Jurnal Smart Teknologi
Vol. 7, No. 1, Nopember 2025, Halaman 23 – 32
ISSN: 2774-1702, http://jurnal.unmuhjember.ac.id/index.php/JST

Implementasi Algoritma *Random Forest* Berbasis *Oversampling* SMOTE Untuk Klasifikasi Penyakit Alzheimer

Implementation of the Random Forest Algorithm with SMOTE-Based Oversampling for Alzheimer's Disease Classification

Galung Reinan[1)], Agung Nilogiri[2)], Wiwik Suharso[3)]
[1]Teknik Informatika, Universitas Muhammadiyah Jember
email: galung.reiarta23@gmail.com
[2]Teknik Informatika, Universitas Muhammadiyah Jember
email: agungnilogiri@unmuhjember.ac.id
[3]Teknik Informatika, Universitas Muhammadiyah Jember
email: wiwiksuharso@unmuhjember.ac.id

Abstrak

Penyakit Alzheimer merupakan bentuk paling umum dari demensia yang menjadi masalah kesehatan global, terutama pada lanjut usia. Deteksi dini sangat penting untuk memperlambat perkembangan penyakit dan meningkatkan kualitas hidup pasien melalui intervensi medis yang tepat waktu. Namun, diagnosis konvensional sering kali memerlukan waktu lama, biaya tinggi, serta bergantung pada tenaga ahli. Penelitian ini bertujuan mengimplementasikan algoritma *Random Forest* untuk klasifikasi penyakit Alzheimer secara otomatis, efisien, dan akurat. *Dataset* yang digunakan berasal dari Kaggle, terdiri atas 2.149 data pasien dengan 35 atribut terkait. Proses pengolahan data mengikuti kerangka kerja CRISP-DM yang mencakup enam tahap: *Business Understanding*, *Data Understanding*, *Data Preparation*, *Modeling*, *Evaluation*, dan *Deployment*. Untuk mengatasi ketidakseimbangan kelas, digunakan teknik *oversampling* SMOTE. Evaluasi model dilakukan menggunakan metrik akurasi, presisi, *recall*, dan *F1-score*. Hasil yang diperoleh menunjukkan bahwa algoritma *Random Forest* dapat digunakan sebagai sistem pendukung keputusan dalam deteksi dini Alzheimer secara praktis dan berbasis data.

Kata Kunci: Alzheimer, Klasifikasi, *Machine Learning, Random Forest.*

Abstract

Alzheimer's disease is the most common form of dementia and has become a global health concern, particularly among the elderly. Early detection is crucial to slowing disease progression and improving patients' quality of life through timely medical intervention. However, conventional diagnostic procedures often require significant time, high costs, and reliance on medical experts. This study aims to implement the Random Forest algorithm for automated, efficient, and accurate classification of Alzheimer's disease. The dataset used was obtained from Kaggle, consisting of 2,149 patient records with 35 related attributes. Data processing follows the CRISP-DM framework, which includes six stages: Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, and Deployment. To address class imbalance, the SMOTE (Synthetic Minority Oversampling Technique) method was applied. Model performance was evaluated using accuracy, precision, recall, and F1-score metrics. The results indicate that the Random Forest algorithm can serve as a data-driven decision support system for the early detection of Alzheimer's disease in a more practical and accessible manner.

Keywords*: Alzheimer, Classification, Machine Learning, Random Forest.*

Jurnal Smart Teknologi
Vol. 7, No. 1, Nopember 2025, Halaman 23 – 32
ISSN: 2774-1702, http://jurnal.unmuhjember.ac.id/index.php/JST

1. PENDAHULUAN

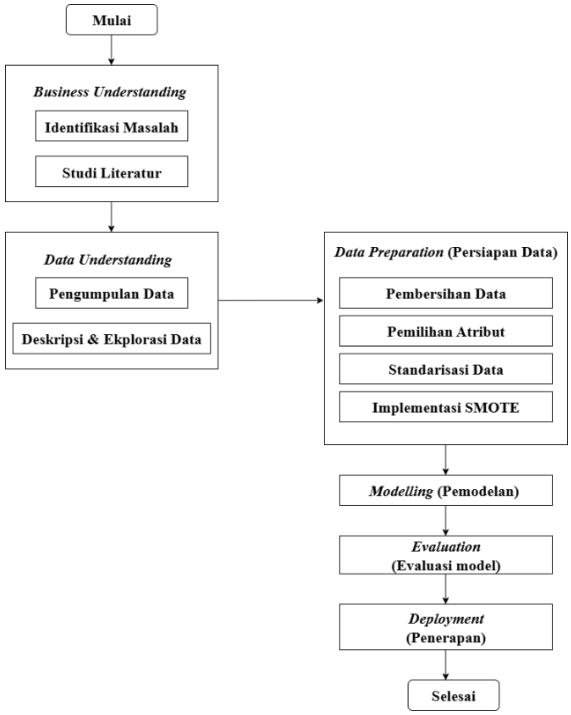
Penyakit Alzheimer merupakan bentuk demensia paling umum dan menjadi salah satu masalah kesehatan global yang serius. Dengan jumlah penderita yang diperkirakan mencapai 139 juta secara global pada tahun 2050, deteksi dini menjadi sangat penting untuk memperlambat progresivitas penyakit dan meningkatkan kualitas hidup pasien. Sayangnya, metode diagnosis konvensional masih memiliki keterbatasan dari segi biaya, waktu, dan aksesibilitas.

Seiring perkembangan teknologi, *machine learning* menawarkan solusi alternatif yang lebih efisien dan akurat dalam proses diagnosis. Salah satu algoritma yang banyak digunakan adalah *Random Forest*, karena kemampuannya yang andal dalam menangani data kompleks. Namun, tantangan berupa ketidakseimbangan kelas dalam data medis kerap mengurangi performa model. Untuk mengatasi hal tersebut, diterapkan teknik SMOTE (*Synthetic Minority Over-sampling Technique*) guna menyeimbangkan distribusi data.

Penelitian ini bertujuan untuk mengimplementasikan algoritma *Random Forest* dengan pendekatan *oversampling* SMOTE dalam klasifikasi penyakit Alzheimer, serta mengevaluasi pengaruh variasi *hyperparameter n_estimators* dan *max_depth* terhadap performa model berdasarkan metrik akurasi, *precision*, *recall*, dan *F1 score*. Diharapkan hasil penelitian ini dapat memberikan kontribusi nyata dalam pengembangan sistem pendukung keputusan berbasis data untuk deteksi dini penyakit Alzheimer.

2. METODOLOGI PENELITIAN

Metode penelitian yang digunakan dalam studi ini adalah CRISP-DM (*Cross Industry Standard for Data Mining*), sebuah pendekatan yang banyak digunakan karena menyediakan tahapan dan kerangka kerja yang terstruktur untuk proses pemodelan data. Metode ini membantu peneliti menjalankan langkah-langkah penelitian secara terarah. Tahapan-tahapan penelitian ditampilkan pada Gambar 3.1 berikut ini:



Gambar 1. Metodologi Penelitian

A. Jenis Penelitian

Penelitian ini menggunakan pendekatan kuantitatif, yaitu metode yang berlandaskan pada data konkrit berupa angka untuk menjawab hipotesis. Data dikumpulkan, diolah, dan dianalisis menggunakan teknik statistik, matematika, dan komputasi. Hasilnya disajikan dalam bentuk visual seperti tabel, grafik, atau gambar guna mendukung proses penarikan kesimpulan secara objektif.

B. *Business Understanding*

Tahap *Business Understanding* merupakan langkah awal yang sangat krusial dalam proses *data mining* menggunakan metode CRISP-DM. Pada tahap ini, akan dilakukan pemahaman yang mendalam mengenai masalah yang ingin diselesaikan menggunakan *data mining*. Adapun hal-hal yang dilakukan pada penelitian ini untuk mendapatkan pemahaman masalah yang mendalam seperti identifikasi masalah dan studi literatur.

1. Identifikasi Masalah

Permasalahan utama dalam klasifikasi penyakit Alzheimer adalah ketidakseimbangan

kelas pada *dataset* medis, di mana jumlah pasien non-Alzheimer jauh lebih banyak dibandingkan penderita Alzheimer. Ketidakseimbangan ini berisiko menyebabkan model bias terhadap kelas mayoritas. Untuk itu, pemilihan algoritma klasifikasi yang efektif sangat penting. Algoritma *Random Forest* digunakan karena mampu menangani data dengan fitur kompleks, termasuk data hilang dan *outlier*. Namun, untuk mengatasi ketidakseimbangan kelas, diperlukan penerapan metode *oversampling* yang tepat, seperti SMOTE, yang menghasilkan data sintetis dari kelas minoritas agar model dapat belajar secara seimbang. Selain itu, performa model sangat dipengaruhi oleh kombinasi parameter seperti *n_estimators* dan *max_depth*, sehingga diperlukan pengujian menyeluruh untuk memperoleh konfigurasi terbaik.

2. Studi Literatur

Studi literatur dalam tugas akhir ini dilakukan dengan mengumpulkan dan mengkaji berbagai referensi ilmiah yang berkaitan dengan penerapan algoritma *Random Forest* dan klasifikasi penyakit Alzheimer. Studi literatur yang dilakukan pada penelitian ini yaitu dengan mencari dan mempelajari jurnal-jurnal atau penelitian terdahulu yang nantinya dapat mendukung dan dapat dijadikan rujukan yang memperkuat argumentasi-argumentasi yang ada.

C. *Data Understanding* (Pemahaman Data)

Data Understanding atau pemahaman data adalah tahapan mengumpulkan data awal serta mempelajari dan mendeskripsikan data tersebut untuk bisa mengenal dan memahami data yang akan dipakai. Tahap ini juga mencoba melakukan identifikasi awal mengenai masalah yang berkaitan dengan kualitas data dan mencoba mendeteksi adanya bagian yang menarik dari data untuk membuat hipotesa awal.

1. Pengumpulan Data

Pengumpulan data pada penelitian ini menggunakan data sekunder yang diperoleh dari *platform online* (*website*) Kaggle.com dengan format .csv yang dibagikan oleh Rabie El Kharoua. *Dataset* penyakit alzheimer ini memiliki 35 atribut atau variabel, dan 2.149 *record* atau data pasien.

2. Deskripsi dan Ekplorasi *Dataset*

Deskripsi dan eksplorasi data dilakukan untuk memahami apa saja variabel yang ada pada *dataset*, tipe data masing-masing atribut (numerik, kategorikal), serta distribusi data. Secara umum langkah-langkah proses deskripsi dan ekplorasi data adalah dengan membuat profil *dataset*, visualisasi data, analisis data, serta mencari pola dan anomali data seperti korelasi, *outlier*, dan data yang hilang (*missing value*).

D. *Data Preparation* (Persiapan Data)

Pada tahap *data preparation* dilakukan persiapan data dengan menyesuaikan *dataset* agar dapat sesuai dengan kebutuhan yang akan digunakan saat tahap pemodelan. Ada beberapa hal yang akan dilakukan seperti melakukan pembersihan data, melakukan pemilihan atribut yang akan digunakan, standarisasi data, dan penerapan teknik *oversampling* menggunakan SMOTE (*Synthetic Minority Oversampling Technique*). SMOTE digunakan untuk mengatasi permasalahan data yang tidak seimbang pada kelas target, dengan cara membuat sintetis data baru pada kelas minoritas, sehingga distribusi kelas menjadi lebih seimbang dan performa model klasifikasi dapat ditingkatkan.

E. *Modeling* (Pemodelan)

Pada tahap pemodelan, peneliti membangun model klasifikasi menggunakan algoritma *Random Forest* dan melakukan pengujian terhadap berbagai kombinasi parameter untuk memperoleh performa model yang terbaik. Parameter yang diuji meliputi *split ratio* / *train:test* (90:10, 80:20, 70:30), *random state* (20, 42, dan 60), jumlah pohon keputusan atau *n_estimators* (200, 300, dan 500), serta *max_depth* (10, 12, dan 15) yang berfungsi membatasi kedalaman pohon guna mencegah *overfitting*.

Pengujian *split ratio* bertujuan mengevaluasi pengaruh proporsi data latih dan uji terhadap performa model, sementara variasi *random state* digunakan untuk memastikan kestabilan model terhadap pembagian data yang berbeda. Evaluasi terhadap parameter *n_estimators* dan *max_depth* dilakukan untuk

mengukur kontribusi kompleksitas model terhadap akurasi klasifikasi. Melalui pengujian menyeluruh ini, peneliti berupaya menghasilkan model *Random Forest* yang tidak hanya akurat, tetapi juga stabil dalam membedakan antara penderita Alzheimer dan Tidak-Alzheimer.

F. Evaluasi Model

Setelah membangun model *Random Forest* dengan berbagai kombinasi parameter, peneliti melanjutkan ke tahap evaluasi model untuk menilai performa masing-masing model. Evaluasi dilakukan menggunakan beberapa metrik klasifikasi, yaitu akurasi, presisi, *recall*, dan *f1-score*, untuk memberikan gambaran menyeluruh terhadap kualitas prediksi. Evaluasi dilakukan melalui dua skenario utama, yaitu:

1. Skenario pengujian tanpa menggunakan SMOTE.
2. Skenario pengujian menggunakan SMOTE.

G. *Deployment*

Deployment merupakan tahapan terakhir pada metode CRISP-DM. Pada penelitian ini model yang telah dibuat akan diterapkan pada aplikasi *website* diagnosa dini penyakit alzheimer menggunakan *streamlit*. Dimana nantinya pengguna akan memasukkan data klinis pengguna untuk mendapatkan hasil prediksi penyakit alzheimer dari penerapan algoritma *Random forest*.

3. HASIL DAN PEMBAHASAN

Bab ini menyajikan hasil dan pembahasan dari seluruh tahapan yang telah dilakukan dalam penelitian, mulai dari lingkungan pengujian, eksplorasi *dataset*, proses persiapan data (*preprocessing*), penerapan metode *oversampling* SMOTE, pelatihan model *Random Forest*, hingga evaluasi performa model.

Selain itu, dibahas pula skenario pengujian yang dirancang, analisis terhadap hasil yang diperoleh, serta proses *deployment* sistem. Seluruh hasil dianalisis untuk mengukur sejauh mana metode *Random Forest* berbasis *oversampling* SMOTE mampu meningkatkan akurasi dalam mendeteksi penyakit Alzheimer.

A. Lingkungan Pengujian

Lingkungan pengujian pada penelitian ini meliputi spesifikasi perangkat keras yang ditunjukkan dalam Tabel 1 dan spesifikasi perangkat lunak yang ditunjukkan dalam Tabel 2 berikut ini:

Table 1. Spesifikasi Perangkat Keras

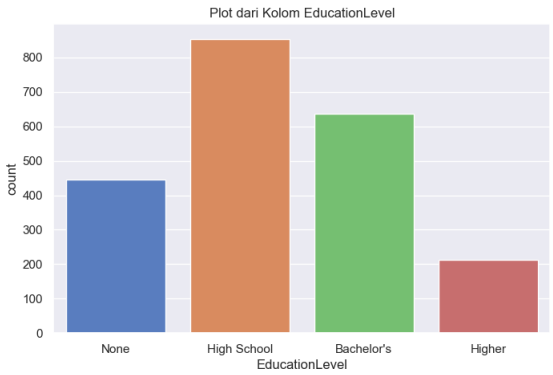
No.	Nama	Spesifikasi
1.	Laptop	ASUS-PO7DD8PT.
2.	Processor	Intel(R) Core(TM) i3-1005G1 CPU @ 1.20GHz (1.19 GHz).
3.	Memori	8.00 GB.
4.	Harddisk	238 GB.

Table 2. Spesifikasi Perangkat Lunak

No.	Nama	Spesifikasi
1.	Sistem Operasi	Windows 11 Home Single Language.
2.	Development	Visual Code, Jupyter Notebook, Python.
3.	Library	Pandas, Numpy, Matplotlib, Seaborn, Sklearn, Joblib, Streamlit.

B. Eksplorasi *Dataset*

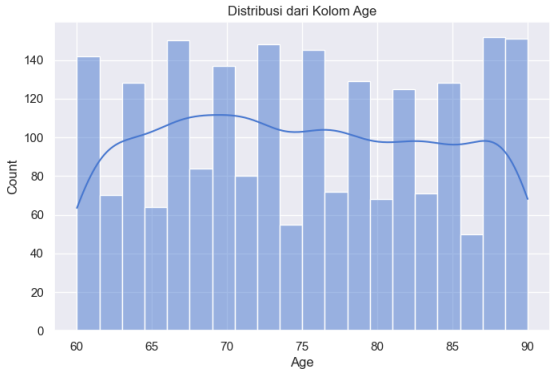
Pada tahap eksplorasi *dataset* peneliti berupaya mengidentifikasi kecenderungan umum dalam data, maupun kecocokan terhadap distribusi normal. Pemahaman terhadap distribusi data menjadi aspek krusial dalam proses persiapan data (*preprocessing*) yang tepat sebelum memasuki tahap pemodelan. Berikut merupakan contoh visualisasi distribusi data dari *dataset* yang digunakan:



Gambar 2. Distribusi Data Kategorikal

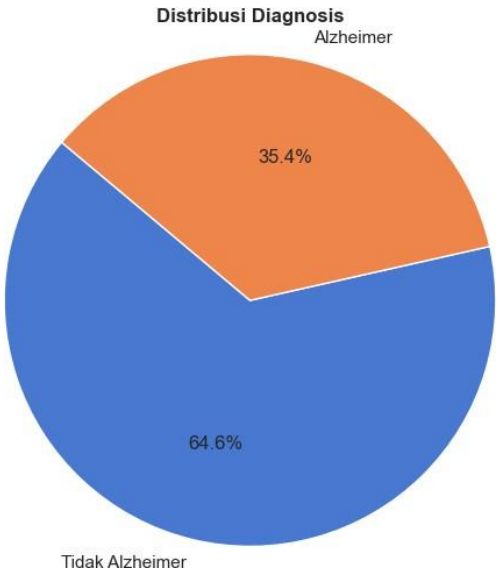
Jurnal Smart Teknologi
Vol. 7, No. 1, Nopember 2025, Halaman 23 – 32
ISSN: 2774-1702, http://jurnal.unmuhjember.ac.id/index.php/JST

Eksplorasi distribusi data kategorikal yang telah dilakukan menunjukkan bahwa sebagian besar individu dalam kumpulan data tidak mengalami Alzheimer. Dari sisi demografi, kelompok etnis Kaukasia (*Caucasian*) merupakan yang paling dominan dalam *dataset*. Berdasarkan tingkat pendidikan, lulusan sekolah menengah atas (*high school*) menjadi kelompok terbesar, disusul oleh individu yang memiliki gelar sarjana (*bachelor's degree*).



Gambar 3. Distribusi Data Numerik

Selanjutnya, peneliti mengidentifikasi adanya data yang tidak seimbang pada variabel target. Distribusi kelas pada fitur *Diagnosis* menunjukkan kondisi yang tidak seimbang (*imbalanced*), di mana terdapat 64,6% data merupakan kelas 0 (Tidak Alzheimer) dan hanya 35,4% merupakan kelas 1 (Alzheimer).



Gambar 4. Distribusi Kelas Diagnosis

Ketidakseimbangan ini dapat memengaruhi performa model klasifikasi, terutama dalam mendeteksi kasus minoritas (Alzheimer), sehingga perlu dipertimbangkan teknik penyeimbangan data seperti SMOTE pada tahap *data preparation* / *data preprocessing*.

C. Data Preparation

Pada penelitian ini, *dataset* yang digunakan telah bersih, tidak mengandung nilai kosong (*missing values*), sehingga tahapan pembersihan data dapat dilewati. Selanjutnya pada tahapan *preparation data* peneliti melakukan seleksi fitur, standarisasi fitur, dan nantinya akan dilanjutkan dengan implementasi SMOTE untuk menangani ketidakseimbangan data.

1. Seleksi Fitur

Pada tahap seleksi fitur, peneliti menghapus fitur yang tidak relevan untuk proses klasifikasi, yaitu *PatientID* dan *DoctorInCharge*, karena tidak memberikan kontribusi informatif terhadap prediksi diagnosis. Setelah itu, dilakukan pemisahan antara fitur dan target. Fitur-fitur yang digunakan dalam proses klasifikasi meliputi:

Table 3. Tabel Fitur Yang Digunakan

No.	Fitur/Atribut
1.	*Age*
2.	*Gender*
3.	*Ethnicity*
4.	*EducationLevel*
5.	*BMI*
6.	*Smoking*
7.	*AlcoholConsumption*
8.	*PhysicalActivity*
9.	*DietQuality*
10.	*SleepQuality*
11.	*FamilyHistoryAlzheimers*
12.	*CardiovascularDisease*
13.	*Diabetes*
14.	*Depression*
15.	*HeadInjury*
16.	*Hypertension*
17.	*SystolicBP*
18.	*DiastolicBP*
19.	*CholesterolTotal*
20.	*CholesterolLDL*

21. *CholesterolHDL*
22. *CholesterolTriglycerides*
23. *MMSE*
24. *FunctionalAssessment*
25. *MemoryComplaints*
26. *BehavioralProblems*
27. *ADL*
28. *Confusion*
29. *Disorientation*
30. *PersonalityChanges*
31. *DifficultyCompletingTasks*
32. *Forgetfulness*

2. Standarisasi Fitur

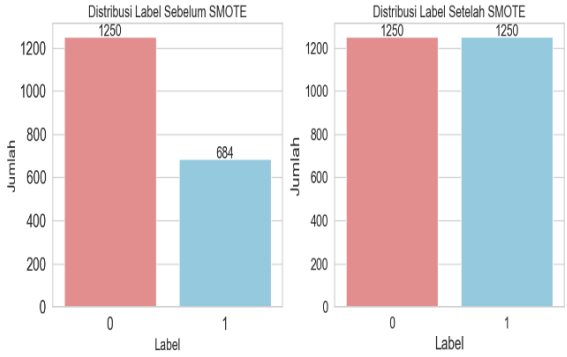
Setelah memisahkan fitur dan target, peneliti melakukan standarisasi fitur numerik menggunakan StandardScaler dari library scikit-learn. Proses ini mengubah distribusi nilai setiap fitur agar memiliki rata-rata nol dan standar deviasi satu, sehingga semua fitur berada dalam skala yang seragam. Standarisasi diperlukan untuk mencegah dominasi fitur bernilai besar dalam proses pembelajaran model, yang dapat menyebabkan bias dan meningkatkan risiko overfitting. Berikut merupakan hasil dari proses standarisasi terhadap fitur numerik:

	Age	Gender	Ethnicity	EducationLevel	BMI	Smoking	AlcoholConsumption
0	-0.212368	0	0	2	-0.655225	0	0.565923
1	1.567757	0	0	0	-0.114751	0	-0.954895
2	-0.212368	0	3	1	-1.366428	0	1.653006
3	-0.101111	1	0	1	0.851625	1	0.376930
4	1.567757	0	0	0	-0.961607	0	1.461793
5	1.233984	1	1	1	0.411764	0	-1.024794
6	-0.768658	0	3	2	1.487290	1	-1.631769
7	0.010147	0	0	1	-1.230597	0	0.640031
8	-0.323626	1	1	0	0.024598	0	0.369735
9	1.345242	0	0	0	1.081051	1	1.040419
10	1.567757	0	3	1	1.636327	0	-0.039633
11	0.343921	0	0	2	-0.719580	1	1.608725
12	1.011468	1	0	1	-0.122614	0	0.163045
13	0.343921	1	0	1	0.168375	1	0.026972
14	-1.213689	1	0	2	0.039797	0	-1.386032

Gambar 5. Standarisasi fitur

3. Penerapan *Oversampling* SMOTE

Selanjutnya, peneliti menerapkan teknik SMOTE (Synthetic Minority Over-sampling Technique) untuk mengatasi ketidakseimbangan kelas, di mana jumlah data penderita Alzheimer jauh lebih sedikit dibandingkan kelas Tidak-Alzheimer. Ketidakseimbangan ini berpotensi menimbulkan bias model terhadap kelas mayoritas. Visualisasi distribusi data sebelum dan sesudah SMOTE ditampilkan pada gambar berikut.

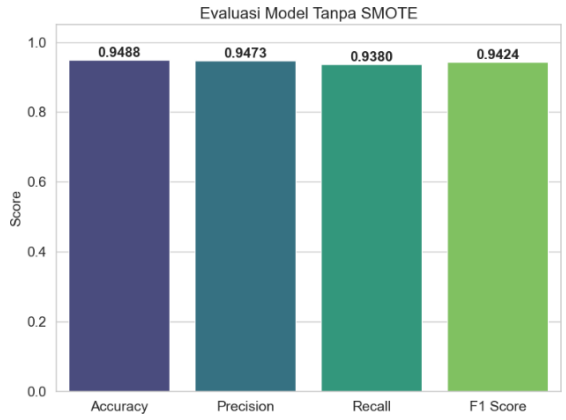


Gambar 6. Perbandingan Hasil Evaluasi Menggunakan SMOTE dan Tanpa Menggunakan SMOTE

D. Skenario Pengujian

Pada skenario pengujian, model dievaluasi menggunakan kombinasi parameter rasio data *train:test* (90:10, 80:20, 70:30), *random_state* (20, 42, 60), *n_estimators* (200, 300, 500), dan *max_depth* (10, 12, 15). Setiap konfigurasi diuji menggunakan metrik akurasi, presisi, *recall*, dan *f1-score* untuk mencari performa model terbaik dalam mendeteksi penyakit Alzheimer. Evaluasi dilakukan melalui dua skenario utama, yaitu:

1. Skenario uji tanpa menggunakan SMOTE.



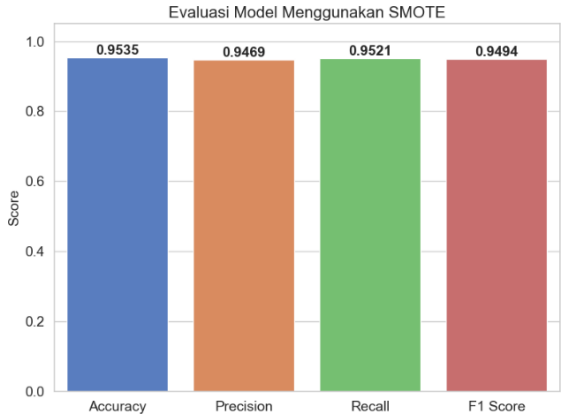
Gambar 7. Hasil Evaluasi Tanpa SMOTE

Hasil evaluasi menunjukkan bahwa konfigurasi terbaik berdasarkan keseimbangan seluruh metrik diperoleh pada kombinasi

Jurnal Smart Teknologi
Vol. 7, No. 1, Nopember 2025, Halaman 23 – 32
ISSN: 2774-1702, http://jurnal.unmuhjember.ac.id/index.php/JST

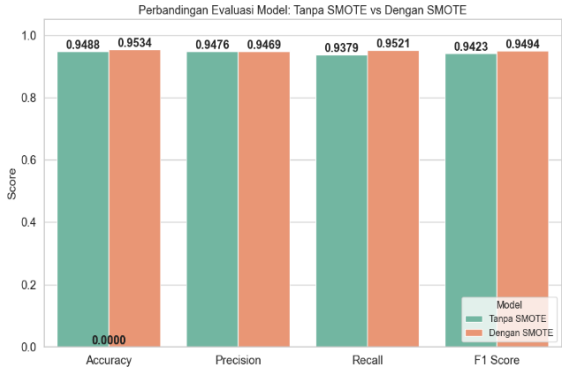
parameter *n_estimators* = 200 dan 300, *max_depth* = 15, *split ratio* = 0,1, serta *random_state* = 42. Pada konfigurasi ini, model berhasil mencapai akurasi sebesar 94,8%, *recall* 93,8%, dan *f1-score* 94,2%. Sementara itu, nilai presisi tertinggi diperoleh pada konfigurasi berbeda, yaitu *n_estimators* = 500, *max_depth* = 12, *split ratio* = 0,2, dan *random_state* = 42, dengan hasil presisi sebesar 94,7%.

2. Skenario uji menggunakan SMOTE.



Gambar 8. Hasil Evalusi Menggunakan SMOTE

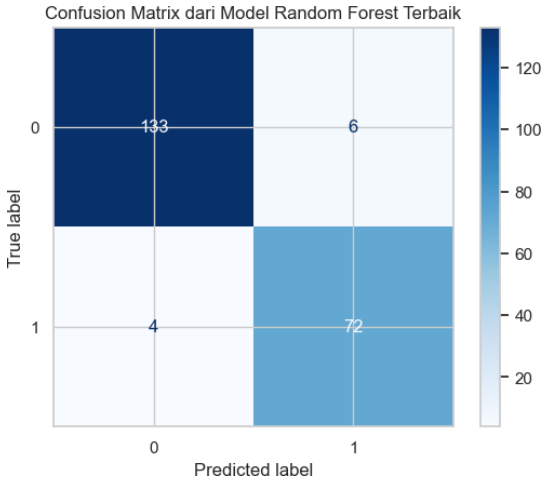
Hasil evaluasi yang dilakukan dengan menerapkan teknik *oversampling* SMOTE, konfigurasi terbaik algoritma *Random Forest* diperoleh pada kombinasi parameter *n_estimators* = 200, *max_depth* = 12, *split ratio* = 0,1, dan *random_state* = 42. Dengan konfigurasi tersebut, model berhasil mencapai akurasi sebesar 95,3%, presisi 94,69%, *recall* 95,2%, dan *f1-score* 94,9%.



Gambar 9. Perbandingan Evaluasi Model Tanpa SMOTE dan Dengan SMOTE

Berdasarkan hasil pengujian yang telah dilakukan, penerapan teknik SMOTE terbukti memberikan peningkatan yang relatif efektif terhadap performa model klasifikasi. Pada model yang tidak menggunakan SMOTE, diperoleh nilai akurasi sebesar 94,8%, presisi 94,7%, *recall* 93,8%, dan *f1-score* 94,2%. Sementara itu, setelah diterapkan teknik oversampling SMOTE, performa model meningkat menjadi akurasi 95,3%, recall 95,2%, dan f1-score 94,9%. Namun untuk nilai presisi menunjukkan penurunan nilai menjadi 94,69%. Berikut merupakan visualisasi dari perbandingan model tanpa SMOTE dan menggunakan SMOTE.

Untuk memberikan gambaran yang lebih jelas terhadap performa model klasifikasi dengan penerapan SMOTE dan kombinasi nilai parameter terbaik, ditampilkan visualisasi *confusion matrix.* Matriks ini memperlihatkan jumlah prediksi benar (*True Positives* dan *True Negatives*) serta kesalahan prediksi (*False Positives* dan *False Negatives*), yang menjadi indikator penting dalam mengevaluasi ketepatan model, terutama dalam konteks diagnosis penyakit Alzheimer.



Gambar 10. *Confusion Matrix*

Berdasarkan *confusion matrix* yang dihasilkan dari model dengan penerapan SMOTE, diketahui bahwa nilai *True Positive* (TP), yaitu jumlah kasus Alzheimer yang berhasil diprediksi dengan benar, adalah sebanyak 72. Sementara itu, nilai *True Negative*

Jurnal Smart Teknologi
Vol. 7, No. 1, Nopember 2025, Halaman 23 – 32
ISSN: 2774-1702, http://jurnal.unmuhjember.ac.id/index.php/JST

(TN), yaitu jumlah kasus Tidak-Alzheimer yang diprediksi benar sebagai Tidak-Alzheimer, tercatat sebanyak 133. Untuk kesalahan prediksi, nilai *False Positive* (FP), yaitu jumlah kasus Tidak-Alzheimer yang salah diprediksi sebagai Alzheimer, berjumlah 6, sedangkan *False Negative* (FN), yaitu jumlah kasus Alzheimer yang salah diklasifikasikan sebagai non-Alzheimer, sebanyak 4.

E. *Deployment*

Setelah model klasifikasi Alzheimer berhasil dilatih dan dievaluasi, peneliti melanjutkan ke tahap *deployment* dengan mengembangkan aplikasi web berbasis *streamlit* agar model dapat digunakan secara praktis oleh pengguna. *Streamlit* dipilih karena merupakan *framework open-source* berbasis Python yang mendukung pengembangan aplikasi interaktif secara cepat, serta mudah diintegrasikan dengan model *machine learning*. Selain itu, *Streamlit* memungkinkan penyajian antarmuka pengguna yang sederhana dan responsif. Pada tahap *deployment* ini, sebelum peneliti menggunakan model klasifikasi dalam aplikasi peneliti menyimpan model yang telah dilatih ke dalam file berekstensi .pkl dengan menggunakan *library joblib*. Selain itu, peneliti juga menyimpan variabel *standard_scaler* yang sebelumnya digunakan untuk menstandarisasi data masukan, agar proses transformasi terhadap data baru tetap konsisten dengan proses saat pelatihan model. Berikut merupakan kode Python yang peneliti gunakan untuk menyimpan model klasifikasi *Random Forest* beserta variabel *standard_scaler*:

```
# === Simpan Standarisasi ===
joblib.dump(standard_scaler, 'scaler.pkl')
print("berhasil")

# === Simpan Model ===
joblib.dump(best_model, 'model_random_forest_alzheimer.pkl')
print("Model disimpan sebagai 'model_random_forest_alzheimer.pkl")

✓ 0.1s

berhasil
Model disimpan sebagai 'model_random_forest_alzheimer.pkl
```

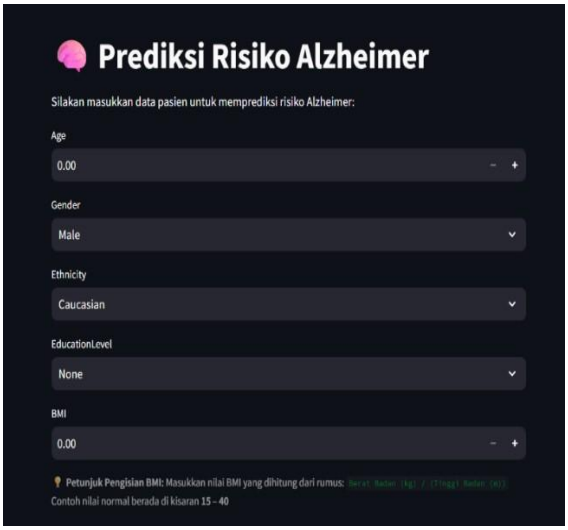
Gambar 11. Penyimpanan Variabel *Standard_scaler* dan Model

Setelah menyimpan model klasifikasi dan variabel *StandardScaler*, peneliti melanjutkan tahap implementasi dengan merancang antarmuka aplikasi web menggunakan *streamlit*. Proses diawali dengan mengimpor *library* yang diperlukan, seperti *streamlit*, *joblib*, dan *library* pendukung klasifikasi lainnya. Selanjutnya, peneliti memuat model *machine learning* beserta objek *StandardScaler* yang telah disimpan sebelumnya menggunakan *joblib*. Berikut merupakan kode Python untuk proses *import library* yang dibutuhkan:

```
import streamlit as st
import pandas as pd
import numpy as np
import joblib
# === Load Model, Scaler ===
model = joblib.load("model_random_forest_alzheimer.pkl")
scaler = joblib.load("scaler.pkl")
```

Gambar 12. *Import Library* dan Memuat Model

Setelah semua komponen berhasil dimuat, langkah selanjutnya adalah membangun tampilan antarmuka aplikasi, yang dapat menyajikan hasil prediksi secara langsung berdasarkan input yang diberikan. Berikut merupakan tampilan aplikasi web berbasis *streamlit* yang sudah dibuat.



Gambar 13. Tampilan Web Berbasis *Streamlit*

4. KESIMPULAN

Berdasarkan hasil penelitian mengenai klasifikasi penyakit Alzheimer menggunakan algoritma *Random Forest* yang dikombinasikan dengan teknik *oversampling* SMOTE, dapat disimpulkan bahwa model mampu

menghasilkan performa klasifikasi yang sangat baik. Penerapan SMOTE terbukti efektif dalam mengatasi ketidakseimbangan kelas pada data, yang secara efektif meningkatkan akurasi dan keandalan model. Model menghasilkan akurasi sebesar 95,3%, presisi 94,6%, *recall* 95,2%, dan *f1-score* 94,9%, yang menunjukkan bahwa model tidak hanya akurat, tetapi juga seimbang dalam mendeteksi kasus positif dan negatif. Selain itu, pengujian dengan berbagai kombinasi nilai hyperparameter menunjukkan bahwa konfigurasi terbaik diperoleh pada *n_estimators* sebesar 200 dan *max_depth* sebesar 12, dengan rasio pembagian data *train-test* sebesar 90:10 dan *random_state* = 42.

5. REFERENSI

Alzheimer's Association. (2019). 2019 Alzheimer's disease facts and figures includes a special report on Alzheimer's detection in the primary care setting: connecting patients and physicians. Retrieved from https://alz.org/media/Documents/alzheimers-facts-andfigures-2019-r.pdf.

Aziz, M.I., et al. 2023. "Analisis Metode Ensemble Pada Klasifikasi Penyakit Jantung Berbasis Decision Tree", *Jurnal Media Informatika Budidarma*, 7(1), 1 – 12.

Berry, M. J. ., & Linoff, G. S. (2004). Data Mining Techniques. United States: Wiley Publishing, Inc.

Breiman, L. (2001). *Random forests*. *Machine Learning*, 45(1), 5–32.

Devella, S. Y. Yohannes, and F. N. Rahmawati, "Implementasi Random Forest Untuk Klasifikasi Motif Songket Palembang Berdasarkan SIFT," JATISI (Jurnal Tek. Inform. dan Sist. Informasi), vol.7, no. 2, pp. 310–320, 2020.

Dewi, N.R., et al. 2023. "Algoritma *K-Nearest Neighbor (K-NN)* dan *Single Layer Perceptron* (SLP) Untuk Klasifikasi Penyakit Alzheimer", 9(2), 92 – 101.

Dinova, D. B., & Prasetiyo, B. 2024. "Implementasi Random Forest dalam Klasifikasi Kanker Paru-Paru". Journal Of Informatics Engineering, 5(2), 27 – 31.

Dominicus, D. A., Setiawan, N. Y., & Wicaksono, S. A. (2020). Prediksi Kecenderungan Pelanggan Telat Bayar pada Layanan Pembiayaan Adira Finance Saluran E-Commerce. Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer, 4(4), 1300–1307.

Freund, Y., & Schapire, R. E. (1999). A Short Introduction to Boosting. Journal of Japanase Society for Artificial Intelligence, 14(5), 771–780.

Handayani, P., dkk. 2024. "*Machine Learning* Klasifikasi Status Gizi Balita Menggunakan Algoritma *Random Forest* ". Kajian Ilmiah Informatika dan Komputer, 4(6), 3064 – 3072.

Lakhan, S. E. (2019, December 4). Alzheimer Disease Clinical Presentation: History, Physical Examination, Stages of Alzheimer Disease. Retrieved January 12, 2020, from https://emedicine.medscape.com/article/1134817-clinical#b4.

Liu, J., Kong, X., Xia, F., Bai, X., Wang, L., Qing, Q., & Lee, I. 2018. Artificial Intelligence in the 21st century. *IEEE Acces*, 34403-34421.

Lorase, D. T. 2005. Discovering Knowledge In Data. Canada: Wiley Interscience.

M. A. Hasanah, S. Soim, dan A. S. Handayani. (2021). "Implementasi CRISP-DM Model Menggunakan Metode Decision Tree dengan Algoritma CART untuk Prediksi Curah Hujan Berpotensi Banjir," vol. 5, no. 2.

Mantas, C. J., & Abellan, J. 2014. Credal-C4.5: Decision Tree Based on Imprecise Probabilities to Classify Noisy Data. Expert System With Application, 4625–4637.

Muzaky, A., dkk. 2024. "Menerapkan Metode Klasifikasi pada Data Uji Emisi Kendaraan di Jakarta dengan Menggunakan Jupyter Notebook". *JPSII (Jurnal Pengembangan Sistem Informasi dan Informatika)*, 5(2), 74 – 84.

Prasetyo, E. 2014. Mengolah Data Menjadi Informasi Menggunakan Matlab. Yogyakarta: Andi.

Putri, R, A,. 2024. "Pemodelan Algoritma Random Forest Untuk Klasifikasi Log Acces Jenis Domain Pada PANDI (Pengelola Nama Domain Internet Indonesia)". Jakarta: Program Studi Teknik Informatika Universitas Islam Negeri Syarif Hidayatullah.

Rahman, S., et al. 2023. Python: Dasar-dasar Pemrograman Berorientasi Objek. Deli Serdang: CV TAHTA MEDIA GROUP.

Rozy, A. 2024. "Penerapan Random Forest Untuk Prediksi Virus Hepatitis C". FIMERKOM: Journal of Information Systems and Technology, 1(1), 19 -23.

Santosa, B. 2007. Data Mining : Teknik Pemanfaatan Data untuk Keperluan Bisnis. Yogyakarta: Graha Ilmu.

Sidiq, S., Alfian, A. and Mabrur, N.S., 2025. Pengembangan Model Prediksi Risiko Diabetes Menggunakan Pendekatan AdaBoost dan Teknik Oversampling SMOTE. *Jurnal Ilmiah Informatika dan Ilmu Komputer (JIMA-ILKOM)*, *4*(1), pp.13-23.

Utami, S.F. 2020. "Penerapan Data Mining Algoritma Decision Tree Berbasis PSO". *Jurnal SAINTEKS (Seminar Nasional Teknologi Komputer & Sains)*, Februari: 677 – 681.

Wildah, S.K., et al. 2020. "Deteksi Penyakit Alzheimer Menggunakan Algoritma *Naïve Bayes* dan *Correlation Based Feature Selection*", 7(2), 166 – 173.

Wu, Xi., Kumar, V., Quinlan, J. R., Ghosh, J., Yang, Q., Motoda, H., … Steinbarg. (2008). Top 10 algorithm in data mining. Knowl Inf Syst, 14, 1–37.