

## ANALISIS SENTIMEN TERHADAP INFLUENCER MENGGUNAKAN LSTM DAN SVM (LINIER SVC) PADA KOMENTAR PLATFORM X

GINNA ANGGRIANI BUULOLO<sup>1\*</sup>, UCTA PRADEMA SANJAYA<sup>2</sup>

Universitas Ngudi Waluyo<sup>12</sup>

E-mail: [ginnaanggriani04@gmail.com](mailto:ginnaanggriani04@gmail.com)<sup>1\*</sup>, [uctapradema@unw.ac.id](mailto:uctapradema@unw.ac.id)<sup>2</sup>

### Abstrak

Media sosial X telah menjadi platform penting bagi influencer dalam membentuk opini publik, tetapi komentar pengguna umumnya tidak terstruktur dan menggunakan Bahasa yang informal, dan memiliki sentimen yang beragam sehingga sulit untuk dianalisis. Penelitian ini bertujuan untuk mengklasifikasikan sentiment terhadap influencer di platform X serta membandingkan kinerja model *Deep Learning* LSTM dan *Machine Learning* Klasik SVM. Dengan menggunakan 8.252 komentar yang sudah melewati proses *Text Preprocessing*. Dan Untuk mengatasi ketidakseimbangan kelas, model BiLSTM menggunakan *embedding FastText* dan *Focal Loss*, sedangkan model SVM menerapkan *oversampling SMOTE*. Kinerja dievaluasi menggunakan akurasi dan F1-macro, serta diuji perbedaannya dengan uji McNemar. Hasil menunjukkan BiLSTM mencapai akurasi 0,68 dan F1-macro 0,66, sedikit lebih tinggi daripada SVM (akurasi 0,67; F1-macro 0,65). Namun, uji McNemar mengindikasikan perbedaan tidak signifikan ( $p > 0,05$ ). Temuan ini mengimplikasikan bahwa SVM tetap layak sebagai alternatif yang efisien untuk klasifikasi teks pendek pada sumber daya komputasi terbatas, sementara BiLSTM memberi peningkatan kecil. Batasan penelitian ini terletak pada cakupan data yang dibatasi oleh kata kunci dan rentang waktu pengambilan, sehingga generalisasi ke domain lain memerlukan studi lanjutan. Kebaruan penelitian terdapat pada perbandingan komprehensif kedua pendekatan pada komentar influencer yang informal dengan strategi penanganan imbalance yang berbeda.

**Kata Kunci:** *FastText; Focal Loss; Influencer; LSTM; SVM.*

### Abstract

*Social media platform X has become an important platform for influencers in shaping public opinion, but user comments are generally unstructured and use informal language, and have diverse sentiments that are difficult to analyze. This study aims to classify sentiment towards influencers on the X platform and compare the performance of the Deep Learning LSTM model and the Classic Machine Learning SVM model. Using 8,252 comments that have undergone text pre-processing. To overcome class imbalance, the BiLSTM model uses FastText embedding and Focal Loss, while the SVM model applies SMOTE oversampling. Performance is evaluated using accuracy and F1-macro, and the differences are tested using the McNemar test. The results show that BiLSTM*

51

Buulolo, G.A., Sanjaya, U. P. (2026). ANALISIS SENTIMEN TERHADAP INFLUENCER MENGGUNAKAN LSTM DAN SVM (LINIER SVC) PADA KOMENTAR PLATFORM X. *Jurnal Inovasi Pendidikan dan Teknologi Informasi (JIPTI)*, 7(1), 51-64. <https://doi.org/10.52060/jipti.v7i1.4009>

<http://ejournal.ummba.ac.id/index.php/JIPTI/>

---

achieves an accuracy of 0.68 and an F1-macro of 0.66, slightly higher than SVM (accuracy 0.67; F1-macro 0.65). However, the McNemar test indicated that the difference was not significant ( $p > 0.05$ ). These findings imply that SVM remains a viable alternative for efficient short text classification on limited computational resources, while BiLSTM provides a slight improvement. The limitations of this study lie in the scope of data, which is restricted by keywords and the time range of collection, so generalization to other domains requires further study. The novelty of this study lies in the comprehensive comparison of the two approaches to informal influencer comments with different imbalance handling strategies.

**Keywords:** FastText; Focal Loss; Influencer; LSTM; SVM.

---

**Submitted:** 2026-02-23. **Revision:** 2026-03-08. **Accepted:** 2026-03-19. **Publish:** 2026-04-01.

---

## PENDAHULUAN

Perkembangan media sosial telah mengubah cara individu berkomunikasi sekaligus membentuk opini secara cepat dan masif. Platform X menjadi salah satu ruang diskusi yang dominan karena interaksinya yang terbuka melalui balasan (reply), kutipan (quote). Dalam konteks ini, influencer berperan sebagai aktor yang mampu memengaruhi sikap dan perilaku audiens melalui narasi dan interaksi yang dibangun di ruang publik digital. Namun karena komentar pengguna di X biasanya tidak terstruktur seperti mengandung bahasa informal, singkatan, emoji dan simbol nonteks, sehingga analisis manual menjadi tidak efisien. Oleh karena itu, pendekatan otomatis berbasis *Natural Language Processing* (NLP) diperlukan untuk memproses opini dalam jumlah besar secara konsisten dan objektif (Arwindarti et al., 2024; Xu et al., 2022).

Beberapa penelitian telah mengaplikasikan NLP untuk analisis sentimen pada media sosial. (Prasetyo et al., 2023) menganalisis polaritas opini publik di Twitter terkait pemindahan Ibu Kota

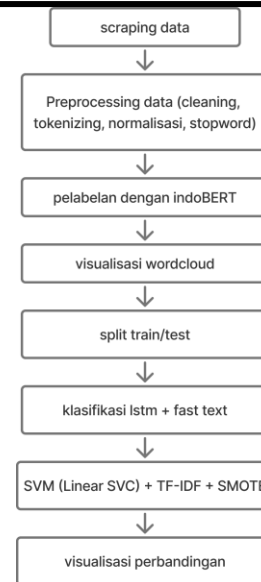
Nusantara menggunakan K-Nearest Neighbor, mencapai akurasi 88,12%. (Larasati et al., 2022) menerapkan Random Forest untuk mengekstrak keluhan pengguna dompet digital DANA, sementara (Safira & Hasan, 2023) menunjukkan bahwa metode klasik seperti *Naïve Bayes* tetap stabil untuk teks berbahasa Indonesia nonformal. Temuan-temuan tersebut menegaskan bahwa metode klasik masih relevan untuk klasifikasi sentimen, tetapi karakter data, konteks domain, serta bentuk interaksi pengguna dapat memengaruhi kinerja model. Di sisi lain, model *deep learning Long Short-Term Memory* (LSTM) terbukti lebih mampu menangkap konteks dan urutan kata dalam teks sekuensial (Irawan et al., 2025; Putra et al., 2022). Pra-pemrosesan teks yang tepat, termasuk tokenisasi, normalisasi kata, dan penghapusan *stopword*, juga berperan penting untuk mengurangi *noise* dan meningkatkan performa model (Eliviani & Wazaumi, 2024). Evaluasi menggunakan metrik akurasi, *precision*, *recall*, dan F1-score menjadi standar dalam penelitian terkini (Hasan et al., 2021).

Meskipun penelitian sebelumnya telah menunjukkan efektivitas berbagai metode

klasifikasi, masih terdapat kesenjangan dalam konteks komentar pengguna terhadap influencer di Platform X, terutama pada teks pendek berbahasa informal yang mengandung emoji, singkatan, dan campuran bahasa. Penelitian ini bertujuan untuk mengklasifikasikan sentimen komentar pengguna terhadap influencer di Platform X ke dalam kelas positif, negatif, dan netral, serta membandingkan kinerja model BiLSTM dengan *embedding FastText* dan *machine learning* klasik *Support Vector Machine* (SVM). Perbandingan ini diharapkan memperjelas pemilihan pendekatan klasifikasi yang lebih efektif untuk domain media sosial Indonesia yang berkarakter informal dan dinamis.

## METODE PENELITIAN

Metode penelitian yang digunakan dalam penelitian ini terdiri dari beberapa tahapan yang dimulai dari proses pengumpulan data komentar X hingga tahap evaluasi hasil klasifikasi sentimen. Secara umum, alur penelitian dirancang untuk mengolah data teks tidak terstruktur dan mengklasifikasikan sentimen pengguna terhadap influencer. Tahapan metode penelitian divisualisasikan dalam bentuk diagram alir (flowchart) dijabarkan sebagai berikut:



Gambar 1. *Flowchart* Metode Penelitian

Dalam konteks perkembangan teknologi informasi, *text mining* menjadi salah satu pendekatan penting dalam menggali pengetahuan dari data teks tidak terstruktur; kajian empiris menunjukkan efektivitas pendekatan ini dalam ekstraksi pola dan opini dari media sosial (Derbentsev et al., 2023). *Text mining* didefinisikan sebagai proses eksplorasi dan ekstraksi informasi dari kumpulan dokumen melalui penerapan teknik analisis tertentu untuk menemukan pola, hubungan, atau kecenderungan yang tersembunyi.

Pada penelitian ini, data yang digunakan berupa komentar pengguna pada unggahan Platform X influencer. Pengumpulan data dilakukan dengan cara *scraping* komentar publik pada akun X influencer yang dipilih, dengan mempertimbangkan aspek etika penelitian dan kebijakan privasi platform. Dalam praktik pengumpulan data daring, pedoman etika dan kebijakan institusi menjadi penting untuk memastikan kepatuhan terhadap privasi dan persyaratan penggunaan platform (Prasetyo et al., 2023).

Data komentar yang diperoleh kemudian disimpan dalam format terstruktur untuk memudahkan proses pengolahan dan analisis lebih lanjut. Sebelum dilakukan proses klasifikasi sentimen, data komentar diproses melalui tahap pra-pemrosesan teks. Tahapan ini meliputi *case folding* untuk mengubah seluruh teks menjadi huruf kecil, *tokenization* untuk memisahkan teks menjadi kata-kata, penghapusan *stopword*, Setelah itu, dilakukan normalisasi untuk mengurangi pengulangan karakter yang berlebihan (misalnya "baguuuuu" menjadi "baguus"). Penelitian tentang pengaruh pra-pemrosesan terhadap performa model menegaskan bahwa tahapan ini krusial untuk meningkatkan akurasi model klasifikasi teks, khususnya pada data media sosial yang penuh *noise* dan variasi bahasa (Brown et al., 2025).

Setelah tahap pra-pemrosesan, teks yang telah bersih direpresentasikan dalam bentuk numerik menggunakan teknik *word embedding*. Representasi berbasis *embedding* terbukti menangkap informasi semantik kata dan meningkatkan performa model deep learning pada tugas klasifikasi sentimen (Hadiprakoso et al., 2023). Representasi vektor (mis. Word2Vec, fastText, atau embedding pretrained lain) digunakan sebagai masukan ke model LSTM untuk mempelajari pola sekuensial dan konteks kata. model LSTM digunakan untuk melakukan klasifikasi sentimen komentar ke dalam tiga kelas, yaitu positif, netral, dan negatif. LSTM dipilih karena kemampuannya dalam memproses data sekuensial dan menangkap ketergantungan konteks jangka panjang pada teks. Selain itu, kajian komparatif pada teks pendek media sosial menunjukkan bahwa arsitektur

sekuensial (LSTM/BiLSTM) masih kompetitif untuk sejumlah kasus domain tertentu, terutama bila data dilatih dengan representasi fitur yang baik (Asyaky et al., 2025). Proses pelatihan model dilakukan menggunakan data latih, sedangkan pengujian model dilakukan pada data uji untuk mengukur kemampuan generalisasi model. Dan dari (Wen & Li, 2023) LSTM (Long Short-Term Memory) merupakan varian dari *Recurrent Neural Network* (RNN) yang dirancang untuk mengatasi masalah *vanishing gradient* dan *exploding gradient* melalui struktur *gate* (forget gate, input gate, dan output gate) yang menggunakan fungsi aktivasi sigmoid.

Pelabelan sentimen dilakukan secara otomatis (pseudo-labeling) menggunakan model transformer Indo-RoBERTa w11wo/indonesian-roberta-base-sentiment-classifier. Data yang dilabeli adalah teks pada kolom clean hasil preprocessing. Proses pelabelan dimulai dengan tokenisasi menggunakan *AutoTokenizer* kemudian dilakukan inferensi menggunakan *AutoModelForSequenceClassification* pada mode evaluasi. Output model berupa logits diubah menjadi probabilitas kelas melalui fungsi *softmax*, dan kelas sentimen ditentukan menggunakan argmax dengan tiga kategori, yaitu negatif, netral, dan positif. Dan dilakukan visualisasi kata yang paling sering muncul pada setiap kelas sentimen (negatif, netral, positif) menggunakan WordCloud. Setelah visualisasi, label sentimen pada kolom sentiment diubah menjadi format numerik menggunakan LabelEncoder sehingga setiap kelas memiliki kode angka tertentu. Data teks untuk pemodelan diambil dari kolom clean, kemudian dibagi menjadi data

latih dan data uji menggunakan `train_test_split` dengan rasio 80:20. Yang kemudian akan dilakukan klasifikasi menggunakan dua model LSTM dengan `fast text` dan `svm (linear SVC)` dengan `smote` dan `TF-IDF`. pada Tahap akhir dari penelitian ini adalah evaluasi dan analisis hasil klasifikasi sentimen. Evaluasi dilakukan menggunakan metrik akurasi, *precision*, *recall*, dan F1-score untuk menilai kinerja model secara komprehensif.

Dan pada jurnal (Perwira et al., 2025) menggunakan *IndoBERT* sebagai pelabelan otomatis pada dataset ulasan pariwisata berbahasa Indonesia. Proses labelling dilakukan dengan merepresentasikan teks ulasan sebagai *contextualized embeddings* menggunakan model *IndoBERT* pra-latih, kemudian menghitung *cosine similarity* antara *embedding* frasa ulasan dengan kata kunci aspek yang telah ditentukan. Aspek dengan skor kemiripan tertinggi ditetapkan sebagai label. Untuk memverifikasi kualitas label, dilakukan anotasi manual oleh tiga anotator dengan *Cohen's kappa* sebesar 0.82, menunjukkan tingkat kesepakatan yang kuat.

### Validasi Kualitas Label

Berdasarkan uji validasi manual pada sampel stratified sebanyak 286 data, reliabilitas anotasi manusia tergolong sangat tinggi dengan nilai *Cohen's Kappa* = 0,8422, yang menunjukkan konsistensi penilaian antar anotator kuat. Selanjutnya, kualitas pseudo-label *Indo-RoBERTa* dibandingkan terhadap gold label (majority vote anotator) menghasilkan akurasi 0,9301 dan F1-macro 0,9312, menandakan bahwa label otomatis memiliki kesesuaian yang sangat baik dengan penilaian manual. Secara per kelas, performa

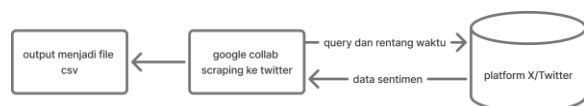
tetap stabil: negatif (F1=0,90; recall=0,86), netral (F1=0,97; recall=1,00), dan positif (F1=0,92; recall=0,95), dengan kesalahan utama terjadi pada sebagian kecil data negatif yang terprediksi sebagai positif (10 kasus) serta pertukaran kecil antara negatif–positif (masing-masing 5 kasus). Temuan ini mengindikasikan bahwa pseudo-label yang digunakan cukup reliable untuk dijadikan dasar evaluasi dan perbandingan model (SVM dan BiLSTM) pada eksperimen selanjutnya.

### Konfigurasi Model

Pada tahap eksperimen, data dibagi menggunakan stratified train-test split agar proporsi kelas negatif, netral, dan positif tetap seimbang pada data latih dan data uji, dengan komposisi 80% data latih dan 20% data uji, `random_state = 42`, serta `stratify = y`. Untuk menjaga konsistensi hasil, penelitian menetapkan `seed random 42` pada NumPy dan *TensorFlow*. Pada model klasik, representasi teks dibentuk menggunakan *TF-IDF* dengan maksimum 20.000 fitur dan ngram 1–2, kemudian untuk menangani ketidakseimbangan kelas penulis menggunakan SMOTE dengan parameter `k_neighbors = 5` dan `random_state = 42` sebelum dilatih menggunakan SVM kernel linear. Sementara itu, pada model *deep learning*, BiLSTM dilatih menggunakan *FastText embedding* 300 dimensi, maksimum 6.000 kata, dan panjang *sequence* 80 token, dengan fungsi kerugian categorical *focal loss* yang menggunakan `gamma = 2,0` dan `alpha = 0,25` untuk menekankan pembelajaran pada sampel yang sulit. Proses pelatihan dilakukan menggunakan batch size 64 hingga maksimum 15 epoch, serta EarlyStopping (`patience = 3`) dan ReduceLRonPlateau (`factor = 0,5`;

patience = 2; min\_lr = 1e-6) agar pelatihan lebih stabil dan mengurangi risiko *overfitting*.

### Pengumpulan Data



Gambar 2. *Scrapping* Sentimen Influencer di Platform X

Pengumpulan data dilakukan dengan cara melakukan scraping dari platform X/Twitter menggunakan Tweet Harvest yang dijalankan di Google Collab. Proses dimulai dari instalasi dependensi yang diperlukan oleh Tweet Harvest, kemudian *scrapping* dijalankan berdasarkan beberapa kata kunci dengan rentang waktu yang sama dari 01 Januari 2024 sampai 31 Desember 2025 dengan *query* “timothy”, “kalimasada”, “timothy saham”, “timothy bitcoin” untuk mengumpulkan sentimen yang relevan dan mendapat 45.745 data. Hasil *scrapping* kemudian disimpan ke dalam format csv. Yang nantinya akan digunakan pada proses analisis sentimen.

## HASIL DAN PEMBAHASAN

### Hasil

Dari total 45.745 data komentar yang didapat dari *scrapping* sentimen dilakukan filter lebih lanjut pada data. Filter yang dilakukan meliputi lang:id untuk menyaring sentimen yang memiliki lang:id saja. dan filter duplikat pada kolom full\_text dimana sentimen pengguna berada di sini karena saat melakukan *scrapping* berulang ada kemungkinan duplikat saat *scrapping*. dan di lanjut dengan proses pelabelan sentimen menggunakan Indo-RoBERTa diperoleh distribusi label sebagai berikut:

Tabel 1. Distribusi Label *IndoBERT*

Negatif	3.933	47,66%
Netral	2.373	28,76%
Positif	1.946	23,58%

### Preprocessing Data

Tahap *preprocessing* dilakukan untuk membersihkan teks komentar/sentimen sebelum dipakai oleh model klasifikasi. Pertama, teks sentimen akan diubah menjadi huruf kecil (*case folding*), lalu mengubah emoji yang ada pada sentimen ke teks agar tidak mengurangi makna atau emosi yang ada pada sentimen tersebut, selanjutnya penghapusan link/url seperti http dan www, dan mention @username, lalu menormalisasi sentimen seperti kata "sangat bagusss" menjadi "bagus" untuk mengurangi noise, dan diakhir teks akan di tokenisasi dengan NLTK `word_tokenize`. Hasil dari *preprocessing* ini akan berupa kolom `clean` dan `normalized_text`. Dan menurut (Safitri et al., 2025) Metode *preprocessing* teks berperan penting dalam pemodelan analisis sentimen karena berfungsi untuk menghilangkan *noise* dan menyelaraskan format data agar menjadi lebih bersih dan terstruktur sebelum dianalisis lebih lanjut. Implementasi *pipeline preprocessing* yang komprehensif seperti pembersihan karakter yang tidak relevan, *case folding*, tokenisasi, penghapusan stopword, hingga normalisasi kata slang atau singkatan terbukti secara empiris mampu meningkatkan kualitas sinyal data.

Tabel 2. Tahapan *Text Pre-processing*

Teks Asli	@DonTimoteo1902 @BudiBukanIntel Kalo mau kaya harus eksploitasi. Gw tanya andrew susanto
-----------	--

	timothy ronald kalimasada gabriel rey ajik krisna oleh2 itu apa yg di exploitasi?????????
Cleaned	kalo mau kaya harus exploitasi gw tanya andrew susanto timothy ronald kalimasada gabriel rey ajik krisna oleh2 itu apa yg di exploitasi
Normalized	kalau mau kaya harus eksploitasi saya tanya andrew susanto timothy ronald kalimasada gabriel rey ajik krisna oleh-oleh itu apa yang dieksploitasi
Stemmed + Stopword	kaya eksploitasi tanya andrew susanto timothy ronald kalimasada gabriel rey ajik krisna oleh oleh eksploitasi

### WordCloud

Menurut (Turki & Roy, 2022) word cloud sangat penting dalam analisis teks eksploratif karena mampu mewakili representasi kata-kata yang sering muncul, sehingga membantu kita memahami tema atau kata-kata menonjol yang membuat suatu ucapan menjadi penuh kebencian. Dalam penerapannya pada kasus ini, *word cloud* kelas negatif didominasi oleh kata-kata bernada serangan, penilaian keras, dan ujaran kasar seperti “bodoh”, “goblok”, dan “anjing”, serta penyebutan figur seperti “timothy ronald”. Pola tersebut menunjukkan bahwa sentimen negatif tidak hanya berisi kritik terhadap sebuah konten, tetapi juga dapat mengarah pada

komentar personal dan kata-kata bernada agresif.

Sebaliknya, pada kelas netral dan positif, *word cloud* menunjukkan pola interaksi yang sangat berbeda. Kelas netral lebih menonjolkan penyebutan tokoh/entitas seperti “timothy” dan “timothy ronald”, serta kata-kata informatif seperti “investasi”, yang mengindikasikan bahwa komentar netral cenderung bersifat informatif dan membahas konteks tanpa ekspresi evaluasi emosi yang kuat. Sementara itu, *word cloud* kelas positif menampilkan kata-kata apresiatif seperti “good” dan “love” serta kata-kata yang terkait dengan kepercayaan. Hal ini membuktikan bahwa sentimen positif sering kali muncul sebagai bentuk dukungan, kepuasan, atau keyakinan terhadap informasi tertentu, seperti kelas crypto yang sedang ramai diperbincangkan.

### Penggunaan FastText Indonesia (cc.id.300.vec)

FastText memiliki keunggulan utama dibanding Word2Vec klasik karena memanfaatkan subword information (n-gram karakter), sehingga model tetap bisa membentuk representasi vektor meskipun kata yang muncul tidak persis sama dengan yang ada di kamus. Dengan mekanisme ini, kata tidak baku, salah ketik, atau bentuk ekspresif seperti “baguss”, “baguus”, “manteppp” cenderung masih mendapatkan vektor yang relevan, sekaligus lebih efektif menangani variasi morfologi bahasa Indonesia yang kaya imbuhan, misalnya “membantu”, “bantuan”, dan “dibantu”, karena kata-kata tersebut berbagi pola subword yang mirip.

### Model (SVM + SMOTE)

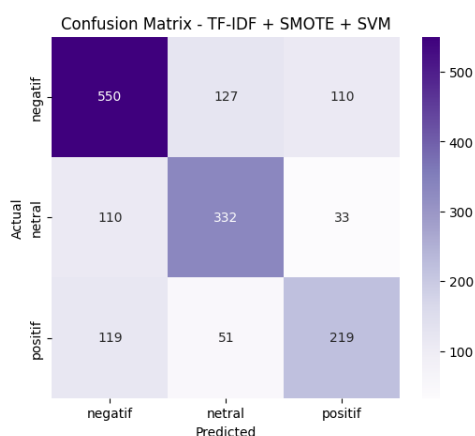
Konfigurasi model svm pada penelitian ini menggunakan representasi fitur

*TF-IDF* dengan  $\text{max\_features} = 20000$  dan  $\text{ngram\_range} = (1,2)$  untuk menangkap informasi kata tunggal maupun frasa dua kata, kemudian diklasifikasikan menggunakan SVM kernel linear (Linear SVC). Untuk mengatasi ketidakseimbangan kelas, diterapkan SMOTE pada data latih, yaitu teknik *oversampling* yang membuat sampel sintetis pada kelas minoritas agar distribusi kelas menjadi lebih seimbang, sehingga model tidak hanya “belajar” dari dominasi kelas mayoritas dan mampu mengenali pola pada kelas minoritas dengan lebih baik.

### Hasil Klasifikasi SVM + SMOTE

Pada data uji, model SVM yang menggunakan representasi TF-IDF dan penanganan imbalance dengan SMOTE menghasilkan *accuracy* sebesar 0,6669, F1-macro 0,6533, MCC 0,4753, Hamming Loss 0,3331, dan ROC-AUC micro sebesar 0,8521. Pada tingkat kelas, model memperoleh F1-score 0,70 untuk kelas negatif, 0,67 untuk kelas netral, dan 0,58 untuk kelas positif.

### Confusion Matrix SVM



Gambar 3. *Confusion Matrix SVM*

Berdasarkan confusion matrix, model mampu mengklasifikasikan dengan benar

550 dari 787 data negatif, 332 dari 475 data netral, dan 219 dari 389 data positif. Kesalahan prediksi terbesar terjadi pada kelas positif yang cukup sering diprediksi sebagai negatif, yaitu sebanyak 119 data. Temuan ini menunjukkan bahwa model SVM memiliki kinerja yang cukup seimbang pada kelas negatif dan netral, tetapi masih mengalami kesulitan dalam mengenali kelas positif.

### Model (BiLSTM + FastText + Focal Loss)

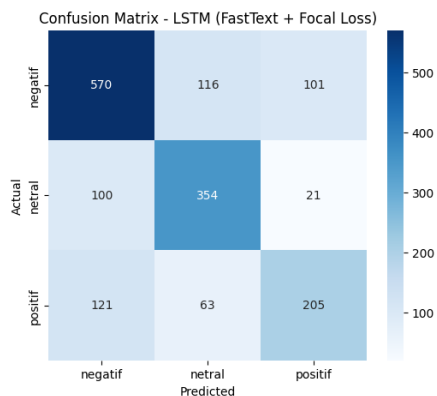
Model deep learning yang diterapkan menggunakan arsitektur Bidirectional LSTM dengan embedding FastText 300 dimensi, serta dimodifikasi untuk meningkatkan generalisasi melalui penambahan layer Spatial-Dropout1D(0.25) dan GlobalMaxPooling1D. Konfigurasi pelatihan memanfaatkan loss function Focal Loss dan optimizer Adam dengan learning rate awal  $1e-3$ , yang didukung oleh mekanisme ReduceLROnPlateau serta EarlyStopping untuk efisiensi. Selama proses pelatihan, akurasi training tercatat meningkat hingga mencapai angka 0,86 pada epoch ke-5, sementara akurasi validasi bergerak stabil pada kisaran 0,66–0,69 dengan puncak capaian sekitar 0,685; adapun validation loss yang sempat membaik pada epoch ke-2 namun kemudian perlahan naik memicu sistem untuk menurunkan learning rate dan mengaktifkan EarlyStopping guna mencegah pelatihan berjalan terlalu lama tanpa perbaikan performa.

### Hasil Klasifikasi LSTM

Model BiLSTM yang menggunakan *embedding FastText* dan *Focal Loss* menghasilkan *accuracy* sebesar 0,6838, F1-macro 0,6658, MCC 0,5008, Hamming Loss 0,3162, dan ROC-AUC micro sebesar 0,8525. Pada tingkat kelas, model

memperoleh F1-score 0,72 untuk kelas negatif, 0,70 untuk kelas netral, dan 0,57 untuk kelas positif.

### Confusion Matrix LSTM



Gambar 4. *Confusion Matrix LSTM*

Berdasarkan confusion matrix LSTM (FastText + Focal Loss) pada gambar, model mampu mengklasifikasikan kelas Negatif dengan benar sebanyak 570 dari 787 data, namun masih terjadi kesalahan ketika sentimen Negatif diprediksi sebagai Netral (116) maupun Positif (101). Pada kelas Netral, prediksi benar mencapai 354 dari 475 data, dengan kesalahan yang paling dominan berupa Netral yang diprediksi sebagai Negatif (100), sedangkan Netral yang dianggap Positif (21) relatif lebih kecil. Sementara itu, performa pada kelas Positif masih lebih menantang karena hanya 205 dari 389 data yang terklasifikasi benar, dengan kesalahan utama berupa Positif yang salah diklasifikasikan menjadi Negatif (121) dan Positif menjadi Netral (63). Pola keseluruhan ini menunjukkan bahwa tantangan terbesar model adalah membedakan sentimen Positif dari Negatif atau Netral pada komentar media sosial yang cenderung ambigu, sarkastik, atau menggunakan bahasa informal. Dari (Nurmawati & Prayogi, 2021) menjelaskan bahwa

confusion matrix merupakan tabel yang digunakan untuk menyatakan jumlah data uji yang benar dan salah diklasifikasikan, dengan komponen utama berupa *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN). Metrik evaluasi yang dihitung dari *confusion matrix* meliputi akurasi, presisi, dan *recall*.

### Fungsi Focal Loss

Penggunaan *Focal Loss* bertujuan untuk mengatasi ketidakseimbangan kelas dengan cara mengurangi kontribusi sampel yang mudah (easy samples) dan meningkatkan fokus pada sampel sulit (hard samples). Hal ini membantu model tidak terlalu hafal memprediksi kelas mayoritas saja.

### Fungsi SpatialDropout1D dan GlobalMaxPooling1D

Penerapan *SpatialDropout1D* dan *GlobalMaxPooling1D* memiliki peran krusial dalam mengoptimalkan arsitektur model. *SpatialDropout1D* berfungsi melakukan dropout pada dimensi embedding secara konsisten di seluruh urutan untuk mengurangi co-adaptation antar fitur, sehingga model menjadi lebih tangguh terhadap variasi kata, terutama pada teks pendek bernoise seperti komentar. Selanjutnya, *GlobalMaxPooling1D* berguna untuk mengambil fitur terkuat dari keluaran *BiLSTM*, yang efektif menonjolkan sinyal paling informatif sekaligus mengurangi jumlah parameter dibandingkan penggunaan dense layer pada sekuens panjang. penggunaan kedua lapisan ini terbukti membantu mencegah *overfitting* pada dataset berukuran menengah, yang diindikasikan oleh perbedaan kurva train-validation yang tidak divergen secara ekstrem serta akurasi validasi yang relatif stabil pada kisaran 0,66–

0,69, menjadikan performa model lebih terkontrol.

### Perbandingan Model dan Hasil

#### Hasil Klasifikasi

Tabel 3. Tabel *Matrix* tiap Model

Topik & Model	Jumlah Data	Accuracy	Precision	F1-Macro	MCC	ROC-AUC	Hamming Loss
Influencer, LSTM (FastText + Focal)	8.252	0,6838	Negatif: 0.72 Netral: 0.66 Positif: 0.63	0,6658	0.500	0,8525	0,3162
Influencer, SVM (TF-IDF + SMOTE)	8.252	0,6669	Negatif: 0.71 Netral: 0.65 Positif: 0.60	0,6533	0.475	0,8521	0.3331
Aplikasi Dana, Random Forest	1.354	84%	84%	84%	-	-	-
Layanan Paylater, NB & Text Blob	405	NB: 91% TextBlob: 61%	-	-	-	-	-
Relokasi Ibu Kota, NB & KNN	800	NB: 82% KNN: 88%	NB: 86% KNN: 93%	-	-	-	-

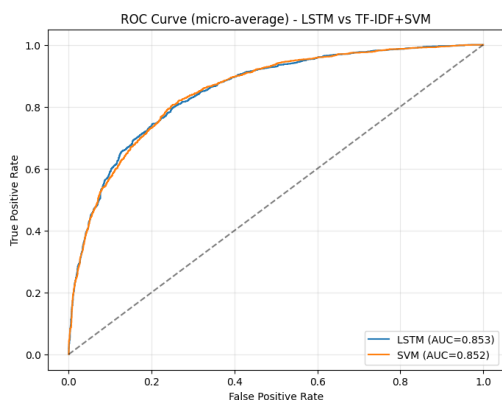
Jika dibandingkan dengan beberapa penelitian terdahulu, performa pada penelitian ini cenderung lebih rendah dari sisi accuracy. Penelitian pada ulasan aplikasi DANA dengan Random Forest melaporkan akurasi 84%, penelitian layanan paylater dengan Naive Bayes mencapai 91%, dan penelitian terkait relokasi Ibu Kota dengan KNN mencapai 88%. Namun, perbedaan ini tidak dapat dibaca secara langsung sebagai

kelemahan model pada penelitian ini, karena karakteristik data yang digunakan berbeda. Dataset komentar influencer di platform X cenderung lebih beragam, lebih informal, lebih noisy, dan mengandung ambiguitas yang lebih tinggi dibandingkan ulasan aplikasi atau topik yang polaritas sentimennya lebih eksplisit.

Dengan demikian, hasil penelitian ini justru menunjukkan bahwa pada domain komentar media sosial yang kompleks, baik pendekatan machine learning klasik maupun deep learning sama-sama memiliki potensi yang baik, tetapi keunggulan deep learning tidak selalu muncul secara drastis. Temuan ini mendukung pandangan bahwa pemilihan model tidak hanya ditentukan oleh kompleksitas algoritma, melainkan juga oleh karakter data, skema pelabelan, keseimbangan kelas, dan bentuk ekspresi bahasa pada korpus yang dianalisis.

Di sisi lain, hasil uji McNemar yang tidak signifikan menunjukkan bahwa meskipun BiLSTM memperoleh metrik yang sedikit lebih tinggi, SVM masih dapat dianggap sebagai alternatif yang layak. Ini berarti bahwa dalam konteks klasifikasi sentimen komentar influencer, peningkatan kompleksitas model belum tentu menghasilkan lompatan performa yang besar. Posisi temuan penelitian ini berada di antara dua kecenderungan: mendukung studi yang menunjukkan manfaat representasi konteks dan embedding pada teks informal, tetapi sekaligus menunjukkan bahwa model klasik berbasis TF-IDF masih sangat kompetitif untuk tugas klasifikasi sentimen berbahasa Indonesia

### Kurva ROC



Gambar 5. ROC

Kurva ROC micro-average pada Gambar 4 memperlihatkan kemampuan model dalam membedakan kelas-kelas sentimen berdasarkan probabilitas prediksi; pada grafik terlihat bahwa kedua kurva (BiLSTM dan SVM) berada jauh di atas garis diagonal (baseline tebakan acak) sehingga keduanya memiliki kemampuan diskriminasi yang baik, dengan nilai ROC-AUC (micro) yang sangat berdekatan yaitu BiLSTM sebesar 0,8525 ( $\approx 0,853$  pada grafik) dan SVM sebesar 0,8521 ( $\approx 0,852$  pada grafik), sehingga dari sisi kemampuan pemisahan kelas secara probabilistik keduanya hampir setara; dengan kata lain, meskipun BiLSTM sedikit lebih unggul pada metrik klasifikasi (Accuracy, F1-macro, MCC, dan Hamming Loss), SVM tetap menunjukkan daya diskriminasi probabilitas yang sebanding dengan BiLSTM.

### McNemar

Uji McNemar digunakan untuk mengevaluasi apakah perbedaan prediksi benar/salah dari dua model pada sampel yang sama bersifat signifikan secara statistik. Hasil perhitungan menunjukkan pasangan prediksi Yes/Yes = 938, Yes/No = 184, No/Yes = 163, dan No/No = 366, dengan p-value = 0,28298 yang lebih besar dari 0,05. Berdasarkan uji McNemar tersebut, tidak terdapat perbedaan yang signifikan secara statistik antara kinerja SVM (berbasis SMOTE) dan LSTM (berbasis Focal Loss) pada taraf signifikansi 5%, sehingga pendekatan Deep Learning yang telah dioptimasi mampu menyamai stabilitas Machine Learning klasik, namun tidak serta-merta mengunggulinya secara drastis karena model Deep Learning umumnya

membutuhkan data yang jauh lebih masif agar keunggulannya muncul secara konsisten.

### Pembahasan

Hasil penelitian menunjukkan bahwa model BiLSTM dengan *FastText* dan *Focal Loss* memperoleh kinerja sedikit lebih baik dibandingkan SVM berbasis *TF-IDF* dan SMOTE, terlihat dari nilai *accuracy*, F1-macro, MCC, dan *Hamming Loss* yang relatif lebih unggul. Temuan ini mengindikasikan bahwa pendekatan *deep learning* memiliki kelebihan dalam menangkap konteks urutan kata dan variasi bentuk kata pada komentar media sosial yang cenderung pendek, informal, serta penuh *noise*. Penggunaan *FastText* turut berkontribusi karena mampu merepresentasikan kata berdasarkan *subword*, sehingga lebih adaptif terhadap kata tidak baku, singkatan, maupun variasi penulisan yang sering muncul pada komentar di Platform X. Selain itu, penerapan *Focal Loss* membantu model untuk lebih memperhatikan sampel-sampel yang sulit dan kelas minoritas, sehingga performa klasifikasi menjadi lebih seimbang.

Di sisi lain, hasil uji McNemar yang menunjukkan tidak adanya perbedaan signifikan antara kedua model menegaskan bahwa peningkatan performa BiLSTM belum cukup kuat untuk menyatakan keunggulan yang meyakinkan atas SVM. Hal ini menunjukkan bahwa pada domain komentar pendek berbahasa Indonesia, model klasik seperti SVM masih sangat kompetitif, terutama ketika dipadukan dengan representasi *TF-IDF* yang efektif untuk menangkap kata-kata penting dan strategi SMOTE untuk mengatasi ketidakseimbangan kelas. Temuan ini juga

memperlihatkan bahwa pemilihan model terbaik tidak hanya ditentukan oleh kompleksitas algoritma, tetapi sangat dipengaruhi oleh karakteristik data, kualitas label, distribusi kelas, dan bentuk ekspresi bahasa pada korpus yang dianalisis. Dengan demikian, penelitian ini menempatkan BiLSTM sebagai model dengan performa terbaik secara metrik, tetapi SVM tetap layak dipertimbangkan sebagai alternatif yang lebih sederhana, khususnya ketika sumber daya komputasi terbatas atau ketika selisih performa antar model relatif kecil.

### KESIMPULAN

Hasil penelitian menunjukkan bahwa klasifikasi sentimen komentar pada Platform X terhadap influencer menghadapi ketidakseimbangan kelas yang tinggi, dengan dominasi sentimen negatif. *Preprocessing* teks, termasuk *demojize* dan normalisasi, serta penggunaan *FastText embedding* terbukti efektif dalam menangani noise dan variasi bahasa media sosial. Arsitektur BiLSTM dengan dukungan *SpatialDropout1D*, *GlobalMaxPooling1D*, dan *Focal Loss* memberikan performa terbaik dengan akurasi 0,68 dan F1-macro 0,66, sementara model SVM (Linear SVC) berbasis *TF-IDF* yang dipadukan dengan SMOTE menunjukkan performa yang kompetitif. Uji McNemar mengindikasikan perbedaan kinerja kedua pendekatan tidak signifikan secara statistik, menegaskan bahwa baik *deep learning* maupun *machine learning* klasik sama-sama efektif untuk dataset tidak seimbang. Temuan ini menegaskan bahwa strategi penanganan kelas minoritas yang tepat dapat menjaga kestabilan kinerja model, sekaligus menjadi referensi bagi pengembangan sistem analisis

sentimen yang lebih akurat pada komentar media sosial.

## DAFTAR PUSTAKA

- Arwindarti, T., Setiawan, E. I., & Imron, S. (2024). Klasifikasi Sentimen Opini Publik Pada Instagram Pemerintah Kabupaten Bojonegoro Menggunakan LSTM. *Teknika*, 13(1), 1–9. <https://doi.org/10.34148/teknika.v13i1.699>
- Asyaky, M. S., Al-Husaini, M., & Lukmana, H. H. (2025). Sentiment Analysis on Short Social Media Texts Using DistilBERT. *Journal of Computer Networks, Architecture and High Performance Computing*, 7(2), 524–533. <https://doi.org/10.47709/cnahpc.v7i2.5836>
- Brown, M. A., Gruen, A., Maldoff, G., Messing, S., Sanderson, Z., & Zimmer, M. (2025). Web scraping for research: Legal, ethical, institutional, and scientific considerations. *Big Data & Society*, 12(4), 20539517251381690. <https://doi.org/10.1177/20539517251381686>
- Derbentsev, V. D., Bezkorovainyi, V. S., Matviychuk, A. V, Pomazun, O. M., Hrabariev, A. V, & Hostryk, A. M. (2023). Sentiment analysis of electronic social media based on deep learning. *No. M3e2*, 163–175. <https://doi.org/10.5220/0011932300003432>
- Eliviani, R., & Wazaumi, D. D. (2024). Exploring sentiment trends: Deep learning analysis of social media reviews on Google Play Store by netizens. *International Journal of Advances in Data and Information Systems*, 5(1), 62–70. <https://doi.org/10.59395/ijadis.v5i1.131>
- Hadiprakoso, R. B., Setiawan, H., Yasa, R. N., & Girinoto. (2023). Text preprocessing for optimal accuracy in Indonesian sentiment analysis using a deep learning model with word embedding. *AIP Conference Proceedings*, 2680(1), 20050. <https://doi.org/10.1063/5.0126116>
- Hasan, M., Milawati, Mp., Darodjat, Mp., & DrTuti Khairani Harahap, Ma. (2021). *Makna Peran Media Dalam Komunikasi dan Pembelajaran | i MEDIA PEMBELAJARAN*.
- Irawan, T., Amelia, P. R., Afriandi, A., Jusiaty, A., Octaviani, V., Maryaningsih, M., & Narti, S. (2025). Edukasi Bentuk Kampanye Sosial Menggunakan Media Sosial Di SMA Negeri 8 Kota Bengkulu. *Jurnal Dehasen Mengabdi*, 4(1), 9–12. <https://doi.org/10.37676/jdm.v4i1.7784>
- Larasati, F. A., Ratnawati, D. E., & Hanggara, B. T. (2022). Analisis sentimen ulasan aplikasi Dana dengan metode Random Forest. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 6(9), 4305–4313.
- Perwira, R. I., Permadi, V. A., Purnamasari, D. I., & Agusdin, R. P. (2025). 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*, 11(1), 30–40.
- Prasetyo, S. D., Hilabi, S. S., & Nurapriani, F. (2023). Analisis sentimen relokasi Ibukota Nusantara menggunakan algoritma Naïve Bayes dan KNN. *Jurnal KomtekInfo*, 1–7. <https://doi.org/10.35134/komtekinfo.v10i1.330>

- 
- Putra, Y. I., Hakiki, M., Ridoh, A., Fauziah, Fadli, R., & Sundahry. (2022). *Konsep Interaksi Manusia Dan Komputer*. Penerbit Lakeisha.
- Safira, A., & Hasan, F. N. (2023). Analisis Sentimen Masyarakat Terhadap Paylater Menggunakan Metode Naive Bayes Classifier. *ZONasi: Jurnal Sistem Informasi*, 5(1), 59–70. <https://doi.org/10.31849/zn.v5i1.12856>
- Safitri, A., Firmansyah, I., Yani, F., Kurniawan, M. A., & Nuryamin, Y. (2025). ANALISIS SENTIMEN RESPON PENGGUNA CHAT GPT MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE. *Jurnal Inovasi Pendidikan Dan Teknologi Informasi (JIPTI)*, 6(2), 553–566. <https://doi.org/10.52060/jipti.v6i2.3810>
- Turki, T., & Roy, S. S. (2022). Novel hate speech detection using word cloud visualization and ensemble learning coupled with count vectorizer. *Applied Sciences*, 12(13), 6611. <https://doi.org/10.20473/jisebi.11.1.30-40>
- Wen, X., & Li, W. (2023). Time series prediction based on LSTM-attention-LSTM model. *IEEE Access*, 11, 48322–48331. <https://doi.org/10.1109/ACCESS.2023.3276628>
- Xu, Q. A., Chang, V., & Jayne, C. (2022). A systematic review of social media-based sentiment analysis: Emerging trends and challenges. *Decision Analytics Journal*, 3, 100073. <https://doi.org/10.1016/j.dajour.2022.100073>