

Evaluasi Performa XGBoost dengan *Oversampling* dan *Hyperparameter Tuning* untuk Prediksi Alzheimer

Performance Evaluation of XGBoost with Oversampling and Hyperparameter Tuning for Alzheimer's Prediction

Furqon Nurbaril Yahya¹, Mochammad Anshori², Ahsanun Naseh Khudori³
^{1,2,3}Informatika, Fakultas Sains dan Teknologi, Institut Teknologi, Sains, dan Kesehatan RS.DR.
Soepraoen Kesdaam V/BRW
E-mail: ¹furqonyahya1@gmail.com, ²moanshori@itsk-soepraoen.ac.id, ³ahsanunnaseh@itsk-soepraoen.ac.id

Abstrak

Alzheimer adalah gangguan neurodegeneratif yang mempengaruhi kemampuan kognitif dan memori, deteksi dini sangat penting untuk pengobatan yang tepat. Namun, untuk mendeteksi Alzheimer memerlukan biaya yang tinggi, sehingga penggunaan *machine learning* bisa menjadi alternatif yang lebih efisien. Salah satu tantangan utama dalam penerapan *machine learning* untuk mendeteksi Alzheimer adalah ketidakseimbangan data, di mana jumlah kasus positif (Alzheimer) jauh lebih sedikit daripada kasus negatif (sehat), yang berdampak pada kinerja model. Penelitian ini bertujuan untuk mengidentifikasi pengaruh teknik *oversampling* dan *hyperparameter tuning* terhadap hasil model XGBoost dalam prediksi Alzheimer. Empat eksperimen dilakukan untuk melihat masing-masing performa terhadap model, yaitu: (1) model dasar XGBoost, (2) XGBoost dengan *oversampling*, (3) XGBoost dengan *hyperparameter tuning*, dan (4) XGBoost dengan kombinasi *oversampling* dan *hyperparameter tuning*. Hasil penelitian menunjukkan bahwa eksperimen kedua (XGBoost + *Oversampling*) menghasilkan performa terbaik yaitu dengan *recall* 96,1%, *Persisi* 94%, akurasi 95,3%, dan *F1-Score* 95%. Temuan ini menunjukkan bahwa penerapan *oversampling* dapat meningkatkan kinerja model dalam mengatasi masalah ketidakseimbangan data. Penelitian ini memberikan kontribusi dalam pengembangan model deteksi Alzheimer dengan menekankan pentingnya penanganan ketidakseimbangan data.

Kata kunci: *XGBoost, Oversampling, Hyperparameter Tuning, Prediksi Dini, Alzheimer.*

Abstract

Alzheimer's is a neurodegenerative disorder that affects cognitive ability and memory, early detection is crucial for proper treatment. However, detecting Alzheimer's is costly, so using machine learning can be a more efficient alternative. One of the main challenges in applying machine learning to detect Alzheimer's is data imbalance, where the number of positive (Alzheimer's) cases is much less than negative (healthy) cases, which impacts the performance of the model. This study aims to assess the effect of oversampling techniques using SMOTE and hyperparameter tuning on the results of the XGBoost model in Alzheimer's prediction. Four experiments were conducted to see the performance of each model, namely: (1) XGBoost model, (2) XGBoost with oversampling, (3) XGBoost with hyperparameter tuning, and (4) XGBoost with a combination of oversampling and hyperparameter tuning. The results showed that the second experiment (XGBoost + Oversampling) produced the best performance with recall 96.1%, Precision 94%, accuracy 95.3%, and F1-Score 95%. This finding shows that the application of oversampling can improve the performance of the model in overcoming data imbalance problems. This research contributes to the development of Alzheimer's detection models by emphasizing the importance of handling data imbalance.

Keywords: *XGBoost, Oversampling, Hyperparameter Tuning, Prediksi Dini, Alzheimer*

1. PENDAHULUAN

Alzheimer merupakan penyakit neurodegeneratif yang ditandai oleh penurunan fungsi kognitif dan kemampuan individu untuk menjalani aktivitas sehari-hari secara mandiri. Berbagai faktor risiko seperti genetik, usia lanjut, gangguan pembuluh darah, cedera kepala, faktor lingkungan, dan infeksi, berperan penting dalam perkembangannya[1]. Berdasarkan data Alzheimer's Disease International, terdapat sekitar 50 juta kasus demensia pada tahun 2018 di seluruh dunia, dan angka ini diproyeksikan meningkat tiga kali lipat pada tahun 2050, dengan sebagian besar kasus terjadi di negara berpenghasilan rendah dan menengah[2]. Dampaknya yang signifikan, baik dari segi individu, keluarga, maupun ekonomi global, dengan biaya tahunan mencapai US\$1 triliun[3], menjadikan Alzheimer sebagai fokus penting untuk penelitian.

Diagnosis Alzheimer biasanya membutuhkan intervensi tenaga medis profesional menggunakan teknologi canggih seperti MRI. Metode ini tidak hanya mahal, tetapi juga memerlukan kualitas data yang tinggi, banyak pemeriksaan biomarker, dan sesi pencitraan yang kompleks[4]. Tantangan ini memunculkan kebutuhan akan solusi yang lebih efisien dan terjangkau. Teknologi digital, khususnya *machine learning*, telah menawarkan pendekatan inovatif untuk menganalisis data medis secara akurat dan efisien[5]. *Machine learning* mampu mengidentifikasi pola yang kompleks, memproses data skala besar, dan menghasilkan wawasan yang relevan[6]. Hal ini berpotensi mempercepat proses diagnosis dan mengurangi biaya keseluruhan. Dengan integrasi teknologi ini, diagnosis dapat dilakukan lebih cepat, mengurangi biaya, serta memberikan hasil yang konsisten tanpa risiko kesalahan manusia[7].

Machine learning mempunyai 3 jenis metode yaitu *supervised learning*, *unsupervised learning*, dan *reinforcement learning*. *Supervised Learning* merupakan algoritma yang dirancang untuk memahami hubungan antar fitur-fitur dalam dataset pelatihan, sehingga mampu memprediksi label pada data baru[8]. Sementara itu, *unsupervised learning* memanfaatkan data tanpa label untuk mengidentifikasi pola-pola tertentu, yang bertujuan untuk mengelompokkan fitur yang memiliki kesamaan nilai[9]. Berbeda dengan kedua pendekatan tersebut, *reinforcement learning* melibatkan agen yang mempelajari strategi optimal melalui interaksi dengan lingkungan, kemudian menerima umpan balik berupa hadiah atau penalti atas tindakannya, dan menggunakan informasi tersebut untuk memperbaiki strategi pengambilan keputusan secara bertahap[10].

Salah satu metode dalam *machine learning* yang relevan untuk kasus ini adalah *supervised learning*, karena metode ini memanfaatkan data berlabel untuk tujuan klasifikasi dan prediksi[11]. Di antara algoritma *supervised learning* yang banyak digunakan, Extreme Gradient Boosting (XGBoost) menonjol karena kecepatannya dan efisiensinya, dengan performa sepuluh kali lebih cepat dibandingkan algoritma lain[12]. Namun, performa XGBoost sangat bergantung pada pemilihan nilai hyperparameter yang optimal. Proses *hyperparameter tuning* menjadi langkah penting untuk meningkatkan akurasi dan keandalan model dalam klasifikasi[13].

Namun, penerapan *machine learning* tidak lepas dari tantangan, salah satunya adalah masalah ketidakseimbangan data (*imbalanced data*) yang sering ditemukan dalam bidang medis. Ketidakseimbangan ini terjadi ketika distribusi data antara kelas mayoritas dan minoritas tidak seimbang, yang dapat menyebabkan model lebih memahami data mayoritas dibandingkan dengan data minoritas[14][15]. Teknik *oversampling*, seperti *Synthetic Minority Oversampling Technique* (SMOTE), digunakan untuk mengatasi masalah ini dengan menghasilkan data sintetis untuk kelas minoritas, sehingga distribusi data menjadi lebih seimbang[16].

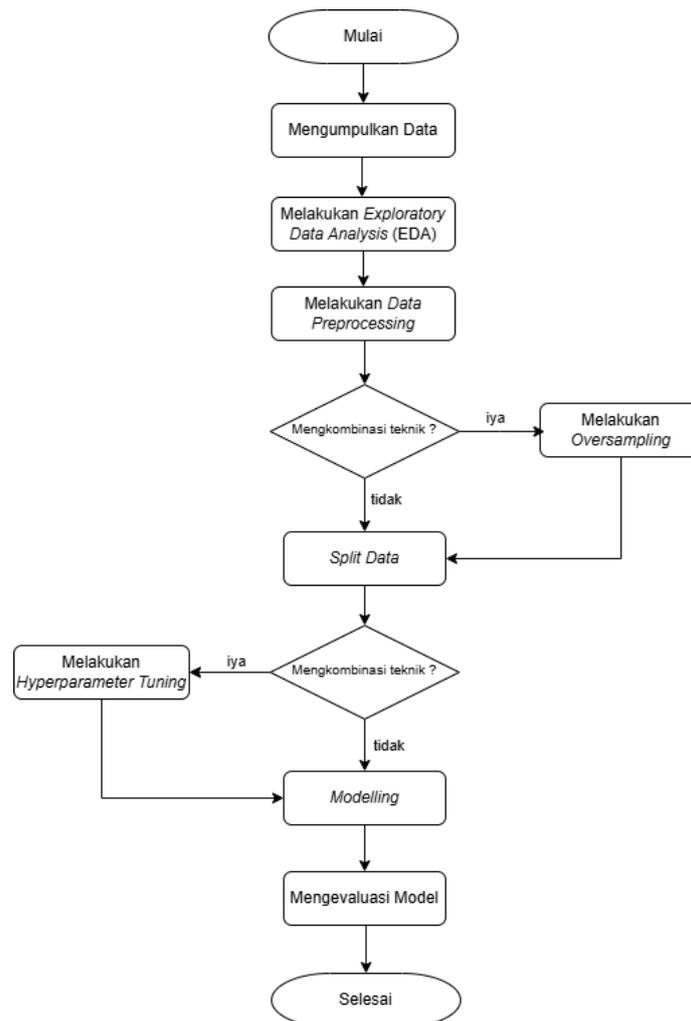
Studi terdahulu oleh [17] dan [18] telah mengeksplorasi metode klasifikasi penyakit Alzheimer dengan berbagai pendekatan. Namun, pendekatan yang digunakan belum memanfaatkan teknik *oversampling* yang berpotensi memperbaiki akurasi model prediksi pada kedua kelas. Selain itu, teknik *hyperparameter tuning* yang bertujuan untuk mengoptimalkan performa model juga belum diterapkan dalam penelitian tersebut.

Berdasarkan latar belakang yang sudah dipaparkan di atas, penelitian ini dilakukan untuk mengidentifikasi kombinasi terbaik antara teknik *oversampling* dan *hyperparameter tuning* pada model XGBoost dalam mendeteksi dini penyakit Alzheimer. Diharapkan, hasil penelitian ini dapat memberikan kontribusi berarti dalam pengembangan alat prediksi yang lebih presisi dan

efisien, sehingga membantu perencanaan intervensi yang lebih tepat.

2. METODE PENELITIAN

Penelitian ini akan mengaplikasikan kombinasi antara teknik *oversampling* (SMOTE) dan *hyperparameter tuning* (Grid Search) untuk mengidentifikasi pengaruh dari setiap kombinasi teknik yang diterapkan pada model XGBoost untuk memprediksi dini Alzheimer. SMOTE digunakan untuk mengatasi masalah ketidakseimbangan data dengan cara menambah jumlah sampel pada kelas minoritas. Sedangkan, Grid Search akan digunakan untuk mencari kombinasi nilai *hyperparameter* terbaik, guna meningkatkan efektivitas dan performa model secara keseluruhan.



Gambar 1. Diagram Penelitian

Gambar 1 menunjukkan tahapan penelitian yang dilakukan mulai dari pengumpulan data, *data preprocessing*, *split data*, *oversampling*, *hyperparameter tuning*, *modelling*, hingga evaluasi model. Setiap tahapan memiliki peran penting dalam memastikan bahwa model yang dibangun dapat memberikan hasil yang optimal dan akurat. Berbagai eksperimen kombinasi teknik, seperti

penggunaan *oversampling* dan *hyperparameter tuning*, akan dilakukan untuk menguji pengaruh masing-masing kombinasi teknik terhadap performa model dalam prediksi Alzheimer.

2.1 Pengumpulan Data

Tahap awal penelitian ini adalah pengumpulan data. Dataset yang digunakan pada penelitian ini merupakan dataset tentang penyakit Alzheimer yang diperoleh dari *platform* Kaggle yang berjudul [Alzheimer Disease](#)[19]. Dataset dikumpulkan dari berbagai sumber, termasuk fasilitas kesehatan seperti rumah sakit, klinik, serta hasil penelitian yang mencatat riwayat medis pasien dengan rentang usia 60 hingga 100 tahun. Dataset ini memiliki 2149 sampel data, dengan 34 atribut yang terdiri dari 33 variabel sebagai fitur dan 1 variabel target yaitu Diagnosis.

2.2 Exploratory Data Analysis (EDA)

Proses ini bertujuan untuk memahami sifat dan karakteristik data, serta mengidentifikasi wawasan awal yang dapat diperoleh dari dataset[20]. Langkah pertama adalah melihat distribusi data pada variabel numerik maupun kategorikal dan pengecekan nilai yang kosong. Selanjutnya, dilakukan analisis distribusi pada fitur target yaitu Diagnosis untuk mendeteksi adanya ketidakseimbangan kelas, yang dapat memengaruhi saat pelatihan model.

2.3 Data Preprocessing

Pada tahap awal *data preprocessing*, data diperiksa untuk mengidentifikasi adanya duplikasi data. Selain tidak diperlukan, data duplikat hanya meningkatkan waktu komputasi, menghabiskan memori, dan dapat menyebabkan ketidakseimbangan dalam data[21]. Setelah data terbebas duplikasi, dilakukan evaluasi terhadap relevansi fitur yang tersedia. Pemilihan fitur yang tepat dapat meningkatkan efisiensi komputasi, menyederhanakan model, dan meningkatkan akurasi prediksi[22].

Proses berikutnya adalah normalisasi, yang bertujuan untuk memastikan setiap fitur berada dalam skala yang seragam sebelum digunakan dalam pelatihan model, sehingga mampu meningkatkan kinerja secara signifikan sekaligus mempercepat proses pembelajaran pada model *machine learning*[23][24][25]. Pendekatan yang paling banyak digunakan untuk normalisasi data adalah MinMax Scaler. Dimana nilai tertinggi diubah menjadi 1, sedangkan nilai terendah menjadi 0 dan nilai lainnya berada dalam rentang desimal antara 0 dan 1[26].

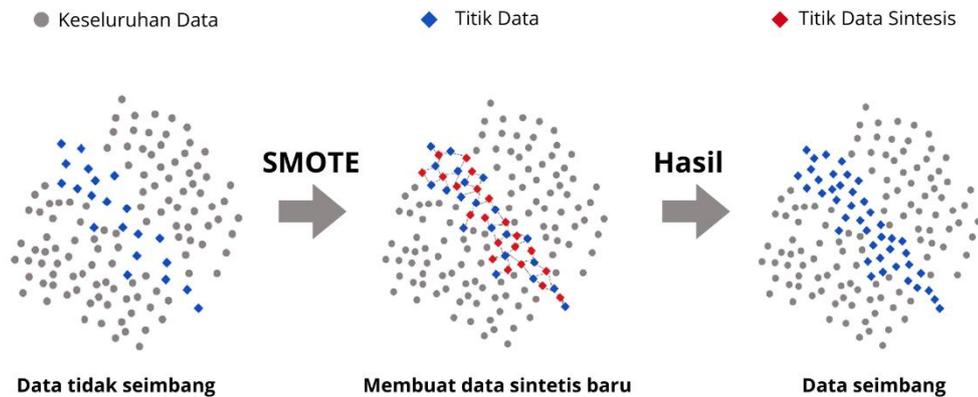
$$X_{scaled} = \frac{(X - X_{min})}{(X_{max} - X_{min})} \quad (1)$$

Dalam rumus (1), X_{scaled} merupakan nilai yang sudah dinormalisasi atau diubah skalanya. Proses normalisasi dilakukan dengan mengurangi nilai X dengan nilai minimum X_{min} yang terdapat pada fitur tersebut, kemudian hasilnya dibagi dengan selisih antara nilai maksimum X_{max} dan nilai minimum X_{min} dari fitur yang bersangkutan. Hal ini meningkatkan keseragaman skala antar fitur, yang dapat mempercepat proses pelatihan dalam algoritma *machine learning* serta meningkatkan efisiensi pada saat pemodelan.

2.4 Oversampling

Dalam kasus dataset yang tidak seimbang, di mana kelas minoritas memiliki jumlah sampel yang jauh lebih sedikit, peningkatan akurasi prediksi pada kelas minoritas menjadi tantangan tersendiri. Oleh karena itu, teknik *oversampling* seperti SMOTE digunakan untuk menyeimbangkan distribusi kelas dengan menghasilkan sampel baru dari kelas minoritas,

sehingga dapat meningkatkan kemampuan model untuk mengklasifikasikan kelas minoritas[27].



Gambar 2. Proses *oversampling* dengan SMOTE

Merujuk ilustrasi pada Gambar 2, dimana distribusi data yang awalnya tidak seimbang pada kelas target, kemudian dilakukan teknik *oversampling* dengan SMOTE untuk menghasilkan data sintesis baru pada kelas minoritas. Dengan adanya data sintesis, menghasilkan distribusi data yang lebih seimbang pada kelas target.

2.5 Split Data

Untuk memastikan bahwa *dataset* dapat dievaluasi dengan akurat oleh model, *dataset* dibagi menjadi dua bagian. 80% baris *dataset* dipilih secara acak untuk pelatihan dan 20% sisanya digunakan untuk pengujian[28]. Teknik ini adalah *best practice* yang umum dilakukan pada penelitian.

2.6 Hyperparameter Tuning

Setelah proses *split data*, langkah selanjutnya adalah *hyperparameter tuning* untuk mencari dan menetapkan kombinasi nilai *hyperparameter* terbaik untuk model yang akan dilatih, karena pemilihan nilai *hyperparameter* memengaruhi performa dari model[29]. *Hyperparameter* seperti *n_estimators* (menentukan jumlah pohon dalam model), *max_depth* (mengontrol kedalaman maksimum pohon), serta *learning rate* (mengatur seberapa besar langkah yang diambil untuk memperbarui model pada setiap iterasi)[30] dipilih dalam penelitian ini. Untuk menemukan nilai *hyperparameter* terbaik, metode Grid Search diterapkan dalam proses ini. Metode tersebut mampu menguji seluruh kombinasi nilai *hyperparameter* dalam ruang pencarian yang telah ditentukan. Metode ini mampu memberikan hasil yang lebih komprehensif daripada metode lain, seperti Random Search yang hanya menguji kombinasi secara acak dan tidak menjamin menemukan *hyperparameter* terbaik[31].

2.7 Modelling

Tahap ini melibatkan proses pelatihan model dengan didasarkan pada beragam kombinasi teknik dari proses sebelumnya. Kombinasi tersebut mencakup penggunaan model XGBoost sebagai model dasar, penerapan *oversampling* pada XGBoost, *hyperparameter tuning* pada XGBoost, serta XGBoost yang dikombinasikan dengan *oversampling* dan *hyperparameter tuning*. Proses ini bertujuan untuk melatih model menggunakan pendekatan yang paling

sesuai, agar model mampu mengenali pola data secara optimal dalam memprediksi kelas target.

2.8 Evaluasi Model

Setelah model dilatih, langkah penting berikutnya adalah mengevaluasi performa model. Berbagai metrik evaluasi digunakan untuk mengukur kinerja model[32], yaitu akurasi, presisi, *recall* dan F1-score dengan rumus (2), (3), (4), dan (5).

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \quad (2)$$

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

$$Presisi = \frac{TP}{TP+FP} \quad (4)$$

$$F1 - Score = 2 * \frac{Presisi*Recall}{Presisi+Recall} \quad (5)$$

Akurasi, rasio jumlah data yang diprediksi benar terhadap jumlah total data. Recall, yakni rasio jumlah data prediksi benar terhadap jumlah data yang sebenarnya positif. Presisi, adalah rasio jumlah data prediksi yang benar terhadap jumlah data yang diprediksi positif. *F1-Score* adalah hasil perhitungan kombinasi antara *recall* dan presisi. Dengan mengevaluasi model, kita dapat mengukur hasil prediksi berdasarkan empat kemungkinan hasil. *True Positive* (TP) adalah ketika model benar dalam menebak sesuatu sebagai positif, sesuai dengan kondisi sebenarnya. *True Negative* (TN) adalah ketika model benar dalam menebak sesuatu sebagai negatif, sesuai kenyataan juga. *False Positive* (FP) terjadi ketika model salah menebak sesuatu sebagai positif, padahal sebenarnya negatif. Terakhir, *False Negative* (FN) adalah ketika model salah menebak sesuatu sebagai negatif, padahal sebenarnya positif. Keempat hasil ini membantu kita memahami kapan model benar atau salah dalam memprediksi sesuatu.

3. HASIL DAN PEMBAHASAN

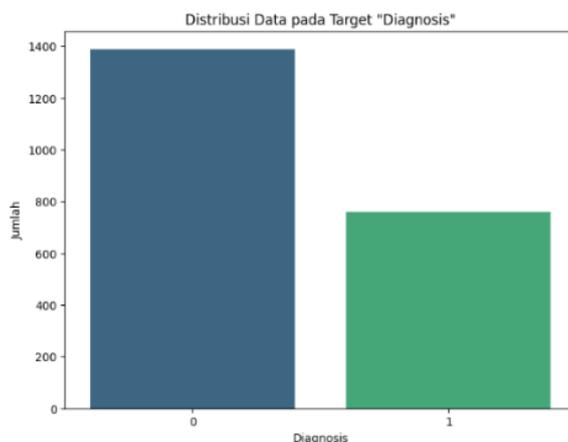
Dataset Alzheimer diperoleh dari *website* Kaggle Dataset. Dataset terdiri dari 2149 data dengan 33 variabel prediktor dan 1 variabel target yaitu Diagnosis. Berikut ini adalah penjelasan lengkap mengenai fitur – fitur yang digunakan pada pada dataset Alzheimer beserta tipe data atributnya.

Tabel 1. Deskripsi fitur pada dataset

Tipe Data	Nama Fitur	Nilai Kosong
Numerikal (<i>Int64, float64</i>)	<i>Patient ID, AgeGender, Ethnicity, EducationLevel, BMI, Smoking, AlcoholConsumption, PhysicalActivity, DietQuality, SleepQuality, FamilyHistoryAlzheimers, CardiovascularDisease, Diabetes, Depression, HeadInjury, Hypertension, SystolicBP, DiastolicBP, CholesterolTotal, CholesterolLDL, CholesterolHDL, CholesterolTriglycerides, MMSE, FunctionalAssessment, MemoryComplaints, BehavioralProblems, ADL, Confusion, Disorientation, PersonalityChanges, DifficultyCompletingTasks, Forgetfulness, Diagnosis</i>	Tidak ada
Kategorikal (<i>object</i>)	<i>DoctorInCharge</i>	Tidak ada

Pada tabel 1, menyajikan deskripsi mengenai setiap fitur yang terdapat dalam dataset, mencakup penjelasan tentang tipe data dari setiap fitur, serta informasi mengenai jumlah nilai yang kosong. Setelah pengumpulan data adalah tahap *Exploratory Data Analysis* (EDA) yang

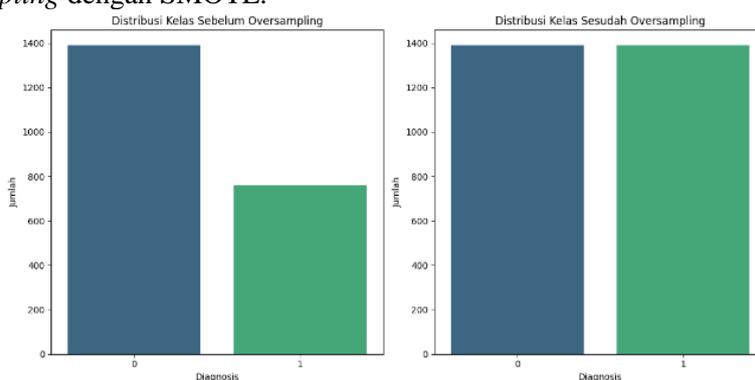
bertujuan untuk memahami distribusi data, memperoleh wawasan, serta membangun intuisi terhadap data yang akan diolah.



Gambar 3. Distribusi data pada fitur target

Berdasarkan proses EDA, diperoleh temuan bahwa terjadi ketidakseimbangan dalam distribusi data pada kelas target yang bisa dilihat pada Gambar 3, dimana jumlah kelas yang sehat (kelas 0) lebih banyak yaitu 1389 data, dibandingkan dengan jumlah dari kelas yang sakit (kelas 1) yaitu 760 data.

Ketidakseimbangan distribusi data ini menjadi perhatian utama, sehingga tahap *data preprocessing* diperlukan untuk memastikan *dataset* siap digunakan untuk proses selanjutnya. Berbagai prosedur dilakukan selama tahap *data preprocessing*. Hasil pemeriksaan menunjukkan bahwa dataset tidak mengandung data duplikat. Selain itu, ditemukan fitur yang tidak relevan seperti fitur *PatientID* yang bernilai unik dan fitur *DoctorInCharge* mempunyai nilai keseluruhan yang sama. Oleh karena itu, fitur tersebut diputuskan untuk dihapus dari dataset. Proses terakhir adalah normalisasi data menggunakan metode *Min-Max Scaler*. Proses ini mengubah semua nilai ke dalam rentang 0 hingga 1. Dengan menyamakan skala nilai pada setiap fitur, dapat meningkatkan kinerja model selama proses pelatihan. Setelah nilai-nilai pada setiap fitur dinormalisasi untuk menyamakan skala, langkah berikutnya adalah menangani ketidakseimbangan kelas dalam dataset. Teknik yang digunakan untuk mengatasinya adalah teknik *oversampling* dengan *SMOTE*.



Gambar 4. Distribusi kelas Diagnosis sebelum dan setelah *oversampling*

Perbandingan antara distribusi kelas sebelum dan setelah *oversampling* dapat dilihat pada gambar 4, di mana kedua kelas kini memiliki jumlah kelas yang lebih seimbang. Setelah penerapan *SMOTE*, kelas 1 yang sebelumnya minoritas berhasil diseimbangkan, sehingga model dapat dilatih dengan representasi kelas yang lebih proporsional. Dengan distribusi kelas yang kini lebih seimbang, fokus selanjutnya adalah mempersiapkan dataset untuk proses pelatihan dan

evaluasi dengan komposisi 80% untuk data latih dan 20% untuk data uji. Pembagian ini dilakukan untuk memastikan bahwa model memiliki proporsi data latih yang cukup tanpa mengurangi data uji yang akan digunakan untuk mengevaluasi model.

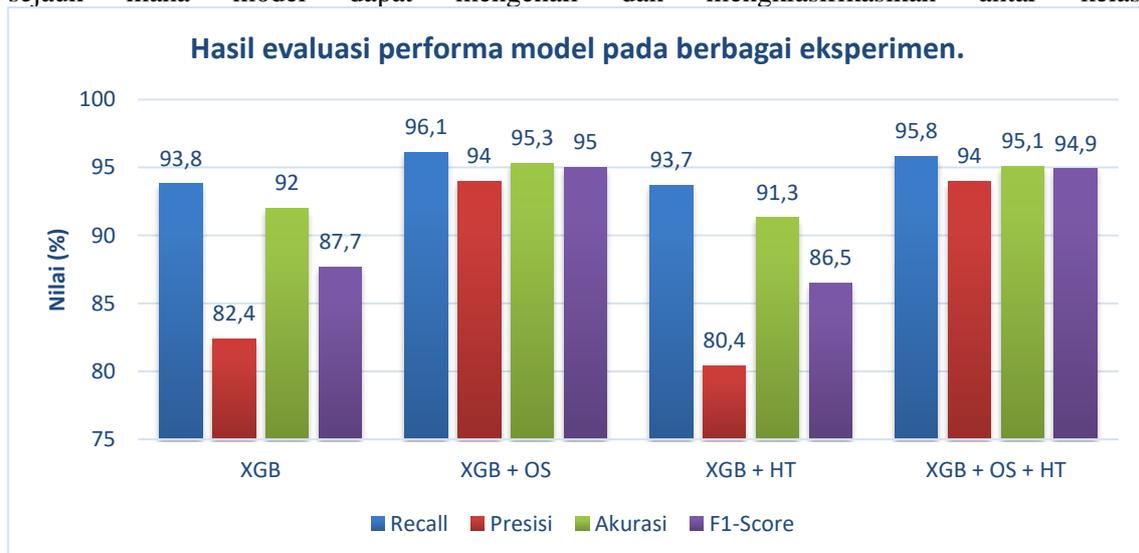
Dengan distribusi kelas yang seimbang pada data latih dan data uji, proses selanjutnya adalah *hyperparameter tuning*. Pada tahap *hyperparameter tuning*, diterapkan Grid Search untuk menemukan kombinasi nilai *hyperparameter* terbaik yang harapannya dapat meningkatkan akurasi model. Beberapa nilai *hyperparameter* dipilih dengan masing-masing nilai pada setiap hyperparameter yaitu $n_estimators = [50, 100, 150, 200, 250, 300, 350, 400, 450, 500]$, $max_depth = [3, 6, 9]$, dan $learning_rate = [0.1, 0.2, 0.3, 0.4, 0.5]$. Setelah serangkaian eksperimen, lima kombinasi nilai hyperparameter terbaik berhasil diidentifikasi berdasarkan peningkatan akurasi yang signifikan pada data latih.

param_n_estimators	param_max_depth	param_learning_rate	mean_test_score	rank_test_score	
59	500	9	0.2	0.949149	1
58	450	9	0.2	0.949149	1
57	400	9	0.2	0.949149	1
56	350	9	0.2	0.948698	4
88	450	9	0.3	0.948249	5

Gambar 5. Hasil pemeringkatan kombinasi hyperparameter terbaik

Hasil pemeringkatan ini menunjukkan kombinasi nilai *hyperparameter* optimal dan performa model yang lebih baik, yang dapat dilihat pada Gambar 5. Gambar tersebut memberikan gambaran jelas mengenai pengaruh setiap kombinasi hyperparameter terhadap kualitas model yang dibangun.

Setelah melihat pengaruh kombinasi nilai *hyperparameter* terhadap kualitas model, langkah selanjutnya adalah mengevaluasi performa model. Untuk mengevaluasi dampak dari setiap kombinasi teknik yang diterapkan pada model, penulis menggunakan berbagai metrik evaluasi seperti akurasi, presisi, *recall*, dan *F1-score* dalam berbagai konfigurasi eksperimen, baik secara terpisah maupun dengan menggabungkan teknik-teknik lain. Terdapat empat eksperimen pada penelitian ini, yaitu pemodelan dengan XGBoost (XGB); XGBoost dan *Oversampling* (XGB + OS); XGBoost dan *Hyperparameter Tuning* (XGB + HT); XGBoost, *Oversampling* dan *Hyperparameter Tuning* (XGB + OS + HT). Setiap metrik memberikan gambaran mengenai sejauh mana model dapat mengenali dan mengklasifikasikan antar kelas.



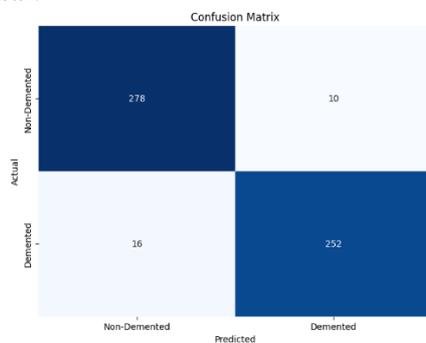
Gambar 6. Hasil evaluasi model pada berbagai eksperimen

Pada eksperimen pertama, dilakukan pemodelan dengan algoritma XGBoost. Proses ini bertujuan untuk memperoleh gambaran awal mengenai performa model dengan menggunakan parameter bawaan yang disediakan oleh XGBoost. Berdasarkan Gambar 6, hasil performa model XGBoost tanpa kombinasi tambahan memperoleh nilai *recall* 93,8%, presisi 82,4%, akurasi sebesar 92%, dan *F1-Score* sebesar 87,7%.

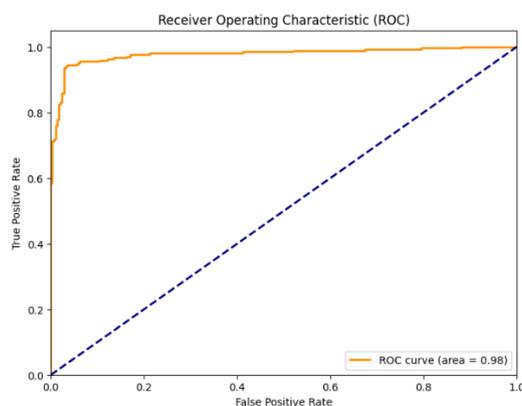
Eksperimen kedua, mengkombinasikan teknik *oversampling* dengan XGBoost untuk mengatasi masalah ketidakseimbangan kelas pada dataset. SMOTE digunakan untuk menambah jumlah data pada kelas minoritas, dengan tujuan untuk memperoleh distribusi kelas yang lebih seimbang. Model kemudian dilatih untuk mengevaluasi dampak penerapan *oversampling* pada performa model. Dari Gambar 6 menunjukkan bahwa dengan mengaplikasikan SMOTE, performa model meningkat dengan masing – masing nilai *recall* 96,1%, presisi 94%, akurasi 95,3%, dan *F1-Score* 95%.

Pada eksperimen ketiga, mengkombinasikan teknik *hyperparameter tuning* menggunakan Grid Search dengan XGBoost untuk mencari kombinasi nilai *hyperparameter* terbaik. Berdasarkan Gambar 6, model menunjukkan penurunan performa dengan nilai *recall* 93,7%, presisi 80,4%, akurasi 91,3%, dan *F1-Score* 86,5%. Penurunan ini menunjukkan bahwa konfigurasi *hyperparameter* yang diuji mungkin belum cocok dengan karakteristik data tidak seimbang yang digunakan.

Eksperimen keempat, mengkombinasikan teknik *oversampling* menggunakan SMOTE dengan *hyperparameter tuning* untuk melihat apakah gabungan kedua teknik ini dapat memberikan peningkatan yang lebih signifikan dalam kinerja model. Pada Gambar 6, hasil eksperimen keempat kembali menunjukkan adanya peningkatan dibandingkan dengan eksperimen ketiga, yaitu nilai *recall* 95,8%, presisi 94%, akurasi 95,1%, dan *F1-Score* 94,9% dengan penerapan kedua teknik tersebut. Namun, meskipun terjadi peningkatan, model yang hanya menggunakan teknik *oversampling* tanpa *hyperparameter tuning* tetap sedikit lebih unggul, dengan selisih 0,3% pada *recall*, 0,2% pada akurasi, dan 0,1% pada *F1-Score*. Hal ini menunjukkan bahwa performa kombinasi XGBoost dan *oversampling* sedikit lebih baik, namun perbedaannya tidak terlalu besar.



Gambar 7. Confusion matrix kombinasi XGBoost dan Oversampling



Gambar 8. Nilai AUC kombinasi XGBoost dan Oversampling

Hasil confusion matrix pada kombinasi XGBoost dan *oversampling* ditunjukkan pada Gambar 7, yang memperlihatkan jumlah prediksi benar yang tinggi untuk kedua kelas, terutama kelas minoritas yang sebelumnya sulit terdeteksi. Selain itu, Gambar 8 menunjukkan nilai *Area Under Curve* (AUC) sebesar 0.98, yang mengindikasikan performa model yang sangat baik dalam membedakan antara kelas positif dan negatif. Pendekatan ini terbukti efektif dalam meningkatkan kinerja model, menghasilkan metrik evaluasi yang lebih baik dibandingkan eksperimen lainnya.

4. KESIMPULAN DAN SARAN

Penelitian ini membuktikan bahwa penerapan teknik *oversampling*, khususnya SMOTE, secara efektif meningkatkan kinerja model XGBoost dalam memprediksi Alzheimer, sehingga dapat mengatasi masalah ketidakseimbangan data yang sering menjadi hambatan dalam data medis. Hasil utama menunjukkan bahwa eksperimen kedua (XGBoost dengan *Oversampling*) memberikan hasil terbaik, sedikit lebih unggul dibandingkan eksperimen keempat (XGBoost dengan *Oversampling* dan *Hyperparameter Tuning*). Sebaliknya, eksperimen pertama dan ketiga yang tidak menggunakan *oversampling* memperlihatkan performa yang lebih rendah. Temuan ini menegaskan bahwa ketidakseimbangan data mempengaruhi kinerja model prediksi secara signifikan. Dalam proses *hyperparameter tuning*, penting untuk memastikan bahwa data memiliki distribusi kelas yang seimbang agar hasil *hyperparameter tuning* lebih optimal. Hal ini terlihat dari eksperimen keempat yang memberikan hasil lebih baik dibandingkan eksperimen ketiga yang tidak menggunakan *oversampling*. Pengembangan penelitian ke depan dapat dilakukan dengan mengeksplorasi teknik *hyperparameter tuning* lain seperti bayesian optimization, sehingga kombinasi antara *oversampling* dan *hyperparameter tuning* dapat menghasilkan performa model yang lebih unggul dibandingkan keempat eksperimen yang telah dilakukan.

DAFTAR PUSTAKA

- [1] Z. Breijyeh and R. Karaman, "Comprehensive Review on Alzheimer's Disease: Causes and Treatment," *Molecules*, vol. 25, no. 24, 2020, doi: 10.3390/MOLECULES25245789.
- [2] P. Scheltens *et al.*, "Alzheimer's disease," *Lancet*, vol. 397, no. 10284, pp. 1577–1590, 2021, doi: 10.1016/S0140-6736(20)32205-4.
- [3] G. Livingston *et al.*, "Dementia prevention, intervention, and care: 2020 report of the Lancet Commission," *Lancet*, vol. 396, no. 10248, pp. 413–446, 2020, doi: 10.1016/S0140-6736(20)30367-6.
- [4] C. Kavitha, V. Mani, S. R. Srividhya, O. I. Khalaf, and C. A. Tavera Romero, "Early-Stage Alzheimer's Disease Prediction Using Machine Learning Models," *Front. Public Heal.*, vol. 10, no. March, pp. 1–13, 2022, doi: 10.3389/fpubh.2022.853294.
- [5] X. Y. Liew, N. Hameed, and J. Clos, "An investigation of XGBoost-based algorithm for breast cancer classification," *Mach. Learn. with Appl.*, vol. 6, no. August, p. 100154, 2021, doi: 10.1016/j.mlwa.2021.100154.
- [6] N. Kiran *et al.*, "Digital Pathology: Transforming Diagnosis in the Digital Age," *Cureus*, vol. 15, no. 9, 2023, doi: 10.7759/cureus.44620.
- [7] Z. Ahmad, S. Rahim, M. Zubair, and J. Abdul-ghafar, "Artificial intelligence (AI) in medicine , current applications and future role with special emphasis on its potential and promise in pathology : present and future impact , obstacles including costs and acceptance among pathologists , practical and phi," pp. 1–16, 2021, doi: <https://doi.org/10.1186/s13000-021-01085-4>.
- [8] O. A. Alimi, K. Ouahada, A. M. Abu-Mahfouz, S. Rimer, and K. O. A. Alimi, "A review of research works on supervised learning algorithms for scada intrusion detection and classification," *Sustain.*, vol. 13, no. 17, pp. 1–19, 2021, doi: 10.3390/su13179597.
- [9] J. Wang and F. Biljecki, "Unsupervised machine learning in urban studies: A systematic review of applications," *Cities*, vol. 129, no. December 2021, 2022, doi:

- 10.1016/j.cities.2022.103925.
- [10] S. S. Kuna, "Reinforcement Management Learning for Optimizing Insurance Portfolio," *African J. Artif. Intell. Sustain. Dev.*, vol. 2, no. 2, pp. 289–333, 2022, [Online]. Available: <https://africansciencegroup.com/index.php/AJAISD/article/view/152>
- [11] J. E. Black, J. K. Kueper, and T. S. Williamson, "An introduction to machine learning for classification and prediction," *Fam. Pract.*, vol. 40, no. 1, pp. 200–204, 2023, doi: 10.1093/fampra/cmac104.
- [12] X. Dairu and Z. Shilong, "Machine Learning Model for Sales Forecasting by Using XGBoost," *2021 IEEE Int. Conf. Consum. Electron. Comput. Eng. ICCECE 2021*, no. Icece, pp. 480–483, 2021, doi: 10.1109/ICCECE51280.2021.9342304.
- [13] B. S. G, Siji George, "Grid Search Tuning of Hyperparameters in Random Forest Classifier for Customer Feedback Sentiment Prediction," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 9, pp. 173–178, 2020, doi: 10.14569/IJACSA.2020.0110920.
- [14] M. Khushi *et al.*, "A Comparative Performance Analysis of Data Resampling Methods on Imbalance Medical Data," *IEEE Access*, vol. 9, pp. 109960–109975, 2021, doi: 10.1109/ACCESS.2021.3102399.
- [15] 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. (Ny)*, vol. 572, pp. 574–589, 2021, doi: 10.1016/j.ins.2021.02.056.
- [16] L. Hussain, K. J. Lone, I. A. Awan, A. A. Abbasi, and J. Pirzada, "Detecting congestive heart failure by extracting multimodal features with synthetic minority oversampling technique (SMOTE) for imbalanced data using robust machine learning techniques," *Waves in Random and Complex Media*, vol. 32, no. 3, pp. 1079–1102, May 2022, doi: 10.1080/17455030.2020.1810364.
- [17] A. A. Mortara, M. Permatasari, A. Desiani, Y. Andriani, and M. Arhami, "Perbandingan Algoritma C4.5 dan Adaptive Boosting dalam Klasifikasi Penyakit Alzheimer," *J. Teknol. dan Inf.*, vol. 13, no. 2, pp. 196–207, 2023, doi: 10.34010/jati.v13i2.10525.
- [18] R. J. Alfirdausy and S. Bahri, "Implementasi Algoritma K-Nearest Neighbor untuk Klasifikasi Diagnosis Penyakit Alzheimer," *Techno.Com*, vol. 22, no. 3, pp. 635–642, Aug. 2023, doi: 10.33633/tc.v22i3.8393.
- [19] A. Hebishy, "alzheimers_disease_data," 2024, *Kaggle*. doi: 10.34740/kaggle/dsv/8738477.
- [20] T. Milo and A. Somech, "Automating Exploratory Data Analysis via Machine Learning: An Overview," in *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, New York, NY, USA: ACM, Jun. 2020, pp. 2617–2622. doi: 10.1145/3318464.3383126.
- [21] N. Gupta *et al.*, "Data Quality Toolkit: Automatic assessment of data quality and remediation for machine learning datasets," Aug. 2021, [Online]. Available: <https://doi.org/10.48550/arXiv.2108.05935>
- [22] N. Pudjihartono, T. Fadason, A. W. Kempa-Liehr, and J. M. O'Sullivan, "A Review of Feature Selection Methods for Machine Learning-Based Disease Risk Prediction," *Front. Bioinforma.*, vol. 2, no. June, pp. 1–17, Jun. 2022, doi: 10.3389/fbinf.2022.927312.
- [23] M. Anshori, N. Rikatsih, and M. S. Haris, "PREDIKSI PASIEN DENGAN PENYAKIT KARDIOVASKULAR MENGGUNAKAN RANDOM FOREST," *TEKTRIKA - J. Penelit. dan Pengemb. Telekomun. Kendali, Komputer, Elektr. dan Elektron.*, vol. 7, no. 2, p. 58, Apr. 2023, doi: 10.25124/tektrika.v7i2.5279.
- [24] A. Ambarwari, Q. J. Adrian, and Y. Herdiyeni, "Analisis Pengaruh Data Scaling Terhadap Performa Algoritme Machine Learning untuk Identifikasi Tanaman," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 4, no. 1, pp. 117–122, 2020, [Online]. Available: <https://doi.org/10.29207/resti.v4i1.1517>
- [25] M. M. Ahsan, M. A. P. Mahmud, P. K. Saha, K. D. Gupta, and Z. Siddique, "Effect of Data Scaling Methods on Machine Learning Algorithms and Model Performance," *Technologies*, vol. 9, no. 3, pp. 5–9, 2021, doi: 10.3390/technologies9030052.

- [26] V. N. G. Raju, K. P. Lakshmi, V. M. Jain, A. Kalidindi, and V. Padma, "Study the Influence of Normalization/Transformation process on the Accuracy of Supervised Classification," in *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*, IEEE, Aug. 2020, pp. 729–735. doi: 10.1109/ICSSIT48917.2020.9214160.
- [27] J. Chen, H. Huang, A. G. Cohn, D. Zhang, and M. Zhou, "Machine learning-based classification of rock discontinuity trace: SMOTE oversampling integrated with GBT ensemble learning," *Int. J. Min. Sci. Technol.*, vol. 32, no. 2, pp. 309–322, 2022, doi: 10.1016/j.ijmst.2021.08.004.
- [28] V. R. Joseph and A. Vakayil, "SPlit: An Optimal Method for Data Splitting," *Technometrics*, vol. 64, no. 2, pp. 166–176, 2022, doi: 10.1080/00401706.2021.1921037.
- [29] W. Nugraha and A. Sasongko, "Hyperparameter Tuning on Classification Algorithm with Grid Search," *SISTEMASI*, vol. 11, no. 2, p. 391, May 2022, doi: 10.32520/stmsi.v11i2.1750.
- [30] H. Dong, D. He, and F. Wang, "SMOTE-XGBoost using Tree Parzen Estimator optimization for copper flotation method classification," *Powder Technol.*, vol. 375, pp. 174–181, Sep. 2020, doi: 10.1016/j.powtec.2020.07.065.
- [31] M. Muntasir Nishat *et al.*, "A Comprehensive Investigation of the Performances of Different Machine Learning Classifiers with SMOTE-ENN Oversampling Technique and Hyperparameter Optimization for Imbalanced Heart Failure Dataset," *Sci. Program.*, vol. 2022, no. Cvd, pp. 1–17, Mar. 2022, doi: 10.1155/2022/3649406.
- [32] S. A. Hicks *et al.*, "On evaluation metrics for medical applications of artificial intelligence," *Sci. Rep.*, vol. 12, no. 1, p. 5979, Apr. 2022, doi: 10.1038/s41598-022-09954-8.