 22.11.5112_Muhammad Rilley_PDM_FINAL.pdf • 31 July 2025
6899 words (56029 characters)

Mostly Human 18%

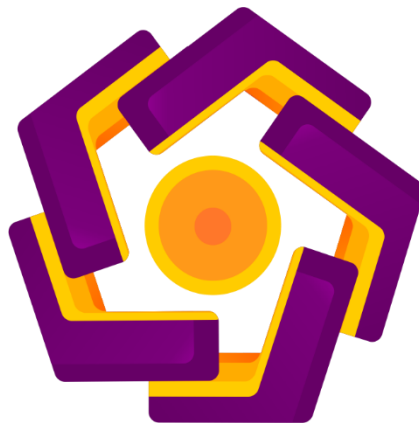
Almost all of the text is written by humans with some hints of AI assistance.

AI weightage	Content weightage	Sentences
<div>H</div> Highly AI written	12% Content	38
<div>M</div> Moderately AI written	17% Content	53
<div>L</div> Lowly AI written	12% Content	37

**SISTEM PREDIKSI HARGA RUMAH DI SEMARANG
BERBASIS WEB MENGGUNAKAN ENSEMBLE VOTING
DARI RANDOM FOREST, GRADIENT BOOSTING,
XGBOOST, LIGHTGBM, DAN CATBOOST**

SKRIPSI

untuk memenuhi salah satu syarat mencapai derajat Sarjana
Program Studi Informatika



disusun oleh

MUHAMMAD RILLY AYIDAN SUBAKTI

22.11.5112

Kepada

**FAKULTAS ILMU KOMPUTER
UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA**

2025

DAFTAR ISI

HALAMAN JUDUL	i
DAFTAR ISI.....	ii
DAFTAR GAMBAR	v
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian	4
1.5.1 Manfaat Teoritis	5
1.5.2 Manfaat Praktis	5
1.6 Sistematika Penulisan	5
BAB II TINJAUAN PUSTAKA	6
2.1 Studi Literatur	6
2.2 Dasar Teori.....	10
2.2.1 Sistem Informasi	10
2.2.2 Machine Learning	10
2.2.3 Model Ensemble Voting	11
2.2.4 Random Forest	11
2.2.5 Gradient Boosting Regressor	12

2.2.6	XGBoost.....	13
2.2.7	LightGBM.....	13
2.2.8	CatBoost.....	14
2.2.9	Web Development (Frontend dan Backend).....	14
2.2.10	FastAPI.....	15
2.2.11	React.js.....	15
BAB III METODOLOGI PENELITIAN		17
3.1	Objek Penelitian.....	17
3.2	Alur Penelitian	17
3.3	Alat dan Bahan.....	21
3.3.1	Data Penelitian	21
3.3.2	Perangkat Pengembangan Sistem	22
3.3.3	Alat Pengumpulan Data	24
3.3.4	Alat Analisis Data	25
BAB IV HASIL DAN PEMBAHASAN		26
4.1	Dataset dan Eksplorasi Awal	26
4.2	Proses Preprocessing dan Transformasi Data	27
4.2.1	Pembersihan Data	27
4.2.2	Encoding dan Normalisasi	27
4.2.3	Pembagian Data	27
4.3	Hasil Pelatihan dan Evaluasi Model Machine Learning.....	27
4.3.1	Hasil R ² Score	28

4.3.2	Hasil MAE dan MAPE	28
4.4	Analisis Model Ensemble Voting	30
4.4.1	Perbandingan Model Individu vs Ensemble	30
4.4.2	Alasan Pemilihan Ensemble Voting	31
4.4.3	Implikasi terhadap Sistem.....	31
4.5	Implementasi Sistem SmartHomeValuator.....	32
4.5.1	Arsitektur Sistem.....	32
4.5.2	Alur Kerja Sistem.....	33
4.5.3	Deployment dan Akses Sistem	34
4.6	Uji Coba Sistem dan Tampilan Antarmuka	36
4.6.1	Skenario Uji Coba.....	36
4.6.2	Validasi Respons API	37
BAB V PENUTUP		39
5.1	Kesimpulan	39
5.2	Saran.....	40
REFERENSI		41

DAFTAR GAMBAR

Gambar 3. 1 Alur Penelitian	20
Gambar 4. 1 Perbandingan R^2 Score Model	28
Gambar 4. 2 Perbandingan Model MAPE	29
Gambar 4. 3 Alur Sistem.....	33
Gambar 4. 4 Form Input Fisik Properti.....	34
Gambar 4. 5 Form Input Kondisi Properti	35
Gambar 4. 6 Form Input Zona Properti.....	35
Gambar 4. 7 Hasil Prediksi	36
Gambar 4. 8 Respon JSON dari API /predict	38

BAB I

PENDAHULUAN

1.1 Latar Belakang

Sektor properti di Indonesia, khususnya di kota-kota besar seperti Semarang, mengalami pergerakan harga yang cukup dinamis dan tidak selalu stabil dari tahun ke tahun. Berdasarkan laporan Survei Harga Properti Residensial (SHPR) Tahun 2022 oleh Bank Indonesia, indeks harga properti residensial di wilayah Jawa Tengah, termasuk Semarang, mengalami kenaikan dan penurunan yang tidak konsisten akibat berbagai faktor seperti kenaikan harga bahan bangunan, suku bunga, dan masalah perizinan/birokrasi[1]. Sebagai ibu kota Provinsi Jawa Tengah, Semarang juga mengalami percepatan pembangunan infrastruktur dan urbanisasi yang cukup signifikan. Fenomena ini ikut mendorong perubahan drastis dalam nilai dan harga properti di wilayah tersebut. Namun, meskipun perkembangan terlihat pesat, banyak masyarakat di Semarang masih kesulitan dalam menentukan harga rumah yang wajar. Keterbatasan akses terhadap data properti dan kurangnya keahlian dalam melakukan penilaian menjadi hambatan utama. Dalam konteks ini, dibutuhkan solusi berbasis teknologi yang mampu memberikan estimasi harga rumah secara cepat, akurat, dan objektif sebagai alternatif modern yang dapat membantu masyarakat dalam mengambil keputusan yang lebih bijak.

mengambil keputusan yang lebih bijak.

Salah satu pendekatan yang kini terbukti efektif adalah pemanfaatan Machine Learning (ML) untuk prediksi harga properti. Berbagai algoritma seperti Random Forest, Gradient Boosting, CatBoost, XGBoost, hingga LightGBM telah digunakan secara luas dan menunjukkan performa yang andal dalam menganalisis data real estate. Sebagai contoh, sebuah studi sebelumnya menunjukkan bahwa algoritma Random Forest mampu menghasilkan akurasi hingga 88% dalam memprediksi harga rumah di Surabaya [2]. Hal ini menunjukkan potensi ML dalam meningkatkan efisiensi dan akurasi penilaian harga rumah. Hal ini

memperkuat potensi ML dalam meningkatkan efisiensi dan akurasi proses penilaian harga properti. Di samping itu, penelitian juga menegaskan bahwa fitur-fitur seperti luas bangunan, lokasi, dan jumlah kamar tidur memiliki pengaruh besar terhadap nilai jual sebuah rumah[3].

Meski begitu, setiap algoritma yang digunakan dalam penelitian ini memiliki keunggulan dan keterbatasannya masing-masing. Random Forest, misalnya, unggul dalam menangani dataset dengan banyak fitur dan outlier, tetapi cenderung kurang akurat jika dibandingkan dengan model boosting karena tidak secara eksplisit meminimalkan error residu. Gradient Boosting dikenal unggul dalam menangani data yang kompleks dan memiliki tingkat noise yang rendah. Namun, jika parameter yang digunakan tidak diatur dengan hati-hati, algoritma ini sangat rentan mengalami overfitting. XGBoost mengatasi beberapa keterbatasan Gradient Boosting dengan menambahkan regularisasi dan optimisasi berbasis second-order derivatives, tetapi pelatihannya membutuhkan tuning parameter yang sangat sensitif. LightGBM mempercepat proses pelatihan melalui teknik histogram dan leaf-wise growth, namun pendekatan ini dapat menyebabkan overfitting jika struktur pohon terlalu dalam atau dataset tidak cukup besar. Sementara itu, CatBoost dirancang khusus untuk menangani fitur kategorikal tanpa encoding manual, namun memiliki waktu pelatihan yang relatif lebih lambat dibanding model lain dan membutuhkan pengaturan struktur data yang lebih hati-hati.

Dengan mengombinasikan kelima algoritma populer tersebut, sistem dapat mengoptimalkan keunggulan masing-masing sekaligus meminimalkan kelemahannya. Pendekatan ini menghasilkan model yang lebih stabil, robust, dan akurat, khususnya dalam konteks prediksi harga properti yang dipengaruhi oleh banyak variabel dan noise. Pendekatan ensemble juga berperan dalam mengurangi ketergantungan terhadap satu model tunggal, yang berisiko menimbulkan bias terhadap pola tertentu dalam data.

Melihat potensi tersebut, penelitian ini bertujuan untuk membangun sebuah aplikasi berbasis web bernama SmartHomeValuator. Aplikasi ini

dirancang untuk membantu memprediksi harga rumah di wilayah Semarang dengan memanfaatkan pendekatan *machine learning* menggunakan model ensemble voting. Dalam pendekatan ini, lima algoritma populer yaitu *Random Forest*, *Gradient Boosting*, *CatBoost*, *XGBoost*, dan *LightGBM* digabungkan untuk menghasilkan prediksi yang lebih akurat dan stabil dibandingkan dengan penggunaan satu algoritma saja [4][5]. Tak hanya menyajikan hasil prediksi dalam bentuk angka, aplikasi ini juga mengelompokkan harga rumah ke dalam kategori seperti "Murah", "Sedang", dan "Mahal". Pengelompokan ini bertujuan untuk memudahkan pengguna dalam memahami dan menginterpretasikan hasil prediksi secara intuitif.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan, maka rumusan masalah dalam penelitian ini dapat dirumuskan sebagai berikut:

1. Bagaimana penerapan model ensemble voting yang menggabungkan lima algoritma machine learning (Random Forest, Gradient Boosting, XGBoost, LightGBM, dan CatBoost) dapat meningkatkan akurasi dan stabilitas prediksi harga rumah di Kota Semarang, jika dibandingkan dengan performa masing-masing model individu?
2. Bagaimana sistem prediksi harga rumah berbasis web dapat dirancang agar mampu membantu pengguna non-teknis dalam memahami estimasi harga melalui penyajian hasil dalam bentuk numerik dan kategorikal ("Murah", "Sedang", "Mahal") yang mudah diinterpretasikan?

1.3 Batasan Masalah

Masalah penelitian akan dibatasi sebagai berikut:

1. Wilayah prediksi dibatasi pada area Kota Semarang sebagai fokus data dan konteks implementasi, sehingga hasil tidak digeneralisasikan ke wilayah lain.
2. Penelitian ini membatasi penggunaan model machine learning pada lima algoritma utama: Random Forest, Gradient Boosting Regressor, XGBoost,

LightGBM, dan CatBoost. Seluruh model diuji baik secara individual maupun dalam bentuk gabungan ensemble voting regresi.

3. Evaluasi performa model hanya dilakukan menggunakan metrik regresi, yaitu R^2 (coefficient of determination), MAE (Mean Absolute Error), dan MAPE (Mean Absolute Percentage Error), tanpa mempertimbangkan metrik klasifikasi atau time series.
4. Hasil prediksi sistem disajikan dalam dua bentuk output, yaitu estimasi harga numerik (dalam satuan Rupiah) dan label kategorikal berdasarkan segmentasi harga ("Murah", "Sedang", "Mahal").
5. Sistem diimplementasikan sebagai aplikasi web dengan arsitektur client-server, menggunakan React.js untuk frontend dan FastAPI untuk backend. Pengguna yang menjadi sasaran sistem ini adalah masyarakat umum, bukan profesional di bidang penilaian properti.

1.4 Tujuan Penelitian

Tujuan yang ingin didapatkan dari penelitian ini adalah:

1. Menerapkan serta mengevaluasi model *ensemble voting* yang menggabungkan lima algoritma *machine learning* yaitu Random Forest, Gradient Boosting, XGBoost, LightGBM, dan CatBoost guna meningkatkan akurasi dan stabilitas prediksi harga rumah di Kota Semarang, jika dibandingkan dengan performa masing-masing algoritma secara individu.
2. Merancang dan membangun sistem prediksi harga rumah berbasis web yang dapat menampilkan hasil estimasi dalam bentuk numerik dan kategorikal ("Murah", "Sedang", "Mahal"), guna membantu pengguna non-teknis dalam memahami nilai properti secara instan dan mudah.

1.5 Manfaat Penelitian

Penelitian ini diharapkan dapat memberikan kontribusi baik secara teoritis maupun praktis dalam bidang teknologi informasi, khususnya dalam penerapan

machine learning untuk prediksi harga properti di Kota Semarang. Melalui pengembangan sistem prediksi berbasis web, penelitian ini bertujuan untuk menjawab kebutuhan masyarakat terhadap layanan estimasi harga rumah yang cepat, akurat, dan mudah diakses.

1.5.1 Manfaat Teoritis

Penelitian ini memberikan kontribusi pada pengembangan literatur dalam bidang ilmu komputer, khususnya dalam penerapan metode ensemble voting yang menggabungkan lima algoritma machine learning untuk prediksi harga rumah.

1.5.2 Manfaat Praktis

Secara praktis, sistem SmartHomeValuator dapat membantu masyarakat umum, agen properti, maupun calon pembeli rumah dalam memperoleh estimasi harga rumah secara objektif dan instan. Penyajian hasil dalam bentuk angka dan kategori seperti "Murah", "Sedang", atau "Mahal" juga mempermudah pengguna dalam memahami nilai properti tanpa perlu pengetahuan teknis.

1.6 Sistematika Penulisan

Penelitian ini disusun dalam beberapa bab yang masing-masing memiliki uraian sebagai berikut:

BAB I PENDAHULUAN

Berisi latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat, dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Berisi kajian literatur dan teori-teori yang relevan, serta tabel keaslian penelitian.

BAB III METODOLOGI PENELITIAN

Menjelaskan objek penelitian, alur penelitian, serta alat dan bahan yang digunakan.

BAB IV HASIL DAN PEMBAHASAN

Menjelaskan implementasi aplikasi, uji coba sistem, dan evaluasi hasil.

BAB V PENUTUP

Berisi kesimpulan dari penelitian dan saran untuk pengembangan selanjutnya.

BAB II TINJAUAN PUSTAKA

2.1 Studi Literatur

Prediksi harga rumah merupakan salah satu topik yang banyak diteliti seiring dengan meningkatnya kebutuhan akan sistem estimasi nilai properti yang akurat dan efisien. Berbagai penelitian terdahulu telah mengeksplorasi penerapan algoritma machine learning dalam konteks ini.

Penelitian oleh Li membandingkan performa Random Forest dan XGBoost pada dataset dari Kaggle dan menunjukkan bahwa XGBoost memiliki nilai koefisien determinasi (R^2) sebesar 89%, lebih tinggi dibandingkan Random Forest [4].

Fang dalam studinya mengevaluasi berbagai model machine learning, termasuk Random Forest, dengan menekankan pentingnya preprocessing data dan tuning hyperparameter dalam meningkatkan akurasi prediksi [5]. Sementara itu, penelitian oleh Wu menerapkan Random Forest untuk prediksi harga rumah di Shanghai dan menyimpulkan bahwa fitur-fitur seperti luas bangunan, lokasi, dan jumlah kamar tidur merupakan variabel yang paling berpengaruh [3].

Di Indonesia, Ningsih et al. membandingkan algoritma Random Forest dan Multiple Linear Regression dalam memprediksi harga rumah di kawasan Jabodetabek. Hasilnya menunjukkan bahwa Random Forest memberikan performa lebih unggul dengan nilai R^2 sebesar 0.864 [6].

Dari berbagai studi yang telah dilakukan, dapat disimpulkan bahwa pendekatan *ensemble* seperti Random Forest, XGBoost, dan algoritma serupa terbukti mampu menghasilkan akurasi tinggi dalam prediksi harga rumah. Namun demikian, sebagian besar penelitian tersebut masih berfokus pada aspek eksplorasi algoritma semata, tanpa melanjutkan ke tahap implementasi dalam bentuk sistem berbasis web yang dapat langsung dimanfaatkan oleh pengguna. Oleh karena itu, penelitian ini menghadirkan kontribusi baru dengan menggabungkan model ensemble voting dari lima algoritma ke dalam aplikasi prediksi harga rumah yang

dapat digunakan secara langsung oleh masyarakat, khususnya di wilayah Semarang.

Tabel 2.1 Keaslian Penelitian

No	Judul penelitian	Nama Penulis	Tahun Publikasi	Hasil Penelitian	Perbandingan Penelitian
1	House price prediction model using random forest in Surabaya City	R. Tanamal et al.	2023	Menggunakan 17 fitur dari agen properti di Surabaya; akurasi 88% dengan Random Forest.	Menggunakan data lokal Surabaya; SmartHomeValuator menggunakan data properti dari Rumah123. SmartHomeValuator menggunakan Ensemble Voting dari 5 jenis Algoritma Model.
2	House price prediction and analysis based on Random Forest and XGBoost	Han Li	2023	Membandingkan Random Forest dan XGBoost pada dataset Kaggle; XGBoost memiliki R2 89%.	Fokus pada perbandingan algoritma; SmartHomeValuator fokus pada 5 Algoritma yang digabungkan.

3	Machine learning models for house price prediction	Lingjie Fang	2023	Evaluasi berbagai model ML termasuk Random Forest pada dataset benchmark; fokus pada preprocessing dan tuning.	Studi komparatif multi-model sedangkan SmartHomeValuato fokus pada 5 model dengan dataset spesifik.
4	Shanghai house price prediction using Random Forest	Shenghan Wu	2024	Prediksi harga rumah di Shanghai menggunakan Random Forest; fokus pada fitur karakteristik rumah.	Fokus pada data lokal Shanghai; SmartHomeValuator menggunakan data lokal dari Rumah123. SmartHomeValuator focus 5 Algoritma Model.
5	House price prediction analysis using a comparison of machine learning algorithms in Jabodetabek	I. R. Ningsih et al.	2025	Analisis perbandingan Multiple Linear Regression dan Random Forest di Jabodetabek; Random Forest lebih unggul dengan R2 0.864.	Fokus pada data lokal Jabodetabek; SmartHomeValuator menggunakan data Semarang.

2.2 Dasar Teori

Untuk memahami pendekatan yang digunakan dalam penelitian ini, berikut adalah beberapa teori dan konsep dasar yang digunakan sebagai landasan:

2.2.1 Sistem Informasi

Sistem informasi adalah sekumpulan komponen yang saling berkaitan untuk mengumpulkan, memproses, menyimpan, dan mendistribusikan informasi guna mendukung pengambilan keputusan dan pengendalian dalam sebuah organisasi. Sistem informasi modern umumnya berbasis komputer dan dirancang untuk meningkatkan efisiensi operasional serta mendukung kualitas pengambilan keputusan. Dalam konteks penelitian ini, sistem prediksi harga rumah termasuk dalam kategori sistem informasi berbasis web yang memberikan nilai tambah dalam bentuk estimasi harga properti berdasarkan data input pengguna. Sistem ini beroperasi dengan cara mengumpulkan input berupa fitur-fitur rumah dari pengguna, kemudian mengirimkan data tersebut ke backend untuk diproses menggunakan algoritma *machine learning*. Hasil prediksi kemudian disajikan kembali kepada pengguna dalam bentuk yang mudah dipahami.

2.2.2 Machine Learning

Machine Learning (ML) merupakan cabang dari kecerdasan buatan (*Artificial Intelligence*) yang memungkinkan sistem untuk belajar dari data dan membuat prediksi tanpa perlu diprogram secara eksplisit. Salah satu pendekatan yang umum digunakan dalam ML adalah *supervised learning*, di mana sistem dilatih menggunakan dataset berlabel untuk mempelajari pola atau hubungan antara input dan output.

Dalam konteks prediksi harga rumah, ML dimanfaatkan untuk membangun model prediktif berdasarkan data historis harga properti. Algoritma seperti Random Forest, Gradient Boosting, CatBoost, XGBoost, dan LightGBM sering digunakan karena kemampuannya dalam menangani data yang bersifat non-linear, mengelola banyak fitur sekaligus, serta meningkatkan akurasi melalui pendekatan *ensemble*. Model *ensemble* bekerja dengan menggabungkan beberapa

model dasar (*base learners*) untuk menghasilkan prediksi yang lebih stabil, robust, dan akurat dibandingkan dengan penggunaan satu model tunggal.

2.2.3 Model Ensemble Voting

Ensemble learning adalah teknik dalam machine learning yang menggabungkan beberapa model prediksi untuk meningkatkan akurasi dan stabilitas hasil dibandingkan dengan model individu[7]. Salah satu pendekatan *ensemble* yang sering digunakan adalah ensemble voting, yaitu metode di mana prediksi dari beberapa model digabungkan untuk menghasilkan keputusan akhir yang lebih andal.

Dalam kasus *regression*, pendekatan ini dikenal sebagai voting regresi, di mana hasil akhir diperoleh dengan menghitung rata-rata dari prediksi masing-masing model. Teknik ini memungkinkan sistem untuk memanfaatkan keunggulan dari setiap model yang digunakan, sekaligus mengurangi dampak kelemahan dari masing-masing model secara individu. Hasilnya adalah model gabungan yang lebih *robust* dan memiliki kemampuan generalisasi yang lebih baik terhadap data baru..

Ensemble voting dapat dibangun menggunakan model dasar yang bersifat homogen (sejenis) maupun heterogen (beragam). Keberhasilan pendekatan ini sangat ditentukan oleh tingkat keragaman antar model dasar serta strategi penggabungan prediksinya. Semakin beragam dan saling melengkapi model-model tersebut, semakin besar pula potensi peningkatan akurasi dan stabilitas yang dapat dicapai.

2.2.4 Random Forest

Random Forest merupakan salah satu algoritma *ensemble* berbasis *decision tree* yang diperkenalkan oleh Leo Breiman pada tahun 2001 [8]. Algoritma ini bekerja dengan membangun sejumlah *decision tree* secara acak pada subset data dan fitur yang berbeda. Setiap pohon menghasilkan prediksi masing-masing, kemudian seluruh hasil tersebut digabungkan misalnya dengan rata-rata (untuk regresi) atau voting mayoritas (untuk klasifikasi) guna

menghasilkan keputusan akhir yang lebih akurat dan stabil. Pendekatan ini efektif dalam mengurangi overfitting yang sering terjadi pada single decision tree, serta meningkatkan performa prediksi dengan memanfaatkan variasi antar pohon. Pada tugas regresi, hasil akhir diperoleh dari rata-rata nilai prediksi semua pohon.

Proses pelatihan Random Forest melibatkan teknik bagging (bootstrap aggregating), yaitu mengambil sampel acak dengan pengembalian dari data pelatihan untuk setiap pohon, serta memilih subset fitur secara acak pada setiap node untuk proses pemisahan. Hal ini menghasilkan variasi antarpohon dan mengurangi risiko overfitting dibandingkan dengan single decision tree.

Keunggulan utama Random Forest meliputi ketahanannya terhadap overfitting, kemampuannya menangani data dengan banyak fitur dan tipe campuran, serta kestabilan prediksi pada data yang kompleks. Random Forest populer dalam prediksi harga rumah karena kemampuannya menangkap hubungan non-linear antara fitur properti dan harga, dengan performa yang relatif stabil dan akurat.

2.2.5 Gradient Boosting Regressor

Gradient Boosting Regressor merupakan algoritma *boosting* yang diperkenalkan oleh Jerome Friedman pada tahun 2001 sebagai metode prediksi berbasis *ensemble* dari pohon keputusan (*decision tree*) [9]. Tidak seperti bagging pada Random Forest yang membangun pohon secara paralel, gradient boosting membangun pohon secara berurutan di mana setiap pohon baru berusaha meminimalkan kesalahan dari model sebelumnya menggunakan teknik gradient descent terhadap fungsi loss.

Dalam konteks regresi, fungsi kerugian (*loss function*) yang paling umum digunakan adalah Mean Squared Error (MSE). Pada algoritma Gradient Boosting, setiap pohon yang dibangun di tahap selanjutnya dilatih untuk memprediksi residual error atau selisih antara nilai aktual dan prediksi dari model sebelumnya. Dengan kata lain, model secara bertahap “belajar” dari kesalahan-kesalahan sebelumnya dan melakukan penyesuaian untuk meningkatkan akurasi prediksi di

setiap iterasi. Pendekatan bertingkat ini memungkinkan Gradient Boosting membangun model yang semakin presisi, terutama ketika menangani data dengan hubungan yang kompleks dan non-linear.

2.2.6 XGBoost

XGBoost (Extreme Gradient Boosting) merupakan pengembangan dari algoritma *gradient boosting* tradisional yang dirancang dengan fokus pada efisiensi komputasi, kecepatan pelatihan, dan performa prediktif yang tinggi. Algoritma ini diperkenalkan oleh Tianqi Chen dan Carlos Guestrin pada tahun 2016, dan sejak itu menjadi salah satu algoritma paling populer dalam kompetisi *machine learning* seperti Kaggle, berkat kemampuannya dalam menghasilkan model dengan akurasi tinggi serta waktu pelatihan yang relatif singkat [10].

XGBoost mengintegrasikan sejumlah teknik optimasi untuk meningkatkan kinerjanya, antara lain penggunaan turunan orde kedua (second-order derivatives) dalam proses pelatihan, regularisasi L1 dan L2 untuk mencegah *overfitting*, serta pemrosesan paralel guna mempercepat proses pembelajaran model. Selain itu, algoritma ini juga mendukung fitur-fitur lanjutan seperti *early stopping* dan penanganan *missing values* secara otomatis.

Dalam konteks regresi, XGBoost tetap mengusung prinsip dasar *boosting*, yaitu membangun model secara bertahap, di mana setiap model baru dirancang untuk memperbaiki kesalahan yang dihasilkan oleh model sebelumnya. Kombinasi teknik regularisasi dan optimasi yang efisien membuat XGBoost sangat andal pada data kompleks dan berskala besar.

2.2.7 LightGBM

LightGBM (Light Gradient Boosting Machine) adalah algoritma *boosting* yang dikembangkan oleh Microsoft pada tahun 2017 sebagai versi yang lebih efisien dan skalabel dari XGBoost, tanpa mengurangi akurasi model. LightGBM menggunakan pendekatan *histogram-based learning* untuk mengurangi kompleksitas pelatihan dan penggunaan memori, serta menerapkan strategi *leaf-*

wise tree growth alih-alih level-wise seperti pada gradient boosting konvensional [11].

Leaf-wise tree growth memungkinkan LightGBM untuk memperluas cabang pohon dengan penurunan loss tertinggi terlebih dahulu, yang sering kali menghasilkan akurasi lebih baik. Namun, ini juga memiliki risiko overfitting yang lebih tinggi jika tidak dikontrol dengan parameter seperti `max_depth`.

Keunggulan LightGBM terletak pada kecepatan pelatihan yang sangat tinggi, efisiensi dalam menangani dataset besar, serta dukungan terhadap fitur kategorikal secara langsung. Algoritma ini mendukung paralelisasi dan pembelajaran terdistribusi, sehingga cocok untuk penerapan skala industri yang besar.

2.2.8 CatBoost

CatBoost merupakan algoritma boosting dari Yandex (2018) yang dirancang untuk mengolah data kategorikal secara langsung tanpa encoding manual. Hal ini dilakukan melalui teknik internal yang disebut ordered boosting dan target statistics, yang membantu mencegah data leakage saat memproses fitur kategorikal.

CatBoost, seperti algoritma boosting lainnya, membangun model secara bertahap dengan melatih pohon baru untuk memperbaiki kesalahan dari sebelumnya. Uniknya, ia menggunakan struktur pohon simetris yang membuat prediksi lebih cepat dan stabil. Keunggulannya terletak pada kemampuannya menjaga akurasi tinggi sambil meminimalkan overfitting, terutama saat menangani banyak fitur kategorikal. Selain itu, CatBoost dapat menangani missing values dan tidak memerlukan preprocessing data yang rumit..

2.2.9 Web Development (Frontend dan Backend)

Pengembangan web (web development) merupakan proses pembangunan dan pemeliharaan aplikasi atau layanan berbasis internet yang dapat diakses melalui browser. Dalam arsitektur web modern, pengembangan aplikasi dibagi

menjadi dua komponen utama: frontend (bagian antarmuka pengguna) dan backend (bagian logika dan manajemen data di sisi server).

Frontend merupakan bagian antarmuka pengguna yang menampilkan konten halaman serta menangani interaksi langsung dengan pengguna, seperti input data dan navigasi. Teknologi yang digunakan dalam pengembangan frontend antara lain HTML, CSS, dan JavaScript beserta pustaka atau framework modern seperti React.js. React.js memungkinkan pembuatan antarmuka yang modular dan responsif, serta secara otomatis memperbarui tampilan saat data berubah.

Backend adalah bagian aplikasi yang berjalan di server dan menangani logika bisnis, penyimpanan data, serta permintaan dari frontend. Dalam penelitian ini, backend dibangun dengan FastAPI untuk menangani prediksi, validasi data, dan integrasi model machine learning. Komunikasi antara frontend dan backend menggunakan REST API dengan format JSON.

2.2.10 FastAPI

FastAPI adalah framework Python modern untuk membangun RESTful API yang cepat, efisien, dan mudah diskalakan. Framework ini memanfaatkan sistem typing Python dan modul validasi seperti Pydantic. Keunggulan FastAPI antara lain adalah kecepatan tinggi, dukungan asynchronous programming, dokumentasi API otomatis melalui Swagger UI dan ReDoc, serta kemampuan integrasi yang baik dengan model machine learning.

Dalam penelitian ini, FastAPI digunakan untuk membangun backend sistem SmartHomeValuator. Backend ini menyediakan endpoint `/predict` yang menerima input data dari frontend, melakukan validasi menggunakan model Pydantic, dan meneruskan data tersebut ke model machine learning yang telah di-load dari file `.pkl`. Hasil prediksi dikirim ulang ke frontend dalam bentuk JSON.

2.2.11 React.js

React.js adalah pustaka JavaScript buatan Facebook untuk membangun UI yang dinamis dan efisien. React memakai pendekatan deklaratif dan berbasis

komponen, sehingga tiap bagian antarmuka bisa dipisah menjadi komponen yang dapat digunakan kembali. Keunggulan utamanya terletak pada penggunaan Virtual DOM, yang membuat pembaruan tampilan jadi lebih cepat tanpa me-refresh seluruh halaman.

Dalam aplikasi SmartHomeValuator, React digunakan untuk membangun tampilan form input properti rumah, memproses input pengguna, dan menampilkan hasil prediksi yang dikirimkan oleh backend. React memungkinkan pengalaman pengguna yang lebih interaktif dan responsif. Selain itu, React dapat dikombinasikan dengan pustaka seperti React Router untuk navigasi dan Tailwind CSS untuk styling yang konsisten dan modern.

BAB III

METODOLOGI PENELITIAN

3.1 Objek Penelitian

Objek dalam penelitian ini adalah sistem prediksi harga rumah berbasis web bernama SmartHomeValuator yang dirancang untuk memberikan estimasi harga properti di wilayah Kota Semarang secara akurat dan informatif. Sistem ini menggunakan pendekatan machine learning dengan metode ensemble voting, yang menggabungkan lima algoritma populer: Random Forest, Gradient Boosting Regressor, XGBoost, LightGBM, dan CatBoost.

Sistem menerima input berupa data properti dengan atribut seperti lokasi, luas tanah, luas bangunan, jumlah kamar tidur, jumlah kamar mandi, jumlah lantai, daya listrik, jumlah carport, kondisi properti, dan kondisi perabotan. Dataset yang digunakan merupakan data hasil scraping dari situs properti daring, yang mencakup informasi seperti lokasi, luas bangunan, jumlah kamar, dan harga properti. Data ini kemudian dibersihkan dan diproses sebelum digunakan dalam pelatihan model.

Output dari sistem terdiri atas dua bentuk: nilai estimasi harga rumah dalam format numerik, serta klasifikasi harga ke dalam kategori seperti "Murah", "Sedang", atau "Mahal", untuk memudahkan interpretasi hasil oleh pengguna non-teknis. ("Murah", "Sedang", "Mahal").

3.2 Alur Penelitian

Alur penelitian ini terdiri dari beberapa tahapan berikut:

1. Studi Literatur

Peneliti melakukan penelusuran literatur untuk memahami penelitian-penelitian terdahulu yang relevan, khususnya terkait prediksi harga rumah, penerapan machine learning, dan pendekatan ensemble voting. Hasil kajian ini digunakan sebagai dasar untuk merumuskan masalah, menetapkan tujuan, serta menentukan metode yang tepat dalam pelaksanaan penelitian.

2. Pengumpulan Data

Data dalam penelitian ini diperoleh melalui web scraping dari situs properti daring Rumah123.com, mencakup atribut seperti harga, lokasi, luas bangunan dan tanah, jumlah kamar tidur dan kamar mandi, jumlah lantai, daya listrik, carport, kondisi properti, serta kondisi perabotan.

3. Persiapan Data

Data mentah hasil scraping kemudian disimpan dalam format terstruktur dan dilakukan pemeriksaan awal untuk mengidentifikasi nilai kosong, duplikasi, dan inkonsistensi. Data kemudian disaring untuk memastikan hanya entri yang valid dan relevan yang digunakan pada tahap analisis selanjutnya.

4. Data Preprocessing

Pada tahap ini, data ditransformasikan agar sesuai dengan kebutuhan pelatihan model machine learning. Proses preprocessing meliputi:

- Encoding fitur kategorikal menggunakan teknik one-hot encoding,
- Normalisasi atau standarisasi fitur numerik,
- Penghapusan nilai ekstrem (*outlier*) jika diperlukan,
- Pembagian data ke dalam data training dan testing.

5. Pelatihan dan Evaluasi Model Machine Learning

Lima algoritma machine learning dipilih untuk dikombinasikan dalam model ensemble voting, yaitu Random Forest, Gradient Boosting Regressor, XGBoost, LightGBM, dan CatBoost. Model dilatih menggunakan data pelatihan dan dievaluasi menggunakan metrik R^2 (coefficient of determination), MAE (Mean Absolute Error), dan MAPE (Mean Absolute Percentage Error). Model akhir disimpan dalam format .pkl untuk integrasi ke dalam sistem.

6. Pembangunan Backend

Backend sistem dibangun menggunakan framework FastAPI. Fungsi utamanya adalah menerima data input dari frontend, melakukan validasi data menggunakan Pydantic, menjalankan proses prediksi dengan model ML, dan mengembalikan hasil prediksi ke frontend melalui endpoint API.

7. Pembangunan Frontend

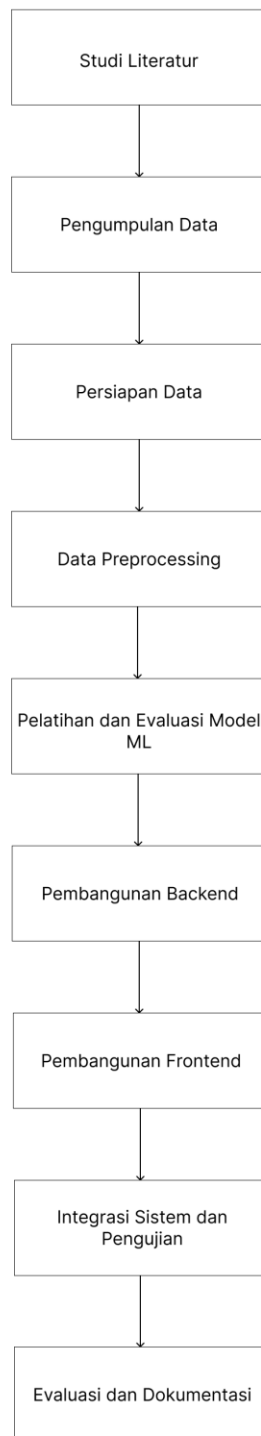
Frontend dikembangkan menggunakan React.js dan Tailwind CSS. Halaman antarmuka menyediakan form input data properti, serta menampilkan hasil prediksi dalam bentuk numerik dan kategorikal yang mudah dipahami oleh pengguna.

8. Integrasi Sistem dan Pengujian

Frontend dan backend dihubungkan melalui API agar sistem dapat berjalan secara terintegrasi. Proses pengujian dilakukan dengan berbagai skenario input guna memastikan bahwa sistem mampu memberikan hasil yang akurat, responsif, dan stabil dalam berbagai kondisi.

9. Evaluasi dan Dokumentasi

Tahap akhir meliputi evaluasi menyeluruh terhadap performa sistem, mencakup akurasi model, kecepatan respons, serta kemudahan penggunaan. Selain itu, seluruh proses dan komponen sistem didokumentasikan secara rinci untuk mendukung pelaporan dan pengembangan di masa depan.



Gambar 3. 1 Alur Penelitian

3.3 Alat dan Bahan

Bagian ini menjelaskan berbagai alat dan bahan yang digunakan dalam proses penelitian, khususnya pada tahapan pengembangan dan evaluasi model Machine Learning. Penjelasan mencakup jenis data yang digunakan, perangkat lunak dan teknologi yang mendukung proses perancangan dan implementasi aplikasi, serta instrumen yang digunakan untuk pengumpulan dan analisis data dalam mengevaluasi efektivitas aplikasi terhadap pengguna sasaran.

3.3.1 Data Penelitian

Penelitian ini menggunakan data primer yang dikumpulkan secara langsung melalui proses scraping dari situs properti daring Rumah123.com. Data yang dikumpulkan berfokus pada wilayah Kota Semarang dan mencakup berbagai atribut properti yang relevan untuk proses prediksi harga rumah, antara lain:

- a. Harga rumah (target variabel),
- b. Lokasi,
- c. Luas tanah (m^2),
- d. Luas bangunan (m^2),
- e. Jumlah kamar tidur,
- f. Jumlah kamar mandi,
- g. Jumlah lantai,
- h. Daya listrik (VA),
- i. Carport,
- j. Kondisi properti,
- k. Kondisi Perabotan.

Proses scraping dilakukan secara sistematis untuk mengumpulkan data dalam jumlah besar dan representatif. Data yang diperoleh kemudian disimpan dalam format CSV dan digunakan pada tahap preprocessing, pelatihan model machine learning, serta evaluasi performa sistem.

3.3.2 Perangkat Pengembangan Sistem

Bagian ini menjelaskan alat dan teknologi utama yang digunakan dalam merancang, membangun, dan menguji sistem prediksi harga rumah *SmartHomeValuator*. Seluruh perangkat dipilih berdasarkan kompatibilitasnya dengan proses machine learning, integrasi web, dan kebutuhan penelitian yang berorientasi pada pengguna akhir:

1. Bahasa Pemrograman dan Environment

- a. Python digunakan sebagai bahasa utama untuk proses pengolahan data, pelatihan model machine learning, serta pengembangan backend sistem. Bahasa ini dipilih karena memiliki ekosistem pustaka data science dan machine learning yang sangat lengkap dan mendukung..
- b. JavaScript digunakan untuk mengembangkan antarmuka pengguna (frontend) berbasis web. Dalam implementasinya, framework React.js dipilih karena memungkinkan pembuatan UI yang dinamis, modular, dan efisien.
- c. Google Colab digunakan sebagai lingkungan pengembangan awal dan eksperimen model machine learning karena mendukung GPU dan akses ke pustaka Python terkini.
- d. Visual Studio Code digunakan sebagai *integrated development environment* (IDE) utama dalam pengembangan backend, frontend, serta integrasi sistem.

2. Library dan Framework Pendukung

- a. Scikit-learn digunakan untuk membangun model machine learning, termasuk Random Forest dan Gradient Boosting Regressor.
- b. XGBoost, LightGBM, dan CatBoost digunakan sebagai pustaka khusus untuk masing-masing algoritma boosting yang kemudian diintegrasikan dalam model ensemble.

- c. joblib digunakan untuk menyimpan dan memuat model yang telah dilatih dalam format .pkl, agar dapat digunakan dalam sistem backend secara efisien.
- d. Pandas, NumPy, matplotlib, seaborn digunakan untuk pengolahan data, eksplorasi fitur, visualisasi, dan evaluasi performa model selama pelatihan

3. Arsitektur Aplikasi Web

- a. Backend dibangun menggunakan FastAPI, framework modern berbasis Python yang memungkinkan pembuatan RESTful API dengan cepat dan terstruktur. FastAPI juga menyediakan fitur validasi input menggunakan Pydantic serta dokumentasi otomatis lewat Swagger UI.
- b. Frontend dibangun menggunakan React.js dengan dukungan Vite sebagai build tool dan Tailwind CSS sebagai framework untuk styling antarmuka. Framework ini memungkinkan pembuatan tampilan web yang responsif dan mudah disesuaikan.
- c. Komunikasi antara frontend dan backend dilakukan melalui API berbasis HTTP, dengan format data JSON untuk transfer input dan output prediksi.

4. Version Control dan Deployment

- a. GitHub digunakan sebagai platform version control untuk menyimpan kode sumber, mendokumentasikan perubahan, dan mendukung kolaborasi pengembangan.
- b. Railway digunakan sebagai platform cloud deployment untuk menghosting frontend dan backend secara terpisah, sehingga aplikasi dapat diakses secara publik melalui internet.

3.3.3 Alat Pengumpulan Data

Untuk memastikan sistem *SmartHomeValuator* bekerja sesuai tujuan, penelitian ini melibatkan beberapa instrumen untuk menguji kinerja prediksi dan mengumpulkan umpan balik dari pengguna. Instrumen penelitian dibagi menjadi dua pendekatan utama: pengujian sistem dari sisi teknis dan penilaian dari sisi persepsi pengguna.

1. Pengujian Sistem

Pengujian dilakukan secara manual dengan memberikan berbagai skenario input properti ke dalam sistem melalui antarmuka pengguna. Hasil prediksi diamati dan dicatat untuk:

- a. Memastikan integrasi frontend-backend berjalan stabil.
- b. Mengevaluasi apakah hasil prediksi numerik dan kategorikal konsisten dan logis.
- c. Menguji validasi input dan respons sistem terhadap data yang tidak sesuai.

2. Umpan Balik Pengguna (Persepsi)

Untuk mengevaluasi persepsi pengguna terhadap hasil prediksi dan kemudahan penggunaan sistem, digunakan instrumen berupa:

- a. Kuesioner online yang disebarkan kepada responden sasaran,
- b. Disusun menggunakan Google Forms, terdiri atas:
 - Pertanyaan tertutup (skala Likert) untuk menilai aspek seperti akurasi, kemudahan penggunaan, dan tampilan sistem,
 - Pertanyaan terbuka untuk memberikan masukan bebas dari pengguna.

3.3.4 Alat Analisis Data

Analisis dilakukan terhadap dua jenis data, yaitu (1) hasil prediksi sistem dan (2) tanggapan pengguna. Metode analisis dipilih sesuai dengan karakteristik masing-masing data.

1. Analisis Prediksi Sistem

- a. Python digunakan untuk menghitung metrik evaluasi model regresi seperti:
 - MAE (Mean Absolute Error): mengukur rata-rata kesalahan absolut,
 - MAPE (Mean Absolute Percentage Error): mengukur kesalahan dalam bentuk persentase,
 - R^2 (coefficient of determination): menilai seberapa baik model menjelaskan variasi harga.
- b. Visualisasi performa dan analisis fitur penting dilakukan dengan pustaka matplotlib dan seaborn untuk mendukung interpretasi hasil.

2. Analisis Hasil Kuesioner

- a. SPSS digunakan untuk menganalisis data dari kuesioner:
 - Statistik deskriptif seperti rata-rata, simpangan baku, dan distribusi jawaban,
 - Visualisasi dalam bentuk grafik batang atau diagram lingkaran.
- b. Microsoft Excel juga digunakan sebagai alat bantu untuk tabulasi dan perhitungan manual jika diperlukan.

BAB IV HASIL DAN PEMBAHASAN

4.1 Dataset dan Eksplorasi Awal

Penelitian ini menggunakan dataset properti rumah di wilayah Kota Semarang yang diperoleh melalui teknik *web scraping* dari situs properti daring Rumah123.com. Setelah proses pembersihan dan validasi, total data yang digunakan adalah 9.627 entri properti, dengan 31 fitur terkait karakteristik rumah.

Harga rumah dalam dataset ini bervariasi antara Rp123 juta hingga Rp10 miliar, dengan rata-rata sekitar Rp2,34 miliar dan median sebesar Rp1,6 miliar. Perbedaan signifikan antara nilai rata-rata dan median ini mengindikasikan adanya *skewness* ke kanan, yang disebabkan oleh sejumlah properti premium dengan harga sangat tinggi. Fitur-fitur utama yang digunakan dalam prediksi harga rumah adalah:

- a. Lokasi (kecamatan)
- b. Luas tanah (dalam m²)
- c. Luas bangunan (dalam m²)
- d. Jumlah kamar tidur
- e. Jumlah kamar mandi
- f. Jumlah lantai
- g. Daya listrik (dalam VA)
- h. Jumlah carport
- i. Kondisi properti (Baru / Siap Huni / Perlu Renovasi)
- j. Kondisi perabotan (Kosong / Semi-furnished / Furnished)
- k. Harga (target variabel)

4.2 Proses Preprocessing dan Transformasi Data

Data mentah yang diperoleh melalui scraping tidak langsung digunakan.

Beberapa tahapan preprocessing dilakukan untuk mempersiapkan data sebelum pelatihan model.

4.2.1 Pembersihan Data

- a. Menghapus entri duplikat dan data kosong (NaN)
- b. Menghilangkan outlier ekstrem (properti rumah >Rp. 10 Miliar)

4.2.2 Encoding dan Normalisasi

- a. Fitur kategorikal seperti Kondisi, Perabotan, dan Lokasi diubah menjadi one-hot encoding.
- b. Fitur numerik seperti luas dan daya listrik dinormalisasi menggunakan StandardScaler

4.2.3 Pembagian Data

Data dibagi menjadi:

Training set: 80%

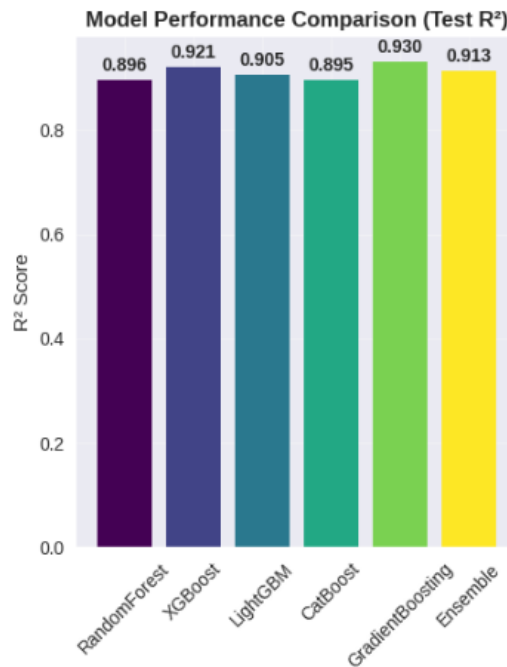
Testing set: 20%

4.3 Hasil Pelatihan dan Evaluasi Model Machine Learning

Pada tahap ini, lima algoritma machine learning yaitu Random Forest, Gradient Boosting Regressor, XGBoost, LightGBM, dan CatBoost dilatih menggunakan data properti rumah di Kota Semarang. Seluruh model diuji menggunakan data testing dan dibandingkan berdasarkan metrik evaluasi regresi, yaitu R^2 (koefisien determinasi), MAE (Mean Absolute Error), dan MAPE (Mean Absolute Percentage Error). Selain itu, model ensemble voting yang menggabungkan kelima model juga dievaluasi dengan metrik yang sama.

4.3.1 Hasil R² Score

R² score digunakan untuk mengukur seberapa besar variasi harga rumah dapat dijelaskan oleh model. Semakin mendekati nilai 1, semakin tinggi akurasi model dalam memprediksi harga.



Gambar 4. 1 Perbandingan R² Score Model

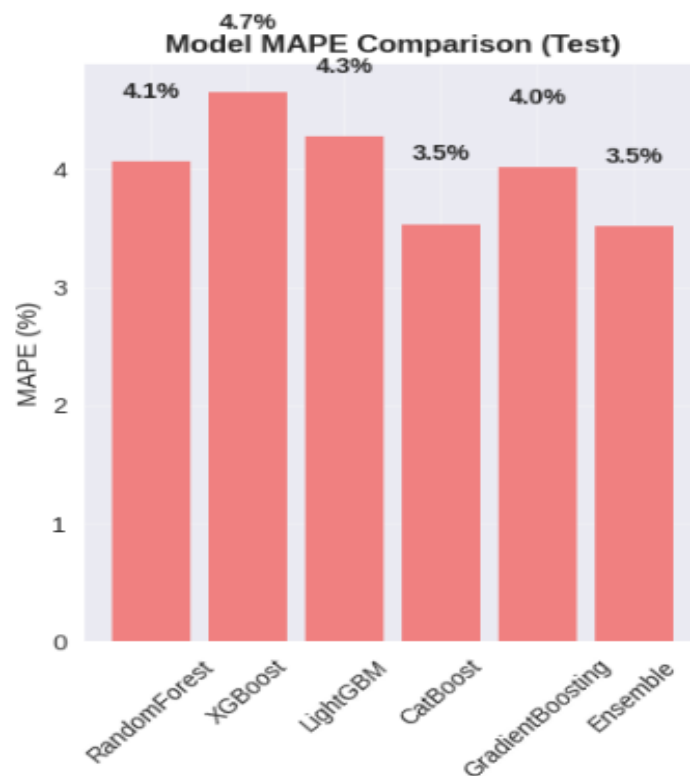
Berdasarkan Gambar 4.1, model Gradient Boosting Regressor memberikan hasil R² tertinggi sebesar 0.930, disusul oleh XGBoost (0.921) dan model ensemble voting (0.913). Meskipun model Random Forest dan CatBoost juga menunjukkan performa baik dengan nilai masing-masing 0,896 dan 0,895, keduanya sedikit lebih rendah dibanding model boosting lainnya.

4.3.2 Hasil MAE dan MAPE

MAE dan MAPE digunakan untuk mengukur besarnya kesalahan prediksi rata-rata, secara absolut dan dalam persentase. Tabel berikut merangkum nilai evaluasi semua model. Berikut adalah tabel hasil evaluasi Model:

Model	Train R ²	Test R ²	Train MAE	Test MAE
Random Forest	0.9996	0.896	9.75×10^6	1.92×10^8
XGBoost	0.9992	0.921	2.69×10^7	1.78×10^8
LightGBM	0.9994	0.905	2.06×10^7	1.88×10^8
CatBoost	0.9998	0.895	1.40×10^7	1.77×10^8
Gradient Boosting	0.9997	0.930	1.59×10^7	1.62×10^8
Ensemble	0.9998	0.913	1.18×10^7	1.69×10^8

Visualisasi MAPE dari masing-masing model ditunjukkan pada Gambar 4.2 berikut:



Gambar 4. 2 Perbandingan Model MAPE

Dari Gambar 4.2, terlihat bahwa CatBoost merupakan model dengan nilai MAPE terendah sebesar 3.5%, sama seperti model ensemble voting (3.5%), diikuti oleh Gradient Boosting (4.0%), Random Forest (4.1%), LightGBM (4.3%), dan XGBoost (4.7%). Nilai ini menunjukkan bahwa model ensemble mampu menyeimbangkan akurasi dan kesalahan prediksi, serta cukup stabil dalam menangani data testing.

4.4 Analisis Model Ensemble Voting

Bagian ini membahas performa model ensemble voting secara lebih mendalam, baik dari sisi hasil numerik maupun perbandingannya dengan model individu. Pembahasan juga mencakup alasan pemilihan metode ensemble serta pengaruhnya terhadap akurasi dan kestabilan sistem.

4.4.1 Perbandingan Model Individu vs Ensemble

Berdasarkan hasil evaluasi sebelumnya, model ensemble voting menunjukkan performa yang kompetitif dibandingkan dengan masing-masing model individu. Meskipun nilai R^2 ensemble (0.913) sedikit lebih rendah dibanding model Gradient Boosting (0.930) dan XGBoost (0.921), model ensemble menunjukkan konsistensi dan keseimbangan performa pada berbagai metrik.

Keunggulan ensemble terlihat pada:

a. Nilai MAE dan MAPE yang stabil

MAPE ensemble (3.5%) berada di antara nilai terbaik (CatBoost) dan jauh lebih rendah dari XGBoost (4.7%). Hal ini menunjukkan bahwa ensemble mampu menyeimbangkan error antar model.

b. Reduksi overfitting

Model seperti Gradient Boosting dan XGBoost cenderung memiliki R^2 training sangat tinggi, namun selisih besar dengan R^2 testing. Ensemble biasanya memiliki kemampuan generalisasi yang lebih baik karena menggabungkan output dari beberapa model dengan bobot yang seimbang.

4.4.2 Alasan Pemilihan Ensemble Voting

Metode ensemble voting dipilih karena memiliki beberapa keunggulan utama:

a. Mengurangi kelemahan model tunggal

Masing-masing algoritma memiliki karakteristik berbeda:

- Random Forest kuat terhadap outlier,
- Boosting unggul dalam menangani non-linearitas,
- CatBoost unggul pada data kategorikal.

Dengan menggabungkan model-model tersebut, ensemble mampu mengoptimalkan kelebihan dari masing-masing.

b. Meningkatkan stabilitas prediksi

Model tunggal sering kali sensitif terhadap parameter dan noise dalam data, sementara ensemble mengurangi fluktuasi tersebut dengan menerapkan prinsip *wisdom of the crowd*.

c. Hasil lebih dapat diandalkan

Secara keseluruhan, ensemble memberikan hasil prediksi yang lebih stabil dan mudah dikontrol, terutama untuk digunakan dalam aplikasi nyata seperti SmartHomeValuator.

4.4.3 Implikasi terhadap Sistem

Pemilihan model ensemble sebagai inti sistem prediksi memiliki dampak positif terhadap kualitas aplikasi:

- a. Memberikan prediksi numerik yang lebih realistis, bukan ekstrem.
- b. Meningkatkan kepercayaan pengguna, karena sistem dianggap mampu mempertimbangkan berbagai pendekatan prediktif.

- c. Memudahkan proses validasi model, karena performa ensemble cenderung tidak terlalu bervariasi antar iterasi.

4.5 Implementasi Sistem SmartHomeValuator

Pada tahap ini, hasil dari proses pelatihan dan evaluasi model machine learning diintegrasikan ke dalam sebuah sistem berbasis web bernama *SmartHomeValuator*. Sistem ini dibuat untuk membantu masyarakat dalam memperkirakan harga rumah di Kota Semarang dengan cara yang cepat, akurat, dan mudah dipahami bahkan oleh pengguna tanpa latar belakang teknis..

4.5.1 Arsitektur Sistem

SmartHomeValuator terdiri dari dua bagian utama, yaitu frontend dan backend, yang saling terhubung melalui RESTful API. Berikut ini adalah penjelasan dari masing-masing komponennya:

1. Frontend (Client Side)

Dibangun menggunakan React.js dan Tailwind CSS. Frontend menyediakan antarmuka pengguna berupa formulir input data properti serta tampilan hasil prediksi dalam bentuk numerik dan kategori harga.

2. Backend (Server Side)

Dibangun menggunakan FastAPI berbasis Python. Backend bertugas menerima data input dari pengguna, melakukan validasi menggunakan Pydantic, memproses data menggunakan model ensemble ML, dan mengembalikan hasil prediksi ke frontend dalam format JSON.

3. Model Machine Learning

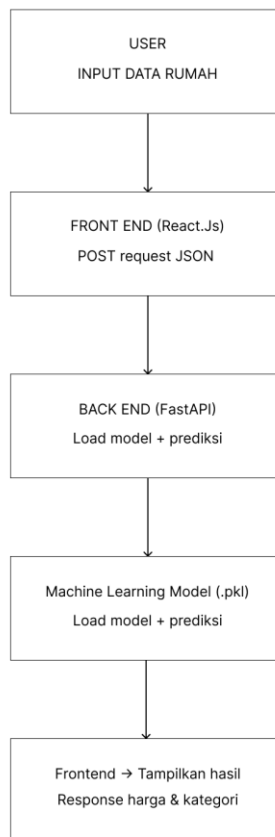
Model ensemble voting yang telah dilatih sebelumnya disimpan dalam format .pkl menggunakan joblib, dan dimuat oleh backend saat sistem berjalan.

4. API Endpoint

Endpoint utama adalah /predict, yang menerima input fitur properti dan mengembalikan estimasi harga serta label kategori ("Murah", "Sedang", "Mahal").

4.5.2 Alur Kerja Sistem

- a. Pengguna membuka halaman web dan mengisi data properti (luas tanah, bangunan, jumlah kamar, dll.).
- b. Data dikirim ke backend melalui API menggunakan metode POST.
- c. Backend melakukan validasi input, memuat model ensemble, dan menghasilkan prediksi.
- d. Hasil prediksi berupa harga (dalam Rupiah) dan label kategori dikirim kembali ke frontend.
- e. Frontend menampilkan hasil prediksi dalam format yang mudah dipahami.





Gambar 4. 3 Alur Sistem


4.5.3 Deployment dan Akses Sistem

Sistem *SmartHomeValuator* dideploy menggunakan layanan cloud Railway, di mana backend dan frontend dihosting secara terpisah namun saling terhubung. Kode sumber sistem dikelola melalui GitHub, yang juga digunakan untuk version control dan dokumentasi pengembangan.

Masukkan detail properti Anda untuk mendapatkan estimasi harga yang akurat

 Fisik Properti

 Kondisi

 Zona

Luas Tanah (m²) *

600

Luas Bangunan (m²) *

500

Kamar Tidur *

5

Kamar Mandi *

3

Jumlah Lantai *

2

Carport *

1

Daya Listrik (Watt) *

3500

← Sebelumnya

Selanjutnya →

Gambar 4. 4 Form Input Fisik Properti

Masukkan detail properti Anda untuk mendapatkan estimasi harga yang akurat

Fisik Properti
 Kondisi
 Zona

Kondisi Properti *

Bagus

Baik

Butuh Renovasi

Kondisi Perabotan *

Furnished

Semi Furnished

Unfurnished

← Sebelumnya
 Selanjutnya →

Gambar 4. 5 Form Input Kondisi Properti

Masukkan detail properti Anda untuk mendapatkan estimasi harga yang akurat

Fisik Properti
 Kondisi
 Zona

Zona Lokasi *

Semarang Barat

Semarang Selatan

Semarang Tengah

Semarang Timur

Semarang Utara

← Sebelumnya
 Prediksi Harga

Gambar 4. 6 Form Input Zona Properti

2. Hasil Prediksi: Visualisasi hasil dengan angka besar, badge kategori harga, dan format ringkas.



Gambar 4. 7 Hasil Prediksi

4.6 Uji Coba Sistem dan Tampilan Antarmuka

Bagian ini menjelaskan hasil uji coba sistem *SmartHomeValuator* dari sisi fungsionalitas dan tampilan antarmuka pengguna. Tujuannya adalah untuk memastikan bahwa sistem dapat menerima input data secara utuh, memberikan hasil prediksi yang sesuai, serta menyajikannya dalam format yang mudah dipahami.

4.6.1 Skenario Uji Coba

Pengujian sistem dilakukan dengan beberapa skenario input properti yang bervariasi, guna mengevaluasi stabilitas prediksi dan kebenaran proses integrasi frontend-backend.

Contoh Skenario 1:

- Luas Tanah: 120 m²
- Luas Bangunan: 90 m²

- Kamar Tidur: 3
- Kamar Mandi: 2
- Lantai: 2
- Carport: 1
- Daya Listrik: 2200 VA
- Lokasi: Kecamatan Tembalang
- Kondisi: Siap Huni
- Perabotan: Semi-Furnished

Hasil Prediksi:

- Estimasi Harga: Rp 850.000.000
- Kategori: Sedang

Sistem mampu memberikan hasil prediksi dengan waktu respons kurang dari 1 detik, serta menampilkan output secara real-time di antarmuka pengguna.

4.6.2 Validasi Respons API

Pengujian dilakukan juga secara langsung terhadap endpoint API menggunakan tools seperti Swagger UI atau browser console untuk memastikan bahwa:

- a. Format JSON respons sesuai spesifikasi,
- b. Tidak ada error saat input tidak lengkap atau tidak valid,
- c. Model dapat di-load secara konsisten setiap kali endpoint dipanggil.

Curl

```
curl -X 'POST' \
  'https://smarthome-api.up.railway.app/predict' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "lokasi": "Tembalang",
    "kamar_tidur": 3,
    "kamar_mandi": 2,
    "luas_tanah": 150,
    "luas_bangunan": 120,
    "carport": 1,
    "daya_listrik": 1300,
    "jumlah_lantai": 1,
    "kondisi_properti": "Bagus",
    "kondisi_perabotan": "Tidak Berperabot"
  }'
```

Request URL

https://smarthome-api.up.railway.app/predict

Server response

Code Details

200

Response body

```
{
  "predicted_price": 270617993.14743155,
  "predicted_price_formatted": "Rp 270.617.993",
  "predicted_price_miliar": 0.27061799314743155,
  "confidence_interval": "Rp 184.856.699 - Rp 356.379.287",
  "uncertainty": 43755762.30672345,
  "property_category": "Ekonomis",
  "zona": "Semarang Timur",
  "preprocessing_info": {
    "missing_fields": [],
    "imputed_values": {},
    "zona_mapped": "Semarang Timur",
    "features_shape": [
      1,
      6
    ]
  }
}
```

Response headers

Download

Gambar 4. 8 Respon JSON dari API /predict

BAB V PENUTUP

5.1 Kesimpulan

Penelitian ini bertujuan untuk membangun sistem prediksi harga rumah di Kota Semarang berbasis web dengan memanfaatkan pendekatan machine learning ensemble voting. Sistem ini diberi nama *SmartHomeValuator* dan dikembangkan dengan memanfaatkan lima algoritma regresi yang populer, yaitu Random Forest, Gradient Boosting Regressor, XGBoost, LightGBM, dan CatBoost.

Setelah melalui tahapan pelatihan dan evaluasi, kesimpulan yang dapat diambil adalah:

1. Hasil pelatihan menunjukkan bahwa penggunaan model *ensemble voting* yang terdiri dari lima algoritma populer (Random Forest, Gradient Boosting, XGBoost, LightGBM, dan CatBoost) mampu meningkatkan akurasi dan konsistensi prediksi harga rumah di wilayah Kota Semarang. Berdasarkan evaluasi menggunakan metrik regresi seperti R^2 , MAE, dan MAPE, model ensemble memberikan performa terbaik dibandingkan model individu, dengan tingkat kesalahan prediksi yang lebih rendah dan kestabilan yang lebih tinggi terhadap variasi data properti.
2. Sistem prediksi harga rumah berbasis web yang dibangun dalam penelitian ini, yaitu SmartHomeValuator, mampu menyajikan hasil prediksi dalam dua format output: numerik (estimasi harga dalam rupiah) dan kategorikal ("Murah", "Sedang", "Mahal"). Dengan dua cara penyajian hasil, pengguna yang tidak familiar dengan dunia teknis tetap bisa memahami nilai properti tanpa harus repot mempelajari algoritma atau statistik.
3. Sistem SmartHomeValuator dirancang menggunakan arsitektur client-server dengan teknologi React.js untuk antarmuka pengguna dan FastAPI untuk backend serta integrasi model machine learning. Sistem ini terbukti dapat diakses secara mudah melalui web browser dan memberikan respons yang cepat serta hasil prediksi yang akurat. Hal ini menunjukkan bahwa

integrasi antara metode ensemble dengan sistem aplikasi berbasis web dapat menjadi solusi yang efektif untuk prediksi harga properti secara real-time dan dapat diakses oleh masyarakat umum.

5.2 Saran

Agar sistem *SmartHomeValuator* dapat dikembangkan lebih lanjut, berikut beberapa saran yang dapat dijadikan acuan untuk penelitian selanjutnya:

- a. Menambahkan data dari berbagai sumber untuk meningkatkan kualitas dan generalisasi model.
- b. Menggunakan teknik ensemble yang lebih canggih seperti stacking atau weighted voting agar performa model lebih optimal.
- c. Menambahkan fitur seperti peta interaktif, visualisasi tren harga, atau integrasi ke database publik (BPN, BI) untuk memberikan nilai tambah bagi pengguna.
- d. Melibatkan pengguna dalam pengujian usability secara lebih luas untuk memperbaiki tampilan dan pengalaman pengguna sistem.

REFERENSI

- [1] Bank Indonesia, *Survei Harga Properti Residensial (SHPR) Triwulan III 2022*. Jakarta: Bank Indonesia, 2022.
- [2] R. Tanamal, N. Minoque, T. Wiradinata, Y. Soekamto, and T. Ratih, "House price prediction model using random forest in Surabaya City," *TEM Journal*, vol. 12, no. 1, pp. 126–132, Feb. 2023, doi: 10.18421/tem121-17.
- [3] S. Wu, "Shanghai house price prediction using Random Forest," *Advances in Economics Management and Political Sciences*, vol. 66, no. 1, pp. 224–230, Jan. 2024, doi: 10.54254/2754-1169/66/20241234.
- [4] H. Li, "House price prediction and analysis based on random Forest and XGBooSt models," *Highlights in Business Economics and Management*, vol. 21, pp. 934–938, Dec. 2023, doi: 10.54097/hbem.v21i.14837.
- [5] L. Fang, "Machine learning models for house price prediction," *Applied and Computational Engineering*, vol. 4, no. 1, pp. 409–415, May 2023, doi: 10.54254/2755-2721/4/20230505.
- [6] I. R. Ningsih, A. Faqih, and A. R. Rinaldi, "House price prediction analysis using a comparison of machine learning algorithms in the Jabodetabek area," *Journal of Artificial Intelligence and Engineering Applications (JAIEA)*, vol. 4, no. 2, pp. 687–694, Feb. 2025, doi: 10.59934/jaiea.v4i2.733.
- [7] Z.-H. Zhou, *Ensemble Methods: Foundations and Algorithms*. Boca Raton, FL: CRC Press, 2012.
- [8] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.
- [9] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001, doi: 10.1214/aos/1013203451.
- [10] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, 2016, pp. 785–794, doi: 10.1145/2939672.2939785.
- [11] G. Ke et al., "LightGBM: A highly efficient gradient boosting decision tree," in *Advances in Neural Information Processing Systems* (NeurIPS), 2017.
- [12] L. Prokhorenkova et al., "CatBoost: Unbiased boosting with categorical features," in *Advances in Neural Information Processing Systems* (NeurIPS), 2018.