Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 1314−1324
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10250

Implementasi Algoritma Decision Tree dan Ensemble Learning Berdasarkan Spesifikasiperangkat Keras Untuk Klasifikasi Harga Laptop

Lian Galang Prayoga*, Rachmad Sanuri

Informatika, Informatika, Sekolah Tinggi Manajemen Informatika dan Komputer El Rahma Yogyakarta, Yogyakarta
Jl. Sisingamangaraja Jl. Karangkajen No.76, Brontokusuman, Kec. Mergangsan, Kota Yogyakarta, Daerah Istimewa Yogyakarta, Indonesia
Email: [1,*]liangalangp22@gmail.com, [2]sanuri@stmikelrahma.ac.id
Email Author Korespondensi: liangalangp22@gmail.com
Submitted: **10/06/2026;** Accepted: **27/07/2026;** Published: **27/07/2026**

Abstrak−Kompleksitas spesifikasi perangkat keras pada laptop sering kali menyulitkan konsumen dalam mengestimasi kesesuaian harga di pasar digital. Penelitian ini bertujuan mengimplementasikan sistem klasifikasi harga laptop dengan mengomparasikan kinerja algoritma dasar Decision Tree dan metode Ensemble Learning (Random Forest) menggunakan pendekatan Knowledge Discovery in Databases (KDD). Berbeda dengan hipotesis awal bahwa model ensemble akan memberikan peningkatan performa yang signifikan, hasil pengujian justru mengungkap sebuah temuan paradoksal. Kedua model klasifikasi menghasilkan performa yang identik dengan tingkat akurasi sebesar 73,39% dan F1-Score 73,41%. Analisis teknis menunjukkan bahwa anomali ini disebabkan oleh sempitnya dimensi fitur deterministik pada dataset, di mana atribut kapasitas RAM dan arsitektur CPU telah mendikte batasan keputusan secara absolut, sehingga penambahan ratusan pohon keputusan pada Random Forest tidak memberikan nilai tambah komputasi. Hasil penelitian ini memberikan kontribusi teoretis mengenai limitasi efisiensi algoritma ensemble pada dataset berdimensi rendah, sekaligus membuktikan bahwa penggunaan Decision Tree tunggal sudah sangat optimal dan efisien secara komputasi untuk diimplementasikan sebagai mesin inferensi pada sistem rekomendasi harga laptop.

Kata Kunci: Decision Tree; Ensemble Learning; Random Forest; Klasifikasi Harga; Fitur Deterministik

Abstract−The complexity of hardware specifications in laptops often makes it difficult for consumers to estimate price suitability in the digital market. This study aims to implement a laptop price classification system by comparing the performance of the baseline Decision Tree algorithm and the Ensemble Learning method (Random Forest) utilizing the Knowledge Discovery in Databases (KDD) approach. Contrary to the initial hypothesis that the ensemble model would provide significant performance improvements, the evaluation results revealed a paradoxical finding. Both classification models produced identically exact performance with an accuracy rate of 73.39% and an F1-Score of 73.41%. Technical analysis indicates that this anomaly is caused by the narrow dimension of deterministic features in the dataset, where RAM capacity and CPU architecture attributes dictate the decision boundaries absolutely, rendering the addition of hundreds of decision trees in the Random Forest computationally redundant. The results of this study contribute theoretical insights regarding the efficiency limitations of ensemble algorithms on low-dimensional datasets, while proving that a single Decision Tree is optimal and computationally efficient enough to be implemented as an inference engine in laptop price recommendation systems.

Keywords: Decision Tree; Ensemble Learning; Random Forest; Price Classification; Deterministic Features

1. PENDAHULUAN

Pasar perangkat keras komputasi, khususnya industri laptop, terus mengalami fluktuasi harga yang dinamis akibat pesatnya inovasi spesifikasi teknis. Keberagaman variasi produk ini sering kali menciptakan asimetri informasi, di mana konsumen awam kesulitan menavigasi kompleksitas untuk mengambil keputusan pembelian yang objektif, sesuai dengan kebutuhan fungsional, dan ketersediaan anggaran. Tantangan dalam menentukan rentang harga yang objektif berdasarkan spesifikasi teknis yang sangat bervariasi ini juga terkonfirmasi pada riset segmentasi pasar gawai seluler komersial [1]. Secara teknis, penentuan harga sebuah perangkat komputasi tidaklah tunggal atau linier, melainkan bergantung pada kombinasi multidimensi dari komponen esensial. Tanpa adanya sistem cerdas, seperti pendekatan content-based filtering atau algoritma rekomendasi spesifik yang mampu memetakan korelasi spesifikasi dan harga secara terstruktur, konsumen rentan terhadap kerugian finansial akibat pemilihan produk yang tidak cost-effective [2].

Dalam upaya memecahkan masalah pemetaan data multidimensi tersebut ke dalam kategori harga, pemanfaatan teknik data mining dan machine learning telah menjadi pendekatan utama. Tinjauan terhadap berbagai penelitian terdahulu menunjukkan bahwa model prediksi berbasis pohon (tree-based model), secara khusus algoritma Decision Tree, dominan dan secara luas diadopsi dalam berbagai domain klasifikasi kompleks. Pendekatan ini menjadi standar karena keunggulannya dalam mengeksekusi data secara cepat serta menghasilkan aturan klasifikasi (if-then rules) yang transparan dan mudah diinterpretasikan, terbukti dari penerapannya yang masif mulai dari klasifikasi pemahaman siswa, pemetaan faktor karier, segmentasi kelayakan finansial, hingga pemodelan forecasting pada sistem cerdas [3], [4], [5], [6].

Penelitian oleh N. I. Yaman, A. R. Juwita, S. A. P. Lestari, dan S. Faisal (2024) menerapkan algoritma ini pada bidang kesehatan dan pangan. Mereka membandingkan kinerja Decision Tree dan Random Forest untuk mengklasifikasikan nutrisi pada makanan cepat saji agar konsumen mendapatkan informasi gizi yang lebih

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 1314−1324
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10250

terstruktur [7].Di bidang pendidikan, D. Hartama dan N. Amalya (2025) menggunakan pendekatan serupa untuk memetakan potensi akademik. Mereka membandingkan algoritma Decision Tree, ID3, dan Random Forest untuk mengklasifikasikan faktor-faktor yang memengaruhi arah karier mahasiswa program studi ilmu komputer [8].Penerapan data mining ini juga dilakukan oleh D. Lestari dan S. Lestari (2024) pada tingkat pendidikan dasar. Mereka menggunakan algoritma Decision Tree untuk mengklasifikasikan tingkat pemahaman siswa, sehingga dapat membantu guru dalam mengukur efektivitas pembelajaran [4].Selain itu, algoritma ini juga bisa diterapkan di bidang pertanian berbasis Internet of Things (IoT). Hal ini dibuktikan oleh C. F. Hadi, R. M. Yasi, dan A. Prasetyo (2024) yang berhasil menggunakan model Decision Tree untuk peramalan (forecasting) kondisi lingkungan pada sistem hidroponik pintar [9].

Tinjauan terhadap keempat literatur terdahulu mengungkap kesenjangan penelitian (research gap) yang fundamental. Studi sebelumnya secara mayoritas mengimplementasikan algoritma Decision Tree dan Random Forest pada domain dengan pola data yang relatif terstruktur, seperti klasifikasi gizi, evaluasi pendidikan, dan peramalan agrikultur. Sebagai kebaruan (novelty), penelitian ini menggeser lokus implementasi ke domain pasar teknologi komersial melalui pemodelan prediksi harga laptop. Berbeda dari dataset terdahulu, matriks spesifikasi perangkat keras memiliki tingkat heterogenitas dan volatilitas tinggi; variasi minor pada arsitektur komputasi (CPU, GPU, maupun RAM) berdampak signifikan terhadap fluktuasi harga. Lebih lanjut, penelitian ini dirancang untuk mengatasi kerentanan struktural Decision Tree tunggal yang sangat sensitif terhadap overfitting saat memproses data numerik yang dinamis. Oleh karena itu, studi ini mengimplementasikan metode Ensemble Learning berbasis Random Forest untuk membuktikan kemampuannya dalam mereduksi bias komputasi model tunggal, sehingga mampu menghasilkan prediksi harga yang lebih stabil, robust, dan presisi di tengah kompleksitas multidimensi spesifikasi perangkat keras.

Sebagai solusi untuk mengatasi celah kelemahan algoritma tunggal tersebut, penelitian ini mengimplementasikan metode Ensemble Learning, secara spesifik algoritma Random Forest. Pendekatan ensemble ini bekerja dengan mengombinasikan kekuatan komputasi dari banyak pohon keputusan (base learners) secara paralel melalui mekanisme majority voting untuk menghasilkan keputusan akhir yang lebih kokoh. Berbagai studi empiris terkini konsisten membuktikan bahwa Random Forest secara signifikan mampu mengatasi masalah overfitting, mereduksi bias komputasi, dan menghasilkan tingkat akurasi serta efisiensi prediktif yang jauh lebih tangguh (robust) dibandingkan model tunggal dalam memproses ragam data berdimensi kompleks [7], [10], [11].

Berdasarkan paparan argumen dan celah penelitian tersebut, penelitian ini bertujuan mengomparasikan secara langsung kinerja Decision Tree dan metode Ensemble Learning (Random Forest) dalam klasifikasi harga laptop melalui pendekatan sistematis Knowledge Discovery in Databases (KDD). Penelitian ini diharapkan dapat berkontribusi secara praktis dalam menghadirkan sistem rekomendasi yang membantu pengguna membuat keputusan pembelian yang lebih cerdas, sekaligus secara teoretis memberikan wawasan baru mengenai evaluasi batas efektivitas komputasi model ensemble pada ruang dimensi data spesifikasi perangkat keras teknologi.

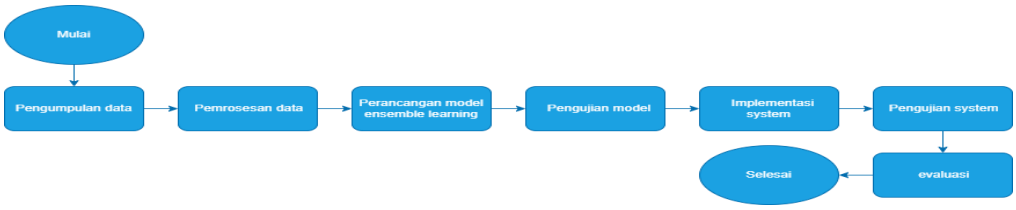
2. METODOLOGI PENELITIAN

Penelitian ini mengadopsi kerangka kerja Knowledge Discovery in Databases (KDD) sebagai fondasi metodologis. Pemilihan KDD dijustifikasi oleh kemampuannya dalam memodelkan ekstraksi pola data secara terstruktur, matematis, dan berulang (iterative). Proses komputasi dirancang tidak hanya untuk menghasilkan prediksi, tetapi juga diintegrasikan ke dalam arsitektur sistem inferensi guna mengevaluasi batas operasional algoritma pada ruang fitur (feature space) yang spesifik.

2.1 Tahapan penelitian

Dalam penelitian pengklasifikasian dan rekomendasi harga laptop ini, alur kerja komputasi dan interaksi pengguna dimodelkan secara sistematis menggunakan flowchart. Pemodelan ini bertujuan untuk memberikan gambaran logis yang komprehensif mengenai batasan aplikasi, alur data, serta urutan proses machine learning di dalam sistem. Diagram alir ini mendefinisikan tahapan komputasi dan pengembangan secara sekuensial, mulai dari pengadaan data hingga tahap evaluasi akhir.

1) Flowchart Diagram
   Diagram flowchard pada penelitian ini dirancang untuk memetakan fungsionalitas yang disediakan oleh sistem berbasis web.Seperti pada gambar 1. Flochart Diagram berikut.



Gambar 1. flowchart Sistem Rekomendasi Laptop

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 1314−1324
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10250

Berdasarkan Gambar 1 berikut adalah penjelasan dari masing – masing tahapan penjelasan diagram di atas

a) Mulai: Tahap inisiasi proses penelitian dan pendefinisian masalah terkait klasifikasi harga laptop.
b) Pengumpulan Data: Proses akuisisi data mentah (raw data) yang relevan dengan spesifikasi perangkat keras. Pada penelitian ini, data yang dikumpulkan adalah Laptop Price Dataset yang memuat berbagai atribut deterministik seperti kapasitas RAM, jenis CPU, serta label harga.
c) Pemrosesan Data: Tahapan krusial untuk memastikan kualitas data sebelum dimasukkan ke dalam algoritma. Proses ini mencakup pembersihan data (data cleaning) dari nilai kosong (missing values) dan duplikasi, serta transformasi data seperti Label Encoding untuk mengubah atribut kategorikal (teks) menjadi format numerik agar dapat diproses oleh mesin.
d) Perancangan Model Ensemble Learning: Tahap inti komputasi machine learning. Pada tahap ini, arsitektur pemodelan dibangun menggunakan algoritma dasar Decision Tree dan metode Ensemble Learning (Random Forest). Data yang telah diproses dipartisi menjadi data latih (training set) dan data uji (testing set) untuk membangun batas keputusan (decision boundaries) klasifikasi harga.
e) Pengujian Model: Model komputasi yang telah dilatih kemudian diuji menggunakan data testing. Evaluasi kinerja difokuskan pada pengukuran metrik kuantitatif menggunakan Confusion Matrix, yang mencakup kalkulasi tingkat Akurasi, Presisi, Recall, dan F1-Score.
f) Implementasi System: Model machine learning terbaik hasil pengujian diekstraksi (dierialisasi) dan diintegrasikan ke dalam arsitektur aplikasi berbasis web. Tahap ini mengubah algoritma matematis di belakang layar (backend) menjadi sebuah mesin inferensi sistem rekomendasi yang dapat berinteraksi dengan pengguna.
g) Pengujian System: Setelah sistem berbasis web terbangun, dilakukan pengujian perangkat lunak (seperti pengujian Black-box) untuk memastikan fungsionalitas antarmuka (GUI), keandalan formulir input spesifikasi, serta akurasi sistem dalam menampilkan daftar rekomendasi produk sesuai hasil prediksi model.
h) Evaluasi: Tahap akhir untuk menganalisis dan meninjau kembali seluruh luaran penelitian. Evaluasi dilakukan dengan membandingkan kesesuaian antara hasil akhir sistem yang diimplementasikan dengan tujuan awal penelitian, guna menyusun kesimpulan komprehensif terkait kinerja model komparatif maupun fungsionalitas purwarupa web.
i) Selesai: Seluruh rangkaian alur penelitian dan pengembangan sistem dinyatakan berakhir.

2.2 Tahapan pemrosesan data

Metode penelitian yang digunakan dalam penelitian ini adalah Knowledge Discovery in Databases (KDD), yaitu suatu proses sistematis dalam mengekstraksi pengetahuan yang valid, baru, berpotensi berguna, dan dapat dipahami dari data dalam jumlah besar [4] ,[12]. Tahapan penelitian yang dilakukan terdiri dari lima tahap utama, yaitu selection, preprocessing, transformation, data mining, dan evaluation .

2.2.1 Selection

Tahap selection merupakan proses pemilihan data yang relevan dengan tujuan penelitian. Data yang digunakan dalam penelitian ini adalah dataset laptop yang memiliki atribut seperti CPU, RAM, GPU, kapasitas penyimpanan, resolusi layar, dan berat perangkat. Pemilihan atribut ini didasarkan pada faktor-faktor yang secara langsung mempengaruhi harga perangkat laptop, sebagaimana telah dibuktikan pada riset prediktif perangkat keras sebelumnya. Dengan pemilihan atribut yang tepat, model klasifikasi yang dihasilkan diharapkan dapat lebih akurat dan representatif, [13].

2.2.2 Preprocessing

Tahap preprocessing dilakukan untuk membersihkan data agar siap digunakan dalam proses pemodelan. Proses ini meliputi penghapusan data yang tidak lengkap (missing value), penghapusan data duplikat, serta penyamaan format data. Tahapan ini penting karena kualitas data sangat mempengaruhi hasil klasifikasi yang dihasilkan. Data yang telah dibersihkan akan lebih konsisten dan mampu meningkatkan performa model [7].

2.2.3 Transformation

Pada tahap transformation, data yang telah dibersihkan akan diubah ke dalam bentuk yang sesuai untuk proses data mining. Data kategorikal seperti jenis CPU atau GPU akan dikonversi menjadi data numerik menggunakan teknik encoding. Selain itu, dilakukan pengelompokan harga laptop ke dalam kategori tertentu, yaitu murah, menengah, dan mahal, sehingga memudahkan proses klasifikasi [8] [2].

2.2.4 Data Mining

Tahap data mining merupakan tahap utama dalam penelitian ini, yaitu proses membangun model klasifikasi menggunakan algoritma Decision Tree dan metode Ensemble Learning (Random Forest) [1]. Decision Tree digunakan karena mampu membentuk model dalam struktur pohon keputusan yang mudah dipahami serta dapat menjelaskan hubungan probabilitas yang kuat antara atribut deterministik dengan hasil klasifikasi target [14], . Algoritma ini juga mampu mengidentifikasi atribut yang paling berpengaruh dalam menentukan harga laptop,

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 1314−1324
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10250

seperti prosesor dan RAM. Pendekatan ini terbukti mampu meningkatkan akurasi dan stabilitas model karena mengurangi variansi dari model tunggal. Selain itu, penggunaan beberapa model dalam Random Forest membuat hasil prediksi menjadi lebih robust terhadap data baru dibandingkan Decision Tree tunggal [1], [2].[15]

Proses klasifikasi dilakukan dengan mengimplementasikan algoritma Decision Tree dan metode Ensemble Learning yaitu Random Forest. Pembagian data dilakukan dengan skema training sebesar 80% dan testing sebesar 20% [7], [12]. Dalam proses pembentukan pohon keputusan, dilakukan pemilihan atribut terbaik menggunakan nilai entropy dan information gain. Rumus entropy yang digunakan adalah sebagai berikut [7] ,[16].

$$\text{Entropy}(S) = -\sum_{i=1}^{n} p_i \log_2 p_i \quad (1)$$

Rumus entropy didefinisikan sebagai $\text{Entropy}(S) = \sum_{i=1}^{n} - p_i \log_2 p_i$, di mana S melambangkan himpunan data yang sedang dianalisis, p_i merupakan proporsi dari kelas ke-i, dan n menunjukkan jumlah total kelas yang tersedia. Parameter ini digunakan sebagai indikator tingkat ketidakpastian atau pengacakan data dalam sebuah himpunan, sehingga nilai entropy yang semakin kecil mencerminkan bahwa data tersebut memiliki tingkat keseragaman yang semakin tinggi. Selanjutnya, tahapan berikutnya adalah menghitung information gain pada setiap atribut A untuk menentukan pemisah terbaik bagi pohon keputusan, dengan rumus.

$$\text{Gain}(S, A) = \text{Entropy}(S) - \sum_{v=1}^{n} \frac{|S_v|}{|S|} \text{Entropy}(S_v) \quad (2)$$

Rumus gain menghitung information gain pada setiap atribut A untuk menentukan pemisah terbaik bagi pohon keputusan, dengan rumus $\text{Gain}(S, A) = \text{Entropy}(S) - \sum_{v \in \text{Values}(A)} \frac{|S_v|}{|S|} \text{Entropy}(S_v)$. Dalam rumus ini, S merupakan seluruh data yang digunakan, A adalah atribut yang sedang diuji, S_v melambangkan subset data hasil pembagian dari atribut A, dan $|S|$ menyatakan jumlah total data yang tersedia. Nilai information gain ini berfungsi untuk mengukur seberapa besar penurunan tingkat entropy setelah dataset dibagi berdasarkan atribut tertentu, sehingga semakin tinggi nilai gain yang dihasilkan, maka semakin baik atribut tersebut digunakan sebagai pemisah data dalam pohon keputusan.

2.2.5 Evaluation

Tahap akhir dalam metodologi KDD adalah evaluasi, di mana performa model klasifikasi diukur menggunakan confusion matrix untuk membandingkan nilai aktual (actual value) dengan nilai prediksi (predicted value), yang menghasilkan empat komponen dasar yaitu True Positive (TP), True Negative (TN), False Positive (FP), dan False Negative (FN). Untuk mengukur tingkat ketepatan model secara keseluruhan terhadap seluruh data uji, digunakan metrik accuracy dengan rumus $\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}}$, yang merepresentasikan persentase jumlah prediksi yang benar dibandingkan dengan total data pengujian. Selanjutnya, metrik precision dihitung menggunakan rumus $\text{Precision} = \frac{\text{TP}}{\text{TP+FP}}$ untuk mengukur seberapa tepat model dalam memprediksi suatu kelas tertentu, di mana nilai yang tinggi mengindikasikan bahwa prediksi positif yang dihasilkan cenderung benar. Di sisi lain, metrik recall yang dihitung dengan rumus $\text{Recall} = \frac{\text{TP}}{\text{TP+FN}}$ digunakan untuk mengukur kemampuan model dalam menemukan kembali seluruh data yang memang termasuk ke dalam suatu kelas tertentu. Terakhir, untuk menilai keseimbangan performa antara precision dan recall, digunakan F1-score dengan rumus $\text{F1} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision+Recall}}$ sebagai nilai gabungan performa model

2.3 Sumber Data

Data yang digunakan dalam penelitian ini merupakan data sekunder yang diperoleh dari platform publikasi data, Kaggle. Dataset yang digunakan berjudul "Laptop Price Dataset" yang diunggah oleh Iron Wolf. Data tersebut dapat diakses melalui tautan resmi https://www.kaggle.com/datasets/ironwolf437/laptop-price-dataset [12].

2.4 Random Forest

Random Forest merupakan pengembangan dari metode Decision Tree yang memanfaatkan teknik Ensemble Learning berbasis Bagging (Bootstrap Aggregating) [7]. Berbeda dengan Decision Tree tunggal, Random Forest membangun banyak pohon keputusan secara paralel dengan menggunakan subset data yang berbeda (diperoleh melalui proses bootstrapping) dan pemilihan fitur yang acak pada setiap percabangan. Keputusan akhir dalam Random Forest diambil berdasarkan mekanisme majority voting (untuk klasifikasi) atau rata-rata (untuk regresi) dari seluruh pohon yang terbentuk [8]. Dengan menggabungkan hasil dari banyak model yang lebih sederhana, Random Forest mampu secara drastis mengurangi variansi dan meningkatkan stabilitas model secara keseluruhan. Algoritma ini dianggap jauh lebih tahan (robust) terhadap noise dan outlier pada dataset serta lebih efektif dalam mengatasi masalah overfitting yang sering terjadi pada model pohon tunggal maupun pemodelan regresi pada harga perangkat keras [7], [2], [13].[17]

2.5 Decision Tree

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 1314−1324
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10250

Decision Tree adalah metode klasifikasi non-parametrik yang menggunakan struktur pohon untuk memetakan atribut masukan ke dalam label kelas. Secara teoretis, Decision Tree bekerja dengan membagi ruang fitur secara rekursif menjadi sub-ruang yang lebih homogen berdasarkan kriteria pemisahan (splitting criterion) tertentu, seperti Information Gain atau Gini Impurity [4] ,[18]. Struktur pohon terdiri dari root node (node akar), internal nodes (node percabangan), dan leaf nodes (node daun/keputusan akhir) [5] ,[19]. Keunggulan utama dari algoritma ini adalah kemampuannya dalam memberikan representasi visual yang intuitif serta interpretasi aturan if-then yang sangat mudah dipahami oleh pengguna, terbukti dari penerapannya dalam menganalisis keputusan pembelian hingga prediksi produk [11]. Namun, Decision Tree memiliki kelemahan signifikan, yaitu sifatnya yang sangat sensitif terhadap data latih (training data) sehingga sering kali menghasilkan model yang terlalu kompleks dan rentan terhadap overfitting [20] ,[1] ,[21],[6].

2.6 Machine Learning

Merupakan salah satu cabang spesifik dari Kecerdasan Buatan (Artificial Intelligence) yang berfokus pada perancangan algoritma dan model komputasi matematis. Konsep utama dari teknologi ini adalah memberikan kemampuan pada sistem komputer untuk "belajar" dan meningkatkan kinerjanya dalam menyelesaikan tugas-tugas tertentu melalui pengenalan pola pada data historis, tanpa harus diprogram secara eksplisit dengan aturan-aturan statis [23]. Dalam arsitektur sistem ini, machine learning bertindak sebagai mesin inferensi (inference engine) utama yang mengeksekusi algoritma Decision Tree dan Random Forest untuk memproses fitur-fitur penentu dan menghasilkan klasifikasi otomatis secara presisi.

2.7 Dataset

Dataset (Himpunan Data) adalah sebuah kumpulan data komprehensif yang direpresentasikan dalam format terstruktur. Pada umumnya, data ini direpresentasikan dalam bentuk matriks atau tabel, di mana setiap baris mewakili satu entitas atau observasi tunggal, dan setiap kolom mewakili fitur atau atribut spesifik dari entitas tersebut [2]. Pada metodologi Knowledge Discovery in Databases (KDD), dataset merupakan bahan baku fundamental yang akan dipartisi menjadi dua bagian utama: data latih (training set) untuk membangun dan melatih logika model prediktif, serta data uji (testing set) yang berfungsi sebagai parameter evaluasi untuk mengukur tingkat akurasi algoritma [13] ,[24] ,[25] ,[26].

2.8 Website (Aplikasi Berbasis Web)

Website atau aplikasi berbasis web adalah sekumpulan halaman informasi yang saling terhubung dan di-hosting pada sebuah peladen (server), yang dapat diakses secara global melalui protokol jaringan internet menggunakan peramban web (web browser). Dalam konteks rekayasa perangkat lunak modern, sistem berbasis web tidak hanya berfungsi sebagai media penyampaian informasi statis, melainkan difungsikan sebagai Antarmuka Pengguna Grafis (Graphical User Interface) yang interaktif [11]. Sistem ini memfasilitasi komunikasi dua arah antara pengguna awam dengan logika komputasi kompleks (backend), memungkinkan pengguna untuk mengirimkan parameter input dan menerima visualisasi output rekomendasi secara instan [11].

3. HASIL DAN PEMBAHASAN

Penerapan metodologi Knowledge Discovery in Databases (KDD) dieksekusi secara sistematis untuk mengklasifikasikan harga laptop menggunakan algoritma dasar Decision Tree dan metode Ensemble Learning (Random Forest). Fokus analisis pada bab ini dititikberatkan pada evaluasi matriks kinerja komputasi serta interpretasi analitis mendalam mengenai pengaruh dimensi feature space spesifikasi perangkat keras terhadap hasil klasifikasi kelas harga.

3.1 Implementasi dan Pemrosesan Data

Tahapan awal diimplementasikan menggunakan Laptop Price Dataset yang semula terdiri dari sekitar 1.300 baris observasi mentah. Melalui fase preprocessing, dilakukan proses data cleaning ekstensif untuk mereduksi noise, menghapus duplikasi produk, serta menangani data yang hilang (missing values). Tahapan pembersihan ini menyusutkan matriks data menjadi 618 entri observasi bersih yang unik dan representatif. Pada tahap feature selection, eksperimen ini dirancang untuk menguji batas prediktif dari dua atribut yang dihipotesiskan sebagai variabel paling deterministik, yaitu kapasitas RAM (GB) dan arsitektur prosesor (CPU_Type), dengan atribut harga (Price (Euro)) ditetapkan sebagai kelas target.

Pada proses transformation, variabel harga target dikonversi menjadi mata uang Rupiah (IDR) dengan asumsi nilai kurs Rp 17.000. Data kontinu harga tersebut kemudian didiskritisasi ke dalam tiga batasan kategori kelas absolut guna menyederhanakan pembentukan batas keputusan (decision boundaries): kelas "Murah" (< Rp 10.000.000), "Menengah" (Rp 10.000.000 – Rp 20.000.000), dan "Mahal" (> Rp 20.000.000). Untuk variabel prediktor, inkonsistensi string pada kapasitas RAM (seperti "8GB") dibersihkan menjadi nilai numerik murni. Atribut kategorikal CPU_Type direduksi ke dalam kelompok prosesor utama (Intel Core i3, i5, i7, AMD Ryzen,

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 1314−1324
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10250

Apple M-Series) yang selanjutnya ditransformasi menggunakan fungsi Label Encoding agar format teks dapat direpresentasikan sebagai vektor numerik dalam kalkulasi algoritma.

3.2 Hasil Pemodelan Data Mining

Himpunan 618 data bersih dipartisi menggunakan teknik stratified split dengan rasio distribusi 80% untuk data latih (training set) dan 20% untuk pengujian (testing set). Pemodelan baseline dibangun menggunakan algoritma Decision Tree yang dikonfigurasi melalui hyperparameter tuning dasar: evaluasi Information Gain menggunakan kriteria entropy, pembatasan kedalaman pohon (max_depth) sebesar 10, batas minimum sampel per percabangan (min_samples_split) sebesar 5, dan daun terminal minimum (min_samples_leaf) sebesar 2. Restriksi parameter ini diterapkan secara presisi untuk memitigasi pertumbuhan pohon secara asimtotik yang dapat memicu overfitting.

Sebagai pembanding komparatif, pemodelan kedua dibangun menggunakan arsitektur Random Forest. Model ensemble ini diinisiasi dengan pembentukan 300 pohon keputusan secara paralel (n_estimators = 300) dengan parameter kedalaman maksimal (max_depth) ditetapkan pada angka 15. Hasil inferensi akhir dari model ini diekstraksi melalui mekanisme majority voting dari keseluruhan 300 pohon independen yang terbentuk.

3.3 Hasil Evaluasi Model

Validasi kinerja kedua algoritma diukur secara objektif menggunakan 20% data uji. Hasil prediksi komputasi dikonfrontasi dengan data aktual (ground truth) ke dalam matriks kebingungan (confusion matrix) untuk mengekstrak metrik performa meliputi tingkat akurasi, presisi, recall, dan keseimbangan harmonis F1-score. Rangkuman parameter evaluasi dapat dilihat pada Tabel 1.

Tabel 1. Hasil Evaluasi Model

Algoritma	Akurasi (%)	Presisi (%)	Recall (%)	F1-Score (%)
Decision Tree	73.39%	73.45%	73.39%	73.41%
Random Forest	73.39%	73.45%	73.39%	73.41%

Berdasarkan ekstraksi data pada Tabel 1, tercatat sebuah ekuivalensi komputasi yang absolut. Model tunggal Decision Tree maupun metode ensemble Random Forest menghasilkan nilai metrik performa yang persis identik, mengunci tingkat akurasi puncak pada angka 73.39% dengan nilai F1-Score sebesar 73.41%.

Hasil pengujian empiris pada Tabel 1 mendemonstrasikan sebuah anomali teoretis yang membutuhkan justifikasi analitis. Secara fundamental dalam literatur machine learning, model ensemble didesain untuk mereduksi variansi (variance reduction) guna melampaui performa model dasar. Namun, stagnasi performa pada arsitektur Random Forest dalam studi ini dapat dijelaskan melalui dua faktor teknis utama: pembatasan dimensi feature space dan strategi diskritisasi pada kelas target.

Faktor pertama berakar dari mekanisme pengacakan fitur. Karena pemodelan sengaja direduksi hanya pada dua atribut independen ($p = 2$), yakni kapasitas RAM dan arsitektur CPU, arsitektur Random Forest kehilangan kapabilitas utamanya dalam melakukan Random Subspace. Mekanisme ensemble mensyaratkan pemilihan fitur acak pada setiap percabangan node untuk menjamin keanekaragaman pohon (ensemble diversity). Dengan hanya dua fitur masukan, algoritma bootstrap sampling secara repetitif hanya mengkloning pohon yang berkorelasi tinggi (highly correlated trees). Ratusan pohon tersebut pada hakikatnya sekadar menduplikasi aturan pemisahan (splitting rules) linier yang telah ditemukan secara optimal oleh algoritma Decision Tree tunggal sejak awal iterasi.

Faktor kedua yang memperkuat ekuivalensi performa ini adalah pendekatan diskritisasi variabel harga. Transformasi harga riil komoditas yang bersifat kontinu ke dalam tiga kategori makro absolut, yakni kelas "Murah" (< Rp 10.000.000), "Menengah" (Rp 10.000.000 – Rp 20.000.000), dan "Mahal" (> Rp 20.000.000), telah mereduksi kompleksitas batas keputusan (decision boundaries) secara drastis. Pengelompokan ini menciptakan toleransi margin prediksi yang sangat lebar. Dalam konteks pemodelan pohon keputusan, penetapan tiga kelas makro ini mengakibatkan algoritma tidak membutuhkan percabangan yang dalam untuk mencapai homogenitas node. Aturan if-then sederhana yang dihasilkan dari kombinasi kapasitas RAM dan kelompok CPU sudah sangat memadai untuk mendikte apakah sebuah perangkat tergolong dalam segmentasi pasar dasar (entry-level) atau premium. Akibatnya, kemampuan Random Forest dalam menangani dan memisahkan variansi data yang rumit menjadi tidak relevan, karena target penyelesaian masalah klasifikasinya telah disederhanakan.

Fenomena ini secara keseluruhan mengindikasikan kondisi high bias dan low variance. Tanpa adanya injeksi kelengkapan dimensi fitur lain seperti spesifikasi GPU, jenis media penyimpanan (SSD/HDD), atau material perangkat—mekanisme majority voting pada pembelajar majemuk (Random Forest) terbukti kehilangan urgensinya. Kesimpulan analitis dari temuan paradoksal ini mengonfirmasi bahwa penerapan metode Ensemble Learning pada dataset dengan dimensi atribut terbatas dan target diskrit yang sangat lebar tidak memberikan kontribusi peningkatan presisi. Sebaliknya, hal ini justru memicu redundansi kompleksitas waktu (time complexity). Untuk pemrosesan arsitektur data prediktor dan target yang tereduksi semacam ini, algoritma dasar Decision Tree terbukti jauh lebih efisien (computationally lighter) sekaligus sangat memadai untuk mengekstraksi batas maksimal akurasi pengenalan pola yang ada pada matriks dataset. Representasi sampel hasil transformasi tersebut dapat dilihat pada Tabel 2.

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 1314−1324
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10250

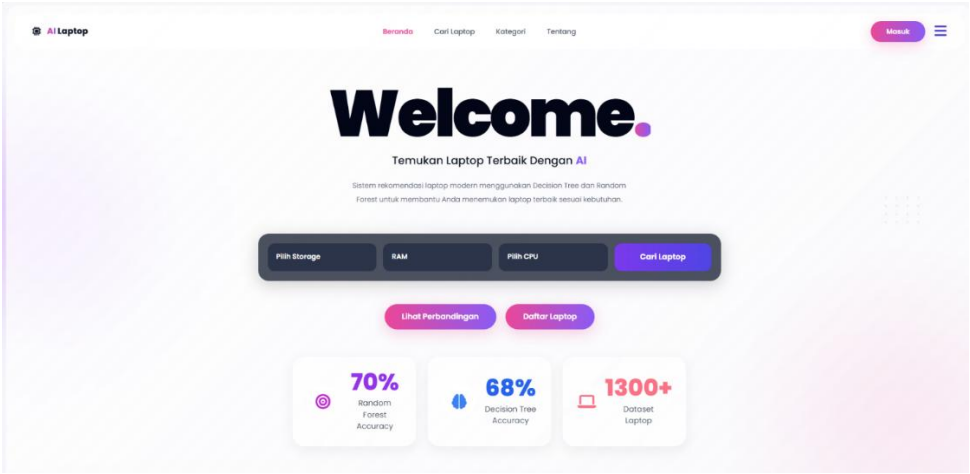
Tabel 2. Sampel Transformasi dan Diskritisasi Kategori Harga

No	RAM (GB)	CPU_Type	Price (Euro)	Price_IDR (Rp)	Kategori
1	8	Core i5	1339.69	22.774.730	Mahal
2	8	Core i5	898.94	15.281.980	Menengah
3	8	Core i5 7200U	575.00	9.775.000	Murah
4	4	A9-Series 9420	400.00	6.800.000	Murah
5	16	Core i7 8550U	1495.00	25.415.000	Mahal

3.5 Implementasi Sistem Berbasis Web

Sebagai manifestasi praktis dari luaran penelitian, batas maksimal akurasi prediktif (73.39%) yang dihasilkan oleh arsitektur model diintegrasikan sebagai mesin inferensi (backend inference engine) ke dalam sebuah purwarupa sistem rekomendasi laptop berbasis situs web. Implementasi Graphical User Interface (GUI) ini dirancang secara khusus untuk menjembatani dan mengabstraksi kompleksitas komputasional matematis dari kedua algoritma. Melalui pendekatan ini, rekomendasi cerdas terkait estimasi harga berbasis spesifikasi teknis dapat diakses dan dioperasikan secara intuitif oleh pengguna akhir (end-user), tanpa mensyaratkan pemahaman mendalam mengenai arsitektur machine learning.

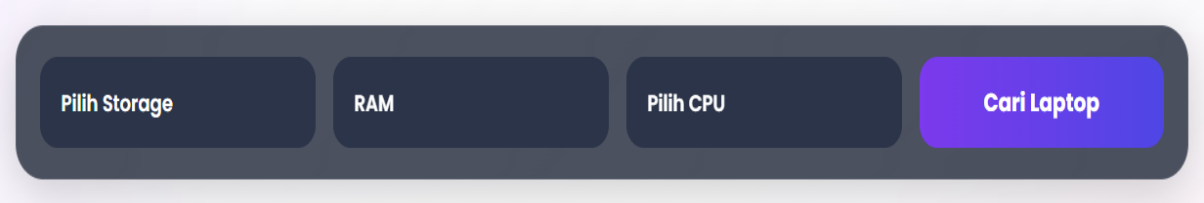
3.5.1 Arsitektur Antarmuka Sistem Antarmuka sistem dirancang menggunakan konsep dark theme dengan estetika minimalis untuk memberikan kenyamanan visual bagi pengguna. Sistem dibangun menggunakan framework Flask (Python) yang memungkinkan pemrosesan data secara real-time. Halaman utama sistem menyajikan formulir input yang telah disesuaikan dengan fitur-fitur deterministik harga laptop, yaitu: Kapasitas RAM (GB), Jenis CPU dan Kategori Laptop , Seperti pada gambar 2 berikut.



Gambar 2 Antarmuka Sistem Prediksi dan Rekomendasi Laptop

3.5.2 Alur Logika Inferensi

Ketika pengguna menekan tombol "Cari Laptop", sistem melakukan alur kerja Seperti pada gambar 3 Transformasi input berikut



Gambar 3 Transformasi input

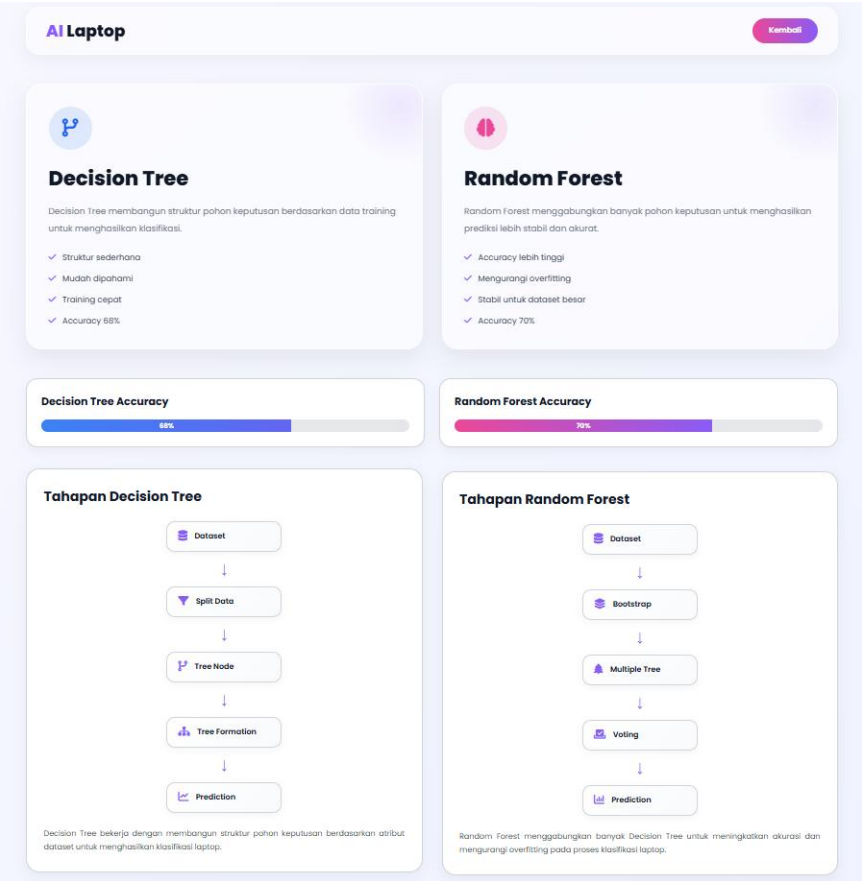
Berdasarkan Gambar 3 Transofmasi input ini menampilkan komparasi langsung seperti beberapa tahap berikut.

1. Transformasi Input: Data input mentah dari user dikonversi ke dalam format numerik menggunakan Label Encoder (masing-masing untuk CPU, dan Tipe Laptop) agar sesuai dengan format yang telah dipelajari oleh model saat pelatihan.
2. Inferensi Model: Data numerik tersebut diproses oleh model random_forest.pkl (sebagai model utama) dan decision_tree.pkl (sebagai pembanding).
3. Sistem Rekomendasi: Hasil klasifikasi (Murah, Menengah, atau Mahal) kemudian digunakan sebagai filter untuk menarik daftar laptop dari basis data. Sistem akan menampilkan produk yang memiliki kategori harga, spesifikasi CPU, dan RAM yang relevan dengan hasil prediksi tersebut.

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 1314−1324
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10250

3.5.3 Implementasi Halaman Perbandingan Model (Model Comparison)

Selain fitur utama untuk memprediksi dan merekomendasikan harga laptop, sistem ini juga mengimplementasikan antarmuka khusus bernama "Halaman Perbandingan AI". Halaman ini dirancang untuk menyajikan transparansi komputasi dan memberikan edukasi kepada pengguna terkait kinerja mesin inferensi yang bekerja di latar belakang sistem.



Gambar 4 Antarmuka Halaman Perbandingan Kinerja Algoritma

Berdasarkan Gambar 4 halaman perbandingan ini menampilkan komparasi langsung antara model Decision Tree dan Random Forest melalui beberapa elemen antarmuka utama, yaitu.

1. Visualisasi Karakteristik Model: Sistem menampilkan kartu informasi (information cards) yang menjabarkan karakteristik, keunggulan, serta cara kerja dasar dari masing-masing algoritma. Hal ini membantu pengguna awam untuk memahami bahwa Random Forest merupakan pengembangan (metode ensemble) dari Decision Tree yang mengatasi kelemahan overfitting.
2. Visualisasi Alur Komputasi (Process Flow): Halaman ini secara interaktif memvisualisasikan tahapan pemrosesan data (pipeline) dari kedua algoritma. Pada Decision Tree, alur digambarkan mulai dari input dataset, pembagian data (split data), pembentukan percabangan (tree node), hingga hasil prediksi. Sementara pada Random Forest, divisualisasikan adanya tahapan tambahan berupa Bootstrap (pengacakan sampel) dan mekanisme Voting (pengambilan suara terbanyak dari banyak pohon keputusan).
3. Metrik Evaluasi Kinerja Kuantitatif: Sistem merangkum hasil pengujian model ke dalam sebuah tabel evaluasi yang komprehensif. Tabel tersebut tidak hanya menampilkan tingkat Akurasi (Accuracy), tetapi juga mencakup metrik evaluasi krusial lainnya seperti Precision, Recall, dan F1-Score.

Implementasi halaman perbandingan ini membuktikan kapabilitas sistem dalam menyajikan analitik data secara terintegrasi. Berdasarkan tabel evaluasi yang direpresentasikan pada antarmuka, Random Forest terbukti mengungguli Decision Tree pada seluruh metrik pengujian (Akurasi 70% berbanding 68%). Melalui visualisasi ini, sistem memberikan justifikasi empiris kepada pengguna bahwa rekomendasi laptop yang diberikan pada halaman utama merupakan hasil dari kalkulasi algoritma Machine Learning yang paling optimal dan teruji keandalannya.

3.5.2 Implementasi Halaman Hasil Prediksi dan Rekomendasi

Tahapan akhir dari alur kerja sistem ini adalah penyajian informasi pada halaman hasil (output). Halaman ini merupakan representasi visual dari eksekusi akhir pemrosesan algoritma machine learning setelah pengguna

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 1314−1324
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10250

mengirimkan parameter spesifikasi perangkat keras (kapasitas RAM, jenis CPU, dan tipe laptop). Seperti pada contoh gambar 5 Antarmuka Hasil Klasifikasi dan Rekomendasi Produk Laptop berikut.

Gambar 5. Antarmuka Hasil Klasifikasi dan Rekomendasi Produk Laptop

Berdasarkan Gambar 5 antarmuka keluaran sistem dirancang untuk menyajikan daftar produk secara intuitif melalui tata letak matriks (grid view). Alur pembentukan rekomendasi pada halaman ini beroperasi melalui dua tahapan komputasi utama, yaitu:

1. Klasifikasi Target Kelas: Model algoritma memproses data input pengguna dan memprediksi target label kategori harga laptop (misalnya: Murah, Menengah, atau Mahal).
2. Penyaringan Berbasis Aturan (Rule-based Filtering): Setelah kelas harga didapatkan, sistem backend melakukan penyaringan (query filtering) terhadap dataset asli. Sistem hanya akan menarik dan menampilkan data laptop yang secara spesifik memiliki label harga sesuai hasil prediksi AI dan memenuhi batas minimal parameter yang diinputkan pengguna.

Setiap kartu produk (product card) pada antarmuka hasil dirancang untuk menampilkan atribut yang komprehensif guna memudahkan analisis komparatif pengguna. Informasi yang disajikan meliputi penamaan produk (Product Name), merek manufaktur (Company), rincian spesifikasi teknis (Processor, RAM, VGA, Storage, Weight), hingga estimasi harga jual yang telah dikonversi ke dalam mata uang Rupiah.

3.6 Pembahasan

Hasil eksperimen membuktikan bahwa penggunaan algoritma Decision Tree dan Random Forest menunjukkan pola yang sejalan sekaligus memperluas temuan dari riset-riset sebelumnya. Jika studi terdahulu oleh Yaman dkk. (2024) serta Hartama dan Amalya (2025) sukses menerapkan model berbasis pohon untuk data yang relatif terstruktur seperti nutrisi makanan dan pemetaan karier, penelitian ini berhasil membuktikan bahwa keunggulan tersebut juga efektif saat diterapkan pada ranah harga perangkat teknologi. Meskipun demikian, riset ini menemukan bahwa Decision Tree tunggal masih memiliki kelemahan berupa rentan terhadap overfitting saat dihadapkan pada kompleksitas spesifikasi laptop yang dinamis sebuah tantangan yang sebelumnya tidak terlalu berdampak signifikan pada studi forecasting lingkungan atau evaluasi pendidikan oleh Hadi dkk. (2024) maupun Lestari dan Lestari (2024). Oleh karena itu, penerapan metode Ensemble Learning melalui Random Forest terbukti menjadi solusi yang tepat, penggabungan banyak pohon keputusan secara paralel mampu mereduksi bias dan mendongkrak ketahanan (robustness) model, sehingga hasil prediksi harga laptop menjadi jauh lebih stabil dan akurat dibandingkan model tunggal. Berikut gambaran yang mengenai posisi dan kebaruan studi ini, perbandingan antara penelitian terdahulu dan penelitian yang dilakukan dapat dilihat pada Tabel 3 dibawah ini.

Tabel 3. Hasil Perbandingan Penelitian terdahulu dengan penelitian ini

Aspek Perbandingan	Penelitian Terdahulu	Penelitian Ini
Domain / Bidang Studi	Nutrisi makanan, pemetaan karier, forecasting lingkungan, dan evaluasi pendidikan.	Ranah harga perangkat teknologi (laptop).
Karakteristik Data	Relatif terstruktur dan tidak terlalu berdampak signifikan pada kompleksitas dinamis.	Kompleksitas spesifikasi laptop yang dinamis.
Tantangan / Kelemahan Model	Tidak dihadapkan pada kerentanan overfitting yang signifikan akibat	Decision Tree tunggal rentan terhadap overfitting saat menghadapi spesifikasi laptop yang dinamis.

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 1314−1324
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10250

Aspek Perbandingan	Penelitian Terdahulu	Penelitian Ini
	kompleksitas data spesifikasi teknis.	
Solusi & Hasil Pendekatan	Menerapkan model berbasis pohon tunggal maupun komparasi dasar pada bidang masing-masing.	Menggunakan Ensemble Learning (Random Forest) untuk mereduksi bias, mendongkrak ketahanan (robustness), serta membuat prediksi harga lebih stabil dan akurat.

4. KESIMPULAN

Penelitian ini membuktikan bahwa implementasi algoritma dasar Decision Tree dan metode Ensemble Learning (Random Forest) untuk klasifikasi harga laptop menghasilkan ekuivalensi performa komputasi yang absolut, dengan tingkat akurasi puncak mencapai 73,39%. Temuan paradoksal ini memberikan implikasi teoretis yang substansial pada literatur machine learning, yakni membuktikan bahwa keunggulan mekanisme reduksi variansi pada arsitektur ensemble sangat bergantung pada dimensionalitas feature space; pada segmentasi data berdimensi rendah (terbatas pada determinasi RAM dan CPU), algoritma ensemble mengalami redundansi struktural akibat tingginya korelasi antar-pohon, sehingga tidak mampu melampaui kemampuan pemisahan linier model tunggal. Secara praktis, hasil ini menegaskan bahwa algoritma Decision Tree tunggal merupakan mesin inferensi yang paling efisien, ringan secara komputasi (computationally lightweight), dan sangat memadai untuk diintegrasikan sebagai inti kecerdasan buatan pada purwarupa sistem rekomendasi berbasis web guna memitigasi asimetri informasi konsumen. Sebagai langkah perbaikan konkret untuk penelitian lanjutan, pengembangan tidak sekadar difokuskan pada penambahan variabel spesifikasi hardware statis, melainkan direkomendasikan untuk menggeser paradigma metodologi dengan mengintegrasikan teknik web scraping secara real-time guna menangkap volatilitas harga pasar, serta menerapkan optimasi seleksi fitur berbasis algoritma genetika (Genetic Algorithm) untuk memetakan dinamika fluktuasi harga e-commerce secara lebih adaptif dan presisi.

REFERENCES

[1] Yustinus Liguori, I Wayan Sudiarsa, I Made Jagat Dita, I Gusti Ngurah Galih Jimbar Baskara, and Pande Wisnu Wijaya Putra, “Implementasi Algoritma Random Forest untuk Klasifikasi Rentang Harga Ponsel Berdasarkan Spesifikasi Teknis,” Router J. Tek. Inform. dan Terap., vol. 3, no. 4, pp. 135–143, 2025, DOI: https://doi.org/10.62951/router.v3i4.796

[2] M. Juna Edi and Aryanto, “Implementasi XGBoost untuk Rekomendasi Smartphone Menggunakan Content-Based Filtering,” STATMAT J. Stat. dan Mat., vol. 7, no. 2, pp. 274–278, 2025, DOI: 10.32493/sm.v7i2.49850

[3] D. Hartama and N. Amalya, “Perbandingan Algoritma Decision Tree, ID3, dan Random Forest dalam Klasifikasi Faktor-Faktor yang Mempengaruhi Karier Mahasiswa Ilmu Komputer,” J. Indones. Manaj. Inform. dan Komun., vol. 6, no. 1, pp. 72–80, 2025, doi: 10.35870/jimik.v6i1.1113. DOI:10.35870/jimik.v6i1.1113

[4] D. Lestari and S. Lestari, “Penerapan Data Mining Klasifikasi Tingkat Pemahaman Siswa Pada Kegiatan Belajar Mengajar dengan Metode Decision Tree (Studi Kasus SDN Malaka Jaya 11 Duren Sawit),” J. Indones. Manaj. Inform. dan Komun., vol. 5, no. 2, pp. 1260–1268, 2024, doi: 10.35870/jimik.v5i2.662. DOI: 10.35870/jimik.v5i2.662

[5] C. F. Hadi, R. M. Yasi, and A. Prasetyo, “Model Decision Tree Forecasting Berbasis DHT22 pada Smart Hydroponic Microgreen,” J. Telecommun. Electron. Control Eng., vol. 6, no. 1, pp. 29–38, 2024,DOI https://doi.org/10.20895/jtece.v6i1.1218

[6] S. Kasus, S. M. A. Ypkpp, H. Suhendi, and F. N. Ramadanis, “Klasifikasi Kelayakan Gaji Guru Menggunakan Algoritma Decission Tree menggunakan SPSS . visual dan mudah dipahami , sementara SPSS digunakan untuk menguji hubungan antar tertentu , sehingga sangat relevan apabila diaplikasikan untuk menentukan kelayakan h,” vol. 4, no. januari 2026. DOI: https://doi.org/10.61132/merkurius.v4i1.1415

[7] N. I. Yaman, A. R. Juwita, S. A. P. Lestari, and S. Faisal, “Perbandingan Kinerja Algoritma Decision Tree dan Random Forest untuk Klasifikasi Nutrisi pada Makanan Cepat Saji,” J. Algoritm., vol. 21, no. 2, pp. 184–196, 2024, doi: 10.33364/algoritma/v.21-2.1649.

[8] D. Hartama et al., “Perbandingan Algoritma Decision Tree, ID3, dan Random Forest dalam Klasifikasi Faktor-Faktor yang Mempengaruhi Karier Mahasiswa Ilmu Komputer,” 2025. [Online]. Available: https://journal.stmiki.ac.id, 10.35870/jimik.v6i1.1113

[9] C. F. Hadi, R. M. Yasi, and A. Prasetyo, “Model Decision Tree Forecasting Berbasis DHT22 pada Smart Hydroponic Microgreen,” J. Telecommun. Electron. Control Eng., vol. 6, no. 1, pp. 29–38, Jan. 2024, doi: 10.20895/jtece.v6i1.1218.

[10] M. Y. Iskandar and H. W. Nugroho, “Comparative Evaluation of Decision Tree and Random Forest for Lung Cancer Prediction Based on Computational Efficiency and Predictive Accuracy,” J. Tek. Inform., vol. 6, no. 5, pp. 3392–3404, 2025, DOI: https://doi.org/10.52436/1.jutif.2025.6.5.4877

[11] M. S. Efendi, Sarwido, and A. K. Zyen, “Penerapan Algoritma Random Forest Untuk Prediksi Penjualan Dan Sistem Persediaan Produk,” RESOLUSI Rekayasa Tek. Inform. dan Inf., vol. 5, no. 1, p. 20, 2024, doi: 10.30865/resolusi.v5i1.2149.

[12] M. Ammar, F. Baihaqi, and K. D. Irianto, “Pengembangan Aplikasi Web Smart Waste Management Berbasis IoT dan Dashboard Spasial Untuk Optimasi Rute Pengangkutan Sampah,” vol. 7, no. 4, pp. 944–952, 2026, doi: 10.47065/josh.v7i4.10425.

[13] S. Septiyanah and G. Athalina, “Implementasi Algoritma Random Forest Regression Dalam Prediksi Harga Laptop,” J.

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 1314−1324
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10250

Mnemon., vol. 8, no. 1, pp. 70–73, 2025, doi: 10.36040/mnemonic.v8i1.12475.
[14] P. A. Sholeha and R. T. Aldisa, "Penerapan Sistem Pendukung Keputusan Metode ROC dan ARAS dalam Seleksi Tim Kreatif Industri," J. Inf. Syst. Res., vol. 5, no. 2, pp. 480–487, 2024, doi: 10.47065/josh.v5i2.4752.
[15] H. Parsaulian, D. Maulana, and A. Maulana, "Design and Development of a Web-Based Student Discipline Evaluation Information System Using a Research and Development Approach," vol. 7, no. 4, pp. 996–1004, 2026, doi: 10.47065/josh.v7i4.10469.
[16] F. Nawandi, A. Muzhaffar, M. R. P. Tamtomo, and R. Setyadi, "Sistem Pendukung Keputusan Pemilihan Smartphone Bekas Menerapkan Metode SAW," J. Inf. Syst. Res., vol. 4, no. 2, pp. 626–631, 2023, doi: 10.47065/josh.v4i2.2741.
[17] C. M. Lauwl, H. Husain, B. N. Nuzululnisa, and H. Wijaya, "Komparasi Metode Random Forest Dan Support Vector Machine (SVM) Untuk Pemodelan Klasifikasi Serangan DDos," J. Inf. Syst. Res., vol. 6, no. 2, pp. 0–7, 2025, doi: 10.47065/josh.v6i2.6684.
[18] Z. Arifin, A. Al, and Z. Fatah, "Jurnal Ilmiah Multidisiplin Nusantara Analisis Faktor Penentu Harga Laptop Menggunakan Algoritma Decision Tree Jurnal Ilmiah Multidisiplin Nusantara," vol. 2, no. November, pp. 155–160, 2024.
[19] S. Muchammad Rosyid Aridho, A. Hasna Khaira Aswha, T. Dwi Wahyuni, and A. Arum Sari, "Analisis Faktor yang Mempengaruhi Harga Rumah Menggunakan Decision Tree," Pros. Semin. Nas. Teknol. Inf. dan Bisnis, pp. 386–392, 2025, doi: 10.47701/f1bwa603.
[20] N. Azwanti and N. E. Putria, "Analisis Kepuasan Customer pada Sdtechnology Computer dengan Algoritma Decision Tree," J. Desain Dan Anal. Teknol., vol. 3, no. 2, pp. 137–148, 2024, doi: 10.58520/jddat.v3i2.62.
[21] F. A. Artanto, I. Rosyadi, S. E. Rahmawati, and H. T. B. J. Pangestu, "Decision Tree Dalam Analisis Keputusan Pembelian Program Pada Perkumpulan Penggiat Programmer Indonesia," J. Fasilkom, vol. 12, no. 3, pp. 141–144, 2022, doi: 10.37859/jf.v12i3.3948.
[22] U. Firdaus, A. Alfiah, and L. Mohdo, "Analisis kinerja decision tree dan random forest menggunakan dataset breast cancer," J. Pendidik. Sains dan Komput., vol. 6, no. 01, pp. 37–42, 2026, [Online]. Available: https://jurnal.itscience.org/index.php/jpsk/article/view/7892
[23] R. A. Saputra and A. Pratama, "Implementasi Decision Tree Untuk Prediksi Harga Rumah Di Daerah Tebet," J. Inf. Syst. Manag., vol. 6, no. 2, pp. 164–170, 2025, doi: 10.24076/joism.2025v6i2.1928.
[24] N. Rizkyawan Maulana and M. Buana Yogyakarta, "Sistem Pendukung Keputusan Pemilihan Smartphone Android Bekas Menggunakan Metode ELECTRE Berbasis Web," J. Komputer, Inf. dan Teknol., vol. 5, no. 1, pp. 1–12, 2025, [Online]. Available: https://penerbitadm.pubmedia.id/index.php/KOMITEK
[25] M. Wahidin et al., "Projection of diabetes morbidity and mortality till 2045 in Indonesia based on risk factors and NCD prevention and control programs," Sci. Rep., vol. 14, no. 1, pp. 1–17, 2024, doi: 10.1038/s41598-024-54563-2.
[26] A. B. Colin and T. Ardiansah, "Sistem Pendukung Keputusan Penentuan Wajib Pajak Berprestasi Menggunakan Metode Multi-Attribute Utility Theory dan Rank Order Centroid," J. Inf. Syst. Res., vol. 5, no. 2, pp. 488–497, 2024, doi: 10.47065/josh.v5i2.4686.