

# Efektivitas Optimasi Hyperparameter Dalam Prediksi Pembakaran Kalori : Data Aktivitas Fisik

Ulfatun Nadifa<sup>1\*</sup>, Rahmat Deddy Rianto Dako<sup>1</sup>, Ade Irawaty Tolago<sup>2</sup>, Rahmat Hidayat Dongka<sup>2</sup>

<sup>1</sup>Teknik Komputer, Fakultas Teknik, Universitas Negeri Gorontalo

<sup>2</sup>Teknik Elektro, Fakultas Teknik, Universitas Negeri Gorontalo

Jl. Jendral Sudirman No. 6 Kota Gorontalo

E-mail: [ulfatun@ung.ac.id](mailto:ulfatun@ung.ac.id) (\*)

Naskah Masuk: 02 Juli 2025; Diterima: 27 Agustus 2025; Terbit: 31 Agustus 2025

## ABSTRAK

**Abstrak** - Prediksi pembakaran kalori berdasarkan data aktivitas fisik merupakan topik penting di bidang kesehatan dan olahraga. Fokus utama adalah kemampuan untuk mempersonalisasi program kebugaran dan memantau kondisi metabolisme secara lebih akurat. Namun, tantangan utama dalam pengembangan model prediksi terletak pada akurasi dan konsistensinya, mengingat keragaman jenis dan kompleksitas data. Penelitian ini mengembangkan pendekatan membangun model prediksi yang efektif dengan Optimasi Hyperparameter otomatis menggunakan Optuna. Langkah pertama adalah membuat Pipeline Preprocessing berbasis Tree Model untuk menangani data secara lebih baik. Selanjutnya, digunakan metode Stacking Ensemble dengan tiga model dasar berbasis Gradient Boosting: CatBoost, LightGBM, dan XGBoost. Model divalidasi dengan teknik K-Fold Cross Validation dan metrik Root Mean Squared Logarithmic Error (RMSLE). Hasilnya menunjukkan nilai RMSLE sebesar 0.00599 dan waktu prediksi 0.0091 detik. Selain itu, untuk menguji konsistensi model, diterapkan dua teknik Encoding fitur kategorikal, yaitu LabelEncoder dengan Standard Scaler dan OrdinalEncoder tanpa Standard Scaler. Perbedaan skala nilai yang dihasilkan dari Teknik ini menjadi alasan utama untuk melihat pengaruhnya terhadap waktu dan konsistensi prediksi. Ditemukan model tetap menunjukkan konsistensi yang baik dilihat dari RMSLE dan waktu prediksi tidak jauh berbeda, ini mengindikasikan bahwa kombinasi Pipeline yang tepat, Model Ensemble, dan Optimasi Hyperparameter dapat menghasilkan sistem prediksi yang akurat dan stabil.

**Kata kunci:** Hyperparameter, Stacking Ensemble, Gradient Boosting, Pipeline, Machine Learning

## ABSTRACT

**Abstract** - Predicting calorie burn based on physical activity data is an important topic in the fields of health and fitness. The main focus is the ability to personalize fitness programs and monitor metabolic conditions more accurately. However, the main challenge in developing prediction models lies in accuracy and consistency, given the diversity and complexity of the data. This research develops an approach to building an effective prediction model using automated hyperparameter optimization with Optuna. The first step is to create a Tree-Based Pipeline Preprocessing to better handle the data. Next, a Stacking Ensemble method is applied using three base models based on Gradient Boosting: CatBoost, LightGBM, and XGBoost. The model is validated using K-Fold Cross Validation and Root Mean Squared Logarithmic Error (RMSLE) metrics. The results show an RMSLE value of 0.00599 and a prediction time of 0.0091 seconds. Furthermore, to test the consistency of the model, two techniques for encoding categorical features are applied: LabelEncoder with Standard Scaler and OrdinalEncoder without Standard Scaler. The differences in the scale of values produced by these techniques are the main reason for evaluating their impact on prediction time and consistency. The results reveal that the model remains consistently good, as indicated by the RMSLE and prediction time, which are not significantly different. This suggests that the combination of the right pipeline, ensemble models, and hyperparameter optimization can produce an accurate and stable prediction system.

**Keywords:** Hyperparameter, Stacking Ensemble, Gradient Boosting, Pipeline, Machine Learning

Copyright © 2025 Jurnal Teknik Elektro dan Komputasi (ELKOM)

## 1. PENDAHULUAN

Semakin banyak orang yang menyadari pentingnya hidup sehat dan menjaga kebugaran tubuh. Hal ini mendorong munculnya berbagai inovasi dalam alat yang dapat mengukur pembakaran kalori [1]. Banyak pusat kebugaran dan tempat kesehatan kini memanfaatkan alat-alat ini sebagai daya tarik tambahan bagi

pelanggan, dengan menawarkan pemantauan kalori yang lebih akurat sebagai bagian dari program kebugaran mereka. Prediksi pembakaran kalori berdasarkan aktivitas fisik yang dilakukan dapat membantu individu menentukan program kebugaran yang paling sesuai [2]. Dengan demikian, manfaat besar yang dapat diperoleh dari hasil prediksi pembakaran kalori mendorong pengembangan model prediksi yang lebih akurat dan andal.

Aktivitas fisik memiliki peran penting dalam menjaga kesehatan dan mencegah penyakit degeneratif. Menurut Badan Pusat Statistik (BPS, 2023), prevalensi obesitas dan penyakit tidak menular di Indonesia terus meningkat seiring dengan menurunnya aktivitas fisik masyarakat. Data Survei Sosial Ekonomi Nasional (Susenas) menunjukkan bahwa lebih dari 30% penduduk usia produktif kurang melakukan aktivitas olahraga teratur, sehingga risiko kelebihan berat badan dan gangguan metabolik semakin tinggi. Kondisi ini menegaskan urgensi ketersediaan metode monitoring aktivitas fisik yang akurat, terutama terkait estimasi pembakaran kalori.

Penelitian ini memiliki potensi aplikasi yang luas dalam industri kebugaran dan kesehatan. Dengan adanya model prediksi pembakaran kalori yang lebih akurat melalui optimasi hyperparameter, penyedia layanan kebugaran seperti aplikasi fitness tracker, smart wearable (jam pintar, gelang kesehatan), maupun platform olahraga daring dapat memberikan rekomendasi latihan yang lebih personal. Dalam bidang kesehatan, hasil prediksi ini dapat dimanfaatkan oleh dokter, ahli gizi, dan fisioterapis untuk memantau pasien dalam program diet, rehabilitasi, atau terapi berbasis aktivitas fisik. Integrasi model prediksi ini ke dalam sistem monitoring kesehatan digital akan meningkatkan efektivitas perencanaan program pengelolaan berat badan dan pencegahan penyakit metabolik seperti obesitas, diabetes, dan hipertensi. Dengan demikian, penelitian ini bukan hanya memberikan kontribusi teoretis pada pengembangan model machine learning, tetapi juga bernilai praktis dalam menunjang gaya hidup sehat, olahraga berbasis data, serta pelayanan kesehatan preventif dan kuratif.

Namun, meskipun banyak penelitian yang telah mengembangkan model prediksi pembakaran kalori, tantangan utama dalam bidang ini adalah mendapatkan model yang tidak hanya akurat, tetapi juga konsisten dan dapat diandalkan dalam berbagai kondisi. Keberagaman jenis dan kompleksitas data yang berbeda menjadi tantangan tersendiri. Selain itu, proses optimasi model untuk meningkatkan akurasi prediksi seringkali memerlukan waktu komputasi yang tinggi [3]. Oleh karena itu, analisis efisiensi model penting dilakukan dengan pendekatan dan pengujian yang berbeda untuk melihat konsistensi model dalam memprediksi data baru.

Beberapa penelitian sebelumnya juga mengembangkan model prediksi kalori yang terbakar berdasarkan data aktivitas fisik. Sebagai contoh, penelitian Aziz et al. (2023) menggunakan model XGBoost, SVM, dan Linear Regressor dengan hasil akurasi prediksi 99,67% untuk model XGBoost [2]. Sementara itu, Ratnakar et al. (2022) menggunakan model XGBoost yang menghasilkan nilai Mean Absolute Error (MAE) sebesar 2.7, atau akurasi sekitar 97,3% [4]. Penelitian Kadam et al. (2023) menggunakan model Random Forest Regressor, yang menghasilkan nilai Relative Mean Absolute Error (RMAE) sebesar 8,3 [3]. Perbedaan akurasi prediksi ini dapat disebabkan oleh teknik preprocessing data yang berbeda, serta pemilihan metode dan algoritma yang digunakan dalam pengembangan model.

Penelitian Reddy et al. (2024) mengombinasikan model XGBoost dan Random Forest Regressor serta menggunakan metode Lasso untuk penetapan fitur data dengan korelasi terbaik. Hasilnya adalah akurasi 83,73% [5]. Zhang et al. (2019) menggunakan algoritma Random Forest dan Support Vector Machine (SVM) untuk memperoleh model yang memiliki akurasi cukup baik, meskipun masih menghadapi tantangan dalam hal konsistensi dan waktu komputasi dengan data yang lebih besar [6]. Selain itu, penelitian Wang et al. (2020) mengembangkan metode deep learning dengan menggunakan Long Short-Term Memory (LSTM) untuk memodelkan data urutan aktivitas fisik. Meskipun LSTM dapat mengatasi masalah kompleksitas data, metode ini membutuhkan waktu pelatihan yang lebih lama [7]. Pendekatan optimasi hyperparameter untuk meningkatkan kinerja model prediksi kalori juga telah banyak dieksplorasi. Penelitian oleh Li et al. (2021) menggunakan algoritma optimasi berbasis Bayesian Optimization untuk meningkatkan akurasi model prediksi kalori [8], Vasudevan, S. et al. (2019) kemudian Bergstra, J. et al. (2012) membahas metode pencarian acak untuk optimasi hyperparameter dalam pembelajaran mesin [10]. Namun, optimasi hyperparameter sering menghadapi tantangan dalam hal efisiensi waktu dan akurasi pada data dengan dimensi fitur yang besar. Oleh karena itu, penelitian ini akan menggunakan metode yang sesuai untuk memilih hyperparameter terbaik secara otomatis.

Tujuan dari penelitian ini adalah untuk mengatasi beberapa tantangan tersebut dengan mengembangkan model prediksi pembakaran kalori yang lebih akurat dan stabil. Pendekatan yang diusulkan melibatkan pembuatan pipeline preprocessing berbasis tree model, penerapan metode stacking ensemble dengan model-model CatBoost, LightGBM, dan XGBoost, serta optimasi hyperparameter otomatis menggunakan Optuna. Dengan kombinasi ini, diharapkan dapat diperoleh model prediksi yang tidak hanya efektif dalam hal akurasi, tetapi juga konsisten dalam merespons data baru dengan waktu prediksi yang efisien.

## 2. METODE PENELITIAN

Berikut ini adalah metode penelitian yang digunakan sebagai pendekatan untuk mendapatkan model prediksi pembakaran kalori dari data aktivitas fisik yang memiliki akurasi dan stabilitas dalam memprediksi data baru.

### a. Pendekatan Penelitian

Penelitian ini menggunakan pendekatan eksperimental dan kuantitatif untuk mengembangkan model prediksi pembakaran kalori yang lebih akurat, stabil, dan efisien. Penelitian ini bertujuan untuk membandingkan berbagai teknik pemodelan dan optimasi hyperparameter untuk menghasilkan model yang konsisten dalam memprediksi kalori yang terbakar dari data aktivitas fisik.

### b. Dataset

Data yang digunakan dalam penelitian ini diambil dari salah satu data kompetisi yang diadakan di Kaggle Competition <https://www.kaggle.com/competitions/playground-series-s5e5>, yang diselenggarakan dari 1 Mei 2025 – 1 Juni 2025. Didalam dataset ini terdapat 3 file data yaitu data training, data testing, dan data submission. Untuk data training memiliki 9 feature dan berjumlah sekitar 750.000 data. Berikut ini adalah informasi setiap feature dalam dataset:

Data Training :

	id	Sex	Age	Height	Weight	Duration	Heart_Rate	Body_Temp	Calories
0	0	male	36	189.0	82.0	26.0	101.0	41.0	150.0
1	1	female	64	163.0	60.0	8.0	85.0	39.7	34.0
2	2	female	51	161.0	64.0	7.0	84.0	39.8	29.0
3	3	male	20	192.0	90.0	25.0	105.0	40.7	140.0
4	4	female	38	166.0	61.0	25.0	102.0	40.6	146.0

Gambar 1. Tampilan Dataset Pembakaran Kalori Dari Data Aktivitas Fisik

- 1) Id : berisi data id dari setiap individu yang melakukan rekam data aktifitas fisiknya.
- 2) Sex : data terkait jenis kelamin, dengan tipe data kategorikal yang terbagi atas male dan female.
- 3) Height : data terkait tinggi badan, dalam satuan cm
- 4) Weight : data terkait berat badan, dalam satuan kg
- 5) Duration : Lama waktu aktifitas fisik, dalam satuan menit
- 6) Heart\_Rate : Detak jantung dalam satuan bps
- 7) Body\_Temp : Suhu Tubuh setelah aktivitas fisik
- 8) Calories : Jumlah kalori yang terbakar

Feature data yang digunakan sama dengan feature dataset pada penelitian sebelumnya (misalnya, dataset yang digunakan oleh Aziz et al. [2], Ratnakar et al. [4], atau Zhang et al. [6]). Namun terdapat perbedaan jumlah dimensi data karena pada penelitian sebelumnya jumlah data hanya 15.000, sedangkan pada dataset yang digunakan ini lebih besar, untuk data training sebesar 750.000 dan data testing 250.000.

### c. Proses Preprocessing Data

Dataset akan melalui beberapa tahapan preprocessing untuk mempersiapkan data sebelum dimasukkan ke dalam model prediksi. Beberapa tahapan utama dalam preprocessing data adalah sebagai berikut:

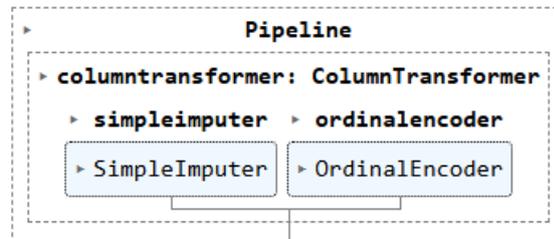
#### 1) Eksplorasi Data Analysis (EDA)

Tahap pertama adalah melakukan Eksplorasi Data Analysis (EDA) untuk memahami kondisi data secara mendalam. Pada langkah ini, kita akan menganalisis distribusi data, kompleksitas setiap fitur, serta hubungan atau korelasi antara fitur-fitur yang ada. Hasil dari EDA ini akan memberikan wawasan tentang karakteristik data, seperti apakah ada nilai yang hilang, apakah ada distribusi yang tidak merata, dan apakah fitur-fitur tersebut memiliki korelasi yang signifikan. Informasi ini akan digunakan untuk menentukan metode preprocessing yang tepat. Misalnya, dalam penelitian ini, beberapa metode yang digunakan antara lain: Ordinal Encoder dan Label Encoder untuk menangani fitur kategorikal. Simple Imputer untuk mengisi data yang hilang (missing values) dan Standard Scaler untuk menstandarkan dimensi data, khususnya bila menggunakan linear model Bishop, C. (2006) [11].

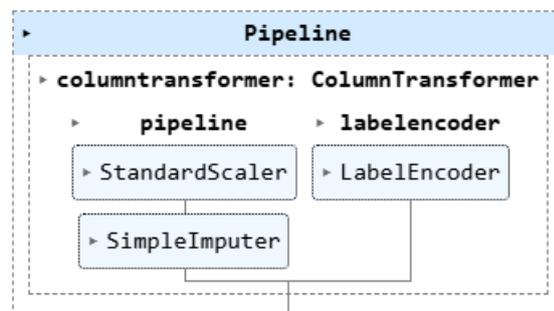
#### 2) Pembuatan Pipeline Preprocessing

Selanjutnya, akan dibuat Pipeline Preprocessing yang berfungsi untuk memproses data secara efisien. Pipeline ini memungkinkan transformasi data secara otomatis dan berurutan, mulai dari

menangani masalah distribusi data, pengkodean (labeling) data, hingga mengatasi nilai yang hilang. Dengan menggunakan pipeline, proses preprocessing menjadi lebih efisien karena kita dapat menerapkan serangkaian transformasi yang sama pada data numerik maupun kategorikal dalam satu langkah. Hal ini memastikan bahwa data yang diproses memiliki kualitas yang baik dan siap untuk dimasukkan ke dalam model prediksi.



Gambar 2. Pipeline Preprocessing Tanpa Standard Scaler



Gambar 3. Pipeline Preprocessing Dengan Standard Scaler

### 3) Pemilihan Fitur Data (X) dan Label (Y)

Setelah pipeline preprocessing selesai dibuat, langkah berikutnya adalah memilih fitur yang akan digunakan sebagai input (X) dan target atau label (Y) yang akan diprediksi. Dalam penelitian ini, pemilihan fitur dilakukan dengan bantuan metode Lasso (Least Absolute Shrinkage and Selection Operator). Metode Lasso ini berguna untuk memilih fitur-fitur yang memiliki korelasi yang kuat dengan label target, sekaligus mengurangi kompleksitas model dengan menghilangkan fitur-fitur yang kurang relevan. Lasso bekerja dengan cara memberikan batasan pada koefisien regresi, sehingga hanya fitur-fitur dengan kontribusi yang signifikan yang dipertahankan. Dengan cara ini, kita bisa memilih fitur terbaik yang membantu model menghasilkan prediksi yang lebih akurat.

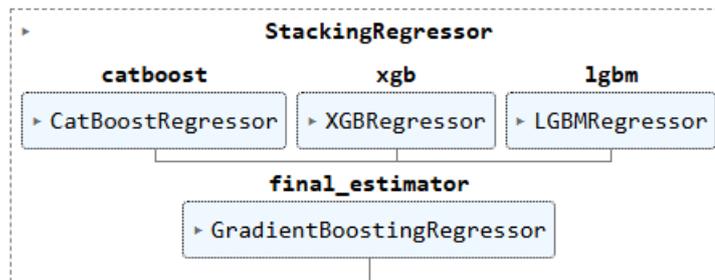
### 4) Proses Pemilihan Hyperparameter dengan Optuna

Optuna adalah metode yang digunakan untuk membantu memilih hyperparameter terbaik melalui serangkaian percobaan atau trial. Semakin banyak trial yang dilakukan, semakin optimal hasil pemilihan hyperparameter yang dapat diperoleh Akiba, T. et al. (2019) [12]. Dalam penelitian ini, dilakukan percobaan sebanyak 50 kali untuk menguji berbagai kombinasi hyperparameter pada tiga algoritma yang digunakan, yaitu *CatBoost*, *XGBoost*, dan *LGBM*.

Setiap percobaan (trial) akan mengevaluasi kombinasi parameter seperti learning rate, n-estimator, dan max-depth untuk menemukan konfigurasi yang paling optimal. Setelah 50 trial, model terbaik akan dipilih berdasarkan hasil evaluasi dari masing-masing kombinasi hyperparameter yang diuji. Dengan menggunakan Optuna, proses ini dapat dilakukan secara otomatis dan lebih efisien, sehingga memungkinkan pemilihan model dengan kinerja terbaik tanpa harus mencoba satu per satu secara manual.

### 5) Proses Training dan Validasi Model dengan Stacking Model

Stacking Ensemble model adalah teknik yang menggabungkan beberapa model dasar untuk menghasilkan prediksi akhir yang lebih akurat. Cara kerjanya adalah dengan melatih meta-model menggunakan output prediksi dari model-model dasar yang sudah dilatih sebelumnya. Dengan pendekatan ini, kita memanfaatkan kekuatan dan keunggulan masing-masing model dasar untuk menghasilkan prediksi yang lebih baik. Tujuannya adalah untuk meningkatkan akurasi secara keseluruhan dengan menggabungkan berbagai perspektif yang diberikan oleh model-model dasar. Meta-model akan belajar bagaimana menggabungkan hasil-hasil prediksi tersebut dengan cara yang optimal, sehingga menghasilkan prediksi yang lebih tepat.



Gambar 4. Stacking Model Dengan Meta-Model GBR

6) Analisis dan Evaluasi Stabilitas Model

Analisis dan evaluasi kinerja model dilakukan menggunakan metode *K-Fold Cross Validation* dan mengukur nilai *Root Mean Squared Logarithmic Error (RMSLE)*. Metode ini membantu memastikan bahwa model diuji dengan cara yang lebih representatif dan stabil, memberikan gambaran yang lebih akurat tentang performanya. Selain itu, pengujian stabilitas model dilakukan dengan membandingkan cara penanganan data kategorikal menggunakan dua metode encoding yang berbeda, yaitu *LabelEncoder* dengan *Standard Scaler* dan *OrdinalEncoder* Tanpa *Standard Scaler*. Kedua metode ini menghasilkan jumlah skala nilai fitur kategorikal yang berbeda.

Pada pengujian pertama, dengan *OrdinalEncoder* digunakan 7 fitur, sementara pada pengujian kedua, *LabelEncoder* dilakukan penambahan fitur menjadi 10 fitur, seperti fitur BMI, *Activity\_Level*, *Age\_Group*, dan *Heart\_Rate\_Group* yang ditambahkan untuk memperkaya informasi yang diberikan kepada model. Perbedaan jumlah fitur dan dimensi fitur inilah yang digunakan sebagai indikator stabilitas model, melihat seberapa baik model beradaptasi dengan kompleksitas data yang berubah-ubah. Untuk mengukur stabilitas model, dilakukan perhitungan rata-rata waktu prediksi untuk data baru dan membandingkan nilai RMSLE antara kedua percobaan. Dengan melihat perbedaan persentase pada RMSLE dan waktu prediksi antara kedua metode, kita dapat menilai apakah model memberikan hasil yang konsisten dan stabil saat berhadapan dengan data yang berbeda dalam hal dimensi dan kompleksitas.

**3. HASIL DAN PEMBAHASAN**

**3.1 Hasil Pipeline Preprocessing dan Subset Fitur dengan Metode Lasso**

Dalam penelitian ini, pada tahap pembuatan Pipeline Preprocessing, kami menyiapkan dua teknik dan metode Preprocessing yang berbeda, terutama untuk menangani tipe data kategorikal. Perbedaan dalam arsitektur pipeline ini dapat dilihat pada Gambar 2 dan Gambar 3. Kedua Pipeline Preprocessing ini diterapkan pada dataset yang sama, namun dengan penambahan beberapa fitur baru untuk meningkatkan kompleksitas data. Tujuan dari tahap ini adalah untuk menghasilkan fitur data dengan tingkat kompleksitas yang berbeda-beda, sehingga dapat digunakan sebagai tolak ukur untuk mengukur stabilitas model yang dihasilkan.

Setelah pipeline berhasil dibuat, langkah selanjutnya adalah menerapkan metode Lasso untuk memilih subset fitur yang memiliki korelasi yang baik dengan label prediksi. Berikut ini adalah hasil dari pipeline preprocessing dan pemilihan fitur menggunakan metode Lasso, yang kemudian diterapkan pada model prediksi.

	Sex	Age	Height	Weight	Duration	Heart_Rate	Body_Temp
0	36.0	189.0	82.0	26.0	101.0	41.0	1.0
1	64.0	163.0	60.0	8.0	85.0	39.7	0.0
2	51.0	161.0	64.0	7.0	84.0	39.8	0.0
3	20.0	192.0	90.0	25.0	105.0	40.7	1.0
4	38.0	166.0	61.0	25.0	102.0	40.6	0.0

Gambar 5. Skala Nilai Dengan Metode Ordinal Encoder

	Sex	Age	Weight	Duration	Heart_Rate	Body_Temp	BMI	Activity_Level	Age_Group	Heart_Rate_Group
0	1.001925	-0.357192	0.490201	1.266324	0.583714	1.235772	-0.939009	1.162994	0.012893	0.230261
1	-0.998079	1.487943	-1.083172	-0.888309	-1.109436	-0.431163	-1.185799	-0.923974	1.323360	0.230261
2	-0.998079	0.631273	-0.797104	-1.008011	-1.215258	-0.302938	0.208818	-1.022639	1.323360	0.230261
3	1.001925	-1.411555	1.062337	1.146622	1.007002	0.851095	0.025968	1.161922	-1.297573	0.230261
4	-0.998079	-0.225397	-1.011655	1.146622	0.689536	0.722869	-1.480887	1.081489	0.012893	0.230261

Gambar 6. Skala Nilai Dengan Metode Label Encoder

Berdasarkan gambar 6, dapat dilihat bahwa pada Pipeline Preprocessing dengan metode Ordinal Encoder, skala nilai untuk setiap fitur bersifat ordinal, yaitu memiliki urutan tertentu. Sementara itu, pada metode Label Encoder, skala nilai setiap fitur berupa bilangan desimal positif dan negatif. Hal ini terjadi karena pada pipeline preprocessing dengan metode Label Encoder ditambahkan langkah Standard Scaler, yang mengubah skala nilai fitur menjadi angka desimal. Selain itu, terlihat juga bahwa pada metode Label Encoder terdapat penambahan fitur baru untuk meningkatkan kompleksitas data, sehingga dapat digunakan sebagai standar dalam pengujian model.

### 3.2 Analisis Pengaruh Optimasi Hyperparameter dengan Optuna

Berdasarkan hasil optimasi hyperparameter menggunakan Optuna dengan trial sebanyak 50 kali pada setiap pengujian, berikut ini adalah tabel hasil optimasi Hyperparameter pada masing-masing pengujian stabilitas model.

Tabel 1. Hasil Trial Optuna Untuk Optimasi Hyperparameter

No.	Pengujian	Optimasi Hyperparameter			RMSLE
		learning Rate	n-Estimator	max-depth	
1.	LabelEncoder/Standard Scaler/10 Feature	0.035	2218	9	0.0594
2.	OrdinalEncoder/7 Feature	0.018	2993	11	0.0591

Pada pengujian pertama, pipeline preprocessing menggunakan LabelEncoder yang dikombinasikan dengan Standard Scaler dan 10 fitur yang dipilih dengan metode Lasso. Hasil Optimasi Hyperparameter menunjukkan nilai learning rate sebesar 0.035, dengan n-estimators 2218 dan kedalaman maksimum pohon keputusan (max-depth) sebesar 9. Dengan nilai RMSLE 0.0594, model ini memberikan hasil yang cukup baik.

Pengujian kedua yang menggunakan OrdinalEncoder dan hanya 7 fitur yang dipilih dengan metode Lasso. Dalam hal Optimasi Hyperparameter, learning rate yang digunakan adalah 0.018, dengan 2993 n-estimators dan kedalaman maksimum pohon keputusan (max-depth) sebesar 11. Nilai RMSLE yang lebih rendah, 0.0591, menunjukkan bahwa model ini sedikit lebih baik dalam memprediksi dibandingkan dengan pengujian pertama, tetapi perbedaan ini juga tidak terlalu signifikan Hastie, T. et al. (2009) [13].

Namun, pada pengujian kedua, dapat dilihat bahwa nilai n-estimator dan max-depth yang lebih tinggi dibandingkan dengan pengujian pertama menunjukkan bahwa model yang dihasilkan memiliki kompleksitas yang lebih besar. Hal ini bisa menjadi keuntungan atau malah sebaliknya, karena semakin kompleks model, semakin besar pula waktu prediksi dan beban komputasi yang dibutuhkan. Perbedaan ini juga tercermin pada nilai learning rate di setiap pengujian, yang biasanya berbanding lurus dengan waktu komputasi dan prediksi model. Artinya, model dengan learning rate lebih kecil cenderung membutuhkan waktu komputasi yang lebih lama, sementara learning rate yang lebih besar bisa mempercepat proses pelatihan, namun berisiko menyebabkan overfitting jika model tidak mampu mempelajari pola data secara baik. Untuk melihat performa model yang sebenarnya maka selanjutnya dilakukan pengujian model dengan data baru.

### 3.3 Hasil Stacking Model dengan Algoritma Gradient Boosting

Setelah memperoleh Optimasi Hyperparameter yang berbeda pada setiap pengujian sebelumnya, langkah selanjutnya adalah menerapkan Stacking Ensemble dengan menggunakan model Gradient Boosting Regressor sebagai meta-model (final estimator), sementara algoritma yang digunakan pada pengujian sebelumnya diterapkan sebagai model estimator (base-model). Untuk melihat lebih jelas tentang arsitektur stacking model yang digunakan, dapat dilihat pada Gambar 4. Implementasi selanjutnya adalah melakukan training dan evaluasi pada model stacking yang telah dibuat. Berikut ini adalah tabel hasil Testing Data Baru yang diuji menggunakan Stacking Model.

Tabel 2. Hasil Pengujian Stacking Model

No.	Pengujian	Stacking Model		
		RMSE	STD	Rata-Rata Waktu Prediksi (/detik)
1.	LabelEncoder/Standard Scaler /10 Feature	0.0599	0.00035	0.0091
2.	OrdinalEncoder/7 Feature	0.0597	0.00033	0.023

### 3.4 Evaluasi Konsistensi Model

Berdasarkan Tabel 2, hasil prediksi model Stacking terhadap pembakaran kalori menunjukkan kesalahan prediksi yang kecil dengan akurasi yang baik. Bahkan, jika dilihat dari pengujian kompleksitas fitur yang diterapkan, model ini tetap menghasilkan nilai RMSLE yang tidak berbeda secara signifikan. RMSLE (Root Mean Squared Logarithmic Error) adalah salah satu metrik evaluasi yang digunakan untuk mengukur kesalahan prediksi, dengan fokus pada jarak antara nilai aktual (ground truth) dan nilai prediksi model. Metrik ini sangat berguna ketika kita ingin mengukur kesalahan pada data yang memiliki distribusi yang tidak merata, seperti data yang melibatkan pembakaran kalori atau data eksponensial lainnya Sharma, S. et al. (2020) [14].

1. Sebagai contoh, misalkan hasil prediksi kalori terbakar adalah 95, dan nilai RMSLE yang diperoleh adalah 0.0599. Berdasarkan rumus RMSLE, kita dapat memperkirakan bahwa nilai aktual atau kalori yang sebenarnya berada dalam rentang sekitar 83.0 hingga 108.1. Dengan kata lain, kesalahan prediksi kalori dari nilai aktual dapat berkisar antara 12 hingga 13 kalori. Meskipun terdapat perbedaan antara nilai prediksi dan nilai aktual, kesalahan ini tidak dianggap signifikan atau fatal, terutama dalam konteks prediksi pembakaran kalori, karena selisih ini relatif kecil dan tidak memberikan dampak besar dalam pengambilan keputusan terkait program kebugaran atau pemantauan kesehatan. Dalam konteks ini, model prediksi kalori terbakar dapat dianggap cukup akurat untuk tujuan aplikasi praktis. Secara matematis, RMSLE dihitung dengan rumus berikut Geron, A. (2019) [15].

$$\text{RMSLE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (\log(p_i + 1) - \log(a_i + 1))^2} \quad (1)$$

## 4. KESIMPULAN

Penelitian ini berhasil mengembangkan model prediksi pembakaran kalori dengan dua teknik Pipeline Preprocessing yang berbeda, yaitu LabelEncoder dengan Standard Scaler dan OrdinalEncoder. Kedua teknik ini menambahkan fitur baru untuk meningkatkan kompleksitas data dan mengukur stabilitas model. Meskipun terdapat perbedaan dalam skala nilai dan kompleksitas, keduanya menghasilkan model yang cukup akurat. Pada tahap optimasi hyperparameter menggunakan Optuna, model dengan LabelEncoder/Standard Scaler dan 10 fitur menghasilkan nilai RMSLE 0.0594, sementara model dengan OrdinalEncoder dan 7 fitur menghasilkan nilai RMSLE 0.0591. Perbedaan nilai RMSLE yang sangat kecil menunjukkan bahwa kedua model memiliki akurasi prediksi yang hampir setara.

Selanjutnya, penerapan Stacking Ensemble dengan Gradient Boosting Regressor sebagai meta-model menunjukkan hasil yang sangat stabil, dengan nilai RMSLE yang hampir sama (0.0599 dan 0.0597) untuk kedua model. Ini mengindikasikan bahwa kedua model memberikan hasil prediksi yang konsisten meskipun ada perbedaan dalam kompleksitas dan jumlah fitur. Secara keseluruhan, model ini memiliki kesalahan prediksi yang sangat kecil dan dapat digunakan dengan baik untuk aplikasi praktis, seperti prediksi pembakaran kalori dalam bidang kebugaran atau pemantauan kesehatan.

Namun, penelitian ini juga memiliki beberapa kekurangan. Salah satunya adalah waktu pelatihan yang cukup lama, terutama karena banyaknya percobaan yang dilakukan selama proses optimasi Optuna. Hal ini disebabkan oleh keterbatasan pada sistem yang hanya mendukung pemrosesan menggunakan CPU. Selain itu, meskipun model telah menunjukkan stabilitas yang baik pada dataset yang ada, performa model pada dataset yang lebih besar dan lebih beragam masih perlu diuji lebih lanjut untuk memastikan konsistensinya di berbagai kondisi.

Untuk pengembangan penelitian selanjutnya, disarankan untuk mengeksplorasi teknik optimasi hyperparameter lainnya, seperti Bayesian Optimization atau Genetic Algorithms, untuk membandingkan efisiensi dan efektivitasnya dalam memilih hyperparameter terbaik. Penelitian lebih lanjut juga bisa mengeksplorasi penggunaan dataset yang lebih besar dan lebih beragam, serta memasukkan lebih banyak variabel, untuk meningkatkan akurasi prediksi kalori yang lebih realistis dan lebih dapat diandalkan di dunia nyata.

## REFERENSI

- [1] G. K. Reddy, "Proceedings of 2015 IEEE 9th International Conference on Intelligent Systems and Control, ISCO 2015," *Proc. 2015 IEEE 9th Int. Conf. Intell. Syst. Control. ISCO 2015*, 2015.
- [2] C. I. Engineering et al., "Calories Burnt Prediction Using Machine Learning Ap- Proach Literature Review :," vol. 1, no. 1, pp. 29–36.
- [3] A. Kadam, A. Shrivastava, S. K. Pawar, V. H. Patil, J. Michaelson, and A. Singh, "Calories Burned Prediction Using Machine Learning," *Proc. Int. Conf. Contemp. Comput. Informatics, IC3I 2023*, vol. 6, no. September, pp. 1712–1717, 2023..
- [4] S. S. Ratnakar and S. Vidya, "Calorie Burn Prediction using Machine Learning," vol. 9, no. 6, pp. 781–

- 787, 2022.
- [5] Reddy, M. R., Prakash, M., & Kumar, R. S. (2024). A hybrid approach combining XGBoost and Random Forest Regressor for calorie burn prediction. *Journal of Artificial Intelligence and Data Science*, 45(2), 321-335.
  - [6] Zhang, L., & Chen, X. (2019). Prediction of calorie expenditure based on Random Forest and Support Vector Machine models. *International Journal of Machine Learning and Computing*, 9(4), 505-514.
  - [7] Wang, Y., Li, Q., & Zhang, J. (2020). Activity prediction using Long Short-Term Memory (LSTM) for physical activity monitoring. *Neural Computing and Applications*, 32(8), 2451-2460.
  - [8] Li, Z., Zhang, W., & Wang, Y. (2021). Hyperparameter optimization for calorie prediction using Bayesian optimization techniques. *Computational Intelligence*, 37(3), 684-693.
  - [9] Vasudevan, S., & Thomas, S. (2019). Predicting Caloric Burn Based on User's Physical Activity using Decision Trees and Neural Networks. *Health Informatics Journal*, 25(4), 1415-1425.
  - [10] Bergstra, J., & Bengio, Y. (2012). Random Search for Hyper-Parameter Optimization. *Journal of Machine Learning Research*, 13(1), 281-305.
  - [11] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.
  - [12] Akiba, T., Sano, S., & Koyama, M. (2019). Optuna: A Next-Generation Hyperparameter Optimization Framework. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019.
  - [13] Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction (2nd ed.)*. Springer.
  - [14] Sharma, S., & Shukla, P. (2020). Calorie Burn Prediction Using Machine Learning Algorithms: A Case Study of Fitness Tracking Applications. *Proceedings of the 2020 IEEE International Conference on Artificial Intelligence and Data Science (AIDS)*, 2020.
  - [15] Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (2nd ed.)*. O'Reilly Media.