10.30865/json.v7i3.9390



Gambar 8. *Confusion Matrix* Model LSTM dan BiLSTM

Confusion matrix pada sisi kiri Gambar 8 menunjukkan hasil evaluasi model LSTM, sedangkan sisi kanan menunjukkan hasil evaluasi model BiLSTM. Kedua model digunakan untuk mengklasifikasikan sentimen ke dalam tiga kategori, yaitu positif, netral, dan negatif. *Confusion matrix* memberikan gambaran menyeluruh mengenai jumlah prediksi yang benar dan salah dari masing-masing kelas.

a. Untuk kelas negatif, model LSTM mengklasifikasikan 1856 data negatif secara benar. Namun, masih terdapat 198 data yang diklasifikasikan salah, yaitu 44 data netral dan 154 data positif. Sementara itu, model BiLSTM menunjukkan performa yang lebih baik dengan berhasil mengklasifikasikan 1918 data negatif secara benar. Kesalahan klasifikasi yang terjadi berjumlah 136 data, yaitu 51 data diklasifikasikan netral dan 85 data diklasifikasikan positif.
b. Untuk kelas netral, model mengklasifikasikan 4 data netral secara benar. Terdapat total 47 data diklasifikasikan salah, yaitu 29 diklasifikasikan negatif dan 18 data diklasifikasikan positif. Model BiLSTM menunjukkan sedikit peningkatan performa dengan mampu mengklasifikasikan 7 data netral secara benar. Namun masih terdapat 44 data yang salah diklasifikasikan, terdiri dari 28 data diklasifikasikan negatif dan 16 data diklasifikasikan positif.
c. Pada kelas positif, model LSTM berhasil mengklasifikasikan 298 data dengan benar. Namun, masih terdapat 45 data yang diklasifikasikan salah, yaitu 31 diklasifikasikan negatif dan 14 diklasifikasikan netral. Pada model BiLSTM, sebanyak 286 data positif berhasil diklasifikasikan dengan benar. Namun terdapat 57 data yang salah diklasifikasikan, yaitu 48 data diklasifikasikan negatif dan 9 data diklasifikasikan netral.

Setelah mendapatkan jumlah data yang telah evaluasi setiap kelas, Tabel 6 berikut adalah hasil perhitungan metrik evaluasi berupa presisi, *recall*, dan F1 *score* pada model LSTM untuk kelas positif, netral, dan negatif.

Tabel 6. Metrik Presisi, *Recall*, dan F1 *Score* Model LSTM untuk Setiap Kelas

Kelas	Presisi	*Recall*	F1 *Score*
Positif	0.63	0.87	0.73
Netral	0.06	0.08	0.07
Negatif	0.97	0.90	0.94

Berdasarkan nilai F1 *score* pada Tabel 6, model menunjukkan performa yang sangat baik pada kelas negatif dan baik pada kelas positif. Sebaliknya, performa pada kelas netral masih rendah, yang menunjukkan bahwa model masih mengalami kesulitan dalam mengidentifikasi sentimen netral secara akurat. Secara keseluruhan, model LSTM yang dievaluasi memperoleh akurasi pengujian sebesar 88.15%. Setelah memperoleh nilai metrik evaluasi dan akurasi pada model LSTM, evaluasi juga dilakukan pada model BiLSTM. Hasil perhitungan metrik presisi, *recall*, dan F1 *Score* untuk model BiLSTM ditampilkan pada Tabel 7.

Tabel 7. Metrik Presisi, *Recall*, dan F1 *Score* Model BiLSTM untuk Setiap Kelas

Kelas	Presisi	*Recall*	F1 *Score*
Positif	0.74	0.83	0.78
Netral	0.10	0.14	0.12
Negatif	0.96	0.93	0.95

Berdasarkan nilai F1 *score* pada Tabel 7, performa model BiLSTM menunjukkan pola yang hampir sama dengan model LSTM, namun terdapat sedikit peningkatan pada kelas netral. Secara keseluruhan, model BiLSTM memperoleh akurasi pengujian sebesar 90.32%, yang menunjukkan performa yang sedikit lebih baik dibandingkan model LSTM.

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 3, Maret 2026
e-ISSN 2685-998X
DOI 10.30865/json.v7i3.9390

3.4 Visualisasi Data Prediksi

Pada tahap ini dilakukan visualisasi data prediksi dengan menampilkan *word cloud* untuk menampilkan kata-kata yang paling sering muncul pada ulasan pengguna Gojek. Data yang ditampilkan dalam bentuk *word cloud* adalah data hasil prediksi Model LSTM dan BiLSTM pada data uji. Gambar 9 berikut menampilkan hasil *word cloud* LSTM untuk sentimen positif, netral, dan negatif.

Gambar 9. Visualisasi *Word Cloud* LSTM

Pada Gambar 9 ditampilkan tiga *word cloud* yang merepresentasikan sentimen positif, netral, dan negatif dari hasil prediksi model LSTM. *Word cloud* pada bagian kiri menunjukkan sentimen positif, bagian tengah menunjukkan sentimen netral, dan bagian kanan menunjukkan sentimen negatif. Pada sentimen positif, kata-kata yang sering muncul antara lain "mudah", "banyak", "pesan", "driver", "gojek", dan "aplikasi". Namun, terdapat pula kata "tapi" yang cenderung memiliki makna negatif. Hal ini menunjukkan bahwa model LSTM masih memprediksi sebagian ulasan negatif sebagai positif. Pada sentimen netral, muncul kata-kata seperti "tidak bisa", "tapi", dan "tolong" yang cenderung mengandung makna keluhan. Sementara itu, pada sentimen negatif kata yang dominan adalah "tidak bisa", "tapi", "aplikasi", "pesan", "driver", dan "gojek", yang menunjukkan adanya keluhan pengguna terhadap layanan atau aplikasi. Gambar 10 berikut menampilkan *word cloud* dari hasil prediksi model BiLSTM untuk kelas sentimen positif, netral, dan negatif.

Gambar 10. Visualisasi *Word Cloud* BiLSTM

Pada Gambar 10 ditampilkan tiga *word cloud* hasil prediksi model BiLSTM, di mana bagian kiri menunjukkan sentimen positif, bagian tengah menunjukkan sentimen netral, dan bagian kanan menunjukkan sentimen negatif. Pada sentimen positif terlihat kata-kata yang sesuai dengan sentimen positif seperti "cepat", "bagus", dan "mudah". Selain itu, kemunculan kata yang cenderung bermakna negatif seperti "tapi" lebih sedikit dibandingkan pada model LSTM. Pada sentimen netral, kata-kata yang sering muncul antara lain "pesan", "saran", dan "tolong". Sedangkan pada sentimen negatif, kata yang dominan adalah "tidak bisa" dan "tapi". Hal ini menunjukkan bahwa model BiLSTM mampu menangkap kata-kata yang berkaitan dengan keluhan pengguna dengan lebih baik.

3.5 Pembahasan Hasil

Berdasarkan hasil pelabelan data, kelas negatif merupakan kelas yang paling dominan, sedangkan kelas netral memiliki jumlah data paling sedikit. Ketidakseimbangan jumlah data tersebut dapat menyebabkan model kesulitan memahami pola pada kelas netral sehingga performa prediksi pada kelas tersebut menjadi rendah. Untuk mengatasi permasalahan tersebut, penelitian ini menerapkan teknik penyeimbangan data menggunakan *Random Oversampling*. Sebelum proses penyeimbangan data dilakukan, seluruh teks ulasan terlebih dahulu melalui tahap *preprocessing* yang meliputi seleksi fitur, pembersihan teks, tokenisasi, normalisasi kata, penghapusan *stopword*, dan *stemming*. Setelah itu dilakukan pembagian data dan proses *Random Oversampling* pada data latih sehingga jumlah data pada setiap kelas menjadi seimbang. Distribusi data yang lebih seimbang memungkinkan model mempelajari pola pada kelas minoritas, khususnya kelas netral, secara lebih optimal [19]. Selanjutnya, setiap teks ulasan dilakukan ekstraksi fitur menggunakan *word embedding FastText*. Pada tahap ini teks yang sebelumnya berbentuk kata ditransformasikan ke dalam bentuk numerik serta dilakukan *padding* hingga panjang maksimum

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 3, Maret 2026
e-ISSN 2685-998X
DOI 10.30865/json.v7i3.9390

100 token. Representasi numerik tersebut kemudian digunakan sebagai masukan pada *embedding layer* sehingga model dapat mempelajari hubungan antar kata selama proses pelatihan.

Hasil pelatihan menunjukkan bahwa model BiLSTM memiliki akurasi yang lebih tinggi dibandingkan LSTM. Model BiLSTM mencapai akurasi pelatihan sebesar 98% dan akurasi validasi sebesar 89%, sedangkan model LSTM mencapai akurasi sebesar 88%. Meskipun pelatihan telah menggunakan teknik *Random Oversampling* untuk menyeimbangkan jumlah data antar kelas, nilai *loss* pada kedua model masih menunjukkan indikasi *overfitting*, di mana *training loss* jauh lebih rendah dibandingkan *validation loss* yang masih berada di sekitar 0.5. Hal ini menunjukkan bahwa model mampu mempelajari pola pada data latih dengan baik, namun performanya pada data validasi masih belum optimal. Hasil pada tahap pengujian menunjukkan bahwa model BiLSTM memiliki akurasi yang lebih tinggi pada data uji, yaitu sebesar 90.32%, sedangkan model LSTM hanya mencapai 88.15%. Berdasarkan nilai F1 *score*, model LSTM menunjukkan performa yang cukup baik pada kelas positif (73%) dan sangat baik pada kelas negatif (94%). Namun, pada kelas netral performanya sangat rendah, yaitu hanya sebesar 7%, sehingga model mengalami kesulitan dalam memprediksi ulasan dengan sentimen netral. Sementara itu, model BiLSTM menunjukkan peningkatan dibandingkan model LSTM. Nilai F1 *score* pada kelas positif, netral, dan negatif masing-masing mencapai 78%, 12%, dan 95%. Meskipun teknik *Random Oversampling* telah diterapkan, model LSTM dan BiLSTM mampu memahami dan memprediksi beberapa data pada kelas netral, namun kelas netral tetap sulit diprediksi secara akurat berdasarkan hasil evaluasi.

Berdasarkan hasil visualisasi data, terlihat perbedaan pola kata yang ditangkap oleh model LSTM dan BiLSTM pada masing-masing kelas sentimen. Pada model LSTM masih terdapat kata yang cenderung bermakna negatif muncul pada sentimen positif, sehingga mengindikasikan adanya kesalahan dalam memahami konteks ulasan. Pada kelas netral, kedua model masih menampilkan kata-kata yang berkaitan dengan keluhan pengguna. Kondisi ini menunjukkan bahwa model mengalami kesulitan dalam membedakan antara sentimen netral dan negatif. Sementara itu, model BiLSTM menampilkan pola kata yang lebih sesuai dengan masing-masing kelas sentimen, sehingga mengindikasikan bahwa model tersebut mampu menangkap konteks kata dengan lebih baik dibandingkan model LSTM.

Penelitian oleh Musfiroh, Tholib, dan Arifin menunjukkan bahwa model LSTM memperoleh akurasi sebesar 83% pada klasifikasi tiga kelas, yaitu positif, netral, dan negatif. Penelitian tersebut menghadapi keterbatasan pada kelas netral yang jumlah datanya sangat sedikit serta tidak menerapkan teknik penyeimbangan data. Selain itu, pelabelan dilakukan secara otomatis berdasarkan rating sehingga kelas netral masih mengandung sentimen campuran, berbeda dengan penelitian ini yang menggunakan pelabelan manual. Meskipun penelitian ini menerapkan penyeimbangan data dan *word embedding FastText*, akurasi pengujian yang diperoleh adalah 88.15% [8]. Penelitian Alghifari, Edi, dan Firmansyah melaporkan bahwa metode BiLSTM memperoleh akurasi sebesar 91%. Namun berdasarkan *confusion matrix*, kelas netral tidak terprediksi oleh model dan sebagian besar diklasifikasikan sebagai positif atau negatif [6]. Dalam penelitian ini, model BiLSTM memperoleh akurasi pengujian sebesar 90.32%, sedikit lebih tinggi dibandingkan LSTM. Meskipun performa pada kelas netral masih rendah, model tetap mampu memprediksi sebagian data netral dengan benar.

4. KESIMPULAN

Hasil akhir penelitian ini menunjukkan bahwa sistem analisis sentimen pada ulasan pengguna Gojek menggunakan algoritma *Long Short-Term Memory* (LSTM) mampu mengelompokkan ulasan ke dalam tiga kategori sentimen, yaitu positif, netral, dan negatif. Model LSTM menunjukkan performa yang baik dengan akurasi 88.15% dan F1 *score* masing-masing positif 73%, netral 7%, dan negatif 94%, sedangkan model *Bidirectional Long Short-Term Memory* (BiLSTM) sebagai pembanding memberikan peningkatan performa dengan akurasi 90.32% dan F1 *score* positif 78%, netral 12%, dan negatif 95%. Teknik penyeimbangan data seperti *Random Oversampling* membantu model memahami dan memprediksi kelas netral, meskipun kelas ini masih sulit diprediksi secara akurat. Implikasi utama penelitian ini adalah rekomendasi penggunaan BiLSTM untuk dataset ulasan Gojek, penerapan teknik penyeimbangan data untuk membantu model memahami kelas yang minoritas, serta pemanfaatan *word embedding FastText* dan *preprocessing* untuk meningkatkan performa klasifikasi tiga kelas. Selain itu, proses *labeling* secara manual membantu menangkap sentimen positif, netral, dan negatif secara lebih akurat. Namun, kelas netral masih sulit diprediksi karena jumlah data relatif sedikit dan ulasan yang benar-benar bersifat netral lebih sulit ditemukan. Keterbatasan penelitian ini terletak pada jumlah data kelas netral yang terbatas. Oleh karena itu, penelitian selanjutnya disarankan menambah data netral yang memiliki kata-kata unik serta menggunakan teknik penyeimbangan lain atau variasi *preprocessing* agar model dapat memprediksi kelas netral secara lebih akurat.

REFERENCES

[1] Annatasya, D. Hanggoro, M. H. Hasmira, A. S. Putera, A. Adriyani, dan N. S. Kamal, "Perkembangan Teknologi Informasi Menciptakan Inovasi di Bidang Transportasi Online: Ojek Online," *Social Empirical*, vol. 1, no. 2, hlm. 154–160, Des 2024, doi: 10.24036/scemp.v1i2.39.

[2] J. Jumhadi dan A. S. Mulyani, "Perkembangan Industri Transportasi Ojek Online di Era 5.0 Dari Pt. Gojek Indonesia," *Jurnal Cakrawala Ilmiah*, vol. 2, no. 6, hlm. 2393–2402, Feb 2023, doi: 10.53625/jcijurnalcakrawalailmiah.v2i6.4907.

[3] V. Umarani, A. Julian, dan J. Deepa, "Sentiment Analysis using various Machine Learning and Deep Learning Techniques," *Journal of the Nigerian Society of Physical Sciences*, vol. 3, no. 4, hlm. 386–395, Nov 2021, doi: 10.46481/jnsps.2021.308.

[4] Y. Bengio, Y. Lecun, dan G. Hinton, "Deep learning for AI," *Commun. ACM*, vol. 64, no. 7, hlm. 58–65, Jul 2021, doi: 10.1145/3448250.

[5] S. J. Pipin dan H. Kurniawan, "Analisis Sentimen Kebijakan MBKM Berdasarkan Opini Masyarakat di Twitter Menggunakan LSTM," *Jurnal SIFO Mikroskil*, vol. 23, no. 2, hlm. 197–208, Okt 2022, doi: 10.55601/jsm.v23i2.900.

[6] D. R. Alghifari, M. Edi, dan L. Firmansyah, "Implementasi Bidirectional LSTM untuk Analisis Sentimen Terhadap Layanan Grab Indonesia," *Jurnal Manajemen Informatika (JAMIKA)*, vol. 12, no. 2, hlm. 89–99, Okt 2022, doi: 10.34010/jamika.v12i2.7764.

[7] A. R. Isnain, H. Sulistiani, B. M. Hurohman, A. Nurkholis, dan S. Styawati, "Analisis Perbandingan Algoritma LSTM dan Naive Bayes untuk Analisis Sentimen," *Jurnal Edukasi dan Penelitian Informatika (JEPIN)*, vol. 8, no. 2, hlm. 299–303, Agu 2022, doi: 10.26418/jp.v8i2.54704.

[8] M. Musfiroh, A. Tholib, dan Z. Arifin, "Analisis Sentimen Terhadap Ulasan Aplikasi Shopee di Google Play Store Menggunakan Metode TF-IDF dan Long Short-Term Memory (LSTM)," *Journal of Electrical Engineering and Computer (JEECOM)*, vol. 6, no. 2, hlm. 371–381, Okt 2024, doi: 10.33650/jeecom.v6i2.8713.

[9] I. G. B. A. Budaya, L. P. S. Pratiwi, dan D. P. Agustino, "Klasifikasi Sentimen untuk Analisis Kepuasan Pelayanan Puskesmas Berbasis Arsitektur LSTM," *Smart Comp: Jurnalnya Orang Pintar Komputer*, vol. 12, no. 4, hlm. 941–948, Okt 2023, doi: 10.30591/smartcomp.v12i4.5361.

[10] M. P. Purba dan Y. T. Wijaya, "Analisis Basic Emotion Masyarakat Pada Masa Pandemi COVID-19 di Media Sosial Twitter Dengan Metode LSTM-FastText," *Seminar Nasional Official Statistics*, vol. 2022, no. 1, hlm. 643–654, Nov 2022, doi: 10.34123/semnasoffstat.v2022i1.1524.

[11] M. N. P. Ma'ady, D. D. Rizaldy, R. F. Satria, dan P. Anaking, "SPARRING: Sistem Rekomendasi Peneliti Terintegrasi Google Scholar via SerpAPI dan Latent Dirichlet Allocation pada Konteks Perguruan Tinggi," *Jurnal Teknologi dan Manajemen Informatika*, vol. 9, no. 2, hlm. 161–171, Des 2023, doi: 10.26905/jtmi.v9i2.11111.

[12] A. Chusen, Moh. R. Advan, L. Chantika, M. A. S. N., dan F. Faadihilah, "Analisis Pengukuran Kepuasan Pengguna Aplikasi Gojek di Surabaya Menggunakan Metode End User Computing Satisfaction (EUCS)," *Prosiding Seminar Nasional Teknologi dan Sistem Informasi*, vol. 2, no. 1, hlm. 120–130, Sep 2022, doi: 10.33005/sitasi.v2i1.278.

[13] I. Menarianti *dkk.*, *Konsep Dasar Machine Learning di Era Digital*. Batam: Cendikia Mulia Mandiri, 2025.

[14] E. Alshdaifat, D. Alshdaifat, A. Alsarhan, F. Hussein, dan S. M. F. S. El-Salhi, "The effect of preprocessing techniques, applied to numeric features, on classification algorithms' performance," *Data (Basel).*, vol. 6, no. 2, hlm. 1–23, Feb 2021, doi: 10.3390/data6020011.

[15] D. Kurniawan, *Pengenalan Machine Learning dengan Python*. Jakarta: PT Elex Media Komputindo, 2023.

[16] M. A. Nur dan N. Wardhani, "Optimasi Normalisasi Kata Pada Data Twitter Untuk Meningkatkan Akurasi Analisis Sentimen (Studi Kasus Respon Masyarakat Terhadap Layanan Teman Bus)," *Jurnal Fokus Elektroda : Energi Listrik, Telekomunikasi, Komputer, Elektronika dan Kendali)*, vol. 7, no. 4, hlm. 237–243, Nov 2022.

[17] Rianto, A. B. Mutiara, E. P. Wibowo, dan P. I. Santosa, "Improving the accuracy of text classification using stemming method, a case of non-formal Indonesian conversation," *J. Big Data*, vol. 8, no. 1, hlm. 1–16, Des 2021, doi: 10.1186/s40537-021-00413-1.

[18] E. Utami, A. D. Hartanto, S. Adi, R. B. S. Putra, dan S. Raharjo, "Formal and Non-Formal Indonesian Word Usage Frequency in Twitter Profile Using Non-Formal Affix Rule," dalam *2019 1st International Conference on Cybernetics and Intelligent System, ICORIS 2019*, IEEE, Agu 2019, hlm. 173–176. doi: 10.1109/ICORIS.2019.8874908.

[19] Md. A. Talukder *dkk.*, "A hybrid deep learning model for sentiment analysis of COVID-19 tweets with class balancing," *Sci. Rep.*, vol. 15, no. 1, hlm. 27788, 2025, doi: 10.1038/s41598-025-97778-7.

[20] J. Sangeetha dan U. Kumaran, "A hybrid optimization algorithm using BiLSTM structure for sentiment analysis," *Measurement: Sensors*, vol. 25, hlm. 100619, Feb 2023, doi: 10.1016/j.measen.2022.100619.

[21] T. R. Mahesh *dkk.*, "Transformative Breast Cancer Diagnosis using CNNs with Optimized ReduceLROnPlateau and Early Stopping Enhancements," *International Journal of Computational Intelligence Systems*, vol. 17, no. 1, hlm. 14, 2024, doi: 10.1007/s44196-023-00397-1.

[22] M. Fahmy Amin, "Confusion Matrix in Three-class Classification Problems: A Step-by-Step Tutorial," *Journal of Engineering Research*, vol. 7, no. 1, 2023, doi: 10.21608/erjeng.2023.296718.

[23] J. A. Wibowo, V. C. Mawardi, dan T. Sutrisno, "Visualisasi Word Cloud Hasil Analisis Sentimen Berbasis Fitur Layanan Aplikasi Gojek dengan Support Vector Machine," *Jurnal Serina Sains, Teknik dan Kedokteran*, vol. 2, no. 1, hlm. 61–70, Mar 2024, doi: 10.24912/jsstk.v2i1.32058.