



SNESTIK

Seminar Nasional Teknik Elektro, Sistem Informasi,
dan Teknik Informatika

<https://ejurnal.itats.ac.id/snestik> dan <https://snestik.itats.ac.id>



Informasi Pelaksanaan :

SNESTIK V - Surabaya, 26 April 2025

Fakultas Teknik Elektro dan Teknologi Informasi, Institut Teknologi Adhi Tama Surabaya

Informasi Artikel:

DOI : 10.31284/p.snestik.2025.7380

Prosiding ISSN 2775-5126

Fakultas Teknik Elektro dan Teknologi Informasi-Institut Teknologi Adhi Tama Surabaya
Gedung A-ITATS, Jl. Arief Rachman Hakim 100 Surabaya 60117 Telp. (031) 5945043
Email : snestik@itats.ac.id

Advanced Sleep Disorder Classification: An ML-Based Study with Optuna for Model Optimization

Amalan Fadil Gaib¹, I Gusti Ngurah Bagus Ferry Mahayudha², Anggito Karta Wijaya³,
Nurul Andini⁴, Ando Zamhariro Royan⁵

State University of Gorontalo¹, Udayana University², University of Jember³, State
University of Jakarta⁴, University Of Jember⁵

e-mail: amalanfadil@gmail.com¹

ABSTRACT

Hyperparameter optimization plays a crucial role in improving the performance of machine learning models, particularly in sleep disorder classification. However, searching for optimal hyperparameters often requires extensive computational resources and prolonged execution time. To address this issue, this study implements Optuna, a hyperparameter optimization framework based on the Tree-structured Parzen Estimator (TPE) and pruning mechanisms to enhance the efficiency of model configuration search adaptively. This study compares the performance of Support Vector Machine (SVM), Extreme Gradient Boosting (XGBoost), Random Forest, and Multi-Layer Perceptron (MLP) in classifying sleep disorders based on health and lifestyle variables. The data undergoes several preprocessing steps, including handling missing values, encoding, normalization (StandardScaler), and class balancing using SMOTE. The models are then developed and optimized using Optuna to determine the best hyperparameter configurations. Evaluation is conducted using Accuracy, Precision, Recall, and F1-score. Experimental results show that before optimization, the Random Forest model achieved an accuracy of 94%, XGBoost 96%, SVM 93%, and MLP 96%. After being optimized with Optuna, accuracy increased to 97% for Random Forest, 97% for XGBoost, 98% for SVM, and 97% for MLP. This improvement indicates that Optuna effectively enhances model performance, especially for SVM, which experienced the most significant accuracy boost after optimization. Thus, the use of Optuna not only accelerates hyperparameter tuning but also improves the efficiency and accuracy of machine learning models in sleep disorder classification. This approach has great potential in supporting AI-based medical diagnosis systems, enabling faster and more accurate detection of sleep disorders.

Keywords: *Optuna, sleep disorder, SVM, Random Forest, XGBoost, MLP*

ABSTRAK

Optimasi hyperparameter berperan penting dalam meningkatkan kinerja model pembelajaran mesin, terutama dalam klasifikasi gangguan tidur. Namun, pencarian hyperparameter optimal sering membutuhkan sumber daya komputasi besar dan waktu eksekusi yang lama. Untuk mengatasi hal ini, penelitian ini menerapkan Optuna, framework optimasi berbasis Tree-structured Parzen Estimator (TPE) dan pruning untuk meningkatkan efisiensi pencarian konfigurasi model. Penelitian ini membandingkan performa Support Vector Machine (SVM), Extreme Gradient Boosting (XGBoost), Random Forest, dan Multi-Layer Perceptron (MLP) dalam klasifikasi gangguan tidur berbasis variabel kesehatan dan gaya hidup. Data diproses melalui penanganan nilai kosong, encoding, normalisasi (StandardScaler), dan penyeimbangan kelas menggunakan SMOTE. Model kemudian dikembangkan dan dioptimalkan menggunakan Optuna untuk menemukan kombinasi hyperparameter terbaik. Evaluasi dilakukan menggunakan Akurasi, Presisi, Recall, dan F1-score. Hasil eksperimen menunjukkan bahwa sebelum optimasi, model Random Forest memiliki akurasi 94%, XGBoost 96%, SVM 93%, dan MLP 96%. Setelah dioptimasi dengan Optuna, akurasi meningkat menjadi 97% untuk Random Forest, 97% untuk XGBoost, 98% untuk SVM, dan 97% untuk MLP. Peningkatan ini menunjukkan bahwa Optuna secara efektif meningkatkan performa model, terutama pada SVM, yang mengalami lonjakan akurasi paling signifikan setelah optimasi. Dengan demikian, penggunaan Optuna tidak hanya mempercepat proses tuning hyperparameter tetapi juga meningkatkan efisiensi dan akurasi model dalam klasifikasi gangguan tidur. Pendekatan ini memiliki potensi besar dalam mendukung sistem diagnosis medis berbasis kecerdasan buatan, memungkinkan deteksi gangguan tidur yang lebih cepat dan akurat.

Kata kunci: Optuna, gangguan tidur, SVM, Random Forest, XGBoost, MLP.

PENDAHULUAN

Gangguan tidur telah menjadi masalah kesehatan global yang berdampak signifikan pada kualitas hidup jutaan orang. Sekitar 31% populasi mengalami gangguan tidur dengan 30% diantaranya mengalami obstructive sleep apnea (OSA), 28% insomnia, dan excessive daytime sleepiness (EDS). Insomnia, yang paling umum, ditandai dengan kesulitan memulai atau mempertahankan tidur (Roth, 2007) [1]. Sleep apnea sering tidak terdiagnosis dengan baik dan dapat meningkatkan risiko penyakit kardiovaskular (Peppard et al., 2013) [2]. Narcolepsy, yang menyebabkan kantuk berlebihan di siang hari, juga mempengaruhi produktivitas dan kualitas hidup (Dauvilliers et al., 2007) [3]. Gangguan tidur berdampak tidak hanya pada kesehatan fisik, tetapi juga kesehatan mental, produktivitas, dan interaksi sosial. Individu dengan insomnia sering mengalami gangguan kognitif, atensi, dan memori (Fortier-Brochu, 2014) [4]. Selain itu, mereka cenderung mengalami kelelahan dan penurunan produktivitas, yang meningkatkan risiko kecelakaan kerja serta menambah beban ekonomi. Gangguan tidur kronis juga meningkatkan risiko penyakit serius, seperti hipertensi, diabetes, dan penyakit jantung, serta gangguan emosi seperti depresi dan kecemasan. Konsekuensi sosial dan ekonomi gangguan tidur juga signifikan. Kesulitan berkonsentrasi dan penurunan fungsi kognitif akibat kurang tidur dapat mengganggu aktivitas sehari-hari dan kinerja kerja. Studi oleh Rand Corporation menunjukkan bahwa kurang tidur menyebabkan penurunan produktivitas hingga miliaran dolar setiap tahunnya, akibat absensi kerja dan pengambilan keputusan yang buruk (Hafner et al., 2017) [5]. Sebagai respons terhadap masalah ini, berbagai lembaga kesehatan dan pemerintah, termasuk Kementerian Kesehatan Republik Indonesia (2017) telah mengeluarkan pedoman penanganan gangguan tidur sebagai bagian dari strategi nasional untuk meningkatkan kesehatan Masyarakat [6].

Perkembangan kecerdasan buatan (AI) dan pembelajaran mesin (ML) dalam beberapa tahun terakhir telah membuka peluang baru dalam diagnosis dan klasifikasi gangguan tidur. Metode pembelajaran mesin meningkatkan akurasi diagnosis melalui analisis data elektrofisiologis, pola tidur, serta faktor lingkungan dan biologis. Penelitian oleh Huang et al.

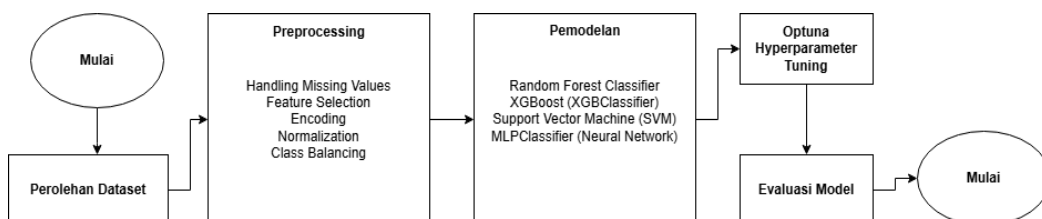
(2019) menunjukkan bahwa deep learning dapat menghasilkan klasifikasi lebih akurat dibandingkan metode konvensional [7]. Biswal et al. (2018) juga menemukan bahwa teknik seperti Random Forest dan Support Vector Machine (SVM) efektif dalam mengidentifikasi gangguan tidur dengan akurasi tinggi [8].

Penelitian ini mengimplementasikan algoritma pembelajaran mesin seperti SVM, Extreme Gradient Boosting (XGBoost), Random Forest, dan Artificial Neural Network (ANN). SVM efektif dalam menangani data berdimensi tinggi dan menemukan batas keputusan optimal (Cortes & Vapnik, 1995) [9]. XGBoost unggul dalam mengelola data kompleks dengan teknik boosting yang meningkatkan akurasi prediksi (Chen & Guestrin, 2016) [10]. Random Forest menawarkan stabilitas dengan mengurangi risiko overfitting melalui penggabungan beberapa pohon keputusan (Breiman, 2001) [11]. Sementara itu, ANN mampu menangkap pola non-linear dalam data fisiologis, menjadikannya alat yang ampuh dalam analisis data kompleks (LeCun et al., 2015) [12]. Seiring dengan perkembangan teknologi machine learning dalam diagnosis gangguan tidur, berbagai penelitian telah mengevaluasi keunggulan algoritma yang berbeda dalam mengklasifikasikan kondisi tidur. Penelitian oleh Chen et al. (2020) menunjukkan bahwa penggunaan SVM dalam klasifikasi tahap tidur berdasarkan sinyal EEG mencapai akurasi sekitar 92%, yang diperoleh melalui pemilihan kernel yang optimal dan kemampuannya dalam mengelola data berdimensi tinggi [13]. Selain itu, dalam konteks Extreme Gradient Boosting (XGBoost) studi oleh Liu et al. (2020) mengungkapkan bahwa algoritma ini mampu mendeteksi sleep apnea dengan akurasi berkisar antara 91 hingga 94% berkat penerapan metode boosting iteratif dan fitur regularisasi yang efektif dalam menangani kompleksitas data, seperti variabilitas denyut jantung [14]. Penelitian oleh Rahman et al. (2021) menunjukkan bahwa Random Forest memberikan hasil yang stabil dengan akurasi antara 90 hingga 92% dalam pengklasifikasian gangguan tidur karena kemampuannya mengurangi risiko overfitting melalui kombinasi beberapa pohon keputusan [15]. Di sisi lain, penelitian oleh Supratak et al. (2017) dengan model DeepSleepNet menunjukkan bahwa pendekatan ANN dapat mencapai akurasi hingga 94% dalam pengklasifikasian tahap tidur berkat kemampuannya menangkap pola non-linear yang kompleks pada data EEG [16]. Penelitian oleh Akiba et al. (2019) tentang Optuna menunjukkan keunggulan signifikan dalam optimasi hyperparameter untuk model machine learning, berkat pendekatan define-by-run yang memberikan fleksibilitas tinggi dalam mendefinisikan ruang pencarian hyperparameter secara dinamis [17]. Framework ini mengimplementasikan algoritma Tree-structured Parzen Estimator (TPE) untuk proses sampling dan pruning, yang mempercepat optimasi dengan mengeliminasi kombinasi hyperparameter yang kurang potensial sejak dini. Selain itu, Optuna mendukung komputasi terdistribusi, memungkinkan eksperimen berskala besar maupun ringan, serta kompatibel dengan berbagai framework machine learning seperti Scikit-Learn, Keras, PyTorch, TensorFlow, XGBoost, dan LightGBM. Dengan kemampuan tersebut, Optuna secara efektif meningkatkan akurasi dan keandalan model machine learning, termasuk dalam klasifikasi gangguan tidur, sehingga mendukung pengambilan keputusan medis yang lebih cepat, akurat, dan efisien.

Secara keseluruhan, gangguan tidur telah menjadi masalah kesehatan global yang signifikan, yang mempengaruhi kualitas hidup jutaan orang. Penelitian ini berfokus pada penerapan algoritma machine learning, seperti SVM, XGBoost, Random Forest, dan ANN, untuk mengklasifikasikan gangguan tidur. Optuna digunakan sebagai framework optimasi hyperparameter untuk meningkatkan performa model. Matrik evaluasi yang digunakan mencakup akurasi, presisi, recall, dan F1-score. Hasil analisis akan disajikan melalui visualisasi grafik untuk memudahkan interpretasi. Penelitian ini diharapkan dapat mendukung tenaga medis dalam mendiagnosa gangguan tidur secara lebih akurat dan efisien, sehingga meningkatkan kualitas layanan kesehatan bagi pasien.

METODE PENELITIAN

Penelitian ini mengimplementasikan pendekatan pembelajaran mesin untuk mengklasifikasikan gangguan tidur berdasarkan variabel kesehatan dan gaya hidup. Pada Gambar 1, tahapan penelitian ini merupakan diagram alur dalam pengembangan model machine learning. Penelitian dimulai dengan perencanaan dan pengumpulan data dari sumber relevan. Tahap preprocessing mencakup handling missing values, feature selection, feature engineering, encoding, normalisasi, dan class balancing. Model machine learning yang dibangun meliputi Random Forest, XGBoost, SVM, dan ANN. Proses tuning hyperparameter menggunakan Optuna, dan evaluasi model dilakukan dengan akurasi, precision, recall, F1-score, atau AUC-ROC, diakhiri dengan analisis hasil dan kesimpulan.



Gambar 1. Tahapan Penelitian

1. Datasets

Pada tahap pengumpulan data, penelitian ini menggunakan dataset Sleep Health and Lifestyle dari Kaggle. Dataset ini mencakup 374 data dengan 13 kolom, yang terdiri dari berbagai variabel terkait kesehatan tidur dan gaya hidup, seperti durasi tidur, kualitas tidur, aktivitas fisik, pola makan, tingkat stres, dan faktor kesehatan lainnya. Dataset ini dipilih karena kelengkapan variabel yang mendukung penelitian dalam klasifikasi gangguan tidur berbasis machine learning, sehingga memberikan dasar yang kuat untuk analisis dan pengembangan model prediktif.

2. Tahap Processing Data

Pada tahap ini, dilakukan berbagai langkah pemrosesan awal terhadap data untuk meningkatkan kualitasnya sebelum digunakan dalam pemodelan. Preprocessing data sangat penting agar model yang digunakan dapat menghasilkan prediksi yang akurat dan menghindari bias akibat kualitas data yang buruk.

2.1 Handling Missing Values

Langkah pertama dalam preprocessing adalah menangani nilai yang hilang agar tidak menyebabkan bias atau mengurangi kualitas analisis. Metode yang digunakan meliputi pengisian dengan rata-rata, median, atau modus, serta interpolasi jika diperlukan. Jika jumlah nilai yang hilang terlalu banyak dan tidak dapat diperbaiki, data tersebut dapat dihapus dengan pertimbangan yang matang.

2.2 Feature Selection

Selanjutnya, dilakukan seleksi fitur untuk memilih variabel yang paling berpengaruh dalam analisis. Teknik yang digunakan mencakup analisis korelasi, uji statistik, dan metode berbasis machine learning guna menghilangkan fitur yang kurang relevan atau tidak memiliki kontribusi signifikan terhadap hasil penelitian.

2.3 Encoding

Agar data kategori dapat digunakan dalam pemodelan, dilakukan proses encoding untuk mengubah data non-numerik menjadi bentuk numerik. Teknik yang digunakan meliputi one-hot encoding untuk kategori nominal dan label encoding untuk kategori ordinal.

2.4 Normalisasi

Untuk memastikan bahwa setiap fitur memiliki skala yang seimbang, dilakukan normalisasi data. Metode yang umum digunakan mencakup Min-Max Scaling, yang merentangkan nilai dalam rentang tertentu, serta standarisasi dengan Z-score agar distribusi data lebih seimbang dan tidak ada variabel yang mendominasi dalam analisis.

2.5 Class Balancing

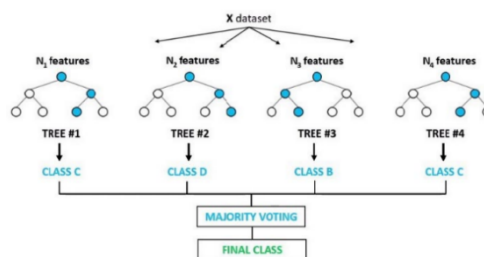
Distribusi kelas yang tidak seimbang dapat menyebabkan model cenderung memprediksi kelas mayoritas, sehingga perlu dilakukan penyeimbangan kelas. Teknik yang digunakan dalam penelitian ini adalah Synthetic Minority Over-sampling Technique (SMOTE), yang bekerja dengan mensintesis sampel baru berdasarkan sampel minoritas yang sudah ada. Dengan metode ini, model dapat belajar dari kedua kelas secara lebih seimbang, sehingga mengurangi risiko bias dalam pemodelan.

3. Pemodelan

Tahap ini, dilakukan pembangunan model pembelajaran mesin untuk melaksanakan proses klasifikasi. Proses pemodelan mencakup tiga aspek utama: pemilihan algoritma, pelatihan model, dan evaluasi kinerja. Dalam implementasinya, dipilih empat algoritma utama yaitu Random Forest Classifier, XGBoost, Support Vector Machine (SVM), dan MLPClassifier (Neural Network).

3.1 Random Forest Classifier

Random Forest Classifier adalah algoritma ensemble berbasis pohon keputusan yang membangun banyak pohon secara acak dan menggabungkan hasilnya untuk meningkatkan akurasi serta mengurangi risiko overfitting. Algoritma ini menggunakan teknik Bootstrap Aggregating (Bagging), dengan pemilihan fitur secara acak pada setiap pemisahan node menggunakan kriteria seperti Gini Impurity atau Information Gain. Pada Gambar 2 ditampilkan arsitektur Random Forest, di mana dataset dibagi menjadi beberapa subset secara acak, lalu setiap subset digunakan untuk melatih pohon keputusan secara independen. Hasil prediksi dari seluruh pohon digabungkan menggunakan metode voting mayoritas untuk klasifikasi atau rata-rata (averaging) untuk regresi. Pendekatan ini mampu meningkatkan stabilitas dan akurasi model serta mengurangi risiko overfitting.



Gambar 2. Random Forest Classifier

Perhitungan entropy untuk menentukan tingkat polusi atribut dan informasi yang diperoleh dimulai dengan membangun pohon keputusan. Entropy sendiri mengukur tingkat ketidakmurnian data, yaitu sejauh mana distribusi kelas dalam suatu atribut tidak homogen atau beragam (Adi & Wintarti, 2022). Berikut adalah rumus yang diterapkan untuk menghitung nilai entropy:

$$\hat{y} = \arg \left(\sum_{i=1}^n I(h_i(x) = k) \right) \quad (1)$$

Keterangan dari rumus Random Forest yaitu :

$\hat{y}$ = Hasil prediksi akhir

K = Kelas target (label) yang mungkin.

n = Jumlah pohon dalam hutan.

$h_i(x)$ = Prediksi dari pohon ke- i untuk input x .

I = Fungsi indikator yang bernilai 1 jika kondisi terpenuhi dan 0 jika tidak.

3.2 XGBoost

XGBoost (Extreme Gradient Boosting) adalah algoritma pembelajaran mesin berbasis teknik boosting yang dikembangkan untuk meningkatkan performa prediksi model dengan cara menggabungkan beberapa pohon keputusan (decision trees) yang lebih lemah menjadi model yang lebih kuat. Teknik ini bekerja dengan memperbaiki model secara iteratif, di mana setiap model yang baru dibangun berfokus pada kesalahan yang dibuat oleh model sebelumnya. XGBoost terkenal karena kemampuannya dalam menangani dataset yang besar dan kompleks serta memiliki performa yang tinggi dalam berbagai kompetisi analisis data.

$$L(\theta) = \sum_{i=1}^n l(y_i, \hat{y}_i) + \sum_{k=1}^K \Omega(f_k) \quad (2)$$

Keterangan dari rumus XGBoost yaitu :

$L(\theta)$ = adalah fungsi objektif total yang ingin diminimalkan.

$l(y_i, \hat{y}_i)$ = adalah fungsi kehilangan (loss function) yang mengukur perbedaan antara nilai yang diprediksi (adalah fungsi kehilangan (loss function) yang mengukur perbedaan antara nilai yang diprediksi ($\hat{y}_i$) dan nilai aktual (y_i))

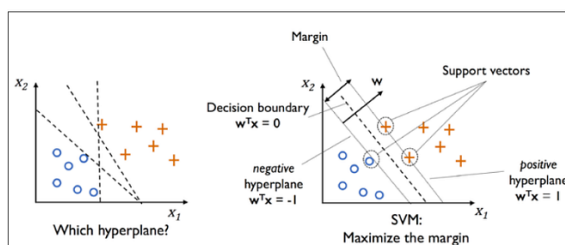
$\Omega(f_k)$ = adalah fungsi regularisasi yang mengontrol kompleksitas pohon keputusan ke- k untuk menghindari overfitting.

n = adalah jumlah sampel data.

K = adalah jumlah pohon yang dibangun selama proses boosting.

3.3 Support Vector Machine (SVM)

Support Vector Machine (SVM) adalah algoritma pembelajaran mesin untuk klasifikasi dan regresi yang mencari hyperplane optimal guna memisahkan dua kelas data dengan margin maksimal. Hyperplane ini berfungsi sebagai batas keputusan, sementara support vectors adalah titik data terdekat dari masing-masing kelas. SVM efektif untuk dataset kecil maupun besar serta bekerja baik pada data berdimensi tinggi (Abdusyukur, 2023). Algoritma ini diperkenalkan oleh Vladimir Vapnik dan Alexey Chervonenkis pada 1963, kemudian dikembangkan pada 1990-an dengan kernel trick untuk menangani data yang tidak terpisahkan secara linear. Pada gambar 3, arsitektur SVM memperlihatkan beberapa kemungkinan hyperplane di sebelah kiri, tetapi tidak semuanya memiliki margin optimal. Di sebelah kanan, ditunjukkan konsep utama SVM, yaitu hyperplane yang memaksimalkan margin antara dua kelas. aris putus-putus menunjukkan hyperplane positif dan negatif dengan persamaan $w^T x = 1$ dan $1w^T x = -1$, sementara garis tengah $w^T x = 0$ adalah decision boundary. Titik terdekat dengan hyperplane disebut support vectors dan berperan menentukan margin optimal.



Gambar 3 Arsitektur Support Vektor Machine (SVM)

Rumus Svm:

$$(w \cdot xi) + b = 0 \quad (3)$$

Di mana:

1. w adalah vektor bobot yang menentukan orientasi hyperplane.
2. xi adalah vektor fitur dari data ke- i .
3. b adalah bias, yang menentukan pergeseran hyperplane dari titik asal.
4. 0 menunjukkan bahwa hyperplane berada di tengah antara dua kelas.
- 5.

Data yang termasuk dalam kelas (-1) dapat dirumuskan dalam persamaan berikut:

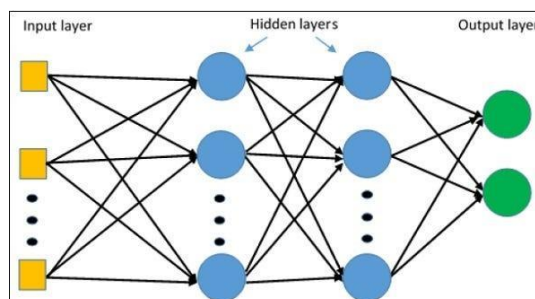
$$(w \cdot xi + b) \leq -1 \quad (4)$$

Sedangkan data yang termasuk dalam kelas (+1) dapat dirumuskan dalam persamaan berikut:

$$(w \cdot xi + b) \geq 1 \quad (5)$$

3.4 MLPClassifier

MLPClassifier (Neural Network) adalah jaringan saraf tiruan yang menggunakan backpropagation untuk mengenali pola non-linear dalam data. Algoritma ini memungkinkan penyesuaian jumlah hidden layers, neuron, dan fungsi aktivasi, tetapi memerlukan waktu komputasi lebih lama dan sensitif terhadap skala data, sehingga normalisasi penting sebelum pelatihan. Model ini diimplementasikan menggunakan scikit-learn dan XGBoost dengan optimasi parameter untuk klasifikasi penyakit tiroid. Pada gambar 4, arsitektur Multi-Layer Perceptron terdiri dari input layer yang menerima data, hidden layers yang memproses pola dengan bobot dan fungsi aktivasi, serta output layer yang menghasilkan prediksi akhir.



Gambar 4 Arsitektur MultiLayer Perceptron (Neural Network)

4. Optuna Hyperparameter Tuning

Pada tahap ini, dilakukan penyesuaian hyperparameter model menggunakan Optuna, pustaka berbasis Python yang memanfaatkan teknik pencarian efisien seperti Tree-structured Parzen Estimator (TPE) untuk menemukan kombinasi parameter terbaik secara otomatis. Optuna bekerja dengan mengeksplorasi berbagai kombinasi hyperparameter secara adaptif menggunakan

mekanisme berbasis bayesian optimization untuk memandu pencarian ke arah parameter optimal secara lebih efektif. Proses tuning dilakukan dengan mendefinisikan ruang pencarian parameter, menjalankan simulasi percobaan, dan mengevaluasi hasilnya hingga diperoleh konfigurasi optimal. Keunggulan Optuna adalah kemampuannya dalam menyesuaikan strategi pencarian berdasarkan hasil sebelumnya dan mendukung paralelisasi untuk mempercepat proses pencarian parameter terbaik.

5. Confusion Matrix

Confusion Matrix adalah matriks yang digunakan untuk mengevaluasi kinerja model klasifikasi dengan membandingkan hasil prediksi model terhadap nilai aktual dari dataset. Matriks ini memberikan gambaran rinci tentang bagaimana model mengklasifikasikan setiap kelas dalam dataset, termasuk kesalahan yang terjadi.

Dari *confusion matrix*, ada beberapa metrik kinerja model yang dapat dihitung seperti akurasi, *recall* (sensitivitas), presisi, dan *F1 score* yang umumnya digunakan untuk mengukur performa model. Untuk model klasifikasi biner, confusion matrix memiliki struktur seperti pada Tabel 1 berikut.

Tabel 1. Struktur *Confusion Matrix*

Kelas Prediksi	Kelas Aktual	
	Label Positif	Label Negatif
Prediksi Positif	<i>True Positive</i> (TP)	<i>False Positive</i> (FP)
Prediksi Negatif	<i>False Negative</i> (FN)	<i>True Negative</i> (TN)

Keterangan :

- TN : Data negatif dan diprediksi benar sebagai negatif
- TP : Data negatif dan diprediksi benar sebagai positif
- FP : Data negatif dan diprediksi salah sebagai positif
- FN : Data positif dan diprediksi salah sebagai negative

5.1 Interpretasi Confusion Matrix

Dari confusion matrix, kita dapat menghitung beberapa metrik evaluasi utama yang digunakan dalam penelitian ini, yaitu:

1. Accuracy (Akurasi)

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \quad (6)$$

2. Precision (Presisi)

$$Presisi = \frac{TP}{TP+FP} \quad (7)$$

3. Recall (Sensitivitas / True Positive Rate)

$$Recall = \frac{TP}{TP+FN} \quad (8)$$

4. F1-score

$$F1\ Score = 2 \times \frac{Recall \times Precision}{Recall+Precision} \quad (9)$$

HASIL DAN PEMBAHASAN

Penelitian ini dilakukan dengan menerapkan berbagai metode normalisasi fitur, pengkodean kategorikal, dan teknik penyeimbangan data untuk mengidentifikasi kombinasi preprocessing yang menghasilkan akurasi terbaik dalam klasifikasi gangguan tidur. Normalisasi fitur dilakukan menggunakan StandardScaler, MinMaxScaler, dan RobustScaler untuk

memastikan distribusi data yang lebih seimbang dan meningkatkan efektivitas algoritma pembelajaran mesin. Selain itu, pengkodean data kategorikal diuji dengan OneHotEncoder dan OrdinalEncoder guna memahami dampaknya terhadap performa model. Untuk mengatasi ketidakseimbangan kelas dalam dataset, diterapkan teknik SMOTE (Synthetic Minority Over-sampling Technique), yang bertujuan untuk meningkatkan representasi kelas minoritas. Melalui kombinasi berbagai metode ini, penelitian bertujuan untuk menentukan konfigurasi preprocessing terbaik yang dapat meningkatkan akurasi serta matrik evaluasi lainnya dalam membangun model klasifikasi yang lebih optimal.

1. Evaluasi Model Random Forest

Model Random Forest diuji menggunakan berbagai kombinasi teknik scaling, encoding, dan sampling untuk menentukan konfigurasi preprocessing yang optimal. Berdasarkan Tabel 2, performa model Random Forest tanpa tambahan optimasi Optuna menghasilkan akurasi 94%, recall 94%, presisi 94%, dan F1-score 94%. Sedangkan model Random Forest dengan tambahan optimasi Optuna menghasilkan akurasi 97%, recall 97%, presisi 97%, dan F1-score 97%. Hasil ini menunjukkan bahwa penggunaan Optuna sebagai alat optimasi hiperparameter memberikan peningkatan kinerja pada model Random Forest dalam penelitian ini, meskipun peningkatannya tidak terlalu signifikan. Hal ini dapat mengindikasikan bahwa konfigurasi hiperparameter awal sudah cukup optimal, namun tuning lebih lanjut dengan Optuna masih mampu menemukan kombinasi parameter yang sedikit lebih baik.

Tabel 2. Perbandingan Model Random Forest dengan Optuna

Model	Accuracy	Recall	Precision	F1-Score
Random Forest	0.94	0.94	0.94	0.94
Random Forest+Optuna	0.97	0.97	0.97	0.97

2. Evaluasi Model XGBoost

XGBoost (Extreme Gradient Boosting) adalah algoritma pembelajaran mesin berbasis pohon keputusan yang menggabungkan teknik boosting untuk meningkatkan akurasi dan mengurangi overfitting. Pada Tabel 3, model XGBoost menunjukkan performa terbaik dalam klasifikasi gangguan tidur. Sebelum optimasi, model ini memiliki akurasi 96%, recall 96%, presisi 96%, dan F1-score 96%. Setelah optimasi dengan Optuna, performanya meningkat sedikit dengan semua metrik mencapai 97%. Model ini menggunakan preprocessing seperti StandardScaler untuk normalisasi, OrdinalEncoder untuk encoding variabel kategorikal, dan SMOTE untuk menangani ketidakseimbangan kelas. Hasil ini menunjukkan bahwa Optuna memberikan sedikit peningkatan performa, mengindikasikan bahwa konfigurasi awal XGBoost sudah cukup optimal.

Tabel 3. Perbandingan Model XGBoost dengan Optuna

Model	Accuracy	Recall	Precision	F1-Score
XGBoost	0.96	0.96	0.96	0.96
XGBoost+Optuna	0.97	0.97	0.97	0.97

3. Evaluasi Model Support Vector Machine (SVM)

Support Vector Machine (SVM) adalah algoritma pembelajaran mesin yang mencari hyperplane optimal untuk memisahkan kelas dalam data, sehingga efektif dalam tugas klasifikasi. Pada Tabel 4, model SVM menunjukkan peningkatan performa yang signifikan setelah penerapan Optuna Hyperparameter Tuning. Sebelum optimasi, model SVM memiliki akurasi 93%, recall 93%, presisi 93%, dan F1-score 93%. Setelah optimasi dengan Optuna,

semua metrik meningkat drastis hingga 98%. Model ini menggunakan preprocessing seperti StandardScaler untuk normalisasi, OrdinalEncoder untuk encoding variabel kategorikal, dan SMOTE untuk menangani ketidakseimbangan kelas. Hasil ini menunjukkan bahwa tuning hiperparameter dengan Optuna mampu secara signifikan meningkatkan kinerja model SVM dalam penelitian ini.

Tabel 4. Perbandingan Model SVM dengan Optuna

Model	Accuracy	Recall	Precision	F1-Score
SVM	0.93	0.93	0.93	0.93
SVM+Optuna	0.98	0.98	0.98	0.98

4. Evaluasi Model MLP

Model Multi-Layer Perceptron (MLP) diuji menggunakan berbagai kombinasi teknik preprocessing untuk menentukan konfigurasi yang optimal. Berdasarkan Tabel 5, performa model MLP tanpa tambahan optimasi Optuna menghasilkan akurasi 96%, recall 96%, presisi 96%, dan F1-score 96%. Sedangkan model MLP dengan tambahan optimasi Optuna menghasilkan akurasi 97%, recall 97%, presisi 97%, dan F1-score 97%. Hasil ini menunjukkan bahwa penggunaan Optuna sebagai alat optimasi hiperparameter mampu meningkatkan kinerja model MLP secara signifikan dalam penelitian ini. Hal ini mengindikasikan bahwa tuning hiperparameter dengan Optuna dapat membantu menemukan konfigurasi parameter yang lebih baik untuk meningkatkan performa model.

Tabel 5. Perbandingan Model MLP dengan Optuna

Model	Accuracy	Recall	Precision	F1-Score
MLP	0.96	0.96	0.96	0.96
MLP +Optuna	0.97	0.97	0.97	0.97

KESIMPULAN

Penelitian Penelitian ini menunjukkan bahwa optimasi hiperparameter menggunakan Optuna dapat meningkatkan performa model dalam klasifikasi gangguan tidur. Berdasarkan hasil eksperimen, sebelum optimasi, model Random Forest memiliki akurasi 94%, XGBoost 96%, SVM 93%, dan MLP 96%. Setelah dioptimasi dengan Optuna, performa model meningkat menjadi 97% untuk Random Forest, 97% untuk XGBoost, 98% untuk SVM, dan 97% untuk MLP. Teknik preprocessing data, seperti StandardScaler untuk normalisasi, OrdinalEncoder untuk encoding variabel kategorikal, serta SMOTE untuk menangani ketidakseimbangan kelas, terbukti meningkatkan stabilitas dan generalisasi model. Selain itu, Optuna secara signifikan meningkatkan performa model, terutama pada SVM, yang mengalami peningkatan akurasi paling besar setelah optimasi. Hasil penelitian ini membuktikan bahwa Optuna tidak hanya mengurangi kompleksitas eksplorasi hiperparameter secara manual tetapi juga meningkatkan efisiensi dan akurasi model dalam klasifikasi gangguan tidur. Pendekatan ini memiliki potensi besar dalam mendukung sistem diagnosis medis berbasis kecerdasan buatan, memungkinkan deteksi gangguan tidur yang lebih cepat dan akurat.

DAFTAR PUSTAKA

- [1] Roth, T. (2007). Insomnia: Definition, Prevalence, Etiology, and Consequences. *Journal of Clinical Sleep Medicine*, 3(5), S7-S101..

- [2] Jean-Louis, G., Zizi, F., Clark, L.T., Brown, C.D., & McFarlane, S.I. (2008). Obstructive sleep apnea and cardiovascular disease: role of the metabolic syndrome and its components. *Journal of Clinical Sleep Medicine*, 4(3), 261–272.
- [3] Dauvilliers, Y., et al. (2021). Work productivity and activity impairment in patients with narcolepsy. *Journal of Sleep Research*, 30(6), e140873.
- [4] Fortier-Brochu, É., & Morin, C.M. (2014). Cognitive impairment in individuals with insomnia: clinical significance and correlates. *SLEEP*, 37(11), 1787-1794.
- [5] Farrell, S. (2021). The Detrimental Effects of Inadequate Sleep on the Economy. *Sacred Heart University*.
- [6] Kementerian Kesehatan Republik Indonesia. (2017). *Pedoman Penanganan Gangguan Tidur*. Jakarta: Kemenkes RI.
- [7] X. Huang, Z. Wang, and S. Z. Li, "Deep Learning for Sleep Apnea Detection Using Electrocardiogram," *IEEE Transactions on Biomedical Engineering*, vol. 66, no. 1, pp. 1-12, 2019.
- [8] A. Biswal, H. Nagarajan, and S. Chandran, "A Machine Learning Approach for Detecting Sleep Disorders: A Review," *International Journal of Computer Applications*, vol. 180, no. 10, pp. 44-48, 2018.
- [9] L. Cortes and V. Vapnik, "Support Vector Networks," *Machine Learning*, vol. 20, no. 3, pp. 273-297, 1995.
- [10] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 785-794.
- [11] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5-32, 2001.
- [12] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," *Nature*, vol. 521, no. 7553, pp. 436-444, 2015.
- [13] Y. Chen, C. K. Xie, and P. Zhou, "Support Vector Machine-Based Sleep Stage Classification Using EEG Signals," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 28, no. 2, pp. 299-308, 2020.
- [14] Y. Liu, W. Liu, and H. Li, "Detection of Sleep Apnea Using Extreme Gradient Boosting," *IEEE Access*, vol. 8, pp. 132122-132130, 2020.
- [15] M. Rahman, M. A. Khan, and M. N. Ahmed, "Using Random Forest for Sleep Disorder Classification," *Journal of Medical Systems*, vol. 45, no. 3, pp. 72-80, 2021.
- [16] R. Supratak, Y. Dong, and C. Wu, "DeepSleepNet: A Deep Learning Framework for Sleep Stage Scoring," *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 5, pp. 1077-1086, 2017.
- [17] K. Akiba, S. Sano, T. Yanase, and T. Ohta, "Optuna: A Next-generation Hyperparameter Optimization Framework," *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 2623-2631.
- [18] Chawla, N. V., K. W. Bowyer, L. Hall, dan W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321-357, 2002.

Halaman ini sengaja dikosongkan