

BAB II TINJAUAN PUSTAKA

2.1. Landasan Teori

2.1.1. Prediksi

Prediksi merupakan proses untuk memperkirakan kejadian di masa depan berdasarkan pola atau preferensi dari data historis, yang bertujuan untuk mendukung pengambilan keputusan di berbagai bidang. Proses ini dilakukan melalui pengumpulan data historis, yang dapat diperoleh dari berbagai sumber seperti basis data (*database*) maupun repositori (Rahayu, 2023).

Prediksi harga dengan dibantu menggunakan algoritma *machine learning* dapat membantu dalam membuat sebuah keputusan. Perusahaan dalam memprediksi harga produk tidak dapat dilakukan dengan hanya perkiraan semata. *Machine Learning* mampu membantu dalam menentukan harga produk dengan menggunakan algoritma yang mempunyai akurasi yang akurat.

2.1.2. Tren Harga

Secara umum, prediksi tren harga merupakan teknik analisis yang memanfaatkan data historis untuk memperkirakan pergerakan harga di masa mendatang. Pendekatan ini dilakukan secara sistematis dengan dasar data relevan dari periode sebelumnya, sehingga diharapkan mampu memberikan tingkat objektivitas yang lebih tinggi dalam menghasilkan proyeksi harga di masa depan (Rozi et al., 2023).

2.1.3. Hubungan Tren Harga dengan Supermarket

Mengidentifikasi pola tren harga, supermarket dapat mendapatkan wawasan penting mengenai periode penjualan produk. Tren harga merupakan perubahan terhadap harga produk yang secara signifikan tidak dapat ditebak. Maka dari itu supermarket menerapkan harga awal dan secara otomatis nantinya menyesuaikan berdasarkan reaksi dan tren yang diperoleh dari pelanggan (Algoritma, 2024).

Prediksi harga produk dilakukan dengan melihat tren harga historis serta membandingkannya dengan harga produk serupa di toko pembanding. Pendekatan ini bertujuan untuk mengidentifikasi pola harga yang relevan dan membantu menghasilkan prediksi yang lebih akurat untuk tiga hari kedepan. Dengan memanfaatkan data harga dari toko lain sebagai acuan, supermarket dapat menyesuaikan strategi harga secara lebih kompetitif di pasar.

2.1.4. Rekomendasi Produk

Rekomendasi produk adalah memberikan rekomendasi produk yang akan dipilih berdasarkan preferensi masa lalu, riwayat pembelian, dan informasi (Yaya Suharya et al., 2021). Sistem rekomendasi sendiri dibagi menjadi tiga tipe, yaitu *Content-Based-Filtering*, *Collaborative Filtering*, dan *Item-Based Filtering* (Suhailah & Hartatik, 2023).

A. Content-Based-Filtering

Content-Based-Filtering memberikan rekomendasi produk yang sesuai dengan historis produk yang disukai sebelumnya.

B. Collaborative Filterings

Collaborative Filtering dalam sistem rekomendasi merupakan salah satu metode yang memanfaatkan informasi dari pengguna yang berupa nilai terhadap produk.

C. Item-Based Filtering

Item-Based Filtering memberikan rekomendasi kepada pengguna lain berdasarkan nilai kemiripan *item* atau produk yang dihitung berdasarkan nilai peringkat yang diberikan pengguna.

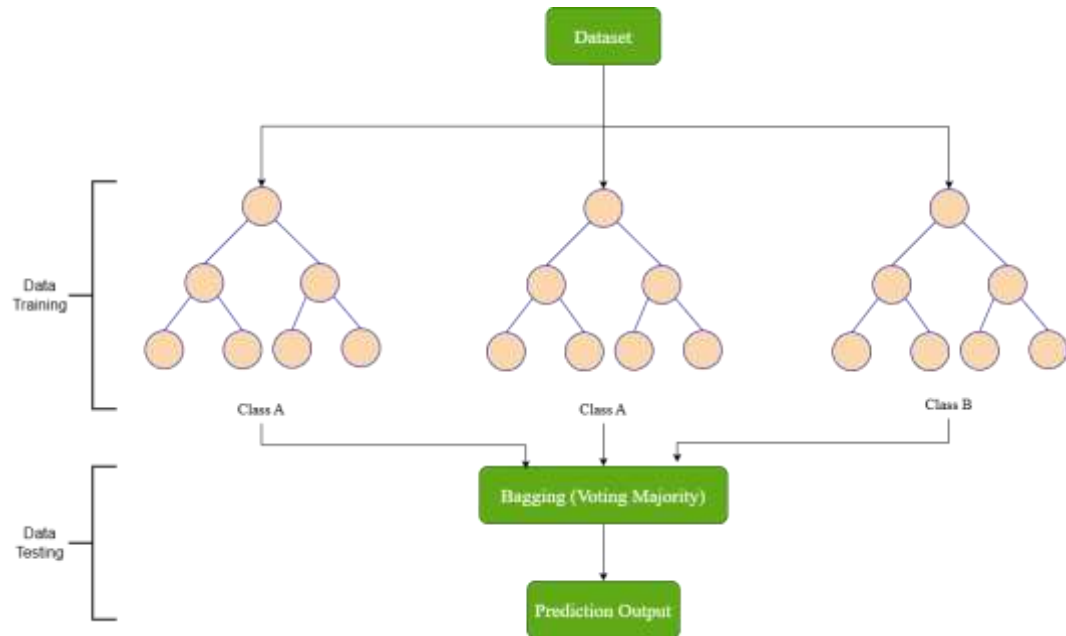
2.1.5. Machine learning

Machine learning merupakan salah satu cabang dari ilmu komputer yang berfokus pada pengembangan kemampuan mesin untuk belajar secara cerdas dari data. Salah satu metode dalam machine learning adalah *concept learning*, yaitu pendekatan yang termasuk dalam kategori *supervised learning* karena memerlukan data latih yang mencakup contoh positif maupun negatif (Muhammad et al., 2023). *Machine learning* mampu membantu membuat keputusan khususnya dalam memprediksi harga agar lebih akurat dikarenakan berbagai model yang mempunyai akurasi cukup akurat. Pada penelitian ini pendekatan *machine learning* yang digunakan diantaranya model *Random Forest*, *XGBoost*, *Gradient Boosting*, *Decision Tree* dan *Regresi Linear*.

A. Random Forest

Random Forest merupakan metode *ensemble* yang memanfaatkan sejumlah pohon keputusan untuk melakukan prediksi, dengan membangun beberapa pohon selama proses pelatihan dan menghasilkan prediksi akhir berupa rata-rata

dari seluruh pohon pada kasus regresi (Aditya et al., 2024). Cara kerja dari algoritma *random forest* sendiri di visualisasikan pada Gambar 2.1.



Gambar 2. 1 Algoritma *Random Forest* (Aditya et al., 2024)

Pada Gambar 2.1, secara garis besar cara kerja algoritma *random forest* diantaranya:

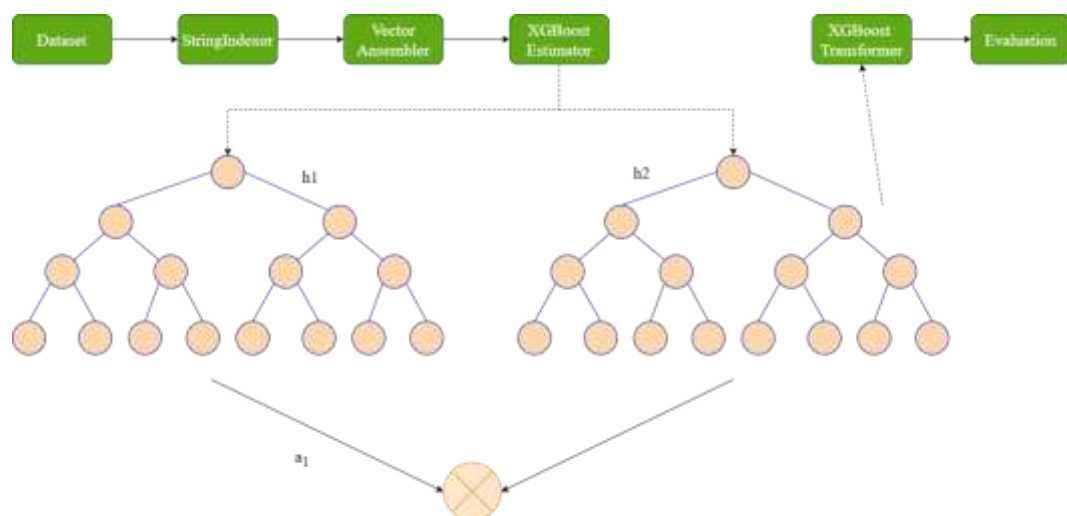
- 1) Membagi dataset menjadi beberapa subset menggunakan *bootstrap sampling* atau membagi dataset menjadi data *train* dan *test*.
- 2) Pada data *training* setiap subset digunakan untuk melatih sebuah *decision tree*, dan hanya sebagian fitur yang dipilih secara acak untuk setiap *split*. Hal ini akan membantu dalam mengurangi *overfitting* dengan membuat setiap *tree* beragam.
- 3) Setelah semua *tree* terlatih, *tree* tersebut akan digunakan untuk melakukan prediksi. Untuk masalah klasifikasi, akan dipilih variabel atau kelas mana

yang memiliki suara terbanyak (*voting majority*). Kemudian untuk masalah regresi, rata-rata semua *tree* digunakan sebagai hasil akhir.

- 4) Hasil akhir dari proses agregasi prediksi dari semua *decision tree*, yang menghasilkan prediksi yang lebih stabil dan akurat dibandingkan hanya menggunakan satu *decision tree*.

B. XGBoost

XGBoost adalah versi yang disempurnakan dari *Gradient Boosting*, dengan kemampuan lebih tinggi dalam menyelesaikan tugas regresi dan klasifikasi melalui pendekatan *Gradient Boosting* (Andriansyah, 2023). Cara kerja dari algoritma *XGBoost* di visualiasikan pada Gambar 2.2.



Gambar 2. 2 Algoritma *XGBoost* (Andriansyah, 2023)

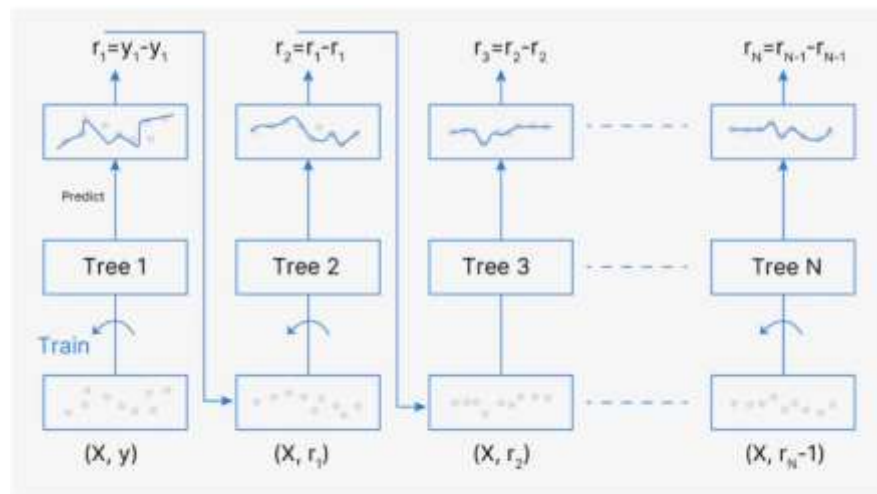
Pada Gambar 2.2, alur dari algoritma *XGBoost*, sebagai berikut:

- 1) **Dataset:** Dataset di inputkan ke sistem.

- 2) *StringIndexer*: Data kategorikal dalam dataset diubah menjadi format numerik. Langkah ini penting agar data bisa diproses oleh pendekatan *machine learning*.
- 3) *VectorAssembler*: Fitur-fitur numerik digabungkan ke dalam suatu vektor. Vektor ini menjadi representasi tunggal dari data yang akan digunakan untuk pelatihan model.
- 4) *XGBoost Estimator*: Setelah data diubah menjadi format vektor, data diberikan ke *XGBoost Estimator*, yang dimana pada bagian ini melatih model *XGBoost* berdasarkan *tree* yang telah di bentuk.
- 5) *XGBoost Transformer*: Setelah model terlatih, kemudian akan digunakan sebagai *transformer* untuk menerapkan prediksi pada data baru.
- 6) *Evaluation*: Hasil prediksi dari *transformer* dievaluasi menggunakan matrix evaluasi tertentu untuk menilai kinerja model.

C. Gradient Boosting

Gradient Boosting adalah salah satu metode *ensemble learning* untuk melakukan prediksi, metode ini cocok digunakan pada data yang sangat besar (Septian, 2023). Untuk gambaran proses *Gradient Boosting* dapat dilihat pada Gambar 3.



Gambar 2. 3 Algoritma *Gradient Boosting* (Septian, 2023)

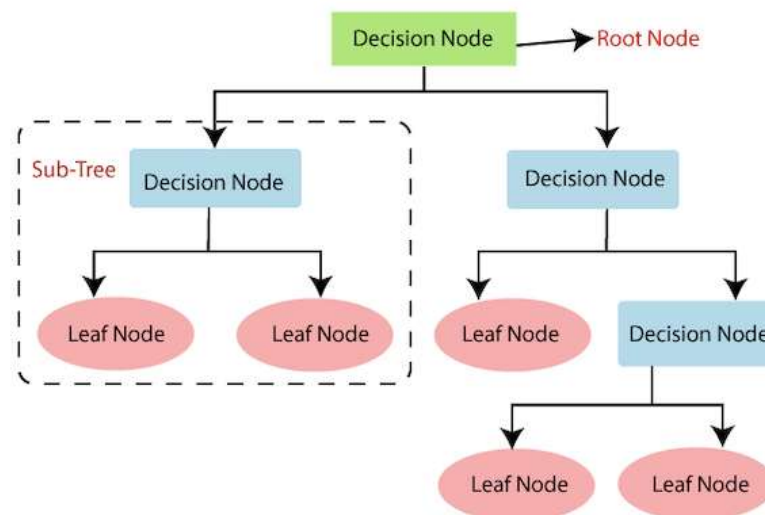
Pada Gambar 2.3, menjelaskan gambaran dari alur algoritma *gradient boosting* diantaranya:

- 1) Inisialisasi Model (*Tree 1*): *Gradient Boosting* dimulai dengan melatih *tree 1* pada dataset awal (X, y) . Model pertama mencoba memprediksi target y berdasarkan fitur X . Setelah model memprediksi ($-y_1$), residu (r_1) dihitung sebagai selisih antara nilai aktual (y_1) dan prediksi awal ($-y_1$).
- 2) Model Kedua (*Tree 2*): Model *Tree 2* dilatih untuk mempelajari residu (r_1) dari model pertama. Prediksi dari *Tree 2* menghasilkan residu baru (r_2) dengan menghitung selisih antara residu sebelumnya (r_1) dan prediksi model kedua ($-r_1$).
- 3) Iterasi *Tree 3* hingga *Tree N*: Proses ini diulangi untuk model berikutnya. Setiap model dilatih pada residu yang tersisa dari model sebelumnya. Residu baru dihitung pada setiap iterasi, *residual learning* ini memastikan bahwa setiap model bertugas untuk memperbaiki kesalahan dari model sebelumnya.

- 4) *Tree N*: Pada model terakhir (*Tree N*), residu akhir (r_N) dihitung. Proses ini akan berhenti ketika jumlah *tree* telah mencapai batas tertentu atau ketika residu sudah cukup kecil.
- 5) Kombinasi prediksi: Dengan mengkombinasikan semua *tree* yang telah dilatih. Bobot kontribusi setiap *tree* dapat diatur menggunakan faktor *learning rate*.

D. Decision Tree

Decision Tree merupakan salah satu algoritma dalam machine learning yang digunakan untuk membangun model klasifikasi dan regresi dalam bentuk struktur pohon. Pendekatan ini memungkinkan interpretasi proses pengambilan keputusan secara terstruktur, bertahap, dan rasional, sehingga mendukung prediksi terhadap data di masa depan (Nurani et al., 2023). Gambaran visualisasi *decision tree* dapat dilihat pada Gambar 2.4.



Gambar 2. 4 Algoritma *Decision Tree* (Nurani et al., 2023)

Pada Gambar 2.4, merupakan elemen-elemen yang terdapat pada algoritma *decision tree*, diantaranya:

- 1) *Root node (node akar)*: Node pertama dalam pohon, digunakan untuk membagi dataset berdasarkan fitur tertentu. Keputusan pertama dibuat di *node* ini, yang kemudian mengarahkan data ke *node* berikutnya.
- 2) *Decision node (node keputusan)*: *Node* bercabang, *node* ini mewakili tes atau keputusan berdasarkan fitur dataset. Setiap cabang dari *node* keputusan mengarah ke sub-*node*, yang bisa berupa *leaf node* atau *node* keputusan lainnya.
- 3) *Leaf node (node daun)*: *node* akhir pada pohon, *node* ini tidak bercabang lebih lanjut dan mewakili keputusan atau hasil akhir. Dalam klasifikasi, *leaf node* berisi label kelas, sedangkan dalam regresi, *leaf node* berisi nilai numerik.
- 4) *Sub-Tree (Sub-Pohon)*: Bagian dari pohon keputusan yang merupakan cabang dari *root node*. Sub-tree ini dianggap sebagai pohon keputusan kecil yang berdiri sendiri.

Untuk proses alur algoritma *decision tree*, dimulai dari *root node*, kemudian dibagi secara berulang melalui *decision nodes* berdasarkan kriteria tertentu. Proses berlanjut sampai mencapai *leaf node*, dimana keputusan akhir dibuat.

E. Regresi Linear

Regresi linear merupakan metode analisis statistik yang digunakan untuk mempelajari hubungan antara satu variabel *independen* dan satu variabel *dependen* (Andriani et al., 2023). Metode ini berguna untuk menguji sejauh

mana variabel independen (X) memengaruhi variabel dependen (Y) dalam konteks hubungan sebab-akibat. Pada persamaan prediksi (2.1), konstanta (2.2), dan koefisien (2.3) *regresi linier* adalah sebagai berikut:

Prediksi :

$$Y = a + bX \quad (2.1)$$

Konstanta :

$$a = \frac{(\sum Y)(\sum X^2) - (\sum X)(\sum XY)}{n \sum X^2 - (\sum X)^2} \quad (2.2)$$

Koefisien :

$$b = \frac{n \sum XY - (\sum X)(\sum Y)}{n \sum X^2 - (\sum X)^2} \quad (2.3)$$

Keterangan :

Y = Variabel dependen (variabel terikat).

X = Variabel independen (variabel bebas).

a = Konstanta (nilai dari Y apabila X = 0).

b = Koefisien regresi (pengaruh positif atau negatif).

Alur dari algoritma *regresi linier* diantaranya sebagai berikut:

- 1) Mengumpulkan data: Kumpulkan pasangan data X (independen) dan Y (dependen).
- 2) Menghitung parameter a dan b: untuk menghitung parameter a menggunakan rumus parameter 2 dan untuk menghitung parameter b menggunakan rumus parameter 3.

- 3) Membentuk persamaan *linear*: setelah nilai a dan b diketahui, bentuk persamaan linear pada rumus persamaan 1.
- 4) Melakukan prediksi: Masukkan nilai X kedalam persamaan Y .
- 5) Evaluasi model: Bandingkan nilai Y prediksi ($\hat{Y}$) dengan nilai aktual Y untuk mengavaluasi keakuratan model.

2.1.6. Feature Importance

Feature importance merupakan teknik yang digunakan untuk menilai tingkat kontribusi masing-masing fitur dalam proses prediksi model, sehingga memungkinkan identifikasi fitur-fitur yang paling relevan dan signifikan (Hamid Rahman, Ramdani Agusman, 2024).

Pengujian variabel untuk mengecek korelasi antara variabel target dan variabel yang mempengaruhi dibantu dengan *heatmap*. Variabel yang mempunyai nilai pengujian <0.01 , maka variabel tersebut disarankan untuk dilakukan penghapusan, dikarenakan tidak adanya korelasi terhadap variabel target sehingga tidak akan mempengaruhi hasil akurasi.

2.1.7. Feature Selection

Pemilihan fitur atau variabel dalam dataset bertujuan untuk mengurangi beban komputasi. Proses ini merupakan bagian penting dalam *machine learning* karena dalam beberapa kasus, sebuah dataset mungkin mengandung banyak fitur yang tidak relevan, yang dapat mempengaruhi akurasi algoritma secara *negative* (Sudarman & Budi, 2023).

Proses sebelumnya dalam pengujian variabel menggunakan *feature importance*, tahapan selanjutnya yaitu dengan memilih fitur berdasarkan hasil dari pengujian *feature importance* yang mengambil fitur-fitur yang mempunyai nilai tinggi berdasarkan hasil pengujian untuk meningkatkan performa model, pada model yang mempunyai akurasi awal terbaik.

2.1.8. **Matrix Evaluation**

Matrix evaluation bertujuan untuk mengevaluasi performa model terhadap hasil akurasi prediksi (Edris Effendi et al., 2023). Pada penelitian ini digunakan beberapa *matrix evaluation* diantaranya:

A. *R-Squared*

R-Squared digunakan untuk mengukur keefektifan seberapa akurat algoritma tersebut dalam memprediksi (Rosita & Moonlight, 2024). Pada persamaan *R-Squared* (2.4):

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (2.4)$$

Keterangan:

- y = nilai aktual
- y = nilai perkiraan
- n = nilai aktual
- i = indeks data

Tabel 2.1 menunjukkan rentang nilai R^2 , angka yang berkisar 0 sampai 1 mengindikasikan besar kombinasi variabel independen secara bersama-sama mempengaruhi nilai variabel independen (Nurani et al., 2023).

Tabel 2. 1 Rentang Nilai R²

Interval Koefisien	Hubungan
1 – 0,8	Sangat kuat
0,6 – 0,79	Kuat
0,4 – 0,59	Cukup kuat
0,2 – 0,39	Lemah
0 – 0,19	Sangat lemah

B. Mean Absolute Percentage Error (MAPE)

Mean Absolute Percentage Error (MAPE) dapat membantu mengukur kesalahan dalam prediksi (Triya et al., 2024). Pada persamaan MAPE (2.5):

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left(\frac{|y_i - \hat{y}_i|}{y_i} \times 100 \right) \quad (2.5)$$

Keterangan:

y = nilai aktual

-y = nilai perkiraan

n = nilai aktual

i = indeks data

Semakin kecil nilai *Mean Absolute Percentage Error* (MAPE), maka semakin tinggi tingkat akurasi dari hasil peramalan. Tabel 2.2 menunjukkan rentang interpretasi nilai MAPE, di mana MAPE kurang dari 10% dikategorikan sebagai prediksi dengan tingkat akurasi sangat baik. Nilai MAPE antara 10% hingga 20% menunjukkan akurasi yang baik, sedangkan nilai antara 20% hingga 50% mencerminkan tingkat akurasi yang cukup. Sementara itu, nilai MAPE yang melebihi 50% menunjukkan bahwa tingkat akurasi prediksi tergolong buruk (Nurani et al., 2023).

Tabel 2. 2 Rentang Nilai MAPE

Nilai MAPE	Akurasi Prediksi
<10%	Sangat baik
10% - 20%	Baik
20% - 50%	Cukup
>50%	Buruk

C. Root Mean Squared Error (RMSE)

Root Mean Squared Error mempunyai fungsi untuk mengevaluasi seberapa baik model yang digunakan dalam memprediksi (Rosita & Moonlight, 2024). Pada persamaan RMSE (2.6):

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|} \quad (2.6)$$

Keterangan:

y = nilai aktual

-y = nilai perkiraan

n = nilai aktual

i = indeks data

2.2. State-of-the-Art

Terdapat beberapa penelitian yang signifikan menggunakan berbagai pendekatan untuk melakukan prediksi harga dan juga merekomendasikan sebuah produk. Kontribusi penting dari penelitian-penelitian tersebut terletak pada pemahaman dan faktor-faktor yang mempengaruhi terhadap prediksi harga produk. Pada penelitian sebelumnya memiliki potensi untuk menjadi dasar penting dalam pengembangan metode prediksi untuk meningkatkan hasil akurasi yang lebih akurat. Berikut Tabel 2.3 mengenai penelitian terkait yang menjadi landasan dalam penelitian ini.

Tabel 2. 3 State-of-The-Art

No	Nama Peneliti/Journal		Hasil Penelitian
1	Peneliti (tahun)	(Warjiyono et al., 2024)	Penelitian ini menggunakan algoritma <i>Random Forest</i> untuk memprediksi harga rumah berdasarkan dataset dari <i>Kaggle</i> . Variabel dependen adalah harga rumah, sementara variabel independen meliputi luas tanah, luas bangunan, jumlah kamar tidur, jumlah kamar mandi, garasi, dan kota. Model <i>Random Forest</i> menghasilkan akurasi sebesar 75,10%.
	Judul	Analisa Prediksi Harga Jual Rumah Menggunakan Algoritma <i>Random Forest Machine learning</i>	
	Algoritma/Metode	<i>Random Forest</i>	
2	Peneliti (tahun)	(Radithya et al., 2024)	

	Judul	Prediksi Target Penjualan Mobil Menggunakan <i>Decision Tree</i>	.Penelitian ini menggunakan algoritma <i>Decision Tree</i> untuk memprediksi target penjualan. Untuk meningkatkan performa model dibantu dengan metode <i>feature scalling</i> , dan mendapatkan akurasi 91%.
	Algoritma/Metode	<i>Decision Tree + Feature Scaling</i>	
3	Peneliti (tahun)	(Hidayah et al., 2024)	Penelitian ini menggunakan dataset dari paket MASS di RStudio dengan 7.048 data dan 14 variabel, dibagi menjadi 70% data training dan 30% data testing. Model <i>Regresi Linear</i> menghasilkan R^2 sebesar 0.71 dan RMSE sebesar 4.64%, model SVR mencapai R^2 sebesar 0.80 dan RMSE sebesar 3.83%, model <i>Decision Tree</i> memiliki R^2 sebesar 0.59 dan RMSE sebesar 5.49%, sedangkan model <i>Random Forest</i> menunjukkan performa terbaik dengan R^2 sebesar 0.86 dan RMSE sebesar 3.20%. Disarankan untuk penelitian selanjutnya menggunakan pembagian data 80:20 untuk meningkatkan akurasi.
	Judul	Prediksi Harga Rumah di Boston Menggunakan Metode <i>Linear Regression</i> , SVR, <i>Decision Tree</i> , <i>Random Forest Regression</i>	
	Algoritma/Metode	<i>Regresi Linear</i> , SVR, <i>Decision Tree</i> , <i>Random Forest</i>	
4	Peneliti (tahun)	(Simamora et al., 2024)	Berdasarkan hasil kutipan pada penelitian untuk meningkatkan hasil akurasi prediksi harga emas dengan menggabungkan algoritma <i>random forest</i> dan <i>gradient boosting</i> , variabel yang ada pada dataset diantaranya terdapat 7 variabel. Untuk hasil akurasi
	Judul	Peningkatan dan Optimalisasi Prediksi Harga Emas Menggunakan Metode <i>combine</i>	

		<i>Machine learning Random Forest</i> dan <i>Gradient Boosting</i>	dari penggabungan kedua algoritma ini yaitu mendapatkan nilai RMSE 0.07%.
	Algoritma/Metode	<i>Random Forest</i> dan <i>Gradient Boosting</i>	
5	Peneliti (tahun)	(Yanisa Putri et al., 2024)	Penelitian ini menggunakan data dari website <i>FemaleDaily</i> untuk membangun sistem rekomendasi produk. Model <i>Content-Based Filtering</i> tidak memerlukan data dari pengguna, sedangkan <i>Collaborative Filtering</i> bergantung pada ulasan pengguna. Hasil akurasi menunjukkan bahwa <i>Content-Based Filtering</i> memiliki skor akurasi tinggi sebesar 80%, sementara <i>Collaborative Filtering</i> hanya mencapai skor 20%.
	Judul	Sistem Rekomendasi <i>Skincare</i> Menggunakan Metode <i>Content Based Filtering</i> dan <i>Coolaborative Filtering</i>	
	Algoritma/Metode	<i>Content-Based Filtering</i> dan <i>Collaborative Filtering</i>	
6	Peneliti (tahun)	(Sari et al., 2024)	Penelitian ini menggunakan dataset dari <i>Kaggle</i> dengan variabel harga laptop dan spesifikasi untuk mengimplementasikan model <i>regresi linear</i> . Hasil evaluasi menunjukkan nilai R^2 sebesar 0.68% dan <i>Mean Absolute Error</i> (MAE) sebesar 0.18. Pengembangan lebih lanjut disarankan dengan memperluas validasi model.
	Judul	Implementasi <i>Machine learning</i> untuk Prediksi Harga Laptop Menggunakan Algoritma Regresi Linear Berganda	
	Algoritma/Metode	Algoritma <i>Regresi Linear</i> Berganda.	

7	Peneliti (tahun)	(Radithya et al., 2024)	Pada penelitian ini, data yang digunakan diambil dari dataset yang telah tersedia di platform pembelajaran Udemy. Untuk variabel yang digunakan pada prediksi target penjualan menggunakan <i>decision tree</i> ini yaitu berdasarkan profil dan penghasilan. Untuk hasil akurasi pada prediksi target penjualan sebesar 91%.
	Judul	Prediksi Target Penjualan Mobil Menggunakan <i>Decision Tree</i>	
	Algoritma/Metode	<i>Decision Tree</i>	
8	Peneliti (tahun)	(Nasyuli et al., 2023)	Penelitian ini menggunakan dataset dari <i>Kaggle</i> untuk mengimplementasikan model <i>Gradient Boosting</i> dan Linear Regression dalam memprediksi harga mobil. Prosesnya melibatkan <i>encoding</i> menggunakan <i>Linear Regression</i> , diikuti evaluasi dengan <i>Gradient Boosting</i> yang menghasilkan akurasi RMSE sebesar 5.81% dan MAPE sebesar 7.5%. Pengembangan penelitian selanjutnya disarankan dengan menambahkan beberapa variabel lain.
	Judul	Penerapan Model <i>Machine learning</i> Algoritma <i>Gradient Boosting</i> dan <i>Linear Regression</i> Melakukan Prediksi Harga Kendaran Bekas	
	Algoritma/Metode	<i>Gradient Boosting</i> dan <i>Linear Regression</i>	
9	Peneliti (tahun)	(Wijaya et al., 2024)	Penelitian ini menggunakan dataset publik dari <i>Kaggle</i> dan mengoptimasi model <i>XGBoost</i> dengan kombinasi <i>Random Search</i> untuk meningkatkan akurasi prediksi. Hasil akurasi menunjukkan nilai R^2 sebesar 97,2%, MSE sebesar 30,9%, dan MAE sebesar 10,2%. Namun, hasil tersebut dianggap kurang efektif karena
	Judul	Meningkatkan Prediksi Penjualan Retail Xyz Dengan Teknik Optimasi <i>Random Search</i> Pada Model <i>XGBoost</i>	

	Algoritma/Metode	<i>XGBoost</i> dan <i>Random Search</i>	teknik optimasi terbatas pada <i>Random Search</i> dan dataset yang kurang variatif
10	Peneliti (tahun)	(Tamami & Arifin, 2024)	Penelitian ini menggunakan dataset penjualan tahunan dari <i>Kaggle</i> dengan pembagian data 80% untuk <i>training</i> dan 20% untuk <i>testing</i>. Variabel yang digunakan adalah tanggal (<i>date</i>) dan penjualan (<i>sales</i>). Model <i>Long Short-Term Memory</i> (LSTM) menghasilkan akurasi untuk kategori <i>Low-End</i> dengan MSE sebesar 83.3%, RMSE sebesar 91.2%, dan R^2 sebesar 0.82%; kategori <i>Mid-End</i> dengan MSE sebesar 50.1%, RMSE sebesar 22.3%, dan R^2 sebesar 0.98%; serta kategori <i>High-End</i> dengan MSE sebesar 93.7%, RMSE sebesar 96.8%, dan R^2 sebesar 0.84%.
	Judul	Penggunaan LSTM dalam Membangun Prediksi Penjualan Untuk Aplikasi Laptop Lens	
	Algoritma/Metode	<i>Long Short-Term Memory</i>	
11	Peneliti (tahun)	(Rayhan Rizal Mahendra et al., 2024)	Penelitian ini bertujuan mengimplementasikan <i>metode Item-Based Collaborative Filtering</i> pada dataset rekomendasi produk. Dataset yang digunakan berasal dari <i>Kaggle</i>, berupa data toko <i>e-commerce</i> dengan total 100.000 data yang mencakup user ID, movie ID, dan rating. Hasil evaluasi model menunjukkan performa yang baik berdasarkan matriks MAE dan RMSE, namun disarankan untuk menggunakan pendekatan metode lain guna meningkatkan hasil.
	Judul	Implementasi <i>Item-Based Collaborative Filtering</i> Untuk Rekomendasi Film	
	Algoritma/Metode	<i>Collaborative Filtering</i>	
12	Peneliti (tahun)	(Nuris, 2024)	

	Judul	Analisis Prediksi Harga Rumah Pada <i>Machine learning</i> Menggunakan Metode Regresi Linear	Berdasarkan kutipan penelitian analisis prediksi harga rumah, data yang diperoleh bersumber dari rumah123.com dengan total data 361. Variabel yang digunakan dalam pengimplementasian algoritma <i>regresi linear</i> yaitu luas tanah dan harga jual rumah. Didapatkan hasil akurasi dari model ini yaitu MSE 46.4%, R^2 0.78 %, MAE 31.3 %.
	Algoritma/Metode	<i>Regresi Linear</i>	
13	Peneliti (tahun)	(Negara & Mardiansyah, 2024)	Penelitian ini membagi dataset menjadi dua bagian, yaitu 80% untuk data train dan 20% untuk data test. Hasil implementasi kedua model menunjukkan bahwa <i>Content-Based Filtering</i> menggunakan teknik <i>bag of words</i> dan <i>cosine similarity</i> memberikan hasil yang memadai untuk halaman detail tempat wisata. Sementara itu, model <i>Collaborative Filtering</i> berhasil memberikan hasil memuaskan dengan nilai error yang rendah pada data train dan data test, yaitu 0.1162 dan 0.1215 secara berturut-turut.
	Judul	Implementasi <i>Machine learning</i> Dengan Metode <i>Collaborative Filtering</i> Dan <i>Content-Based Filtering</i> Pada Aplikasi Mobile Travel (Bangkit Academy)	
	Algoritma/Metode	<i>Collaborative Filtering</i> dan <i>Content-Based Filtering</i>	
14	Peneliti (tahun)	(Nouvalina & Hati, 2024)	Penelitian ini menggunakan dataset hasil web scraping dari berbagai website brand kosmetik dengan total lebih dari 1200 produk skin care. Metode <i>Content-Based Filtering</i> diterapkan dalam sistem rekomendasi tanpa feedback, menggunakan TF-IDF dan <i>Cosine Similarity</i> . Hasil rekomendasi menunjukkan bahwa
	Judul	Sistem Rekomendasi Produk <i>Skin Care</i> Berdasarkan Permasalahan Kulit Wajah Dengan Metode <i>Content Based Filtering</i>	

	Algoritma/Metode	<i>Content Based Filtering</i>	sistem mampu memberikan produk yang sesuai dengan kebutuhan konsumen.
15	Peneliti (tahun)	(Saputro & Amin, 2024)	Penelitian ini bertujuan memberikan rekomendasi produk <i>skincare face wash</i> pria yang mirip dengan produk yang biasanya digunakan. Data yang digunakan bersumber dari <i>official store</i> di <i>e-commerce Shopee</i>, terdiri dari 15 produk dari berbagai merek. Hasil dari model <i>Content-Based Filtering</i> menunjukkan nilai similaritas 0,5611, yang menunjukkan bahwa <i>Scarlet Whitening Facial Wash</i> adalah produk dengan akurasi tertinggi dibandingkan produk lainnya.
	Judul	Sistem Rekomendasi <i>Content-Based Filtering Skincare Pria di E-Commerce Shopee</i>	
	Algoritma/Metode	<i>Content-Based Filtering</i>	
16	Peneliti (tahun)	(Oktavian & Amin, 2023)	Penelitian ini menggunakan dataset yang diperoleh melalui web scraping dari <i>website Els Computer Shop</i>, dengan tujuan memberikan tiga rekomendasi laptop menggunakan metode <i>Content-Based Filtering</i> berdasarkan perhitungan <i>cosine similarity</i>. Hasil dari metode ini cukup baik, menghasilkan tiga rekomendasi laptop berdasarkan indeks kemiripan. Namun, untuk penelitian selanjutnya, disarankan untuk menggunakan dataset yang lebih besar agar implementasi model <i>Content-Based Filtering</i> dapat lebih maksimal, karena penelitian ini hanya menggunakan 20 data laptop.
	Judul	Perancangan Sistem Rekomendasi Laptop dengan Model <i>Prototyping</i> dan Penerapan <i>Content-Based Filtering Approach</i> pada ELS <i>Computer Shop Semarang</i>	
	Algoritma/Metode	<i>Content-Based Filtering</i>	
17	Peneliti (tahun)	(Madani et al., 2024)	

	Judul	Sistem Rekomendasi Musik Menggunakan <i>Machine learning</i>	Berdasarkan hasil kutipan pada penelitian ini, dataset ini memiliki data sebanyak 2391 data. Tujuan dari penelitian ini yaitu memberikan rekomendasi musik kepada pengguna berdasarkan prefensi dari pengguna itu sendiri. Untuk hasil akurasi menggunakan model <i>Collaborative Filtering</i> dapat memberikan rekomendasi yang akurat dan relevan bagi pengguna, yang ditunjukkan dari hasil akurasi mendapatkan nilai MAE dan RMSE yang rendah.
	Algoritma/Metode	<i>Collaborative Filtering</i>	
18	Peneliti (tahun)	(Anwari & Nurma Yulita, 2023)	Penelitian ini bertujuan untuk mengetahui metode ARIMA untuk melakukan prediksi harga telur ayam ras di masa depan. Dataset yang digunakan bersumber dari website resmi kemendagri. Untuk hasil akurasi prediksi dengan menggunakan metode ARIMA menghasilkan nilai RMSE 0.82% dan nilai R^2 yaitu 0.87% dapat disimpulkan metode ARIMA dalam memprediksi harga telur dapat dikatakan cukup.
	Judul	Penggunaan <i>Machine learning</i> Untuk Prediksi Harga Telur Ayam Ras di Kota Bandung	
	Algoritma/Metode	ARIMA	
19	Peneliti (tahun)	(Kusuma & Hidayat, 2024)	Berdasarkan Kutipan dari penelitian ini mengenai prediksi harga mobil menggunakan model <i>Regresi Linier</i> bertujuan untuk meningkatkan pemahaman mengenai pasar mobil bekas di negara
	Judul	Penerapan Model <i>Regresi Linier</i> dalam Prediksi Harga Mobil Bekas	

		di India dan Visualisasi dengan Menggunakan Power BI	tersebut. Dataset yang digunakan pada penelitian ini bersumber dari situs <i>kaggle</i> . Untuk hasil akurasi prediksi mengenai harga mobil tersebut dengan menggunakan metode Regresi Linier yaitu 71.09%.
	Algoritma/Metode	<i>Regresi Linier</i>	
20	Peneliti (tahun)	(Pratama et al., 2023)	Pada peneltian ini berfokus pada menemukan algoritma terbaik antara algoritma <i>Linear Regresi</i> dan <i>Random Forest</i> dalam memprediksi harga bawang merah. Dataset yang digunakan dalam penelitian ini didapatkan dari Dinas Perdagangan dan UMKM Kota Samarinda yang berisi 152 data yang terdiri dari 4 variabel. Hasil dari akurasi prediksi harga bawang untuk <i>model Regresi Linear</i> mendapatkan nilai RMSE 53.7% dan algoritma <i>Random Forest</i> mendapatkan nilai RMSE 84.1%.
	Judul	Perbandingan Performa Algoritma <i>Linear Regresi</i> dan <i>Random Forest</i> Untuk Prediksi Harga Bawang Merah di Kota Samarinda	
	Algoritma/Metode	<i>Linear Regresi</i> dan <i>Random Forest</i>	

Tabel 2.3, merangkum berbagai penelitian yang relevan dengan penelitian ini dalam hal prediksi harga dan rekomendasi. Berbagai algoritma *machine learning* telah digunakan, mulai dari model *regresi* sederhana hingga model *ensemble* seperti *Random Forest*, *XGBoost*, dan *Gradient Boosting*. Penelitian (Sari et al., 2024), dan (Nuris, 2024) menggunakan regresi linear untuk memprediksi harga laptop dan rumah, dengan hasil akurasi yang baik namun masih memerlukan validasi lebih luas. *Random Forest* digunakan (Warjiyono et al., 2024) dan (Pratama et al., 2023), menunjukkan akurasi tinggi dalam prediksi harga, namun perlu perbaikan dalam aspek optimasi.

Model lain seperti LSTM dan *XGBoost* yang dioptimasi dengan teknik *random search* digunakan (Tamami & Arifin, 2024), menghasilkan peningkatan akurasi yang signifikan. Pendekatan optimasi seperti *feature scaling* juga diterapkan (Radithya et al., 2024) pada *Decision Tree* dan mencapai akurasi hingga 91%. Sementara itu, (Simamora et al., 2024) menunjukkan bahwa kombinasi model *Random Forest* dan *Gradient Boosting* dapat meningkatkan performa prediksi secara signifikan. Dalam konteks rekomendasi, *collaborative filtering* yang digunakan (Yanisa Putri et al., 2024) kurang efektif dalam merekomendasikan produk.

Mengacu pada temuan penelitian sebelumnya, penelitian ini mengimplementasikan base model, hasil akurasi awal model yang cukup akurat dilakukan tahapan optimasi menggunakan *feature importance* untuk mengevaluasi kontribusi variabel terhadap hasil prediksi, serta *feature selection* untuk memilih fitur yang paling relevan. Dalam rekomendasi, metode *content-based filtering*

digunakan untuk memberikan rekomendasi produk yang lebih relevan. Seluruh hasil prediksi dan rekomendasi kemudian divisualisasikan menggunakan *Tableau* agar lebih mudah dipahami dalam menyusun strategi pemasarannya.

2.3.Matrix Penelitian

Tabel 2. 4 Matrix Penelitian

No	Penelitian	Algoritma	Tujuan Penelitian			
			Perbandingan Algoritma	Pengujian Variabel	Optimalisasi Algoritma	Sistem Rekomendasi
1	(Warjiyono et al., 2024)	<i>Random Forest</i>	-	✓	-	-
2	(Radithya et al., 2024)	<i>Decision Tree + Feature Scalling</i>	-	-	✓	-
3	(Hidayah et al., 2024)	<i>Random Forest, Regresi Linear, dan Decision Tree</i>	✓	✓	✓	-
4	(Simamora et al., 2024)	<i>Random Forest, dan Gradient Boosting</i>	-	-	✓	-
5	(Yanisa Putri et al., 2024)	<i>Collaborative Filtering</i>	-	-	-	✓
6	(Sari et al., 2024)	<i>Regresi Linear</i>	-	-	-	-

7	(Radithya et al., 2024)	<i>Decision Tree</i>	-	-	-	-
8	(Nasyuli et al., 2023)	<i>Gradient Boosting dan Regresi Linear</i>	✓	-	-	-
9	(Wijaya et al., 2024)	<i>XGBoost</i>	-	-	✓	-
10	(Tamami & Arifin, 2024)	LSTM	-	✓	-	-
11	(Rayhan Rizal Mahendra et al., 2024)	<i>Collaborative Filtering</i>	-	-	-	✓
12	(Nuris, 2024)	<i>Regresi Linear</i>	-	✓	-	-
13	(Negara & Mardiansyah, 2024)	<i>Collaborative Filtering, dan Content-Based Filtering</i>	-	-	-	✓
14	(Nouvalina & Hati, 2024)	<i>Content-Based Filtering</i>	-	-	-	✓
15	(Saputro & Amin, 2024)	<i>Content-Based Filtering</i>	-	-	-	✓
16	(Oktavian & Amin, 2023)	<i>Content-Based Filtering</i>	-	-	-	✓
17	(Madani et al., 2024)	<i>Collaborative Filtering</i>	-	-	-	✓

18	(Anwari & Nurma Yulita, 2023)	ARIMA	-	-	-	-
19	(Kusuma & Hidayat, 2024)	<i>Regresi Linear</i>	-	✓	-	-
20	(Aditya Pratama et al., 2024)	<i>Random Forest</i> , dan <i>Regresi Linear</i>	✓	✓	-	-
21	<i>Our research</i>	<i>Random Forest</i> , <i>Decision Tree</i> , <i>Gradient Boosting</i> , <i>XGBoost</i> , <i>Linear Regresi</i> , <i>Content Based</i> + <i>Optimasi</i>	✓	✓	✓	✓

Tabel 2.4, dapat disimpulkan bahwa penelitian yang dilakukan memiliki keterkaitan paling dekat dengan penelitian (Warjiyono et al., 2024) yang menggunakan algoritma *Random Forest* untuk prediksi harga, namun belum menerapkan teknik optimasi terhadap model untuk meningkatkan performa akurasi model. Selain itu, (Radithya et al., 2024) mengoptimasi *Decision Tree* dengan *feature scaling* untuk meningkatkan akurasi, tetapi hasilnya perlu ditingkatkan kembali. Pada penelitian (Yanisa Putri et al., 2024), metode *collaborative filtering*, masih belum cukup relevan dalam merekomendasikan produk.

Merujuk pada keterbatasan dari penelitian sebelumnya, penelitian ini bertujuan untuk memberikan kontribusi dan mengatasi permasalahan pada penelitian terdahulu, dengan menguji beberapa algoritma, seperti *Random Forest*, *XGBoost*, *Gradient Boosting*, *Decision Tree*, dan *Regresi Linear*. Model dengan akurasi awal yang cukup akurat, kemudian dioptimasi menggunakan teknik *feature importance* dan *feature selection* guna meningkatkan performa prediksi, sebagai alternatif dari teknik optimasi *feature scaling* yang digunakan pada penelitian sebelumnya. Selain itu, dalam rekomendasi, penelitian ini menggunakan *content-based filtering* yang dianggap mampu memberikan rekomendasi produk lebih relevan dibandingkan metode *collaborative filtering*.