

# Performance Comparison of IndoBERT and Bi-LSTM Models for Sentiment Analysis of Shopee App Users in Indonesia

Taufik Ramadhan <sup>1\*</sup>, Kusnawi <sup>2\*</sup>

\* Ilmu Komputer, Universitas Amikom Yogyakarta  
[taufikramadhan@students.amikom.ac.id](mailto:taufikramadhan@students.amikom.ac.id)<sup>1</sup>, [khusnawi@amikom.ac.id](mailto:khusnawi@amikom.ac.id)<sup>2</sup>

## Article Info

### Article history:

Received 2025-09-01  
Revised 2026-04-19  
Accepted 2026-04-29

### Keyword:

*BERT,*  
*IndoBERT,*  
*Sentiment Classification,*  
*Natural Language Processing,*  
*Ecommerce Reviews.*

## ABSTRACT

This study compares the performance of IndoBERT and Bidirectional Long Short-Term Memory (Bi-LSTM) models for sentiment classification of Indonesian-language product reviews from the Shopee e-commerce platform. The original dataset employed a 1–5 rating scale, which was reduced to two sentiment categories: negative (ratings 1–2) and positive (ratings 4–5), while reviews with a rating of 3 were excluded due to their ambiguous nature. IndoBERT (indobenchmark/indobert-base-p1) was applied through a fine-tuning process, whereas the Bi-LSTM model was trained from scratch using comprehensive text preprocessing, including case folding, stopword removal, stemming, tokenization, and padding. The dataset consisted of 7,225 reviews, divided into 5,780 training samples and 1,445 testing samples. Model performance was evaluated using accuracy, precision, recall, and F1-score metrics. Experimental results demonstrate that IndoBERT outperforms Bi-LSTM, achieving an accuracy and F1-score of 85.61%, compared to 78.27% obtained by the Bi-LSTM model. These findings indicate that transformer-based models are more effective in capturing contextual semantics in Indonesian text than recurrent neural network-based approaches.



This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

## I. PENDAHULUAN

Dalam era digital saat ini, e-commerce telah menjadi bagian penting dalam kehidupan masyarakat, dengan berbagai platform yang menawarkan kemudahan dalam berbelanja secara daring. Shopee, sebagai salah satu platform e-commerce terbesar di Asia Tenggara, memiliki jutaan pengguna yang aktif memberikan ulasan mengenai pengalaman merek. Ulasan ini mengandung informasi berharga terkait kepuasan pelanggan, kendala penggunaan, serta aspek layanan yang perlu ditingkatkan[1]. Oleh karena itu, analisis sentimen terhadap ulasan pengguna menjadi penting untuk memahami persepsi pelanggan secara lebih mendalam dan memberikan wawasan bagi pengembangan layanan yang lebih baik.

Analisis sentimen adalah teknik dalam pemrosesan bahasa alami (Natural Language Processing/NLP) yang bertujuan untuk mengidentifikasi dan mengkategorikan opini dalam suatu teks, terutama untuk menentukan apakah sentimen tersebut bersifat positif, negatif, atau netral[2]. Metode

konvensional dalam analisis sentimen, seperti pendekatan berbasis kamus dan pembelajaran mesin tradisional, memiliki keterbatasan dalam menangkap kompleksitas bahasa dan konteks semantik dalam teks [3]. Untuk mengatasi tantangan tersebut, pendekatan berbasis deep learning mulai banyak digunakan, salah satunya adalah Long Short-Term Memory (LSTM) dan pengembangannya, yaitu Bidirectional LSTM (Bi-LSTM). Bi-LSTM mampu memproses urutan kata secara dua arah, yaitu dari depan ke belakang dan dari belakang ke depan, sehingga dapat menangkap konteks kata secara lebih komprehensif dibandingkan LSTM satu arah. Pendekatan ini terbukti efektif dalam berbagai penelitian analisis sentimen karena kemampuannya memahami dependensi jangka panjang dalam teks[4]. Selain model berbasis Recurrent Neural Networks (RNN) seperti Bi-LSTM, pendekatan berbasis transformer seperti BERT (Bidirectional Encoder Representations from Transformers) juga menjadi solusi unggulan dalam memahami konteks kalimat secara lebih akurat[5]. BERT adalah model transformer yang mampu memahami hubungan antar kata dalam sebuah teks secara

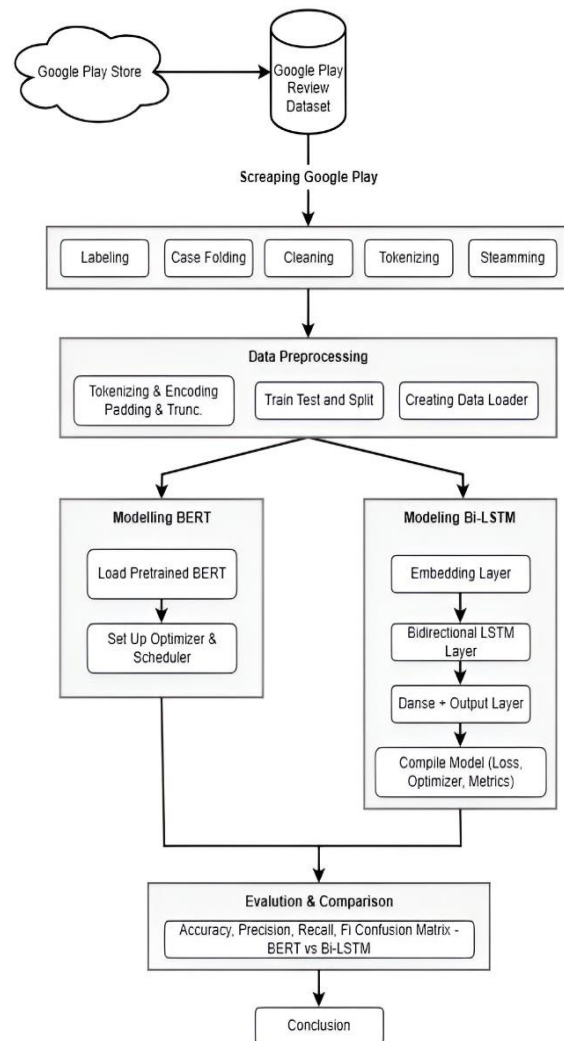
kontekstual dengan mempertimbangkan makna kata sebelum dan sesudahnya secara simultan. Kemampuan ini menjadikan BERT lebih unggul dibandingkan pendekatan RNN konvensional dalam menangkap konteks semantik yang kompleks[6]. Beberapa penelitian sebelumnya telah membuktikan efektivitas Bi-LSTM dan BERT dalam tugas analisis sentimen, terutama dalam bahasa Inggris. Namun, implementasi dan perbandingan kedua algoritma tersebut dalam analisis sentimen berbahasa Indonesia, khususnya pada ulasan e-commerce, masih memerlukan eksplorasi lebih lanjut[7]. Penelitian ini bertujuan untuk menganalisis sentimen pengguna aplikasi Shopee dengan menggunakan model BERT, khususnya IndoBERT, serta membandingkannya dengan algoritma Bi-LSTM yang telah banyak digunakan dalam penelitian NLP. Penelitian ini mencakup tahap pengumpulan data melalui teknik web scraping, preprocessing teks, pelabelan sentimen, serta pelatihan dan evaluasi model BERT dan Bi-LSTM dalam mengklasifikasikan sentimen pengguna. Dengan penelitian ini, diharapkan dapat diperoleh model analisis sentimen yang lebih akurat dan mampu memberikan wawasan yang lebih baik dalam memahami opini pengguna Shopee.

## II. METODE

Penelitian ini menggunakan pendekatan berbasis pemodelan *deep learning*, khususnya model IndoBERT, untuk mengklasifikasikan sentimen dalam ulasan pengguna aplikasi Shopee. Selain itu, penelitian ini juga menggunakan algoritma *Bidirectional Long Short-Term Memory* (Bi-LSTM) sebagai model pembanding untuk mengevaluasi efektivitas pendekatan berbasis *transformer* dibandingkan model berbasis *Recurrent Neural Network* (RNN). Proses dilakukan melalui serangkaian tahapan yang sistematis dan berurutan, dimulai dari pengumpulan data hingga evaluasi performa model [8]. Setiap tahap memiliki fungsi spesifik dalam mendukung validitas dan akurasi hasil akhir. Berikut ini adalah uraian rinci dari masing-masing tahap beserta pengertian dan fungsinya:

### A. Alur Penelitian

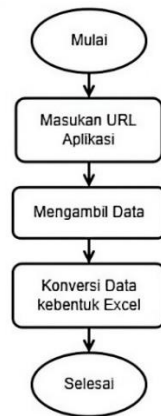
Pada alur penelitian dibawah ini merupakan tahapan dalam proses penelitian yang dilakukan, dari pengumpulan data hingga proses permodelan. Tahapan penelitian ditampilkan pada gambar 1.



Gambar 1. Alur Penelitian

### B. Scraping

Didalam penelitian ini proses pengumpulan data atau informasi pengguna aplikasi shopee dilakukan dengan teknik scraping[9] dimana teknik scraping sendiri itu merupakan teknik pengumpulan data yang diambil dengan cara otomatis pada google playstore menggunakan api dari google colab yang bernama google scrapper dan didapatkan 8000 ulasan pengguna aplikasi shopee.



Gambar 2. Tahapan Scraping

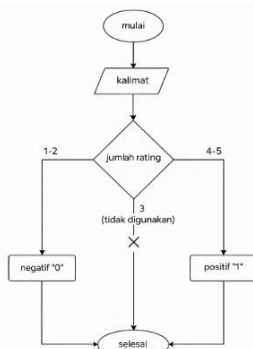
### C. Preprocessing Data

Tahap preprocessing dalam penelitian ini dibedakan berdasarkan jenis model yang digunakan. Pada model Bi-LSTM, dilakukan preprocessing lengkap yang meliputi case folding, stopword removal, stemming, dan tokenization untuk meningkatkan kualitas data dan mengurangi. Sementara itu, pada model IndoBERT, preprocessing dilakukan secara minimal, yaitu hanya mencakup case folding dan tokenization, tanpa stopword removal dan stemming, karena model BERT telah dirancang untuk memahami konteks bahasa secara langsung melalui mekanisme tokenizer dan pretraining[10]. Tujuan utama dari penelitian ini agar kualitas data menjadi meningkat dan juga mengurangi potensi masalah atau noise yang dapat mempengaruhi hasil kinerja modeling. Sehingga meningkatkan akurasi dan efektifitas analisis sentiment yang dilakukan.

#### 1) Labeling

Pada proses labeling mengacu pada penentuan kategori sentiment untuk setiap kalimat ulasan [11]. Yang mana proses labeling dilakukan secara otomatis menggunakan script dalam bahasa pemrograman python dengan skema labeling berikut :

- Rating 1-2 : label 0 (Negativ)
- Rating 3 : Label 2 (Tidak digunakan)
- Rating 4-5 Label 1 (Positif)
- 



Gambar 3. Tahapan Labeling

#### 2) Case Folding

Pada tahap Case Folding mengubah semua huruf dalam teks menjadi huruf kecil [12]. Case Folding penting dikarenakan pada tahap preprocessing data Case folding dapat membantu meningkatkan konsistensi, akurasi, dan efisiensi pada analisis sentiment.

#### 3) Tokenizing

Tahap tokenizing dalam penelitian analisis sentimen ulasan Shopee ini mengacu pada proses memecah teks menjadi unit-unit lebih kecil yang disebut token. Token ini dapat berupa kata atau frasa. Tujuan utama tokenizing adalah untuk membuat teks lebih mudah diolah, dianalisis, dan dipahami oleh sistem atau program computer[13]. Dengan memecah teks menjadi token, sistem dapat lebih mudah mengidentifikasi pola, menentukan makna, dan melakukan analisis sentiment terhadap teks tersebut.

#### 4) Steaming

Tahap stemming bertujuan untuk menghapus imbuhan dan mengembalikan kata-kata ke bentuk dasar [14]. Dalam Bahasa Indonesia, infleksi kata sering kali muncul dalam bentuk awalan, akhiran, atau pembentukan kata turunan yang dapat memengaruhi bentuk kata secara keseluruhan.

### D. Data Preparation

Persiapan data merupakan tahapan penting dalam penelitian analisis sentiment shopee. Tahap ini melibatkan serangkaian langkah untuk mengelola data mentah agar siap untuk digunakan dalam pembuatan model [15]. Tujuan utamanya adalah memastikan kualitas, konsistensi, dan relevansi data untuk menghasilkan analisis yang lebih akurat.

#### 1) Tokenizing and Encoding, Padding and trunc

Tahap tokenisasi pada penelitian ini bertujuan untuk mengubah kalimat ulasan menjadi unit token yang dapat diproses oleh model. Pada BERT, tokenisasi dilakukan menggunakan *WordPiece tokenizer*, sedangkan pada Bi-LSTM tokenisasi dilakukan dengan memecah kalimat menjadi kata-kata. Selanjutnya, tahap *encoding* mengubah token menjadi representasi numerik, berupa *token ID* pada BERT dan indeks kata atau vektor embedding pada Bi-LSTM [16]. Tahap *padding* dan *truncation* diterapkan pada kedua model untuk menyamakan panjang urutan input. *Padding* menambahkan token atau nilai nol pada kalimat yang lebih pendek, sedangkan *truncation* memotong kalimat yang melebihi panjang maksimum, sehingga data dapat diproses secara konsisten selama pelatihan dan evaluasi model.

#### 2) Train Test & Splitting

Pembagian data dalam penelitian analisis sentiment ulasan Shopee berperan penting untuk memastikan objektivitas evaluasi dan akurasi model[15]. Tahap ini melibatkan pemisahan dataset menjadi dua bagian utama, yaitu data *training* dan data *testing*. Pembagian data ini bertujuan untuk memastikan bahwa model bahasa BERT dan model Bi-LSTM

tidak hanya menghafal pola pada data pelatihan, tetapi juga mampu menggeneralisasi pengetahuan yang diperoleh pada data baru. Dengan demikian, evaluasi yang dihasilkan mencerminkan kemampuan model secara nyata dalam mengklasifikasikan sentimen ulasan pengguna secara akurat dan andal[17].

### 3) *Creating Data Loader*

Tahap *creating data loader* bertujuan untuk menyiapkan data agar dapat digunakan secara efisien pada proses pelatihan dan evaluasi model BERT dan Bi-LSTM. Data yang telah melalui proses tokenisasi, *encoding*, serta *padding* dikemas ke dalam *data loader* untuk mendukung proses *batching* dan *shuffling* [18]. Penggunaan *data loader* memungkinkan kedua model memproses data secara konsisten dan terstruktur, sehingga meningkatkan efisiensi pelatihan serta menjaga keandalan evaluasi model.

## E. Modeling Bert

Pada tahap ini, model IndoBERT digunakan untuk melakukan klasifikasi sentimen terhadap ulasan pengguna aplikasi Shopee. IndoBERT merupakan model pretrained berbasis transformer yang telah dilatih secara khusus pada korpus bahasa Indonesia, sehingga mampu menangkap konteks dan struktur kalimat dalam bahasa Indonesia dengan baik[19].

### 1) *Load Pretrained BERT*

Tahap *load pretrained BERT* bertujuan untuk memanfaatkan model BERT yang telah dilatih sebelumnya pada korpus teks berskala besar[20]. Dalam penelitian ini digunakan model IndoBERT yang telah disesuaikan dengan karakteristik bahasa Indonesia. Dengan memanfaatkan model *pretrained*, proses pelatihan menjadi lebih efisien karena model telah memiliki pemahaman awal terhadap struktur bahasa, makna kata, dan konteks kalimat. Selanjutnya, model tersebut dilakukan *fine-tuning* menggunakan data ulasan Shopee agar mampu mengklasifikasikan sentimen sesuai dengan konteks domain e-commerce secara lebih akurat.

### 2) *Setting-up Optimizer and Schedule*

Tahap *set-up BERT pretrained model serta optimizer dan scheduler* bertujuan untuk mempersiapkan model dan mekanisme pelatihan secara menyeluruh. Pada tahap ini, model IndoBERT pra-latih dimuat melalui pustaka yang sesuai sebagai dasar pemodelan *Natural Language Processing*. Penggunaan model pra-latih memungkinkan model memiliki pemahaman awal terhadap struktur bahasa dan konteks kalimat sehingga proses pelatihan menjadi lebih efisien. Selanjutnya, optimizer digunakan untuk meminimalkan nilai fungsi kerugian dengan memperbaiki parameter model secara iteratif. Pengaturan hyperparameter seperti *batch size*, *epoch*, dan *learning rate* dilakukan untuk mengontrol proses pembelajaran model[21]. Selain itu, scheduler diterapkan untuk mengatur penyesuaian *learning rate* selama pelatihan agar model dapat mencapai konvergensi yang stabil serta menghasilkan performa klasifikasi sentimen yang optimal.

## G. Modelling Bi-LSTM

Model Bi-LSTM dibangun menggunakan framework Keras TensorFlow 2.19.0. Arsitektur utama terdiri dari sebuah embedding layer diikuti oleh lapisan Bidirectional LSTM dan beberapa lapisan dense sebagai klasifikator. Detail arsitektur yang digunakan dalam penelitian ini adalah sebagai berikut: embedding layer berdimensi 100 (embedding trainable), lapisan Bidirectional LSTM dengan 128 unit, dropout 0.5 untuk regularisasi, sebuah dense layer tersembunyi dengan 64 neuron menggunakan aktivasi ReLU, serta lapisan output dengan aktivasi softmax untuk klasifikasi multi-kelas[21]. Panjang input teks distandarisasi dengan padding ke panjang maksimum 128 token. Jumlah parameter total model dilaporkan untuk memberikan gambaran kompleksitas model.

### 1) *Embedding Layer*

Embedding layer pada model LSTM atau Bi-LSTM berfungsi untuk mengubah setiap token kata yang telah diencode menjadi vektor numerik berdimensi tetap. Representasi vektor ini memungkinkan model menangkap makna semantik serta hubungan antar kata dalam suatu kalimat. Dengan menggunakan embedding layer, kata-kata yang memiliki makna serupa akan direpresentasikan dalam ruang vektor yang berdekatan, sehingga membantu LSTM atau Bi-LSTM dalam mempelajari pola sekuensial dan konteks sentimen secara lebih efektif selama proses pelatihan[22].

### 2) *Bidirectional LSTM Layer*

Lapisan *Bidirectional Long Short-Term Memory* (Bi-LSTM) berfungsi untuk mengekstraksi informasi sekuensial dari representasi kata yang dihasilkan oleh embedding layer. Bi-LSTM memproses urutan kata dari dua arah, yaitu dari awal ke akhir dan dari akhir ke awal kalimat, sehingga mampu menangkap konteks masa lalu dan masa depan secara simultan. Pendekatan ini memungkinkan model memperoleh representasi konteks yang lebih komprehensif, yang sangat penting dalam memahami makna keseluruhan kalimat pada tugas klasifikasi sentiment [23].

### 3) *Dense + Output Layer*

Lapisan *Dense* berperan sebagai lapisan *fully connected* yang mengolah fitur hasil ekstraksi dari Bi-LSTM menjadi representasi yang lebih abstrak dan diskriminatif. Fungsi aktivasi ReLU digunakan untuk mempelajari pola non-linear secara efektif. Selanjutnya, lapisan output menggunakan fungsi aktivasi softmax dengan jumlah neuron yang disesuaikan dengan jumlah kelas sentimen, sehingga menghasilkan probabilitas untuk setiap kelas dan memungkinkan penentuan label sentimen akhir [24].

### 4) *Compile Model (Loss, Optimizer, Metrics)*

Tahap kompilasi model bertujuan untuk menentukan mekanisme pembelajaran dan evaluasi model Bi-LSTM. Fungsi loss categorical *crossentropy* digunakan untuk mengukur kesalahan prediksi pada tugas klasifikasi, sementara *optimizer Adam* diterapkan untuk memperbaiki parameter model secara adaptif dan efisien. Selain itu, *metrik*

*accuracy* digunakan untuk mengevaluasi kinerja model dengan mengukur tingkat ketepatan prediksi terhadap label sentimen yang sebenarnya [25].

### III. HASIL DAN PEMBAHASAN

#### A. Input Dataset

Penelitian ini menggunakan dataset ulasan pengguna aplikasi shopee pada google store. Upaya yang dilakukan penulis untuk mendapatkan data tersebut yaitu dengan metode *screaping* dengan *library Google-Play-Screaper*, dari metode tersebut didapatkan 8000 *records* komentar ulasan dengan periode waktu mulai dari tanggal 18 Januari 2025 hingga 6 Februari 2025.

#### B. Data Preprocessing

Pra-pemrosesan data merupakan tahap yang penting dalam membersihkan dan mengadaptasi data teks sebelum diolah oleh algoritma analisis sentimen. Tujuan dari prapemrosesan ini adalah untuk meningkatkan kualitas data serta membantu algoritma dalam pemahaman teks dengan lebih efektif. Pada preprocessing dilakukan labelisasi, case folding, filterisasi dengan stopwords, tokenisasi, dan stemming.

##### 1) Labeling

Pada tahap pemberian label, klasifikasi sentimen dilakukan berdasarkan nilai rating ulasan. Rating 1–2 diberi label 0 dan dikategorikan sebagai sentimen negatif, sedangkan rating 4–5 diberi label 1 dan dikategorikan sebagai sentimen positif. Sementara itu, ulasan dengan rating 3 tidak disertakan dalam proses pelabelan karena dianggap bersifat netral dan ambigu, sehingga berpotensi menurunkan kejelasan kelas sentimen. Hasil dari proses pelabelan dapat dilihat pada tabel.

Tabel 1

Hasil Labeling

Score	Label	Category
1	0	Negative
2	0	Negatif
3	1	Netral/Tidak digunakan
4	2	Positif
5	2	Positif

Dan didapatkan hasil dari proses labeling 3.835 merupakan data positif, 3.390 data negative, serta 775 data netral/Tidak digunakan. Setelah menghapus data dengan rating 3 (netral), jumlah dataset menjadi 7.225 data.

##### 2) Case Folding

Tahap case folding dilakukan dengan mengubah seluruh huruf pada teks ulasan menjadi huruf kecil guna menyeragamkan data dan mengurangi kompleksitas kosakata. Normalisasi ini penting terutama pada pemodelan berbasis LSTM/Bi-LSTM agar token yang sama dikenali secara

konsisten. Sementara pada BERT, meskipun tokenizer telah menangani variasi huruf secara internal, proses ini tetap diterapkan untuk menjaga konsistensi data pada tahap awal penelitian.

##### 3) Filtering

Tahap filtering dilakukan dengan menghapus stopwords Bahasa Indonesia yang tidak memiliki kontribusi signifikan terhadap penentuan sentimen. Proses ini bertujuan untuk mengurangi noise dan meningkatkan fokus model LSTM/Bi-LSTM pada kata-kata bermakna. Pada model BERT, proses filtering dilakukan secara terbatas karena BERT mampu memanfaatkan konteks kata secara lebih komprehensif.

##### 4) Tokenizing

Tahap tokenizing bertujuan untuk memecah teks ulasan menjadi unit-unit token yang dapat diproses oleh model. Pada LSTM/Bi-LSTM, tokenisasi dilakukan berbasis kata yang selanjutnya diubah menjadi indeks atau vektor embedding. Sementara itu, pada BERT digunakan subword tokenization (WordPiece) yang memungkinkan model menangani kata tidak baku dan variasi bahasa secara lebih efektif.

##### 5) Steaming

Tahap stemming dilakukan untuk mengubah kata ke bentuk dasarnya guna mengurangi variasi kosakata dan menyederhanakan representasi teks. Proses ini terutama diterapkan pada pemodelan LSTM/Bi-LSTM untuk meningkatkan efektivitas pembelajaran pola sekuensial. Pada BERT, stemming tidak diterapkan secara eksplisit karena model telah dilatih untuk memahami bentuk kata secara kontekstual.

#### C. Data Preparation

Tahap persiapan data dilakukan untuk mengubah data mentah menjadi format yang siap digunakan dalam pemodelan. Penelitian ini menggunakan dua pendekatan, yaitu BERT dan Bi-LSTM. Pada model BERT, persiapan data meliputi proses tokenisasi dan encoding menggunakan tokenizer *IndoBERT-base-uncased*, pembagian data, serta pembentukan *data loader* guna mendukung proses pelatihan model. Sementara itu, pada model Bi-LSTM, data diproses melalui tahap prapemrosesan teks, tokenisasi berbasis kata, dan pembentukan representasi numerik menggunakan *embedding layer* sebelum dilakukan pembagian data. Perbedaan proses ini mencerminkan karakteristik masing-masing model dalam membangun representasi tekstual.

##### 1) Tokenizer and Encode

Langkah tokenisasi disini melibatkan penggunaan tokenizer BERT, dimana data awal berupa kalimat dipecah menjadi token atau kata-kata. Proses ini mencakup encoding, yang bertujuan untuk mengubah kalimat yang sudah dalam bentuk token menjadi token IDs agar dapat dipahami oleh model. Proses tersebut juga melibatkan penambahan token spesial, yaitu [CLS] di awal kalimat dan [SEP] di akhir kalimat. Hasil dari tahap ini dapat dilihat pada gambar 4.



```

BertForSequenceClassification(
  (bert): BertModel(
    (embeddings): BertEmbeddings(
      (word_embeddings): Embedding(50000, 768, padding_idx=0)
      (position_embeddings): Embedding(512, 768)
      (token_type_embeddings): Embedding(2, 768)
      (LayerNorm): LayerNorm((768,), eps=1e-12, elementwise_affine=True)
      (dropout): Dropout(p=0.1, inplace=False)
    )
    (encoder): BertEncoder(
      (layer): ModuleList(
        (0-11): 12 x BertLayer(
          (attention): BertAttention(
            (self): BertSdpaSelfAttention(
              (query): Linear(in_features=768, out_features=768, bias=True)
              (key): Linear(in_features=768, out_features=768, bias=True)
              (value): Linear(in_features=768, out_features=768, bias=True)
              (dropout): Dropout(p=0.1, inplace=False)
            )
            (output): BertSelfOutput(
              (dense): Linear(in_features=768, out_features=768, bias=True)
              (LayerNorm): LayerNorm((768,), eps=1e-12, elementwise_affine=True)
              (dropout): Dropout(p=0.1, inplace=False)
            )
          )
          (intermediate): BertIntermediate(
            (dense): Linear(in_features=768, out_features=3072, bias=True)
            (intermediate_act_fn): GELUActivation()
          )
          (output): BertOutput(
            (dense): Linear(in_features=3072, out_features=768, bias=True)
            (LayerNorm): LayerNorm((768,), eps=1e-12, elementwise_affine=True)
            (dropout): Dropout(p=0.1, inplace=False)
          )
        )
      )
      (pooler): BertPooler(
        (dense): Linear(in_features=768, out_features=768, bias=True)
        (activation): Tanh()
      )
      (dropout): Dropout(p=0.1, inplace=False)
      (classifier): Linear(in_features=768, out_features=2, bias=True)
    )
  )
)

```

Gambar 8. Load BERT pre-trained model

## 2) Setting-up Optimizer and Scheduler

Proses analisis sentimen menggunakan BERT pada penelitian ini menerapkan optimizer AdamW, yang dipilih karena kemampuannya dalam mengelola pembaruan bobot secara adaptif serta mengatasi ketidakstabilan *learning rate* melalui mekanisme berbasis Root Mean Square Propagation (RMSProp). Dibandingkan dengan *Stochastic Gradient Descent*, AdamW terbukti lebih efektif dalam mempercepat konvergensi dan menjaga stabilitas pelatihan model. Hyperparameter yang digunakan meliputi *batch size* sebesar 16, *learning rate*  $2e-5$ , dan 10 epoch. Penentuan jumlah epoch didasarkan pada hasil pengujian dan studi terdahulu, di mana konfigurasi 10 epoch menunjukkan kinerja terbaik dibandingkan dengan penggunaan 4 dan 16 epoch. Pemilihan *learning rate*  $2e-5$  bertujuan untuk meminimalkan risiko *catastrophic forgetting* pada proses fine-tuning BERT. Selain itu, diterapkan *learning rate scheduler* yang terintegrasi dengan optimizer untuk mengatur laju pembelajaran secara dinamis selama pelatihan guna meningkatkan performa model secara optimal.

## E. Modeling Bi-LSTM

### 1). Embedding Layer

```

Output dari Embedding Layer (Vektor Embedding):
Shape Output Embedding: (1, 128, 100)
Embedding untuk Token ID pertama (20 dimensi pertama): [ 0.01921305 -0.00752329 -0.04749007
-0.00800828 0.01800796 -0.00163504 0.01382903 -0.00244762 -0.02618865
-0.02553268 0.01555808 0.0115742 0.03823802 -0.00230552 0.02379278
-0.00662927 0.03985539]...
Embedding untuk Token ID kedua (20 dimensi pertama): [ 0.01921305 -0.00752329 -0.04749007 -0.00800828
0.01800796 -0.00163504 0.01382903 -0.00244762 -0.02618865
-0.02553268 0.01555808 0.0115742 0.03823802 -0.00230552 0.02379278
-0.00662927 0.03985539]...

```

Gambar 9. Output Embedding Layer

Gambar 9 menunjukkan hasil keluaran dari Embedding Layer pada pemodelan Bi-LSTM. Embedding layer menghasilkan representasi vektor berdimensi tetap dengan ukuran (1, 128, 100), di mana setiap token direpresentasikan sebagai vektor numerik berdimensi 100. Representasi ini memungkinkan model menangkap informasi semantik antar kata secara lebih efektif sebelum diproses oleh lapisan Bidirectional LSTM.

### 2). Bidirectional LSTM Layer

```

1/1 ----- 0s 170ms/step
Output dari Bidirectional LSTM Layer:
Shape Output Bi-LSTM: (1, 256)
Hasil Vektor Output Bi-LSTM (20 dimensi pertama): [ 3.0571602e-02 2.3843573e-02 3.4845982e-02 6.2480317e-03
-1.6496414e-02 7.2587014e-04 -1.3724494e-02 -2.2076258e-05
-2.1177113e-02 -7.7610072e-03 2.2248864e-02 -1.0213311e-02
6.2537696e-03 1.1919469e-02 -4.3451937e-04 8.0819624e-03
-6.5173274e-03 2.9632398e-03 9.9611944e-03 -8.4162373e-03]...

```

Gambar 10. Hasil Bidirectional LSTM Layer

Gambar 10 menunjukkan keluaran dari lapisan Bidirectional LSTM. Output yang dihasilkan berupa vektor berdimensi (1, 256), yang merupakan hasil penggabungan representasi sekuens dari arah maju dan mundur. Vektor ini merepresentasikan ringkasan kontekstual dari seluruh ulasan dan selanjutnya digunakan sebagai input pada lapisan Dense untuk proses klasifikasi sentimen.

### 3). Dense & Output Layer

```

1/1 ----- 0s 292ms/step
Output dari Lapisan Dense (64 Unit, ReLU):
Shape Output Dense (64): (1, 64)
Hasil Vektor Output Dense (64) (20 dimensi pertama): [0.00704397 0.
0. 0.01201519 0. 0.00479577 0.0350907 0.00409242
0.00428393 0.01152526 0. 0. 0.01403184 0.0131815
0.00426046 0.0132403 ]...

```

Gambar 11. Hasil Dense &amp; Output Layer

Berdasarkan hasil yang ditunjukkan pada Gambar 10, lapisan *Dense* dengan 64 unit neuron dan fungsi aktivasi ReLU menghasilkan keluaran dengan bentuk vektor (1, 64). Bentuk ini menunjukkan bahwa satu sampel input direpresentasikan sebagai 64 fitur numerik hasil transformasi lanjutan dari keluaran Bidirectional LSTM. Nilai-nilai pada vektor output Dense, seperti 0.00704397, 0.01201519, dan 0.0350907, mencerminkan aktivasi neuron yang merespons pola sentimen tertentu yang telah dipelajari model.

Adanya nilai nol pada beberapa dimensi output menunjukkan karakteristik fungsi aktivasi ReLU, di mana neuron dengan nilai input negatif akan dinonaktifkan. Mekanisme ini membantu model melakukan seleksi fitur secara implisit dengan hanya mempertahankan fitur yang memiliki kontribusi positif terhadap proses klasifikasi. Dengan demikian, lapisan Dense berperan sebagai *feature refinement layer* yang mengekstraksi representasi sentimen yang lebih diskriminatif sebelum tahap klasifikasi akhir.

Vektor hasil dari lapisan Dense ini kemudian diteruskan ke *output layer*, yang berfungsi untuk menghasilkan prediksi sentimen akhir. Output layer mengonversi representasi fitur berdimensi 64 tersebut menjadi nilai keluaran sesuai dengan jumlah kelas sentimen yang ditentukan. Nilai keluaran pada lapisan ini merepresentasikan skor atau probabilitas prediksi

sentimen, yang menjadi dasar dalam menentukan label akhir pada proses analisis sentimen.

Secara keseluruhan, hasil keluaran lapisan Dense dan output layer menunjukkan bahwa model Bi-LSTM mampu mengolah representasi sekuensial teks menjadi fitur tingkat tinggi yang informatif, serta melakukan pemetaan fitur tersebut ke dalam keputusan klasifikasi sentimen secara efektif.

4). *Compile Model (Loss, Optimizer, Metrics)*

Berdasarkan ringkasan model yang ditampilkan, arsitektur Bi-LSTM yang digunakan terdiri dari beberapa lapisan utama, yaitu *Embedding layer*, *Bidirectional LSTM layer*, *Dropout*, dan *Dense layer* sebagai lapisan klasifikasi akhir. Model menerima input berupa sekuens teks dengan panjang maksimum 128 token.

Lapisan *Embedding* menghasilkan output berdimensi (None, 128, 100) dengan jumlah parameter sebanyak 865.800, yang menunjukkan bahwa setiap token direpresentasikan ke dalam vektor berdimensi 100. Representasi ini memungkinkan model menangkap informasi semantik dari setiap kata dalam ulasan.

Selanjutnya, lapisan *Bidirectional LSTM* menghasilkan output berdimensi (None, 256) dengan total parameter 234.496, yang berasal dari penggabungan representasi LSTM arah maju dan mundur. Hal ini memungkinkan model mempelajari ketergantungan konteks baik dari sisi kiri maupun kanan sekuens teks.

Lapisan *Dense* terakhir memiliki output (None, 2) dengan 130 parameter, yang menunjukkan bahwa model dirancang untuk melakukan klasifikasi biner. Secara keseluruhan, model memiliki 1.116.874 parameter yang dapat dilatih, yang sepenuhnya berkontribusi dalam proses pembelajaran sentimen.

Model: "sequential\_8"

Layer (type)	Output Shape	Param #
embedding_8 (Embedding)	(None, 128, 100)	865,800
bidirectional_6 (Bidirectional)	(None, 256)	234,496
dropout_12 (Dropout)	(None, 256)	0
dense_12 (Dense)	(None, 64)	16,448
dropout_13 (Dropout)	(None, 64)	0
dense_13 (Dense)	(None, 2)	130

Total params: 1,116,874 (4.26 MB)  
 Trainable params: 1,116,874 (4.26 MB)  
 Non-trainable params: 0 (0.00 B)

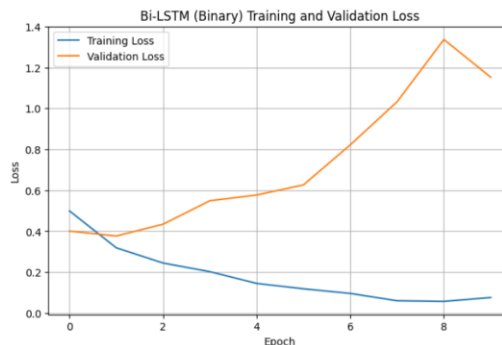
Gambar 12. *Compile Model (Loss, Optimizer, Metrics)*

• Analisis Proses Pelatihan (Training dan Validation Loss)

Grafik *training loss* dan *validation loss* menunjukkan pola pembelajaran yang kontras. Nilai *training loss* mengalami penurunan yang konsisten dari epoch awal hingga akhir, yang mengindikasikan bahwa model mampu mempelajari pola dari data latih secara efektif.

Sebaliknya, *validation loss* justru mengalami peningkatan signifikan setelah beberapa epoch awal. Kondisi ini mengindikasikan terjadinya *overfitting*, di mana model menjadi terlalu menyesuaikan diri terhadap data latih namun kehilangan kemampuan generalisasi pada data uji. Fenomena

ini menunjukkan bahwa meskipun Bi-LSTM memiliki kapasitas representasi yang cukup kuat, model ini masih sensitif terhadap kompleksitas data dan jumlah epoch yang digunakan.



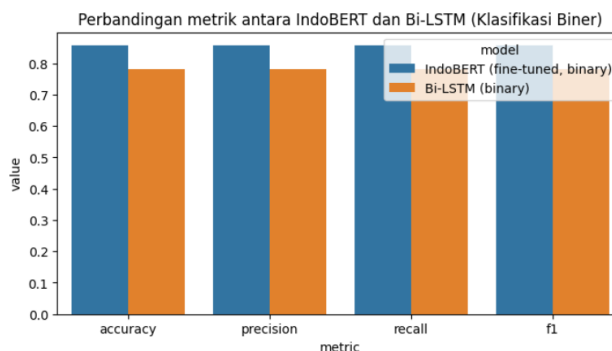
Gambar 13. Hasil Training and Validation loss

• Evaluasi Kinerja Model Bi-LSTM dan IndoBERT

Hasil evaluasi kinerja menunjukkan bahwa model IndoBERT (fine-tuned) menghasilkan performa yang lebih unggul dibandingkan dengan model Bi-LSTM pada seluruh metrik evaluasi. IndoBERT mencapai nilai akurasi 0,8561, *precision* 0,8562, *recall* 0,8561, dan *F1-score* 0,8561, yang menunjukkan performa yang seimbang dan stabil dalam klasifikasi sentimen biner.

Sementara itu, model Bi-LSTM memperoleh akurasi 0,7827, dengan nilai *precision* 0,7825, *recall* 0,7827, dan *F1-score* 0,7825. Perbedaan ini menunjukkan bahwa Bi-LSTM memiliki keterbatasan dalam menangkap konteks semantik yang kompleks dibandingkan IndoBERT, yang secara arsitektural memang dirancang untuk memahami konteks secara mendalam melalui mekanisme *self-attention*. Perbandingan ini menegaskan bahwa pendekatan berbasis *transformer* seperti IndoBERT lebih efektif dalam menangani data teks Bahasa Indonesia yang bersifat kontekstual dan ambigu dibandingkan pendekatan sekuensial konvensional seperti Bi-LSTM.

	model	accuracy	precision	recall	f1
0	IndoBERT (fine-tuned, binary)	0.856055	0.856154	0.856055	0.856090
1	Bi-LSTM (binary)	0.782699	0.782544	0.782699	0.782471



Gambar 14. Perbandingan Algoritma IndoBERT dan Bi-LSTM

- Pembahasan Compile Model (Loss, Optimizer, dan Metrics)

Model Bi-LSTM dikompilasi menggunakan fungsi *loss* untuk klasifikasi biner, yang selaras dengan jumlah output kelas sebanyak dua. Pemilihan *optimizer* memungkinkan pembaruan bobot dilakukan secara efisien selama proses pelatihan, sedangkan metrik evaluasi seperti *accuracy* digunakan untuk memantau performa model secara kuantitatif.

Hasil yang diperoleh menunjukkan bahwa meskipun konfigurasi *compile* telah sesuai, keterbatasan Bi-LSTM dalam mempertahankan performa pada data validasi mengindikasikan perlunya strategi tambahan, seperti *regularization* atau *early stopping*, untuk meningkatkan kemampuan generalisasi model.

#### IV. KESIMPULAN

Penelitian ini bertujuan untuk membandingkan kinerja model berbasis *transformer* dan *recurrent neural network* dalam melakukan analisis sentimen berbahasa Indonesia, khususnya antara IndoBERT dan Bidirectional Long Short-Term Memory (Bi-LSTM) pada skema klasifikasi sentimen biner. Berdasarkan hasil eksperimen yang telah dilakukan, kedua model mampu melakukan klasifikasi sentimen, namun menunjukkan perbedaan performa yang cukup signifikan.

Hasil evaluasi menunjukkan bahwa model IndoBERT yang telah melalui proses fine-tuning memberikan kinerja yang lebih unggul dibandingkan model Bi-LSTM pada seluruh metrik evaluasi. IndoBERT mencapai nilai akurasi sebesar 85,61%, precision 85,62%, recall 85,61%, dan F1-score 85,61%. Sebaliknya, model Bi-LSTM memperoleh nilai akurasi sebesar 78,27%, precision 78,25%, recall 78,27%, dan F1-score 78,25%. Perbedaan ini menunjukkan adanya peningkatan performa sekitar 7% pada seluruh metrik ketika menggunakan pendekatan berbasis *transformer*.

Keunggulan IndoBERT tersebut mengindikasikan bahwa model berbasis *transformer* lebih efektif dalam menangkap konteks semantik dan hubungan antar kata pada teks ulasan yang bersifat kompleks dan kontekstual. Hal ini didukung oleh mekanisme *self-attention* yang memungkinkan IndoBERT memahami dependensi kata secara menyeluruh tanpa bergantung pada urutan sekuens secara ketat, sehingga mampu menghasilkan representasi teks yang lebih kaya dibandingkan Bi-LSTM.

Sementara itu, meskipun model Bi-LSTM mampu mempelajari pola sekuensial dua arah dengan cukup baik, hasil pelatihan menunjukkan adanya kecenderungan *overfitting*, yang ditandai dengan penurunan *training loss* secara konsisten, namun diikuti oleh peningkatan *validation loss* pada epoch selanjutnya. Kondisi ini mengindikasikan bahwa model Bi-LSTM memiliki keterbatasan dalam melakukan generalisasi terhadap data yang tidak terlihat selama proses pelatihan, terutama ketika dihadapkan pada variasi bahasa yang tinggi.

Secara keseluruhan, hasil penelitian ini menegaskan bahwa pendekatan berbasis *transformer*, khususnya IndoBERT, lebih unggul dan lebih sesuai untuk tugas analisis sentimen Bahasa Indonesia dibandingkan pendekatan berbasis *recurrent neural network* seperti Bi-LSTM. Temuan ini diharapkan dapat menjadi referensi bagi penelitian selanjutnya dalam pemilihan arsitektur model analisis sentimen, serta mendorong pemanfaatan model pra-latih berbasis *transformer* untuk meningkatkan performa pengolahan bahasa alami berbahasa Indonesia.

#### DAFTAR PUSTAKA

- [1] A. H. Ruger, M. Suyanto, and M. P. Kurniawan, "Sentimen Analisis Pelanggan Shopee di Twitter dengan Algoritma Naive Bayes," *J. Inf. Technol.*, vol. 1, no. 2, pp. 26–29, 2021, doi: 10.46229/jifotech.v1i2.282.
- [2] R. C. Rivaldi, T. D. Wismarini, J. T. Lomba, and J. Semarang, "Analisis Sentimen Pada Ulasan Produk Dengan Metode Natural Language Processing (NLP) (Studi Kasus Zalika Store 88 Shopee)," vol. 17, no. 1, pp. 120–128, 2024.
- [3] N. Firdausy, I. Yuadi, and I. Puspitasari, "Analisis Sentimen Evaluasi Reaksi E-Learning Menggunakan Algoritma Naive Bayes, Support Vector Machine Dan Deep Learning," *Techno.Com*, vol. 22, no. 3, pp. 677–689, 2023, doi: 10.33633/tc.v22i3.8160.
- [4] O. Boas, J. Putro, A. Jacobus, and F. D. Kambe, "Aspect-Based Sentiment Analysis Product Review Using CNN and Bidirectional LSTM," vol. 20, no. 2, pp. 117–124, 2025.
- [5] S. W. Nadya Sikana, "Analisis Sentimen untuk Ulasan Produk E-Commerce," vol. 26, no. 2, pp. 223–238, 2025.
- [6] M. M. Shivaji Alaparthy, "Bidirectional Encoder Representations from Transformers (BERT): A sentiment analysis odyssey Shivaji," *arXiv Prepr.*, no. 1, pp. 1–15, 2020, doi: <https://doi.org/10.48550/arXiv.2007.01127>.
- [7] D. Yuliana, A. Ningrum, E. Daniati, M. N. Muzaki, and S. Informasi, "Perbandingan Model BERT dan RNN-LSTM pada Analisis Sentimen Aplikasi BRI Mobile," vol. 4, no. 2, pp. 75–85, 2025.
- [8] J. Saquer, "A Comparative Analysis of Transformer and LSTM Models for Detecting Suicidal Ideation on Reddit".
- [9] S. Alfari and Kusnawi, "Komparasi Metode KNN dan Naive Bayes Terhadap Analisis Sentimen Pengguna Aplikasi Shopee," *Indones. J. Comput. Sci.*, vol. 12, no. 5, pp. 2766–2776, 2023, doi: 10.33022/ijcs.v12i5.3304.
- [10] R. Illahi, S. Agustian, S. K. Riau, S. Baru, and K. Pekanbaru, "Bidirectional Lstm Dan Indobert Dengan Dataset Terbatas," vol. 7, no. 1, pp. 74–84.
- [11] N. Wayan, A. Sekar, and A. R. Isnain, "Analisis Sentimen Terhadap Media Sosial Twitter dengan Kasus Kampanye Anti-Korupsi di Indonesia Menggunakan Naive Bayes," *J. Media Inform. Budidarma*, vol. 8, no. April, pp. 695–703, 2024, doi: 10.30865/mib.v8i2.7582.
- [12] G. Darmawan, S. Alam, and M. I. Sulisty, "Analisis Sentimen Berdasarkan Ulasan Pengguna Aplikasi MyPertamina Pada Google Playstore Menggunakan Metode Naive Bayes," *STORAGE – J. Ilm. Tek. dan Ilmu Komput.*, vol. 2, no. 3, pp. 100–108, 2023.
- [13] M. R. R. Lillah, D. S. Maylawati, W. B. Zulfikar, W. Uriawan, and A. Wahana, "Implementasi Algoritma K-Nearest Neighbor (KNN) untuk analisis sentimen pengguna aplikasi Tokopedia," *Intellect Indones. J. Learn. Technological Innov.*, vol. 02, no. 02, pp. 171–184, 2023, [Online]. Available: [https://digilib.uinsgd.ac.id/77056/%0Ahttps://digilib.uinsgd.ac.id/77056/6/4\\_BAB I.pdf](https://digilib.uinsgd.ac.id/77056/%0Ahttps://digilib.uinsgd.ac.id/77056/6/4_BAB%20I.pdf)
- [14] N. Z. B. Jannah and K. Kusnawi, "Comparison of Naive Bayes and SVM in Sentiment Analysis of Product Reviews on Marketplaces," *Sinkron*, vol. 8, no. 2, pp. 727–733, 2024, doi: 10.33395/sinkron.v8i2.13559.
- [15] A. Ananta Firdaus, A. Id Hadiana, and A. Kania Ningsih,

- “Klasifikasi Sentimen pada Aplikasi Shopee Menggunakan Fitur Bag of Word dan Algoritma Random Forest,” *Ranah Res. J. Multidiscip. Res. Dev.*, vol. 6, no. 5, pp. 1678–1683, 2024, doi: 10.38035/rrj.v6i5.994.
- [16] N. Ray, A. Tambunan, D. Retno, and S. Saputro, “Hybrid Integration Of Bert And Bilstm Models For,” vol. 20, no. 2, pp. 1719–1730, 2026.
- [17] M. Mustafa and S. Kumar, “BERT-Enhanced Bi-LSTM with weighted cross-entropy for multilingual sentiment classification,” vol. 11, no. 3, pp. 396–416, 2025.
- [18] Y. Huang, “Sentiment Analysis of News Content Based on,” pp. 616–620, 2025, doi: 10.1145/3783669.3783764.
- [19] M. F. Cahyadi and T. H. Rochadiani, “Implementasi Ensemble Deep Learning Untuk Analisis Sentimen Terhadap Genre Game Mobile,” *J. Media Inform. Budidarma*, vol. 8, no. 3, p. 1512, 2024, doi: 10.30865/mib.v8i3.7832.
- [20] J. Kecerdasan, A. Rekeyasa, and K. Wau, “Application of Fine-Tuned IndoBERT for Sentiment Classification Local Product Reviews on Tokopedia Marketplace with Limited Dataset,” vol. 5, no. 1, pp. 1–5, 2025.
- [21] S. Aras, R. Ruimassa, E. Agustinus, B. Wambraw, and E. B. Palalangan, “Sentiment Analysis on Shopee Product Reviews Using IndoBERT,” vol. 6, no. 3, pp. 1616–1627, 2024, doi: 10.51519/journalisi.v6i3.814.
- [22] A. C. Adamuthe, “Improved Text Classification using Long Short-Term Memory and Word Embedding Technique,” vol. 13, no. 1, pp. 19–32, 2020.
- [23] Y. Wang, X. Cheng, and X. Meng, “Sentiment Analysis with An Integrated Model of BERT and Bi-LSTM Based on Multi-Head Attention Mechanism,” vol. 50, no. 1, 2023.
- [24] A. Sami, S. Buyrukoğlu, and M. Rashad, “Advanced deep learning techniques for sentiment analysis : combining Bi-LSTM , CNN , and attention layers,” vol. 11, no. 1, pp. 55–71, 2025.
- [25] C. Fri, R. Elouahbi, Y. Taki, and A. Remaida, “Enhanced Bidirectional LSTM for Sentiment Analysis of Learners ’ Posts in MOOCs,” vol. 16, no. 5, pp. 163–172, 2025.