Journal of Applied Informatics and Computing (JAIC)
Vol.9, No.6, December 2025, pp. 3283~3294
e-ISSN: 2548-6861

Analysis of Stacking Ensemble Method in Machine Learning Algorithms to Predict Student Depression

Naouthla Asia Levina [1]*, Majid Rahardi[2]*
* Informatika, Fakultas Ilmu Komputer, Universitas Amikom Yogyakarta
naouthlaasia@students.amikom.ac.id[1], majid@amikom.ac.id[2]

Article Info

Article history:

Received 2025-10-07
Revised 2025-11-03
Accepted 2025-11-08

Keyword:

Depression,
University Students,
Machine Learning,
Stacking Ensemble,
Prediction.

ABSTRACT

Mental health issues, particularly depression among university students, require early detection and intervention due to their profound impact on academic performance and overall well-being. Although machine learning has been utilized in previous studies to predict depression, most research still relies on single-model approaches and rarely employs publicly available datasets that have undergone comprehensive preprocessing. This study aims to develop a depression prediction model for university students using a two-level stacking ensemble technique with cross-validation stacking, integrating Random Forest, Gradient Boosting, and XGBoost as base learners, and Logistic Regression as the meta-learner. A public dataset from Kaggle was utilized, consisting of 502 student records and 10 multidimensional predictor variables. Data preprocessing included cleaning, feature encoding, and standardization. Model performance was evaluated using accuracy, precision, recall, F1-score, and ROC-AUC metrics. The proposed stacking ensemble model achieved excellent performance, with an accuracy of 98.02%, ROC-AUC of 99.8%, precision of 96%, recall of 100%, and an F1-score of 98% for the depression class. These results demonstrate that the stacking ensemble method is highly effective for early depression detection among university students and has strong potential for implementation as a decision-support tool in academic environments.

I. PENDAHULUAN

Kesehatan mental menjadi faktor yang penting dalam kehidupan manusia karena dapat memengaruhi cara berpikir, emosi, dan juga tindakan [1]. Gangguan kesehatan mental seperti depresi di kalangan mahasiswa perlu adanya perhatian serius [2]. Depresi biasanya terjadi akibat adanya gangguan suasana hati yang ditandai dengan munculnya perasaan sedih berkepanjangan, perubahan pola makan atau tidur, perasaan lelah yang berlebihan [3]. Kondisi ini tidak hanya memengaruhi kesejahteraan individu, tetapi juga dapat berdampak pada prestasi akademik dan hubungan sosial mahasiswa.

Banyak mahasiswa yang menghadapi tekanan akademik yang menyebabkan peningkatan risiko depresi serta memperburuk kondisi kesehatan mental mereka [4]. Berdasarkan data dari World Health Organization (WHO), depresi menjadi penyebab utama masalah kesehatan di seluruh dunia, lebih dari 800.000 orang meninggal setiap tahunnya karena bunuh diri [5]. Tingginya angka tersebut menunjukkan bahwa depresi bukan hanya persoalan individu, melainkan juga menjadi tantangan serius dalam sistem kesehatan global. Hal ini menggarisbawahi pentingnya upaya deteksi dan penanganan dini terhadap depresi, khususnya di kalangan mahasiswa sebagai kelompok yang rentan.

Seiring perkembangan teknologi informasi, penerapan Machine Learning (ML) dalam bidang kesehatan mental menjadi topik yang semakin banyak dikaji. Teknik ML menawarkan keunggulan dalam mengolah data berdimensi besar dan menemukan pola prediktif yang sulit dideteksi secara manual, sehingga mampu membantu lembaga pendidikan atau tenaga medis dalam melakukan deteksi awal [6]. *Review* sistematis terbaru menyatakan bahwa model ML dapat mencapai akurasi performa lebih dari 70% dalam mendeteksi depresi, kecemasan, dan stres pada mahasiswa [7]. Namun, sebagian besar penelitian masih terbatas pada penggunaan model tunggal (seperti Logistic

Regression atau Decision Tree) tanpa mengeksplorasi secara mendalam penggabungan algoritma. Pendekatan model tunggal kerap menghadapi keterbatasan dalam hal stabilitas dan kemampuan generalisasi, terutama ketika menghadapi kerumitan data psikologis.

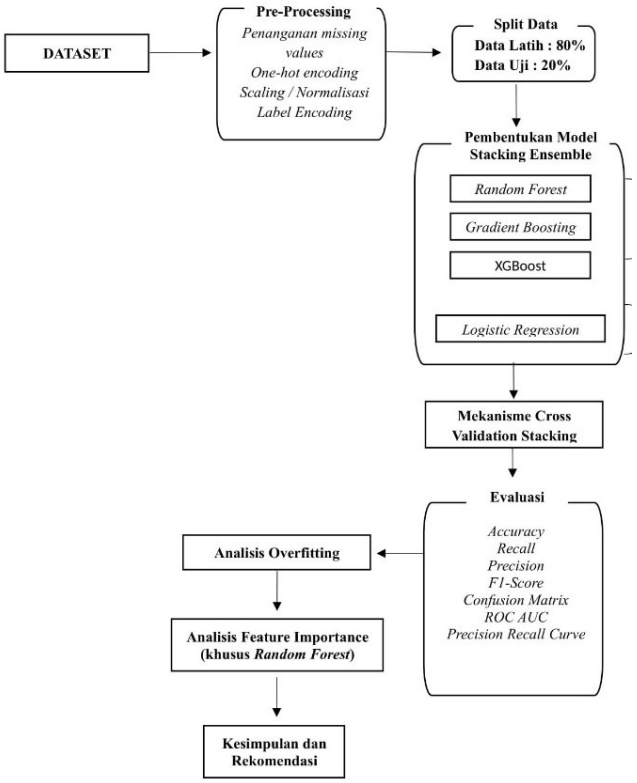
Untuk mengatasi keterbatasan model tunggal digunakan metode ensemble learning, khususnya Stacking Ensemble, dianggap lebih potensial. Stacking Ensemble bekerja dengan menggabungkan keunggulan dari beberapa model dasar (base learner), sehingga menghasilkan model prediksi yang lebih akurat dan stabil [8]. Metode ini memungkinkan integrasi berbagai algoritma untuk saling melengkapi kelemahan masing-masing.

Beberapa studi terdahulu telah menunjukkan efektivitas pendekatan ensemble dalam prediksi depresi. Penelitian dari Vergaray et al. [9] pada tahun 2023 memperoleh akurasi 94,69% menggunakan *stacking ensemble*. Selanjutnya, Selvaraj & Mohandoss [10] pada tahun 2024, mencapai akurasi 98% dengan pendekatan serupa. Penelitian oleh Hasan et al. [11] juga menemukan bahwa metode *ensemble* seperti Random Forest dan Gradient Boosting memberikan performa yang lebih baik daripada model tunggal. Temuan-temuan ini menegaskan bahwa metode *ensemble* dapat meningkatkan keakuratan deteksi depresi.

Penelitian ini bertujuan untuk mengembangkan dan menganalisis kinerja model prediksi depresi pada mahasiswa menggunakan teknik Stacking Ensemble dua tingkat yang lebih robust. Model ini mengintegrasikan Random Forest, Gradient Boosting, dan XGBoost sebagai base learner yang heterogen, serta Logistic Regression sebagai meta learner. Dengan memanfaatkan dataset publik yang telah melalui proses pre-processing menyeluruh dan mengimplementasikan Cross-Validation Stacking (CVS) untuk menjamin validitas, diharapkan model yang dikembangkan dapat menghasilkan prediksi depresi yang sangat akurat dan stabil. Model ini diharapkan dapat dimanfaatkan sebagai alat bantu skrining dini yang efektif di lingkungan kampus.

Dengan demikian, penelitian ini diharapkan memberikan kontribusi nyata dalam pengembangan sistem pendeteksian dini depresi berbasis machine learning yang dapat diimplementasikan di lingkungan kampus. Pendekatan ini memungkinkan lembaga pendidikan untuk secara proaktif memantau kondisi psikologis mahasiswa melalui analisis data akademik dan perilaku belajar. Selain itu, hasil penelitian ini dapat menjadi landasan bagi pengembangan decision support system di bidang kesehatan mental berbasis data, yang mampu memberikan rekomendasi intervensi lebih awal bagi mahasiswa berisiko. Secara keseluruhan, penelitian ini tidak hanya berfokus pada peningkatan akurasi model prediksi, tetapi juga pada penerapan praktisnya sebagai inovasi dalam upaya pencegahan dan penanganan depresi di lingkungan pendidikan tinggi.

II. METODE



Gambar 1. Alur Penelitian

A. Dataset

Dataset adalah sekumpulan data yang telah dikumpulkan dan disusun oleh pihak lain, yang kemudian digunakan kembali oleh peneliti untuk tujuan tertentu [12]. Dataset yang digunakan dalam penelitian ini diperoleh dari Kaggle melalui tautan https://www.kaggle.com/datasets/ikynahidwin/depressionstunt-dataset. Kaggle merupakan platform berbasis daring yang menyediakan kumpulan dataset publik untuk keperluan data mining dan data science, sehingga dapat dimanfaatkan sebagai sumber pembelajaran dan penelitian [13]. Dataset ini berisi 502 data mahasiswa dengan total 11 atribut, yang terdiri atas 10 variabel independent dan 1 variabel dependen (target), yaitu Depression yang menunjukkan status depresi mahasiswa. Variabel-variabel tersebut mencakup faktor akademik (Academic Pressure, Study Satisfaction, Study Hours), psikologis (Suicidal Thoughts, Sleep Duration), gaya hidup (Dietary Habits, Financial Stress), serta demografis dan genetik (Gender, Age, Family History). Dataset telah melalui proses pembersihan awal oleh penyedia sehingga tidak memiliki missing values maupun duplikasi data. Selain itu, dataset bersifat publik dan dapat diakses secara terbuka tanpa batasan lisensi, sehingga aman digunakan untuk kepentingan penelitian akademik.

Distribusi kelas dalam dataset relatif seimbang, yaitu 252 mahasiswa depresi dan 250 tidak depresi, sehingga risiko bias

klasifikasi dapat diminimalkan. Deskripsi rinci setiap fitur ditampilkan pada Tabel 1.

TABEL I
DESKRIPSI DATASET

No	Fitur	Keterangan
1	Gender	Male/Female
2	Age	Age
3	Academic Pressure	1 (Low) – 5 (High)
4	Study Satisfaction	1 (Low) – 5 (High)
5	Sleep Duration	< 5 jam, 5-6 jam, 7-8 jam, < 8 jam
6	Dietary Habits	Healthy, Moderate, Unhealthy
7	Suicidal Thoughts	Yes/No
8	Study Hours	Work hours per day
9	Financial Stress	1 (Low) – 5 (High)
10	Family History	Yes/No
11	Depression	Yes/No

B. Pre-Processing Data

Pra-pemrosesan data merupakan proses penting karena pengolahan data yang baik dapat meningkatkan akurasi dan mengurangi bias pada saat pelatihan model [14]. Pada penelitian ini, proses pra-pemrosesan dilakukan secara sistematis untuk memastikan dataset bersih, terstandardisasi, dan siap digunakan oleh algoritma machine learning. Tahapan yang dilakukan dijelaskan sebagai berikut.

1) Pengecekan Nilai Kosong (Missing Value)

Langkah awal pra-pemrosesan data yang harus dilakukan adalah mengecek nilai kosong yang ada di setiap kolom dalam dataset, untuk memastikan data bersih. Proses ini dilakukan dengan menggunakan fungsi 'df.isnull().sum()' untuk memastikan seluruh kolom terisi dan tidak kosong.

2) One-hot Encoding

Terdapat fitur-fitur kategorikal seperti, 'Gender', 'Sleep Duration', 'Dietary Habits', 'Suicidal Thoughts', 'Family History of Mental Illness' yang harus diubah menjadi angka karena tidak dapat langsung digunakan oleh model. Oleh karena itu, menggunakan teknik One-Hot Encoding, yaitu mengubah kategori menjadi numerik (0 atau 1).

3) Scaling/Normalisasi fitur numerik

Terdapat fitur-fitur 'Age', 'Academic Pressure', 'Study Satisfaction', 'Study Hours', 'Financial Stress', memiliki skala yang berbeda-beda. Misalnya, didalam dataset terdapat kolom age yang bernilai puluhan, sementara kolom academic pressure berada dalam sekala 1-5. Perbedaan skala ini dapat menyebabkan model bias terhadap fitur tertentu. Jadi, untuk mengatasi hal tersebut agar tidak terjadi, dilakukan proses normalisasi dengan menggunakan 'StandardScaler', yang dapat mengubah nilai menjadi distribusi normal.

4) Label Encoding

Variabel dependen dalam dataset yaitu kolom depression, yang berisi nilai 'Yes' dan 'No' untuk menunjukkan apakah mahasiswa mengalami depresi atau tidak. Karena model hanya dapat memproses data numerik, maka nilai tersebut diubah menjadi angka 1 (depresi) dan 0 (tidak depresi).

Seluruh proses pre-processing data yang sudah dilakukan sebelumnya, yaitu (One-Hot Encoding dan Standardization) kemudian diatur dengan menggunakan 'ColumnTransformer'. Dengan menggunakan fungsi tersebut memastikan agar proses yang dilakukan menjadi konsisten pada data latih maupun uji.

C. Split Data

Setelah pre-processing data, langkah selanjutnya yaitu membagi data menjadi data latih dan data uji. Pembagian data dengan rasio 80:20, artinya 80% data untuk pelatihan model dan 20% data untuk pengujian model, yang dapat dilihat pada Tabel 2. Tujuan dari pembagian data ini adalah untuk melatih dan menguji model secara adil, sehingga kinerjanya dapat dievaluasi berdasarkan data yang belum pernah dilihat sebelumnya.

TABEL 2
PEMBAGIAN DATA LATIH & UJI

Deskripsi	Data Latih	Data Uji	Total
Proporsi	80%	20%	100%
Jumlah Data	401	101	502

D. Komponen Stacking

Model dikembangkan menggunakan pendekatan Stacking Ensemble dua tingkat dengan strategi cross-validation stacking untuk meningkatkan kemampuan generalisasi dan mencegah data leakage. Metode stacking dipilih karena dapat menggabungkan kekuatan dari base learner dan dapat menghasilkan akurasi prediksi yang lebih akurat dibandingkan dengan model tunggal [15].

Seluruh proses analisis, preprocessing, dan pengembangan model Stacking Ensemble ini diimplementasikan menggunakan bahasa pemrograman Python dengan library utama dari ekosistem machine learning seperti Scikit-learn (digunakan untuk pipeline, preprocessing, Random Forest, Gradient Boosting, Logistic Regression, dan StackingClassifier) dan XGBoost. Komputasi model dijalankan pada lingkungan CPU standar.

1) Base Learner (level-0)

Base learner bertugas untuk mempelajari pola dari data latih secara independen dan menghasilkan prediksi awal yang akan digunakan sebagai input bagi meta-learner pada tahap selanjutnya. Selain itu, pendekatan multi-model memungkinkan setiap algoritma memberikan kontribusi berbeda berdasarkan karakteristik pendekatan masing-masing, sehingga meningkatkan kemampuan generalisasi sistem secara keseluruhan. Dipilih tiga algoritma klasifikasi yang heterogen untuk memanfaatkan keragaman dalam pemodelan:

a) Random Forest

Random Forest merupakan algoritma ensemble berbasis *bagging* yang membangun sejumlah pohon keputusan secara acak dari subset data dan fitur yang berbeda. Hasil akhir ditentukan melalui proses voting mayoritas dari seluruh pohon. Algoritma ini unggul dalam menangani data dengan hubungan non-linear serta memiliki tingkat *robustness* yang tinggi terhadap keberadaan *outlier* maupun *noise*. Selain itu, karena memanfaatkan banyak pohon secara paralel, Random Forest juga mampu mengurangi risiko *overfitting* yang sering terjadi pada model pohon tunggal [16]. Dalam penelitian ini, Random Forest dikonfigurasi dengan parameter: n_estimators=50, random_state=42.

b) Gradient Boosting

Gradient Boosting bekerja secara iteratif dengan menggabungkan sejumlah model pohon keputusan yang lemah (*weak learners*) untuk membentuk model yang lebih kuat. Setiap pohon baru dibangun untuk memperbaiki kesalahan prediksi dari model sebelumnya melalui *proses* minimisasi *loss function*. Pendekatan ini secara bertahap meningkatkan akurasi prediksi dengan memanfaatkan teknik *boosting*, sehingga menghasilkan model yang mampu menangkap pola kompleks pada data dan memiliki performa prediksi yang lebih tinggi dibanding model individual [17]. Dalam penelitian ini, Gradient Boosting dikonfigurasi dengan parameter: n_estimators=50, use_label_encoder=False, eval_matric='logloss', random_state=42.

c) XGBoost

XGBoost merupakan pengembangan dari algoritma Gradient Boosting dengan sejumlah optimasi yang meningkatkan efisiensi dan performa. Algoritma ini memiliki kemampuan regularization (L1 dan L2) untuk mencegah overfitting, mendukung paralelisasi proses pelatihan, serta memiliki efisiensi *komputasi* tinggi melalui penggunaan tree pruning dan cache optimization. Keunggulan ini menjadikan XGBoost salah satu algoritma ensemble yang sangat kompetitif dalam berbagai tugas klasifikasi dan prediksi [16]. Dalam penelitian ini, XGBooat dikonfigurasi dengan parameter: n_estimators=50, random_state=42.

2) Meta Learner (level-1)

Meta learner bertugas untuk menggabungkan prediksi awal dari base learners (model-level 0) menjadi prediksi final. Dengan kata lain, setelah base learners menghasilkan prediksi mereka masing-masing dari data latih, prediksi tersebut digunakan sebagai input (meta-features) ke meta learner. Tujuannya adalah mencari mekanisme optimal baik bobot maupun kombinasi untuk menyatukan kontribusi dari setiap base learner sehingga sistem keseluruhan memiliki kemampuan generalisasi yang lebih baik. Meta learner yang tepat harus cukup sederhana agar tidak over-fit terhadap meta-features yang kadang kompleks, dan cukup fleksibel untuk menggabungkan prediksi secara efektif [18]. Digunakan algoritma untuk meta learner, yaitu:

a) Logistic Regression

Logistic Regression (LR) sebagai meta learner karena sifatnya yang linier, sederhana, dan stabil. LR mampu mencegah overfitting pada meta-features yang kompleks serta mencari kombinasi bobot paling optimal dari hasil prediksi base learners. Selain itu, model ini mudah diinterpretasikan karena setiap koefisien menunjukkan kontribusi masing-masing base learner terhadap prediksi akhir. Beberapa penelitian menunjukkan bahwa LR merupakan meta learner yang umum digunakan karena performanya yang konsisten dan stabil dalam metode stacking [19][20].

E. Mekanisme Cross-Validation Stacking

Stacking ensemble pada penelitian ini dibangun menggunakan Random Forest, Gradient Boosting, dan XGBoost sebagai base learners, serta Logistic Regression sebagai meta learner. Metode yang digunakan adalah cross-validation stacking dengan 5-fold cross-validation, di mana meta learner dilatih menggunakan prediksi *out-of-fold* dari base learners agar proses pelatihan tetap independen dan menghindari *data leakage*. sehingga setiap prediksi yang diterima oleh meta learner berasal dari sampel yang tidak digunakan saat pelatihan base learners.

Pada metode stacking, data leakage dapat terjadi apabila meta learner dilatih menggunakan prediksi dari base learner yang telah melihat sampel yang sama saat pelatihan. Kondisi ini dapat menyebabkan performa model tampak lebih baik daripada kemampuan sebenarnya [21]. Untuk mencegahnya, penelitian ini menggunakan mekanisme out-of-fold predictions, sehingga meta learner belajar dari prediksi yang benar-benar independent, dan informasi dari data training base learner tidak bocor ke meta learner. Data testing berjumlah 101 sampel tidak terlibat dalam proses pelatihan dan hanya digunakan untuk evaluasi akhir.

F. Evaluasi Model

Evaluasi dilakukan untuk menilai seberapa baik model dapat mengklasifikasikan atau memprediksi data baru yang belum pernah dilihat sebelumnya. Tanpa evaluasi yang tepat, sulit untuk menentukan apakah model yang dikembangkan benar-benar memberikan hasil yang akurat dan dapat diandalkan [22]. Beberapa metrik evaluasi yang digunakan dalam prediksi, yaitu:

1) Accuracy

Total keseluruhan yang menunjukkan seberapa sering model klasifikasi memberikan prediksi.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (1)$$

2) Precision

Rasio yang mengukur seberapa tepat model memprediksi positif, seberapa sering prediksi benar.

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

3) Recall
Seberapa baik model dalam memprediksi data yang benar-benar termasuk dalam kelas positif.

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

4) F1-Score
Rata-rata harmonik yang diperoleh dari Precision dan Recall.

$$F1 - Score = 2x\frac{Precision\ x\ Recall}{Precision + Recall} \quad (4)$$

5) ROC-AUC
Mengukur seberapa baik model dalam membedakan positif dan negatif.

$$PR\ (Recall) = \frac{TP}{TP + FN} \quad (5)$$

$$FPR\ = \frac{FP}{FP + TN} \quad (6)$$

G. Analisis Overfitting

Analisis overfitting dilakukan untuk memastikan bahwa performa tinggi model tidak disebabkan oleh kemampuan model menghafal data latih (memorization) tetapi oleh kemampuan generalisasi yang baik terhadap data baru. Evaluasi dilakukan dengan membandingkan hasil performa pada data training dan data testing menggunakan metrik akurasi, precision, recall, F1-score, dan ROC AUC. Perbedaan nilai yang kecil antara kedua dataset menunjukkan model yang stabil dan tidak mengalami overfitting [23].

Selain itu, digunakan strategi 5-fold cross-validation stacking untuk memastikan bahwa setiap data pada tahap pelatihan dan validasi bersifat independen. Regularisasi pada algoritma Logistic Regression sebagai meta learner juga membantu menekan kompleksitas model agar tidak terlalu menyesuaikan diri terhadap data latih. Jika perbedaan performa antara training dan testing set melebihi ambang batas 2–3%, maka model akan dievaluasi ulang untuk mengidentifikasi kemungkinan overfitting.

H. Analisis Feature Importance (khusus Random Forest)

Untuk mengidentifikasi faktor-faktor yang paling berpengaruh terhadap prediksi depresi mahasiswa, dilakukan analisis feature importance menggunakan algoritma Random Forest, yang merupakan salah satu base learner dalam model stacking ensemble. Random Forest dipilih karena kemampuannya menghitung tingkat kepentingan fitur berdasarkan kontribusinya dalam mengurangi impurity (Gini Importance) pada setiap node pohon keputusan [24].

Proses analisis dilakukan dengan menghitung rata-rata penurunan impurity dari setiap fitur pada seluruh pohon dalam model. Nilai importance dinormalisasi agar totalnya mencapai 100%, sehingga dapat diketahui proporsi pengaruh relatif setiap variabel terhadap hasil prediksi. Fitur dengan nilai penting tertinggi dianggap memiliki kontribusi terbesar dalam memengaruhi klasifikasi depresi.

III. HASIL DAN PEMBAHASAN

A. Dataset

Dataset yang digunakan dalam penelitian ini merupakan data publik yang diperoleh dari platform Kaggle melalui tautan https://www.kaggle.com/datasets/ikynahidwin/depressionstunt-dataset dan terdiri dari 502 data. Dataset ini terdapat 11 atribut, yang terdiri dari 10 variabel independent dan 1 variabel dependen (target). Variabel independen mencakup jenis kelamin, umur, tekanan akademik, kepuasan bealajar, durasi tidur, kebiasaan makan, pemikiran bunuh diri, durasi belajar, stress finansial, riwayat keluarga dengan gangguan mental. Sedangkan, variabel dependen adalah Depression, yang menunjukkan mahasiswa mengalami depresi (1) atau tidak (0). Gambar 2 menampilkan sebagian contoh struktur data yang digunakan dalam penelitian ini.

	Gender	Age	Academic Pressure	Study Satisfaction	Sleep Duration
0	Male	28	2	4	7-8 hours
1	Male	28	4	5	5-6 hours
2	Male	25	1	3	5-6 hours
3	Male	23	1	4	More than 8 hours
4	Female	31	1	5	More than 8 hours

Dietary Habits	Have you ever had suicidal thoughts ?	Study Hours	Financial Stress	Family History of Mental Illness	Depression
Moderate	Yes	9	2	Yes	No
Healthy	Yes	7	1	Yes	No
Unhealthy	Yes	10	4	No	Yes
Unhealthy	Yes	7	2	Yes	No

Gambar 2. Sebagian data pada Dataset

B. Hasil Pre-Processing Data

1) Pengecekan Nilai Kosong (Missing Value)

Pemeriksaan menggunakan fungsi df.isnull().sum() menunjukkan bahwa seluruh fitur tidak memiliki missing value, sehingga data dapat digunakan secara penuh tanpa proses imputasi maupun penghapusan sampel. Gambar 3 merupakan hasil pengecekan missing value.

```
Jumlah missing value per kolom:
Gender                                  0
Age                                     0
Academic Pressure                       0
Study Satisfaction                      0
Sleep Duration                          0
Dietary Habits                          0
Have you ever had suicidal thoughts ?   0
Study Hours                             0
Financial Stress                        0
Family History of Mental Illness        0
Depression                              0
dtype: int64
```

Gambar 3. Hasil Pengecekan Missing Value

2) One-hot Encoding

Fitur kategorikal seperti Gender, Sleep Duration, Dietary Habits, Suicidal Thoughts, dan Family History dikonversi menjadi representasi numerik melalui teknik one-hot encoding agar dapat diproses oleh algoritma machine learning tanpa menimbulkan hubungan urutan antar kategori. Gambar 4 merupakan hasil Encoding kolom kategorikal.

Gender_Male	Sleep Duration_7-8 hours	Sleep Duration_Less than 5 hours
0.0	0.0	0.0
1.0	0.0	0.0
1.0	0.0	1.0
0.0	0.0	0.0
1.0	0.0	0.0

Sleep Duration_More than 8 hours	Dietary Habits_Moderate	Dietary Habits_Unhealthy
1.0	1.0	0.0
1.0	0.0	1.0
0.0	1.0	0.0
0.0	0.0	0.0
1.0	1.0	0.0

Have you ever had suicidal thoughts ?_Yes	Family History of Mental Illness_Yes
0.0	0.0
1.0	0.0
0.0	0.0
0.0	0.0
1.0	1.0

Gambar 4. Hasil Encoding kolom kategorikal

3) Scaling/Normalisasi fitur numerik

Fitur numerik seperti Age, Academic Pressure, Study Satisfaction, Study Hours, dan Financial Stress dinormalisasi menggunakan StandardScaler untuk menyamakan skala nilai, sehingga tidak ada fitur yang mendominasi proses pelatihan model dan proses komputasi menjadi lebih optimal. Gambar 5 menampilkan hasil normalisasi fitur numerik.

Age	Academic Pressure	Study Satisfaction	Study Hours	Financial Stress
0.376157	0.734357	-0.793329	-1.691083	0.037079
0.580270	0.734357	-0.075214	-1.422838	0.037079
1.396720	0.734357	-1.511443	-0.618102	1.453148
1.600832	-0.688238	1.361015	1.259616	0.745113
0.580270	0.023060	-1.511443	-0.618102	-1.378989

Gambar 5. Hasil normalisasi fitur numerik

Hasil normalisasi menunjukkan bahwa fitur numerik telah diubah ke dalam rentang standar dengan rata-rata mendekati 0 dan standar deviasi mendekati 1. Nilai-nilai positif menunjukkan variabel di atas rata-rata populasi, sedangkan nilai negatif menunjukkan di bawah rata-rata. Normalisasi ini memastikan semua fitur numerik memiliki kontribusi yang seimbang dalam proses pelatihan model.

4) Label Encoding

Label target Depression diubah dari nilai kategorikal (“Yes”,”No”) menjadi numerik (1/0) agar dapat diproses dalam klasifikasi biner. Tabel 4 menampilkan hasil encoding kolom target pada sebagian dataset.

TABEL 3
HASIL SPLIT DATA

Target	Keterangan
0	No
0	Yes
1	No
0	No
0	No

Seluruh proses encoding dan normalisasi diintegrasikan menggunakan ColumnTransformer sehingga penerapan langkah pra-premosesan berlangsung secara konsisten baik pada data latih maupun data uji.

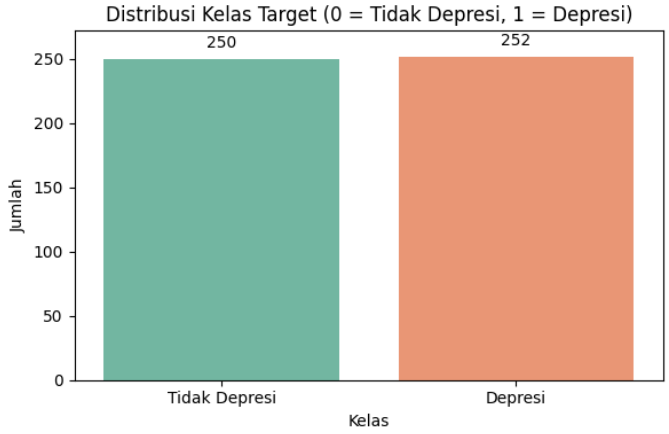
C. Hasil Pembagian Data

Dataset kemudian dibagi menjadi 80% data latih (401 sampel) dan 20% data uji (101 sampel). Proporsi ini dipilih untuk menjaga keseimbangan antara pelatihan dan evaluasi model. di mana hasil split data tersebut disajikan pada Tabel 4. Komposisi kelas yang seimbang pada data latih membantu mencegah bias model terhadap salah satu kelas.

TABEL 4
HASIL SPLIT DATA

Data	Length
Xtrain	401
Xtest	101

Distribusi target kelas Depression pada data latih juga cukup seimbang, yaitu 252 data pada kelas Depresi (label 1) dan 250 data pada kelas Tidak Depresi (label 0). Keseimbangan ini menjadi faktor penting dalam mencegah bias prediksi model terhadap salah satu kelas dan memastikan performa klasifikasi yang lebih adil. Hasil dari distribusi data pada Gambar 6.



Gambar 6. Distribusi kelas target 'Depression'

D. Komponen Stacking

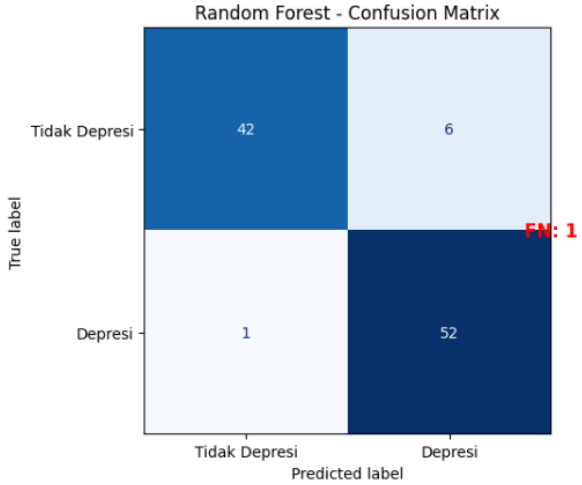
Proses pelatihan model dilakukan menggunakan pendekatan stacking ensemble dua tingkat. Pada tahap ini digunakan tiga algoritma sebagai base learner, yaitu Random Forest (RF), Gradient Boostingg (GB), dan XGBoost (XGB), sedangkan Logistic Regression (LR) digunakan sebagai meta learner.

1) Base Model (level-0)

Sebelum proses penggabungan dilakukan, ketiga base learner diuji secara individual untuk mengetahui performanya masing-masing terhadap data pelatihan.

a) Random Forest

Model Random Forest memperoleh akurasi 93% dengan performa yang seimbang antar kelas. Pada kelas Tidak Depresi, nilai precision 98% dan recall 88% menunjukkan prediksi yang cukup tepat meski masih ada beberapa kesalahan deteksi. Sementara itu, kelas Depresi memiliki precision 90% dan recall 98%, menandakan kemampuan tinggi dalam mengenali individu depresi. Nilai macro dan weighted average 93% mengindikasikan kinerja stabil tanpa bias terhadap salah satu kelas. Berdasarkan ROC curve dengan AUC 99.4% dan Precision–Recall curve dengan AP 99.5%, model ini menunjukkan kemampuan klasifikasi yang sangat baik dengan keseimbangan tinggi antara precision dan recall. Gambar 7 merupakan hasil confusion matrix untuk model Random Forest sebelum dilakukan stacking.



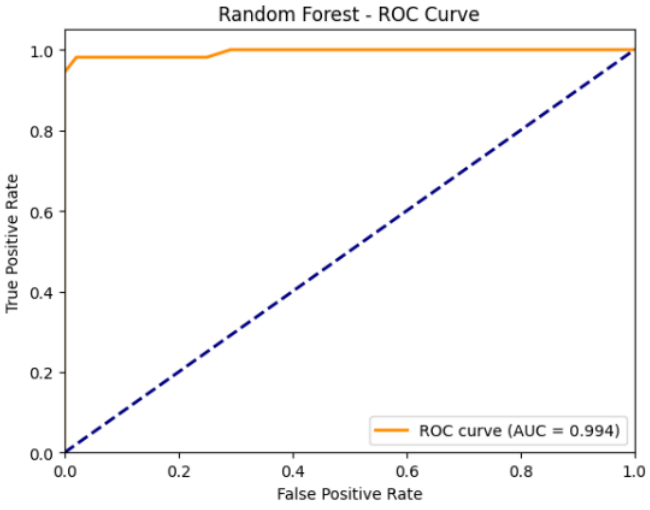
Gambar 7. Confusion Matrix Random Forest

Berdasarkan Confusion Matrix, maka didapatkan akurasi, recall, precision, dan F1-score sebagaimana tertera pada Tabel 5.

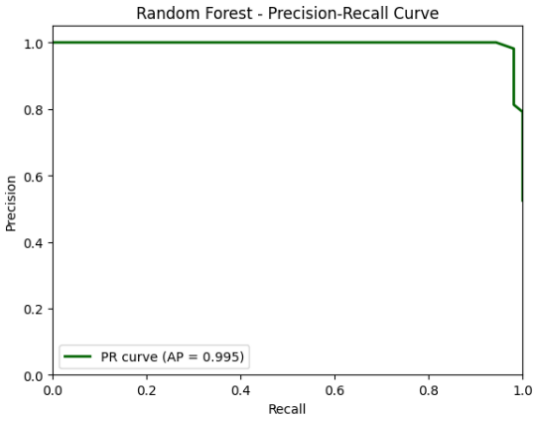
TABEL 5
HASIL EVALUASI RANDOM FOREST

METRIK EVALUASI	NILAI
Confussion Matrix	[[42, 6], [1,52]]
Accuracy	93%
Recall	98%
Precision	90%
F1-score	94%

Gambar 8 dan Gambar 9 menunjukkan kurva ROC dan Precision-Recall dari pemodelan Random Forest.



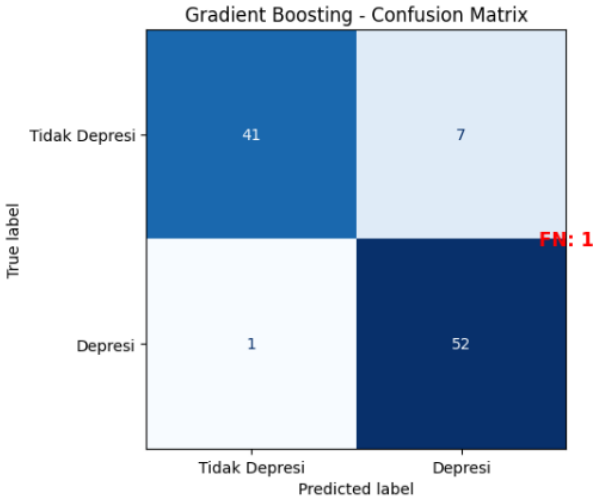
Gambar 8. Kurva ROC Random Forest



Gambar 9. Kurva Precision-Recall Random Forest

b) Gradient Boosting

Model Gradient Boosting mencatat akurasi 92% dengan performa yang sedikit lebih rendah dibanding Random Forest. Kelas Tidak Depresi memiliki precision 98% dan recall 85%, sedangkan kelas Depresi menunjukkan precision 88% dan recall 98%. Hasil ini menandakan model lebih sensitif terhadap kelas depresi dan cenderung melewatkan sebagian kecil kasus Tidak Depresi. Nilai rata-rata 92% menunjukkan kinerja yang tetap konsisten dan seimbang. Nilai AUC 99% pada ROC curve dan AP 99.2% pada Precision–Recall curve memperkuat bahwa model memiliki kemampuan generalisasi yang baik dalam membedakan kedua kelas. Gambar 10 merupakan hasil confusion matrix untuk model Random Forest sebelum dilakukan stacking.



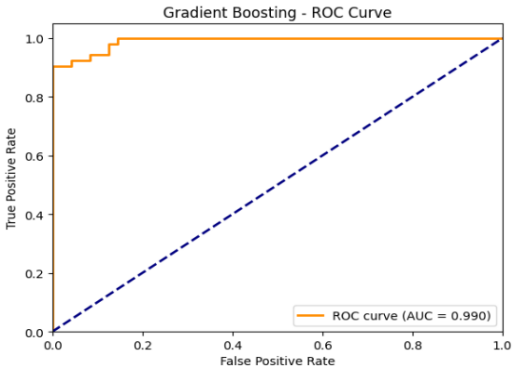
Gambar 10. Confusion Matrix Gradient Boosting

Berdasarkan Confusion Matrix, maka didapatkan akurasi, recall, precision, dan F1-score sebagaimana tertera pada Tabel 6.

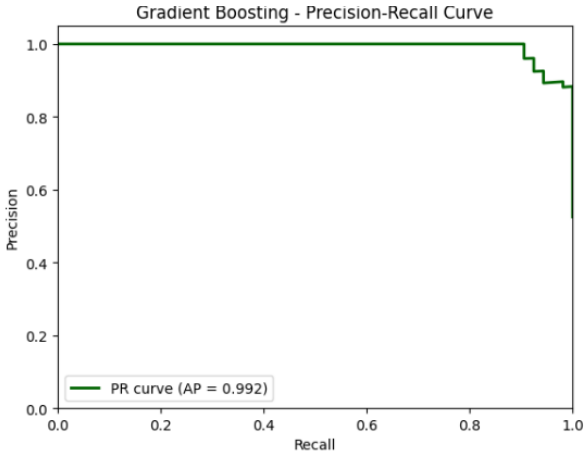
TABEL 6
HASIL EVALUASI GRADIENT BOOSTING

METRIK EVALUASI	NILAI
Confussion Matrix	[[41, 7], [1,52]]
Accuracy	92%
Recall	98%
Precision	88%
F1-score	943%

Gambar 11 dan Gambar 12 menunjukkan kurva ROC dan Precision-Recall dari pemodelan Gradient Boosting.



Gambar 11. Kurva ROC Gradient Boosting

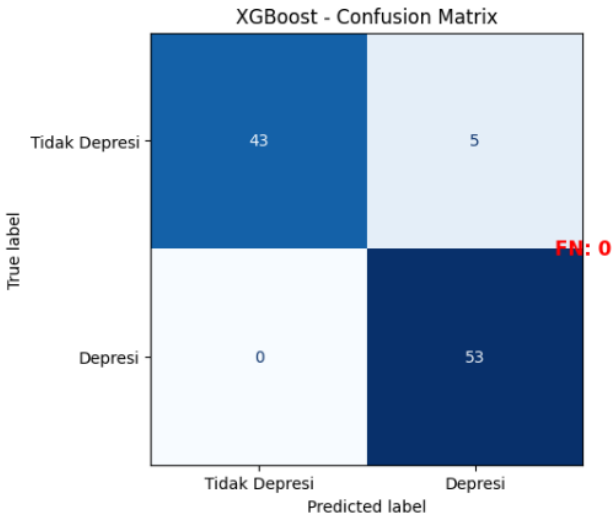


Gambar 12. Kurva Precision-Recall Gradient Boosting

c) XGBoost

Model XGBoost memberikan hasil terbaik dengan akurasi 95%. Kelas Tidak Depresi mencapai precision 100% dan recall 90%, sementara kelas Depresi memiliki precision 91% dan recall 100%, menunjukkan deteksi sempurna terhadap seluruh kasus depresi. Nilai macro dan weighted average sebesar 95% menegaskan kinerja unggul dan seimbang di kedua kelas. Hasil ROC curve dengan AUC 99.5% dan Precision–Recall curve dengan AP 99.6% menunjukkan kemampuan klasifikasi yang hampir sempurna serta kestabilan model dalam menjaga keseimbangan antara

precision dan recall. Gambar 13 merupakan hasil confusion matrix untuk model Random Forest sebelum dilakukan stacking.



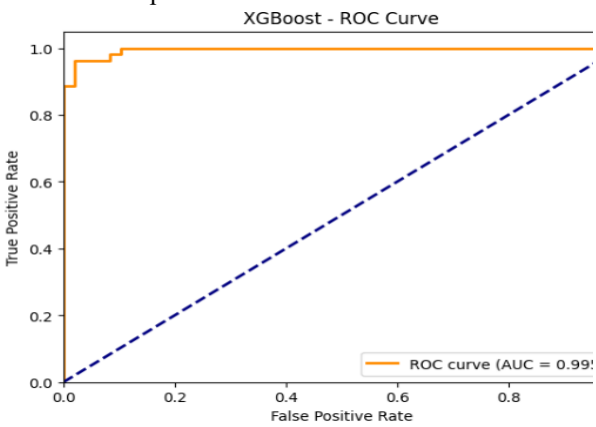
Gambar 13. Confusion Matrix XGBoost

Berdasarkan Confusion Matrix, maka didapatkan akurasi, recall, precision, dan F1-score sebagaimana tertera pada Tabel 7.

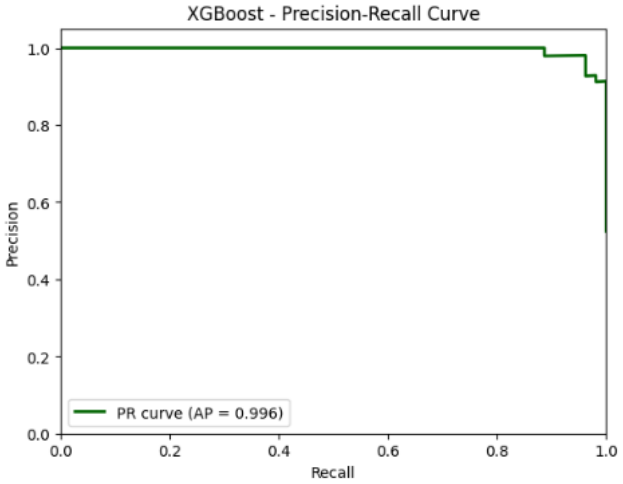
TABEL 7
HASIL EVALUASI XGBOOST

METRIK EVALUASI	NILAI
Confussion Matrix	[[43, 5], [0,53]]
Accuracy	95%
Recall	100%
Precision	91%
F1-score	95%

Gambar 14 dan Gambar 15 menunjukkan kurva ROC dan Precision-Recall dari pemodelan XGBoost.



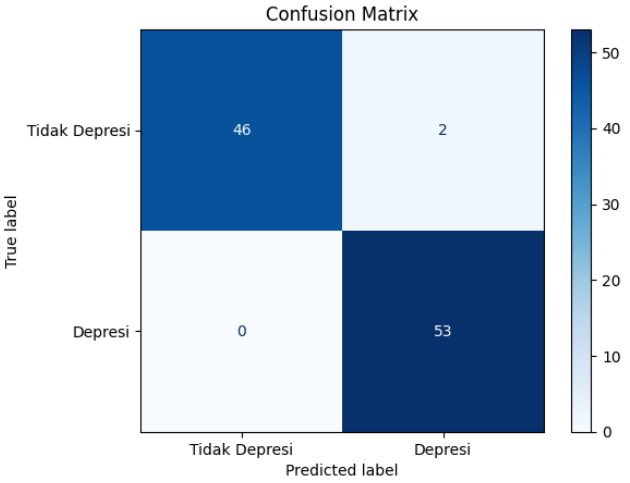
Gambar 14. Kurva ROC XGBoost



Gambar 15. Kurva Precision-Recall XGBoost

2) Meta Learner dan Mekanisme Stacking Ensemble menggunakan Cross Validation

Model stacking ensemble menunjukkan performa yang sangat baik dengan akurasi sebesar 98.02%, serta keseimbangan performa antar kelas yang tinggi. Pada kelas tidak depresi (label 0), diperoleh precision sebesar 100% dan recall 96%, sedangkan pada kelas depresi (label 1), precision mencapai 96% dan recall 100%. Hasil ini diperoleh melalui penerapan mekanisme *cross-validation stacking*, di mana meta learner dilatih menggunakan prediksi *out-of-fold* dari base learners agar proses pelatihan tetap independen dan menghindari *data leakage*. Gambar 16 merupakan hasil confusion matrix setelah dilakukan stacking ensemble.



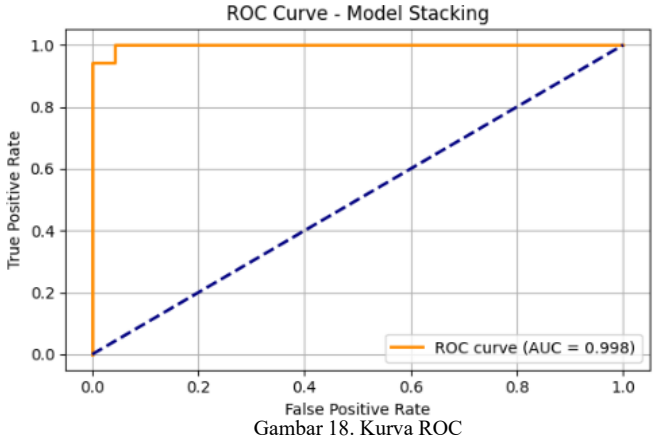
Gambar 16. Confusion Matrix Model Stacking Ensemble

Berdasarkan Confusion Matrix, maka didapatkan akurasi, recall, precision, dan F1-score sebagaimana tertera pada Gambar 17.

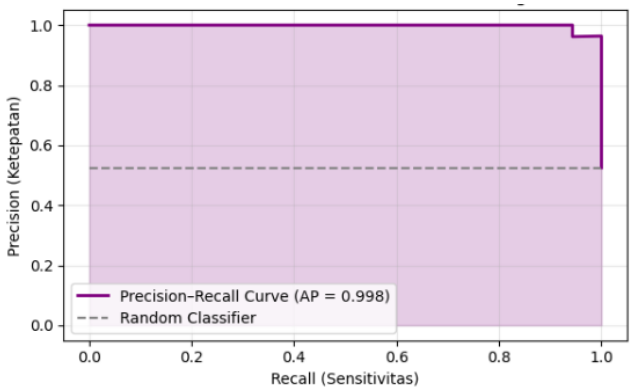
```
Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.96      0.98        48
           1       0.96      1.00      0.98        53

    accuracy                           0.98       101
   macro avg       0.98      0.98      0.98       101
weighted avg       0.98      0.98      0.98       101

Accuracy: 0.9801980198019802
```

Gambar 17. Hasil Evaluasi Stacking Ensemble

Hasil *ROC curve* (AUC = 99.8%) dan *Precision–Recall curve* (AP = 99.8%) menunjukkan kemampuan model yang sangat baik dalam membedakan kelas serta menjaga keseimbangan antara *precision* dan *recall*, menegaskan konsistensi performa tinggi model stacking. Gambar 18 dan Gambar 19 merupakan hasil kurva ROC dan kurva Precision-Recall model stacking ensemble.



Gambar 18. Kurva ROC



Gambar 19. Kurva Precision-Recall

E. *Perbandingan Kinerja Model*

Bagian ini membahas perbandingan kinerja antara model *Stacking Ensemble* dengan model pembanding lainnya untuk menilai efektivitas pendekatan yang diusulkan.

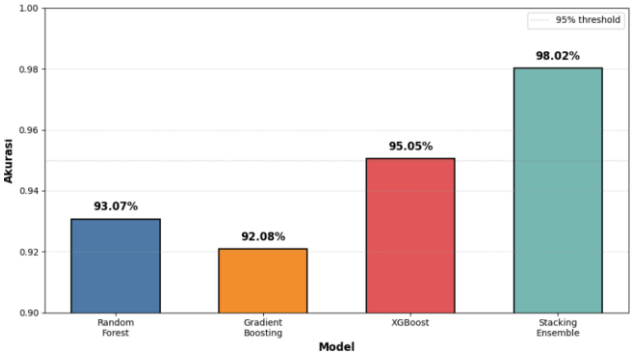
1) *Perbandingan Stacking Ensemble dengan Base Learners*

Hasil evaluasi menunjukkan bahwa stacking ensemble memberikan peningkatan performa yang konsisten dibandingkan dengan model tunggal. Seperti ditunjukkan pada Tabel 8 dan Gambar 20, model stacking mencapai akurasi 98.02%, precision 96.36%, recall 100%, dan F1-score 98.15%, sedangkan model tunggal terbaik (XGBoost) hanya mencapai akurasi 95.05%, yang berarti terdapat peningkatan sebesar 2.97%.

Keberhasilan peningkatan kinerja ini divalidasi oleh mekanisme Cross-Validation Stacking (CVS). CVS sangat penting untuk mencegah data leakage, yaitu kondisi di mana Meta Learner dilatih menggunakan prediksi yang dihasilkan oleh Base Learner dari sampel yang telah dilihat selama pelatihan. Dalam penelitian ini, CVS memastikan bahwa Meta Learner hanya menerima Prediksi Out-of-Fold (OOF), yaitu prediksi yang berasal dari sampel data yang belum pernah dilihat oleh Base Learner.

TABEL 8
HASIL PERBANDINGAN PERFORMA MODEL TUNGGAL DAN STACKING ENSEMBLE (SE)

Model	Accuracy	Precision	Recall	F1-Score	ROC AUC
RF	93.07%	89.66%	98.11%	93.69%	99.45%
GB	92.08%	88.14%	98.11%	92.86%	99.02%
XGB	95.05%	91.38%	100.0%	95.5%	99.49%
SE	98.02%	96.36%	100.0%	98.15%	99.76%



Gambar 20. Perbandingan Akurasi Model Tunggal dan Stacking Ensemble

2) *Perbandingan Base Learners, Stacking Ensemble, dan Bagging*

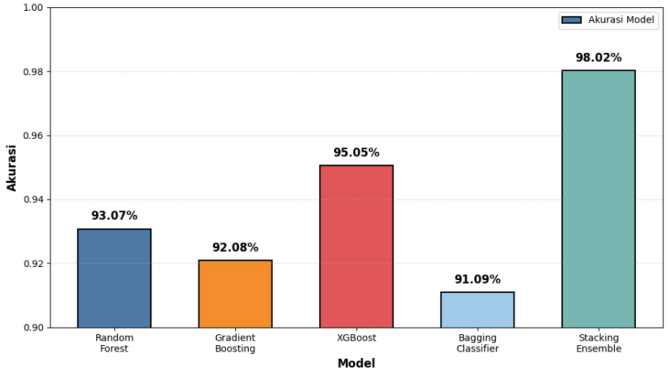
Selain itu, untuk memperluas pembahasan, dilakukan pula perbandingan antara Base Learners, Stacking Ensemble (SE) dengan model Bagging Classifier (BC) yang mewakili pendekatan ensemble sederhana berbasis voting. Hasil perbandingan pada Tabel 9 dan Gambar 21 menunjukkan bahwa SE tetap unggul signifikan, dengan akurasi 98.02%, dibandingkan BC yang hanya mencapai 91.09%.

Perbedaan ini menunjukkan bahwa stacking lebih unggul karena mampu mengombinasikan model yang heterogen dan

melakukan pembelajaran tingkat kedua melalui meta learner, bukan sekadar melakukan voting seperti pada bagging. Oleh karena itu, metode stacking ensemble dapat dinyatakan sebagai pendekatan yang paling efektif dalam penelitian ini untuk klasifikasi depresi pada mahasiswa.

TABEL 9
HASIL PERBANDINGAN PERFORMA BASE LEARNERS, STACKING ENSEMBLE (SE), DAN BAGGING CLASSIFIER (BC)

Model	Accuracy	Precision	Recall	F1-Score	ROC AUC
RF	93.07%	89.66%	98.11%	93.69%	99.45%
GB	92.08%	88.14%	98.11%	92.86%	99.02%
XGB	95.05%	91.38%	100.0%	95.5%	99.49%
BC	91.09%	89.29%	94.34%	91.74%	95.54%
SE	98.02%	96.36%	100.0%	98.15%	99.76%



Gambar 21. Perbandingan Akurasi Base Learners, Bagging, dan Stacking Ensemble

F. *Analisis Overfitting*

Karena performa model sangat tinggi (akurasi 98.02% dan ROC AUC 99.8%), dilakukan analisis mendalam untuk memastikan bahwa hasil tersebut tidak disebabkan oleh overfitting atau bias data. Untuk mengidentifikasi potensi overfitting, dilakukan perbandingan performa antara training set dan testing set, sebagaimana ditunjukkan pada Tabel 10. Perbedaan performa antara keduanya relatif kecil (selisih <2% untuk sebagian besar metrik), yang menandakan tidak terjadi overfitting signifikan. Nilai recall 100% pada kedua dataset juga menunjukkan konsistensi model dalam mendeteksi seluruh kasus depresi tanpa adanya false negative.

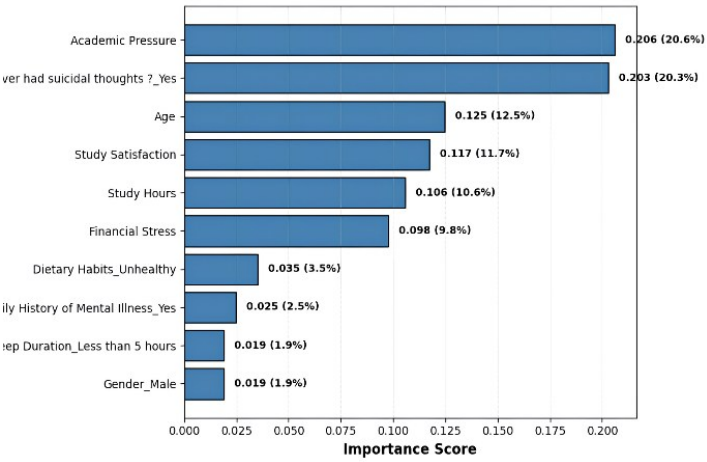
Performa yang stabil ini dipengaruhi oleh penggunaan strategi 5-fold cross-validation stacking, regularisasi pada Logistic Regression sebagai meta learner, serta kompleksitas model yang terkendali. Meskipun performa training set sempurna (100%), hasil pengujian menunjukkan model tetap generalis dan tidak hanya menghafal pola data pelatihan. Validasi lebih lanjut menggunakan dataset eksternal tetap diperlukan untuk memperkuat bukti generalisasi model.

TABEL 10
PERBANDINGAN PERFORMA TRAINING SET DAN TESTING SET

Metrik	Training Set	Testing Set	Selisih
Accuracy	100.00%	98.02%	1.98%
Precision	100%	96%	3.64%
Recall	100%	100%	0.00%
F1-Score	100%	98%	1.85%
ROC AUC	100.0%	99.8%	0.24%

G. Analisis Feature Importance

Untuk memahami faktor-faktor yang paling berpengaruh terhadap prediksi depresi pada mahasiswa, dilakukan analisis *feature importance* menggunakan algoritma Random Forest.



Gambar 22. Feature Importance

Hasil analisis pada Gambar 22 menunjukkan bahwa Academic Pressure (20.6%), Suicidal Thoughts (20.3%), dan Age (12.5%) merupakan faktor paling dominan dalam menentukan prediksi depresi mahasiswa. Faktor lain seperti Financial Stress (9.8%) dan Study Satisfaction (11.7%) juga berkontribusi meski dalam tingkat lebih rendah, mencerminkan peran tekanan ekonomi dan kepuasan belajar terhadap stabilitas emosional. Secara keseluruhan, hasil ini menunjukkan bahwa risiko depresi dipengaruhi oleh kombinasi faktor akademik, psikologis, dan sosial-ekonomi, sehingga diperlukan strategi pencegahan terpadu melalui konseling akademik, pelatihan manajemen stres, dan dukungan finansial bagi mahasiswa.

IV. KESIMPULAN

Penelitian ini mengusulkan model prediksi depresi mahasiswa menggunakan pendekatan stacking ensemble dua tingkat dengan Random Forest, Gradient Boosting, dan

XGBoost sebagai base learners, serta Logistic Regression sebagai meta learner. Penerapan cross-validation stacking terbukti efektif dalam mencegah data leakage dan meningkatkan kemampuan generalisasi model. Hasil evaluasi menunjukkan performa sangat baik dengan akurasi 98.02%, nilai precision dan recall di atas 96%, serta AUC 99.8%, menandakan kemampuan klasifikasi yang hampir sempurna. Analisis feature importance mengungkap bahwa Academic Pressure, Suicidal Thoughts, dan Age merupakan faktor dominan yang memengaruhi risiko depresi, sejalan dengan kondisi nyata di lingkungan kampus di mana tekanan akademik dan beban psikologis sering menjadi pemicu utama menurunnya kesejahteraan mental mahasiswa.

Meskipun performa model tinggi, penelitian ini memiliki keterbatasan pada ukuran dataset yang kecil (502 data) dan sumber data tunggal (Kaggle), sehingga generalisasi hasil masih terbatas. Penelitian selanjutnya disarankan untuk menggunakan dataset yang lebih besar dan beragam serta mengeksplorasi model lanjutan seperti deep learning, semi-supervised learning, atau explainable AI (XAI) agar prediksi lebih akurat dan transparan. Selain itu, model ini berpotensi dikembangkan menjadi sistem deteksi dini berbasis kampus yang terintegrasi dengan sistem informasi akademik atau survei kesehatan mental online untuk membantu identifikasi dan intervensi dini mahasiswa berisiko depresi.

DAFTAR PUSTAKA

[1] K. Vitoasmara, F. Vio Hidayah, R. Yuna Aprillia, and L. A. Dyah Dewi, "Gangguan Mental (Mental Disorders)," *Student Res. J.*, no. 2, pp. 57–68, 2024, [Online]. Available: https://doi.org/10.55606/srjyappi.v2i3.1219

[2] I. Setiawan, I. F. Yasin, Y. T. Desianti, and A. Surakarta, "Komparasi Kinerja Algoritma Random Forest , Decision Tree , Naïve Bayes , dan KNN dalam Prediksi Tingkat Depresi Mahasiswa Menggunakan Student Depression Dataset," vol. 6, no. 1, pp. 47–58, 2025.

[3] I. Zulfahmi, H. Syahputra, S. I. Naibaho, M. A. Maulana, and E. P. Sinaga, "Perbandingan Algoritma Support Vector Machine (SVM) dan Decision Tree Untuk Deteksi Tingkat Depresi Mahasiswa," *Bina Insa. Ict J.*, vol. 10, no. 1, p. 52, 2023, doi: 10.51211/biict.v10i1.2304.

[4] S. N. Abdussamad, N. P. Doholio, W. P. Lasaleng, and P. Ayu, "Klasifikasi Tingkat Depresi Mahasiswa Menggunakan Image Recognition dengan Support Vector Machine," vol. 4, no. 1, pp. 30–36, 2025, doi: 10.55657/rmns.v4i1.193.

[5] D. K. Widyatna, T. Sagirani, and N. Wahyuningtyas, "Sistem Pakar Berbasis Android untuk Mengetahui Tingkat Depresi Dini Mahasiswa Menggunakan Metode Dempster Shafer," vol. 09, no. 01, pp. 1–6, 2024.

[6] M. Rijal, F. Aziz, and S. Abasa, "Prediksi Depresi : Inovasi Terkini Dalam Kesehatan Mental Melalui Metode Machine Learning Depression Prediction : Recent Innovations in Mental Health Journal Pharmacy and Application," *J. Pharm. Appl. Comput. Sci.*, vol. 2, no. 1, pp. 9–14, 2024, [Online]. Available: https://doi.org/10.59823/jopacs.v2i1.47

[7] B. L. Schaab *et al.*, "How do machine learning models perform in the detection of depression, anxiety, and stress among undergraduate students? A systematic review," *Cad. Saude Publica*, vol. 40, no. 11, 2024, doi: 10.1590/0102-311XEN029323.

[8] U. Royal, "Stacking Ensemble Model Machine Learning Deteksi Dini Risiko Kesehatan Mental Di," vol. 4307, no. August, pp. 4256–4266, 2024.

[9] A. Daza Vergaray, J. C. H. Miranda, J. B. Cornelio, A. R. López Carranza, and C. F. Ponce Sánchez, "Predicting the depression in university students using stacking ensemble techniques over oversampling method," Inform. Med. Unlocked, vol. 41, p. 101295, 2023, doi: 10.1016/j.imu.2023.101295.

[10] A. Selvaraj and L. Mohandoss, "Enhancing Depression Detection: A Stacked Ensemble Model with Feature Selection and RF Feature Importance Analysis Using NHANES Data," *Appl. Sci.*, vol. 14, no. 16, 2024, doi: 10.3390/app14167366.

[11] M. E. Hasan *et al.*, "Prevalence, associated factors, and machine learning-based prediction of depression, anxiety, and stress among university students: a cross-sectional study from Bangladesh," *J. Health. Popul. Nutr.*, vol. 44, no. 1, p. 361, 2025, doi: 10.1186/s41043-025-01095-8.

[12] M. Peran, P. Kecerdasan, B. Kualitas, and G. Hermawan, "Memahami Peran Dataset dalam Penelitian Kecerdasan Buatan : Kualitas , Aksesibilitas , dan Tantangan," no. October, 2024, doi: 10.13140/RG.2.2.34468.49288.

[13] H. H. Arrosyid, Z. Pratama, and G. Priambodo, "Penurunan Cancellation Rate Pada City Hotel Menggunakan Metode Issue Tree," vol. 1, no. 1, pp. 31–39, 2025.

[14] V. Chernykh, A. Stepnov, and B. O. Lukyanova, "Data preprocessing for machine learning in seismology," *CEUR Workshop Proc.*, vol. 2930, no. October, pp. 119–123, 2023.

[15] T. Zhou and H. Jiao, "Exploration of the Stacking Ensemble Machine Learning Algorithm for Cheating Detection in Large-Scale Assessment," *Educ. Psychol. Meas.*, vol. 83, no. 4, pp. 831–854, 2023, doi: 10.1177/00131644221117193.

[16] Z. Shao, M. N. Ahmad, and A. Javed, "Comparison of Random Forest and XGBoost Classifiers Using Integrated Optical and SAR Features for Mapping Urban Impervious Surface," *Remote Sens.*, vol. 16, no. 4, 2024, doi: 10.3390/rs16040665.

[17] S. B. Nadkarni, G. S. Vijay, and R. C. Kamath, "Comparative Study of Random Forest and Gradient Boosting Algorithms to Predict Airfoil Self-Noise," *Eng. Proc.*, vol. 59, no. 1, 2023, doi: 10.3390/engproc2023059024.

[18] R. Kablan, H. A. Miller, S. Suliman, and H. B. Frieboes, "Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID- 19 . The COVID-19 resource centre is hosted on Elsevier Connect , the company ' s public news and information ," no. January, 2020.

[19] C. Yang, E. A. Fridgeirsson, J. A. Kors, J. M. Reps, P. R. Rijnbeek, and D. Ross, "An initial investigation into more complex stacking methods to improve transportability of prediction models developed across multiple databases," *Obs. Heal. Data Sci. Informatics (OHDSI)*, 2023.

[20] T. Tong and Z. Li, "Predicting learning achievement using ensemble learning with result explanation," *PLoS One*, vol. 20, no. 1, pp. 1–25, 2025, doi: 10.1371/journal.pone.0312124.

[21] M. Rosenblatt, L. Tejavibulya, R. Jiang, S. Noble, and D. Scheinost, "Data leakage inflates prediction performance in connectome-based machine learning models," *Nat. Commun.*, vol. 15, no. 1, pp. 1–15, 2024, doi: 10.1038/s41467-024-46150-w.

[22] M. R. Sudrajat and M. Zakariyah, "Penerapan Natural Language Processing dan Machine Learning untuk Prediksi Stres Siswa SMA Berdasarkan Analisis Teks," vol. 6, no. 3, 2024, doi: 10.47065/bits.v6i3.6180.

[23] P. Proskura and A. Zaytsev, "Effective Training-Time Stacking for Ensembling of Deep Neural Networks," *ACM Int. Conf. Proceeding Ser.*, pp. 78–82, 2022, doi: 10.1145/3573942.3573954.

[24] X. Yuan, S. Liu, W. Feng, and G. Dauphin, "Feature Importance Ranking of Random Forest-Based End-to-End Learning Algorithm," *Remote Sens.*, vol. 15, no. 21, pp. 1–20, 2023, doi: 10.3390/rs15215203.