

# Perbandingan Kinerja *Word Embedding* dalam Analisis Sentimen Ulasan Pengguna Aplikasi Perjalanan

<http://dx.doi.org/10.28932/jutisi.v11i1.9681>

Riwayat Artikel

Received: 06 Agustus 2024 | Final Revision: 07 Maret 2025 | Accepted: 05 Februari 2025

Creative Commons License 4.0 (CC BY – NC)



Muhammad Agung Maugi Pahendra<sup>✉ #1</sup>, Siska Anraeni<sup>#2</sup>, Lutfi Budi Ilmawan<sup>#3</sup>

<sup>#</sup> Program Studi Teknik Informatika, Universitas Muslim Indonesia  
Jl. Urip Sumoharjo No.km.5, Makassar, 90231, Indonesia

<sup>1</sup>muhagungmaugi60244@gmail.com

<sup>2</sup>siska.anraeni@umi.ac.id

<sup>3</sup>lutfibudi.ilmawan@umi.ac.id

<sup>✉</sup>Corresponding author: muhagungmaugi60244@gmail.com

**Abstrak** —Traveloka, sebagai salah satu platform pemesanan perjalanan terkemuka, telah mencapai lebih dari 50 juta unduhan di Google Play Store. Pencapaian ini menunjukkan tingginya minat dan kepercayaan pengguna terhadap layanan yang ditawarkan. Namun, ulasan pengguna mengindikasikan adanya beberapa isu terkait performa dan kestabilan aplikasi yang perlu diperhatikan. Penelitian ini membandingkan performa metode *Word Embedding Word to Vector* (Word2vec) dan *Embedding from Language Model* (ELMo) menggunakan model *Bidirectional Long Short Term Memory* (BiLSTM) dalam analisis sentimen ulasan aplikasi Traveloka. Hasil penelitian menunjukkan bahwa model BiLSTM dengan Word2vec memiliki akurasi 76,13%, *precision* 75,22%, *recall* 77,99%, dan *F1-measure* 76,58%, lebih baik dibandingkan model dengan ELMo memiliki akurasi 74,38%, *precision* 70,49%, *recall* 78,77% dan *F1-measure* 74,40%. Model BiLSTM dengan Word2vec lebih efektif dalam analisis sentimen ulasan Traveloka, membantu mengidentifikasi dan menangani isu-isu pengguna guna meningkatkan kualitas layanan dan kepuasan pengguna.

**Kata kunci**— Analisis Sentimen; BiLSTM; Perbandingan Algoritma; Word Embedding.

## Performance Comparison of Word Embedding in Travel App User Review Sentiment Analysis

**Abstract** — Traveloka, as one of the leading travel booking platforms, has reached more than 50 million downloads on the Google Play Store. This achievement shows the high user interest and trust in the services offered. However, user reviews indicate that there are some issues related to application performance and stability that need attention. This research compares the performance of the *Word Embedding Word to Vector* (Word2vec) and *Embedding from Language Model* (ELMo) methods using the *Bidirectional Long Short Term Memory* (BiLSTM) model in sentiment analysis of Traveloka application reviews. The results show that the BiLSTM with Word2vec has 76.13% accuracy, 75.22% precision, 77.99% recall, and 76.58% F1-measure, better than the model with ELMo has 74.38% accuracy, 70.49% precision, 78.77% recall and 74.40% F1-measure. The BiLSTM with Word2vec is more effective in sentiment analysis of Traveloka reviews, helping to identify and address user issues to improve service quality and user satisfaction.

**Keywords—** *BiLSTM; Comparison Algorithms; Sentiment Analysis; Word Embedding.*

## I. PENDAHULUAN

Traveloka adalah salah satu aplikasi travel terkemuka di Asia Tenggara yang menyediakan layanan pemesanan tiket pesawat, hotel, dan berbagai produk perjalanan lainnya [1]. Seiring dengan perkembangan teknologi dan kemajuan dalam aplikasi mobile, platform seperti Traveloka menjadi semakin populer di kalangan pengguna. Traveloka telah mencapai lebih dari 50 juta unduhan di Google Play Store, menunjukkan tingginya minat dan kepercayaan pengguna terhadap layanan yang mereka tawarkan. Rating tinggi yang disertai dengan jutaan ulasan, baik positif maupun negatif, menjadi indikator bahwa aplikasi ini telah memenuhi kebutuhan dan harapan pengguna.

Namun, ulasan pengguna mengindikasikan adanya beberapa isu dan tantangan yang perlu diperhatikan. Masalah terkait performa dan kestabilan aplikasi sering kali menjadi keluhan pengguna. Beberapa ulasan menyebutkan bahwa aplikasi terkadang mengalami gangguan teknis, seperti lagging, crash, atau kesulitan dalam proses pembayaran. Gangguan ini dapat mengurangi kenyamanan pengguna dan mempengaruhi persepsi mereka terhadap keandalan aplikasi. Transparansi dan kejelasan informasi yang disediakan dalam aplikasi juga menjadi sorotan. Beberapa pengguna merasa bahwa informasi mengenai harga, syarat dan ketentuan, serta kebijakan pengembalian dana tidak selalu jelas atau mudah ditemukan. Hal ini dapat menimbulkan kebingungan dan ketidakpuasan di kalangan pengguna, terutama ketika terjadi perubahan atau pembatalan perjalanan. Meskipun demikian, banyak ulasan juga memberikan apresiasi terhadap berbagai fitur dan kemudahan yang ditawarkan oleh Traveloka, seperti berbagai pilihan pembayaran, promosi menarik, dan antarmuka yang *user-friendly*. Secara keseluruhan, ulasan pengguna memberikan wawasan berharga yang dapat digunakan Traveloka untuk terus memperbaiki dan mengoptimalkan layanan mereka demi kepuasan dan loyalitas pengguna yang lebih baik.

Untuk mengatasi masalah ini, analisis sentimen muncul sebagai solusi yang efektif. Analisis sentimen menggunakan teknik *word embedding* dan teknik *deep learning* dapat mengelompokkan dan menginterpretasi ulasan pengguna secara efisien, membantu mengidentifikasi pola dan tren dalam *feedback* yang diberikan. Dalam penelitian ini, menggunakan Metode *word embedding* yaitu *Word to Vector* (Word2vec) dan *Embedding from Language Model* (ELMo), dan metode *deep learning*, yaitu *Bidirectional Long Short Term Memory* (BiLSTM) untuk melakukan analisis sentimen. Dengan pendekatan ini, diharapkan dapat memberikan gambaran yang lebih jelas dan komprehensif mengenai opini pengguna terhadap aplikasi Traveloka, sehingga dapat digunakan untuk meningkatkan kualitas layanan dan kepuasan pengguna secara keseluruhan.

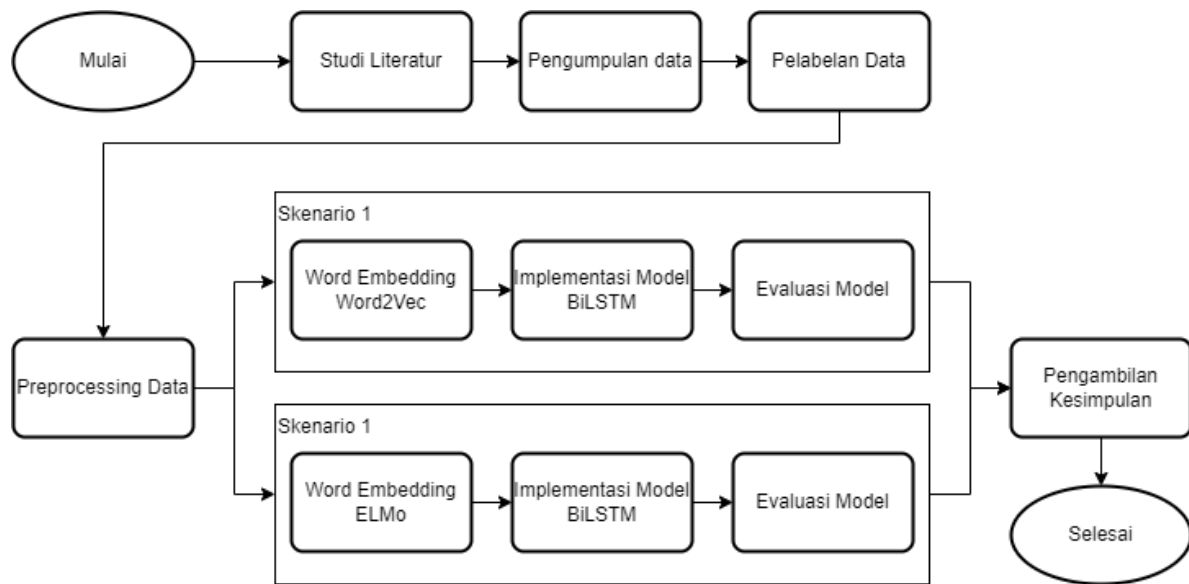
Referensi penelitian sebelumnya menunjukkan bahwa Word2Vec dan ELMo memiliki keunggulan masing-masing dalam tugas analisis sentimen. Misalnya, Peters [2] melaporkan bahwa penggunaan ELMo meningkatkan kinerja model dalam tugas analisis sentimen dan berbagai tugas NLP lainnya dengan signifikan. Hal ini dikonfirmasi oleh studi lain yang menunjukkan bahwa ELMo dapat menangkap makna kata yang lebih dalam dan bervariasi tergantung pada konteks kalimatnya, memberikan keunggulan dalam memahami nuansa sentimen dalam teks.

Penelitian yang dilakukan oleh Jain [3], mengungkapkan bahwa ELMo menunjukkan akurasi yang lebih tinggi dibandingkan metode *word embedding* lainnya dalam deteksi ujaran kebencian, menegaskan keunggulan ELMo dalam menangkap konteks kalimat yang kompleks. Dukungan lebih lanjut datang dari penelitian selanjutnya [4] yang menunjukkan bahwa ELMo melampaui Word2Vec dalam klasifikasi kalimat biomedis, mencerminkan kemampuan ELMo untuk memahami terminologi dan konteks spesifik dengan lebih baik. Namun, penelitian oleh Jain [5], menemukan bahwa akurasi Word2Vec sebanding dengan ELMo dalam klasifikasi teks media sosial terkait bencana, kemungkinan karena sifat teks media sosial yang lebih pendek dan informal, dimana Word2Vec masih dapat menangkap konteks yang cukup. Oleh karena itu, efektivitas ELMo relatif terhadap metode *word embedding* lainnya tampaknya sangat kontekstual dan tergantung pada jenis data serta tugas spesifik yang dihadapi, dengan ELMo cenderung lebih unggul dalam tugas yang memerlukan pemahaman konteks yang lebih mendalam.

Pada penelitian ini mencoba untuk melakukan perbandingan kinerja *word embedding* yaitu Word2vec dan ELMo dengan model BiLSTM untuk analisis sentimen dengan dataset ulasan pengguna Traveloka. Performa yang diukur adalah *accuracy*, *precision*, *recall* dan *f-measure*. Tujuan dari penelitian ini adalah mendapatkan hasil perbandingan performa dari dua metode *word embedding* dengan BiLSTM dan menentukan metode *word embedding* yang paling baik dalam analisis sentimen ulasan pengguna Traveloka. Harapan dari penelitian ini adalah penelitian ini dapat dijadikan referensi bagi para peneliti dalam analisis sentimen pada kasus lainnya.

## II. METODE PENELITIAN

Metodologi Penelitian ini terdiri dari beberapa tahapan seperti pada Gambar 1.



Gambar 1. Tahapan Penelitian

### A. Studi Literatur

Pada tahap ini dilakukan pengkajian berbagai literatur model seperti *e-book* dan jurnal penelitian tentang analisis sentimen, berbagai metode *word embedding* seperti Word2vec dan ELMo, dan juga pengimplementasian metode klasifikasi BiLSTM yang akan digunakan sebagai pendukung teori kajian dalam penelitian.

#### 1. Analisis sentimen

Analisis Sentimen adalah salah satu teknik dalam mengekstrak informasi berupa pandangan (sentimen) seseorang terhadap suatu isu atau kejadian [6]. Tujuan dari adanya analisis sentimen adalah untuk mengelompokkan atau mengkategorikan teks ke dalam kelas yang diberikan.

#### 2. Word2vec

Word2Vec adalah teknik *word embedding* yang digunakan dalam pemrosesan bahasa alami (*Natural Language Processing* - NLP). Teknik ini dikembangkan oleh Tomas Mikolov dan rekan-rekannya pada tahun 2013 [7]. Pendekatan ini bertujuan untuk merepresentasikan kata-kata dalam teks menjadi vektor numerik dalam ruang berdimensi rendah, di mana kata-kata dengan makna yang mirip cenderung memiliki representasi vektor yang lebih dekat satu sama lain.

#### 3. ELMo

ELMo (*Embeddings from Language Models*) adalah pendekatan *word embedding* yang dikembangkan oleh Matthew E. ELMo adalah representasi kata berkonteks mendalam yang dapat memodelkan fitur-fitur kompleks (seperti sintaksis dan semantik) dan perubahan kata dalam konteks bahasa (misalnya, pemodelan polisemi). ELMo menciptakan representasi kontekstual setiap kata dengan menggabungkan status internal biLSTM 2 lapis yang dilatih pada tugas pemodelan bahasa dua arah [2].

#### 4. BiLSTM

BiLSTM juga disebut sebagai tumpukan LSTM, sebuah improvisasi dari LSTM. *Forward layer* yang terdapat pada arsitektur LSTM dimanfaatkan untuk menyimpan konteks informasi setelahnya. Sedangkan, *backward layer* dimanfaatkan untuk menyimpan konteks informasi sebelumnya [8].

### B. Pengumpulan Data

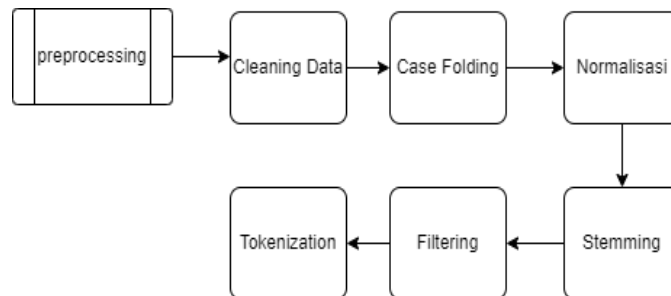
Pengumpulan data dilakukan dengan mendownload data di Kaggle. Data diambil dari kaggle dengan judul *dataset traveloka-id application rating and review* [9], yang didalamnya terdapat 4 kolom yang berisi nama, bintang, tanggal dan waktu serta ulasan dengan jumlah sekitar 63581 data. Pada penelitian ini, hanya menggunakan kolom ulasan dan kolom label sentimen. Pada kolom label diambil data dengan bintang 1 sebanyak 7930 data dan bintang 5 sebanyak 7822 data. Pada penelitian ini jumlah ulasan yang dipakai sebanyak 15752 data.

### C. Pelabelan Data

Data dilabeli berdasarkan bintang dimana jika bintang 1 diberi label 0 yang bermakna sentimen negatif dan jika bintang 5 diberi label 1 yang bermakna sentimen positif.

### D. Preprocessing Data

Pada tahap ini, data akan dilakukan preprocessing data dengan beberapa tahap seperti gambar 2.



Gambar 2. Preprocessing Data

#### 1. Cleaning Data

*Cleaning* adalah proses untuk menghilangkan atribut yang tidak diperlukan dalam teks [10]. Tujuannya adalah untuk mengurangi noise pada dataset. Contoh karakter yang dihapus termasuk tanda baca seperti titik(.), koma(,), mention(@), URL, dan angka.

#### 2. Case Folding

*Case folding* melibatkan pengubahan semua huruf dalam teks menjadi huruf kecil atau *lowercase* [11]. Hanya huruf 'a' sampai 'z' yang diterima. Karakter selain huruf hanya dianggap sebagai delimiter atau pembatas.

#### 3. Normalisation

*Normalisation* melibatkan pengubahan tiap kata yang disingkat dengan kata yang memiliki arti yang sepadan dengan kamus kata yang sesuai dengan kaidah bahasa Indonesia [12].

#### 4. Stemming

*Stemming* melibatkan proses penghilangan imbuhan pada sebuah kata agar kembali ke bentuk dasarnya. sumber kata dasar yang dipakai adalah *library* Sastrawi [13].

#### 5. filtering

*Filtering* adalah proses menyaring kata yang penting dan menghapus kata yang kurang relevan dengan menggunakan *stopwords* yang berisi list kata yang umum digunakan [14].

#### 6. Tokenizing

*Tokenizing* melibatkan pemotongan string masukan berdasarkan kata-kata yang membentuknya. Ini merupakan proses pemecahan kalimat menjadi kata-kata [15]. Pada penelitian ini pemecahan kalimat menjadi unigram kata.

### E. Implementasi Word Embedding

*Word embedding* adalah teknik yang umum digunakan untuk merepresentasikan kata-kata dalam bentuk vektor berdimensi rendah. Pada penelitian ini menerapkan metode Word2vec dengan model Skip-gram dan metode ELMo yang sudah pernah dilatih sebelumnya (ElmoForManyLangs) dengan menggunakan dataset Indonesian CONLL 2017 [16], [17].

### F. Implementasi Model

Pada tahap ini, dilakukan proses pengklasifikasian data dengan menggunakan algoritma BiLSTM.

### G. Evaluasi Model

Dalam tahap ini, dilakukan perhitungan performa berupa *accuracy*, *precision*, *recall*, dan *f-measure* serta membandingkan metode mana yang paling baik diantara dua metode *word embedding* dengan BiLSTM terhadap dataset ulasan aplikasi Traveloka.

#### 1. Accuracy

*Accuracy* adalah nilai perbandingan prediksi *True Positive* (TP) dan *True Negative* (TN) dengan jumlah keseluruhan data [18].

#### 2. Precision

*Precision* adalah nilai perbandingan prediksi *True Positive* (TP) dengan keseluruhan data yang diprediksi positif [18]. *Precision* digunakan untuk menentukan seberapa sering data benar ketika memprediksi positif.

### 3. Recall

*Recall* adalah nilai perbandingan prediksi *True Positive* (TP) dengan keseluruhan data yang benar positif [18]. *Recall* mendefinisikan bagaimana data benar untuk semua data prediksi.

### 4. F-measure

*F-measure* adalah nilai rata-rata harmonik dari *precision* dan *recall* [18].

## H. Pengambilan Kesimpulan

Kesimpulan diambil berdasarkan hasil perbandingan performa (*accuracy*, *precision*, *recall*, dan *f-measure*) metode *word embedding* Word2vec dan ELMo dengan BiLSTM terhadap dataset ulasan aplikasi Traveloka.

## III. HASIL DAN PEMBAHASAN

### A. Pengumpulan Data

Data yang diambil dari kaggle terlihat pada Tabel 1.

TABEL 1  
DATASET DARI KAGGLE

No	Nama	Bintang (Label)	Tanggal dan Waktu	Ulasan
1	Edy Suranto	5.0	11/17/2023 9:33	Aplikasi yang sangat bagus, banyak sekali promo yang ditawarkan, banyak juga pilihan destinasi wisata juga, kalo ada hal mendadak yang menjadikan tidak bisa berangkat ada fitur reschedule, jadi tidak worry lagi, terima kasih Traveloka.
2	Vebrie Anienbie	2.0	11/10/2023 8:13	Sekarang Traveloka masuk nya susah mau lihat harga tiket atau limit aja layar nya putih, padahal dulu lancar2 aja. Alasan nya d suruh instal versi terbaru, padahal saya sudah instal tapi tetap aja Lola, malah mau lihat harga tiket aja gak tersedia terus, makin buruk aja ni aplikasi
3	Zenal Parker	5.0	11/17/2023 10:39	Aplikasi yang sangat bagus, banyak promo yang ditawarkan, banyak pilihan destinasi wisata yang lengkap, pokoknya Anti ribet dah,, ada fitur reschedule jadi tidak worry lagi makasih Traveloka

Data yang akan diolah terlihat pada Tabel 2.

TABEL 2  
DATASET YANG DIOLAH

No	Ulasan	Bintang (Label)
1	Diperbaiki lagi dong sistemnya, jangan minta login ulang terus. Jadi susah cek pesanan dan cek tiket penerbangan dan lain2 ni	1
2	Sangat membantu	1
3	Sangat membantu, dan simpel dalam pemesanan tiket	5

Berikut adalah data ulasan pengguna traveloka berdasarkan bintang yang terlihat pada Tabel 3.

TABEL 3  
JUMLAH ULASAN BERDASARKAN BINTANG

Bintang	Jumlah Ulasan
1	9381
2	1584
3	2383
4	5566
5	44567

### B. Hasil Pelabelan Data

Proses pelabelan data didasarkan pada rating bintang. Data dengan rating bintang 1 diklasifikasikan sebagai sentimen negatif dan diberi label 0, sementara data dengan rating bintang 5 diklasifikasikan sebagai sentimen positif dan diberi label 1. Rating bintang 2, 3, dan 4 dikecualikan dari proses pelabelan karena tidak memberikan informasi sentimen yang konklusif. Hasil pelabelan data terlihat pada Tabel 4.

TABEL 4  
HASIL PELABELAN DATA

No	Ulasan	Bintang (Label)
1	Diperbaiki lagi dong sistemnya, jangan minta login ulang terus. Jadi susah cek pesanan dan cek tiket penerbangan dan lain2 ni	0
2	Sangat membantu	0
3	Sangat membantu,dan simpel dalam pemesanan tiket	1

Pada Tabel 3. menunjukkan bahwa jumlah ulasan di setiap kelas tidak seimbang, maka diperlukan keseimbangan data untuk mengolah data dengan teknik *undersampling*. Teknik ini akan menyeimbangkan data dengan mengurangi jumlah data kelas mayoritas agar seimbang dengan kelas minoritas. Berikut adalah data setelah dilakukan *undersampling* terlihat pada Tabel 5.

TABEL 5  
DATA SETELAH DILAKUKAN UNDERSAMPLING

Bintang	Jumlah Ulasan	Label
1	7930	0 ( Negatif )
5	7822	1 ( Positif )

### C. Hasil Preprocessing data

Setelah dilakukan pelabelan data, selanjutnya data akan masuk pada tahap *preprocessing*. Pada tahap ini data akan diproses dengan teknik *cleaning*, *case folding*, normalisasi, *stemming*, *filtering*, dan *tokenize*. Berikut adalah hasil dari *preprocessing* data yang terlihat pada Tabel 6.

TABEL 6  
HASIL PREPROCESSING DATA

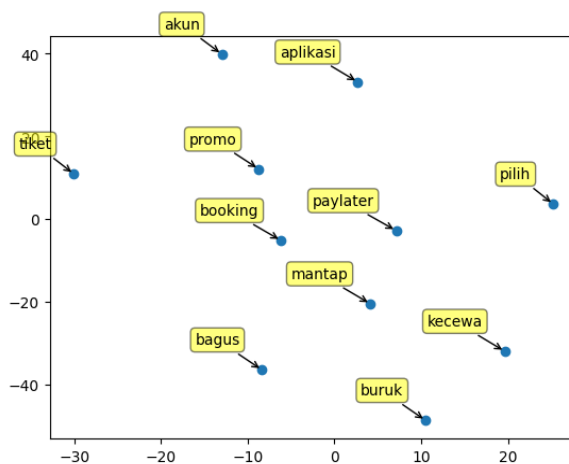
No	Ulasan	Bintang (Label)
1	['baik', 'lagi', 'dong', 'sistem', 'jangan', 'minta', 'login', 'ulang', 'terus', 'jadi', 'susah', 'cek', 'pesan', 'dan', 'cek', 'tiket', 'terbang', 'dan', 'lain', 'ni']	0
2	['sangat', 'bantu']	0
3	['sangat', 'membantudan', 'simpel', 'dalam', 'mesan', 'tiket']	1

### D. Implementasi Word Embedding

#### 1. Word2vec

Penelitian ini mengadopsi pendekatan Word2Vec dengan model *Skip-gram* untuk melakukan representasi vektor kata pada dataset ulasan Traveloka. Proses *word embedding* dilaksanakan dengan memanfaatkan pustaka Gensim. Dataset yang telah melalui tahap *preprocessing* digunakan sebagai input dalam pembentukan model Word2Vec. *Hyperparameter* yang digunakan meliputi *vector\_size* sebesar 100, *window\_size* bernilai 5, *min\_count* ditetapkan 1, *epoch* sebanyak 10, serta parameter *Skip-gram* (*sg*=1) dan *negative sampling* (*negative*=5).

Visualisasi *embedding* Word2vec berbagai kata seperti 'promo', 'akun', 'buruk', 'bagus', 'kecewa', 'paylater', 'aplikasi', 'mantap', 'pilih', 'booking', dan 'tiket' yang terlihat pada gambar 3.



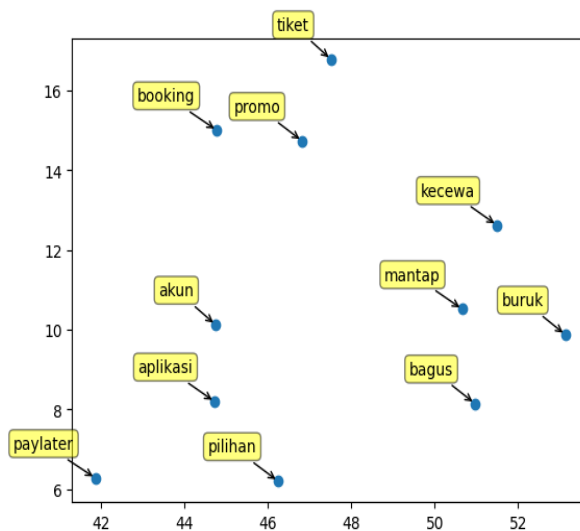
Gambar 3. Visualisasi Word2vec

Pada gambar 3. terlihat hubungan atau kedekatan antar kata seperti kata 'akun', dan 'aplikasi' membentuk kelompok tersendiri. Kata-kata tersebut memiliki kedekatan makna yang sama. Selain itu, kata seperti 'promo', 'booking', 'paylater', dan 'mantap' membentuk kelompok sendiri. Begitu juga dengan kata 'kecewa' dan 'buruk' memiliki makna yang berdekatan. Namun pada kata 'tiket' dan 'pilih' memiliki kedekatan kata yang terpisah dengan kelompok kata lainnya. Visualisasi ini memberikan gambaran tentang bagaimana Word2vec membuat penyematan kata.

## 2. ELMo

Penelitian ini memanfaatkan model bahasa pra-latih ElmoForManyLangs untuk menghasilkan representasi vektor kata (word embedding) pada korpus Indonesian CONLL 2017. Prosesnya melibatkan tokenisasi data, pembuatan *embedding* menggunakan model ElmoForManyLangs, dan penyimpanan embedding dalam format word2vec dengan bantuan pustaka Gensim. Langkah-langkah ini bertujuan untuk memperoleh representasi kata yang kaya secara semantik dan sintaktik, sehingga dapat mendukung berbagai tugas pemrosesan bahasa alami pada teks berbahasa Indonesia.

Visualisasi *embedding* ELMo berbagai kata seperti 'promo', 'akun', 'buruk', 'bagus', 'kecewa', 'paylater', 'aplikasi', 'mantap', 'pilih', 'booking', dan 'tiket' yang terlihat pada gambar 4.



Gambar 4. Visualisasi ELMo

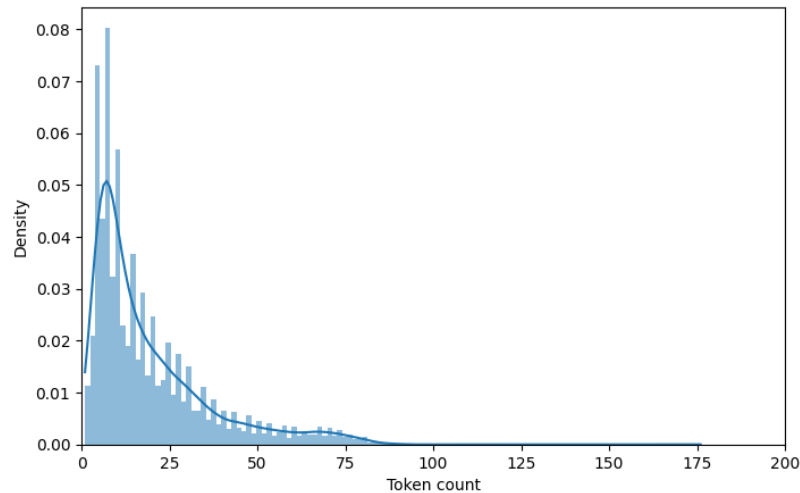
Pada gambar 4. terlihat hubungan atau kedekatan antar kata seperti kata 'booking', 'promo', dan 'tiket' membentuk kelompok tersendiri. Kata-kata tersebut memiliki kedekatan makna yang sama. Selain itu, kata seperti 'kecewa', 'mantap', 'buruk', dan 'bagus' membentuk kelompok sendiri. Begitu juga dengan kata 'akun' dan 'aplikasi' memiliki makna yang

berdekatan. Namun pada kata ‘paylater’ memiliki kedekatan kata yang terpisah dengan kelompok kata lainnya. Visualisasi ini memberikan gambaran tentang bagaimana ELMo membuat penyematan kata.

#### E. Implementasi Model

##### 1. Membuat dictionary

Proses pembuatan kamus kata (*dictionary*) pada tahap *word embedding* merupakan langkah krusial dalam membangun model klasifikasi. Setelah tahap preprocessing data, dataset yang telah dibersihkan dan disiapkan kemudian diproses menggunakan pustaka Keras Preprocessing. Langkah ini melibatkan penentuan panjang maksimum kata yang akan diinput ke model. Analisis distribusi frekuensi kata atau token pada dataset hasil *preprocessing*, seperti yang ditunjukkan pada Gambar 5, memberikan gambaran mengenai keragaman kata dalam dataset dan dapat menjadi dasar dalam memilih teknik pembobotan kata yang sesuai pada tahap selanjutnya.



Gambar 5. Visualisasi Distribusi Panjang Token

Analisis distribusi panjang token pada Gambar 5. menunjukkan bahwa sebagian besar ulasan dalam dataset memiliki panjang token antara 1 hingga 25 token. Meskipun terdapat beberapa ulasan dengan panjang token yang jauh lebih panjang, mencapai maksimum 175 token, frekuensi kemunculan token secara signifikan menurun setelah melewati batas 80 token. Berdasarkan observasi ini, panjang maksimum token ditetapkan sebesar 80 token untuk pemrosesan model. Batas ini dipilih untuk menjaga keseimbangan antara representasi kata yang cukup komprehensif dengan menghindari noise yang mungkin timbul dari token yang jarang muncul.

Proses pembuatan kamus token dilakukan menggunakan pustaka Keras. Kamus token yang dihasilkan berhasil mengidentifikasi sebanyak 14.601 token unik dari dataset yang digunakan. Kamus token ini berfungsi sebagai representasi numerik dari setiap kata atau token dalam dataset, yang selanjutnya akan digunakan sebagai input untuk model pembelajaran mesin. Hasil dari proses pembuatan kamus token ini dapat dilihat pada Tabel 7.

TABEL 7  
KAMUS KATA

Dictionary Kata/token	
Key	Value
<unk>	1
Saya	2
Tidak	3
Dan	4



Proses transformasi token menjadi urutan numerik (sequences) dilakukan dengan memanfaatkan kamus token yang telah dibangun sebelumnya. Setiap token dalam suatu teks akan dipetakan ke representasi numerik yang unik sesuai dengan definisi pada kamus token. Hasil dari proses pemetaan ini adalah sebuah matriks numerik di mana setiap baris mewakili sebuah dokumen atau kalimat, dan setiap kolom mewakili posisi token dalam urutan. Hasil transformasi ini dapat dilihat pada Tabel 8.

TABEL 8  
SEQUENCES

Pengubahan token ke sequences	
Token	sequences
['banyak', 'sekali', 'pop', 'up', 'nya', 'ganggu', 'sekali', 'sumpah']	[42, 24, 1, 1, 18, 1, 24, 1]

Proses padding dan truncation diterapkan pada sequences untuk memastikan semua urutan memiliki panjang yang sama. Padding melibatkan penambahan elemen khusus (biasanya berupa nol) pada awal atau akhir sequence untuk mencapai panjang yang diinginkan. Sebaliknya, truncation dilakukan dengan memangkas bagian awal atau akhir sequence jika panjangnya melebihi batas yang ditentukan. Teknik ini sangat penting dalam persiapan data untuk model deep learning yang memerlukan input dengan panjang yang seragam. Contoh hasil dari proses padding dan truncation dapat dilihat pada Tabel 9.

TABEL 9  
PENAMBAHAN PADDING DAN TRUNCATION KE SEQUENCES

[illegible]

## 2. Pembagian dataset

Setelah proses pembentukan kamus kata, langkah selanjutnya adalah pembagian dataset. Dataset yang telah diproses dalam bentuk sequences numerik kemudian dibagi menjadi dua bagian, yaitu variabel independen (X) dan variabel dependen (Y). Variabel X merepresentasikan sequences yang telah melalui proses padding dan truncation, sedangkan variabel Y mewakili label atau kelas yang ingin diprediksi. Pembagian data dilakukan dengan menggunakan metode hold-out dengan perbandingan 80:20, menghasilkan 12.601 data untuk pelatihan model dan 3.151 data untuk evaluasi model.

3. *Membuat model untuk klasifikasi teks*

Setelah proses pembagian dataset menjadi data latih dan uji, tahap selanjutnya adalah pembangunan model klasifikasi sentimen. Model yang digunakan dalam penelitian ini adalah model *deep learning* yang menggabungkan teknik *word embedding* dengan arsitektur *Bidirectional Long Short-Term Memory* (BiLSTM). Pada tahap awal, *word embedding* pre-trained Word2Vec dan ELMo digunakan untuk merepresentasikan kata-kata dalam bentuk vektor numerik yang bermakna secara semantik. Vektor-vektor ini kemudian dijadikan sebagai input untuk lapisan *embedding* pada model. Lapisan *embedding* ini berperan dalam memetakan kata-kata ke dalam ruang vektor berdimensi rendah.

Selanjutnya, dua lapisan BiLSTM digunakan secara berurutan untuk menangkap informasi kontekstual dari kedua arah (maju dan mundur) pada teks. Setiap lapisan BiLSTM dilengkapi dengan mekanisme *dropout* sebesar 50% untuk mencegah terjadinya *overfitting*. Setelah lapisan BiLSTM, ditambahkan lapisan *fully connected* dengan 32 neuron dan fungsi aktivasi ReLU, diikuti oleh lapisan *dropout* sebesar 50%. Lapisan terakhir adalah lapisan output dengan satu neuron dan fungsi aktivasi sigmoid yang digunakan untuk memprediksi probabilitas biner (2 kelas). Arsitektur model BiLSTM dengan Word2vec dan Arsitektur model BiLSTM dengan ELMo dapat dilihat pada Gambar 6 dan 7.

Model: "sequential"

Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, 80, 100)	1460800
bidirectional (Bidirectional)	(None, 80, 256)	234496
dropout (Dropout)	(None, 80, 256)	0
bidirectional_1 (Bidirectional)	(None, 128)	164352
dropout_1 (Dropout)	(None, 128)	0
dense (Dense)	(None, 32)	4128
dropout_2 (Dropout)	(None, 32)	0
dense_1 (Dense)	(None, 1)	33

=====  
Total params: 1863809 (7.11 MB)  
Trainable params: 1863809 (7.11 MB)  
Non-trainable params: 0 (0.00 Byte)

Gambar 6. Arsitektur BiLSTM dengan Word2vec

Model klasifikasi yang diilustrasikan pada Gambar 6 dan 7 mengadopsi arsitektur BiLSTM. Model ini terdiri dari beberapa lapisan, di antaranya lapisan embedding dengan 1024 dimensi untuk ELMo dan 100 dimensi untuk Word2vec, diikuti oleh dua lapisan BiLSTM berurutan dengan masing-masing 256 dan 128 unit neuron. Jumlah unit neuron pada kedua lapisan BiLSTM ditentukan berdasarkan dimensi embedding yang digunakan. Semakin tinggi dimensi embeddingnya, maka semakin banyak pula neuron yang digunakan. Setelah itu, terdapat lapisan fully connected dengan 32 neuron yang berperan dalam menggabungkan fitur-fitur yang telah diekstrak. Terakhir, lapisan output dengan satu neuron menghasilkan prediksi kelas.

Model: "sequential\_2"

Layer (type)	Output Shape	Param #
embedding_2 (Embedding)	(None, 80, 1024)	176905216
bidirectional_4 (Bidirectional)	(None, 80, 256)	1180672
dropout_6 (Dropout)	(None, 80, 256)	0
bidirectional_5 (Bidirectional)	(None, 128)	164352
dropout_7 (Dropout)	(None, 128)	0
dense_4 (Dense)	(None, 32)	4128
dropout_8 (Dropout)	(None, 32)	0
dense_5 (Dense)	(None, 1)	33

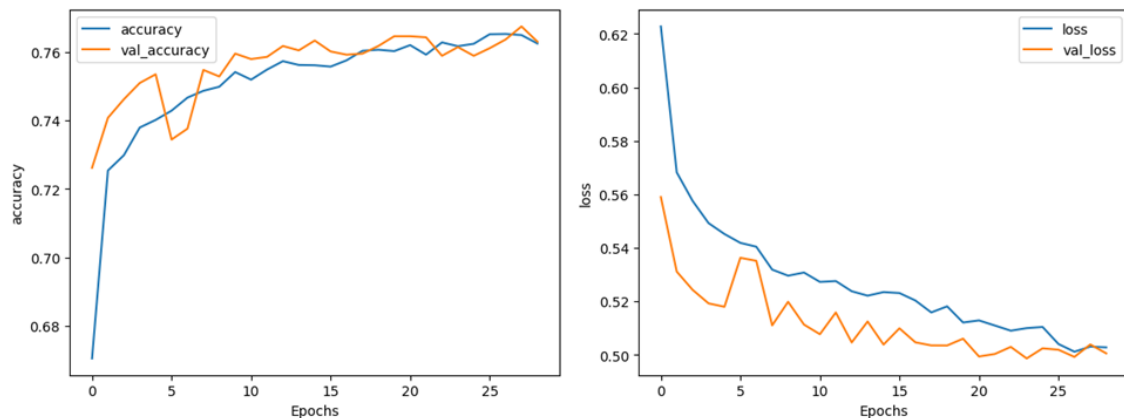
=====  
Total params: 178254401 (679.99 MB)  
Trainable params: 178254401 (679.99 MB)  
Non-trainable params: 0 (0.00 Byte)

Gambar 7. Arsitektur BiLSTM dengan ELMo

Setelah model klasifikasi dibangun, tahap selanjutnya adalah mengoptimalkan proses pelatihan. *Optimizer* yang digunakan adalah Adam dengan *learning rate* 0.0001, dipilih karena kemampuannya dalam mengadaptasi *learning rate* secara adaptif. Fungsi kehilangan yang digunakan adalah *binary cross-entropy* yang sesuai untuk masalah klasifikasi biner. Selain itu, metrik akurasi digunakan untuk memantau kinerja model selama pelatihan.

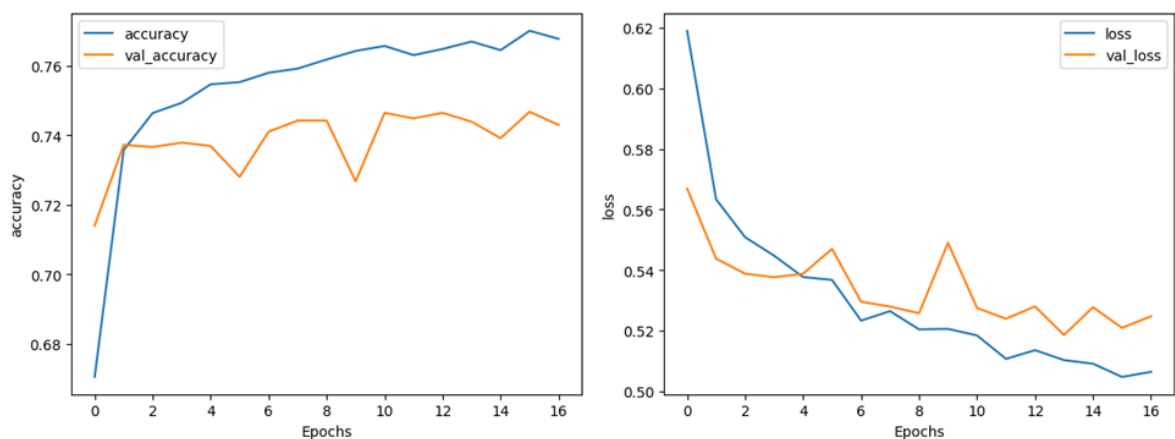
Proses pelatihan dilakukan dengan menggunakan strategi *early stopping* untuk mencegah terjadinya *overfitting*. Parameter *monitor* diatur untuk memantau nilai *loss* pada data validasi, dan proses pelatihan akan dihentikan jika nilai *loss* tidak mengalami penurunan selama 5 *epoch* berturut-turut. Model dilatih selama maksimal 50 *epoch* dengan ukuran *batch*

sebesar 32. Data latih digunakan untuk memperbarui parameter model, sedangkan data validasi digunakan untuk memantau kinerja model secara berkala dan mencegah *overfitting*.



Gambar 8. Grafik Pelatihan Model BiLSTM dengan Word2vec

Grafik pada Gambar 8. menunjukkan performa model BiLSTM dengan *embedding* Word2vec selama proses pelatihan. Grafik sebelah kiri menampilkan matrik akurasi, baik pada data latih (*accuracy*) maupun data validasi (*val\_accuracy*). Sementara itu, grafik sebelah kanan menampilkan matrik loss, yang juga diukur pada data latih (*loss*) dan data validasi (*val\_loss*). Berdasarkan hasil visualisasi proses pelatihan, dapat disimpulkan bahwa model BiLSTM dengan *embedding* Word2Vec telah menunjukkan kinerja yang cukup baik dalam tugas klasifikasi sentimen. Grafik akurasi menunjukkan tren peningkatan yang stabil pada data latih dan validasi, mengindikasikan bahwa model mampu belajar dari data dan menggeneralisasi pada data yang belum pernah ditemui. Meskipun terdapat sedikit *overfitting* yang ditunjukkan oleh gap antara akurasi latih dan validasi, namun hal ini masih dalam batas wajar. Nilai *loss* yang terus menurun pada kedua dataset juga mendukung kesimpulan bahwa model telah berhasil menemukan representasi fitur yang baik. Secara keseluruhan, hasil pelatihan ini mengindikasikan bahwa arsitektur model dan hyperparameter yang dipilih telah cukup efektif. Namun, untuk meningkatkan performa model lebih lanjut, disarankan untuk melakukan eksperimen dengan berbagai teknik regularisasi, augmentasi data, dan penyesuaian hyperparameter.



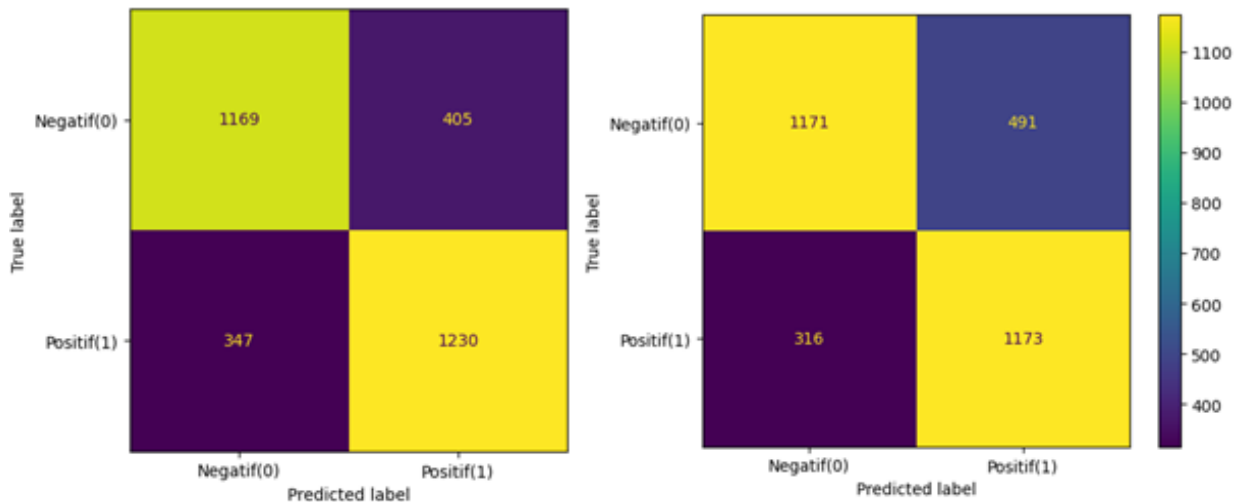
Gambar 9. Grafik Pelatihan Data Model BiLSTM dengan ELMo

Grafik pada Gambar 9. menunjukkan performa model BiLSTM dengan *embedding* ELMo selama proses pelatihan. Grafik sebelah kiri menampilkan metrik akurasi, baik pada data latih (*accuracy*) maupun data validasi (*val\_accuracy*). Sementara itu, grafik sebelah kanan menampilkan matrik loss, yang juga diukur pada data latih (*loss*) dan data validasi (*val\_loss*). Berdasarkan grafik yang disajikan, dapat disimpulkan bahwa model BiLSTM dengan *embedding* ELMo telah dilatih dengan baik. Model mampu mencapai akurasi yang cukup tinggi pada kedua dataset, dan nilai *loss* yang terus menurun menunjukkan bahwa model telah berhasil menemukan representasi fitur yang baik. Meskipun terdapat sedikit

overfitting, sebagaimana terlihat pada Gambar 9. yang menunjukkan gap 2% antara akurasi pelatihan dan validasi, hal ini masih dapat ditoleransi. Pengaruhnya terhadap kinerja model pada data yang belum terlihat dinilai masih relatif rendah.

#### F. Evaluasi Model

Setelah melatih model dengan data *training*, selanjutnya model akan diuji menggunakan data *testing* dan dievaluasi menggunakan matrik evaluasi seperti *accuracy*, *precision*, *recall*, dan *f1-measure* untuk menilai performa model secara keseluruhan. Pada gambar 10. Terlihat ada dua *confusion matrix* yaitu *confusion matrix* dari model BiLSTM dengan Word2vec (kiri) dan *confusion matrix* dari model BiLSTM dengan ELMo (kanan). Model BiLSTM dengan ELMo lebih baik dalam mengklasifikasikan ulasan negatif (TN lebih tinggi, FN lebih rendah). Namun, model ini cenderung lebih sering salah mengklasifikasikan ulasan positif sebagai negatif (FP lebih tinggi). Model BiLSTM dengan Word2vec juga cukup baik dalam mengklasifikasikan ulasan negatif, tetapi sedikit lebih buruk dibandingkan model BiLSTM dengan ELMo. Namun, model ini lebih baik dalam mengklasifikasikan ulasan positif (FP lebih rendah).



Gambar 10. Confusion Matrix BiLSTM + Word2vec (kiri) dan BiLSTM + ELMo (kanan)

Pada penelitian ini menggunakan dataset ulasan traveloka dengan mengambil 15752 dataset (7882 untuk data kelas positif dan 7930 dan untuk kelas negatif). Pembagian dataset dibagi menjadi 80% untuk data *training* dan 20% untuk data *testing*. Artinya sebanyak 12601 data untuk melatih model dan 3151 data untuk menguji performa model. Berikut adalah perbandingan performa model BiLSTM yang dipengaruhi oleh *word embedding* Word2vec Dan ELMo yang terlihat pada Tabel 10.

TABEL 10  
PERBANDINGAN KINERJA WORD2VEC DENGAN ELMO

Model	Matrik Evaluasi			
	Accuracy	Precision	Recall	F1 – measure
BiLSTM + Word2vec	76,13%	75,22%	77,99%	76,58%
BiLSTM + ELMo	74,38%	70,49%	78,77%	74,40%

Berdasarkan Tabel 8. menunjukkan hasil nilai dari performansi pada pengujian dengan penerapan *word embedding* (Word2vec dan ELMo) dengan menggunakan klasifikasi BiLSTM yang dievaluasi dengan matrik evaluasi (akurasi, *precision*, *recall*, dan *F1-measure*). Akurasi menunjukkan persentase prediksi yang benar dari keseluruhan prediksi yang dilakukan oleh model. Dari Tabel 8. dapat dilihat bahwa model BiLSTM dengan *word embedding* Word2Vec memiliki akurasi sebesar 76,13%, yang lebih tinggi dibandingkan dengan model BiLSTM dengan *word embedding* ELMo yang memiliki akurasi sebesar 74,38%. Akurasi yang lebih tinggi menunjukkan bahwa model BiLSTM dengan Word2Vec lebih sering mengklasifikasikan data dengan benar.

*Precision* dari model BiLSTM dengan Word2Vec juga lebih tinggi, yaitu sebesar 75,22%, dibandingkan dengan *precision* model BiLSTM dengan ELMo yang hanya mencapai 70,49%. Hal ini mengindikasikan bahwa model BiLSTM dengan Word2Vec lebih akurat dalam memprediksi kelas positif, dengan lebih sedikit kesalahan prediksi positif palsu.

Sementara itu, *recall* dari model BiLSTM dengan ELMo sedikit lebih tinggi, yaitu sebesar 78,77%, dibandingkan dengan *recall* model BiLSTM dengan Word2Vec yang sebesar 77,99%. Ini menunjukkan bahwa model dengan ELMo lebih baik dalam mendeteksi semua contoh positif yang ada dalam data.

Namun, ketika melihat *F1-measure*, yang merupakan rata-rata harmonis antara *precision* dan *recall*, model BiLSTM dengan Word2Vec kembali menunjukkan performa yang lebih baik dengan nilai sebesar 76,58% dibandingkan dengan model BiLSTM dengan ELMo yang memiliki *F1-measure* sebesar 74,40%. *F1-measure* yang lebih tinggi pada model BiLSTM dengan Word2Vec mengindikasikan keseimbangan yang lebih baik antara *precision* dan *recall*.

Berdasarkan Tabel 11. terlihat adanya perbedaan kinerja yang signifikan antara model Word2Vec dan ELMo dalam memprediksi sentimen ulasan. ELMo menunjukkan keunggulan dalam memprediksi sentimen ulasan panjang, sementara Word2Vec lebih baik dalam memprediksi ulasan pendek. Mengingat dominasi ulasan pendek pada dataset Traveloka dan kecenderungan ulasan panjang memiliki sentimen negatif, hal ini dapat menjelaskan mengapa *precision* model ELMo lebih rendah. Meskipun demikian, Word2Vec juga mampu memprediksi sentimen ulasan panjang dengan cukup baik, sehingga menghasilkan nilai *recall* dan *F1-measure* yang memuaskan.

TABEL 11  
PERBEDAAN PREDIKSI SENTIMEN WORD2VEC DENGAN ELMO

Ulasan	Sentimen	Prediksi Word2vec	Prediksi ELMo
Traveloka sangat membantu tetapi di saat refund sangat lama dan tidak ada kejelasan uang dikembalikan atau bagaimana	Positif	Positif	Negatif
Refund susahhhhhh bangettttt...ditelpon dijawab robot, di message diajak muter muter doang. Mau sampe kapan diberesin refund saya?????bintang 1 ga bakal saya ubah sebelum urusan saya beres. Kapok pake traveloka..sumpah kapok. Belinya doang enak..urusan refund bermasalah!!!!!!	Negatif	Positif	Negatif
Double pesanan yg saya tidak ketahui.sudah di cancel pesanan eh ternyata timbul lagi pesanan yg sama.tiba2 muncul tagihan tanpa notif..kan aneh	Negatif	Negatif	Positif
Sejauh ini pelayanan nya memuaskan lah dan memudahkan saya buat nyari penginapan di tempat yg dituju	Positif	Negatif	Positif
saya sudah lama pengajuan refund, tolong dikembalikan dana tiket pesawat saya... Perusahaan harus profesional lah kembalikan dana refundnya penumpang, klo seperti ini buruknya pelanggan akan berpindah ke aplikasi pemesanan tiket lain yg jujur	Positif	Negatif	Positif

#### IV. SIMPULAN

Penelitian ini telah melakukan perbandingan performa antara dua metode *word embedding*, Word2Vec dan ELMo, yang diterapkan pada model BiLSTM dalam analisis sentimen ulasan aplikasi Traveloka. Berdasarkan hasil evaluasi yang dilakukan, dapat disimpulkan bahwa model BiLSTM dengan *word embedding* Word2Vec menunjukkan performa yang lebih baik secara keseluruhan dibandingkan dengan model BiLSTM dengan *word embedding* ELMo, dengan akurasi sebesar 76,13%, *precision* sebesar 75,22%, *recall* sebesar 77,99%, dan *F1-measure* sebesar 76,58%. Model BiLSTM dengan *word embedding* ELMo memiliki *recall* yang sedikit lebih tinggi sebesar 78,77% dibandingkan dengan model dengan Word2Vec, namun memiliki *precision* dan *F1-measure* yang lebih rendah.

Performa Word2vec lebih baik dari ELMo disebabkan oleh penggunaan pretrained ELMo yang merupakan word embedding dari corpus berita yang kurang relevan dengan dataset ulasan traveloka. Oleh karena itu, penggunaan model BiLSTM dengan Word2Vec lebih disarankan untuk analisis sentimen ulasan aplikasi Traveloka karena keseimbangan yang lebih baik antara *precision* dan *recall*, serta performa keseluruhan yang lebih tinggi.

Saran dari penelitian ini adalah mengembangkan Dataset dengan menambahkan jumlah data yang lebih banyak agar mendapatkan model klasifikasi lebih ideal. Selain itu, melakukan preprocessing data yang lebih baik agar model dapat

belajar lebih baik dan menghasilkan prediksi yang lebih akurat. Rencana penelitian lanjutan adalah menggunakan *pretrained* model *word embedding* yang sesuai dengan dataset dan menggunakan *word embedding* word2vec dengan berbagai dimensi seperti 200 dimensi, 300 dimensi, atau 500 dimensi.

#### DAFTAR PUSTAKA

- [1] Y. Yadi, "Analisa Usability Pada Website Traveloka," J. Ilm. Betrik, vol. 9, no. 03, pp. 172–180, 2018.
- [2] M. E. Peters et al., "Deep contextualized word representations," in NAACL HLT 2018 - 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference, 2018, pp. 2227–2237.
- [3] M. Jain, P. Goel, P. Singla, and R. Tehlan, "Jain, M., Goel, P., Singla, P., Tehlan, R. (2021). Comparison of Various Word Embeddings for Hate-Speech Detection. In: Khanna, A., Gupta, D., Pólkowski, Z., Bhattacharyya, S., Castillo, O. (eds) Data Analytics and Management. Lecture Notes on Data Engineering and Communications Technologies," vol 54. Springer, Singapore, 2021, pp. 251–265.
- [4] L. Jiang, X. Sun, F. Mercaldo, and A. Santone, "DECAB-LSTM: Deep Contextualized Attentional Bidirectional LSTM for cancer hallmark classification," Knowledge-Based Syst., vol. 210, p. 106486, 2020.
- [5] P. Jain, R. Ross, and B. Schoen-Phelan, "Estimating distributed representation performance in disaster-related social media classification," in Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, New York, NY, USA: ACM., 2019, pp. 723–727.
- [6] N. Anizah, Y. Salim, and L. B. Ilmawan, "Analisis Sentimen Terhadap Event Big Sale 11.11 Shopee di Media Sosial Instagram menggunakan Metode Naïve Bayes," Bul. Sist. Inf. dan Teknol. Islam, vol. 4, no. 1, pp. 25–34., 2023.
- [7] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," 2013, [Online]. Available: <http://arxiv.org/abs/1301.3781>
- [8] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, "Neural Architectures for Named Entity Recognition," 2016, [Online]. Available: <http://arxiv.org/abs/1603.01360>
- [9] M. Gustiansyah, "Traveloka-id application rating and review dataset." Kaggle, 2024. [Online]. Available: <https://www.kaggle.com/datasets/mgustiansyah/traveloka-id-application-rating-and-review-dataset>
- [10] T. Widyanto, I. Ristiana, and A. Wibowo, "Komparasi Naïve Bayes dan SVM Analisis Sentimen RUU Kesehatan di Twitter," SINTECH (Science Inf. Technol. J., vol. 6, no. 3, pp. 147–161, 2023.
- [11] M. A. Rosid, A. S. Fitriani, I. R. I. Astutik, N. I. Mulloh, and H. A. Gozali, "Improving Text Preprocessing For Student Complaint Document Classification Using Sastrawi," IOP Conf. Ser. Mater. Sci. Eng., vol. 874, no. 1, p. 012017, 2020.
- [12] Cindy Caterine Yolanda, Syafriandi Syafriandi, Yenni Kurniawati, and Dina Fitria, "Sentiment Analysis of DANA Application Reviews on Google Play Store Using Naïve Bayes Classifier Algorithm Based on Information Gain," UNP J. Stat. Data Sci., vol. 2, no. 1, pp. 48–55, 2024.
- [13] R. Mukarramah, D. Atmajaya, and L. B. Ilmawan, "Performance comparison of support vector machine (SVM) with linear kernel and polynomial kernel for multiclass sentiment analysis on twitter," Ilk. J. Ilm., vol. 13, no. 2, pp. 168–174, 2021.
- [14] L. B. Ilmawan and M. A. Mude, "Perbandingan Metode Klasifikasi Support Vector Machine dan Naïve Bayes untuk Analisis Sentimen pada Ulasan Tekstual di Google Play Store," Ilk. J. Ilm., vol. 12, no. 2, pp. 154–161, 2020.
- [15] K. Sari, Nurdina Rasjid, and Adi Heri, "Pengelompokan Judul Penelitian Mahasiswa menggunakan Algoritma Naïve Bayes pada Program Studi Teknik Informatika," J. Comput. Inf. Syst. ( J-CIS ), vol. 5, no. 2, pp. 1–12, 2022.
- [16] W. Che, Y. Liu, Y. Wang, B. Zheng, and T. Liu, "Towards Better UD Parsing: Deep Contextualized Word Embeddings, Ensemble, and Treebank Concatenation," 2018, [Online]. Available: <http://arxiv.org/abs/1807.03121>
- [17] M. Fares, A. Kutuzov, S. Oepen, and E. Velldal, "Word vectors, reuse, and replicability: Towards a community repository of large-text resources," in Proceedings of the 21st Nordic Conference on Computational Linguistics, Gothenburg, Sweden: Association for Computational Linguistics, 2017, pp. 271–276. [Online]. Available: <http://www.aclweb.org/anthology/W17-0237>
- [18] I. W. Saputro and B. W. Sari, "Uji Performa Algoritma Naïve Bayes untuk Prediksi Masa Studi Mahasiswa," Creat. Inf. Technol. J., vol. 6, no. 1, p. 1, 2020.