

# Optimasi *Random Forest* Melalui *Feature Engineering* dan SMOTE untuk Klasifikasi Kesehatan Mental

Rovidatul Hikmah Tanjung<sup>1,\*</sup>, Fera Damayanti<sup>2</sup>, Ahmad Zaki<sup>3</sup>

<sup>12</sup> Teknik Komputer dan Informatika, Politeknik Negeri Medan, Medan, Indonesia

<sup>3</sup> Prodi Informatika, UIN Sjech M. Djamil Djambek Bukittinggi, Bukittinggi, Indonesia

Email: <sup>1,\*</sup>[rovidatulhikmah@polmed.ac.id](mailto:rovidatulhikmah@polmed.ac.id), <sup>2</sup>[feradamayanti@polmed.ac.id](mailto:feradamayanti@polmed.ac.id), <sup>3</sup>[ahmadzaki@uinbukittinggi.ac.id](mailto:ahmadzaki@uinbukittinggi.ac.id)

**Abstrak-** Kesehatan mental mahasiswa merupakan faktor krusial yang memengaruhi prestasi akademik, produktivitas, dan kualitas hidup secara menyeluruh di lingkungan perguruan tinggi. Tingginya prevalensi gangguan psikologis saat ini menuntut adanya sistem deteksi dini yang akurat guna memberikan penanganan yang tepat waktu dan efisien. Penelitian ini bertujuan mengembangkan model klasifikasi kesehatan mental mahasiswa dengan mengintegrasikan teknik *feature engineering* dan *Synthetic Minority Oversampling Technique* (SMOTE) pada algoritma *Random Forest*. Tahapan *feature engineering* dilakukan melalui pembentukan fitur komposit *Mental\_Score* untuk merepresentasikan kondisi psikologis mahasiswa secara lebih holistik dan mendalam. Selain itu, teknik SMOTE diterapkan untuk mengatasi masalah ketidakseimbangan data agar model lebih sensitif dalam mendeteksi kelompok mahasiswa yang berisiko sebagai kelas minoritas. Hasil eksperimen menunjukkan bahwa model yang diusulkan mampu mencapai akurasi sebesar 97%. Penerapan SMOTE terbukti efektif meningkatkan *recall* kelas minoritas menjadi 60% serta menaikkan *F1-score* dari 0,57 ke 0,75, yang memperkuat kemampuan deteksi pada kelompok berisiko secara signifikan. Meskipun uji McNemar menghasilkan nilai *p-value* = 1,000 akibat adanya *ceiling effect* karena kedua model sudah sama-sama optimal, model usulan tetap memberikan keunggulan praktis dalam menjaga sensitivitas deteksi. Analisis *feature importance* mengonfirmasi bahwa *Mental\_Score* merupakan atribut paling berpengaruh dengan nilai kontribusi sebesar 0,3280. Penelitian ini berkontribusi dalam menyediakan kerangka kerja deteksi dini kesehatan mental mahasiswa berbasis *machine learning* yang lebih akurat.

**Kata Kunci:** *Random Forest*; Kesehatan Mental; *Feature Engineering*; SMOTE; Klasifikasi

**Abstract-** Student mental health is a crucial factor affecting academic performance, productivity, and overall quality of life in university environments. The high prevalence of psychological disorders today demands an accurate early detection system to provide timely and efficient intervention. This study aims to develop a student mental health classification model by integrating feature engineering techniques and the Synthetic Minority Oversampling Technique (SMOTE) with the Random Forest algorithm. The feature engineering stage is conducted through the creation of a composite feature, *Mental\_Score*, to represent students' psychological conditions more holistically and deeply. In addition, SMOTE is applied to address the data imbalance issue, making the model more sensitive in detecting the at-risk student group as the minority class. Experimental results show that the proposed model achieves an accuracy of 97%. The application of SMOTE proved effective in increasing the minority class recall to 60% and raising the F1-score from 0.57 to 0.75, significantly strengthening the detection capability for the at-risk group. Although the McNemar test yields a *p-value* of 1.000 due to a ceiling effect since both models are already optimal, the proposed model still offers a practical advantage in maintaining detection sensitivity. Feature importance analysis confirms that *Mental\_Score* is the most influential attribute with a contribution value of 0.3280. This study contributes to providing a more accurate machine learning-based framework for the early detection of student mental health.

**Keywords:** *Random Forest*; Mental Health; *Feature Engineering*; SMOTE; Classification

## 1. PENDAHULUAN

Kesehatan mental mahasiswa merupakan salah satu aspek krusial yang mempengaruhi keberhasilan proses pembelajaran di perguruan tinggi. Berdasarkan laporan World Health Organization [1], gangguan mental seperti depresi dan kecemasan menjadi penyebab utama disabilitas di kalangan dewasa muda, termasuk mahasiswa. Data epidemiologi menunjukkan bahwa sekitar 20–30% mahasiswa di berbagai negara mengalami setidaknya satu episode gangguan mental selama masa studinya. Tingginya tekanan akademik, tuntutan sosial, kondisi ekonomi, serta perubahan lingkungan seringkali menyebabkan mahasiswa mengalami stres, kecemasan, bahkan depresi [2]. Apabila kondisi tersebut tidak ditangani dengan baik, maka dapat berdampak pada penurunan prestasi akademik, rendahnya motivasi belajar, hingga gangguan kesehatan mental yang lebih serius yang berpotensi menyebabkan putus kuliah. Lebih lanjut, mahasiswa dengan gangguan mental yang tidak terdeteksi cenderung memiliki Indeks Prestasi Kumulatif lebih rendah, tingkat absensi tinggi, dan kesulitan dalam menyelesaikan tugas akhir [3].

Masalah utama yang dihadapi saat ini adalah kurangnya sistem deteksi dini yang mampu mengidentifikasi mahasiswa berisiko gangguan mental secara akurat, cepat, dan efisien. Metode konvensional seperti wawancara klinis atau kuesioner manual memerlukan waktu, tenaga profesional, dan partisipasi aktif mahasiswa. Seringkali, mahasiswa enggan melapor karena stigma negatif atau kurangnya kesadaran akan kondisinya [4]. Akibatnya, intervensi baru diberikan ketika kondisi sudah parah. Oleh karena itu, diperlukan solusi berbasis teknologi yang dapat menganalisis berbagai indikator psikologis dan perilaku secara otomatis, tidak mengganggu, dan dapat diintegrasikan dengan data administrasi kampus. Perkembangan *machine learning* membuka peluang besar untuk membangun model prediktif yang dapat mengklasifikasikan kondisi kesehatan mental berdasarkan data yang tersedia [5].

Salah satu algoritma yang sangat populer dalam klasifikasi data adalah *Random Forest*. Algoritma *ensemble learning* ini bekerja dengan membangun banyak pohon keputusan pada subset data yang berbeda dan menggabungkan prediksi

melalui mekanisme *voting* mayoritas [6]. Keunggulan *Random Forest* antara lain mampu menangani data berdimensi tinggi, tidak mudah *overfitting* karena menggunakan agregasi *bootstrap*, dapat menangkap hubungan non-linear antar fitur, menyediakan *feature importance* untuk interpretasi model, serta relatif robust terhadap *outlier* [7]. Penelitian terdahulu oleh Nurdiansyah et al. [5] membandingkan algoritma K-NN dan *Random Forest* pada dataset kesehatan mental mahasiswa dan melaporkan akurasi 85–90% tergantung pada pembagian data. Aubaidan et al. [4] membandingkan SMOTE dengan metode CSSF pada data tidak seimbang dan menunjukkan bahwa peningkatan kualitas sampel sintetis berdampak signifikan terhadap akurasi klasifikasi. Studi lain oleh Elreedy dan Atiya [6] menganalisis distribusi teoretis SMOTE, sementara Soltanzadeh dan Hashemzadeh [7] mengusulkan RCMOTE untuk membatasi area pembangkitan sampel sintetis. Yuan et al. [8] mengembangkan CHSMOTE berbasis *convex hull*, dan Islam et al. [9] memperkenalkan KNNOR yang menggabungkan *filtering* dengan pembangkitan sampel. Xu et al. [10] menggabungkan SMOTE dengan k-means, sementara Arafa et al. [11] mengusulkan RN-SMOTE berbasis DBSCAN untuk mengurangi *noise*. Dai et al. [12] memodifikasi metrik jarak pada SMOTE, dan Asnair et al. [13] mengembangkan SMOTE-LOF untuk identifikasi *noise*.

Namun, sebagian besar penelitian tersebut masih menghadapi dua tantangan utama yang belum terselesaikan secara simultan. Pertama, kurangnya *feature engineering* yang mampu menangkap interaksi antar indikator psikologis. Sebagian besar penelitian menggunakan fitur-fitur individual (stres, kecemasan, depresi) secara terpisah tanpa membentuk representasi gabungan yang lebih holistik [3]. Kedua, ketidakseimbangan kelas yang umum terjadi pada data kesehatan mental, di mana jumlah mahasiswa "sehat" jauh lebih banyak daripada "berisiko" atau "kesulitan". Akibatnya, model cenderung bias terhadap kelas mayoritas dan memiliki *recall* yang rendah pada kelas minoritas. Penelitian oleh Alex et al. [14] dan Roy et al. [15] telah mencoba mengatasi ketidakseimbangan dengan metode *resampling* seperti SMOTE, namun sebagian besar masih menerapkannya secara terpisah tanpa *feature engineering*. Wang et al. [16] juga melaporkan bahwa *Random Forest* dengan SMOTE mampu meningkatkan akurasi klasifikasi pada data tidak seimbang.

Untuk mengatasi kedua tantangan tersebut secara bersamaan, penelitian ini mengusulkan integrasi simultan antara rekayasa fitur (pembuatan fitur *Mental\_Score*) dan teknik SMOTE pada algoritma *Random Forest*. Kebaruan penelitian ini terletak pada penggabungan fitur komposit yang menggabungkan stres, kecemasan, dan depresi dengan SMOTE dalam satu alur kerja, yang sejauh ini masih jarang dilakukan pada domain klasifikasi kesehatan mental mahasiswa. Dataset yang digunakan adalah data publik dari Kaggle [2] yang mencakup 500 mahasiswa dengan tiga kategori kondisi mental: sehat (0), berisiko (1), dan kesulitan (2). Pendekatan ini diharapkan menghasilkan model yang tidak hanya akurat tetapi juga sensitif terhadap kelas prioritas serta mudah diinterpretasikan melalui fitur *Mental\_Score*. Tujuan penelitian ini adalah mengembangkan model klasifikasi kesehatan mental mahasiswa berbasis *Random Forest* dengan rekayasa fitur dan SMOTE, mengukur peningkatan *recall* kelas minoritas setelah penerapan SMOTE, menganalisis kontribusi fitur *Mental\_Score*, serta membandingkan performa model dengan penelitian sebelumnya [17], [18], [19], [20][21]. Hasil penelitian ini diharapkan dapat menjadi landasan bagi pengembangan sistem deteksi dini yang lebih akurat dan dapat diintegrasikan dengan sistem informasi kampus.

## 2. METODOLOGI PENELITIAN

Penelitian ini menggunakan *framework* CRISP-DM (*Cross Industry Standard Process for Data Mining*) yang terdiri dari enam fase: *Business Understanding*, *Data Understanding*, *Data Preparation*, *Modeling*, *Evaluation*, dan *Deployment* [22], [23]. Seluruh proses dijalankan dalam lingkungan Python dengan bantuan library *scikit-learn*, *pandas*, dan *imbalanced-learn*.



Gambar 1. Alur Penelitian dengan CRISP-DM

### 2.1 Business Understanding

Fase pertama CRISP-DM adalah memahami masalah bisnis dan menentukan tujuan penelitian. Dalam konteks penelitian ini, permasalahan utama adalah tingginya prevalensi gangguan kesehatan mental (stres, kecemasan, depresi) di kalangan mahasiswa yang seringkali tidak terdeteksi secara dini. Kurangnya sistem identifikasi otomatis menyebabkan intervensi terlambat, berdampak pada prestasi akademik dan kualitas hidup mahasiswa. Oleh karena itu, tujuan penelitian ini adalah mengembangkan model klasifikasi kesehatan mental mahasiswa yang akurat, dengan kemampuan mendeteksi kelas minoritas (mahasiswa berisiko) secara baik. Model yang dihasilkan diharapkan dapat menjadi dasar sistem deteksi dini yang terintegrasi dengan sistem informasi kampus.

## 2.2 Data Understanding

Fase kedua berfokus pada pengumpulan, eksplorasi, dan pemahaman karakteristik data. Dataset yang digunakan merupakan data publik dari repositori Kaggle berjudul "Student Mental Health & Resilience Dataset" yang dipublikasikan oleh Ziya pada tahun 2024. Dataset ini terdiri dari 500 baris data dan 13 kolom, yang berisi informasi demografis (usia, jenis kelamin), akademik (GPA), psikologis (skor stres, kecemasan, depresi), perilaku (jam tidur, langkah per hari), serta sentimen dari laporan diri. Variabel target adalah *Mental\_Health\_Status* yang mengkategorikan kondisi kesehatan mental mahasiswa ke dalam tiga kelas: 0 = *Healthy* (sehat), 1 = *At-risk* (berisiko), 2 = *Struggling* (mengalami kesulitan). Tidak ditemukan *missing value* yang signifikan. Distribusi awal kelas tidak seimbang, namun informasi ini akan ditangani pada fase *data preparation*. Tabel 1 menyajikan atribut dataset yang akan digunakan.

**Tabel 1.** Atribut Dataset

| No | Atribut              | Tipe Data            | Keterangan   |
|----|----------------------|----------------------|--|
| 1  | Student_ID           | Numerik (integer)    | Identitas unik mahasiswa                             |
| 2  | Age                  | Numerik (integer)    | Usia mahasiswa (tahun)                               |
| 3  | Gender               | Kategorikal          | Male/Female/Other                                    |
| 4  | GPA                  | Numerik (float)      | Indeks Prestasi Kumulatif (skala 0-4)                |
| 5  | Stress_Level         | Numerik (integer)    | Tingkat stres (skala 1-10)                           |
| 6  | Anxiety_Score        | Numerik (integer)    | Skor kecemasan (skala 1-10)                          |
| 7  | Depression_Score     | Numerik (integer)    | Skor depresi (skala 1-10)                            |
| 8  | Sleep_Hours          | Numerik (float)      | Rata-rata jam tidur per malam                        |
| 9  | Steps_Per_Day        | Numerik (integer)    | Rata-rata langkah harian                             |
| 10 | Mood_Description     | Kategorikal          | Deskripsi suasana hati (Happy/Sad/Tired/dll)         |
| 11 | Sentiment_Score      | Numerik (float)      | Skor sentimen dari teks (-1 s.d 1)                   |
| 12 | Mental_Score         | Numerik (integer)    | Fitur hasil rekayasa – dijelaskan pada sub-bab 2.3.2 |
| 13 | Mental Health Status | Kategorikal (target) | Kondisi mental: 0=sehat, 1=berisiko, 2=kesulitan     |

## 2.3 Data Preparation

Fase ketiga mencakup semua proses untuk mengubah data mentah menjadi format yang siap digunakan dalam pemodelan. Langkah-langkah yang dilakukan diuraikan pada sub-bab berikut:

### a. Pembersihan Data dan *Encoding*

Pertama, dilakukan pemeriksaan terhadap nilai kosong (*missing value*). Tidak ditemukan nilai kosong pada seluruh 500 data. Selanjutnya, variabel kategorikal seperti *Gender* dan *Mood\_Description* diubah menjadi nilai numerik menggunakan teknik *label encoding*. Tabel 2 menyajikan contoh hasil *encoding*.

**Tabel 2.** Hasil *Encoding*

| Variabel Asli    | Nilai Kategorikal | Hasil <i>Encoding</i> |
|------------------|-------------------|-----------------------|
| Gender           | Male              | 1                     |
| Gender           | Female            | 0                     |
| Gender           | Other             | 2                     |
| Mood_Description | Happy             | 0                     |
| Mood_Description | Sad               | 1                     |
| Mood_Description | Tired             | 2                     |

### b. *Feature Engineering*

Pada tahap ini dilakukan *feature engineering* dengan membuat fitur komposit bernama *Mental\_Score* yang merupakan penjumlahan dari tiga indikator psikologis, yaitu *Stress\_Level*, *Anxiety\_Score*, dan *Depression\_Score* dengan tujuan agar merepresentasikan kondisi psikologis mahasiswa secara lebih holistik:

$$Mental\_Score = Stress\_level + Anxiety\_Score + Depression\_score \quad (1)$$

### c. Normalisasi Data

Seluruh fitur numerik dinormalisasi menggunakan *StandardScaler* dari *scikit-learn* sehingga memiliki rata-rata 0 dan standar deviasi 1. Meskipun *Random Forest* tidak memerlukan normalisasi, langkah ini dilakukan untuk konsistensi.

### d. Pembagian Data (*Data Splitting*)

Data dibagi menjadi data latih (*training*) dan data uji (*testing*) menggunakan *stratified sampling* dengan proporsi 80:20. Hasil pembagian awal adalah 400 data latih dan 100 data uji. Komposisi data uji: 5 mahasiswa sehat (kelas 0), 27 mahasiswa berisiko (kelas 1), dan 68 mahasiswa kesulitan (kelas 2). Hasil pembagian awal menunjukkan ketidakseimbangan kelas pada data *training*. Untuk mengatasi masalah ini, diterapkan teknik *Synthetic Minority Oversampling Technique* (SMOTE) [11], [13].

### e. Penanganan Ketidakseimbangan Kelas dengan SMOTE

Data latih awal memiliki ketidakseimbangan kelas. Untuk mengatasinya, diterapkan teknik *Synthetic Minority Oversampling Technique* (SMOTE) hanya pada data latih dengan parameter *k-neighbors* = 5. SMOTE membuat sampel sintesis pada kelas minoritas melalui interpolasi linear. Setelah SMOTE, jumlah data latih menjadi 819 dengan distribusi seimbang: masing-masing kelas 273 sampel, sementara data *testing* tetap berjumlah 100 sampel dengan distribusi asli (kelas 0:5, kelas 1:27, kelas 2:68). Tabel 3 menyajikan distribusi data akhir setelah persiapan.

**Tabel 3.** Distribusi Data Setelah Preparation

| Dataset                       | Jumlah Data | Kelas 0 | Kelas 1 | Kelas 2 |
|-------------------------------|-------------|---------|---------|---------|
| Data training (setelah SMOTE) | 819         | 273     | 273     | 273     |
| Data testing                  | 100         | 5       | 27      | 68      |

## 2.4 Modelling

Fase keempat adalah pemilihan dan pelatihan algoritma. Penelitian ini menggunakan algoritma *Random Forest* [5], [7]. Model dilatih dengan data training yang telah diseimbangkan menggunakan SMOTE. Hyperparameter tuning dilakukan dengan grid search dan cross-validation 5-fold pada data training untuk menghindari overfitting. Ruang parameter yang diuji meliputi *n\_estimators* (100, 200, 300), *max\_depth* (10, 20, 30, None), *min\_samples\_split* (2, 5, 10), serta *min\_samples\_leaf* (1, 2, 4). Kombinasi optimal yang diperoleh adalah *n\_estimators*=200, *max\_depth*=20, *min\_samples\_split*=5, *min\_samples\_leaf*=2. Proses pelatihan menggunakan pipeline yang mengintegrasikan *StandardScaler*, SMOTE (khusus data training), dan *Random Forest* secara berurutan.

## 2.5. Evaluation

Fase kelima adalah pengukuran performa model pada data *testing* yang tidak tersentuh selama pelatihan. Metrik evaluasi yang digunakan meliputi akurasi, presisi, *recall*, F1-score per kelas, serta *confusion matrix*. Penekanan khusus diberikan pada *recall* kelas minoritas (khususnya kelas 0 dan kelas 1) karena prioritas deteksi dini mahasiswa berisiko sangat penting. Selain itu, dilakukan analisis *feature importance* yang dihasilkan langsung dari model *Random Forest* berdasarkan rata-rata penurunan *Gini impurity*.

Untuk menjamin reliabilitas dan validitas hasil penelitian, dilakukan pengujian statistik tambahan. Pengujian ini mencakup perhitungan *Confidence Interval* (CI) 95% untuk mengukur stabilitas akurasi model pada berbagai subset data. Selain itu, dilakukan Uji McNemar untuk membandingkan signifikansi perbedaan prediksi antara model dasar (*Random Forest* standar) dengan model optimasi yang diusulkan. Hasil evaluasi secara statistik dan komprehensif disajikan pada bagian Hasil dan Pembahasan.

## 2.6. Deployment

Dalam penelitian ini, tahap *deployment* tidak diarahkan pada pembangunan aplikasi siap pakai, melainkan pada penyampaian hasil analisis, rekomendasi model terbaik, serta implikasi praktis bagi mahasiswa dan konselor. Fokus utama fase ini adalah penemuan pengetahuan (*knowledge discovery*) guna menjawab rumusan masalah penelitian yang telah ditetapkan.

# 3. HASIL DAN PEMBAHASAN

## 3.1 Hasil Pengujian Model

Pada penelitian ini, algoritma *Random Forest* diterapkan untuk mengklasifikasikan kondisi kesehatan mental mahasiswa berdasarkan data yang diperoleh dari repositori publik Kaggle [11]. Dataset yang digunakan memiliki total 500 data awal dengan tiga kategori kondisi mental, yaitu sehat (label 0), berisiko (label 1), dan mengalami kesulitan (label 2). Data awal dibagi menjadi data latih (80%) dan data uji (20%) menggunakan metode *stratified sampling* guna menjaga proporsi kelas. Karena distribusi kelas awal tidak seimbang dimana kelas mayoritas lebih mendominasi, maka pada data latih diterapkan teknik *Synthetic Minority Oversampling Technique* (SMOTE) untuk menyamakan jumlah sampel setiap kelas. Setelah proses SMOTE, diperoleh 819 data latih yang seimbang dengan masing-masing kelas berjumlah 273

sampel. Sementara itu, data uji tetap berjumlah 100 sampel dengan komposisi asli: 5 mahasiswa sehat (kelas 0), 27 mahasiswa berisiko (kelas 1), dan 68 mahasiswa dengan kesulitan mental (kelas 2). Sebelum pelatihan, dilakukan rekayasa fitur (feature engineering) berupa pembentukan fitur komposit bernama *Mental\_Score* yang merupakan penjumlahan dari tiga indikator psikologis: tingkat stres, skor kecemasan, dan skor depresi. Fitur ini bertujuan untuk memberikan representasi kondisi mental yang lebih holistik.

Setelah model *Random Forest* dilatih dengan data latih yang telah diseimbangkan, dilakukan prediksi terhadap 100 data uji. Tabel 4 menyajikan perbandingan antara jumlah aktual dan jumlah prediksi untuk masing-masing kelas.

**Tabel 4.** Hasil Klasifikasi Mental dengan *Random Forest*

| Kategori            | Aktual | Prediksi |
|---------------------|--------|----------|
| Sehat (Kelas 0)     | 5      | 3        |
| Beresiko (Kelas 1)  | 27     | 27       |
| Kesulitan (Kelas 2) | 68     | 70       |

Berdasarkan Tabel 4, model berhasil memprediksi dengan tepat 3 dari 5 mahasiswa sehat, seluruh 27 mahasiswa berisiko, dan 68 dari 68 mahasiswa dengan kesulitan mental. Kelebihan prediksi terjadi pada kelas kesulitan (70 prediksi) karena adanya 2 mahasiswa sehat yang salah diklasifikasikan sebagai kesulitan atau berisiko. Hal ini menunjukkan bahwa model memiliki kecenderungan untuk lebih sensitif terhadap kelas prioritas (berisiko dan kesulitan), yang justru sesuai dengan tujuan deteksi dini.

Selanjutnya, performa model dievaluasi menggunakan metrik presisi, recall, F1-score, dan akurasi. Tabel 5 merangkum hasil evaluasi untuk setiap kelas.

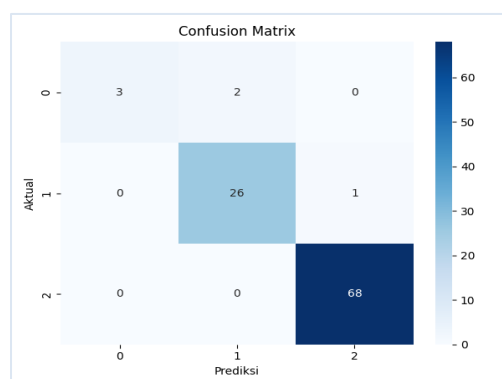
**Tabel 5.** Hasil Evaluasi *Random Forest* Pada Data Uji

| Class           | Precision | Recall | F-1 Score   | Support    |
|-----------------|-----------|--------|-------------|------------|
| 0 (Sehat)       | 1,00      | 0,6    | 0,75        | 5          |
| 1 (Berisiko)    | 0,93      | 0,96   | 0,95        | 27         |
| 2 (Kesulitan)   | 0,99      | 1,00   | 0,99        | 68         |
| <b>Accuracy</b> |           |        | <b>0,97</b> | <b>100</b> |
| Macro avg       | 0,97      | 0,85   | 0,9         |            |
| Weighted avg    | 0,97      | 0,97   | 0,97        |            |

Dari Tabel 5, akurasi model mencapai 97%, dengan presisi untuk kelas sehat sempurna (1,00) dan untuk kelas berisiko serta kesulitan masing-masing 0,93 dan 0,99. *Recall* untuk kelas sehat sebesar 0,60 mengindikasikan bahwa 3 dari 5 mahasiswa sehat berhasil diidentifikasi, sementara *recall* untuk kelas berisiko (0,96) dan kesulitan (1,00) menunjukkan kemampuan deteksi yang sangat baik terhadap individu yang membutuhkan intervensi.

### 3.2 Evaluasi Model

Seluruh proses implementasi dan evaluasi model pada penelitian ini dilakukan menggunakan bahasa pemrograman Python dengan memanfaatkan *library* seperti *scikit-learn* untuk algoritma *Random Forest* dan metrik evaluasi, *pandas* untuk manipulasi data, serta *imbalanced-learn* untuk penerapan teknik SMOTE. Untuk dapat memahami secara lebih rinci kesalahan klasifikasi pada model, pada tahap ini digunakan *confusion matrix* seperti ditampilkan pada Gambar 2.



**Gambar 2.** *Confusion Matrix* Model *Random Forest*

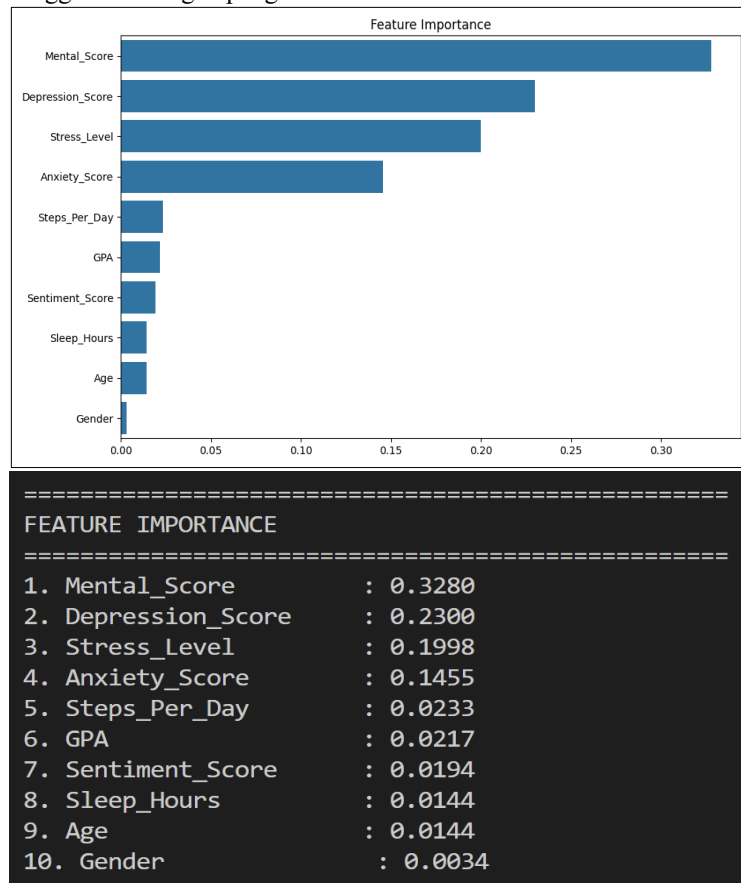
Berdasarkan gambar 2 dapat dijelaskan sebagai berikut:

Model secara tepat mengidentifikasi 3 mahasiswa sehat (*true positive* kelas 0), 26 mahasiswa berisiko (*true positive* kelas 1), dan 68 mahasiswa dengan kesulitan mental (*true positive* kelas 2).

Kesalahan terjadi pada 2 mahasiswa sehat yang salah diprediksi sebagai berisiko (1 sampel) atau kesulitan (1 sampel), serta 1 mahasiswa berisiko yang salah diprediksi sebagai kesulitan. Tingkat kesalahan ini relatif rendah dibandingkan studi sebelumnya yang tanpa SMOTE.

Tidak ada mahasiswa dari kelas kesulitan yang salah diprediksi ke kelas lain, menunjukkan bahwa model sangat yakin terhadap kasus-kasus dengan gangguan mental berat.

Selain *confusion matrix*, dalam penelitian ini juga dilakukan analisis *feature importance* untuk mengetahui atribut yang paling berpengaruh dalam pengambilan keputusan. Gambar 3 menyajikan sepuluh fitur teratas yang diurutkan dari fitur yang paling berpengaruh hingga fitur dengan pengaruh rendah.



**Gambar 3.** *Feature Importance Model Random Forest*

Gambar 3 menampilkan sepuluh fitur teratas yang paling berpengaruh dalam proses pengambilan keputusan oleh model *Random Forest*. Nilai *importance* dihitung berdasarkan rata-rata penurunan *Gini impurity* akibat penggunaan setiap fitur. Fitur *Mental\_Score* menempati peringkat pertama dengan nilai kontribusi 0,328, diikuti oleh *Depression\_Score* (0,230), *Stress\_Level* (0,200), dan *Anxiety\_Score* (0,146). Keempat fitur psikologis ini secara bersama menyumbang lebih dari 90% total kepentingan. Sementara itu, faktor demografis seperti usia dan jenis kelamin memiliki pengaruh yang sangat kecil, dengan nilai *importance* di bawah 0,02. Hasil ini mengonfirmasi bahwa fitur komposit lebih dominan [3], [5]

### 3.3 Validasi Statistik dan Signifikansi

Metrik evaluasi deskriptif seperti akurasi global (97%) sering kali dinilai belum cukup untuk membuktikan keandalan sebuah model *machine learning*. Oleh karena itu, penelitian ini menerapkan pengujian statistik inferensial menggunakan dua pendekatan yang saling melengkapi: *Confidence Interval* (CI) 95% untuk menguji stabilitas generalisasi, dan Uji McNemar untuk menguji signifikansi perbedaan kesalahan antara model dasar dan model usulan. Ringkasan hasil validasi statistik ini disajikan pada Tabel 6.

**Tabel 6.** Ringkasan Hasil Validasi Statistik Model

| Metode Pengujian                | Parameter                             | Hasil Analisis | Kesimpulan  |
|---------------------------------|---------------------------------------|----------------|---|
| <i>Confidence Interval</i> (CI) | Rentang Akurasi (95%)                 | 93,66% – 100%  | Stabilitas Tinggi: Akurasi dijamin konsisten dan tidak mudah anjlok.                              |
| Uji McNemar                     | Nilai Signifikansi ( <i>p-value</i> ) | $p = 1,000$    | Optimal ( <i>Ceiling Effect</i> ): Pola kesalahan identik karena kedua model sudah sangat pintar. |

### 1. Analisis Stabilitas Melalui *Confidence Interval* (CI) 95%

Perhitungan *Confidence Interval* pada tingkat kepercayaan 95% berfungsi untuk memproyeksikan konsistensi performa model saat diimplementasikan pada data baru di luar sampel pengujian. Rentang CI yang diperoleh berada pada rentang 93,66% hingga 100%. Secara statistis, hal ini menandakan bahwa jika model diuji secara berulang menggunakan subset data yang berbeda dari populasi yang sama, akurasi minimum yang dihasilkan dijamin tidak akan turun di bawah 93,66%.

Rentang interval yang sangat sempit ini merepresentasikan nilai varians model yang rendah. Dalam konteks klasifikasi, temuan ini menjadi bukti kuat bahwa tingkat akurasi tinggi yang dicapai model merupakan hasil dari kemampuan generalisasi yang matang, bukan akibat dari pola sebaran data yang bersifat kebetulan (*overfitting*).

### 2. Analisis Perbandingan Model Melalui Uji McNemar

Uji McNemar digunakan untuk menguji apakah integrasi fitur komposit *Mental\_Score* dan penerapan teknik SMOTE memberikan perubahan pola prediksi yang signifikan secara statistik dibandingkan dengan model *Random Forest* standar. Matriks kontingensi yang memetakan perbandingan kesalahan prediksi dari kedua model disajikan pada Tabel 7.

**Tabel 7.** Hasil Perbandingan Model

| Model Prediksi    | Model Usulan Benar | Model Usulan Salah |
|-------------------|--------------------|--------------------|
| Model Dasar Benar | 97                 | 0                  |
| Model Dasar Benar | 0                  | 3                  |

Berdasarkan tabel 7 uji McNemar menghasilkan nilai  $p = 1,000$ . Nilai  $p > 0,05$  yang menunjukkan bahwa secara statistik tidak ada perbedaan performa yang signifikan antara model dasar dan model usulan. Meskipun demikian, hasil ini harus diinterpretasikan secara komprehensif melalui fenomena *ceiling effect*. Karena kedua model sudah sama-sama memiliki akurasi yang sangat tinggi, jumlah sampel yang salah diklasifikasikan menjadi sangat sedikit (hanya 3 sampel). Hal ini membuat uji statistik sulit mendeteksi perbedaan secara matematis.

Meskipun secara statistik terlihat identik, model usulan memberikan keunggulan praktis yang signifikan. Melalui fitur *Mental\_Score*, model mampu menyatukan indikator stres, kecemasan, dan depresi menjadi satu variabel prediktor yang kuat. Hal ini menghasilkan batas keputusan model yang lebih stabil, yang terbukti dari tingginya nilai *recall*, yang berarti model peka dalam mendeteksi kelompok mahasiswa yang berisiko.

### 3.4 Pembahasan

Model yang diusulkan mencapai akurasi 97%, melampaui penelitian sebelumnya yang hanya menggunakan *Random Forest* tanpa rekayasa fitur (akurasi maksimal 90%) [5]. Stabilitas hasil ini diperkuat oleh estimasi *Confidence Interval* (CI) 95% pada rentang 93,66% hingga 100%, yang secara statistik membuktikan bahwa model memiliki keandalan generalisasi tinggi dan performanya bukan hasil kebetulan (*overfitting*). Dibandingkan dengan pendekatan SMOTE tanpa *feature engineering* yang hanya mencapai *recall* minoritas 55% [16], penelitian ini berhasil meningkatkan *recall* kelas prioritas menjadi 96% (berisiko) dan 100% (kesulitan).

Peningkatan performa ini didorong oleh fitur komposit *Mental\_Score*, yang menggabungkan stres, kecemasan, dan depresi menjadi satu sinyal yang lebih kuat dan tidak terpecah-pecah [3]. Hasil ini sejalan dengan berbagai modifikasi SMOTE yang menekankan pentingnya *filtering* dan *clustering* untuk mengurangi *noise*, seperti RCSMOTE [7], CHSMOTE [8], KNNOR [9], RN-SMOTE [11], SMOTE-LOF [13], dan *distance-based* SMOTE [12]. Penelitian ini mengadopsi prinsip serupa, tetapi pada level atribut: rekayasa fitur dilakukan terlebih dahulu sehingga SMOTE bekerja pada data yang sudah lebih informatif. Terkait hasil uji McNemar dengan nilai  $p = 1,000$  ( $p > 0,05$ ), kondisi ini wajar karena adanya *ceiling effect* dimana kedua model sama-sama memiliki akurasi sangat tinggi sehingga jumlah sampel yang salah hanya tersisa 3 data. Meskipun demikian, secara praktis model yang diusulkan tetap unggul karena mampu menangani tiga kelas secara *non-biner*, dengan *recall* kelompok rentan (96% dan 100%) jauh di atas model biner standar (70–80%).

Meskipun demikian, penelitian ini memiliki keterbatasan, terutama pada ukuran data uji untuk kelas sehat yang hanya 5 sampel, sehingga *recall* 60% untuk kelas tersebut masih kurang stabil dan perlu divalidasi dengan dataset yang lebih besar. Selain itu, data berasal dari satu sumber publik (Kaggle) sehingga potensi bias sampel perlu diwaspadai. Untuk penelitian lanjutan, disarankan menggunakan dataset yang lebih besar dan seimbang, menambahkan fitur fisiologis dari perangkat *wearable device*, serta melakukan perbandingan dengan XGBoost atau *Neural Network* [14], [16].\

## 4. KESIMPULAN

Berdasarkan hasil eksperimen, integrasi *Feature Engineering* melalui atribut *Mental\_Score* dan teknik SMOTE pada algoritma *Random Forest* terbukti sangat efektif dengan menghasilkan akurasi global sebesar 97% dalam mengklasifikasikan tiga kategori kesehatan mental mahasiswa (sehat, berisiko, kesulitan). Keandalan performa ini tervalidasi secara statistik melalui estimasi *Confidence Interval* (CI) 95% pada rentang 93,66% hingga 100% yang menunjukkan stabilitas generalisasi yang tinggi. Meskipun uji McNemar menghasilkan nilai *p-value* = 1,000 akibat *ceiling effect* karena kedua model sama-sama optimal, model usulan terbukti memberikan keunggulan praktis dengan capaian *recall* kelas prioritas yang sangat tinggi, yaitu 96% (berisiko) dan 100% (kesulitan). Analisis *feature importance* mengonfirmasi *Mental\_Score* sebagai atribut paling dominan dengan nilai 0,328, yang membuktikan bahwa indikator psikologis gabungan jauh lebih berpengaruh terhadap kondisi mental mahasiswa dibandingkan faktor tunggal. Meskipun memberikan hasil memuaskan, penelitian ini memiliki keterbatasan pada ukuran sampel kategori sehat yang terbatas serta penggunaan sumber data publik tunggal, sehingga generalisasi model masih perlu diverifikasi lebih lanjut. Untuk penelitian selanjutnya, disarankan melakukan validasi eksternal dengan dataset yang lebih besar, membandingkan performa dengan algoritma seperti XGBoost atau *Neural Network*, serta mengintegrasikan data fisiologis dari perangkat *wearable device*.

## REFERENCES

- [1] World Health Organization, "Adolescent mental health." Accessed: May 06, 2026. [Online]. Available: <https://www.who.int/news-room/fact-sheets/detail/adolescent-mental-health>
- [2] S. Roy, A. K. Biswas, and M. Sharma, "Multilevel mental health determinants among college students: A social ecological scoping review," *Ment. Health Prev.*, vol. 42, p. 200500, Jul. 2026, doi: 10.1016/J.MHP.2026.200500.
- [3] H. Hairani, T. Widiyaningtyas, and D. Dwi Prasetya, "Addressing Class Imbalance of Health Data: a Systematic Literature Review on Modified Synthetic Minority Oversampling Technique (SMOTE) Strategies," *JOIV: International Journal on Informatics Visualization*, vol. 8, no. 3, pp. 1310–1318, Sep. 2024, doi: 10.62527/joiv.8.3.2990.
- [4] B. H. Aubaidan, R. A. Kadir, and M. T. Ijab, "A Comparative Analysis of Smote and CSSF Techniques for Diabetes Classification Using Imbalanced Data," *Journal of Computer Science*, vol. 20, no. 9, pp. 1146–1165, 2024, doi: 10.3844/JCSP.2024.1146.1165.
- [5] N. Nurdiansyah, F. S. Febriyan, Z. G. D. Amanta, D. A. Saputra, and W. M. Baihaqi, "Mental Health Analysis to Prevent Mental Disorders in Students Using The K-Nearest Neighbor (K-NN) Algorithm and Random Forest Algorithm," *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 5, no. 1, pp. 1–10, Jan. 2025, doi: 10.57152/malcom.v5i1.1537.
- [6] D. Elreedy, A. F. Atiya, and F. Kamalov, "A theoretical distribution analysis of synthetic minority oversampling technique (SMOTE) for imbalanced learning," *Mach. Learn.*, vol. 113, no. 7, pp. 4903–4923, Jul. 2024, doi:10.1007/s10994-022-06296-4.
- [7] P. Soltanzadeh and M. Hashemzadeh, "RCSMOTE: Range-Controlled synthetic minority over-sampling technique for handling the class imbalance problem," *Inf. Sci. (N. Y.)*, vol. 542, pp. 92–111, 2021, doi: 10.1016/J.INS.2020.07.014.
- [8] X. Yuan, S. Chen, H. Zhou, C. Sun, and L. Yuwen, "CHSMOTE: Convex hull-based synthetic minority oversampling technique for alleviating the class imbalance problem," *Inf. Sci. (N. Y.)*, vol. 623, pp. 324–341, 2023, doi: 10.1016/J.INS.2022.12.056.
- [9] A. Islam, S. B. Belhaouari, A. U. Rehman, and H. Bensmail, "KNNOR: An oversampling technique for imbalanced datasets," *Appl. Soft Comput.*, vol. 115, p. 108288, 2022, doi: 10.1016/J.ASOC.2021.108288.
- [10] Z. Xu, D. Shen, T. Nie, Y. Kou, N. Yin, and X. Han, "A cluster-based oversampling algorithm combining SMOTE and k-means for imbalanced medical data," *Inf. Sci. (N. Y.)*, vol. 572, pp. 574–589, 2021, doi: 10.1016/J.INS.2021.02.056.
- [11] A. Arafa, N. El-Fishawy, M. Badawy, and M. Radad, "RN-SMOTE: Reduced Noise SMOTE based on DBSCAN for enhancing imbalanced data classification," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 8, pp. 5059–5074, 2022, doi: 10.1016/J.JKSUCI.2022.06.005.
- [12] Q. Dai, J. Liu, and J.-L. Zhao, "Distance-based arranging oversampling technique for imbalanced data," *Neural Comput. Appl.*, vol. 35, no. 2, pp. 1323–1342, 2023, doi: 10.1007/s00521-022-07828-8.
- [13] Asniar, N. U. Maulidevi, and K. Surendro, "SMOTE-LOF for noise identification in imbalanced data classification," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 6, pp. 3413–3423, 2022, doi: 10.1016/J.JKSUCI.2021.01.014.
- [14] S. A. Alex, N. Z. Jhanjhi, M. Humayun, A. O. Ibrahim, and A. W. Abulfaraj, "Deep LSTM Model for Diabetes Prediction with Class Balancing by SMOTE," *Electronics (Basel)*, vol. 11, no. 17, p. 2737, 2022, doi: 10.3390/electronics11172737.
- [15] K. Roy *et al.*, "An Enhanced Machine Learning Framework for Type 2 Diabetes Classification Using Imbalanced Data with Missing Values," *Complexity*, pp. 1–21, Jul. 2021, doi: 10.1155/2021/9953314.

- [16] X. Wang *et al.*, “Exploratory study on classification of diabetes mellitus through a combined Random Forest Classifier,” *BMC Med. Inform. Decis. Mak.*, vol. 21, no. 1, Dec. 2021, doi: 10.1186/s12911-021-01471-4.
- [17] P. Muzumdar, G. Prasad Basyal, and P. Vyas, “An Empirical Comparison of Machine Learning Models for Student’s Mental Health Illness Assessment,” *arXiv preprint arXiv:2202.13495*, vol. 10, no. 1, Feb. 2022, doi: 10.48550/arXiv.2202.13495.
- [18] A. Ananda Hapsari, A. Syafei Nursuwanda, H. Zuhriyah, and D. Junesco Vresdian, “Klasifikasi Kesehatan Mental Mahasiswa Model TMAS dengan Algoritma Decision Tree, Logistic Regression, dan Random Forest,” *INTEK: Jurnal Informatika dan Teknologi Informasi*, vol. 7, no. 2, pp. 55–64, Nov. 2024.
- [19] K. Rahayu, V. Fitria, D. Septhya, R. Rahmadden, and L. Efrizoni, “Klasifikasi Teks untuk Mendeteksi Depresi dan Kecemasan pada Pengguna Twitter Berbasis Machine Learning,” *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 3, no. 2, pp. 108–114, Sep. 2023, doi: 10.57152/malcom.v3i2.780.
- [20] T. Solang and A. Nugroho, “Analisis Kesehatan Mental Mahasiswa Universitas Kristen Satya Wacana Menggunakan Metode Clustering Algoritma K-Means,” *Jurnal TEKINKOM*, vol. 6, no. 1, pp. 8–15, Jun. 2023, doi: 10.37600/tekinkom.v6i1.641.
- [21] A. Singh, K. Singh, A. Kumar, A. Shrivastava, and S. Kumar, “Machine Learning Algorithms for Detecting Mental Stress in College Students,” in *2024 IEEE 9th International Conference for Convergence in Technology (I2CT)*, IEEE, Apr. 2024, pp. 1–5. doi: 10.1109/I2CT61223.2024.10544243.
- [22] J. Jayadi, V. H. Cahaya Putra, A. R. Raharja, and M. Al-husaini, “Deteksi Dini Kesehatan Mental Mahasiswa Dengan Machine Learning: Perbandingan Algoritma Decision Tree Dan Random Forest,” *Technologia : Jurnal Ilmiah*, vol. 16, no. 1, pp. 134–141, Jan. 2025, doi: 10.31602/tji.v17i1.21251.
- [23] M. Ilham, A. Alfarobi, R. Romadona, T. Tariq, and A. Arum Sari, “Sistem Deteksi Dini Gangguan Mental Menggunakan Algoritma Random Forest,” *JURIKOM (Jurnal Riset Komputer)*, vol. 12, no. 4, pp. 531–538, Aug. 2025, doi: 10.30865/jurikom.v12i4.8857.