

# Prediksi Emosi Pada Teks Berbahasa Indonesia Menggunakan Model Long-Short Term Memory (LSTM)

Muhammad Ilham Hakiki<sup>1</sup>, Ilham Firmansyah<sup>2</sup>

<sup>1,2</sup>Teknik Informatika, Fakultas Teknik dan Ilmu Komputer, Universitas Nusantara PGRI Kediri

E-mail: <sup>1</sup>[ilhamhakiki2304@gmail.com](mailto:ilhamhakiki2304@gmail.com), <sup>2</sup>[ilhamfirm02@gmail.com](mailto:ilhamfirm02@gmail.com)

**Abstrak** – Analisis emosi pada teks merupakan bagian penting dalam pengolahan bahasa alami karena mampu memberikan pemahaman terhadap opini dan perasaan pengguna dalam berbagai konteks digital. Penelitian ini bertujuan untuk mengklasifikasikan emosi pada teks berbahasa Indonesia dengan memanfaatkan model *Bidirectional Long Short-Term Memory (BiLSTM)* tanpa menggunakan model *pretrained*. Pemilihan topik ini didasarkan pada kebutuhan akan model klasifikasi emosi yang efisien, ringan, dan dapat diimplementasikan pada lingkungan dengan keterbatasan sumber daya komputasi. Metode penelitian meliputi tahap *praproses data*, vektorisasi teks menggunakan *TextVectorization*, pembagian data latih dan validasi secara *terstratifikasi*, serta pelatihan model *BiLSTM* bertingkat dengan penerapan *class weighting* dan *regularisasi* untuk mengatasi ketidakseimbangan kelas dan *overfitting*. Evaluasi dilakukan menggunakan metrik akurasi, grafik pelatihan, dan *confusion matrix*. Hasil eksperimen menunjukkan bahwa model *BiLSTM* yang diusulkan mencapai akurasi validasi sebesar 86.01% dengan proses pelatihan yang stabil berdasarkan grafik akurasi dan *loss*. Analisis *confusion matrix* memperlihatkan bahwa model mampu mengklasifikasikan emosi dengan ciri leksikal yang jelas secara akurat, sementara kesalahan klasifikasi lebih sering terjadi pada kelas emosi yang memiliki kedekatan semantik.

**Kata Kunci** — Analisis Emosi, *BiLSTM*, Klasifikasi Teks, Pengolahan Bahasa Alami

## 1. PENDAHULUAN

Perkembangan teknologi informasi dan meningkatnya penggunaan media digital menyebabkan pertumbuhan data teks dalam jumlah besar, khususnya dalam bentuk opini, komentar, dan ekspresi perasaan pengguna. Data teks tersebut mengandung informasi emosional yang bernilai penting untuk berbagai aplikasi, seperti analisis opini publik, evaluasi layanan, serta sistem pendukung keputusan. Oleh karena itu, analisis emosi pada teks menjadi salah satu topik penting dalam bidang pengolahan bahasa alami (*Natural Language Processing*).

Klasifikasi emosi pada teks berbahasa Indonesia memiliki tantangan tersendiri, antara lain keterbatasan sumber daya bahasa, variasi penggunaan kata, serta ambiguitas makna yang sering muncul dalam kalimat sehari-hari. Beberapa penelitian sebelumnya telah memanfaatkan model berbasis *deep learning* seperti *Convolutional Neural Network (CNN)* dan *Long Short-Term Memory (LSTM)* untuk klasifikasi teks, serta model *pretrained* seperti BERT yang menunjukkan performa tinggi [1], [2], [3], [8]. Namun, penggunaan model *pretrained* umumnya membutuhkan sumber daya komputasi yang besar dan tidak selalu sesuai untuk lingkungan dengan keterbatasan perangkat keras [14].

Berdasarkan permasalahan tersebut, penelitian ini mengusulkan pendekatan klasifikasi emosi teks berbahasa Indonesia menggunakan model *Bidirectional Long Short-Term Memory (BiLSTM)* tanpa memanfaatkan model *pretrained*. Pendekatan ini dipilih karena *BiLSTM* mampu menangkap konteks urutan kata secara dua arah sehingga lebih efektif dalam memahami makna kalimat, sekaligus tetap efisien secara komputasi [7]. Penelitian ini diharapkan dapat memberikan alternatif solusi yang seimbang antara performa dan efisiensi, serta menjadi referensi dalam pengembangan sistem analisis emosi teks berbahasa Indonesia.

## 2. METODE PENELITIAN

Metode penelitian pada artikel ini disusun untuk menjelaskan tahapan yang dilakukan dalam membangun sistem klasifikasi emosi teks berbahasa Indonesia. Tahapan penelitian meliputi analisis data, perancangan arsitektur model, proses pelatihan, serta evaluasi kinerja model.

### 2.1 Analisis Data

Data yang digunakan dalam penelitian ini berupa dataset teks berbahasa Indonesia yang telah memiliki label emosi. Setiap data terdiri dari teks dan kelas emosi yang merepresentasikan perasaan tertentu, seperti senang, sedih, marah, takut, dan netral. Dataset dianalisis untuk mengetahui distribusi kelas emosi guna mengidentifikasi adanya ketidakseimbangan kelas (*class imbalance*). Untuk menjaga proporsi data pada proses pelatihan dan pengujian, data dibagi menjadi data latih dan data validasi dengan perbandingan 80% dan 20% menggunakan metode *stratified splitting*.

Tabel 1. Contoh Dataset Teks Berlabel Emosi

| Text   | Emosi   |
|--|---------|
| Kamu pasti kuat, semangat ya   | Happy   |
| AkTidak apa-apa, aku terbiasa  | Sad     |
| aku kesel banget sama kamu   | Angry   |
| ujian besok bikin aku deg degan  | Fear    |
| Hidup itu lucu, kadang orang yang kita lindungi adalah pemegang pistolnya. | Neutral |

## 2.2 Praproses dan Vektorisasi Teks

Sebelum digunakan dalam proses pelatihan, data teks dipersiapkan melalui tahap praproses untuk memastikan keseragaman dan kualitas data. Tahap praproses dilakukan karena data teks hasil pengambilan dari sumber daring umumnya memiliki variasi penulisan, perbedaan penggunaan huruf kapital, serta spasi berlebih yang dapat memengaruhi kinerja model klasifikasi emosi. Oleh karena itu, praproses teks diperlukan untuk mengurangi noise dan meningkatkan konsistensi data sebelum diproses lebih lanjut. Beberapa penelitian juga menggunakan Word2vec dan GloVe [9],[10].

Tahap praproses yang diterapkan meliputi normalisasi huruf dengan mengubah seluruh teks menjadi huruf kecil (lowercasing), penghapusan spasi ganda, serta pemangkasan spasi di awal dan akhir teks. Langkah-langkah ini bertujuan untuk menghasilkan teks yang lebih bersih dan terstruktur, sehingga model dapat lebih fokus mempelajari pola linguistik yang relevan dengan emosi tanpa terganggu oleh perbedaan format penulisan.

Tabel 2. Tahapan Praproses Teks

| Tahap             | Deskripsi                                 |
|-------------------|---|
| Case folding      | Mengubah seluruh teks menjadi huruf kecil |
| Normalisasi spasi | Menghapus spasi berlebih                  |
| Tokenisasi        | Memecah teks menjadi kata                 |
| Stopword removal  | Menghapus kata umum yang tidak informatif |
| Stemming          | Mengubah kata ke bentuk dasar             |

Setelah melalui tahap praproses, data teks direpresentasikan ke dalam bentuk numerik agar dapat diproses oleh model pembelajaran mendalam. Pada penelitian ini, proses vektorisasi dilakukan menggunakan lapisan TextVectorization dari Keras yang berfungsi untuk membangun kosakata berdasarkan data latih serta melakukan tokenisasi terhadap teks. Lapisan ini mengubah setiap kata menjadi indeks numerik sesuai dengan kosakata yang terbentuk, sehingga teks dapat direpresentasikan secara terstruktur dalam bentuk urutan angka.

Selain itu, lapisan TextVectorization juga digunakan untuk menyamakan panjang urutan teks melalui proses padding. Penyeragaman panjang urutan diperlukan karena model LSTM membutuhkan input dengan dimensi yang tetap pada setiap proses pelatihan. Proses adaptasi vektorisasi hanya dilakukan pada data latih untuk mencegah kebocoran data (data leakage), sehingga evaluasi model dapat dilakukan secara objektif dan mencerminkan kemampuan generalisasi model terhadap data yang belum pernah dilihat sebelumnya.

Tabel 3. Vektorisasi Teks

| Parameter         | Nilai             |
|-------------------|-------------------|
| Metode            | TextVectorization |
| Maksimum kosakata | 9.000 token       |
| Panjang urutan    | 30 token          |
| Mode output       | Integer           |
| Data adaptasi     | Data latih saja   |

### 2.3 Arsitektur Model

Model yang diusulkan menggunakan arsitektur *Bidirectional Long Short-Term Memory* (BiLSTM) [6], [7]. Lapisan *embedding* digunakan sebagai tahap awal untuk mempelajari representasi kata secara dense berdasarkan kosakata yang terbentuk. Selanjutnya, dua lapisan BiLSTM diterapkan secara bertingkat. Lapisan BiLSTM pertama berfungsi menangkap konteks urutan kata secara dua arah, sedangkan lapisan kedua digunakan untuk mempelajari pola emosi yang lebih halus. Untuk mengurangi risiko *overfitting*, diterapkan teknik regularisasi berupa *dropout*, *recurrent dropout*, *batch normalization*, serta *L2 regularization*. Lapisan keluaran menggunakan fungsi aktivasi *softmax* untuk menghasilkan probabilitas pada setiap kelas emosi.

Tabel 4. Arsitektur Model

| Layer               | Konfigurasi                 | Keterangan                 |
|---------------------|-----------------------------|----------------------------|
| Input               | Teks (string)               | Input teks mentah          |
| TextVectorization   | max_tokens=9000, seq_len=30 | Tokenisasi & padding       |
| Embedding           | Dimensi 64                  | Representasi kata          |
| BiLSTM 1            | 48 unit, dropout 0.25       | Menangkap konteks dua arah |
| BiLSTM 2            | 24 unit, dropout 0.25       | Representasi sekuens final |
| Dense               | 32 unit, ReLU, L2           | Transformasi fitur         |
| Batch Normalization | –                           | Stabilisasi training       |
| Dropout             | 0.35                        | Reduksi overfitting        |
| Output              | Softmax                     | Probabilitas kelas emosi   |

### 2.4 Proses Pelatihan Model

Proses pelatihan model dilakukan menggunakan algoritma optimasi Adam dengan laju pembelajaran yang disesuaikan untuk menjaga stabilitas pelatihan. Fungsi kerugian yang digunakan adalah *Sparse Categorical Crossentropy* karena label kelas berbentuk numerik [6]. Untuk mengatasi ketidakseimbangan kelas, diterapkan metode *class weighting* sehingga kelas dengan jumlah data lebih sedikit tetap berkontribusi secara proporsional selama proses pelatihan. Selain itu, mekanisme *Early Stopping* digunakan untuk menghentikan pelatihan ketika kinerja model pada data validasi tidak mengalami peningkatan, sehingga dapat mencegah *overfitting* [11],[12].

Tabel 5. Proses Pelatihan model

| Parameter       | Nilai                            |
|-----------------|----------------------------------|
| Optimizer       | Adam                             |
| Learning rate   | 0.0003                           |
| Batch size      | 32                               |
| Epoch maksimum  | 20                               |
| Loss function   | Sparse Categorical Crossentropy  |
| Class weighting | Balanced (maks. 2.0)             |
| Callback        | EarlyStopping, ReduceLROnPlateau |

### 2.5 Evaluasi Model

Evaluasi kinerja model dilakukan untuk menilai kemampuan sistem dalam mengklasifikasikan emosi pada teks berbahasa Indonesia. Proses evaluasi dilakukan menggunakan data validasi yang tidak terlibat dalam proses pelatihan, sehingga hasil evaluasi dapat mencerminkan kemampuan generalisasi model terhadap data baru.

Metrik evaluasi yang digunakan dalam penelitian ini meliputi akurasi, precision, recall, dan F1-score. Akurasi digunakan untuk mengukur tingkat ketepatan prediksi model secara keseluruhan, sedangkan precision dan recall digunakan untuk menilai kinerja model pada masing-masing kelas emosi. F1-score digunakan sebagai ukuran keseimbangan antara precision dan recall, terutama pada kondisi dataset yang tidak seimbang.

Selain itu, evaluasi model juga dilakukan menggunakan confusion matrix untuk menganalisis pola kesalahan klasifikasi antar kelas emosi. Analisis ini digunakan untuk mengetahui kelas emosi yang paling sering tertukar serta mengidentifikasi keterbatasan model dalam membedakan emosi dengan karakteristik linguistik yang serupa.

## 2.6 Rumus Matematika

Model yang digunakan dalam penelitian ini berbasis Long Short-Term Memory (LSTM) yang dirancang untuk memproses data sekuensial. Pada setiap langkah waktu  $t$ , LSTM menggunakan beberapa gerbang untuk mengatur aliran informasi, yaitu *forget gate*, *input gate*, dan *output gate*.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad 1)$$

Persamaan (1) merepresentasikan *forget gate* yang berfungsi untuk menentukan bagian informasi dari *cell state* sebelumnya yang masih relevan dan perlu dipertahankan. Nilai keluaran gerbang ini berada pada rentang 0 hingga 1 melalui fungsi aktivasi sigmoid. Nilai mendekati 0 menunjukkan bahwa informasi lama diabaikan, sedangkan nilai mendekati 1 menunjukkan informasi tersebut dipertahankan.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad 2)$$

$$C_t = \tanh(x_t \cdot W_c + h_{t-1} \cdot U_c + b_c) \quad 3)$$

Persamaan (2) dan (3) menggambarkan *input gate* dan *candidate cell state*. Input gate mengatur seberapa besar informasi baru yang akan dimasukkan ke dalam *cell state*. Sementara itu, *candidate cell state* merepresentasikan informasi baru yang dihasilkan dari kombinasi input saat ini dan *hidden state* sebelumnya menggunakan fungsi aktivasi tanh. Mekanisme ini memungkinkan model memilih informasi baru yang relevan dengan konteks emosi teks.

$$C_t = f_t \times C_{t-1} + i_t \times C_t \quad 4)$$

Persamaan (4) menunjukkan proses pembaruan *cell state* yang merupakan inti dari LSTM. *Cell state* diperbarui dengan mengombinasikan informasi lama yang dipertahankan oleh *forget gate* dan informasi baru yang dikendalikan oleh *input gate*. Mekanisme ini memungkinkan LSTM menyimpan informasi penting dalam jangka panjang, yang sangat berguna untuk memahami hubungan antar kata dalam kalimat.

$$O_t = \sigma(x_t \times W_o + h_{t-1} \times U_o + b_o) \quad 5)$$

$$h_t = O_t \times \tanh(C_t) \quad 6)$$

Persamaan (5) dan (6) menjelaskan *output gate* dan pembentukan *hidden state*. Output gate menentukan bagian *cell state* yang akan dikeluarkan sebagai *hidden state*. *Hidden state* ini merupakan representasi akhir pada setiap langkah waktu dan digunakan sebagai masukan untuk lapisan berikutnya atau untuk proses klasifikasi.

$$h_t^{bi} = [h_t; h_t] \quad 7)$$

Pada arsitektur Bidirectional LSTM, sebagaimana ditunjukkan pada Persamaan (7), pemrosesan sekuens dilakukan dari dua arah, yaitu maju dan mundur. *Hidden state* dari kedua arah tersebut digabungkan untuk menghasilkan representasi yang lebih kaya. Pendekatan ini memungkinkan model memahami konteks kata tidak hanya dari kata sebelumnya, tetapi juga dari kata setelahnya, sehingga lebih efektif dalam menangkap makna emosi dalam teks.

$$P(y_i) = \frac{e^{z_i}}{\sum_{j=1}^C e^{z_j}} \quad 8)$$

Persamaan (8) merupakan fungsi *softmax* yang digunakan pada lapisan keluaran untuk melakukan klasifikasi emosi. Fungsi ini mengubah skor keluaran model menjadi probabilitas pada setiap kelas emosi. Kelas dengan probabilitas tertinggi dipilih sebagai hasil prediksi emosi teks.

## 2.7 Pengacuan Pustaka

Pengacuan pustaka dalam penelitian ini menggunakan gaya sitasi IEEE dengan penomoran berurutan sesuai urutan kemunculan sitasi pada naskah. Setiap sumber dirujuk menggunakan angka Arab di dalam tanda kurung siku, seperti [1], [2], dan seterusnya, untuk menjaga konsistensi dan kemudahan penelusuran referensi.

Daftar pustaka disusun berdasarkan urutan kemunculan sitasi dan hanya memuat sumber yang benar-benar dirujuk dalam penelitian. Seluruh referensi yang digunakan berkaitan dengan klasifikasi emosi teks, pengolahan bahasa alami, serta penerapan metode Long Short-Term Memory (LSTM) dan Bidirectional LSTM (BiLSTM).

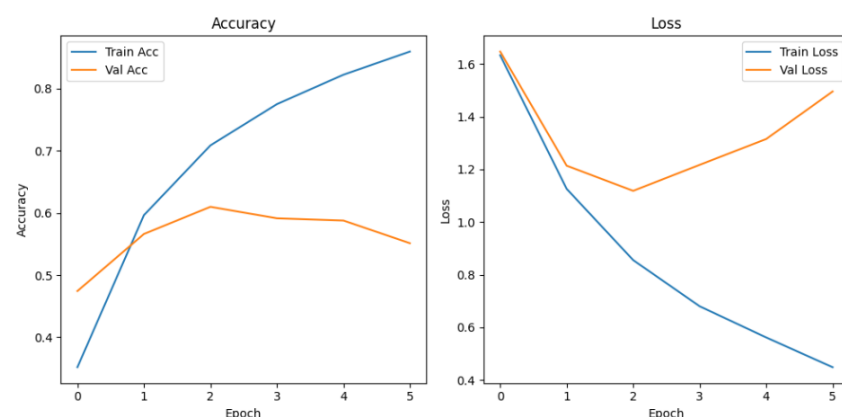
## 2 HASIL DAN PEMBAHASAN

Bagian ini membahas hasil eksperimen yang diperoleh dari penerapan model BiLSTM pada klasifikasi emosi teks berbahasa Indonesia serta analisis terhadap performa model yang dihasilkan. Evaluasi dilakukan menggunakan metrik akurasi, grafik pelatihan, dan *confusion matrix* untuk memberikan gambaran menyeluruh terhadap kinerja sistem.

Grafik fungsi kerugian (*loss*) memperlihatkan penurunan nilai *training loss* secara bertahap, sementara *validation loss* mencapai titik minimum pada epoch tertentu sebelum mengalami peningkatan. Kondisi ini mengindikasikan adanya kecenderungan *overfitting* ringan apabila pelatihan dilanjutkan terlalu lama. Oleh karena itu, penerapan mekanisme *early stopping* terbukti efektif dalam menjaga keseimbangan antara performa dan generalisasi model.

### 3.2 Evaluasi Akurasi

Berdasarkan hasil pengujian, model BiLSTM yang diusulkan mampu mencapai akurasi validasi yang stabil dan lebih tinggi dibandingkan konfigurasi awal tanpa optimalisasi. Peningkatan akurasi ini dipengaruhi oleh penggunaan arsitektur BiLSTM bertingkat yang mampu menangkap konteks kalimat secara dua arah, serta penerapan teknik regularisasi dan *class weighting* yang membantu mengatasi ketidakseimbangan kelas. Meskipun akurasi data latih lebih tinggi dibandingkan data validasi, perbedaan tersebut masih berada dalam batas yang dapat diterima, sehingga model tidak mengalami *overfitting* secara signifikan.



Grafik yang ditampilkan menunjukkan kinerja model selama proses pelatihan berdasarkan beberapa istilah utama, yaitu *epoch*, *accuracy*, dan *loss*, yang masing-masing merepresentasikan aspek berbeda dari proses pembelajaran model. Epoch merujuk pada satu siklus penuh ketika seluruh data latih diproses sekali oleh model selama pelatihan. Peningkatan jumlah epoch memungkinkan model memperbarui bobotnya secara bertahap untuk meminimalkan kesalahan prediksi, namun jumlah epoch yang berlebihan dapat menyebabkan model mengalami *overfitting*.

Istilah *accuracy* menggambarkan tingkat ketepatan prediksi model terhadap data yang diuji, yang dihitung sebagai perbandingan antara jumlah prediksi yang benar dengan total keseluruhan data. *Training accuracy* menunjukkan tingkat akurasi model pada data latih, sedangkan *validation accuracy* mengukur kemampuan model dalam melakukan prediksi pada data validasi yang tidak terlibat dalam proses pelatihan. Perbedaan nilai antara

kedua metrik ini digunakan untuk menilai kemampuan generalisasi model; selisih yang semakin besar mengindikasikan bahwa model cenderung terlalu menyesuaikan diri dengan data latih.

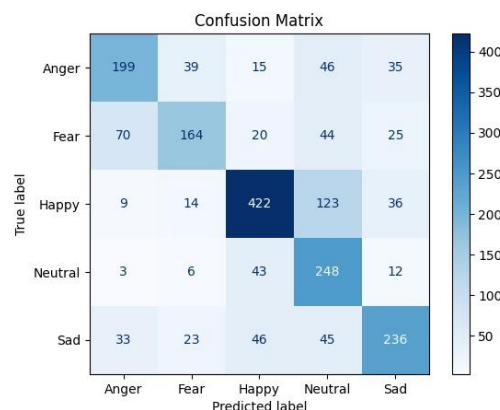
Sementara itu, loss merepresentasikan nilai kesalahan prediksi model yang dihitung menggunakan fungsi kerugian (*loss function*), dalam penelitian ini menggunakan *Sparse Categorical Crossentropy*. Training loss menunjukkan besarnya kesalahan model pada data latih, sedangkan validation loss menggambarkan kesalahan prediksi pada data validasi. Nilai loss yang semakin kecil menandakan bahwa prediksi model semakin mendekati label sebenarnya. Peningkatan validation loss setelah mencapai titik minimum merupakan indikasi bahwa model mulai kehilangan kemampuan generalisasi.

Dengan demikian, kombinasi antara training accuracy, validation accuracy, training loss, dan validation loss digunakan sebagai indikator utama untuk mengevaluasi keseimbangan antara kemampuan pembelajaran dan generalisasi model. Analisis terhadap istilah-istilah ini memungkinkan peneliti mengidentifikasi titik pelatihan optimal serta mendeteksi gejala overfitting secara lebih akurat, sehingga keputusan penghentian pelatihan melalui mekanisme *early stopping* dapat dilakukan secara objektif dan terukur.

### 3.3 Analisis Confusion Matrix

Hasil *confusion matrix* menunjukkan bahwa model memiliki performa yang baik dalam mengklasifikasikan kelas emosi dengan ciri leksikal yang kuat, seperti emosi senang dan cinta. Pada kelas-kelas tersebut, sebagian besar data berhasil diklasifikasikan dengan benar. Sebaliknya, kesalahan klasifikasi lebih sering terjadi pada emosi yang memiliki kedekatan semantik, seperti marah, sedih, dan takut. Hal ini disebabkan oleh kemiripan konteks dan penggunaan kata yang bersifat ambigu dalam teks berbahasa Indonesia.

Selain itu, kelas netral juga menunjukkan beberapa kesalahan klasifikasi ke dalam emosi positif maupun negatif, yang mencerminkan sulitnya membedakan teks netral dari emosi dengan intensitas rendah. Pola kesalahan ini menunjukkan bahwa keterbatasan model bukan semata-mata disebabkan oleh arsitektur, melainkan juga oleh karakteristik linguistik data.



Confusion matrix digunakan untuk menganalisis kinerja model BiLSTM dalam mengklasifikasikan emosi teks berbahasa Indonesia secara lebih rinci pada setiap kelas emosi. Berdasarkan hasil yang ditampilkan, model menunjukkan performa yang cukup baik pada beberapa kelas dengan ciri leksikal yang kuat. Kelas **Happy** memiliki jumlah prediksi benar tertinggi, dengan 422 data berhasil diklasifikasikan secara tepat, menunjukkan bahwa model mampu mengenali ekspresi emosi positif yang eksplisit dengan tingkat akurasi yang tinggi. Hal serupa juga terlihat pada kelas **Neutral** dan **Sad**, yang masing-masing memiliki 248 dan 236 prediksi benar, menandakan bahwa model cukup stabil dalam membedakan teks netral dan emosi negatif yang bersifat eksplisit.

Meskipun demikian, confusion matrix juga memperlihatkan adanya kesalahan klasifikasi yang cukup signifikan pada kelas-kelas emosi yang memiliki kedekatan semantik. Kelas Anger dan Fear menunjukkan tingkat kesalahan yang relatif lebih tinggi, di mana sejumlah data berlabel *Anger* salah diklasifikasikan sebagai *Neutral* dan *Fear*, serta sebaliknya. Hal ini mengindikasikan bahwa model masih mengalami kesulitan dalam membedakan emosi negatif yang memiliki intensitas dan konteks linguistik yang mirip, terutama ketika ekspresi emosi tidak dinyatakan secara eksplisit melalui kata-kata emosional yang kuat.

Selain itu, kelas Happy juga menunjukkan pola kesalahan yang cukup menonjol, khususnya prediksi sebagai *Neutral*. Fenomena ini mencerminkan bahwa sebagian teks beremosi positif memiliki ekspresi yang bersifat implisit atau bernuansa ringan, sehingga sulit dibedakan dari teks netral. Pola serupa juga ditemukan pada kelas

Sad, yang cukup sering tertukar dengan *Neutral* dan *Happy*, menunjukkan adanya ambiguitas makna dalam ekspresi emosi berbahasa Indonesia yang kontekstual.

Secara keseluruhan, analisis confusion matrix menunjukkan bahwa kesalahan klasifikasi yang terjadi bukan semata-mata disebabkan oleh keterbatasan arsitektur model, melainkan juga oleh karakteristik linguistik data yang memiliki kedekatan makna antar kelas emosi. Meskipun demikian, model BiLSTM yang diusulkan tetap menunjukkan kemampuan klasifikasi yang baik secara umum, terutama pada kelas emosi dengan ciri leksikal yang jelas. Temuan ini menegaskan bahwa pendekatan BiLSTM tanpa pretrained embedding mampu memberikan performa yang kompetitif, namun masih memiliki ruang pengembangan untuk meningkatkan pemisahan emosi yang bersifat halus dan kontekstual.

### 3.4 Pembahasan

Secara keseluruhan, hasil penelitian menunjukkan bahwa pendekatan BiLSTM tanpa model *pretrained* tetap mampu memberikan performa yang kompetitif dalam klasifikasi emosi teks berbahasa Indonesia. Optimalisasi arsitektur dan strategi pelatihan berperan penting dalam meningkatkan kinerja model, meskipun masih terdapat keterbatasan dalam membedakan emosi yang memiliki kemiripan makna. Temuan ini menegaskan bahwa BiLSTM dapat menjadi alternatif yang efisien dan efektif pada lingkungan dengan keterbatasan sumber daya komputasi.

## 3 SIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa model *Bidirectional Long Short-Term Memory* (BiLSTM) tanpa memanfaatkan model *pretrained* mampu digunakan secara efektif untuk klasifikasi emosi pada teks berbahasa Indonesia. Penerapan arsitektur BiLSTM bertingkat yang dikombinasikan dengan lapisan *embedding*, teknik regularisasi, serta *class weighting* terbukti dapat meningkatkan performa model dan menjaga stabilitas proses pelatihan. Hasil evaluasi menunjukkan bahwa model mampu mengklasifikasikan emosi dengan baik, khususnya pada kelas emosi yang memiliki ciri leksikal yang jelas, sementara kesalahan klasifikasi umumnya terjadi pada emosi yang memiliki kedekatan semantik. Dengan demikian, penelitian ini menunjukkan bahwa pendekatan BiLSTM non-*pretrained* dapat menjadi solusi yang efisien dan layak untuk analisis emosi teks, terutama pada lingkungan dengan keterbatasan sumber daya komputasi.

## 4 SARAN

Penelitian ini masih memiliki beberapa keterbatasan yang dapat menjadi peluang pengembangan pada penelitian selanjutnya. Pertama, keterbatasan jumlah dan keragaman data menyebabkan model masih mengalami kesulitan dalam membedakan emosi yang memiliki kedekatan semantik, seperti marah, sedih, dan takut. Oleh karena itu, penelitian lanjutan disarankan menggunakan dataset yang lebih besar dan lebih representatif agar model dapat mempelajari variasi ekspresi emosi yang lebih luas.

Kedua, penelitian ini menggunakan pendekatan *embedding* yang dilatih dari awal tanpa memanfaatkan informasi linguistik tambahan. Penelitian selanjutnya dapat mengeksplorasi integrasi fitur linguistik, seperti *part-of-speech tagging* atau informasi sintaksis, untuk membantu model memahami struktur kalimat secara lebih mendalam. Selain itu, pengembangan arsitektur BiLSTM dengan penambahan mekanisme perhatian (*attention mechanism*) juga dapat dipertimbangkan untuk meningkatkan kemampuan model dalam menangkap kata atau frasa yang paling berpengaruh terhadap emosi.

Terakhir, evaluasi pada penelitian ini masih berfokus pada metrik akurasi dan *confusion matrix*. Penelitian mendatang disarankan untuk menambahkan metrik evaluasi lain, seperti *precision*, *recall*, dan *F1-score*, guna memberikan analisis performa yang lebih komprehensif pada setiap kelas emosi.

## DAFTAR PUSTAKA

- [1] A. Glenn and P. LaCasse, "Emotion classification of Indonesian tweets using bidirectional LSTM," *Neural Computing and Applications*, 2023. [Online]. Available: <https://www.semanticscholar.org/paper/Emotion-classification-of-Indonesian-Tweets-using-Glenn-LaCasse/192a1a6f72d935b39bfe8fff809028bab6a4615>

- 
- [2] S. William, K. —, and A. Chowanda, “Emotion recognition Indonesian language from Twitter using IndoBERT and Bi-LSTM,” *Communications in Mathematical Biology and Neuroscience*, 2024. [Online]. Available: <https://scik.org/index.php/cmbn/article/view/7858>
- [3] P. Utami, M. R. Ningsih, D. A. A. Pertiwi, and J. Unjung, “Sentiment-based emotion classification using bidirectional long short-term memory (Bi-LSTM),” *Journal of Soft Computing Explorations*, vol. 5, no. 3, pp. 281–289, 2024. [Online]. Available: <https://shmpublisher.com/index.php/josceex/article/view/461>.
- [4] X. Wang, Y. Liu, C. Sun, B. Wang, and X. Wang, “Predicting sentiment polarity of Chinese microblogs using LSTM,” *Information Processing & Management*, vol. 52, no. 5, pp. 878–887, 2016.
- [5] “Analisis sentimen terhadap berita using LSTM,” JATI: Journal of Applied Informatics Techniques, 2024. [Online]. Available: <https://ejournal.itn.ac.id/jati/article/download/13194/7356/>
- [6] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [7] A. Graves and J. Schmidhuber, “Framewise phoneme classification with bidirectional LSTM networks,” *Neural Networks*, vol. 18, no. 5–6, pp. 602–610, 2005.
- [8] Y. Kim, “Convolutional neural networks for sentence classification,” *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1746–1751, 2014.
- [9] J. Pennington, R. Socher, and C. D. Manning, “GloVe: Global vectors for word representation,” *Proceedings of EMNLP*, pp. 1532–1543, 2014.
- [10] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” *arXiv preprint arXiv:1301.3781*, 2013.
- [11] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, Cambridge, MA, USA: MIT Press, 2016.
- [12] F. Chollet, *Deep Learning with Python*, Shelter Island, NY, USA: Manning Publications, 2018.
- [13] N. Liu, B. Gao, and Y. Liu, “Emotion classification based on LSTM neural network,” *Proceedings of the International Conference on Artificial Intelligence and Big Data*, 2017.
- [14] A. Devlin, M. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” *Proceedings of NAACL-HLT*, pp. 4171–4186, 2019.
- [15] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*, Cambridge, UK: Cambridge University Press, 2008.