Jurnal Riset Aplikasi Ilmu Data dan Sistem 2 (1), 34–54 DOI: 0000
Januari 2027

Analisis Perbandingan Machine Learning dan DistilBERT dalam Deteksi Tingkat Depresi pada Teks Media Sosial

Reinesa Eveniashari Purwasarani[1], Riskyana Dewi Intan Puspitasari[2,*], Elly Matul Imah[2]

[1]*Program Studi Sains Data, Universitas Negeri Surabaya, Indonesia*
[2]*Program Studi Kecerdasan Artifisial, Universitas Negeri Surabaya, Indonesia*

Abstrak

Depresi merupakan salah satu gangguan kesehatan mental yang dapat memengaruhi kondisi emosional, perilaku, dan kualitas hidup seseorang. Meningkatnya penggunaan media sosial mendorong individu untuk mengekspresikan kondisi psikologis melalui unggahan teks, sehingga membuka peluang pemanfaatan *Natural Language Processing* (NLP) untuk melakukan klasifikasi tingkat depresi secara otomatis. Penelitian ini bertujuan membandingkan performa beberapa algoritma klasifikasi dalam mendeteksi tingkat depresi berdasarkan teks media sosial. Dataset yang digunakan merupakan dataset depresi berbahasa Inggris yang terdiri atas tiga kelas, yaitu *not depression*, *moderate*, dan *severe*. Tahapan penelitian meliputi *preprocessing* teks, ekstraksi fitur menggunakan TF-IDF, NRC Emotion Lexicon, LIWC, serta Word2Vec (CBOW dan Skip-Gram), kemudian dilakukan klasifikasi menggunakan *Support Vector Machine* (SVM), Random Forest, dan Na"ive Bayes. Selain itu, penelitian juga mengimplementasikan DistilBERT sebagai pendekatan berbasis Transformer tanpa memerlukan proses ekstraksi fitur secara eksplisit. Evaluasi model dilakukan menggunakan metrik *accuracy*, *precision*, *recall*, *F1-score*, dan *Area Under Curve* (AUC). Hasil pengujian menunjukkan bahwa DistilBERT memperoleh performa terbaik dengan *accuracy* sebesar 64%, *precision* sebesar 62%, *recall* sebesar 61%, *F1-score* sebesar 61%, dan AUC sebesar 0,74. Sementara itu, pada pendekatan *machine learning* klasik, kombinasi SVM dengan TF-IDF dan NRC menghasilkan performa terbaik dengan *accuracy* sebesar 59%, *F1-score* sebesar 57%, dan AUC sebesar 0,74. Model terbaik kemudian diterapkan pada data hasil *scraping* media sosial dan menunjukkan dominasi prediksi pada kelas *severe*. Hasil penelitian menunjukkan bahwa DistilBERT memberikan performa klasifikasi yang lebih baik dibandingkan metode *machine learning* berbasis fitur tradisional pada dataset yang digunakan.

Kata Kunci *Depression; DistilBERT; Machine Learning; Natural Language Processing; Social Media; Word2Vec*

Abstract

Depression is a mental health disorder that can affect an individual's emotional state, behavior, and quality of life. The increasing use of social media encourages individuals to express their psychological conditions through text posts, creating opportunities for the use of *Natural Language Processing* (NLP) to automatically classify depression severity. This study aims to compare the performance of several classification algorithms in detecting depression severity based on social

*Corresponding author. Email: reinesa.22021@mhs.unesa.ac.id or riskyanapuspitasari@unesa.ac.id or elly-matul@unesa.ac.id.

Received 15 Juli 2026; Revised 23 September 2026; Accepted 20 November 2026

media text. The dataset used is an English-language depression dataset consisting of three classes, namely *not depression*, *moderate*, and *severe*. The research process includes text *preprocessing*, feature extraction using TF-IDF, NRC Emotion Lexicon, LIWC, and Word2Vec (CBOW and Skip-Gram), followed by classification using *Support Vector Machine* (SVM), Random Forest, and Na"ive Bayes. In addition, DistilBERT was implemented as a Transformer-based approach without requiring explicit feature extraction. Model evaluation was performed using *accuracy*, *precision*, *recall*, *F1-score*, and *Area Under Curve* (AUC). The results show that DistilBERT achieved the best performance with an *accuracy* of 64%, *precision* of 62%, *recall* of 61%, *F1-score* of 61%, and an AUC of 0.74. Meanwhile, among the classical *machine learning* approaches, the combination of SVM with TF-IDF and NRC achieved the best performance, with an *accuracy* of 59%, *F1-score* of 57%, and an AUC of 0.74. The best-performing model was then applied to scraped social media data and showed a dominance of predictions in the *severe* class. The results indicate that DistilBERT provides better classification performance than traditional feature-based *machine learning* methods on the dataset used in this study.

Keywords *Depresi; DistilBERT; Natural Language Processing; Machine Learning; Media Sosial; Word2Vec*

1 Pendahuluan

Dampak depresi tidak hanya berkaitan dengan kondisi emosional seseorang, tetapi juga dapat memengaruhi aktivitas sehari-hari dan interaksinya dengan lingkungan sosial. World Health Organization (WHO) melaporkan bahwa depresi menjadi salah satu permasalahan kesehatan global yang berdampak pada kualitas hidup dan produktivitas individu. World Health Organization (2025). Gejala yang berkaitan dengan depresi dapat mencakup suasana hati negatif yang berlangsung terus-menerus, berkurangnya minat terhadap aktivitas, penurunan energi, serta kesulitan berkonsentrasi. Kemunculan dan perkembangan kondisi tersebut dapat berkaitan dengan berbagai aspek, termasuk faktor biologis, psikologis, lingkungan, dan sosial Ratajczak dkk. (2024). Pemahaman terhadap gejala dan faktor yang berkaitan dengan depresi diperlukan agar kondisi tersebut dapat dikaji secara lebih tepat dengan mempertimbangkan berbagai aspek yang memengaruhinya.

Unggahan yang dibagikan pengguna di media sosial dapat memuat pengalaman pribadi, pandangan, serta ekspresi emosi yang disampaikan melalui tulisan. Platform seperti Reddit dan X memungkinkan pengguna membagikan pengalaman serta pandangan mereka melalui unggahan tertulis. Data tersebut berpotensi memberikan informasi mengenai kondisi psikologis yang tercermin dalam penggunaan bahasa, meskipun isi unggahan tidak dapat secara langsung dijadikan dasar untuk menetapkan diagnosis klinis Aldkheel dan Zhou (2024). Sejumlah penelitian juga mengkaji hubungan antara karakteristik bahasa di media sosial dan indikasi depresi sebagai salah satu pendekatan untuk mendukung identifikasi awal kondisi kesehatan mental Benjachairat dkk. (2024).

Analisis teks media sosial dapat dilakukan dengan pendekatan Natural Language Processing (NLP), yang membantu mengolah informasi dalam tulisan menjadi representasi yang dapat digunakan pada proses klasifikasi Ghosal dan Jain (2022). Pendekatan ini telah digunakan dalam penelitian deteksi depresi berbasis teks untuk mempelajari pola linguistik yang berkaitan dengan kategori yang diteliti Aldhyani dkk. (2022). Berbagai algoritma klasifikasi, termasuk Support Vector Machine (SVM), Random Forest, dan Naïve Bayes, dapat digunakan untuk mempelajari pola dari representasi teks. Di sisi lain, DistilBERT merupakan model berbasis Transformer

yang menghasilkan representasi kontekstual sehingga makna suatu kata dapat dipertimbangkan berdasarkan konteks penggunaannya dalam kalimat Sanh dkk. (2020). Perbedaan karakteristik tersebut menjadi dasar untuk membandingkan pendekatan klasifikasi berbasis fitur dengan pendekatan berbasis representasi kontekstual.

Setiap metode representasi teks memiliki cara yang berbeda dalam menyajikan informasi dari sebuah dokumen. TF-IDF, misalnya, memberikan bobot berdasarkan kemunculan kata dalam dokumen dan keseluruhan korpus, tetapi tidak secara langsung merepresentasikan konteks emosional maupun hubungan makna antarkata Qaiser dan Ali (2018). Untuk melengkapi informasi tersebut, representasi berbasis leksikon emosi seperti NRC Emotion Lexicon dan pendekatan psikolinguistik seperti Linguistic Inquiry and Word Count (LIWC) dapat digunakan untuk menggambarkan aspek bahasa yang berbeda. Word2Vec menyediakan representasi kata berbentuk vektor, sedangkan model Transformer seperti DistilBERT menghasilkan representasi yang mempertimbangkan konteks. Penggunaan pendekatan representasi yang beragam telah dibahas dalam penelitian sebelumnya, termasuk kajian mengenai representasi teks untuk analisis bahasa dan emosi Troiano dkk. (2023). Namun, perbedaan karakteristik antarrepresentasi tersebut perlu dievaluasi dalam konteks tugas dan dataset yang digunakan agar kinerja masing-masing pendekatan dapat dibandingkan secara terukur.

Berdasarkan latar belakang tersebut, penelitian ini membandingkan beberapa pendekatan representasi teks dan algoritma klasifikasi untuk mendeteksi tingkat depresi pada data teks media sosial. Representasi yang dikaji meliputi TF-IDF, NRC Emotion Lexicon, LIWC, dan Word2Vec, sedangkan pendekatan berbasis Transformer dievaluasi menggunakan DistilBERT. Pada pendekatan machine learning klasik, pengujian dilakukan menggunakan SVM, Random Forest, dan Naïve Bayes dengan variasi representasi fitur yang digunakan. Kinerja setiap pendekatan dibandingkan menggunakan accuracy, precision, recall, F1-score, dan Area Under the Curve (AUC). Perbandingan ini bertujuan mengidentifikasi pendekatan yang memberikan kinerja klasifikasi paling baik pada dataset penelitian.

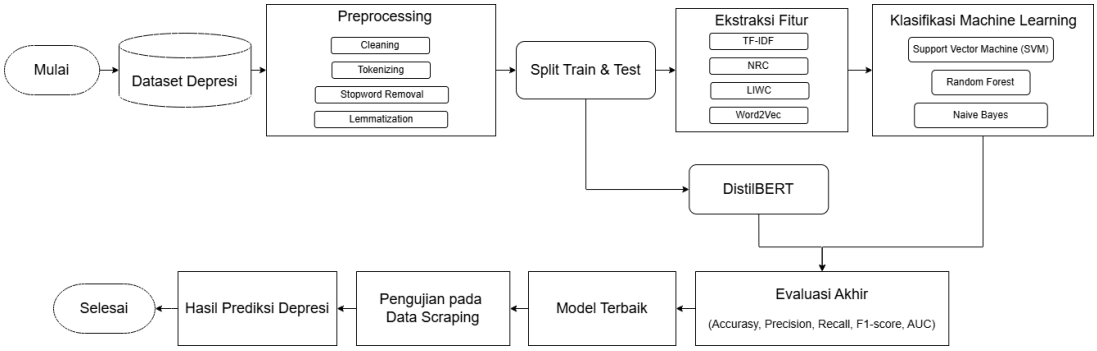
2 Metode

Penelitian ini menggunakan pendekatan *Natural Language Processing* (NLP) untuk mengklasifikasikan tingkat depresi berdasarkan data teks media sosial. Tahapan penelitian meliputi pengumpulan dataset, pra-pemrosesan teks, representasi teks, pelatihan model, dan evaluasi kinerja. Pada algoritma *machine learning* klasik, representasi teks dibentuk menggunakan beberapa teknik ekstraksi fitur. Sementara itu, DistilBERT memanfaatkan representasi kontekstual yang diperoleh melalui arsitektur Transformer. Seluruh model dievaluasi menggunakan metrik *accuracy*, *precision*, *recall*, *F1-score*, dan *Area Under the Curve* (AUC). Alur penelitian secara keseluruhan disajikan pada Gambar 2.

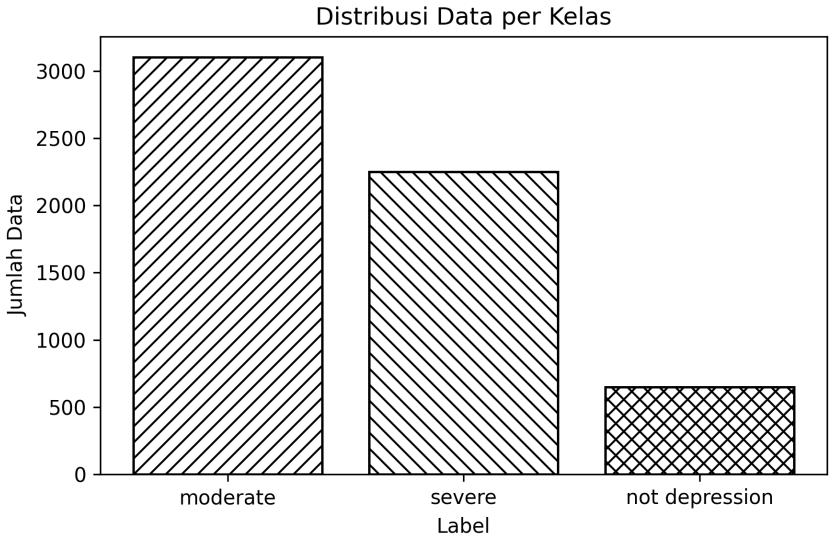
2.1 Dataset

Penelitian ini menggunakan dataset *Depression Detection in Social Media Text* yang berasal dari kompetisi LT-EDI 2022 Poswiata dan Perełkiewicz (2022). Dataset tersebut berisi unggahan pengguna Reddit berbahasa Inggris yang telah dianotasi ke dalam tiga kategori tingkat depresi, yaitu *not depression*, *moderate depression*, dan *severe depression*. Untuk meningkatkan kualitas data, penelitian ini menggunakan versi dataset yang telah dipraproses oleh Poswiata dan Perełkiewicz dengan menghapus data duplikat serta menyusun kembali data pelatihan sehingga diperoleh sebanyak 6.006 data unik yang siap digunakan pada proses klasifikasi.

Setiap data terdiri atas satu teks unggahan beserta label tingkat depresi yang telah diberikan melalui proses anotasi. Seluruh data digunakan sesuai dengan label aslinya tanpa melakukan pelabelan ulang (*re-labeling*). Sebelum proses pelatihan model, dataset dibagi menjadi data latih dan data uji menggunakan rasio 80:20. Pembagian ini bertujuan agar model dapat mempelajari pola bahasa pada data latih dan dievaluasi menggunakan data yang belum pernah dilihat sebelumnya sehingga hasil pengujian dapat menggambarkan kemampuan generalisasi model secara lebih objektif. Dataset yang digunakan memiliki karakteristik bahasa media sosial, seperti penggunaan bahasa informal, singkatan, emoji, serta ekspresi emosional yang beragam. Karakteristik tersebut menjadikan dataset ini sesuai untuk mengevaluasi efektivitas berbagai representasi fitur teks dan algoritma klasifikasi dalam mendeteksi tingkat depresi berdasarkan pola linguistik yang terkandung pada unggahan pengguna media sosial.



Gambar 1: Alur Penelitian



Gambar 2: Dataset Depresi

Tabel 1: Contoh Data pada Dataset Depresi

Tweet	**Label**
feeling super lonely got ghosted one person meant anything im alone without answer hurt feeling need someone sit	Moderate
way mentally commit Suicide like could meditate way could end life conciousness body death meditation cause would decent way go	Severe
truly wish someone would shoot hope die thinking life past like always there getting away except death	Not Depression

2.2 Preprocessing

Tahap *preprocessing* bertujuan untuk meningkatkan kualitas data teks sebelum dilakukan proses ekstraksi fitur dan klasifikasi. Data media sosial umumnya mengandung berbagai elemen yang tidak relevan, seperti URL, mention, tanda baca, karakter khusus, serta variasi penulisan kata yang dapat memengaruhi performa model. Oleh karena itu, dilakukan serangkaian tahapan *preprocessing* untuk menghasilkan representasi teks yang lebih bersih, konsisten, dan siap diproses oleh algoritma machine learning.

Tahapan *preprocessing* meliputi *text cleaning*, *case folding*, *tokenization*, *stopword removal*, dan *lemmatization*. Pada tahap *text cleaning*, URL, mention, hashtag, angka, dan karakter khusus dihapus, sedangkan emoji dipertahankan karena dapat merepresentasikan ekspresi emosional pengguna. Selanjutnya, seluruh teks diubah menjadi huruf kecil (*case folding*) sebelum dipecah menjadi token melalui proses *tokenization*. Tahap *stopword removal* digunakan untuk menghilangkan kata-kata umum yang kurang informatif, sedangkan *lemmatization* mengubah setiap kata ke bentuk dasarnya sehingga variasi morfologi dapat dikurangi tanpa menghilangkan makna semantik. Hasil akhir dari tahapan ini berupa teks yang lebih terstruktur dan siap digunakan pada proses ekstraksi fitur.

Tabel 2: Contoh Hasil Tahapan Preprocessing

Tahap	**Contoh Hasil**
Original	*at this point just genuinely curious what sort of shit throws my way . . . stay tuned*
Tokenization	[at, this, point, just, genuinely, curious, . . . , stay, tuned]
Stopword Removal	[point, genuinely, curious, sort, shit, throws, way, . . . , stay, tuned]
Lemmatization	[point, genuinely, curious, sort, shit, throw, way, . . . , stay, tuned]
Final Text	*point genuinely curious sort shit throw way actually funny used end every previous year . . . stay tuned*

2.3 Ekstraksi Fitur

Tahap ekstraksi fitur bertujuan mengubah data teks hasil *preprocessing* menjadi representasi numerik yang dapat digunakan oleh algoritma klasifikasi. Representasi fitur yang berbeda memungkinkan model mengenali pola frekuensi kata, karakteristik emosional, aspek psikologis, serta hubungan semantik yang terkandung dalam teks. Pada penelitian ini, digunakan empat metode representasi fitur, yaitu TF-IDF, NRC Emotion Lexicon, Linguistic Inquiry and Word Count (LIWC), dan Word2Vec. Sementara itu, DistilBERT menggunakan representasi kontekstual yang diperoleh melalui arsitektur Transformer.

2.3.1 Term Frequency-Inverse Document Frequency (TF-IDF)

Term Frequency–Inverse Document Frequency (TF-IDF) merupakan metode pembobotan kata yang banyak digunakan pada klasifikasi teks karena mampu merepresentasikan tingkat kepentingan suatu kata dalam dokumen. TF-IDF memberikan bobot tinggi pada kata yang sering muncul pada suatu dokumen namun jarang ditemukan pada keseluruhan korpus sehingga kata-kata yang bersifat informatif akan lebih menonjol dibandingkan kata-kata umum (*stopwords*) Saputra dkk. (2021). Nilai TF-IDF dihitung menggunakan persamaan (1).

$$TFIDF(t,d) = TF(t,d) \times \log\left(\frac{N}{DF(t)}\right), \tag{1}$$

dengan $TF(t,d)$ menyatakan frekuensi kemunculan kata t pada dokumen d, $DF(t)$ merupakan jumlah dokumen yang mengandung kata t, sedangkan N adalah jumlah seluruh dokumen pada korpus. Semakin tinggi nilai TF-IDF suatu kata, maka semakin besar kontribusinya dalam membedakan suatu dokumen dari dokumen lainnya. Pada penelitian ini, representasi TF-IDF digunakan sebagai fitur utama pada algoritma Support Vector Machine (SVM), Random Forest, dan Naïve Bayes untuk membentuk representasi statistik dari setiap dokumen sebelum proses klasifikasi dilakukan.

2.3.2 NRC Emotion Lexicon

NRC Emotion Lexicon merupakan kamus leksikal yang memetakan setiap kata ke dalam delapan kategori emosi dasar, yaitu *anger*, *fear*, *sadness*, *joy*, *trust*, *anticipation*, *surprise*, dan *disgust*, serta dua kategori sentimen positif dan negatif Mohammad dan Turney (2013). Pendekatan ini memungkinkan teks direpresentasikan tidak hanya berdasarkan frekuensi kata, tetapi juga berdasarkan kandungan emosi yang terkandung di dalamnya sehingga lebih sesuai untuk analisis kesehatan mental Acheampong dkk. (2020).

$$Score_e = \sum_{i=1}^{n} I(w_i \in e), \tag{2}$$

dengan $Score_e$ merupakan skor untuk kategori emosi e, sedangkan $I(w_i \in e)$ bernilai 1 apabila kata ke-i termasuk ke dalam kategori emosi tersebut dan bernilai 0 apabila tidak termasuk. Nilai skor diperoleh dari akumulasi seluruh kata yang terdapat pada suatu dokumen Syafiqah dkk. (2025). Pada penelitian ini, skor emosi dari NRC Emotion Lexicon digunakan sebagai fitur tambahan untuk memperkaya representasi teks sehingga model mampu mengenali kecenderungan ekspresi emosional yang berkaitan dengan depresi.

2.3.3 LIWC

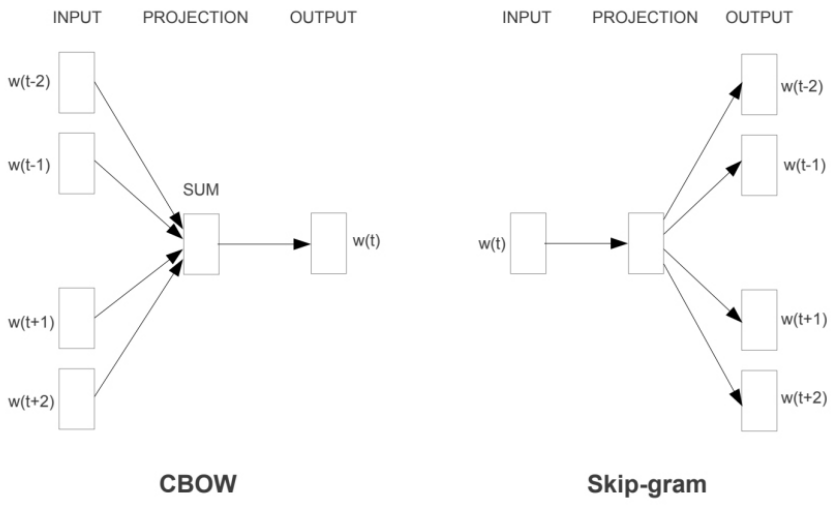
Linguistic Inquiry and Word Count (LIWC) merupakan kamus psikolinguistik yang mengelompokkan kata ke dalam berbagai kategori psikologis, seperti *negative emotion*, *sadness*, *anxiety*, *cognitive process*, dan penggunaan kata ganti orang pertama Tausczik dan Pennebaker (2010). Berbeda dengan pendekatan berbasis frekuensi kata, LIWC berfokus pada karakteristik psikologis yang tercermin melalui pola penggunaan bahasa sehingga banyak dimanfaatkan dalam penelitian kesehatan mental berbasis teks Zhao dkk. (2016).

$$LIWC_c = \frac{N_c}{N} \times 100, \tag{3}$$

dengan N_c merupakan jumlah kata yang termasuk ke dalam kategori psikologis tertentu dan N menyatakan jumlah seluruh kata dalam dokumen. Persentase tersebut menunjukkan proporsi penggunaan kata pada setiap kategori psikologis. Dalam penelitian ini, fitur LIWC digunakan untuk menangkap karakteristik psikologis pengguna media sosial sehingga dapat melengkapi informasi statistik dari TF-IDF dan informasi emosional dari NRC Emotion Lexicon.

2.3.4 Word2Vec

Word2Vec merupakan metode *word embedding* yang merepresentasikan setiap kata ke dalam bentuk vektor kontinu berdasarkan konteks kemunculannya dalam kalimat. Representasi tersebut memungkinkan kata-kata yang memiliki makna serupa berada pada posisi yang berdekatan dalam ruang vektor sehingga mampu menangkap hubungan semantik antar kata Rong (2016). Pada penelitian ini digunakan dua arsitektur Word2Vec, yaitu Continuous Bag of Words (CBOW) dan Skip-Gram.



Gambar 3: Arsitektur Continuous Bag of Words (CBOW) dan Skip-Gram

Arsitektur CBOW memprediksi kata target berdasarkan beberapa kata konteks di sekitarnya. Dengan memanfaatkan informasi konteks, model berusaha menghasilkan representasi kata yang mampu menangkap hubungan semantik antar kata Alin dan Utami (2025). Fungsi objektif CBOW ditunjukkan pada Persamaan (4).

$$J_{CBOW} = \frac{1}{T}\sum_{t=1}^{T} \log P(w_t|Context(w_t)), \tag{4}$$

dengan w_t merupakan kata target dan $Context(w_t)$ adalah kumpulan kata di sekitar kata target. Model dilatih untuk memaksimalkan probabilitas munculnya kata target berdasarkan konteks yang diberikan sehingga representasi kata yang dihasilkan mampu menggambarkan hubungan semantik antar kata.

Berbeda dengan CBOW, arsitektur Skip-Gram menggunakan satu kata target untuk memprediksi beberapa kata konteks di sekitarnya. Pendekatan ini umumnya memberikan representasi yang lebih baik untuk kata-kata yang jarang muncul pada korpus. Fungsi objektif Skip-Gram ditunjukkan pada Persamaan (5).

$$J_{SG} = \frac{1}{T}\sum_{t=1}^{T} \sum_{-c \leqslant j \leqslant c,\, j \neq 0} \log P(w_{t+j}|w_t), \tag{5}$$

dengan c merupakan ukuran *context window*. Pendekatan ini mampu menghasilkan representasi kata yang lebih baik terutama pada kata-kata yang jarang muncul karena setiap kata target digunakan untuk memprediksi beberapa kata konteks di sekitarnya. Pada penelitian ini, representasi Word2Vec yang dihasilkan dari arsitektur CBOW maupun Skip-Gram digunakan sebagai fitur masukan pada algoritma klasifikasi untuk membandingkan performanya terhadap representasi berbasis TF-IDF dan fitur leksikal.

2.4 Metode Klasifikasi

Tahap klasifikasi bertujuan membangun model yang mampu mengidentifikasi tingkat depresi berdasarkan data teks yang telah direpresentasikan sesuai dengan karakteristik masing-masing metode. Pada penelitian ini digunakan tiga algoritma *machine learning*, yaitu Support Vector Machine (SVM), Random Forest, dan Naïve Bayes, serta satu model berbasis *Transformer*, yaitu DistilBERT. Seluruh model dilatih dan dievaluasi menggunakan pembagian data latih dan data uji yang sama agar perbandingan performanya dapat dilakukan secara konsisten berdasarkan metrik evaluasi yang digunakan.

2.4.1 Support Vector Machine (SVM)

Support Vector Machine (SVM) merupakan algoritma klasifikasi yang bekerja dengan mencari *hyperplane* optimal untuk memisahkan data dari kelas yang berbeda dengan margin maksimum Ovirianti dkk. (2022). SVM banyak digunakan pada klasifikasi teks karena mampu menangani data berdimensi tinggi serta tetap memberikan performa yang baik meskipun jumlah fitur sangat besar Tabany dan Gueffal (2024).

Fungsi keputusan SVM dirumuskan sebagai berikut.

$$f(x) = w^T x + b, \tag{6}$$

dengan w merupakan vektor bobot, x adalah vektor fitur, dan b merupakan nilai bias. Tujuan utama SVM adalah menentukan nilai w dan b sehingga diperoleh margin maksimum antar kelas. Proses optimasi SVM dilakukan dengan meminimalkan fungsi berikut.

$$\min \frac{1}{2}||w||^2, \tag{7}$$

dengan kendala

$$y_i(w^T x_i + b) \geqslant 1, ; \forall i, \tag{8}$$

di mana y_i merupakan label kelas dari data ke-i. Pada penelitian ini, SVM diterapkan menggunakan berbagai representasi fitur, meliputi TF-IDF, NRC Emotion Lexicon, LIWC, kombinasi beberapa fitur tersebut, serta Word2Vec dengan arsitektur CBOW dan Skip-Gram. Pengujian ini dilakukan untuk membandingkan pengaruh setiap representasi fitur terhadap performa klasifikasi tingkat depresi. Pada proses pelatihan, SVM menggunakan *linear kernel* dengan parameter *class_ weight=balanced* untuk menangani ketidakseimbangan jumlah data antar kelas. Parameter *probability=True* digunakan untuk menghasilkan probabilitas prediksi yang selanjutnya digunakan dalam perhitungan AUC multiclass, sedangkan *random_ state=42* digunakan untuk menjaga konsistensi hasil eksperimen.

2.4.2 Random Forest

Random Forest merupakan metode *ensemble learning* yang membangun sejumlah pohon keputusan (*decision tree*) dari subset data dan subset fitur yang dipilih secara acak. Prediksi akhir diperoleh melalui mekanisme *majority voting*, sehingga mampu mengurangi risiko *overfitting* dibandingkan penggunaan satu pohon keputusan Bader dkk. (2024).

Prediksi Random Forest dirumuskan sebagai

$$\hat{y} = \text{mode}\left(T_1(x), T_2(x), \ldots, T_n(x)\right), \tag{9}$$

dengan $T_i(x)$ merupakan hasil prediksi pohon keputusan ke-i, sedangkan n menyatakan jumlah pohon yang dibangun.

Semakin banyak pohon keputusan yang digunakan, model cenderung menghasilkan prediksi yang lebih stabil dan memiliki kemampuan generalisasi yang lebih baik. Dalam penelitian ini, Random Forest digunakan untuk mengevaluasi kemampuan metode *ensemble* dalam mendeteksi depresi berdasarkan berbagai representasi fitur teks.

Pada proses pelatihan, Random Forest menggunakan 200 pohon keputusan yang ditetapkan melalui parameter *n_ estimators*=200. Parameter *class_ weight=balanced* digunakan untuk memberikan bobot yang lebih besar pada kelas dengan jumlah data yang lebih sedikit, sehingga pengaruh ketidakseimbangan kelas dapat dikurangi. Parameter *random_ state=42* digunakan untuk menjaga konsistensi hasil eksperimen, sedangkan *n_jobs=-1* digunakan untuk memanfaatkan seluruh inti prosesor yang tersedia selama proses pelatihan.

2.4.3 Naïve Bayes

Naïve Bayes merupakan algoritma klasifikasi probabilistik yang didasarkan pada Teorema Bayes dengan asumsi bahwa setiap fitur bersifat independen terhadap fitur lainnya. Walaupun asumsi tersebut sederhana, Naïve Bayes terbukti efektif untuk klasifikasi teks karena mampu menangani data berdimensi tinggi dengan proses komputasi yang relatif cepat Anugerah dkk. (2024).

Probabilitas posterior dihitung menggunakan persamaan berikut.

$$P(C|X) = \frac{P(X|C)P(C)}{P(X)}, \tag{10}$$

dengan C merupakan kelas dan X adalah vektor fitur.
Model kemudian memilih kelas dengan probabilitas posterior terbesar.

$$\hat{C} = \arg\max_{C} P(C|X). \tag{11}$$

Pada penelitian ini, Naïve Bayes digunakan sebagai model pembanding karena memiliki kompleksitas komputasi yang rendah namun telah banyak digunakan pada berbagai penelitian klasifikasi teks kesehatan mental. Untuk representasi fitur berbasis TF-IDF dan kombinasi fitur TF-IDF dengan NRC Emotion Lexicon maupun LIWC, digunakan Multinomial Naïve Bayes melalui implementasi *MultinomialNB()*. Sementara itu, pada representasi fitur Word2Vec digunakan Gaussian Naïve Bayes melalui `GaussianNB()` karena fitur yang dihasilkan berupa vektor kontinu.

Pada proses pelatihan, model *Multinomial Naïve Bayes* menggunakan parameter bawaan *alpha=1.0*, sedangkan model *Gaussian Naïve Bayes* menggunakan parameter bawaan dari implementasi `GaussianNB()`. Kedua model dilatih menggunakan data pelatihan dan dievaluasi pada data pengujian yang sama dengan algoritma lainnya.

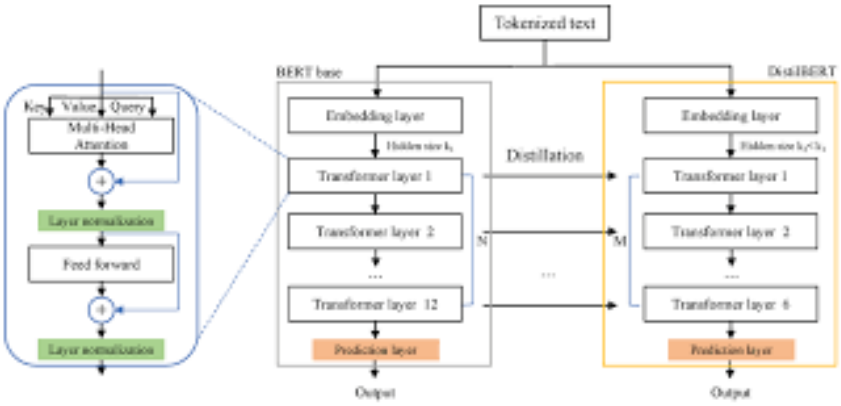
2.4.4 DistilBERT

DistilBERT merupakan model bahasa berbasis Transformer yang dikembangkan melalui pendekatan *knowledge distillation* dari BERT sehingga memiliki jumlah parameter lebih sedikit namun tetap mempertahankan sebagian besar kemampuan representasi kontekstualnya. Dibandingkan BERT-base, DistilBERT menggunakan enam lapisan encoder sehingga proses pelatihan maupun inferensi menjadi lebih cepat dengan kebutuhan memori yang lebih rendah Devlin dkk. (2019).

Mekanisme utama DistilBERT memanfaatkan *self-attention* untuk menangkap hubungan antarkata dalam suatu kalimat. Proses tersebut dirumuskan pada persamaan berikut:

$$Attention(Q, K, V) = Softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{12}$$

dengan Q, K, dan V masing-masing merupakan *query*, *key*, dan *value*, sedangkan d_k adalah dimensi vektor yang digunakan sebagai faktor normalisasi.



Gambar 4: Arsitektur DistilBERT

Seperti ditunjukkan pada Gambar 4, setiap dokumen terlebih dahulu diproses melalui tokenizer DistilBERT untuk menghasilkan representasi token beserta *attention mask*. Selanjutnya

representasi tersebut diproses oleh enam lapisan Transformer encoder sehingga menghasilkan embedding kontekstual. Representasi token *[CLS]* kemudian diteruskan ke lapisan klasifikasi untuk menghasilkan probabilitas kelas menggunakan fungsi *softmax* Hinton dkk. (2015). Pada penelitian ini, DistilBERT digunakan sebagai model *deep learning* untuk membandingkan performanya dengan algoritma machine learning klasik.

Pada proses pelatihan, model menggunakan *pre-trained distilbert-base-uncased* dengan panjang maksimum token (*maximum sequence length*) sebesar 128. Ukuran *batch* yang digunakan adalah 8 untuk data pelatihan dan 16 untuk data pengujian. Model dilatih selama 10 *epoch* menggunakan optimizer AdamW dengan *learning rate* sebesar 1.5×10^{-5}. Lapisan *dropout* sebesar 0.3 diterapkan sebelum lapisan klasifikasi untuk membantu mengurangi risiko *overfitting*. Selain itu, fungsi *cross-entropy loss* diberikan bobot kelas (*class weight*) berdasarkan distribusi kelas pada data pelatihan untuk mengurangi pengaruh ketidakseimbangan jumlah data antarkelas. Selama proses pelatihan, digunakan *linear learning rate scheduler* dengan jumlah langkah *warm-up* sebesar 0.

2.5 Evaluation Metrics

Kinerja model dievaluasi menggunakan *confusion matrix* dan lima metrik evaluasi, yaitu *accuracy*, *precision*, *recall*, *F1-score*, dan *Area Under the Curve* (AUC). Karena dataset yang digunakan dalam penelitian ini memiliki tiga kelas, yaitu *moderate*, *not depression*, dan *severe*, maka evaluasi dilakukan menggunakan pendekatan *multiclass* dengan skema *one-vs-rest* (*OvR*). Pada pendekatan ini, setiap kelas secara bergantian dianggap sebagai kelas positif, sedangkan dua kelas lainnya dianggap sebagai kelas negatif. Dengan demikian, nilai *true positive* (TP), *true negative* (TN), *false positive* (FP), dan *false negative* (FN) dihitung untuk masing-masing kelas.

Untuk suatu kelas k, *true positive* merupakan jumlah data kelas k yang berhasil diprediksi sebagai kelas k. *false positive* merupakan jumlah data dari kelas lain yang salah diprediksi sebagai kelas k. *false negative* merupakan jumlah data kelas k yang diprediksi sebagai kelas lain, sedangkan *true negative* merupakan jumlah data dari kelas lain yang berhasil diprediksi bukan sebagai kelas k. Ilustrasi tersebut dapat dituliskan sebagai berikut:

$$TP_k = CM_{kk}, \tag{13}$$

$$FN_k = \sum_{j \neq k} CM_{kj}, \tag{14}$$

$$FP_k = \sum_{i \neq k} CM_{ik}, \tag{15}$$

$$TN_k = \sum_{i \neq k} \sum_{j \neq k} CM_{ij}, \tag{16}$$

dengan CM_{ij} merupakan jumlah data dengan kelas aktual i yang diprediksi sebagai kelas j, sedangkan k menunjukkan kelas yang sedang dievaluasi.

Berdasarkan nilai tersebut, *precision*, *recall*, dan *F1-score* untuk masing-masing kelas dihitung menggunakan persamaan berikut:

$$Precision_k = \frac{TP_k}{TP_k + FP_k}, \tag{17}$$

$$Recall_k = \frac{TP_k}{TP_k + FN_k}, \tag{18}$$

$$F1_k = 2 \times \frac{Precision_k \times Recall_k}{Precision_k + Recall_k}. \tag{19}$$

Nilai *accuracy* dihitung berdasarkan jumlah prediksi yang benar terhadap keseluruhan data pengujian dan dirumuskan sebagai berikut:

$$Accuracy = \frac{\sum_{k=1}^{K} TP_k}{N}, \tag{20}$$

dengan K merupakan jumlah kelas dan N merupakan jumlah seluruh data pengujian.

Untuk memperoleh nilai evaluasi keseluruhan yang memberikan bobot yang sama kepada setiap kelas, penelitian ini menggunakan *macro average*. Nilai *macro precision*, *macro recall*, dan *macro F1-score* dihitung dengan merata-ratakan nilai metrik dari seluruh kelas:

$$Macro\text{-}F1 = \frac{1}{K} \sum_{k=1}^{K} F1_k. \tag{21}$$

Penggunaan *macro average* memungkinkan setiap kelas memberikan kontribusi yang sama terhadap nilai evaluasi, termasuk kelas dengan jumlah data yang lebih sedikit. Oleh karena itu, *macro F1-score* digunakan sebagai salah satu metrik utama dalam penelitian ini karena distribusi kelas pada dataset tidak seimbang.

Selain itu, kemampuan model dalam membedakan setiap kelas dievaluasi menggunakan kurva *Receiver Operating Characteristic* (ROC) dan nilai *Area Under the Curve* (AUC). Pada klasifikasi *multiclass*, AUC dihitung menggunakan pendekatan *one-vs-rest* untuk setiap kelas, kemudian dirata-ratakan secara *macro*. Nilai *True Positive Rate* (TPR) dan *False Positive Rate* (FPR) untuk masing-masing kelas dirumuskan sebagai berikut:

$$TPR_k = \frac{TP_k}{TP_k + FN_k}, \tag{22}$$

$$FPR_k = \frac{FP_k}{FP_k + TN_k}. \tag{23}$$

Nilai AUC kemudian merepresentasikan kemampuan model dalam membedakan satu kelas terhadap seluruh kelas lainnya. Semakin tinggi nilai AUC, semakin baik kemampuan model dalam membedakan kelas tersebut. Pada penelitian ini, nilai AUC multiclass dihitung menggunakan pendekatan *one-vs-rest* dengan *macro averaging*, sehingga nilai akhir AUC merepresentasikan rata-rata kemampuan model dalam membedakan ketiga kelas depresi.

2.6 Pengumpulan dan Scraping Data

Pengumpulan data dilakukan untuk memperoleh data teks dari platform X yang selanjutnya digunakan sebagai data dunia nyata dalam pengujian model klasifikasi tingkat depresi. Proses scraping dilakukan pada 25 Mei 2026 menggunakan Tweet Harvest versi 2.6.1 dengan memanfaatkan autentikasi akun X. Data yang dikumpulkan menggunakan bahasa Inggris dengan menambahkan parameter `lang:en` pada setiap pencarian. Pencarian dilakukan menggunakan tab *Latest* sehingga

unggahan yang diperoleh berasal dari hasil pencarian terbaru yang tersedia pada saat proses pengumpulan data.

Kata kunci yang digunakan dalam proses scraping terdiri atas enam kombinasi kata kunci yang berkaitan dengan kondisi emosional dan kesehatan mental, yaitu *("depressed" OR "want to die" OR "hopeless" OR "suicidal")*, *("i hate my life" OR "fuck my life" OR "life is meaningless" OR "suicidal thoughts")*, *("emotionally exhausted" OR "mental breakdown" OR "stressed out" OR "empty inside")*, *("anxiety" OR "panic attack" OR "stressed out")*, *("i cant do this anymore" OR "i feel numb" OR "kill myself" OR "end my life")*, serta *("overthinking" OR "emotionally" OR "burnout")*. Setiap kombinasi kata kunci digunakan dalam proses scraping secara terpisah dengan batas pengambilan maksimal 300 unggahan per proses.

Hasil dari enam proses scraping menghasilkan enam berkas data dengan jumlah keseluruhan sebanyak 1.140 unggahan. Seluruh berkas tersebut kemudian digabungkan menjadi satu dataset. Pada tahap penggabungan, dilakukan pembersihan data dengan menghapus data duplikat menggunakan fungsi *drop_ duplicates()* serta menghapus baris yang seluruh nilainya kosong menggunakan *dropna(how='all')*. Setelah proses tersebut, diperoleh sebanyak 1.139 unggahan yang digunakan sebagai data hasil scraping untuk tahap pengujian model.

Data hasil scraping tersebut tidak diberikan label secara manual, melainkan diklasifikasikan menggunakan model terbaik yang telah diperoleh dari hasil pengujian pada dataset depresi. Dengan demikian, data scraping digunakan sebagai data eksternal untuk melihat distribusi hasil prediksi model pada unggahan yang diperoleh dari kondisi data dunia nyata.

3 HASIL DAN PEMBAHASAN

Pada bagian ini disajikan hasil pengujian model klasifikasi depresi menggunakan beberapa algoritma machine learning dan deep learning, yaitu Support Vector Machine (SVM), Random Forest, Naïve Bayes, dan DistilBERT. Setiap algoritma dievaluasi menggunakan berbagai kombinasi fitur, meliputi TF-IDF, NRC Emotion Lexicon, LIWC, Word2Vec (CBOW dan Skip-Gram), serta kombinasi beberapa fitur. Evaluasi dilakukan menggunakan metrik accuracy, precision, recall, F1-score, dan Area Under Curve (AUC) untuk mengetahui kombinasi model dan representasi fitur yang memberikan performa terbaik.

3.1 Hasil Klasifikasi Menggunakan Support Vector Machine

Tabel 3 menunjukkan hasil pengujian algoritma Support Vector Machine pada seluruh skenario ekstraksi fitur yang digunakan. Pengujian dilakukan untuk mengetahui pengaruh masing-masing representasi fitur terhadap kemampuan SVM dalam mengklasifikasikan tingkat depresi.

Tabel 3: Hasil Pengujian Support Vector Machine

Metode	Accuracy	Precision	Recall	F1-score	AUC
SVM + TF-IDF	59%	56%	54%	55%	74%
SVM + NRC	47%	44%	46%	41%	66%
SVM + LIWC	48%	48%	51%	44%	68%
SVM + TF-IDF + NRC	59%	56%	59%	57%	74%
SVM + TF-IDF + LIWC	57%	54%	56%	55%	74%
SVM + TF-IDF + NRC + LIWC	58%	54%	57%	55%	74%
SVM + Word2Vec (CBOW)	52%	49%	55%	49%	72%
SVM + Word2Vec (Skip-Gram)	56%	52%	57%	52%	73%

Berdasarkan Tabel 3, Support Vector Machine menunjukkan performa terbaik pada kombinasi TF-IDF dan NRC Emotion Lexicon dengan accuracy sebesar 59%, precision sebesar 56%, recall sebesar 59%, F1-score sebesar 57%, dan AUC sebesar 74%. Dibandingkan penggunaan TF-IDF saja, penambahan NRC meningkatkan recall dari 54% menjadi 59% dan F1-score dari 55% menjadi 57%, meskipun accuracy tetap sebesar 59%. Hasil tersebut menunjukkan bahwa informasi emosional dari NRC dapat melengkapi representasi TF-IDF dengan memberikan informasi mengenai aspek afektif dalam teks, sehingga membantu model mengenali karakteristik teks yang berkaitan dengan tingkat depresi.

Penggunaan NRC dan LIWC secara tunggal menghasilkan performa yang lebih rendah, dengan accuracy masing-masing sebesar 47% dan 48%. Hal ini menunjukkan bahwa fitur emosional dan psikoleksikal belum cukup untuk merepresentasikan karakteristik teks secara keseluruhan apabila digunakan tanpa representasi kata. Sementara itu, Word2Vec CBOW dan Skip-Gram masing-masing menghasilkan F1-score sebesar 49% dan 52%. Performa yang lebih rendah tersebut dapat berkaitan dengan sifat Word2Vec sebagai *embedding* statis dan penggunaan rata-rata vektor kata sebagai representasi dokumen, sehingga informasi urutan dan konteks antarkata tidak sepenuhnya dipertahankan.

Secara keseluruhan, hasil pengujian menunjukkan bahwa kombinasi representasi berbasis TF-IDF dengan informasi emosional dari NRC memberikan performa paling baik pada SVM. Penambahan LIWC pada kombinasi TF-IDF dan NRC tidak memberikan peningkatan yang berarti, sehingga dalam penelitian ini NRC memberikan kontribusi yang lebih relevan sebagai fitur pelengkap TF-IDF.

3.2 Hasil Klasifikasi Menggunakan Random Forest

Tabel 4 menunjukkan hasil pengujian algoritma Random Forest menggunakan seluruh kombinasi fitur.

Tabel 4: Hasil Pengujian Random Forest

Metode	Accuracy	Precision	Recall	F1-score	AUC
Random Forest + TF-IDF	59%	68%	43%	43%	74%
Random Forest + NRC	55%	43%	40%	38%	50%
Random Forest + LIWC	59%	60%	44%	45%	70%
Random Forest + TF-IDF + NRC	62%	69%	47%	48%	76%
Random Forest + TF-IDF + LIWC	60%	71%	45%	46%	75%
Random Forest + TF-IDF + NRC + LIWC	61%	68%	46%	46%	75%
Random Forest + Word2Vec (CBOW)	62%	65%	51%	53%	74%
Random Forest + Word2Vec (Skip-Gram)	61%	66%	51%	52%	75%

Berdasarkan Tabel 4, Random Forest dengan kombinasi TF-IDF dan NRC menghasilkan nilai AUC tertinggi sebesar 76% dengan accuracy sebesar 62%. Nilai AUC yang tinggi menunjukkan bahwa model memiliki kemampuan diskriminasi kelas yang baik, tetapi nilai F1-score yang diperoleh masih sebesar 48%. Hal ini menunjukkan bahwa kemampuan model dalam membedakan kelas secara keseluruhan belum seimbang, khususnya dalam mendeteksi setiap kelas secara konsisten.

Penggunaan Word2Vec menghasilkan accuracy yang sama, yaitu 62%, dengan F1-score yang lebih tinggi sebesar 53%. Hasil tersebut menunjukkan bahwa AUC yang lebih tinggi tidak selalu menunjukkan performa klasifikasi terbaik karena AUC mengukur kemampuan diskriminasi berdasarkan probabilitas prediksi, sedangkan F1-score mempertimbangkan keseimbangan antara precision dan recall. Oleh karena itu, Random Forest dengan TF-IDF dan NRC belum menjadi kombinasi terbaik secara keseluruhan.

3.3 Hasil Klasifikasi Menggunakan Naïve Bayes

Tabel 5 memperlihatkan hasil klasifikasi menggunakan algoritma Naïve Bayes menggunakan seluruh kombinasi fitur.

Tabel 5: Hasil Pengujian Naïve Bayes

Metode	Accuracy	Precision	Recall	F1-score	AUC
Naïve Bayes + TF-IDF	54%	46%	35%	27%	60%
Naïve Bayes + NRC	51%	43%	37%	35%	54%
Naïve Bayes + LIWC	52%	43%	34%	24%	65%
Naïve Bayes + TF-IDF + NRC	58%	65%	43%	43%	62%
Naïve Bayes + TF-IDF + LIWC	56%	78%	38%	32%	66%
Naïve Bayes + TF-IDF + NRC + LIWC	59%	60%	46%	47%	64%
Naïve Bayes + Word2Vec (CBOW)	41%	46%	52%	41%	69%
Naïve Bayes + Word2Vec (Skip-Gram)	40%	47%	53%	39%	70%

Secara umum, Naïve Bayes menghasilkan performa yang lebih rendah dibandingkan SVM dan Random Forest, terutama pada nilai *F1-score*. Performa terbaik diperoleh pada kombinasi TF-IDF + NRC + LIWC dengan *accuracy* sebesar 59%, *precision* sebesar 60%, *recall* sebesar 46%, dan *F1-score* sebesar 47%.

Rendahnya performa Naïve Bayes pada sebagian besar kombinasi fitur dapat dikaitkan dengan asumsi independensi antarfitur. Pada data teks, kemunculan suatu kata umumnya berkaitan

dengan kata lainnya sehingga asumsi tersebut tidak sepenuhnya terpenuhi. Penambahan fitur NRC dan LIWC juga belum memberikan peningkatan yang konsisten, meskipun kombinasi TF-IDF + NRC + LIWC menghasilkan performa terbaik pada Naïve Bayes. Hal ini menunjukkan bahwa informasi emosional dan linguistik dapat memberikan tambahan informasi, tetapi belum cukup untuk mengatasi keterbatasan asumsi independensi pada model.

Walaupun demikian, Naïve Bayes tetap mampu memberikan waktu pelatihan yang lebih singkat dan dapat dijadikan sebagai model pembanding dalam penelitian klasifikasi teks.

3.4 Hasil Klasifikasi Menggunakan DistilBERT

Berbeda dengan algoritma machine learning klasik, DistilBERT menggunakan representasi kontekstual berbasis Transformer sehingga tidak memerlukan proses ekstraksi fitur secara eksplisit.

Tabel 6: Hasil Pengujian DistilBERT

Model	Accuracy	Precision	Recall	F1-score	AUC
DistilBERT	64%	62%	61%	61%	74%

Berdasarkan Tabel 6, DistilBERT memperoleh *accuracy* sebesar 64%, *precision* sebesar 62%, *recall* sebesar 61%, *F1-score* sebesar 61%, dan AUC sebesar 74%. Hasil tersebut menunjukkan bahwa DistilBERT memberikan performa terbaik dibandingkan algoritma *machine learning* klasik yang digunakan dalam penelitian ini.

Gambar 5a menunjukkan bahwa prediksi benar paling banyak terdapat pada kelas *moderate*, dengan 459 data diklasifikasikan secara tepat. Kesalahan klasifikasi terbesar terjadi pada kelas *severe* yang diprediksi sebagai *moderate* sebanyak 195 data. Sementara itu, kelas *not depression* memiliki 67 prediksi yang tepat dan sebagian besar kesalahannya terjadi ketika data diklasifikasikan sebagai *moderate*. Pola tersebut menunjukkan bahwa model masih mengalami kesulitan dalam membedakan tingkat keparahan depresi yang memiliki karakteristik linguistik yang berdekatan.

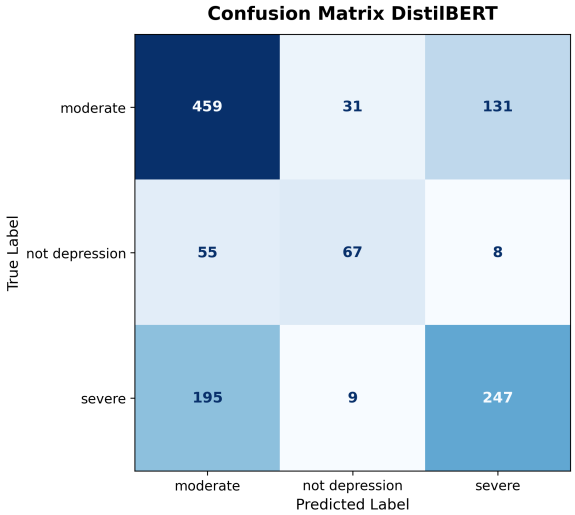
Performa DistilBERT yang lebih tinggi diduga berkaitan dengan kemampuannya menghasilkan representasi kata yang bergantung pada konteks melalui mekanisme *self-attention*. Kemampuan tersebut memungkinkan model mempertimbangkan hubungan antarkata dalam keseluruhan kalimat, sehingga lebih mampu menangkap ekspresi depresi yang tidak selalu ditunjukkan oleh kata-kata tertentu secara langsung. Hal ini berbeda dengan TF-IDF yang merepresentasikan pentingnya kata berdasarkan frekuensi dan Word2Vec yang menghasilkan representasi semantik statis.

3.5 Perbandingan Performa Seluruh Model

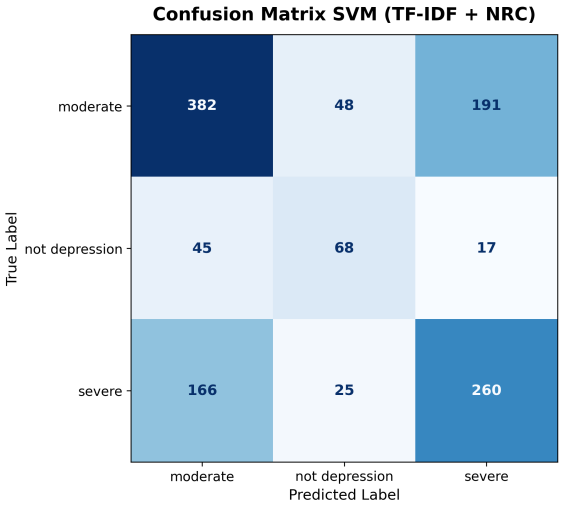
Setiap algoritma pada penelitian ini diuji menggunakan beberapa kombinasi metode ekstraksi fitur untuk memperoleh konfigurasi dengan performa terbaik. Oleh karena itu, pada tahap ini dilakukan perbandingan terhadap model terbaik dari masing-masing algoritma guna mengevaluasi efektivitas pendekatan machine learning klasik maupun Transformer dalam mendeteksi tingkat depresi berdasarkan teks media sosial. Evaluasi dilakukan menggunakan lima metrik, yaitu accuracy, precision, recall, F1-score, dan Area Under Curve (AUC). Ringkasan hasil terbaik dari setiap algoritma disajikan pada Tabel 7.

Tabel 7: Perbandingan Model Terbaik

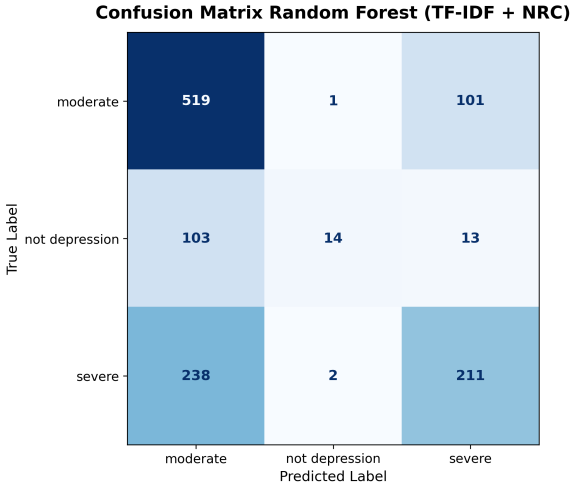
Algoritma	Fitur Terbaik	Accuracy	Precision	Recall	F1-score	AUC
SVM	TF-IDF + NRC	59%	56%	59%	57%	74%
Random Forest	TF-IDF + NRC	62%	69%	47%	48%	76%
Naïve Bayes	TF-IDF + NRC + LIWC	59%	60%	46%	47%	64%
DistilBERT		64%	62%	61%	61%	74%



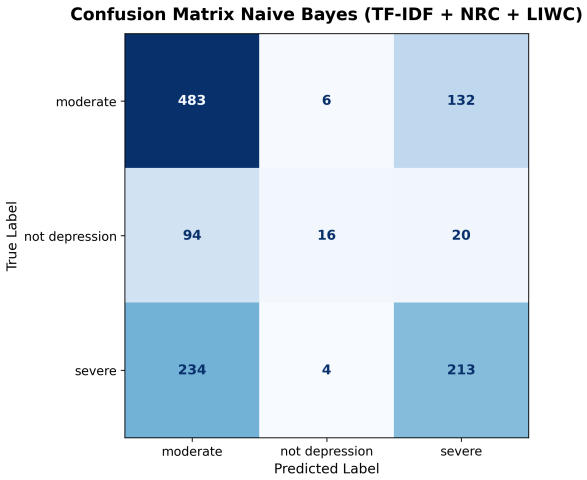
(a) Confusion Matrix DistilBERT



(b) Confusion Matrix SVM + TF-IDF + NRC



(c) Confusion Matrix Random Forest + TF-IDF + NRC



(d) Confusion Matrix Naïve Bayes + TF-IDF + NRC + LIWC

Gambar 5: Perbandingan confusion matrix model DistilBERT, SVM + TF-IDF + NRC, Random Forest + TF-IDF + NRC, dan Naïve Bayes + TF-IDF + NRC + LIWC.

Berdasarkan Tabel 7, DistilBERT menghasilkan performa terbaik berdasarkan accuracy dan

F1-score, masing-masing sebesar 64% dan 61%. Hasil tersebut menunjukkan bahwa representasi kontekstual DistilBERT mampu menangkap hubungan antarkata dan konteks kalimat secara lebih baik dibandingkan representasi fitur pada model machine learning klasik.

Pada kelompok machine learning klasik, SVM dengan kombinasi TF-IDF dan NRC menghasilkan F1-score tertinggi sebesar 57%. Penambahan NRC memberikan informasi emosional yang melengkapi representasi TF-IDF sehingga membantu model membedakan tingkat depresi. Sementara itu, Word2Vec menghasilkan performa yang lebih rendah, yang diduga karena proses perataan vektor kata menjadi representasi dokumen dapat mengurangi informasi konteks dan urutan kata.

Random Forest memperoleh accuracy sebesar 62% dan AUC tertinggi sebesar 76%, tetapi F1-score yang diperoleh hanya 48%. Perbedaan tersebut menunjukkan bahwa AUC yang tinggi tidak selalu mencerminkan kemampuan klasifikasi yang seimbang pada seluruh kelas. Nilai recall Random Forest sebesar 47% menunjukkan bahwa masih terdapat kelas yang sulit dikenali secara konsisten. Oleh karena itu, meskipun Random Forest memiliki kemampuan diskriminasi yang baik berdasarkan AUC, F1-score digunakan sebagai salah satu pertimbangan utama dalam menentukan model terbaik karena lebih mencerminkan keseimbangan antara precision dan recall pada dataset yang tidak seimbang.

Naïve Bayes memperoleh F1-score sebesar 47%, yang merupakan nilai terendah di antara model utama. Kondisi tersebut diduga berkaitan dengan asumsi independensi antarfitur pada Naïve Bayes, sedangkan karakteristik teks memiliki hubungan antarkata dan informasi emosional yang saling berkaitan.

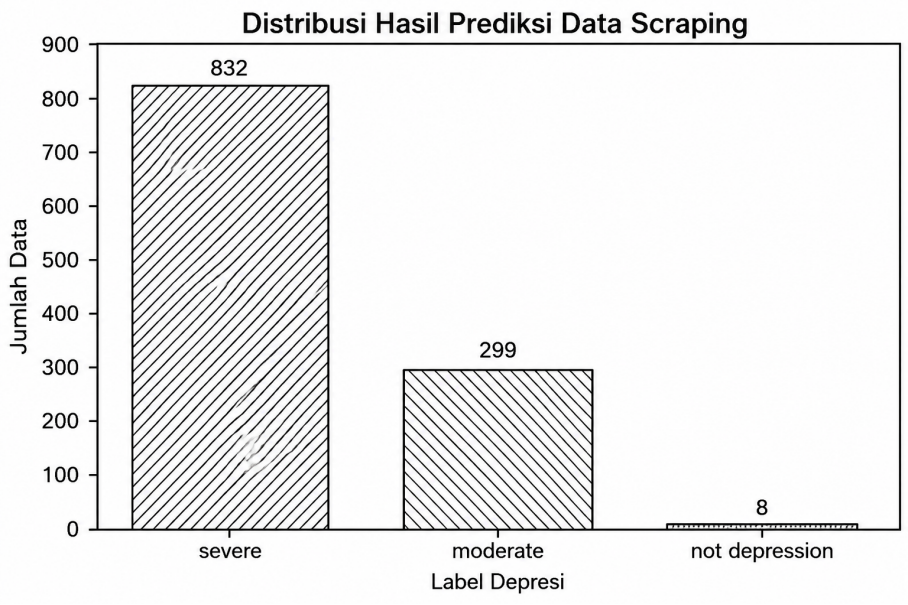
Perbedaan karakteristik linguistik antar kelas turut memengaruhi hasil klasifikasi. Kelas *not depression moderate*, dan *severe* memiliki pola bahasa yang dapat saling tumpang tindih, terutama pada unggahan yang mengekspresikan emosi negatif tanpa secara eksplisit menunjukkan tingkat depresi tertentu. Kondisi tersebut dapat menyebabkan model mengalami kesalahan dalam membedakan kelas *moderate* dan *severe*. Penggunaan fitur NRC pada SVM membantu menambahkan informasi mengenai aspek emosional teks, sedangkan DistilBERT dapat mempertimbangkan konteks kalimat secara lebih menyeluruh

3.6 Pengujian pada Data Scraping

Setelah model terbaik diperoleh berdasarkan hasil pengujian pada dataset depresi, model tersebut kemudian diterapkan pada data hasil scraping dari platform X. Pengujian ini bertujuan untuk mengetahui distribusi tingkat depresi pada data dunia nyata serta mengevaluasi kemampuan model dalam mengidentifikasi unggahan yang mengandung indikasi depresi.

Tabel 8: Distribusi Prediksi Data Scraping

Label Depresi	Jumlah	Persentase (%)
Severe	832	73.05
Moderate	299	26.25
Not Depression	8	0.70
Total	1139	100.00



Gambar 6: Distribusi Hasil Prediksi pada Data Scraping

Tabel 9: Contoh Hasil Prediksi Data Scraping

Teks	Prediksi
I feel exhausted every day and nothing makes me happy anymore...	Severe
I have been feeling lonely lately but still trying to stay strong.	Moderate
Today was a productive day and I'm grateful for everyone around me.	Not Depression

Berdasarkan Gambar 6 dan Tabel 8, ebagian besar data hasil scraping diprediksi ke dalam kategori *severe depression*, yaitu sebanyak 832 unggahan (73,05%), diikuti kategori *moderate depression* sebanyak 299 unggahan (26,25%). Sementara itu, hanya 8 unggahan (0,70%) yang diprediksi sebagai *not depression*. Dominasi kategori *severe depression* menunjukkan bahwa data yang diperoleh melalui kata kunci pencarian cenderung memiliki kandungan emosional negatif yang lebih kuat.

Dominasi prediksi tersebut juga menunjukkan bahwa karakteristik data scraping berbeda dengan distribusi data pada dataset pengujian. Kondisi ini dapat dipengaruhi oleh pemilihan kata kunci yang digunakan dalam proses scraping, seperti *overthinking*, *emotionally*, dan *burnout*, yang cenderung menghasilkan unggahan dengan ekspresi tekanan emosional. Hasil prediksi ini tidak dimaksudkan sebagai diagnosis klinis, melainkan sebagai klasifikasi berdasarkan pola bahasa yang dipelajari oleh model.

4 KESIMPULAN

Penelitian ini menerapkan pendekatan *Natural Language Processing* (NLP) untuk klasifikasi tingkat depresi pada teks media sosial menggunakan beberapa representasi fitur dan algoritma klasifikasi. Representasi yang digunakan meliputi TF-IDF, NRC Emotion Lexicon, LIWC, Word2Vec dengan arsitektur CBOW dan Skip-Gram, serta DistilBERT sebagai model berbasis Transformer.

Berdasarkan hasil pengujian, DistilBERT memberikan performa terbaik dengan *accuracy* sebesar 64%, *precision* sebesar 62%, *recall* sebesar 61%, *F1-score* sebesar 61%, dan AUC sebesar 74%. Pada algoritma *machine learning* klasik, SVM dengan kombinasi TF-IDF dan NRC menghasilkan performa terbaik dengan *accuracy* sebesar 59% dan *F1-score* sebesar 57%. Sementara itu, Random Forest memperoleh AUC tertinggi sebesar 76%, tetapi memiliki *F1-score* sebesar 48%, sehingga belum menjadi model dengan performa keseluruhan terbaik. Pengujian pada data hasil *web scraping* dari platform X menunjukkan dominasi prediksi pada kategori *severe depression*. Hasil tersebut menunjukkan bahwa model dapat digunakan untuk mengklasifikasikan teks baru berdasarkan pola bahasa yang dipelajari, tetapi hasil prediksi tidak dapat digunakan sebagai diagnosis klinis.

Penelitian selanjutnya dapat menggunakan dataset yang lebih beragam dan seimbang, mengeksplorasi model Transformer yang lebih mutakhir, serta melakukan optimasi model dan evaluasi pada data eksternal untuk meningkatkan kemampuan generalisasi.

Ucapan Terima Kasih

Penulis mengucapkan terima kasih kepada Program Studi Sarjana Sains Data Universitas Negeri Surabaya serta seluruh pihak yang telah memberikan dukungan dan masukan selama pelaksanaan penelitian ini.

References

Acheampong, F. A., Wenyu, C., dan Nunoo-Mensah, H. (2020). Text-based emotion detection: Advances, challenges, and opportunities. *Engineering Reports*, 2(7):1–24.

Aldhyani, T. H. H., Alsubari, S. N., Alshebami, A. S., Alkahtani, H., dan Ahmed, Z. A. T. (2022). Detecting and analyzing suicidal ideation on social media using deep learning and machine learning models. *International Journal of Environmental Research and Public Health*, 19(19).

Aldkheel, A. dan Zhou, L. (2024). Depression detection on social media: A classification framework and research challenges and opportunities. *Journal of Healthcare Informatics Research*, 8(1):88–120.

Alin, O. dan Utami, E. (2025). Analisis sentimen pada kurikulum merdeka menggunakan word2vec dan algoritma long short-term memory. *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, 10(3):2794–2805.

Anugerah, S. M., Wijaya, R., dan Bijaksana, M. A. (2024). Sentimen analysis social media for disaster using naïve bayes and indobert. *INTEK: Jurnal Penelitian*, 11(1):51–58.

Bader, M., Abdelwanis, M., Maalouf, M., dan Jelinek, H. F. (2024). Detecting depression severity using weighted random forest and oxidative stress biomarkers. *Scientific Reports*, 14(1):1–18.

Benjachairat, P., Senivongse, T., Taephant, N., Puvapaisankit, J., Maturosjamnan, C., dan Kultananawat, T. (2024). Classification of suicidal ideation severity from twitter messages

using machine learning. *International Journal of Information Management Data Insights*, 4(2):100280.

Devlin, J., Chang, M.-W., Lee, K., dan Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 4171–4186.

Ghosal, S. dan Jain, A. (2022). Depression and suicide risk detection on social media using fasttext embedding and xgboost classifier. *Procedia Computer Science*, 218:1631–1639.

Hinton, G., Vinyals, O., dan Dean, J. (2015). Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Mohammad, S. M. dan Turney, P. D. (2013). Crowdsourcing a word-emotion association lexicon. *Computational Intelligence*, 29(3):436–465.

Ovirianti, N. H., Zarlis, M., dan Mawengkang, H. (2022). Support vector machine using a classification algorithm. *SinkrOn*, 7(3):2103–2107.

Poswiata, R. dan Perełkiewicz, M. (2022). Opi@lt-edi-acl2022: Detecting signs of depression from social media text using roberta pre-trained language models. In *Proceedings of the 2nd Workshop on Language Technology for Equality, Diversity and Inclusion (LT-EDI 2022)*, pages 276–282.

Qaiser, S. dan Ali, R. (2018). Text mining: Use of tf-idf to examine the relevance of words to documents. *International Journal of Computer Applications*, 181(1):25–29.

Ratajczak, P., Martyński, J., Zięba, J. K., Świło, K., Kopciuch, D., Paczkowska, A., Zaprutko, T., dan Kus, K. (2024). Comparative efficacy of animal depression models and antidepressant treatment: A systematic review and meta-analysis. *Pharmaceutics*, 16(9).

Rong, X. (2016). word2vec parameter learning explained. *arXiv preprint arXiv:1411.2738*.

Sanh, V., Debut, L., Chaumond, J., dan Wolf, T. (2020). Distilbert: A distilled version of bert: Smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Saputra, F. T., Nurhadryani, Y., Wijaya, S. H., dan Defina, D. (2021). Analisis sentimen bahasa indonesia pada twitter menggunakan struktur tree berbasis leksikon. *Jurnal Teknologi Informasi dan Ilmu Komputer*, 8(1):135.

Syafiqah, A., Zaman, S., dan Imamudin, M. (2025). Text mining approach to emotion analysis in translation of surah yusuf with nrc emotion lexicon. *IT Journal Research and Development*, 9(2):108–122.

Tabany, M. dan Gueffal, M. (2024). Sentiment analysis and fake amazon reviews classification using svm supervised machine learning model. *Journal of Advances in Information Technology*, 15(1):49–58.

Tausczik, Y. R. dan Pennebaker, J. W. (2010). The psychological meaning of words: Liwc and computerized text analysis methods. *Journal of Language and Social Psychology*, 29(1):24–54.

Troiano, E., Oberländer, L., dan Klinger, R. (2023). Dimensional modeling of emotions in text with appraisal theories: Corpus creation, annotation reliability, and prediction. *Computational Linguistics*, 49(1):1–72.

World Health Organization (2025). Depressive disorder (depression). Accessed August 29, 2025.

Zhao, N., Jiao, D., Bai, S., dan Zhu, T. (2016). Evaluating the validity of simplified chinese version of liwc in detecting psychological expressions in short texts on social network services. *PLoS ONE*, 11(6):1–15.