

Pemodelan Klasifikasi Kategori Harga Rumah Menggunakan Algoritma Decision Tree dengan Pendekatan CRISP-DM

Bunga Aulia Ghaisani¹, Griselda Audrye Damayanti², Rizki Mautya A. Rani³, Nur Fitri⁴, Weiskhy Steven Dharmawan⁵

^{1,2,3,4} Program Studi Informatika, Fakultas Teknik dan Informatika, Universitas Bina Sarana Informatika

⁵ Program Studi Sistem Informasi Akuntansi, Fakultas Teknik dan Informatika, Universitas Bina Sarana Informatika Jl. Abdul Rahman Saleh No.18, Bangka Belitung Laut, Kec. Pontianak Tenggara, Kota Pontianak, Kalimantan Barat, (0561) 583924, Indonesia

E-mail : 15230687@bsi.ac.id¹, 15230680@bsi.ac.id², 15230671@bsi.ac.id³, 15230603@bsi.ac.id⁴, weiskhy.wvn@bsi.ac.id⁵

Abstract— Accurate house price prediction is crucial information for various stakeholders in the property market, including buyers, sellers, and investors, for strategic decision-making. This study aims to design and implement a machine learning model using the Decision Tree Algorithm, known for its ease of interpretation, to classify house prices into three categories: low, medium, and high. This study adds the Year Built variable to fill the gap in previous research. The applied methodology adopts the CRISP-DM (Cross-Industry Standard Process for Data Mining) framework, which includes the stages of data understanding, data preparation, modeling, evaluation, and deployment. The Decision Tree Algorithm is used as the main predictive model, and its performance is measured using classification accuracy metrics. As a final stage, the model has been trained on an interactive web application system using Streamlit, a modern Python-based framework that accelerates the deployment process. The results show that the Decision Tree model is able to classify house price categories with an accuracy rate of 81.74%. The implementation using Streamlit has proven successful in providing an effective, intuitive, and practical interface, allowing non-technical users to interact directly with the predictive model.

Abstrak— Prediksi harga rumah yang akurat merupakan informasi krusial bagi berbagai pemangku kepentingan di pasar properti, termasuk pembeli, penjual, dan investor, untuk pengambilan keputusan strategis. Penelitian ini bertujuan untuk merancang dan mengimplementasikan model machine learning menggunakan Algoritma Decision Tree, yang dikenal karena kemudahan interpretasinya, untuk mengklasifikasikan harga rumah ke dalam tiga kategori: rendah, sedang, dan tinggi. Penelitian ini menambahkan variabel Year Built untuk mengisi gap penelitian sebelumnya. Metodologi yang diterapkan mengadopsi kerangka kerja CRISP-DM (Cross-Industry Standard Process for Data Mining), yang meliputi tahapan pemahaman data, persiapan data, pemodelan, evaluasi, hingga deployment. Algoritma Decision Tree digunakan sebagai model prediktif utama, dan kinerjanya diukur menggunakan metrik akurasi klasifikasi. Sebagai tahap akhir, model yang telah dilatih sistem aplikasi web interaktif menggunakan Streamlit, sebuah framework modern berbasis Python yang mempercepat proses deployment. Hasil penelitian menunjukkan bahwa model Decision Tree mampu mengklasifikasikan kategori harga rumah dengan tingkat akurasi sebesar 81,74%. Implementasi menggunakan Streamlit terbukti berhasil menyediakan antarmuka yang efektif, intuitif, dan praktis, memungkinkan pengguna non-teknis untuk berinteraksi langsung dengan model prediktif.

Kata Kunci— CRISP-DM, Decision Tree, Machine Learning, Prediksi Harga Rumah, Streamlit.

I. PENDAHULUAN

A. Latar Belakang

Industri properti merupakan sektor yang kompleks dan dinamis karena nilai suatu properti dipengaruhi oleh berbagai faktor seperti permintaan pasar, luas bangunan, kualitas material, aksesibilitas, serta lokasi strategis[1]. Penerapan teknik *Machine Learning* terbukti meningkatkan akurasi estimasi harga properti secara signifikan dibandingkan metode statistik konvensional[2]. Berbagai penelitian telah memanfaatkan teknik *Machine Learning* untuk memperkirakan harga rumah berdasarkan atribut fisik dan lokasi[3]. Ketidaktepatan dalam estimasi harga dapat menimbulkan kerugian bagi pembeli maupun penjual, sehingga diperlukan alat prediksi yang akurat untuk mendukung keputusan transaksi properti[4].

Beragam studi sebelumnya menunjukkan bahwa model prediktif seperti *Decision Tree* dapat mengolah data transaksi historis dan menemukan pola yang sulit diidentifikasi secara manual[5], serta dukungan XAI dan data terbuka turut meningkatkan transparansi sistem penilaian harga[6]. Model *Decision Tree* sangat efektif dalam memvisualisasikan kriteria pengambilan keputusan harga, menjadikannya pilihan utama untuk tujuan interpretasi[7]. Namun, sebagian besar penelitian ini masih terbatas pada pendekatan regresi dan hanya menggunakan variabel dasar tanpa mempertimbangkan faktor eksternal seperti Tahun Pembangunan (*Year Built*). Selain itu, model umumnya menghasilkan nilai numerik, sementara penggunaan praktis seringkali membutuhkan kategori harga yang lebih mudah dipahami oleh para

pemangku kepentingan. Implementasi model pada aplikasi web interaktif juga belum banyak dilakukan, sehingga hasil prediksi sulit diakses oleh pengguna non-teknis[8].

Untuk mengatasi keterbatasan ini, penelitian ini menambahkan variabel *Year Built*, mengubah pendekatan dari regresi menjadi klasifikasi (Rendah, Sedang, Tinggi), dan mengintegrasikan model ke dalam aplikasi web berbasis Streamlit untuk kemudahan penggunaan dan transparansi. *Decision Tree* dipilih karena interpretabilitasnya yang tinggi, sehingga logika penentuan harga dapat dipahami tanpa sifat *black-box*[9]. Kebutuhan akan model klasifikasi harga properti yang akurat dan mudah diakses oleh masyarakat umum menjadi fokus utama dalam mengatasi gap antara hasil riset dan aplikasi praktis [10].

Dalam konteks Indonesia, urgensi sistem prediksi berbasis data semakin meningkat karena pasar properti menunjukkan variasi harga yang luas antar wilayah, kurangnya data harga terpusat, dan dominasi harga subjektif yang berkelanjutan. Hingga saat ini, tidak ada *platform machine learning* yang dapat diakses publik yang tersedia untuk menilai keadilan harga properti dengan cepat dan objektif. Hal ini menyulitkan pembeli untuk mendapatkan informasi yang akurat, dan penjual menetapkan harga tanpa referensi yang kuat[11]. Pengembangan model klasifikasi harga rumah yang mudah dipahami dan dapat diakses melalui aplikasi web sangat penting untuk meningkatkan transparansi dan mengurangi asimetri informasi dalam transaksi properti [10]. Inovasi dalam penyediaan *platform prediksi* harga properti berbasis web interaktif sangat dibutuhkan untuk mendukung ekosistem properti yang lebih transparan di Indonesia [12].

Penelitian ini menggunakan kerangka kerja CRISP-DM yang fleksibel untuk berbagai jenis data properti. Selain itu, Streamlit dipilih untuk mengatasi hambatan implementasi karena memungkinkan pembuatan aplikasi berbasis Python tanpa memerlukan keahlian HTML, CSS, atau JavaScript[13], sehingga pengembang dapat fokus pada pengembangan model *machine learning* secara efisien. Streamlit terbukti menjadi solusi *deployment* yang cepat dan efisien untuk model *Machine Learning*, terutama dalam lingkungan pengembangan Python[14].

B. Rumusan Masalah

Berdasarkan latar belakang yang diuraikan, penelitian ini merumuskan masalah sebagai berikut:

1. Bagaimana merancang dan menguji model klasifikasi kategori harga rumah menggunakan Algoritma Pohon Keputusan dengan kerangka kerja CRISP-DM untuk mencapai akurasi yang optimal?
2. Bagaimana membandingkan kinerja Algoritma Pohon Keputusan dengan algoritma klasifikasi lain (*Random Forest*, SVM, Regresi Logistik) dan memberikan justifikasi pemilihan model?
3. Bagaimana mengintegrasikan model Algoritma Pohon Keputusan terpilih ke dalam sistem aplikasi web interaktif menggunakan Streamlit untuk memfasilitasi pengguna non-teknis dalam

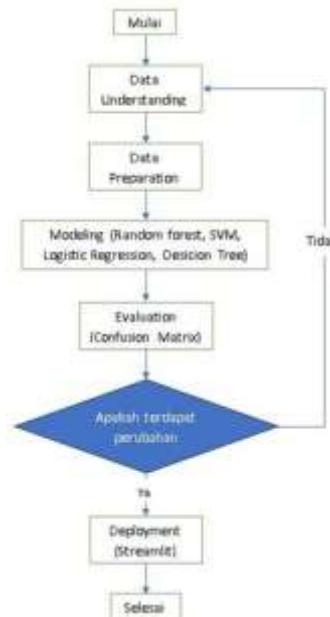
menggunakan prediksi?

C. Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah:

1. Mengembangkan model klasifikasi kategori harga rumah menggunakan Algoritma Pohon Keputusan.
2. Menganalisis dan membandingkan kinerja Algoritma Pohon Keputusan terhadap algoritma lain untuk menentukan model yang paling sesuai berdasarkan kriteria akurasi dan interpretabilitas.
3. Mengimplementasikan model klasifikasi terpilih ke dalam aplikasi web interaktif berbasis Streamlit sebagai solusi *deployment* yang efektif.

II. METODE PENELITIAN



Gambar 1. Alur Metode Penelitian

Pada Gambar 1 menunjukkan alur penelitian untuk membangun sistem prediksi kategori harga rumah menggunakan algoritma klasifikasi *Random Forest*, *Support Vector Machine* (SVM), *Logistic Regression*, dan *Decision Tree*, serta implementasinya melalui aplikasi Streamlit. Penelitian ini mengikuti kerangka kerja CRISP-DM, yang mencakup tahapan data *understanding*, data *preparation*, *modeling*, *evaluation*, dan *deployment*.

A. Data Understanding

Tahap ini bertujuan untuk memahami karakteristik dataset, yaitu Ames *Housing Dataset* dari Kaggle yang berisi 79 variabel properti seperti *Gr Liv Area*, *Overall Qual*, *Garage Cars*, *Year Built*, dan *Neighborhood*. Pada fase ini, nilai yang hilang, duplikat, dan pola distribusi diperiksa untuk memastikan kualitas data sebelum diproses[15].

B. Data Preparation

Pada tahap ini, dilakukan pembersihan data, pemilihan fitur, dan transformasi variabel. *SalePrice* dikategorikan menjadi Rendah, Sedang, dan Tinggi berdasarkan nilai kuantil, sehingga mengubah masalah regresi menjadi masalah klasifikasi[16]. Penggunaan kuantil memungkinkan kategorisasi harga rumah (rendah–sedang–tinggi) dilakukan secara objektif sesuai distribusi data, sehingga proporsi kelas lebih seimbang dan perbandingan antarkelompok menjadi lebih akurat. Metode ini juga menghindari subjektivitas dalam penentuan batas kategori, sebagaimana digunakan pada berbagai studi. Selain itu, disertakan pula tabel deskripsi atribut untuk memperjelas karakteristik data penelitian[17].

C. Pemodelan

Empat algoritma dikembangkan *Random Forest*, SVM, Regresi Logistik, dan *Decision Tree* dan dievaluasi menggunakan metrik akurasi, presisi, *recall*, dan *F1-score*. *Random Forest* memperoleh akurasi tertinggi (85,32%), namun *Decision Tree* dipilih sebagai model utama karena interpretabilitasnya yang tinggi dan kemudahan integrasinya ke Streamlit[18]. *Decision Tree* unggul dalam aspek interpretasi, yang sering kali lebih diprioritaskan daripada akurasi *absolut*, terutama dalam konteks pengambilan keputusan yang memerlukan penjelasan logis[19]. Penggunaan model yang memiliki interpretasi tinggi sangat penting untuk meningkatkan kepercayaan pengguna terhadap hasil prediksi harga properti.

D. Evaluation

Model tersebut divalidasi menggunakan *Confusion Matrix* dan metrik evaluasi lainnya. Setelah memenuhi kriteria kinerja, model diterapkan pada aplikasi web berbasis Streamlit, memungkinkan prediksi interaktif kategori harga rumah. Proporsi 80/20 digunakan untuk membagi data pelatihan dan pengujian guna mencegah kelebihan pas dan menjaga evaluasi yang objektif. Proporsi pembagian data 80:20 merupakan praktik standar dalam pengembangan model *Machine Learning* untuk mencapai keseimbangan antara pelatihan dan pengujian yang efektif[20].

Tabel 1. Tabel Atribut

| Atribut | Penjelasan |
|---------------|---|
| Neighborhood | Lokasi fisik dalam batas kota Ames (misal: <i>CollegeC</i> , <i>OldTown</i>). |
| Overall Qual | Penilaian kualitas material dan <i>finish</i> keseluruhan (skala 1-10). |
| Year Built | Tahun pembangunan asli. |
| Total Bsmt SF | Total luas <i>basement</i> (<i>square feet</i>). |
| Gr Liv Area | <i>Above grade (ground) living area</i> - Total luas ruang hidup di atas permukaan tanah (<i>square feet</i>). |

| | |
|------------------|---|
| Full Bath & Half | Jumlah kamar mandi penuh dan Bath setengah di atas permukaan tanah. |
| TotRms AbvGrd | Total ruangan di atas permukaan tanah (tidak termasuk kamar mandi). |
| Garage Cars | Kapasitas garasi (jumlah mobil). |

III. HASIL DAN PEMBAHASAN

A. Hasil

a) Implementasi Aplikasi Streamlit

Setelah mendapatkan model klasifikasi menggunakan algoritma *Decision Tree*, langkah selanjutnya adalah mengimplementasikan model tersebut ke dalam aplikasi interaktif menggunakan Streamlit.

Aplikasi ini dirancang untuk memudahkan pengguna memprediksi kategori harga rumah berdasarkan atribut yang relevan. Pengguna dapat memasukkan informasi rumah seperti kualitas material, luas bangunan, dan kapasitas garasi untuk mendapatkan hasil klasifikasi ke dalam tiga kategori harga: Rendah, Sedang, dan Tinggi. Antarmuka aplikasi utama dapat dilihat pada Gambar 2, yang menunjukkan antarmuka halaman prediksi beserta navigasi ke halaman analisis data dan perbandingan model.



Gambar 2. Tampilan Interface Aplikasi St

b) Pengujian

Pengujian dilakukan untuk memeriksa seberapa baik model *Decision tree* memprediksi harga rumah saat digunakan dalam aplikasi Streamlit. Setelah model dilatih, model tersebut digunakan untuk mengklasifikasikan harga rumah ke dalam kategori rendah, sedang, atau tinggi berdasarkan data masukan.

Pengujian ini melibatkan pembuatan aplikasi web menggunakan kerangka kerja Streamlit. Model yang sudah dilatih disimpan dalam file pickle (pkl) dan dimuat oleh *backend* Python. Aplikasi ini memiliki antarmuka yang mudah digunakan dengan bagian interaktif seperti *slider* dan kotak centang, memungkinkan pengguna memasukkan detail rumah mereka. Setelah data dimasukkan, aplikasi akan segera menampilkan hasil prediksi dalam format teks dan probabilitas.

Antarmuka untuk halaman prediksi ditunjukkan pada Gambar 3. Antarmuka ini mencakup elemen interaktif yang memungkinkan pengguna memasukkan beberapa fitur utama rumah. *Slider* digunakan untuk nilai numerik seperti Kualitas Keseluruhan, *GrLivArea*, *GarageCars*, dan Tahun Dibangun, sedangkan kotak centang digunakan untuk kategori seperti Lingkungan.

Pengaturan ini memudahkan pengguna memasukkan data dengan jelas dan efektif, membantu memastikan prediksi yang akurat.



Gambar 3. Tampilan Antarmuka Input Data

Setelah pengguna memasukkan semua detail yang diperlukan pada halaman *input*, aplikasi akan menampilkan hasil prediksi, seperti yang ditunjukkan pada Gambar 4. Hasil ini diperlihatkan dengan jelas melalui penanda visual yang menunjukkan kategori harga rumah, yaitu Rendah, Sedang, atau Tinggi berdasarkan keluaran dari model klasifikasi. Selain itu, nilai probabilitas untuk setiap kategori juga ditampilkan, memberikan gambaran tentang seberapa yakin model dalam prediksinya.

Di sisi kiri layar, terdapat menu interaktif yang memungkinkan pengguna beralih ke bagian lain seperti Analisis Data, Perbandingan Model, dan Tentang Aplikasi. Desain aplikasi responsif dan tertata rapi, sehingga memudahkan dan lebih alami bagi pengguna untuk memahami dan menggunakan hasil prediksi.

Di sisi kiri layar, terdapat menu interaktif yang memungkinkan pengguna beralih ke bagian lain seperti Analisis Data, Perbandingan Model, dan Tentang Aplikasi. Desain aplikasi responsif dan tertata dengan baik, sehingga lebih mudah dan alami bagi pengguna untuk memahami dan menggunakan hasil prediksi.



Gambar 4. tampilan hasil prediksi kategori

B. Pembahasan

Pada tahap pra-pemrosesan, kami memilih fitur-fitur yang paling memengaruhi harga rumah. Dari analisis awal, fitur yang digunakan dalam model meliputi Kualitas Keseluruhan, Luas Bangunan, Jumlah Mobil Garasi, Tahun Dibangun, dan Lingkungan Sekitar.

Variabel yang akan diprediksi, yaitu Harga Jual, dikategorikan menjadi tiga kelas: Rendah, Sedang, dan Tinggi. Kategori-kategori ini ditentukan berdasarkan distribusi data secara keseluruhan, dengan batas harga sebagai berikut:

Rendah: Harga di bawah \$129.500

Sedang: Harga antara \$129.500 dan \$180.750

Tinggi: Harga di atas \$180.750

Batas kategori ini bisa disesuaikan tergantung karakteristik

data yang digunakan.

C. Evaluasi Kinerja Model Klasifikasi

a) Perbandingan Akurasi Algoritma

Untuk menemukan model prediksi yang paling optimal, penelitian ini membandingkan kinerja empat algoritma klasifikasi yang berbeda. Model yang diuji termasuk *Random Forest*, *SVM* (*Support Vector Machine*), *Logistic Regression*, dan *Decision Tree*. Perbandingan akurasi setiap model disajikan dalam Tabel 2.

Tabel 2. Perbandingan Akurasi Model.

| Model | Akurasi | Presisi | Recall | F1-score |
|---------------------|---------|---------|--------|----------|
| Random Forest | 85,32% | 84,9% | 85,1% | 85% |
| SVM | 83,45% | 82,8% | 83,2% | 83% |
| Logistic Regression | 83,28% | 82,5% | 82,9% | 82,7% |
| Decision Tree | 81,74% | 80,6% | 81% | 80,8% |

Tabel 2 menunjukkan perbandingan kinerja empat algoritma klasifikasi, yaitu *Random Forest*, *Support Vector Machine* (*SVM*), *Logistic Regression*, dan *Decision Tree*, berdasarkan empat metrik evaluasi utama: akurasi, presisi, recall, dan *F1-score*. Berdasarkan hasil pengujian, model *Random Forest* mencapai skor akurasi tertinggi yaitu 85,32%, diikuti oleh *SVM* dengan 83,45%, *Logistic Regression* dengan 83,28%, dan *Decision Tree* dengan 81,74%.

b) Analisis Confusion Matrix

Untuk memahami distribusi kesalahan klasifikasi yang tidak terlihat hanya dari nilai akurasi, analisis mendalam dilakukan menggunakan *Confusion Matrix* untuk keempat model. Visualisasi ini menyoroti bagaimana setiap algoritma memprediksi kelas target (Rendah, Sedang, Tinggi) dibandingkan dengan data aktual.

1. Analisis Kesalahan Decision Tree (Model Utama)

Visualisasi *Confusion Matrix* untuk model *Decision Tree* dapat dilihat pada Gambar 5.



Gambar 5. Confusion Matrix Model Decision Tree

Berdasarkan Gambar 5, *Decision Tree* menunjukkan kinerja yang sangat baik dalam memprediksi kategori ekstrem. Model berhasil memprediksi 161 titik data kelas "Rendah" dan 202

titik data kelas "Tinggi" dengan benar. Namun, kelemahan utama dari model ini terletak pada klasifikasi kelas "Sedang". Jelas bahwa 36 titik data "Sedang" yang sebenarnya salah diprediksi sebagai "Rendah", dan 20 titik data salah diprediksi sebagai "Tinggi". Tingkat kesalahan yang tinggi dalam kategori menengah ini menunjukkan tumpang tindih fitur dalam batas harga yang sulit dipisahkan secara sempurna oleh satu *Decision Tree* tunggal.

2. Perbandingan dengan Random Forest.

Sebagai bandingan, performa model *Random Forest* disajikan pada Gambar 6.

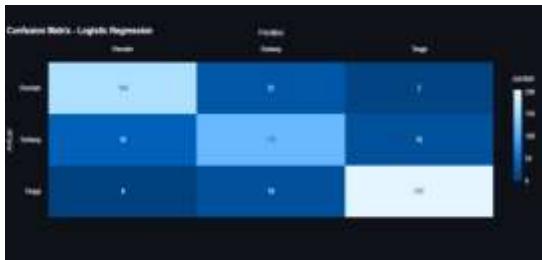


Gambar 6. Confusion Matrix Model Random Forest

Merujuk pada Gambar 6, *Random Forest* lebih baik dalam menangani ambiguitas pada kelas "Sedang". Model ini berhasil memprediksi 124 data kelas "Sedang" dengan benar, yang lebih tinggi dari 116 data yang diprediksi oleh *Decision tree*. Selain itu, *Random Forest* menunjukkan stabilitas tinggi pada kelas "Tinggi" dengan 208 prediksi benar dan 0 kesalahan klasifikasi ke kelas "Rendah". Hal ini wajar mengingat *Random Forest* bekerja berdasarkan prinsip ensemble, sehingga mampu mengurangi variansi pada data batas.

3. Pola Umum pada Model Lain (SVM dan Logistic Regression).

Pola klasifikasi untuk SVM dan *Logistic Regression* dapat dilihat masing-masing pada Gambar 7 dan Gambar 8.



Gambar 7. Confusion Matrix Model Logistic Regression



Gambar 8. Confusion Matrix Model SVM

Kedua model menunjukkan pola yang mirip dengan *Random Forest* dalam mengidentifikasi kelas "Tinggi", dengan SVM mencatat 210 prediksi yang benar dan Regresi Logistik 205 prediksi yang benar. Namun, keduanya juga menghadapi kesulitan yang sama seperti *Decision Tree* dalam membedakan batas kabur antara kelas "Rendah" dan "Sedang", meskipun tidak sesignifikan kesalahan pada *Decision Tree*.

Secara keseluruhan, visualisasi *Confusion Matrix* di atas mengonfirmasi bahwa tantangan utama dalam dataset ini adalah memisahkan kategori harga "Sedang" dari kategori lainnya. Meskipun *Decision Tree* memiliki tingkat kesalahan yang sedikit lebih tinggi dalam segmen ini, model tersebut masih mampu mempertahankan prediksi yang seimbang tanpa bias ekstrem terhadap satu kelas mayoritas, sehingga masih cocok untuk digunakan, terutama karena keunggulan interpretasinya.

D. Justifikasi Pemilihan Model Analisis Trade-off

Meskipun hasil evaluasi kuantitatif pada Tabel 2 dan analisis *Confusion Matrix* menunjukkan bahwa algoritma *Random Forest* memiliki keunggulan akurasi sekitar 3,5% dibandingkan dengan *Decision Tree*, penelitian ini menetapkan *Decision Tree* sebagai model akhir yang diimplementasikan. Keputusan ini didasarkan pada analisis *trade-off* antara akurasi prediksi dan kebutuhan operasional sistem, sebagai berikut:

1. Interpretabilitas Model (*Explainability*).

Dalam konteks pasar properti, pemangku kepentingan (pembeli dan penjual) tidak hanya memerlukan angka prediksi tetapi juga alasan di balik penentuan harga tersebut. *Random Forest* adalah model *black-box* yang kompleks karena menggabungkan ratusan *Decision tree*, sehingga sulit untuk melacak logika spesifik di balik satu prediksi. Sebaliknya, *Decision tree* menawarkan karakteristik *white-box* di mana struktur *Decision tree* dapat divisualisasikan sepenuhnya. Ini memungkinkan pengguna non-teknis untuk memahami aturan logika (misalnya: "Jika Luas Tanah > 200m² dan Lokasi = X, maka Harga = Tinggi") secara transparan dan akuntabel.

2. Efisiensi Komputasi dan Implementasi.

Dari perspektif pengembangan sistem, *Decision Tree* menawarkan kesederhanaan. Algoritma ini tidak memerlukan proses *tuning hyperparameter* yang ekstensif dan memakan waktu seperti *Random Forest* atau SVM. Selain itu, model *Decision Tree* yang dihasilkan memiliki ukuran file yang lebih ringan saat disimpan (format .pkl), yang berdampak pada waktu pemuatan yang lebih cepat saat aplikasi dijalankan di lingkungan produksi web.

3. Fleksibilitas Penanganan Data.

Dataset properti seringkali mengandung tipe data campuran antara numerik (luas, tahun dibangun) dan kategorikal (nama lingkungan, kondisi fisik). *Decision Tree* secara alami mampu menangani tipe data campuran ini tanpa memerlukan pra-pemrosesan yang rumit seperti normalisasi skala yang wajib dilakukan pada algoritma berbasis jarak seperti SVM atau *Logistic Regression*.

Properti dataset sering kali berisi tipe data campuran, termasuk numerik (luas, tahun dibangun) dan kategorikal (nama lingkungan, kondisi fisik). *Decision Tree* secara alami mampu menangani tipe data campuran ini tanpa memerlukan praproses yang kompleks seperti normalisasi skala wajib untuk algoritma berbasis jarak seperti SVM atau *Logistic Regression*.

Dengan mempertimbangkan faktor-faktor di atas, penurunan akurasi sebesar 3,58% dianggap sebagai biaya yang dapat diterima (*acceptable cost*) untuk mendapatkan manfaat interpretabilitas dan efisiensi sistem yang signifikan.

E. Implementasi dan Implikasi Praktis Sistem

Tahap terakhir dari metodologi CRISP-DM adalah *Deployment*. Dalam penelitian ini, model *Decision Tree* yang dipilih diintegrasikan ke dalam aplikasi web berbasis Streamlit.

1. Arsitektur sistem dibangun menggunakan bahasa pemrograman Python. Model yang telah dilatih disimpan dalam format serialisasi *pickle* dan dipanggil kembali oleh *backend* aplikasi. Streamlit dipilih sebagai kerangka kerja *frontend* karena kemampuannya untuk dengan cepat memfasilitasi pembuatan antarmuka data interaktif (*Rapid Prototyping*) tanpa memerlukan kode HTML/CSS/JavaScript terpisah, yang sering menjadi hambatan teknis bagi *data scientist*.
2. Mekanisme Interaksi Pengguna Aplikasi dirancang dengan alur kerja yang intuitif:
 - a) Input Data: Pengguna memasukkan parameter properti melalui elemen visual.
 - b) Pemrosesan Real-time: Saat tombol prediksi ditekan, sistem mengirim data input ke model.
 - c) Visualisasi Output: Hasil klasifikasi ditampilkan bersama dengan nilai probabilitas keyakinan model (*confidence score*), memberikan

pengguna gambaran mengenai seberapa yakin sistem terhadap prediksinya.

3. Dampak Praktis Implementasi ini berhasil menjembatani kesenjangan antara model analitik yang kompleks dan kebutuhan pengguna akhir.

IV. KESIMPULAN

Berdasarkan hasil penelitian dan pembahasan yang telah dilakukan, dapat ditarik beberapa kesimpulan utama terkait aspek ilmiah, dampak praktis, dan arah pengembangan selanjutnya: Pertama, penelitian ini menemukan bahwa algoritma *Decision Tree* efektif untuk mengklasifikasikan kategori harga rumah dengan tingkat akurasi 81,74%. Meskipun analisis komparatif menunjukkan bahwa *Random Forest* memiliki tingkat akurasi yang lebih tinggi (85,32%), *Decision Tree* tetap menjadi pilihan model yang lebih rasional dalam kasus ini karena struktur *white-box*-nya memiliki keunggulan. Hal ini dianggap lebih berharga daripada peningkatan marjinal dalam akurasi karena kemampuan model ini untuk menyajikan aturan keputusan yang jelas dan mudah dipahami, terutama dalam pengambilan keputusan strategis di mana alasan di balik prediksi harga sangat penting. Kedua, dari perspektif praktis, mengintegrasikan model ke dalam aplikasi web berbasis Streamlit berhasil mengurangi kesenjangan antara kebutuhan pengguna akhir dan kompleksitas analisis model. Dengan antarmuka yang mudah dipahami dan ramah pengguna, sistem ini memungkinkan pengguna yang kurang berpengalaman (seperti penjual rumah atau calon pembeli) untuk secara mandiri dan *real-time* mensimulasikan prediksi harga. Keberhasilan peluncuran ini menunjukkan bahwa teknologi *open-source* berbasis Python sumber terbuka dapat mempercepat hilirisasi produk data *Science* tanpa memerlukan infrastruktur pengembangan web yang kompleks. Ketiga, pada studi yang berikutnya disarankan untuk fokus pada peningkatan akurasi prediksi untuk kelas harga "Sedang", yang ditemukan memiliki tingkat klasifikasi yang salah tertinggi. Untuk menjaga transparansi, pengembangan di masa depan dapat mempertimbangkan penggunaan metode ensemble tambahan seperti *Gradient Boosting* (XGBoost atau LightGBM) yang dikombinasikan dengan metode interpretasi model seperti SHAP (*SHapley Addit exPlanations*). Untuk meningkatkan fitur dan akurasi model, disarankan untuk menambahkan lebih banyak variabel eksternal yang relevan dengan konteks lokal, seperti indeks kriminalitas atau jarak ke pusat transportasi.

DAFTAR PUSTAKA

- [1] A. Begum, N. J. Kheya, and M. Z. Rahman, "Housing Price Prediction with Machine Learning," *Int. J. Innov. Technol. Explor. Eng.*, vol. 11, no. 3, pp. 42–46, 2022, doi: 10.35940/ijitee.c9741.0111322.
- [2] S. Yin, Z. Li, D. Song, H. Mu, Y. Niu, and X. Wang, "applied sciences Study on the Energy Evolution Law and Bursting Liability of Coal Failure with Different Joint Inclination Angles," 2024.
- [3] X. Li, *Comparing Linear Regression and Decision Trees for Housing Price Prediction*, no. Dai 2023. Atlantis Press International BV, 2024. doi: 10.2991/978-94-6463-370-2.
- [4] N. Zulkifley, S. Rahman, U. Nor Hasbiah, and I. Ibrahim, "House Price Prediction using a Machine Learning Model: A Survey of Literature," *Int. J. Mod. Educ. Comput. Sci.*, vol. 12, pp. 46–54, 2020, doi: 10.5815/ijmecs.2020.06.04.
- [5] R. A. Saputra, A. Pratama, S. Informasi, U. Nahdlatul, and U. Kalimantan, "IMPLEMENTASI DECISION TREE UNTUK PREDIKSI HARGA RUMAH DI DAERAH TEBET Sistem Informasi STMIK Pontianak Abstraksi Pendahuluan Tinjauan Pustaka," vol. 6, no. 2, pp. 4–10, 2025.
- [6] F. Trindade Neves, M. Aparicio, and M. de Castro Neto, "The Impacts of Open Data and eXplainable AI on Real Estate Price Predictions in Smart Cities," *Appl. Sci.*, vol. 14, no. 5, 2024, doi: 10.3390/app14052209.
- [7] M. Delmas, "Should my automated car drive as I do ? Investigating speed preferences of drivengers in various driving conditions," pp. 1–15, 2023, doi: 10.1371/journal.pone.0281702.
- [8] S. Muchammad, R. Aridho, A. Hasna, K. Aswha, T. D. Wahyuni, and A. A. Sari, "Analisis Faktor yang Mempengaruhi Harga Rumah Menggunakan Decision Tree," pp. 386–392, 2025.
- [9] L. Dipietro *et al.*, "The evolution of Big Data in neuroscience and neurology," *J. Big Data*, vol. 10, no. 1, pp. 1–53, 2023, doi: 10.1186/s40537-023-00751-2.
- [10] G. Bologna, "Transferring CNN Features Maps to Ensembles of Explainable Neural Networks," 2023.
- [11] L. Calcagni, A. Ruggiero, and A. Battisti, "Resilient Waterfront Futures : Mapping Vulnerabilities and Designing Floating Urban Models for Flood Adaptation on the Tiber Delta," pp. 1–22, 2025.
- [12] S. Customizable and O. Source, "A High-Resolution LED Stimulator for Steady-State Visual Stimulation: Customizable, Affordable, and Open Source," pp. 1–15, 2024.
- [13] C. Chitteti, P. Kunapareddy, M. Dharani, B. HimaKeerthi, C. Sravani, and G. Chilukuri, "Crime Prediction using Machine Learning Algorithms," *15th Int. Conf. Adv. Comput. Control. Telecommun. Technol. ACT 2024*, vol. 2, no. 3, pp. 1459–1464, 2024, doi: 10.48175/ijarsct-3620.
- [14] F. N. Hakim, A. Solechan, P. Studi, S. Informasi, and S. Provisi, "Rancang Bangun Sistem Informasi Administrasi Surat Menyurat Kasubbag Tingkat Kecamatan Berbasis Web," vol. 6, no. 1, pp. 2–7, 2020.
- [15] M. Kim and T. Le, "Influence Analysis of Real Exchange Rate Fluctuations on Trade Balance Data Using Feature Important Evaluation Methods," pp. 1–15, 2024.
- [16] J. Li, T. Gao, B. He, W. Zheng, and F. Lin, "applied sciences Power Allocation and User Grouping for NOMA Downlink Systems," 2023.
- [17] S. G. Lee, E. H. Kim, and B. C. Ma, "applied sciences Monitoring Chemical Accidents in Industrial Complexes Using Tower-Installed Infrared System for Remote Chemical Detection and Long-Range Video Surveillance System," 2023.
- [18] "1 , 2, 3," vol. 11, no. 2, 2018.
- [19] T. Rinnert, J. Walsh, C. Fleury, G. Coppin, T. Duval, and B. H. Thomas, "applied sciences AR Presentation of Team Members ' Performance and Inner Status to Their Leader : A Comparative Study," 2024.
- [20] M. Alazab, "Industry 4 . 0 Innovation : A Systematic Literature Review on the Role of Blockchain Technology in Creating Smart and Sustainable Manufacturing Facilities," 2024.