

Optimization of IndoBERT for Sentiment Analysis of FOMO on Social Media Through Fine-Tuning and Hybrid Labeling

Nadhif Fauzil Adhim^{1*}, Nuri Cahyono^{2**}

* Informatika, Universitas Amikom Yogyakarta

** Informatika, Universitas Amikom Yogyakarta

nadhiffauzil@students.amikom.ac.id¹, nuricahyono@amikom.ac.id²

Article Info

Article history:

Received 2025-11-04

Revised 2025-12-01

Accepted 2025-12-10

Keyword:

*Sentiment Analysis,
FOMO,
IndoBERT,
Fine-tuning,
Hybrid Labeling.*

ABSTRACT

The rapid growth of social media in Indonesia has given rise to social phenomena such as Fear of Missing Out (FOMO). Expressions of FOMO on platforms like X (previously Twitter) often written informally, filled with abbreviations, slang, and emotional nuances, posing challenges for traditional Natural Language Processing (NLP) methods. This research aims to develop an optimized sentiment classification model for FOMO-related posts by fine-tuning the IndoBERT architecture and applying comprehensive data enhancement strategies. The study introduces three key innovations: (1) systematic text normalization to handle informal expressions, (2) a hybrid labeling framework combining automated model prediction, lexicon-based validation, and manual annotation to construct high-quality ground-truth data, and (3) hyperparameter tuning using both GridSearchCV for traditional machine learning models and Bayesian Optimization (Optuna) for deep learning models to maximize performance. The experimental results demonstrate that the optimized IndoBERT achieved superior performance with an Accuracy of 94.50%, F1-Score of 94.52%, and Macro AUC of 0.987. These results significantly surpass comparative models, including BiLSTM (Accuracy 86.60%), Support Vector Machine (88.06%), and Naive Bayes (80.73%). These results confirm that integrating hybrid labeling and fine-tuned IndoBERT significantly enhances sentiment classification performance. The findings contribute to developing reliable sentiment analysis systems for detecting social anxiety dynamics and computational social science research in Indonesian contexts.



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

I. PENDAHULUAN

Perkembangan ekosistem digital di Indonesia dalam sepuluh tahun terakhir sangat cepat, sehingga platform media sosial menjadi tempat utama bagi masyarakat untuk menyampaikan pendapat, perasaan, dan pengalaman hidup secara langsung. Platform seperti X (dulunya Twitter) kini bukan hanya alat komunikasi saja, tetapi juga mencerminkan cara masyarakat berperilaku di dunia digital. Dalam aliran informasi yang sangat cepat dan banyak, muncul suatu fenomena sosial yang disebut *Fear of Missing Out* (FOMO), yaitu perasaan cemas karena takut ketinggalan dari kegiatan, tren, atau pengalaman yang sedang ramai di lingkungan sosial baik daring maupun luring. Fenomena ini semakin besar karena penggunaan media sosial semakin intensif, di mana

individu terdorong untuk terus terhubung dan mengikuti perkembangan terkini agar tidak merasa tersisih secara sosial.

Ekspresso FOMO yang muncul di media sosial biasanya memiliki karakteristik linguistik yang khas, seperti penggunaan kalimat singkat, padat, emosional, serta sering kali disampaikan menggunakan bahasa tidak baku, *slang*, atau singkatan yang hanya dipahami oleh komunitas daring tertentu. Pola bahasa seperti ini menimbulkan tantangan tersendiri bagi sistem *Natural Language Processing* (NLP) dalam memahami makna kontekstual secara akurat, terutama karena sistem NLP konvensional cenderung bergantung pada struktur bahasa formal. Selain itu, ambiguitas semantik dan keragaman gaya bahasa pengguna media sosial memperumit proses analisis sentimen yang bertujuan mengidentifikasi emosi atau opini yang terkandung dalam teks.

Pendekatan awal dalam klasifikasi sentimen banyak menggunakan model *machine learning* tradisional yang mengandalkan fitur statistik seperti *Term Frequency–Inverse Document Frequency* (TF-IDF). Algoritma seperti *Support Vector Machine* (SVM) dan *Naïve Bayes* telah digunakan secara luas dan mampu mencapai akurasi antara 85–86% dalam analisis data media sosial berbahasa Indonesia [1]. Meskipun demikian, model-model tersebut masih terbatas dalam menangkap makna implisit yang muncul pada teks informal dan konteks percakapan yang dinamis [2],[3]. Untuk mengatasi keterbatasan ini, berbagai penelitian kemudian mengadopsi pendekatan berbasis *deep learning* dengan memanfaatkan model sekuensial seperti *Bidirectional Long Short-Term Memory* (BiLSTM). Model ini memiliki kemampuan untuk mempertimbangkan urutan kata dan hubungan antar konteks dalam kalimat sehingga memberikan pemahaman yang lebih baik terhadap makna yang tersirat. Sebagai contoh, penelitian pada data Twitter berbahasa Indonesia menunjukkan bahwa model BiLSTM yang dikombinasikan dengan *word embedding* berbasis domain tertentu mampu mencapai akurasi hingga 92,5% [4].

Kemajuan signifikan dalam bidang NLP terjadi ketika diperkenalkannya arsitektur *Transformer* oleh Vaswani et al. [5], yang menjadi landasan bagi berbagai model modern seperti *Bidirectional Encoder Representations from Transformers* (BERT) [6]. Arsitektur ini memungkinkan model untuk memahami konteks kata secara dua arah (kiri dan kanan), sehingga menghasilkan representasi bahasa yang lebih kaya dan mendalam. Untuk bahasa Indonesia, dikembangkan IndoBERT yang telah dilatih (*pre-trained*) pada korpus besar dan beragam sehingga mampu menangkap karakteristik semantik dan sintaktis bahasa Indonesia dengan lebih baik. Penelitian terdahulu menunjukkan bahwa proses *fine-tuning* terhadap IndoBERT secara signifikan meningkatkan kinerja analisis sentimen di berbagai domain. Misalnya, model IndoBERT yang di-*fine-tune* pada ulasan pariwisata mencapai akurasi 84% [7], pada data umum Twitter sebesar 91% [8], dan hingga 94% pada analisis sentimen berbasis aspek terhadap ulasan aplikasi [9]. Hasil-hasil tersebut memperlihatkan bahwa adaptasi model bahasa berskala besar seperti IndoBERT terhadap domain tertentu dapat memberikan peningkatan performa yang substansial.

Meskipun pendekatan *Deep Learning* menjanjikan, penelitian terdahulu masih terbatas dalam menangani kompleksitas linguistik FOMO yang penuh sarkasme dan slang, yang sering disalahartikan dibandingkan sentimen konvensional. Selain itu, model pre-trained seperti IndoBERT memerlukan adaptasi domain (*fine-tuning*) karena basis pelatihannya pada korpus formal berbeda drastis dengan dialek informal media sosial, sehingga rentan gagal menangkap nuansa kecemasan implisit.

Penelitian ini mengisi kekosongan tersebut dengan mengembangkan strategi optimasi model IndoBERT melalui mekanisme *fine-tuning* pada dataset spesifik dan penerapan kerangka kerja *Hybrid Labeling*. Pendekatan ini dirancang untuk memitigasi ambiguitas bahasa dengan menggabungkan

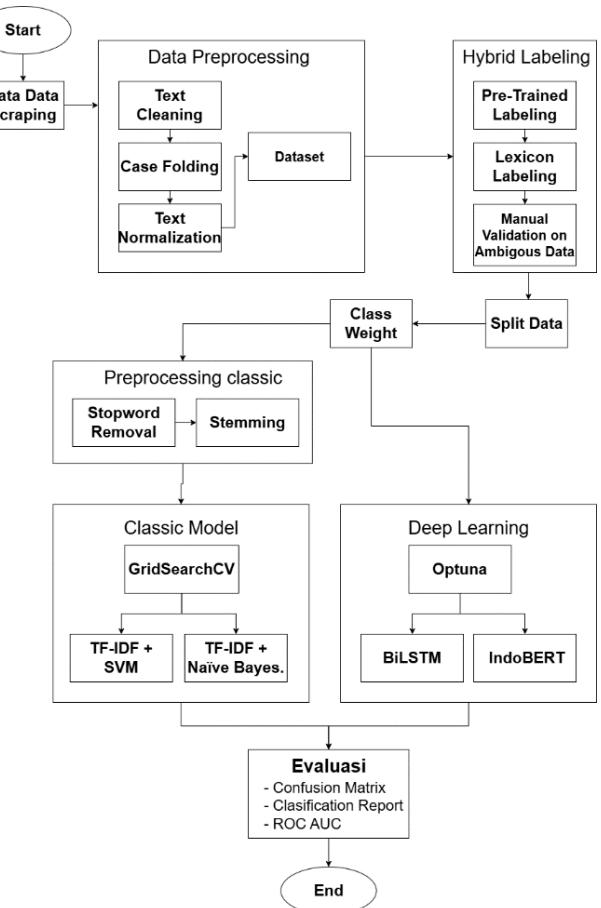
prediksi model, validasi leksikon, dan anotasi manual terstruktur, sehingga menghasilkan ground truth yang lebih akurat dibanding metode pelabelan tunggal.

Untuk memperjelas ruang lingkup data, penelitian ini menggunakan definisi operasional FOMO sebagai ungkapan yang menunjukkan rasa takut ketinggalan informasi, tren, atau aktivitas sosial. Identifikasinya dilakukan melalui dua indikator: (1) eksplisit penggunaan kata seperti “fomo”, “kudet”, “takut ketinggalan”; dan (2) implisit ekspresi negatif seperti penyesalan atau rasa iri, misalnya “nyesel banget nggak ikut war tiket” atau “timeline isinya konser semua, jadi merasa ketinggalan.”.

Penelitian ini memberikan dua kontribusi utama. Pertama, dari sisi metode, penelitian ini menggunakan teknik prapemrosesan dan pelabelan hibrida yang efektif untuk menangani teks Twitter yang penuh *noise*. Kedua, dari sisi penerapan, model IndoBERT yang telah dioptimalkan dapat menjadi langkah awal untuk memahami tanda-tanda kecemasan sosial di ruang digital dan mendukung riset psikoinformatika ke depannya.

II. METODE

Tahapan penelitian ini terdiri dari enam langkah utama seperti divisualisasikan pada diagram penelitian.



Gambar 1. Alur Penelitian

A. Akuisisi Data

Akuisisi data adalah langkah awal untuk mengumpulkan data mentah dari berbagai sumber sebelum diolah dan dianalisis. Dalam penelitian *Natural Language Processing* (NLP) dan analisis sentimen, data utama biasanya berupa teks yang dibuat oleh pengguna di media sosial.

Salah satu sumber data yang sering digunakan adalah platform X (dulu dikenal sebagai Twitter). Platform ini populer karena menyediakan banyak data opini publik yang terus diperbarui secara real-time dan mudah diakses. Data tersebut mencerminkan berbagai emosi serta fenomena sosial, termasuk *Fear of Missing Out* (FOMO).

Untuk mengumpulkan data dari platform seperti ini, peneliti biasanya menggunakan teknik *web scraping*, yaitu proses mengambil data dari halaman web secara otomatis dengan bantuan skrip atau program. Saat ini, sudah banyak library pemrograman yang membantu proses tersebut, seperti tweet-harvest pada Python, yang memudahkan pengumpulan data cuitan publik untuk keperluan analisis dan penelitian.

B. Preprocessing Data

Tahap pra-pemrosesan teks adalah dasar yang penting dalam penelitian ini, bertujuan untuk membersihkan dan membuat data cuitan mentah lebih rapi serta konsisten agar bisa diproses dengan baik oleh model. Teks dari media sosial biasanya tidak teratur, berbahasa santai, dan seringkali mengandung banyak *noise*. [10]. Oleh karena itu, serangkaian proses yang sistematis diperlukan langkah-langkah yang teratur untuk mengubah teks yang masih kasar menjadi bentuk yang bersih.

Proses dimulai dengan Pembersihan Teks (*Text Cleaning*) untuk menghapus elemen gangguan (*noise*) seperti URL, mention pengguna, dan tagar yang tidak relevan dari data mentah. Setelah teks dibersihkan, dilakukan tahap Penyeragaman Kasus (*Case Folding*) untuk mengubah seluruh huruf menjadi huruf kecil.

Langkah krusial dalam penelitian ini adalah Normalisasi Teks (*Text Normalization*). Mengingat banyaknya penggunaan bahasa gaul (*slang*) dan singkatan pada cuitan FOMO, normalisasi dilakukan dengan memetakan kata-kata tidak baku ke dalam bentuk baku Bahasa Indonesia yang sesuai dengan kaidah Ejaan Yang Disempurnakan (EYD). Sebagai contoh yang diterapkan dalam penelitian ini, kata berulang seperti "wihhh gilaaa" dinormalisasi menjadi "wih gila", dan singkatan "pgn" dikonversi menjadi "ingin". Proses ini sangat penting untuk memastikan bahwa variasi penulisan yang berbeda dapat dikenali sebagai fitur semantik yang sama oleh model.

Selanjutnya, alur pra-pemrosesan dipisahkan berdasarkan arsitektur model. Untuk model pembelajaran mesin klasik (SVM dan Naive Bayes), diterapkan proses penghapusan kata hubung (*Stopword Removal*) dan pemotongan imbuhan (*Stemming*) guna mereduksi dimensi fitur. Sebaliknya, untuk model *Deep Learning* (BiLSTM dan IndoBERT), kedua proses terakhir ini ditiadakan. Hal ini bertujuan untuk mempertahankan struktur sintaksis dan urutan kata asli yang

mengandung informasi kontekstual penting bagi model berbasis sekuensial.[10] [11] [12].

C. Hybrid Labeling

Strategi pelabelan data menerapkan pendekatan *Hybrid Labeling* yang mengintegrasikan prediksi model otomatis, validasi berbasis aturan leksikon, dan anotasi manual. Pendekatan ini dirancang untuk memitigasi bias sekaligus menangani volume data yang besar, sejalan dengan temuan bahwa pelabelan manual sepenuhnya membutuhkan waktu lama dan rentan subjektivitas anotator [13], sementara pelabelan otomatis sepenuhnya berpotensi menghasilkan kesalahan pada teks ambigu. Penting untuk dicatat bahwa proses validasi leksikon dilakukan setelah tahap normalisasi teks selesai. Dengan demikian, kata-kata slang yang telah dibakukan pada tahap sebelumnya dapat dicocokkan secara akurat dengan leksikon sentimen, sehingga validitas deteksi emosi tetap terjaga meskipun sumber data berasal dari bahasa informal.

Proses pelabelan berjalan melalui mekanisme penyaringan bertingkat. Pertama, model *pre-trained* memberikan label awal beserta skor keyakinan (*confidence score*), yaitu ukuran tingkat keyakinan model terhadap label yang dihasilkan [14]. Data dengan skor keyakinan tinggi, yaitu di atas ambang batas 0.866, langsung diterima sebagai label valid. Kedua, data dengan skor di bawah ambang batas tersebut divalidasi silang menggunakan metode leksikon.

Apabila terjadi konflik antara prediksi model dan hasil leksikon (misalnya model memprediksi positif sedangkan leksikon mendeteksi kata negatif), data tersebut dikategorikan sebagai Ambigu. Data ambigu ini kemudian diserahkan kepada anotator manusia untuk dilakukan pelabelan manual. Untuk menjamin objektivitas dan mengurangi potensi kesalahan otomatis, pendekatan berbasis aturan seperti validasi leksikon terbukti membantu meningkatkan kualitas dataset [15]. Pada tahap manual ini, anotator diberi pedoman ketat: sentimen 'Positif' diberikan jika terdapat antusiasme tanpa indikasi cemas, 'Negatif' jika terdapat ekspresi kecemasan, iri, atau kesedihan akibat ketertinggalan, dan 'Netral' jika pernyataan bersifat faktual.

D. Splitting Data

Pembagian data dilakukan dengan proporsi 80% untuk data latih (*training data*) dan 20% untuk data uji (*testing data*). Proses pemisahan menggunakan metode *stratified split* dengan tujuan mempertahankan keseimbangan distribusi kelas sentimen (positif, negatif, dan netral) pada kedua subset tersebut. Pendekatan *stratified* dipilih karena mampu menjaga representasi proporsional setiap kelas, sehingga pelatihan model berlangsung lebih stabil dan hasil evaluasi mencerminkan kemampuan generalisasi model terhadap data baru secara lebih akurat.

E. Class Weight

Distribusi data pada *datasets* media sosial umumnya tidak seimbang, di mana satu kelas sentimen seperti positif atau

negatif dapat memiliki jumlah sampel yang jauh lebih banyak dibandingkan dengan kelas lainnya. Ketidakseimbangan kelas tersebut berpotensi menyebabkan model menjadi bias, sehingga lebih akurat dalam memprediksi kelas mayoritas dan mengabaikan kelas minoritas.

Penelitian ini menerapkan teknik penyeimbangan kelas menggunakan metode *class weighting*. Teknik ini bekerja dengan memberikan bobot yang berbeda pada setiap kelas selama proses pelatihan model. Kelas minoritas diberikan bobot yang lebih tinggi, sedangkan kelas mayoritas diberikan bobot yang lebih rendah. Nilai bobot tersebut dihitung sebagai kebalikan dari frekuensi masing-masing kelas dalam data latih. Dengan demikian, kesalahan klasifikasi pada sampel dari kelas minoritas akan memberikan penalti (*loss*) yang lebih besar kepada model, sehingga mendorong model untuk belajar mengenali pola dari seluruh kelas secara lebih seimbang.

Bobot untuk setiap kelas W_j dihitung menggunakan persamaan berikut

$$W_j = \frac{n}{k \times n_j}$$

n = jumlah total sampel.
 k = jumlah total kelas.
 n_j = jumlah sampel dalam kelas ke- j .

Tujuan adalah untuk meningkatkan performa model pada metrik yang sensitif terhadap ketidakseimbangan data, seperti F1-score, serta menghasilkan model klasifikasi yang lebih adil.

F. Experiment Model

Untuk mengevaluasi pendekatan yang paling efektif dalam mengklasifikasikan sentimen FOMO, penelitian ini menerapkan dan membandingkan empat arsitektur model yang berbeda. Setiap model merepresentasikan pendekatan yang berbeda dalam pemrosesan teks, mulai dari model statistik klasik hingga arsitektur *deep learning* berbasis *Transformer*.

1) Support Vector Machine (SVM)

Support Vector Machine (SVM) adalah algoritma pembelajaran mesin yang menggunakan pendekatan terawasi. Algoritma ini bekerja dengan mencari hyperplane optimal yang bisa mengelompokkan data ke dalam kelas-kelas yang berbeda, dengan jarak antar kelas yang terbesar. Dalam konteks klasifikasi teks, setiap dokumen (cuitan) direpresentasikan sebagai vektor dalam ruang berdimensi tinggi, di mana setiap dimensi merepresentasikan bobot fitur seperti TF-IDF.

SVM bertujuan untuk menemukan bidang pemisah yang paling baik dalam membedakan sentimen positif, negatif, dan netral [16]. Secara matematis, *hyperplane* pemisah dapat dinyatakan sebagai

$$w \cdot x - b = 0$$

w = vektor Normal
 x = vektor fitur dari data
 b = parameter bias

2) Naive Bayes

Naive Bayes adalah metode klasifikasi yang menggunakan konsep *Teorema Bayes*. Model ini menghitung kemungkinan sebuah teks termasuk ke dalam kategori sentimen tertentu berdasarkan kata-kata yang ada di dalam teks tersebut. Meskipun disebut "naif" karena mengasumsikan bahwa setiap kata dalam teks tidak saling tergantung, yang sebenarnya kurang tepat dalam bahasa alami, model ini tetap bisa bekerja dengan baik dan cepat dalam analisis sentimen. [17].

Probabilitas *posterior* untuk sebuah kelas c (*sentimen*) diberikan dokumen dapat dihitung menggunakan rumus

$$P(c | d) \propto P(c) \prod_{i=1}^n P(w_i | c)$$

$P(c | d)$ = probabilitas d termasuk dalam kelas c
 $P(c)$ = probabilitas awal
 $P(w_i | c)$ = probabilitas kemunculan kata ke- i
 n = jumlah total kata dalam dokumen

3) Bidirectional Long Short-Term Memory (BiLSTM)

Bidirectional Long Short-Term Memory (BiLSTM) merupakan arsitektur Jaringan Saraf Tiruan Berulang (*Recurrent Neural Network* – RNN) yang dirancang khusus untuk memproses data sekuensial seperti teks. Berbeda dengan model statistik, BiLSTM mampu mempertimbangkan urutan kata serta menangkap ketergantungan kontekstual dalam kalimat.

Arsitektur BiLSTM memproses sekuens teks dari dua arah, yaitu dari depan ke belakang (*forward pass*) dan dari belakang ke depan (*backward pass*). Representasi dari kedua arah ini kemudian digabungkan sehingga model dapat membangun pemahaman kontekstual yang lebih komprehensif terhadap setiap kata. Penelitian sebelumnya menunjukkan bahwa BiLSTM mampu mencapai tingkat akurasi yang tinggi dalam klasifikasi sentimen pada teks berbahasa Indonesia

4) IndoBERT

IndoBERT adalah model bahasa canggih yang menggunakan arsitektur *Transformer* [18] dan telah dilatih secara khusus pada kumpulan teks besar dalam Bahasa Indonesia. Fondasi utama dari arsitektur *Transformer* adalah mekanisme *self-attention*, yang memungkinkan model menimbang tingkat kepentingan setiap kata terhadap kata lainnya dalam satu kalimat. Mekanisme ini memungkinkan IndoBERT menangkap hubungan kontekstual yang kompleks dengan lebih efektif dibandingkan model sekuensial seperti BiLSTM.

Model ini dibangun di atas arsitektur BERT (*Bidirectional Encoder Representations from Transformers*) [19]. Dalam penelitian ini, model IndoBERT yang sudah dilatih diubah lagi agar bisa digunakan untuk tugas klasifikasi sentimen FOMO melalui proses *fine tuning*. Proses ini melibatkan penambahan lapisan klasifikasi di atas arsitektur IndoBERT serta pelatihan ulang parameter model pada himpunan data FOMO yang telah dilabeli. Dengan memanfaatkan pengetahuan bahasa yang telah dipelajari, *fine-tuning* terbukti efektif dalam meningkatkan performa model secara signifikan pada tugas klasifikasi teks berbahasa Indonesia [9].

G. HyperParameter Tuning

Hiperparameter adalah parameter eksternal yang konfigurasinya tidak dipelajari selama proses pelatihan, seperti *learning rate* pada model *deep learning* atau parameter C pada SVM. Proses penalaan ini sangat penting karena konfigurasi yang tepat dapat meningkatkan akurasi dan kemampuan generalisasi model secara signifikan. Dalam penelitian ini, dua pendekatan berbeda digunakan sesuai dengan kompleksitas model yang diuji

1) Grid Search

Untuk model pembelajaran mesin klasik seperti Support Vector Machine (SVM) dan Naive Bayes, metode penyetelan yang digunakan adalah *Grid Search*. Pendekatan ini bekerja dengan mendefinisikan sebuah kisi (*grid*) yang berisi kombinasi kemungkinan hiperparameter. Algoritma kemudian melatih dan mengevaluasi model secara menyeluruh terhadap setiap kombinasi yang ada dalam kisi tersebut.

2) Bayesian Optimization

Model *deep learning* seperti Bidirectional Long Short-Term Memory (BiLSTM) dan IndoBERT memiliki ruang pencarian hiperparameter yang jauh lebih besar serta waktu pelatihan yang lebih lama, sehingga pendekatan *Grid Search* menjadi tidak efisien. Untuk itu, penelitian ini menggunakan Optuna, yaitu kerangka kerja optimasi otomatis yang menerapkan algoritma *Bayesian Optimization* [20].

Untuk memastikan evaluasi yang adil (*fair comparison*) antar berbagai arsitektur model, penelitian ini menerapkan skenario pengujian yang seragam. Seluruh model baik model statistik klasik maupun model *Deep Learning* dilatih dan diuji menggunakan himpunan data yang identik. Pembagian data dilakukan melalui metode *stratified split* dengan proporsi 80% untuk data latih dan 20% untuk data uji, sehingga distribusi kelas sentimen tetap seimbang pada setiap eksperimen.

Perbedaan perlakuan hanya diterapkan pada representasi fitur input (*feature representation*) sesuai karakteristik masing-masing algoritma. Pada model SVM dan Naive Bayes, teks diubah menjadi representasi berbasis TF-IDF (*Term Frequency–Inverse Document Frequency*) untuk menonjolkan signifikansi kata kunci dalam dokumen. Sementara itu, model BiLSTM dan IndoBERT memproses

input dalam bentuk sekuens token secara kontekstual untuk menangkap hubungan semantik antarkata.

Dengan menggunakan data latih dan uji yang sama persis, setiap perbedaan performa yang muncul dapat dikaitkan secara langsung dengan kapasitas masing-masing arsitektur dalam memahami konteks teks, bukan akibat perbedaan kualitas atau bentuk data input.

H. Evaluasi

Tahap evaluasi bertujuan untuk mengukur dan membandingkan kinerja dari setiap model yang telah dilatih secara objektif. Evaluasi dilakukan dengan menggunakan data uji yang tidak pernah digunakan selama proses pelatihan. Tujuannya adalah untuk mendapatkan gambaran yang menyeluruh mengenai kelebihan dan kekurangan dari setiap arsitektur model, penelitian ini menggunakan serangkaian metrik evaluasi standar dalam tugas klasifikasi

Confusion Matrix adalah tabel yang merepresentasikan hasil prediksi model dengan memperlihatkan perbandingan antara kelas aktual dan kelas yang diprediksi. Pada kasus klasifikasi multikelas seperti penelitian ini (positif, negatif, dan netral), matriks berukuran 3×3 digunakan. Komponen utama dari matriks tersebut meliputi

- 1) True Positive (TP) = prediksi benar pada positif
- 2) True Negative (TN) = prediksi benar pada negatif
- 3) False Positif (FP) = negatif diprediksi positif
- 4) False Negative (FN) = positif diprediksi negatif

Dari matriks ini, beberapa metrik kuantitatif dapat diturunkan untuk mengevaluasi kinerja model secara lebih rinci.

- 1) Akurasi (*Accuracy*): Mengukur proporsi total prediksi yang benar dari keseluruhan data. Akurasi adalah metrik yang paling intuitif, namun bisa menyesatkan pada himpunan data yang tidak seimbang.

$$\text{Akurasi} = \frac{TP + TN}{TP + TN + FP + FN}$$

- 2) Presisi (*Precision*): Presisi mengukur proporsi prediksi positif yang benar dari total prediksi yang diklasifikasikan sebagai positif.

$$\text{Presisi} = \frac{TP}{TP + FP}$$

- 3) Recall (*Sensitivity*): Recall mengukur proporsi kasus positif aktual yang berhasil diidentifikasi dengan benar oleh model.

$$\text{Recall} = \frac{TP}{TP + FN}$$

- 4) F1 – Score: F1-Score merupakan rata-rata dari Presisi dan Recall. Metrik ini sangat berguna dalam evaluasi

model pada himpunan data yang tidak seimbang karena menyeimbangkan kedua metrik tersebut.

$$F1\text{-Score} = 2 \times \frac{\text{Presisi} \times \text{Recall}}{\text{Presisi} + \text{Recall}}$$

- 5) Kurva ROC dan AUC: *Receiver Operating Characteristic* (ROC Curve) merupakan representasi grafis yang menunjukkan kemampuan model dalam membedakan antara kelas positif dan negatif berdasarkan berbagai tingkat ambang batas. Kurva ini memetakan *True Positive Rate* (TPR) terhadap *False Positive Rate* (FPR)

$$FPR = \frac{FP}{FP + TN}$$

III. HASIL DAN PEMBAHASAN

A. Pengumpulan Data

Data diperoleh dari platform media sosial X (sebelumnya Twitter) dengan menggunakan pustaka tweet-harvest yang berfungsi untuk melakukan pengambilan data (scraping) secara otomatis. Target pengumpulan data difokuskan pada cuitan berbahasa Indonesia yang relevan dengan topik penelitian.

Kata kunci yang digunakan dalam proses pencarian meliputi “fomo”, “fomotiktok”, “harus ikut tren”, dan “takut ketinggalan”. Proses *scraping* dilakukan dalam rentang waktu 26 November 2024 hingga 24 April 2025. Dari proses tersebut diperoleh sejumlah 9546 cuitan unik yang menjadi himpunan data mentah (*raw dataset*). Data inilah yang selanjutnya digunakan sebagai masukan pada tahap prapemrosesan untuk memastikan kualitas dan konsistensi sebelum digunakan dalam proses analisis lebih lanjut.

TABEL I
DESKRIPSI FITUR DATASETS

No	Fitur	Deskripsi
1	conversation_id_str	ID unik untuk sebuah utas percakapan.
2	created_at	Waktu dan tanggal cuitan dipublikasikan.
3	id_str	ID unik untuk setiap cuitan.
4	full_text	Isi teks lengkap dari cuitan.
5	username	Nama pengguna dari akun pembuat cuitan.

B. Preprocessing Data

Seluruh 9.546 cuitan diproses melalui tiga tahap utama untuk mengubah data mentah menjadi teks yang bersih dan konsisten. Tahap pertama adalah pembersihan teks, yaitu menghapus URL, *mention*, tagar, serta menurunkan semua huruf menjadi kecil. Tahap berikutnya adalah normalisasi bahasa, yang menangani teks informal dengan menyederhanakan elongasi huruf (“wihhh gilaaa” → “wih gila”) dan mengubah singkatan seperti “bgt”, “kmrn”, dan “pgn” menjadi bentuk baku.

Setelah normalisasi, alur pemrosesan dibedakan sesuai kebutuhan model. Pada model klasik (SVM dan Naive Bayes), dilakukan *stopword removal* dan stemming untuk mengurangi dimensi fitur dan mengekstraksi bentuk dasar kata, misalnya “nonton” menjadi “tonton”. Sebaliknya, model *deep learning* (BiLSTM dan IndoBERT) mempertahankan teks asli tanpa penghapusan stopword maupun stemming karena arsitektur sekuensial dan *Transformer* membutuhkan konteks kalimat lengkap untuk memahami nuansa emosi secara optimal.

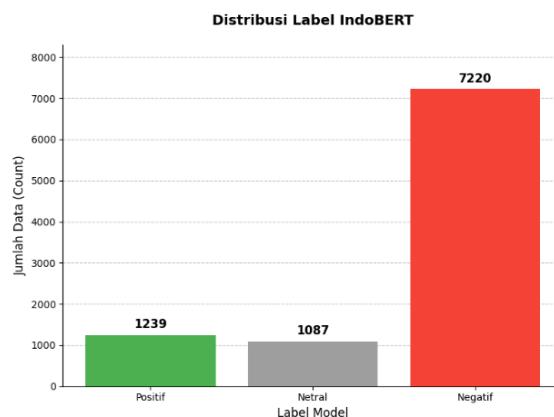
TABEL II
TAHAPAN PREPROCESSING DATA

No	Tahap Proses	Teks Hasil Pra-pemrosesan
	Text Asli	Wihhh gilaaa keran bgt konsernya kmrn, jadi FOMO pgn nonton lagi deh... Cek infonya di #konserkeren
1	Text Cleaning	Wihhh gilaaa keran bgt konsernya kmrn jadi FOMO pgn nonton lagi deh Cek infonya di
2	Case Folding	wihhh gilaaa keran bgt konsernya kmrn jadi fomo pgn nonton lagi deh cek infonya di
3	Normalisasi Slang	wih gila keran banget konsernya kemarin jadi fomo ingin nonton lagi deh cek infonya di
4	Stopword Removal (khusus SVM & NB)	wih gila keran banget konser kemarin fomo ingin nonton
5	Stemming (khusus SVM & NB)	wih gila keran banget konser kemarin fomo ingin tonton

C. Hybrid Labeling

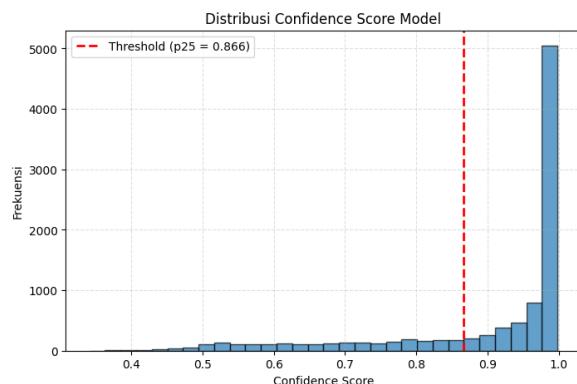
Cuitan yang telah dipra-pemrosesan dilabeli untuk menghasilkan *ground-truth* yang akan digunakan dalam pelatihan model. Proses ini mengikuti kerangka pelabelan hibrida yang telah dirancang, menggabungkan pelabelan otomatis, validasi leksikon, dan verifikasi manual untuk data yang ambigu.

- 1) Pelabelan Otomatis (Model *pretrained*): Pelabelan otomatis dilakukan menggunakan model praterlatih mdhugol/indonesia-bert-sentiment-classification. Model ini memberikan label awal (Positif, Negatif, atau Netral) beserta skor keyakinan (*confidence score*) untuk setiap cuitan. Berdasarkan hasil pelabelan awal terhadap total 9.546 cuitan, teridentifikasi dominasi sentimen negatif yang signifikan, yaitu sebanyak 7.220 cuitan (75,6%). Sementara itu, sentimen positif teridentifikasi sebanyak 1.239 cuitan (13,0%) dan sentimen netral sebanyak 1.087 cuitan (11,4%)



Gambar 2. Distribusi Label Sebelum Hybrid Labeling

- 2) Analisis Ambang Batas dan Validasi Hibrida
 Analisis distribusi skor keyakinan model dilakukan untuk menentukan nilai ambang batas (*threshold*).



Gambar 3. Distribusi Confidence Score dari model

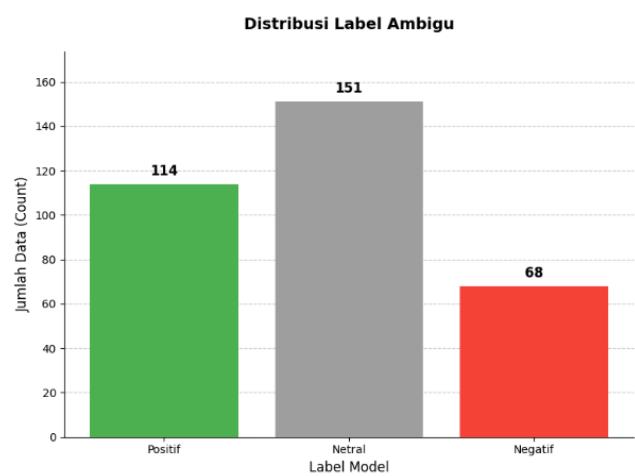
Ditemukan bahwa distribusi skor sangat condong ke kanan (majoritas prediksi memiliki keyakinan tinggi). Oleh karena itu, nilai persentil ke-25 (Q1) dipilih sebagai ambang batas yang representatif, yang mana didapatkan nilai 0.866. Data kemudian dipisahkan berdasarkan nilai ambang batas tersebut. Data dengan skor keyakinan tinggi (≥ 0.866) dianggap valid dan label yang diberikan oleh model diterima tanpa perubahan. Sementara itu, data dengan skor keyakinan rendah (< 0.866) harus melalui proses validasi menggunakan leksikon sentimen. Tabel III menyajikan aturan pelabelan.

TABEL III
 ATURAN PELABELAN HIBRIDA

Kondisi 1: Skor Keyakinan Model	Kondisi 2: Validasi Leksikon	Status Data	Tindakan Akhir
Tinggi (≥ 0.866)	<i>Tidak Diperlukan</i>	Valid	Gunakan Label Model
Rendah (< 0.866)	Konsisten (Label Model == Leksikon)	Valid	Gunakan Label Model

Rendah (< 0.866)	Bertantangan (Positif vs Negatif)	Ambigu	Lakukan Anotasi Manual
Rendah (< 0.866)	Indikasi Emosi (Model Netral, Leksikon memiliki sentimen)	Ambigu	Lakukan Anotasi Manual

Setelah menetapkan ambang batas, ditemukan 333 data yang dikategorikan sebagai ambigu. Berdasarkan prediksi awal model pada subset ini, label Netral mendominasi dengan jumlah 151 data, diikuti oleh Positif sebanyak 114 data, dan Negatif sebanyak 68 data



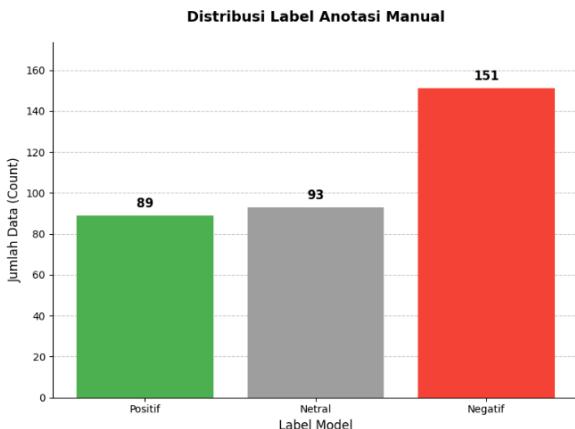
Gambar 4. Distribusi Label pada data ambigu

- 3) Anotasi Manual: Data yang ditandai ambigu sebanyak 333 cuitan, diserahkan kepada anotator manusia untuk divalidasi dan dilabeli ulang. Untuk menjamin konsistensi anotasi manual pada data ambigu dijaga melalui Pedoman Standar Anotasi. Protokol terstandar ini menggantikan intuisi subjektif anotator dengan indikator penilaian objektif, sehingga meminimalisir variabilitas interpretasi dan menjamin validitas *ground truth* yang dihasilkan. dan objektivitas selama proses ini, serangkaian aturan anotasi yang jelas telah ditetapkan disajikan pada tabel IV.

TABEL IV
PEDOMAN STANDAR ANOTASI

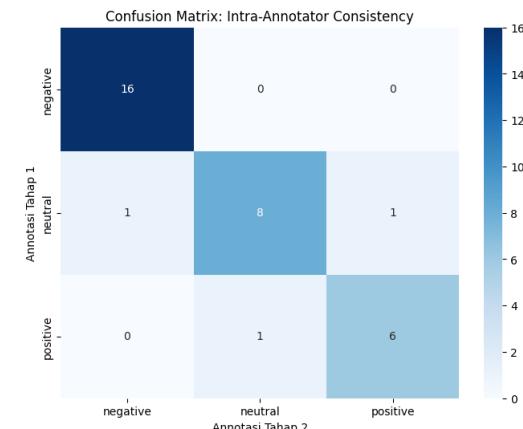
Kelas Sentimen	Definisi	Contoh
Positif	Mengekspresikan antusiasme, kegembiraan, atau partisipasi positif terkait suatu tren/acara. Tidak ada indikasi kecemasan.	Akhirnya bisa ikutan tren ini, seru banget
Negatif	Menggambarkan kecemasan, kesedihan, atau rasa iri akibat ketertinggalan terhadap suatu tren atau acara	Kenapa sih aku nggak diajak, sedih banget.
Netral	Bersifat faktual, informatif, atau menyebut kata fomo tanpa emosi.	Saya sedang membaca artikel penelitian tentang apa itu FOMO.

Ditemukan bahwa mayoritas dari data yang ambigu tersebut, yaitu 151 cuitan (sekitar 45.3%), teridentifikasi sebagai negatif. Sentimen netral teridentifikasi pada 93 cuitan (sekitar 27.9%), dan sentimen positif pada 89 cuitan (sekitar 26.7%). Distribusi ini menyoroti pentingnya validasi manual, karena sebagian besar data yang dianggap ambigu ternyata memiliki sentimen negatif.



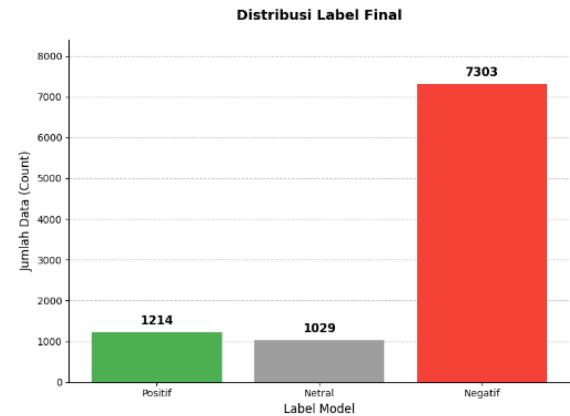
Gambar 5. Distribusi Sentimen Hasil Anotasi Manual

4) Intra-Annotator Consistency Check : dilakukan uji konsistensi internal (*Intra-Annotator Consistency Check*). Proses ini melibatkan pelabelan ulang terhadap sampel acak sebanyak 33 data (10% dari total data ambigu) tanpa merujuk pada keputusan label sebelumnya. Evaluasi statistik menghasilkan tingkat kesepakatan (*Agreement*) sebesar 90,91% dan nilai Cohen's Kappa sebesar 0,854. Berdasarkan standar interpretasi Landis & Koch, skor Kappa di atas 0,80 dikategorikan sebagai '*Almost Perfect Agreement*'.



Gambar 6. Confusion Matrix Intra-Annotator Consistency

5) Data Final: Dataset final diperoleh dengan menggabungkan data valid (yang memiliki skor keyakinan tinggi atau telah terverifikasi leksikon) dengan data ambigu yang telah selesai dianotasi secara manual. Dari total 9.546 cuitan yang siap digunakan untuk pelatihan model, distribusi kelas didominasi oleh sentimen Negatif sebanyak 7.303 cuitan (76,50%). Sementara itu, kelas sentimen lainnya memiliki proporsi yang jauh lebih kecil, yaitu sentimen Positif sebanyak 1.214 cuitan (12,72%) dan sentimen Netral sebanyak 1.029 cuitan (10,78%).



Gambar 7. Distribusi Final

D. Split Data

Seluruh Himpunan data final yang telah dilabeli kemudian dibagi menjadi dua bagian terpisah himpunan data latih (*training set*) dan himpunan data uji (*testing set*).

Pembagian data dilakukan dengan proporsi 80% untuk data latih dan 20% untuk data uji. Metode *stratified split* diterapkan untuk memastikan bahwa distribusi kelas sentimen (positif, negatif, dan netral) pada himpunan data final tetap terjaga secara proporsional di kedua himpunan data yang baru.

Hasil pembagian untuk himpunan data latih disajikan secara rinci pada Tabel V.

TABEL V
DISTRIBUSI KELAS PELATIHAN

Label	Jumlah
Negatif	5842 (76.51%)
Positif	971 (12.72%)
Netral	823 (10.78%)
Total	7636

E. Penyeimbang Kelas (Class Weighting)

Analisis terhadap himpunan data latih yang terdiri dari 7.636 sampel menunjukkan adanya ketidakseimbangan kelas (*class Imbalance*) yang signifikan. Ditemukan bahwa kelas 'Negatif' (5.842 sampel) jauh lebih dominan dibandingkan dengan kelas 'Positif' (971 sampel) dan 'Netral' (823 sampel).

Teknik *class weighting* diterapkan selama fase pelatihan model. Bobot dihitung sebagai kebalikan dari frekuensi setiap kelas. Berdasarkan perhitungan tersebut, bobot yang diterapkan adalah 0.4357 untuk kelas Negatif, 2.6214 untuk kelas Positif, dan 3.0931 untuk kelas Netral. Penggunaan bobot ini bertujuan untuk memberikan penalti yang lebih besar pada kesalahan klasifikasi di kelas minoritas, sehingga mendorong model untuk mengenali semua kelas secara lebih adil dan merata.

TABEL VI
HASIL PERHITUNGAN BOBOT KELAS

Kelas Sentimen	Jumlah Sample Data Latih n_j	Bobot Final W_j
Negatif	5842	0.4357
Positif	971	2.6214
Netral	823	3.0931

F. Hyperparameter Tuning

Untuk memastikan perbandingan yang adil dan untuk mencapai performa puncak dari setiap arsitektur, proses penyetelan *Hyperparameter* dilakukan. Dua strategi optimasi yang berbeda diterapkan sesuai dengan kompleksitas dan tipe model.

1) GridSearchCV untuk Model Statistik: Model *machine learning* statistik (SVM dan Naive Bayes) dioptimalkan menggunakan GridSearchCV. Pendekatan ini dipilih karena model statistik memiliki ruang pencarian yang relatif kecil dan cepat untuk dilatih

TABEL VII
RUANG PENCARIAN HIPERPARAMETER (GRIDSEARCHCV)

Model	Parameter	Nilai Yang Dicari
SVM	C	[0.1, 1, 10]
	kernel	['linear', 'rbf']
Naive Bayes	alpha	[0.01, 0.1, 0.5, 1.0]

2) Optuna untuk Model *Deep Learning*: Untuk arsitektur *deep learning* (BiLSTM dan IndoBERT), yang

memiliki ruang pencarian *hyperparameter* yang jauh lebih besar secara komputasi, Optuna digunakan.

TABEL VIII
RUANG PENCARIAN HIPERPARAMETER (OPTUNA)

Model	Parameter	Nilai Yang Dicari
BiLSTM	emb_dim	[100, 200, 300]
	hid_dim	[128, 256]
	n_layers	[1, 2]
	dropout	[0.3, 0.6]
	lr	[1e-4, 1e-2]
IndoBERT	learning_rate	[1e-5, 5e-5]
	num_train_epch	[2, 3]
	weight_decay	[0.01, 0.1]

3) Hasil Pencarian *Hyperparameter Tuning*: Pada model IndoBERT, pencarian menggunakan *Bayesian Optimization* menemukan bahwa penggunaan learning rate yang sangat rendah, yaitu sebesar 1.45×10^{-5} , merupakan faktor penentu utama dalam mencapai konvergensi optimal. Nilai *learning rate* yang kecil ini mencegah terjadinya *catastrophic forgetting* kondisi di mana model melupakan pengetahuan bahasanya yang sudah ada saat mempelajari data baru yang spesifik. Selain itu, penggunaan *weight decay* sebesar 0.052 memberikan regularisasi yang cukup untuk mencegah *overfitting*, mengingat kompleksitas model yang besar.

TABEL IX
HASIL OPTIMASI HIPERPARAMETER

Model	Metode	Parameter Terbaik
SVM	GridSearchCV	{'C': 10, 'kernel': 'rbf'}
Naïve Bayes	GridSearchCV	{'alpha': 0.1}
BiLSTM	Optuna	{'emb_dim': 200, 'hid_dim': 256, 'n_layers': 2, 'dropout': 0.350, 'lr': 0.0034}
IndoBERT	Optuna	{'learning_rate': 1.45e-05, 'num_train_epochs': 3, 'weight_decay': 0.052}

4) Evaluasi Model dan Analisis Hasil: Evaluasi model dilakukan pada *test set* (20% dari total data) untuk mengukur performa setiap skenario model. Pengujian ini menggunakan dua *datasets* yang berbeda untuk menganalisis kontribusi dari metodologi yang diusulkan. Pertama, *Dataset Awal*, yaitu himpunan data yang dilabeli murni oleh model *pretrained* mdhugol/indonesia-bert-sentiment-classification sebelum proses validasi hibrida. Hasil evaluasi pada datasets ini disajikan pada Tabel X. Kedua, *Dataset Final*, yaitu himpunan data yang telah melalui proses *Hybrid Labeling* dan anotasi manual, yang menghasilkan *ground-truth* yang lebih bersih, hasil evaluasi pada datasets ini disajikan pada tabel XI. Dalam kedua pengujian tersebut, istilah "*Baseline*" merujuk pada model yang dilatih dengan parameter default, sedangkan

"Optimized" merujuk pada model yang telah melalui proses *hyperparameter tuning* (Grid Search atau Optuna).

TABEL X
HASIL EVALUASI PADA DATASETS AWAL

Model	Akurasi	Presisi	Recall	F1-Score
Baseline SVM	0.8534	0.8578	0.8534	0.8298
Optimized SVM	0.8733	0.8671	0.8733	0.8644
Baseline Naive Bayes	0.7738	0.7650	0.7738	0.6841
Optimized Naive Bayes	0.8094	0.7982	0.8094	0.7715
Baseline BiLSTM	0.7613	0.8339	0.7613	0.7820
Optimized BiLSTM	0.8178	0.8320	0.8178	0.8227
Baseline IndoBERT	0.8796	0.8933	0.8796	0.8843
Optimized IndoBERT	0.9322	0.9321	0.9325	0.9322

Hasil evaluasi model yang dilatih menggunakan Dataset Final, yang telah melalui proses *hybrid labeling*

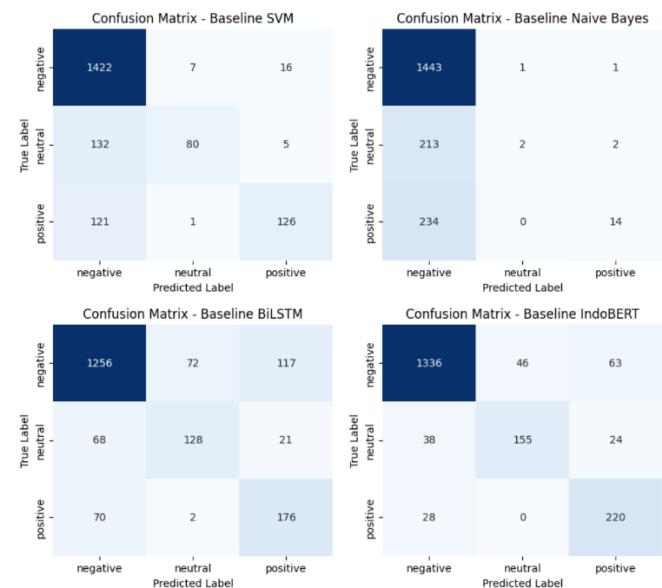
TABEL XI
HASIL EVALUASI PADA DATASETS FINAL

Model	Akurasi	Presisi	Recall	F1-Score
Baseline SVM	0.8524	0.8579	0.8524	0.8315
Optimized SVM	0.8806	0.8768	0.8806	0.8728
Baseline Naive Bayes	0.7644	0.7619	0.7644	0.6716
Optimized Naive Bayes	0.8073	0.7970	0.8073	0.7713
Baseline BiLSTM	0.7984	0.8267	0.7984	0.8079
Optimized BiLSTM	0.8660	0.8651	0.8660	0.8652
Baseline IndoBERT	0.8958	0.9016	0.8958	0.8972
Optimized IndoBERT	0.9450	0.9456	0.9450	0.9452

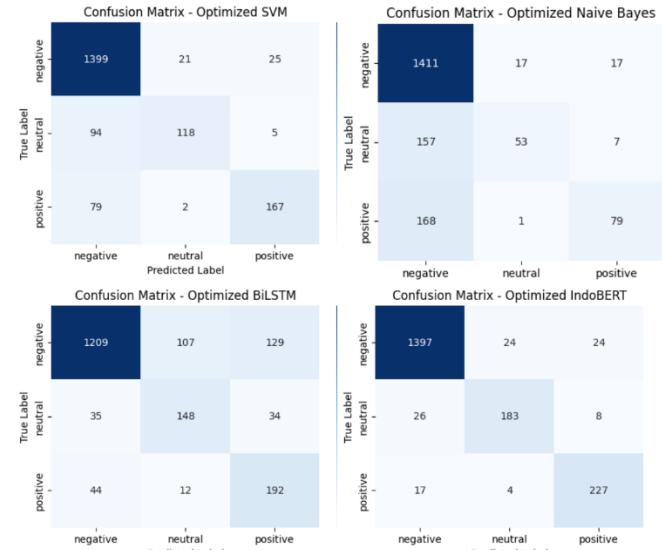
Proses *hyperparameter tuning* memberikan peningkatan performa yang konsisten pada semua arsitektur. IndoBERT menunjukkan peningkatan akurasi sebesar 4.92% (dari 89.58% menjadi 94.50%). BiLSTM mengalami peningkatan paling drastis, dengan akurasi melonjak 6.76% (dari 79.84% menjadi 86.60%). SVM meningkat sebesar 2.82%, dan Naive Bayes meningkat 4.29%. Analisis ini menegaskan bahwa proses *hyperparameter tuning* sangat penting untuk mengekstraksi performa maksimal dari setiap model.

Kontribusi dari kerangka *hybrid labeling* diukur dengan membandingkan performa model "Optimized" pada dataset awal dengan dataset final. Pada Optimized IndoBERT,

performa meningkat dari akurasi 93.22% menjadi 94.50% (peningkatan 1.28%) dan *F1-Score* dari 93.22% menjadi 94.52% (peningkatan 1.30%). Dampak yang lebih signifikan terlihat pada Optimized BiLSTM, dengan akurasi melonjak dari 81.78% menjadi 86.60% (peningkatan 4.82%). Optimized SVM juga mengalami sedikit peningkatan akurasi sebesar 0.73%. Analisis ini membuktikan bahwa proses *hybrid labeling* berhasil meningkatkan kualitas *ground-truth* secara signifikan, yang memberikan dampak paling positif pada model *deep learning* (IndoBERT dan BiLSTM) yang lebih sensitif terhadap kualitas label data

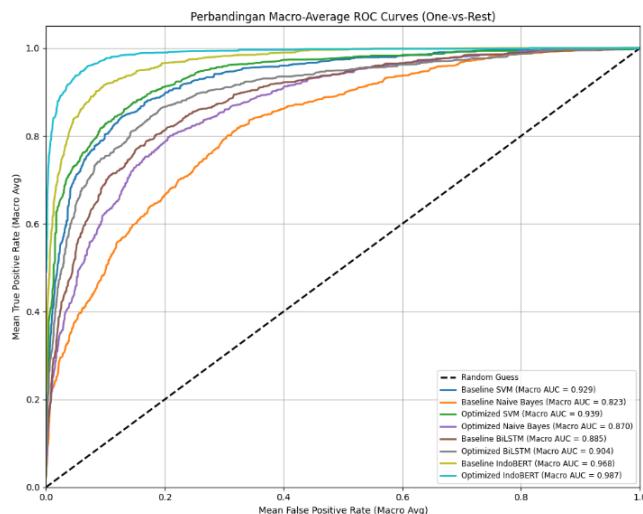


Gambar 8. Confusion Matrix Baseline



Gambar 9. Confusion Matrix Optimized

Analisis performa model diperkuat secara visual melalui kurva *Receiver Operating Characteristic* (ROC) *Macro-Average*. Kurva ROC memetakan *True Positive Rate* terhadap *False Positive Rate* dan *Area Under the Curve* (AUC) mengukur kemampuan diskriminatif model secara keseluruhan. Semakin tinggi nilai AUC dan semakin kurva mendekati sudut kiri atas, semakin baik performa model.



Gambar 10. Perbandingan Kurva ROC pada Datasets final

Model *Optimized IndoBERT* mencapai nilai *Macro AUC* tertinggi (0.987), dengan kurva yang paling curam dan paling mendekati sudut kiri atas, menandakan kemampuan diskriminatif yang superior di semua ambang batas. Model *deep learning* lainnya, *Optimized BiLSTM* (0.904), juga diungguli oleh model statistik terbaik, *Optimized SVM* (0.939). Peringkat AUC secara umum adalah: *IndoBERT* > *SVM* > *BiLSTM* > *Naive Bayes*. Selain itu, kurva untuk setiap model "*Optimized*" (misalnya, *Optimized IndoBERT AUC* 0.987) secara konsisten berada di atas kurva "*Baseline*" (*Baseline IndoBERT AUC* 0.968), yang secara visual menegaskan kontribusi positif dari proses *hyperparameter tuning*.

Hasil evaluasi baik dari metrik tabel maupun kurva ROC, dengan jelas menunjukkan bahwa model *IndoBERT* yang Dioptimalkan (*Optimized IndoBERT*) yang dilatih pada Dataset Final (hasil *Hybrid Labeling*) mencapai performa tertinggi secara keseluruhan. Model ini berhasil mencatatkan Akurasi 94.50%, F1-Score 94.52%, dan Macro AUC 0.987. Hasil ini mengkonfirmasi hipotesis penelitian bahwa kombinasi arsitektur *Transformer* (*IndoBERT*) yang telah disesuaikan dengan domain spesifik melalui *fine-tuning* dan dilatih pada himpunan data berkualitas tinggi hasil pelabelan hibrida merupakan pendekatan yang paling unggul untuk klasifikasi sentimen FOMO di media sosial.

IV. KESIMPULAN

Penelitian ini bertujuan untuk mengembangkan model klasifikasi sentimen yang optimal untuk fenomena *Fear of Missing Out* (FOMO) di media sosial, dengan fokus pada optimisasi arsitektur *IndoBERT* melalui *fine-tuning* dan penerapan kerangka pelabelan hibrida. Berdasarkan hasil evaluasi komprehensif, beberapa kesimpulan utama dapat ditarik.

Model *IndoBERT* yang dioptimalkan (*Optimized IndoBERT*) terbukti mencapai performa terbaik dibandingkan seluruh skenario pengujian. Keunggulan model ini tidak hanya terlihat pada Akurasi (94.50%), tetapi secara lebih signifikan pada perolehan F1-Score sebesar 94.52% dan Macro AUC 0.987. Tingginya nilai F1-Score ini mengindikasikan bahwa model memiliki kemampuan generalisasi yang sangat baik dalam mengenali kelas minoritas (Positif dan Netral) di tengah dominasi data sentimen Negatif, mengatasi kelemahan model *Deep Learning* sekuensial (*BiLSTM*, F1-Score 86.52%) dan model statistik (*SVM*, F1-Score 87.28%) yang cenderung memiliki celah performa antar kelas.

Kontribusi dari metodologi yang diusulkan terkonfirmasi melalui dua aspek kunci. Pertama, pendekatan *hybrid labeling* berhasil meningkatkan kualitas data latih, yang berdampak langsung pada kemampuan model dalam membedakan nuansa emosi ambigu. Hal ini dibuktikan dengan lonjakan performa signifikan pada model *BiLSTM* (kenaikan akurasi 4.82%) dan *IndoBERT* (1.28%) ketika dilatih menggunakan data hasil validasi hibrida dibandingkan data label otomatis semata.

Kedua, proses pencarian hiperparameter (*hyperparameter tuning*) terbukti esensial dalam adaptasi domain. Analisis menggunakan Optuna mengungkap bahwa penyesuaian learning rate ke nilai konservatif 1.45×10^{-5} dan pengaturan *weight decay* (0.052) merupakan parameter paling kritis. Konfigurasi spesifik ini berperan vital dalam mencegah *catastrophic forgetting* dan menjaga stabilitas model terhadap *noise* bahasa informal, yang berkontribusi pada peningkatan akurasi total sebesar 4.92% dibandingkan *baseline*.

Secara keseluruhan, penelitian ini menyimpulkan bahwa integrasi arsitektur *Transformer* (*IndoBERT*) dengan strategi *fine-tuning* presisi dan data berkualitas tinggi merupakan pendekatan yang paling robust untuk mendeteksi dinamika kecemasan sosial di platform digital, melampaui keterbatasan metode konvensional.

DAFTAR PUSTAKA

- [1] R. Syahputra, G. J. Yanris, and D. Irmayani, "SVM and Naïve Bayes Algorithm Comparison for User Sentiment Analysis on Twitter," *Sinkron*, vol. 7, no. 2, pp. 671–678, May 2022, doi: 10.33395/sinkron.v7i2.11430.
- [2] S. Pambudi, P. Setiaji, and W. A. Triyanto, "Sentiment Analysis of Fizzo Novel Application Using Support Vector Machine and Naïve Bayes Algorithm with SEMMA Framework," *Jurnal Teknik Informatika (JUTIF)*, vol. 6, no. 4, 2025, doi: 10.52436/1.jutif.2025.6.4.4875.

- [3] R. A. Fitrianto, A. S. Editya, M. M. Alamin, A. L. Pramana, and A. K. Alhaq, "Classification of Indonesian Sarcasm Tweets on X Platform Using Deep Learning," in *2024 7th International Conference on Informatics and Computational Sciences (ICICoS)*, IEEE, Jul. 2024, pp. 388–393, doi: 10.1109/ICICoS62600.2024.10636904.
- [4] V. P. Kalanjati *et al.*, "Sentiment analysis of Indonesian tweets on COVID-19 and COVID-19 vaccinations," *F1000Res*, vol. 12, 2024, doi: 10.12688/f1000research.130610.4.
- [5] A. Vaswani *et al.*, "Attention is All You Need," *arXiv preprint arXiv:1706.03762*, 2017, [Online]. Available: <https://arxiv.org/abs/1706.03762>
- [6] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *arXiv preprint arXiv:1810.04805*, 2019, [Online]. Available: <https://arxiv.org/abs/1810.04805>
- [7] R. I. Perwira, V. A. Permadji, D. I. Purnamasari, and R. P. Agusdin, "Domain-Specific Fine-Tuning of IndoBERT for Aspect-Based Sentiment Analysis in Indonesian Travel User-Generated Content," *Journal of Information Systems Engineering and Business Intelligence*, vol. 11, no. 1, pp. 30–40, Feb. 2025, doi: 10.20473/jisebi.11.1.30-40.
- [8] L. Afuan, N. Hidayat, H. Hamdani, H. Ismanto, B. C. Purnama, and D. I. Ramdhani, "Optimizing BERT Models with Fine-Tuning for Indonesian Twitter Sentiment Analysis," *J Wirel Mob Netw Ubiquitous Comput Dependable Appl*, vol. 16, no. 2, pp. 248–267, Jun. 2025, doi: 10.58346/JOWUA.2025.I2.016.
- [9] M. Widansyah, Fathia Frazna Az-Zahra, and Agung Pambudi, "Fine-Tuning Model Indobert (Indonesian Bidirectional Encoder Representations from Transformers) untuk Analisis Sentimen Berbasis Aspek pada Aplikasi M-Paspor," *Joutica*, vol. 9, no. 2, pp. 183–195, Sep. 2024, doi: 10.30736/informatika.v9i2.1310.
- [10] A. R. Lubis, Y. Y. Lase, D. A. Rahman, and D. Witorsyah, "Improving Spell Checker Performance for Bahasa Indonesia Using Text Preprocessing Techniques with Deep Learning Models," *Ingénierie des systèmes d'information*, vol. 28, no. 5, pp. 1335–1342, Oct. 2023, doi: 10.18280/isi.280522.
- [11] M. R. Manoppo *et al.*, "Analisis Sentimen Publik Di Media Sosial Terhadap Kenaikan PPN 12% Di Indonesia Menggunakan Indobert," *Jurnal Kecerdasan Buatan dan Teknologi Informasi*, vol. 4, no. 2, pp. 152–163, May 2025, doi: 10.69916/jkbt.v4i2.322.
- [12] A. Nafi', A. T. Harjanta, B. A. Herlambang, and S. Fahmi, "Analisis Sentimen Review Pelanggan Lazada dengan Sastrawi Stemmer dan SVM-PSO untuk Memahami Respon Pengguna," *J-INTECH*, vol. 12, no. 02, pp. 330–339, Dec. 2024, doi: 10.32664/jintech.v12i02.1450.
- [13] S. Saifullah, R. Dreżewski, F. A. Dwiyanto, A. S. Aribowo, Y. Fauziah, and N. H. Cahyana, "Automated Text Annotation Using a Semi-Supervised Approach with Meta Vectorizer and Machine Learning Algorithms for Hate Speech Detection," *Applied Sciences*, vol. 14, no. 3, p. 1078, Jan. 2024, doi: 10.3390/app14031078.
- [14] J. Baan, R. Fernández, B. Plank, and W. Aziz, "Interpreting Predictive Probabilities: Model Confidence or Human Label Variation?," in *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers)*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2024, pp. 268–277, doi: 10.18653/v1/2024.eacl-short.24.
- [15] M. Bosley, S. Kuzushima, T. Enamorado, and Y. Shiraito, "Improving Probabilistic Models In Text Classification Via Active Learning," *American Political Science Review*, vol. 119, no. 2, pp. 985–1002, May 2025, doi: 10.1017/S0003055424000716.
- [16] Q. Li *et al.*, "A Survey on Text Classification: From Traditional to Deep Learning," *ACM Trans Intell Syst Technol*, vol. 13, no. 2, pp. 1–41, Apr. 2022, doi: 10.1145/3495162.
- [17] P. Domingos and M. Pazzani, "On the Optimality of the Simple Bayesian Classifier under Zero-One Loss," *Mach Learn*, vol. 29, no. 2–3, pp. 103–130, Nov. 1997, doi: 10.1023/A:1007413511361.
- [18] B. Wilie, K. R. Vincentio, S. Cahyawijaya, G. I. Winata, Z. Li, and P. Fung, "IndoNLU: Benchmark and Resources for Evaluating Indonesian Natural Language Understanding," *arXiv preprint arXiv:2009.05387*, 2020, [Online]. Available: <https://arxiv.org/abs/2009.05387>
- [19] J. Devlin, M.-W. Chang, K. Lee, K. T. Google, and A. I. Language, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," [Online]. Available: <https://github.com/tensorflow/tensor2tensor>
- [20] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A Next-generation Hyperparameter Optimization Framework," Jul. 2019, [Online]. Available: <http://arxiv.org/abs/1907.10902>.