

# Analisis Sentimen Pengguna X/Twitter Terhadap Timnas Sepakbola Indonesia di Era Shin Tae Yong dengan BERT&RNN

Jahfal Azzuhri Subroto<sup>1</sup>, Ricky Eka Putra<sup>2</sup>

<sup>1,2</sup> Program Studi Teknik Informatika, Fakultas Teknik, Universitas Negeri Surabaya

[1jahfal.21041@mhs.unesa.ac.id](mailto:jahfal.21041@mhs.unesa.ac.id)

[2rickyeka@unesa.ac.id](mailto:rickyeka@unesa.ac.id)

**Abstrak**— Perkembangan media sosial, khususnya Twitter (X), menjadikan platform ini sebagai ruang utama bagi masyarakat untuk mengekspresikan opini terhadap performa Timnas Sepakbola Indonesia di bawah kepelatihan Shin Tae Yong (STY). Penelitian ini bertujuan untuk menganalisis dan membandingkan performa empat model analisis sentimen berbasis kombinasi IndoBERT dan arsitektur Recurrent Neural Network (RNN), yaitu IndoBERT + RNN, IndoBERT + LSTM, IndoBERT + BiLSTM, dan IndoBERT + GRU. Data dikumpulkan menggunakan *Tweet Harvest* dan melalui tahapan *preprocessing* meliputi *cleaning*, *case folding*, *normalisasi*, *tokenizing*, *stopword removal*, *stemming*, serta *labeling* menggunakan InSet Lexicon yang kemudian divalidasi secara manual. Setiap model dilatih menggunakan beberapa konfigurasi *hyperparameter*, seperti variasi *hidden size*, *batch size*, *dropout*, *learning rate*, serta jumlah unit RNN untuk menemukan performa optimal pada tahap pelatihan dan validasi. Konfigurasi terbaik dari masing-masing model kemudian digunakan sebagai model final untuk dievaluasi pada skenario tiga label (positif, netral, negatif) dan dua label (positif, negatif). Evaluasi dilakukan menggunakan *classification report* dan *confusion matrix*. Hasil penelitian menunjukkan bahwa pada skenario tiga label, model IndoBERT+RNN memberikan performa terbaik dengan akurasi 0,69 dan Macro F1-Score 0,68. Sementara itu, pada skenario dua label, model IndoBERT+GRU menghasilkan performa tertinggi dengan akurasi dan Macro F1-Score sebesar 0,83. Temuan ini menegaskan bahwa pemilihan konfigurasi optimal pada kombinasi model berbasis transformer dan jaringan berulang berpengaruh signifikan terhadap peningkatan akurasi analisis sentimen berbahasa Indonesia.

**Kata Kunci**— Analisis sentimen, IndoBERT, RNN, Twitter, Timnas Indonesia

## I. PENDAHULUAN

Sepak bola merupakan olahraga dengan basis penggemar terbesar di Indonesia, di mana lebih dari 69% masyarakat mengikutinya secara aktif [1]. Dalam beberapa tahun terakhir, perhatian publik terhadap Timnas Indonesia meningkat seiring hadirnya Shin Tae Yong (STY) sebagai pelatih sejak akhir 2019. Di bawah kepemimpinannya, Timnas Indonesia mencatat sejumlah pencapaian penting, seperti lolos ke Piala Asia 2023 setelah 47 tahun, mencapai final Piala AFF 2020, serta memperoleh medali perak SEA Games 2021 [2]. Meskipun demikian, berbagai keputusan strategis STY terutama terkait seleksi pemain, rotasi, dan gaya bermain menjadi topik yang memicu beragam opini publik.

Media sosial, khususnya Twitter (X), telah berkembang menjadi ruang utama bagi masyarakat untuk mengekspresikan pandangan mereka secara cepat dan terbuka. Twitter

menghasilkan data teks dalam jumlah besar yang berpotensi memberikan gambaran objektif mengenai dinamika opini publik terhadap performa Timnas Indonesia [3]. Namun demikian, analisis sentimen pada teks berbahasa Indonesia menghadapi tantangan linguistik, seperti penggunaan bahasa informal, slang, singkatan, hingga struktur kalimat yang tidak baku.

Model berbasis transformer seperti Bidirectional Encoder Representations from Transformers (BERT) telah terbukti unggul dalam memahami makna kata secara kontekstual dua arah [4]. Untuk bahasa Indonesia, IndoBERT dikembangkan menggunakan korpus besar berbahasa Indonesia sehingga lebih efektif dalam memodelkan karakteristik linguistik lokal [5]. Di sisi lain, model Recurrent Neural Network (RNN) dan variannya seperti Long Short-Term Memory (LSTM), Bidirectional LSTM (BiLSTM), serta Gated Recurrent Unit (GRU) menunjukkan performa baik dalam menangkap hubungan berurutan pada data teks [6].

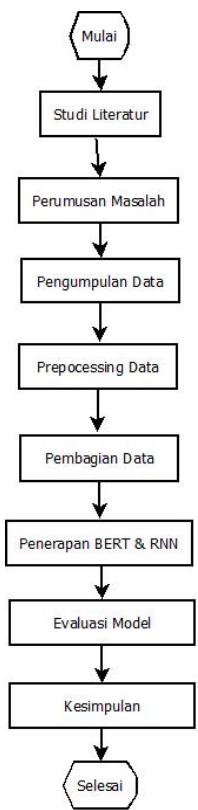
Penelitian-penelitian sebelumnya juga menunjukkan bahwa kombinasi model transformer dengan arsitektur RNN mampu meningkatkan akurasi analisis sentimen dibandingkan penggunaan model tunggal. Penelitian yang dilakukan Bello membandingkan BERT dengan CNN, RNN, dan BiLSTM, dan menemukan bahwa kombinasi BERT dengan RNN dan BiLSTM memberikan performa terbaik dengan akurasi hingga 93% dan F1-score 95% [6]. Penelitian yang dilakukan Topbas dkk juga menunjukkan bahwa model hybrid BERT–RNN mampu meningkatkan akurasi dibandingkan model konvensional pada dataset Twitter terkait COVID-19, dengan BERT menangkap konteks kata secara bidirectional dan RNN memodelkan dependensi sekuensial [7]. Temuan serupa ditunjukkan oleh Md. Jahidul Islam dkk yang melaporkan bahwa kombinasi BERT dengan RNN dan LSTM menghasilkan prediksi sentimen yang lebih akurat pada ulasan aplikasi Zoom dibandingkan model individu [8].

Berdasarkan penelitian terdahulu hasil penelitian menunjukkan bahwa model *hybrid* berbasis BERT dan arsitektur RNN mampu menangkap konteks kata secara mendalam sekaligus memahami struktur sekuensial dalam teks. Penulis tertarik untuk melakukan analisis sentimen pada media social X/twitter terkait timnas sepakbola Indonesia di era Shin Tae Yong. Tujuan dari penelitian ini adalah untuk mengetahui nilai performa yang dihasilkan dari kombinasi IndoBERT dengan RNN, LSTM, BiLSTM, atau GRU untuk mengidentifikasi persepsi publik terhadap Timnas Indonesia di era kepelatihan Shin Tae Yong dengan judul penelitian

“Analisis Sentimen Pengguna X/Twitter terhadap Timnas Sepakbola Indonesia di Era Shin Tae Yong dengan BERT & RNN”.

## II. METODE PENELITIAN

### A. Alur Penelitian



Gbr. 1 Alur Penelitian

Penelitian ini menggunakan pendekatan sistematis untuk menganalisis sentimen pengguna Twitter terhadap Timnas Indonesia di era kepelatihan Shin Tae Yong dengan memanfaatkan kombinasi IndoBERT dan RNN. Alur lengkap metode penelitian ditunjukkan pada Gbr. 1.

### B. Studi Literatur

Peneliti melakukan studi literatur dengan meninjau jurnal, prosiding, dan artikel akademik terkait analisis sentimen berbasis teks. Hasil kajian menunjukkan bahwa BERT memiliki kemampuan unggul dalam memahami konteks secara mendalam, dan beberapa penelitian menemukan bahwa kombinasi BERT dengan RNN, LSTM, atau GRU dapat meningkatkan akurasi klasifikasi sentimen. Studi sebelumnya juga mencatat adanya variasi opini pengguna Twitter terhadap performa Timnas Indonesia di era Shin Tae Yong, sehingga data Twitter berpotensi besar untuk dianalisis lebih lanjut. Temuan ini menjadi dasar bagi pengembangan model analisis sentimen berbasis kombinasi BERT dan RNN pada penelitian ini.

### C. Perumusan Masalah

Permasalahan dalam penelitian ini dirumuskan berdasarkan hasil studi literatur, yaitu belum optimalnya analisis terhadap opini masyarakat Indonesia mengenai Timnas Sepak Bola Indonesia di era kepelatihan Shin Tae Yong yang disampaikan melalui Twitter. Opini tersebut mencakup sentimen positif, negatif, dan netral yang mencerminkan persepsi publik. Penelitian ini memfokuskan perumusan masalah pada penerapan kombinasi BERT dan RNN untuk menangkap konteks teks secara mendalam sekaligus pola urutan sentimen. Pendekatan ini dipilih karena terbukti lebih akurat dibandingkan penggunaan algoritma secara terpisah. Dengan demikian, penelitian ini diharapkan mampu mengklasifikasikan sentimen publik dengan lebih baik dan memberikan gambaran yang lebih komprehensif mengenai persepsi masyarakat terhadap Timnas Indonesia selama era Shin Tae Yong.

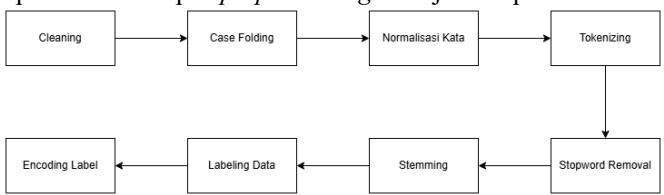
### D. Pengumpulan Data

Data penelitian dikumpulkan menggunakan tools *Tweet Harvest* dengan kata kunci “timnas sty” untuk memperoleh tweet yang membahas persepsi publik terhadap Timnas Indonesia di era kepelatihan Shin Tae Yong. *Tweet Harvest* bekerja melalui mekanisme *crawling* berbasis Playwright dan menghasilkan data dalam format CSV yang mencakup teks tweet, tanggal, jumlah likes, retweets, serta metadata lain yang tersedia. Pengumpulan data dilakukan setiap bulan mulai dari Desember 2019 hingga Januari 2025 untuk menangkap dinamika opini publik sepanjang periode kepelatihan STY. Seluruh data yang diperoleh melalui proses ini kemudian melalui tahap pengecekan duplikasi untuk menjaga kualitas dataset.

*Tweet Harvest* juga telah digunakan secara efektif dalam berbagai penelitian terdahulu, seperti analisis sentimen Pemilu 2024 [9], studi topik Artificial Intelligence [10], serta penelitian mengenai digitalisasi pendidikan [11]. Temuan tersebut memperkuat bahwa *Tweet Harvest* merupakan alat yang andal untuk mengumpulkan data media sosial dalam penelitian analisis sentimen.

### E. Preprocessing Data

Proses *preprocessing* diperlukan agar dataset memiliki kualitas yang baik dan siap digunakan dalam pemodelan. *Preprocessing* digunakan untuk dua tujuan yaitu teks hingga tahap *stemming* dipakai untuk *labeling* dengan InSet, sedangkan untuk model IndoBERT dan RNN hanya digunakan hingga normalisasi karena tokenizer IndoBERT sudah menangani pemecahan subword sehingga *stemming* tidak diperlukan. Tahapan *preprocessing* ditunjukkan pada Gbr. 2.



Gbr. 2 Alur Preprocessing Data

#### 1. Cleaning

Tahap *cleaning* dilakukan untuk menghapus elemen tidak relevan seperti URL, mention, hashtag yang tidak diperlukan, emoji, angka, simbol, dan tanda baca.

#### 2. Case Folding

Semua huruf diubah menjadi huruf kecil untuk menjaga konsistensi dan menghindari perbedaan makna akibat kapitalisasi sehingga model dapat menganggap kata yang sama sebagai entitas yang sama.

#### 3. Normalisasi Kata

Normalisasi dilakukan menggunakan colloquial-indonesian-lexicon untuk mengganti kata tidak baku menjadi kata baku.

#### 4. Tokenizing

*Tokenizing* memecah teks menjadi unit-unit kata (token) sehingga model dapat menganalisis struktur kalimat secara lebih terperinci.

#### 5. Stopword Removal

Stopword umum seperti “di”, “ke”, “yang”, dan kata-kata lain yang tidak memiliki kontribusi signifikan dihapus.

#### 6. Stemming

*Stemming* dilakukan menggunakan library *Sastrawi* untuk mengubah kata ke bentuk dasar dengan menghilangkan imbuhan.

#### 7. Labeling Data

Pelabelan dilakukan menggunakan pendekatan berbasis leksikon dengan Indonesian Sentiment Lexicon (InSet) sebagai acuan dalam menentukan polaritas sentimen. Setiap kata diberi bobot sesuai nilai pada kamus InSet, kemudian bobot tersebut dijumlahkan untuk menghasilkan skor total. Skor positif diklasifikasikan sebagai sentimen positif, skor negatif sebagai negatif, dan skor nol sebagai netral. Hasil pelabelan otomatis ini selanjutnya divalidasi secara manual untuk memastikan kesesuaian label dengan konteks teks.

#### 8. Encoding Label

Setelah proses *preprocessing* dan pelabelan selesai, label sentimen dikonversi ke format numerik agar dapat diolah oleh model pembelajaran mesin. Skema *encoding* yang digunakan adalah: 0 untuk negatif, 1 untuk netral, dan 2 untuk positif.

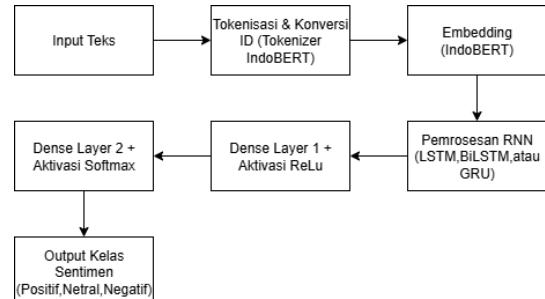
### F. Pembagian Data

Dataset dibagi menjadi tiga subset menggunakan metode random splitting dengan stratifikasi agar distribusi label tetap seimbang. Training set digunakan untuk melatih model, validation set untuk memantau performa selama pelatihan dan mencegah overfitting, sedangkan test set digunakan untuk mengevaluasi performa akhir model pada data yang belum pernah dilihat.

### G. Penerapan BERT & RNN

Penerapan model dilakukan melalui tiga tahap utama, yaitu perancangan model, pelatihan, dan validasi.

#### 1. Desain Model



Gbr. 3 Desain Model

Alur lengkap desain model ditunjukkan pada Gbr. 3. Desain model pada penelitian ini menggabungkan IndoBERT sebagai pembentuk *embedding* kontekstual dan arsitektur RNN (LSTM, BiLSTM, atau GRU) sebagai pemroses urutan kata. IndoBERT menghasilkan *embedding* berdimensi 768 yang mewakili makna tiap token dalam konteks kalimat. *Embedding* tersebut kemudian diteruskan ke model RNN untuk menangkap hubungan antar kata dalam urutan teks. Output dari RNN berupa representasi fitur kemudian diproses oleh Fully Connected Layer dengan aktivasi ReLU, dan lapisan akhir menggunakan Softmax untuk menghasilkan probabilitas tiga kelas sentimen, yaitu negatif, netral, dan positif.

#### 2. Training

Pada tahap training, model dilatih dengan melakukan *fine-tuning* IndoBERT secara menyeluruh bersama arsitektur RNN (LSTM, BiLSTM, atau GRU). *Embedding* yang dihasilkan IndoBERT kemudian diproses oleh RNN untuk memahami urutan kata dalam teks, dan hasilnya diteruskan ke Fully Connected Layer untuk menentukan kelas sentimen. Proses pelatihan menggunakan *Cross-Entropy Loss* dan *optimizer* AdamW. Training berlangsung dalam beberapa *epoch* dengan bantuan *early stopping* agar model tidak overfitting. Setelah proses pelatihan selesai, performa model dievaluasi menggunakan data validasi.

#### 3. Validasi

Validasi dilakukan untuk menilai kemampuan model dalam memprediksi sentimen pada data yang belum pernah dilihat sebelumnya. Nilai loss dan akurasi dari validation set digunakan untuk memonitor overfitting dan menyesuaikan hyperparameter jika diperlukan. Tahap ini membantu memilih konfigurasi terbaik sebelum model diuji pada data uji.

### H. Skenario Ujicoba

Skenario uji coba merangkum parameter yang digunakan dalam training dan validasi model. Pembagian data dilakukan dalam dua variasi, penggunaan IndoBERT dikombinasikan

dengan beberapa arsitektur RNN, serta pengujian berbagai konfigurasi *hidden units*, *dropout*, dan jumlah layer. Pada tahap training, diuji kombinasi *batch size*, *epoch*, *learning rate*, serta penggunaan *Cross-Entropy Loss* dan *optimizer* AdamW. Skenario ini disusun untuk menemukan konfigurasi yang menghasilkan performa terbaik, yang ditunjukkan pada Tabel 1.

TABEL I  
HYPERPARAMETER TUNING

Tahap	Parameter	Nilai
Pembagian Data	Data training, data validasi, data testing	70, 15, dan 15 80, 10, dan 10
BERT	Model	IndoBERT(IndoBERT-base-p2)
RNN	Arsitektur	RNN,LSTM, BiLSTM, GRU
	Hidden Units	128, 256
	Dropout	0.3, 0.5
	Jumlah layer	1, 2
Training Model	Loss Function	Cross-Entropy Loss
	Optimizer	AdamW
	Batch Size	16, 32
	Epoch	2 sd 20 (early stopping)
	Learning Rate	2e-5, 3e-5

### I. Evaluasi Model

Setelah model selesai dilatih, performanya diuji menggunakan data testing yang diperoleh melalui random *stratified splitting*. Evaluasi dilakukan menggunakan *confusion matrix* dan *classification report*. *Confusion matrix* menunjukkan jumlah prediksi benar dan salah pada setiap kelas sentiment positif, netral, dan negatif sehingga dapat terlihat jenis kesalahan yang dilakukan model. Sementara itu, *classification report* memberikan nilai akurasi, precision, recall, dan F1-score sebagai metrik utama untuk menilai kualitas model. Setelah seluruh metrik diperoleh, dilakukan perbandingan performa empat arsitektur model IndoBERT + RNN, IndoBERT + LSTM, IndoBERT + BiLSTM, dan IndoBERT + GRU untuk menentukan kombinasi yang paling optimal dalam menganalisis sentimen pengguna Twitter mengenai Timnas Indonesia di era Shin Tae Yong. Tahap ini menjadi dasar

untuk menarik kesimpulan akhir mengenai efektivitas masing-masing model.

## III. HASIL DAN PEMBAHASAN

### A. Pengumpulan Data

Pengumpulan data dilakukan menggunakan tools *Tweet Harvest* dengan kata kunci utama “timnas sty” untuk memperoleh tweet yang berkaitan dengan opini publik terhadap Timnas Indonesia pada era pelatih Shin Tae Yong. Proses *crawling* dilakukan setiap bulan, dimulai dari Desember 2019 hingga Januari 2025.

```
# Crawl Data

filename = 'timnassty62(1).csv'
search_keyword = 'timnas sty since:2025-01-1 until:2025-01-15 lang:id'
limit = 1000

!npx -y tweet-harvest@2.6.1 -o "{filename}" -s "{search_keyword}" --tab "LATEST" -l {limit} --token {twitter_auth_token}
```

Gbr. 4 Pengumpulan Data Tweet Dengan Tweet Harvest

Pada Gbr 4 menunjukkan bagaimana *Tweet Harvest* digunakan untuk mengambil tweet berdasarkan kata kunci yang telah ditentukan. Tools ini digunakan untuk mengumpulkan tweet per bulan dan menyimpannya dalam bentuk dataset awal sebelum diproses lebih lanjut.

TABEL II  
CONTOH HASIL CRAWLING

No	Tweet
1	@Bali_Football Melihat gaya mai STY di timnas dan stok striker yg ada spaso masih layak di
2	@Krnwn1712 @UpdateBolabola Makanya pada ngebet juara AFF krn udah tw kapasitasnya klo di luar AFF kayak apa... Tar semisal STY udah gak di timnas lagi... Udah kebayang kan hasilnya gimana....
3	DI BAWAH ASUHAN STY #timnas #indonesia #oktober #2023 #bersamagaruda #ke... <a href="https://t.co/qz7dehboLa">https://t.co/qz7dehboLa</a> via @YouTube
4	Lima bintang Timnas Indonesia cedera jelang putaran kedua Kualifikasi Piala Dunia 2026 zona Asia <a href="https://t.co/vSDhSRPMqN">https://t.co/vSDhSRPMqN</a>

Setelah proses *crawling* selesai, langkah berikutnya adalah membaca dan menggabungkan seluruh data hasil scraping bulanan. Berdasarkan proses ini, diperoleh 30.258 tweet, yang kemudian disimpan dalam format CSV (Comma Separated Values). Tabel II menunjukkan contoh hasil data *crawling* yang telah berhasil dikumpulkan.

### B. Preprocessing Data

Setelah data dikumpulkan dan disimpan dalam format CSV, dilakukan beberapa tahapan *preprocessing* untuk menyiapkan teks sebelum masuk ke tahap selanjutnya.

#### 1. Cleaning

Pada tahap *cleaning*, teks dibersihkan dari elemen-elemen yang tidak diperlukan seperti username, emoji,

angka, simbol, serta URL. Sebelum proses *cleaning* dilakukan penghapusan data duplikat. Setelah dilakukan proses penghapusan data duplikat, jumlah data menjadi 30216 data.

```
# Fungsi untuk menghapus emoji
def remove_emoji(tweet):
    if tweet is not None and isinstance(tweet, str):
        emoji_pattern = re.compile("["
            u"\U0001F600-\U0001F64F" # emoticons
            u"\U0001F300-\U0001F5FF" # symbols & pictographs
            u"\U0001F680-\U0001F6FF" # transport & map symbols
            u"\U0001F700-\U0001F77F" # alchemical symbols
            u"\U0001F780-\U0001F7FF" # geometric shapes extended
            u"\U0001F800-\U0001F8FF" # supplemental arrows-C
            u"\U0001F900-\U0001F9FF" # supplemental symbols and pictographs
            u"\U0001FA00-\U0001FAFF" # chess symbols
            u"\U0001FA70-\U0001FAFF" # symbols and pictographs extended-A
            u"\U0001F900-\U0001F900" # additional emoticons
            u"\U0001F1E0-\U0001F1FF" # flags
        "]+", flags=re.UNICODE)
        return emoji_pattern.sub(r'', tweet)
    else:
        return tweet

# Fungsi untuk menghapus simbol
def remove_symbols(tweet):
    if tweet is not None and isinstance(tweet, str):
        tweet = re.sub(r'[^a-zA-Z0-9\s]', '', tweet) # hanya huruf, angka, spasi
    return tweet
```

Gbr. 5 Source Code Cleaning

Pada Gbr 5 menunjukkan source code *cleaning*, yang berisi fungsi untuk menghapus mention, emoji, angka, simbol, dan tautan. Hasil setelah proses *cleaning* dapat dilihat pada Tabel III.

TABEL III  
HASIL SETELAH CLEANING

No	Tweet
1	Melihat gaya mai STY di timnas dan stok striker yg ada spaso masih layak di panggil
2	Makanya pada ngebet juara AFF krn udah tw kapasitasnya klo di luar AFF kayak apa Tar semisal STY udah gak di timnas lagi Udah kebayang kan hasilnya gimana
3	DI BAWAH ASUHAN STY timnas indonesia oktober bersamagaruda ke via
4	Lima bintang Timnas Indonesia cedera jelang putaran kedua Kualifikasi Piala Dunia zona Asia

## 2. Case Folding

*Case folding* bertujuan menyeragamkan teks dengan mengubah seluruh huruf menjadi huruf kecil.

```
# Fungsi untuk mengubah huruf menjadi huruf kecil semua (lowercase)
def case_folding(text):
    if isinstance(text, str):
        lowercase_text = text.lower()
        return lowercase_text
    else:
        return text
```

Gbr. 6 Source Code Case Folding

Pada Gbr 6 menunjukkan source code *case folding*. Fungsi yang digunakan memeriksa apakah input berupa string, lalu mengubahnya menjadi lowercase. Contoh hasil proses *case folding* ditampilkan pada Tabel IV.

TABEL IV  
HASIL SETELAH CASE FOLDING

No	Tweet
1	melihat gaya mai sty di timnas dan stok striker yg ada spaso masih layak di panggil
2	makanya pada ngebet juara aff krn udah tw kapasitasnya klo di luar aff kayak apa tar semisal sty udah gak di timnas lagi udah kebayang kan hasilnya gimana
3	di bawah asuhan sty timnas indonesia oktober bersamagaruda ke via
4	lima bintang timnas indonesia cedera jelang putaran kedua kualifikasi piala dunia zona asia

## 3. Normalisasi Kata

Tahap normalisasi dilakukan untuk mengganti kata tidak baku dengan bentuk bakunya menggunakan kamus colloquial Indonesian lexicon.

```
# Fungsi untuk mengganti kata tidak baku berdasarkan kamus
def replace_taboo_words(text, kamus_tidak_baku):
    if isinstance(text, str):
        words = text.split()
        replaced_words = []
        kamimat_baku = []
        kata_diganti = []
        kata_tidak_baku_hash = []

        for word in words:
            if word in kamus_tidak_baku:
                baku_word = kamus_tidak_baku[word]
                if isinstance(baku_word, str) and all([char.isalpha() or char.isspace() for char in baku_word]):
                    replaced_words.append(baku_word)
                    kamimat_baku.append(baku_word)
                    kata_diganti.append(word)
                    kata_tidak_baku_hash.append(hash(word))
                else:
                    replaced_words.append(word)
            else:
                replaced_words.append(word)

        replaced_text = ' '.join(replaced_words)
    else:
        replaced_text = text
        kamimat_baku = []
        kata_diganti = []
        kata_tidak_baku_hash = []

    return replaced_text, kamimat_baku, kata_diganti, kata_tidak_baku_hash
```

Gbr. 7 Source Code Normalisasi Kata

Pada Gbr 7 berisi source code normalisasi kata yang memeriksa setiap token dan menggantinya bila ada padanan bakunya. Contoh hasil normalisasi dapat dilihat pada Tabel V.

TABEL V  
HASIL SETELAH NORMALISASI KATA

No	Tweet
1	melihat gaya mau sty di timnas dan stok striker yang ada spaso masih layak di panggil
2	makanya pada ngebet juara aff karena sudah tau kapasitasnya kalau di luar aff kayak apa entar semisal sty sudah tidak di timnas lagi sudah terbayang kan hasilnya bagaimana
3	di bawah asuhan sty timnas indonesia oktober bersamagaruda ke via
4	lima bintang timnas indonesia cedera jelang putaran kedua kualifikasi piala dunia zona asia

## 4. Tokenizing

Tokenizing memecah teks menjadi potongan kata (token) agar lebih mudah diproses.

```
def tokenize(text):
    tokens = text.split()
    return tokens

df['tokenize'] = df['hasil_normalisasi'].apply(tokenize)

df.head(5)
```

Gbr. 8 Source Code Tokenizing

Pada Gbr 8 menunjukkan source code *tokenizing* yang memisahkan teks berdasarkan spasi. Contoh hasil *tokenizing* ditampilkan pada Tabel VI.

TABEL VI  
HASIL SETELAH TOKENIZING

No	Tweet
1	['melihat', 'gaya', 'mau', 'sty', 'di', 'timnas', 'dan', 'stok', 'striker', 'yang', 'ada', 'spaso', 'masih', 'layak', 'di', 'panggil']
2	['makanya', 'pada', 'ngebet', 'juara', 'aff', 'karena', 'sudah', 'tau', 'kapasitasnya', 'kalau', 'di', 'luar', 'aff', 'kayak', 'apa', 'entar', 'semisal', 'sty', 'sudah', 'tidak', 'di', 'timnas', 'lagi', 'sudah', 'terbayang', 'kan', 'hasilnya', 'bagaimana']
3	['di', 'bawah', 'asuhan', 'sty', 'timnas', 'indonesia', 'oktober', 'bersamagaruda', 'ke', 'via']
4	['lima', 'bintang', 'timnas', 'indonesia', 'cedera', 'jelang', 'putaran', 'kedua', 'kualifikasi', 'piala', 'dunia', 'zona', 'asia']

##### 5. Stopword Removal

*Stopword removal* menghapus kata umum yang tidak memiliki kontribusi terhadap analisis sentimen, seperti "dan", "yang", atau "di".

```
# Fungsi untuk menghapus stopwords
def remove_stopwords(text):
    return [word for word in text if word not in stop_words]

# Terapkan fungsi pada kolom 'tokenize'
df['stopword_removal'] = df['tokenize'].apply(lambda x: remove_stopwords(x))

# Lihat hasil awal
df.head(5)
```

Gbr. 9 Source Code Stopword Removal

Pada Gbr 9 menampilkan source code *stopword removal* menggunakan daftar stopword NLTK. Contoh hasil implementasi proses ini terlihat pada Tabel VII.

TABEL VII  
HASIL SETELAH STOPWORD REMOVAL

No	Tweet
1	['gaya', 'sty', 'timnas', 'stok', 'striker', 'spaso', 'layak', 'panggil']
2	['ngebet', 'juara', 'aff', 'tau', 'kapasitasnya', 'aff', 'kayak', 'entar', 'sty', 'timnas', 'terbayang', 'hasilnya']
3	['asuhan', 'sty', 'timnas', 'indonesia', 'oktober', 'bersamagaruda', 'via']

4	['bintang', 'timnas', 'indonesia', 'cedera', 'jelang', 'putaran', 'kualifikasi', 'piala', 'dunia', 'zona', 'asia']
---	--

##### 6. Stemming

*Stemming* dilakukan dengan library *Sastrawi* untuk mengubah kata ke bentuk dasar agar lebih konsisten.

```
# Buat objek stemmer
factory = StemmerFactory()
stemmer = factory.create_stemmer()

# Fungsi stemming
def stem_text(text):
    return [stemmer.stem(word) for word in text]

# Terapkan stemming pada kolom 'stopword_removal'
df['stemming_data'] = df['stopword_removal'].apply(lambda x: ' '.join(stem_text(x)))

# Lihat hasil awal
df.head(5)
```

Gbr. 10 Source Code Stemming

Pada Gbr 10 berisi source code *stemming* yang menggunakan *StemmerFactory()*. Contoh hasil *stemming* dapat dilihat pada Tabel VIII.

TABEL VIII  
HASIL SETELAH STEMMING

No	Tweet
1	gaya sty timnas stok striker spaso layak panggil
2	ngebet juara aff tau kapasitas aff kayak entar sty timnas bayang hasil
3	asuh sty timnas indonesia oktober bersamagaruda via
4	bintang timnas indonesia cedera jelang putar kualifikasi piala dunia zona asia

##### 7. Labeling Data

Pada tahap pelabelan, penelitian ini menggunakan pendekatan berbasis leksikon dengan Indonesian Sentiment Lexicon (InSet). Setelah melalui *preprocessing*, setiap kata pada teks dibandingkan dengan daftar kata positif dan negatif dari InSet. Bobot masing-masing kata dijumlahkan untuk memperoleh skor sentimen. Jika skor  $> 0$  maka teks diberi label positif, skor  $< 0$  diberi label negatif, dan skor  $= 0$  diberi label netral.

```
# Fungsi untuk menentukan skor dan sentimen
def determine_sentiment(text):
    if isinstance(text, str):
        score = 0
        for word in text.split():
            if word in positive_lexicon_dict:
                score += positive_lexicon_dict[word]
            elif word in negative_lexicon_dict:
                score += negative_lexicon_dict[word]

        if score > 0:
            sentiment = "Positif"
        elif score < 0:
            sentiment = "Negatif"
        else:
            sentiment = "Netral"
        return score, sentiment
    return 0, "Netral"

# Fungsi untuk menampilkan bobot tiap kata
def word_scores(text):
    results = []
    if isinstance(text, str):
        for word in text.split():
            if word in positive_lexicon_dict:
                results.append((word, positive_lexicon_dict[word]))
            elif word in negative_lexicon_dict:
                results.append((word, negative_lexicon_dict[word]))
            else:
                results.append((word, 0))
    return results
```

Gbr. 11 Source Code Labeling Data

Pada Gbr 11 memperlihatkan potongan source code proses pelabelan. Pada kode tersebut, kamus InSet dimuat dalam format TSV kemudian diubah menjadi dictionary. Fungsi `determine_sentiment()` menghitung skor total berdasarkan bobot kata hasil *stemming*, lalu mengembalikan label sentimen. Hasil dari proses ditampilkan pada Tabel IX.

TABEL IX  
HASIL PELABELAN INSET

No	Tweet	Bobot	Label
1	melihat gaya mau sty di timnas dan stok striker yang ada spaso masih layak di panggil	-1	Negatif
2	makanya pada ngebet juara aff karena sudah tau kapasitasnya kalau di luar aff kayak apa entar semisal sty sudah tidak di timnas lagi sudah terbayang kan hasilnya bagaimana	1	Positif
3	di bawah asuhan sty timnas indonesia oktober bersamagaruda ke via	3	Positif
4	lima bintang timnas indonesia cedera jelang putaran kedua kualifikasi piala dunia zona asia	2	Positif

Hasil dari proses pelabelan dengan InSet menunjukkan distribusi sebagai berikut: 20.291 data positif, 8.159 data negatif, dan 1.766 data netral. Setelah pelabelan otomatis, dilakukan validasi manual oleh dua ahli Bahasa Indonesia untuk memastikan label sesuai konteks kalimat. Pada tahap ini, beberapa tweet mengalami perubahan label karena hasil perhitungan InSet tidak selalu mencerminkan makna konteks. Contoh perubahan label ditampilkan pada Tabel X.

TABEL X  
CONTOH PERUBAHAN LABEL

No	Tweet	InSet	Manual
1	melihat gaya mau sty di timnas dan stok striker yang ada spaso masih layak di panggil	Negatif	Positif
2	sty gabung ke klub korsel tandatanda tinggalkan indonesia selengkapnya di berita lengkap timnas indonesia sty seongnam	Negatif	Netral
3	makanya pada ngebet juara aff karena sudah tau kapasitasnya kalau di luar aff kayak apa entar semisal sty sudah tidak di timnas lagi sudah terbayang kan hasilnya bagaimana	Positif	Negatif
4	lima bintang timnas indonesia cedera jelang putaran kedua kualifikasi piala dunia zona asia	Positif	Netral
5	berharap timnas lolos dan ketemu korea lu pengin lihat profesionalisme sty bagaimana jika ketemu mantan tim yang pernah dia asuh	Netral	Positif
6	negaranya dibantai korea selatan media vietnam sindir timnas indonesia sty pakai formasi aneh lawan brunei	Netral	Negatif

Hasil akhir setelah validasi manual menunjukkan distribusi data sebagai berikut: 9.060 data berlabel negatif, 8.520 data berlabel netral, dan 12.636 data berlabel positif

#### 8. Encoding Label

Tahap terakhir adalah *encoding* label, yaitu mengubah kategori sentimen ke format numerik agar dapat diproses model. Skema yang digunakan yaitu 0 = negatif, 1 = netral, dan 2 = positif.

#### C. Pembagian Data

Dataset dibagi menjadi tiga subset yaitu training, validation, dan test menggunakan metode stratified random splitting agar distribusi label sentimen tetap seimbang. Dua skema pembagian digunakan dalam penelitian, yaitu 80–10–10 dan 70–15–15. Training set digunakan untuk melatih model, validation set untuk mengevaluasi kinerja selama pelatihan, dan test set untuk mengukur performa akhir pada data baru.

```

import pandas as pd
from sklearn.model_selection import train_test_split

# 1. Load dataset
df = pd.read_csv("dataset_STY_encoded_fix.csv")

# 2. Split Train & Temp (Train + Validasi)
train_df, temp_df = train_test_split(
    df,
    test_size=0.20,
    stratify=df['encoded_sentiment'],
    random_state=42
)

# 3. Split Temp jadi Validation & Test
val_df, test_df = train_test_split(
    temp_df,
    test_size=0.50,
    stratify=temp_df['encoded_sentiment'],
    random_state=42
)

# 4. Simpan hasil split ke file
train_df.to_csv("train_split.csv", index=False)
val_df.to_csv("val_split.csv", index=False)
test_df.to_csv("test_split.csv", index=False)

print(f"Jumlah data train: {len(train_df)}, val: {len(val_df)}, test: {len(test_df)}")
print("Pembagian data berhasil disimpan!")

```

Gbr. 12 Source Code Pembagian Data

Potongan kode pada Gbr. 12 menunjukkan proses pembagian dataset menggunakan stratify pada kolom label sehingga setiap subset memiliki proporsi kelas yang seimbang. Pada skema 80–10–10, data awal dibagi menjadi 80% training dan 20% data sementara, yang kemudian kembali dibagi menjadi validation dan test masing-masing 10%. Pada skema 70–15–15, pembagian dilakukan dengan prinsip serupa. Setiap subset disimpan dalam file CSV terpisah untuk kebutuhan training, validasi, dan evaluasi model.

#### D. Implementasi Model

Pada tahap ini dilakukan implementasi model analisis sentimen dengan mengombinasikan IndoBERT sebagai *feature extractor* dan beberapa arsitektur RNN, LSTM, BiLSTM, dan GRU sebagai lapisan klasifikasi. Proses implementasi terdiri dari beberapa tahapan sebagai berikut.

##### 1. Tokenisasi dengan IndoBERT

Teks hasil pembagian data ditokenisasi menggunakan tokenizer IndoBERT-base-p2 untuk mengubah kalimat menjadi `input_ids` dan `attention_mask`. Hasil tokenisasi kemudian dibentuk menjadi dataset yang siap digunakan dalam `DataLoader`.

##### 2. Pembentukan DataLoader

Data yang telah ditokenisasi dibungkus ke dalam `DataLoader` agar dapat diproses secara batch pada tahap training, validasi, dan testing. `Batch size` 16 dan 32 digunakan untuk keperluan eksperimen, dengan pengacakan (shuffle) diterapkan pada data training untuk mencegah model menghafal urutan data.

##### 3. Rancangan Arsitektur Model

Arsitektur model menggabungkan *embedding* IndoBERT dengan lapisan RNN, lalu dilanjutkan dengan *dropout* dan *fully connected layer* untuk menghasilkan prediksi tiga kelas sentimen. Variasi parameter yang digunakan meliputi jumlah hidden units (128 atau 256), jumlah lapisan (1 atau 2), serta nilai *dropout* (0.3 atau 0.5).

#### E. Training dan Validasi Model

Pada tahap ini dilakukan proses pelatihan dan validasi model analisis sentimen menggunakan arsitektur IndoBERT + RNN, IndoBERT + LSTM, IndoBERT + BiLSTM, dan IndoBERT + GRU. Model dikonfigurasi menggunakan `CrossEntropyLoss`, `optimizer AdamW`, dan dua nilai *learning rate* (2e-5 dan 3e-5). Proses training dijalankan dalam mode `train()` untuk menghitung loss dan memperbarui bobot, sedangkan evaluasi dilakukan pada validation set setiap akhir *epoch* untuk memperoleh nilai validasi seperti loss, akurasi, dan macro-F1.

Untuk mencegah overfitting digunakan *early stopping* dengan patience tiga *epoch*, dengan batas maksimum 20 *epoch*. Model dengan performa validasi terbaik disimpan sebagai best model, lalu bobot final diekspor dalam format `.safetensors`. Tahap evaluasi dilanjutkan dengan *classification report* yang menampilkan accuracy, precision, recall, dan F1-score, serta *confusion matrix* untuk melihat distribusi prediksi pada setiap kelas sentimen. Hasil training dan validasi adalah sebagai berikut.

##### 1. Dataset 1 (3 Label)

Pada tahap ini dilakukan pelatihan dan validasi model IndoBERT + RNN, IndoBERT + LSTM, IndoBERT + BiLSTM, dan IndoBERT + GRU menggunakan konfigurasi hyperparameter yang telah diuji. Parameter terbaik untuk masing-masing arsitektur disajikan pada Tabel XI.

TABEL XI  
PARAMETER KONFIGURASI TERBAIK 3 LABEL

Model	Split	Hidden Units	Dro pout	Layer	Batch Size	Leaning Rate
IndoBERT + RNN	80-10-10	128	0,3	1	16	2e-5
IndoBERT+ LSTM	80-10-10	128	0,3	1	32	3e-5
IndoBERT+ BiLSTM	80-10-10	128	0,3	1	32	2e-5
IndoBERT+ GRU	80-10-10	256	0,3	1	32	2e-5

Secara keseluruhan, keempat model menghasilkan performa yang relatif serupa dengan akurasi dan F1-score berada pada kisaran 0.69–0.70.

##### 2. Dataset 2 (2 Label)

Eksperimen dilanjutkan dengan skenario klasifikasi dua label (Positif dan Negatif). Parameter terbaik dari seluruh kombinasi disajikan pada Tabel XII.

TABEL XII PARAMETER KONFIGURASI TERBAIK 2 LABEL

Model	Split	Hidden Units	Drop out	Layer	Batch Size	Leaning Rate
IndoBERT + RNN	80-10-10	256	0,3	1	32	3e-5
IndoBERT + LSTM	80-10-10	256	0,3	1	16	2e-5
IndoBERT + BiLSTM	80-10-10	256	0,5	2	16	3e-5
IndoBERT + GRU	80-10-10	128	0,5	1	32	2e-5

Pada skenario dua label, seluruh model menunjukkan peningkatan performa yang signifikan dengan akurasi dan F1-score pada rentang 0.83–0.84. Hal ini menunjukkan bahwa penghapusan kelas Netral membuat model lebih mudah membedakan dua kelas utama.

#### F. Evaluasi Model

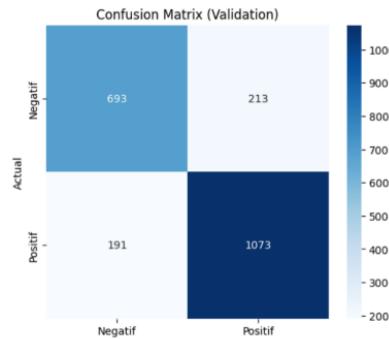
Pada tahap ini, model terbaik dari proses pelatihan diuji menggunakan data uji yang telah ditokenisasi dan disusun dalam *DataLoader*. Bobot model yang telah disimpan sebelumnya dimuat kembali, kemudian digunakan untuk menghasilkan prediksi. Kinerja model dievaluasi menggunakan *classification report* dan *confusion matrix*, sedangkan hasil prediksi disimpan dalam file CSV untuk analisis lebih lanjut. Hasil dari evaluasi model sebagai berikut.

##### 1. Dataset 1 (3 Label)

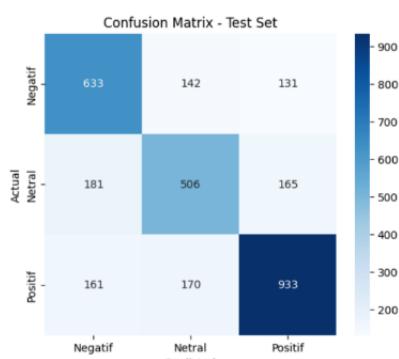
TABEL XIII  
HASIL EVALUASI MODEL DATASET 1 (3 LABEL)

Model	Performa	
	F1-Score	Akurasi
IndoBERT+RNN	0.68	0.69
IndoBERT+LSTM	0.68	0.69
IndoBERT+BiLSTM	0.67	0.68
IndoBERT+GRU	0.67	0.68

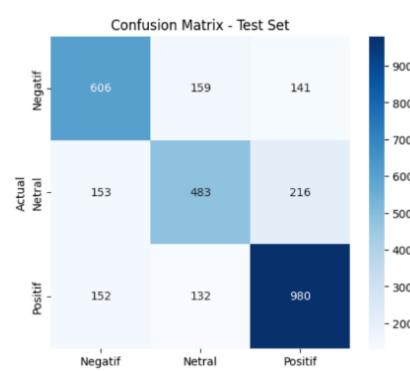
Hasil evaluasi empat model terbaik pada data uji 10% dirangkum pada Tabel XII IndoBERT+RNN dan IndoBERT+LSTM memperoleh akurasi tertinggi sebesar 0.69, diikuti BiLSTM dan GRU dengan akurasi 0.68.



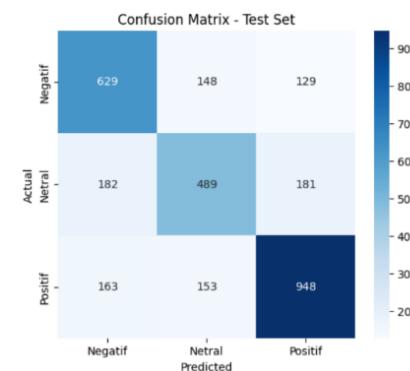
Gbr. 13 Confusion Matrix Testing IndoBERT + RNN



Gbr. 14 Confusion Matrix Testing IndoBERT + LSTM



Gbr. 15 Confusion Matrix Testing IndoBERT + BiLSTM



Gbr. 16 Confusion Matrix Testing IndoBERT + GRU

*Confusion matrix* pada Gbr 13-16 gambaran distribusi prediksi tiap model. Pada Gbr 13, model IndoBERT+RNN menunjukkan klasifikasi paling seimbang, terutama pada kelas Negatif dan Positif, meskipun kelas Netral masih menjadi tantangan. Gbr 14 memperlihatkan bahwa IndoBERT+LSTM memiliki pola prediksi yang serupa dengan RNN, namun sedikit lebih rendah dalam menangani kelas Netral. Pada Gbr 15, model BiLSTM terlihat cenderung menggeser prediksi ke kelas Positif, sehingga akurasi kelas Netral menurun. Sementara itu, *confusion matrix* pada Gbr 16 menunjukkan bahwa GRU memiliki distribusi prediksi yang stabil di ketiga kelas, tetapi tetap sedikit di bawah performa RNN dalam hal keseimbangan antar kelas. Secara keseluruhan, IndoBERT+RNN menjadi model terbaik pada skenario tiga label.

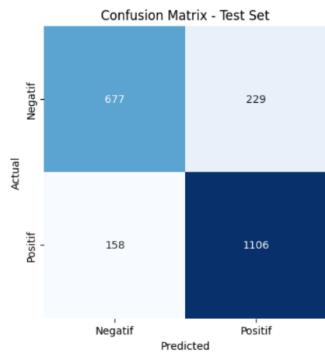
##### 2. Dataset 2 (2 Label)

TABEL XIV  
HASIL EVALUASI MODEL DATASET 2 (2 LABEL)

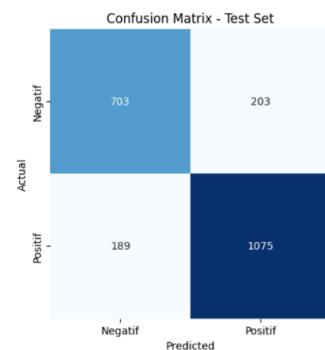
Model	Performa	
	F1-Score	Akurasi
IndoBERT+RNN	0.81	0.82
IndoBERT+LSTM	0.81	0.82
IndoBERT+BiLSTM	0.82	0.83
IndoBERT+GRU	0.83	0.83

Hasil uji coba model pada skenario dua label ditampilkan pada Tabel XIV, di mana IndoBERT+GRU mencapai performa terbaik dengan akurasi 0.83 dan F1-

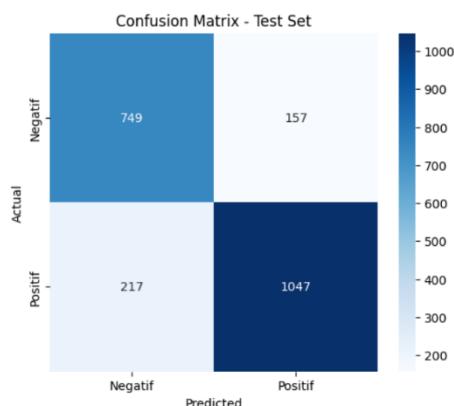
Score 0.83, diikuti BiLSTM dengan akurasi 0.83 dan F1-Score 0.82.



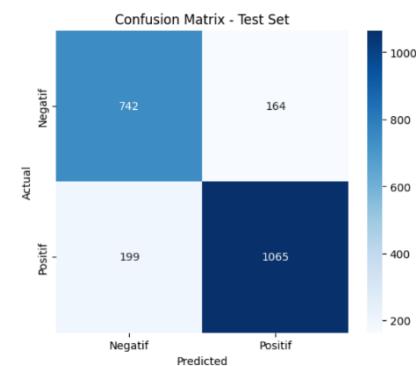
Gbr. 17 Confusion Matrix Testing IndoBERT + RNN 2 Label



Gbr. 18 Confusion Matrix Testing IndoBERT + LSTM 2 Label



Gbr. 19 Confusion Matrix Testing IndoBERT + BiLSTM 2 Label



Gbr. 20 Confusion Matrix Testing IndoBERT +GRU 2 Label

Confusion matrix pada Gbr 17-20 menunjukkan bahwa model IndoBERT + RNN (Gbr 17) dan IndoBERT + LSTM (Gbr 18) memiliki prediksi yang relatif seimbang, meski beberapa tweet ambigu masih salah terkласifikasi. Gbr 19 menunjukkan bahwa IndoBERT + BiLSTM mampu menangkap konteks dua arah dengan baik, menghasilkan peningkatan prediksi benar pada kelas Negatif. Sementara itu, IndoBERT+GRU pada Gbr 20 memperlihatkan distribusi prediksi paling stabil dan tingkat kesalahan terendah, mengonfirmasi bahwa IndoBERT + GRU menjadi model paling optimal pada skenario dua label.

#### IV. KESIMPULAN

Penelitian ini berhasil melakukan analisis sentimen terhadap opini pengguna X/Twitter mengenai Timnas Sepak Bola Indonesia pada era kepelatihan Shin Tae-yong menggunakan model berbasis IndoBERT yang dikombinasikan dengan arsitektur RNN (RNN, LSTM, BiLSTM, dan GRU). Proses *preprocessing*, pelabelan, pelatihan, validasi, dan pengujian dilakukan secara sistematis pada dua skenario dataset, yaitu tiga label (Negatif, Netral, Positif) dan dua label (Negatif, Positif).

Pada skenario tiga label, seluruh model mampu menjalankan tugas klasifikasi dengan performa yang stabil, dengan akurasi dan macro F1-score berada pada rentang 0.67–0.69. Model IndoBERT + RNN menjadi model terbaik dengan akurasi 0.69 dan F1-score 0.68, serta menunjukkan keseimbangan prediksi paling baik berdasarkan *confusion matrix*. Pada skenario dua label, performa model meningkat signifikan karena berkurangnya kompleksitas klasifikasi. Seluruh model mencapai akurasi 0.81–0.83. Model IndoBERT + GRU memberikan hasil terbaik dengan akurasi 0.83 dan F1-score 0.83, serta distribusi prediksi yang paling stabil antar kelas.

Secara keseluruhan, penelitian ini menunjukkan bahwa *finetuning* IndoBERT dengan arsitektur RNN mampu menghasilkan model analisis sentimen yang kuat dan kompetitif untuk teks bahasa Indonesia, khususnya untuk opini publik di media sosial. Model terbaik untuk skenario penelitian ini adalah IndoBERT + RNN (3 label) dan IndoBERT + GRU (2 label).

#### V. SARAN

Berdasarkan kesimpulan dan keterbatasan penelitian, saran yang dapat diberikan adalah sebagai berikut:

1. Penelitian selanjutnya disarankan untuk menggunakan model bahasa yang lebih baru seperti RoBERTa, DistilBERT, atau IndoBART, serta mengeksplorasi arsitektur BiGRU. Model-model tersebut berpotensi memberikan representasi konteks yang lebih baik dan dapat meningkatkan performa klasifikasi, khususnya pada kelas Netral yang masih menjadi tantangan utama pada penelitian ini. [3]
2. Dataset dapat diperluas tidak hanya dari platform Twitter/X, tetapi juga dari media sosial lain seperti Instagram, Facebook, atau forum online. Penambahan sumber data diharapkan mampu menghasilkan analisis sentimen yang lebih beragam, komprehensif, dan merepresentasikan opini masyarakat secara lebih luas. [4]

#### UCAPAN TERIMA KASIH

Peneliti mengucapkan syukur kepada Tuhan Yang Maha Esa atas segala rahmat, berkah, dan pertolongan-Nya sehingga penelitian serta penulisan artikel ilmiah ini dapat diselesaikan dengan baik. Ucapan terima kasih juga disampaikan kepada Dosen Pembimbing dan Koordinator Program Studi yang telah memberikan arahan serta masukan yang sangat berarti bagi peneliti. Penghargaan yang tulus peneliti sampaikan kepada kedua orang tua yang selalu memberikan doa dan dukungan dalam setiap proses yang dijalani. Tidak lupa, peneliti berterima kasih kepada seluruh saudara dan teman-teman di Jurusan Teknik Informatika angkatan 2021 yang senantiasa memberikan bantuan, semangat, dan motivasi hingga penelitian ini dapat diselesaikan dengan lancar.

#### REFERENSI

[1] Nielsen, "What fans want: World Football Report 2022," p. 10, 2022.

[2] Ira Guslina Sufa, "Perjalanan Timnas Indonesia Cetak Sejarah Asia," <https://katadata.co.id/berita/nasional/65b3094146e32/perjalanan-timnas-indonesia-cetak-sejarah-lolos-16-besar-piala-asia> 2024.

[3] A. Ferianto, "Tragedi Supporter Kanjuruhan Malang: Analisis Twitter Sebagai Alat Komunikasi Digital Pemerintah dan Organisasi Sepakbola Indonesia," *J. Soc. Bridg.*, vol. 1, no. 1, pp. 1–16, 2023, doi: 10.59012/jsb.v1i1.1.

[4] J. Devlin, M.-W. Chang, K. Lee, K. T. Google, and A. I. Language, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *Naacl-Hlt 2019*, no. Mlm, pp. 4171–4186, 2018, [Online]. Available: <https://aclanthology.org/N19-1423.pdf>

[5] F. Koto, A. Rahimi, J. H. Lau, and T. Baldwin, "IndoLEM and IndoBERT: A Benchmark Dataset and Pre-trained Language Model for Indonesian NLP," *COLING 2020 - 28th Int. Conf. Comput. Linguist. Proc. Conf.*, pp. 757–770, 2020, doi: 10.18653/v1/2020.coling-main.66.

[6] A. Bello, S. C. Ng, and M. F. Leung, "A BERT Framework to Sentiment Analysis of Tweets," *Sensors*, vol. 23, no. 1, 2023, doi: 10.3390/s23010506.

[7] A. Topbas, A. Jamil, A. A. Hameed, S. M. Ali, S. Bazai, and S. A. Shah, "Sentiment Analysis for COVID-19 Tweets Using Recurrent Neural Network (RNN) and Bidirectional Encoder Representations (BERT) Models," *2021 Int. Conf. Comput. Electron. Electr. Eng. ICE Cube 2021 - Proc.*, no. December, 2021, doi: 10.1109/ICECube53880.2021.9628315.

[8] M. J. Islam, R. Datta, and A. Iqbal, "Actual rating calculation of the zoom cloud meetings app using user reviews on google play store with sentiment annotation of BERT and hybridization of RNN and LSTM," *Expert Syst. Appl.*, vol. 223, no. January, p. 119919, Aug. 2023, doi: 10.1016/j.eswa.2023.119919.

[9] M. R. Ghufron, M. F. Mahabbatka Arsyada, M. R. Lukman, Y. A. Haryono Putra, and N. A. Rakhamwati, "Analisis Sentimen Pengguna Twitter Terhadap Pemilu 2024 Berbasis Model XLM-T," *J-Intech*, vol. 11, no. 2, pp. 307–315, 2023, doi: 10.32664/j-intech.v11i2.1013.

[10] S. A. Putra and A. Wijaya, "Analisis Sentimen Artificial Intelligence (Ai) Pada Media Sosial Twitter Menggunakan Metode Lexicon Based," *JuSiTik J. Sist. dan Teknol. Inf. Komun.*, vol. 7, no. 1, pp. 21–28, 2023, doi: 10.32524/jusitik.v7i1.1042.

[11] O. Muhammad and K. Khatami, "Analisis Sentimen Twitter Menggunakan Naive Bayes Dan Support Vector Machine Terhadap Kpu Pada Pemilihan Umum Presiden 2024," 2024.