

Laporan UAS Kecerdasan Buatan



OLEH :

Azay Agustian

NIM 231011402845

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS PAMULANG**

PAMULANG

2026

Pendahuluan

Kecerdasan Buatan (Artificial Intelligence) merupakan salah satu bidang dalam ilmu komputer yang mempelajari bagaimana membuat sistem atau mesin mampu meniru kecerdasan manusia, seperti belajar dari data, mengambil keputusan, dan melakukan prediksi. Salah satu teknik yang umum digunakan dalam kecerdasan buatan adalah *machine learning*.

Pada tugas UAS ini, digunakan metode *Decision Tree* untuk menyelesaikan permasalahan klasifikasi menggunakan **dataset Iris**. Dataset ini dipilih karena bersifat sederhana, tidak memiliki missing value, dan sering digunakan sebagai dataset pengenalan dalam pembelajaran *machine learning*. Tujuan dari tugas ini adalah memahami konsep Decision Tree, mengimplementasikannya menggunakan Python, serta menganalisis performa model yang dihasilkan.

Bagian 1 – Pemahaman Konsep (Teori)

1.1 Pengertian Decision Tree

Decision Tree adalah metode *machine learning* berbasis pohon keputusan yang digunakan untuk menyelesaikan masalah klasifikasi maupun regresi. Model ini bekerja dengan cara membagi data berdasarkan fitur tertentu hingga mencapai keputusan akhir dalam bentuk kelas atau nilai tertentu.

1.2 Konsep dalam Decision Tree

- **Node**

Node adalah titik pada pohon yang merepresentasikan proses pengambilan keputusan berdasarkan suatu fitur.

- **Root**

Root merupakan node paling atas pada pohon keputusan yang menjadi awal proses pembagian data.

- **Leaf**

Leaf adalah node akhir yang tidak memiliki cabang lagi dan berisi hasil prediksi kelas.

- **Splitting**

Splitting adalah proses membagi data ke dalam beberapa cabang berdasarkan nilai fitur tertentu.

- **Pruning**

Pruning adalah teknik untuk memangkas cabang pohon yang tidak terlalu penting guna mencegah *overfitting*.

1.3 Perbedaan Decision Tree, Random Forest, dan Gradient Boosting

- **DecisionTree**

Menggunakan satu pohon keputusan. Mudah dipahami tetapi rentan terhadap *overfitting*.

- **RandomForest**

Menggabungkan banyak Decision Tree dan mengambil keputusan berdasarkan hasil mayoritas. Lebih stabil dan akurat.

- **GradientBoosting**

Membuat pohon secara bertahap dengan memperbaiki kesalahan model sebelumnya. Akurat tetapi lebih kompleks dan membutuhkan waktu komputasi lebih lama.

1.4 Kelebihan dan Kekurangan Tree-Based Methods

Kelebihan:

- Mudah dipahami dan divisualisasikan
- Tidak memerlukan normalisasi data
- Dapat menangani data numerik dan kategorikal

Kekurangan:

- Rentan terhadap *overfitting*
- Sensitif terhadap perubahan data
- Kurang optimal untuk dataset yang sangat besar tanpa optimasi

Bagian 2 – Implementasi Model

2.1 Dataset

Dataset yang digunakan adalah **Iris Dataset** dari pustaka scikit-learn. Dataset ini memiliki 150 data dengan 4 fitur, yaitu:

- Sepal length
- Sepal width
- Petal length
- Petal width

Target klasifikasi terdiri dari tiga kelas: *setosa*, *versicolor*, dan *virginica*.

2.2 Preprocessing Data

Pada dataset Iris tidak ditemukan missing value sehingga tidak diperlukan proses penanganan data kosong. Seluruh fitur sudah berbentuk numerik sehingga proses encoding juga tidak diperlukan.

2.3 Pembagian Data

Dataset dibagi menjadi:

- **Training set:** 80%
- **Testing set:** 20%

Pembagian ini dilakukan untuk melatih model dan menguji performa model terhadap data yang belum pernah dilihat sebelumnya.

2.4 Pembuatan Model Decision Tree

Model Decision Tree dibangun menggunakan DecisionTreeClassifier dengan parameter:

- criterion = 'gini'
- max_depth = 3
- random_state = 42

Parameter ini digunakan untuk membatasi kedalaman pohon agar menghindari overfitting.

2.5 Evaluasi Model

Model dievaluasi menggunakan metrik:

- Accuracy
- Precision
- Recall
- F1-score

[Gambar 1. Hasil Evaluasi Model (Accuracy & Classification Report)]

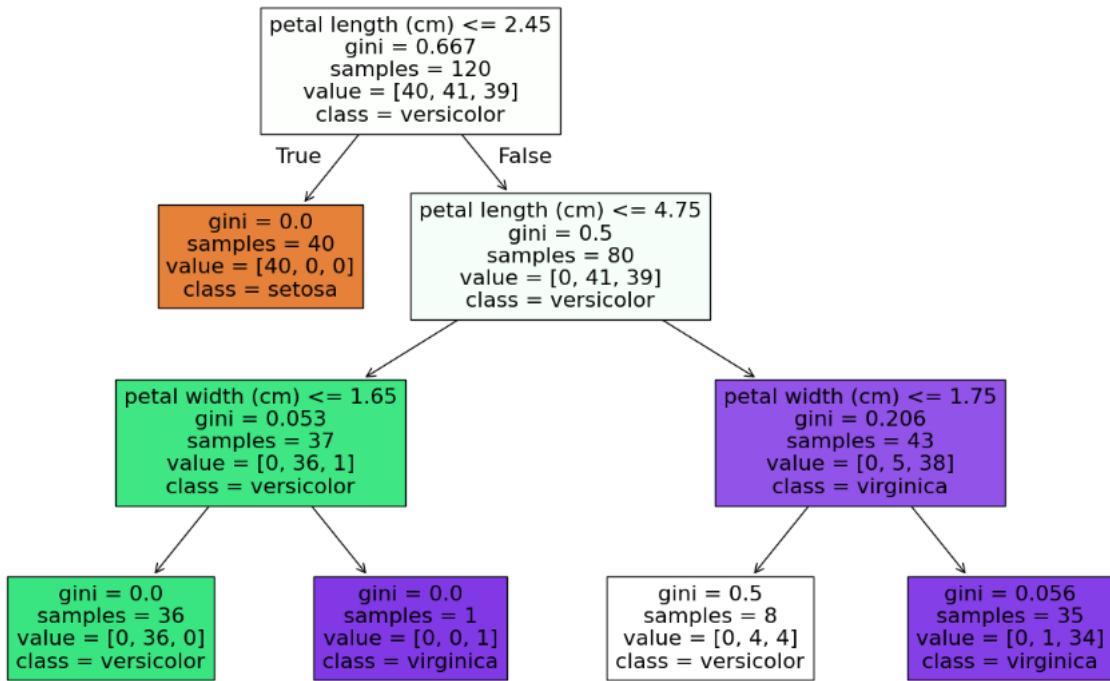
Classification Report:				
	precision	recall	f1-score	support
setosa	1.00	1.00	1.00	10
versicolor	1.00	1.00	1.00	9
virginica	1.00	1.00	1.00	11
accuracy			1.00	30
macro avg	1.00	1.00	1.00	30
weighted avg	1.00	1.00	1.00	30

Hasil evaluasi menunjukkan bahwa model memiliki akurasi yang sangat tinggi pada data uji.

2.6 Visualisasi Pohon Keputusan

Visualisasi pohon keputusan dilakukan untuk melihat bagaimana model mengambil keputusan berdasarkan fitur-fitur yang ada.

[Gambar 2. Visualisasi Pohon Keputusan Dataset Iris]



Dari visualisasi terlihat bahwa fitur *petal length* dan *petal width* memiliki peran penting dalam proses klasifikasi.

Bagian 3 – Analisis dan Kesimpulan

3.1 Model Terbaik

Berdasarkan eksperimen yang dilakukan, model Decision Tree dengan parameter `max_depth = 3` memberikan performa terbaik dengan hasil akurasi tinggi dan struktur pohon yang masih mudah dipahami.

3.2 Faktor yang Mempengaruhi Performa Model

Beberapa faktor yang mempengaruhi performa model antara lain:

- Pemilihan fitur
- Kedalaman pohon
- Pembagian data training dan testing
- Karakteristik dataset

3.3 Kelebihan Tree-Based Methods pada Studi Kasus

Pada dataset Iris, Decision Tree sangat efektif karena:

- Dataset bersih dan sederhana
- Jumlah fitur sedikit
- Pola antar kelas cukup jelas

3.4 Kesimpulan Akhir

Decision Tree merupakan metode yang efektif dan mudah dipahami untuk permasalahan klasifikasi. Pada studi kasus dataset Iris, model mampu mengklasifikasikan data dengan akurasi tinggi. Visualisasi pohon keputusan juga membantu dalam memahami proses pengambilan keputusan oleh model. Metode ini cocok digunakan sebagai dasar pembelajaran *machine learning* sebelum beralih ke metode yang lebih kompleks.

Link Repository GitHub:

<https://github.com/JayyLogic/UAS-Kecerdasan-Buatan-Decision-Tree.git>