

Perbandingan Metode Decision Tree Classifier dan XGBoost Classifier Dalam Memprediksi Penyakit Jantung

Sausan*¹, Desty Mayang Pratiwi², Lulu Mufidah³

Fakultas Informatika, Universitas Telkom Purwokerto
Jalan D.I. Panjaitan No. 128, Kecamatan Purwokerto Selatan, Kota Purwokerto, Jawa Tengah, Indonesia

¹sausans@student.telkomuniversity.ac.id

³lulumufidah@student.telkomuniversity.ac.id

²destymayangpratiwi@student.telkomuniversity.ac.id

Dikirim pada 17-10-2024, Direvisi pada 28-10-2024, Diterima pada 10-12-2024

Abstrak

Penyakit jantung merupakan penyebab utama kematian di seluruh dunia, didorong oleh gaya hidup tidak sehat serta faktor fisiologis seperti hipertensi dan kolesterol tinggi. Deteksi dini menjadi tantangan utama, karena sering kali pasien baru menyadari keberadaan penyakit ini pada tahap lanjut. Penelitian ini berfokus pada pengembangan sistem prediksi berbasis *machine learning* dalam rangka membantu dalam mendiagnosis penyakit jantung secara lebih akurat dan cepat. Dua algoritma, Decision Tree Classifier dan XGBoost Classifier, diterapkan dan dibandingkan efektivitasnya dalam memprediksi risiko penyakit jantung. Dataset *UCI Machine Learning Repository*, yang berisi 920 data observasi dengan 14 fitur, digunakan sebagai sumber data. Melalui serangkaian langkah mulai dari *preprocessing*, *encoding*, hingga teknik *oversampling*, dataset dipersiapkan untuk dianalisis. Data kemudian dipisahkan menjadi data latih dan data uji, sebelum dievaluasi dengan metrik akurasi, *precision*, *recall*, dan *F1-score*. Hasil penelitian mengindikasikan bahwa algoritma XGBoost mencapai akurasi 93%, lebih unggul dibandingkan Decision Tree yang memperoleh akurasi 90%. Dengan demikian, XGBoost direkomendasikan sebagai alternatif yang lebih efektif untuk mendukung proses diagnosis dini yang lebih tepat dan efisien. Diharapkan bahwa hasil penelitian ini dapat berkontribusi dalam pengembangan metode deteksi dini yang lebih efektif dalam layanan kesehatan untuk mengurangi angka kematian akibat penyakit jantung.

Kata Kunci: Decision Tree, Penyakit Jantung, Prediksi, XGBoost Classifier

Ini adalah artikel akses terbuka di bawah lisensi [CC BY-SA](#).



Penulis Koresponden:

Sausan

Universitas Telkom Kampus Purwokerto, Jl. D.I Panjaitan No.128 Purwokerto, 53147 Kabupaten Banyumas, Jawa Tengah, Indonesia
Email: sausans@student.telkomuniversity.ac.id

I. PENDAHULUAN

Penyakit jantung merupakan salah satu masalah kesehatan global utama dan menjadi kematian yang paling didominasi di seluruh dunia [1]. Menurut data dari *World Health Organization* (WHO), penyakit jantung menyumbang sekitar 31% dari total kematian global [1]. Kondisi jantung yang terganggu, termasuk kardiovaskular, jantung koroner, dan serangan jantung, seringkali dipicu oleh gaya hidup tidak sehat. Faktor-faktor seperti kebiasaan merokok, pola makan yang tidak baik, konsumsi alkohol dan kafein berlebihan, begadang, serta stres memberikan dampak negatif pada kesehatan jantung [2]. Di samping itu, faktor-faktor fisiologis seperti hipertensi, kolesterol, obesitas, darah tinggi juga turut memperburuk risiko individu terhadap penyakit ini [3]. Pengobatan penyakit jantung tersedia dalam beberapa metode antara lain operasi, penyinaran, dan kemoterapi [2]. Namun masalah yang timbul dari penyakit jantung adalah sulitnya

deteksi dini, yang sering kali membuat pasien baru menyadari keberadaan penyakit ketika sudah memasuki tahap yang serius dan berbahaya [2].

Tingginya angka kejadian serta sulitnya deteksi dini penyakit jantung mendorong perlunya pendekatan teknologi untuk membantu tenaga medis dalam pencegahan dan pengobatan yang lebih efektif. Diagnosis yang cepat dan tepat dapat meningkatkan peluang pencegahan komplikasi serius pada pasien dengan resiko tinggi [4]. Metode konvensional, misalnya konsultasi medis dan tes laboratorium masih membutuhkan biaya yang besar [5], sehingga pendekatan alternatif yang lebih efisien sangat diperlukan. Dalam konteks ini, teknologi khususnya pembelajaran mesin (*machine learning*) memiliki peran yang semakin penting [6]. Dengan menggunakan kemampuan *machine learning* membuat, memproses, dan menganalisis data pada jumlah besar, sistem prediksi berbasis teknologi ini mampu membuat prediksi yang seksama dan bisa diandalkan pada saat yang lebih singkat. Teknologi ini memungkinkan tenaga medis untuk dapat lebih mudah mengenali pola dan gejala yang berhubungan dengan risiko penyakit jantung, yang pada akhirnya membantu tenaga medis pada pengambilan keputusan yang lebih tepat terkait kesehatan pasien [1].

Algoritma seperti Decision Tree dan XGBoost telah diterapkan dalam prediksi medis dengan hasil yang menjanjikan dalam mengklasifikasikan data kesehatan secara efisien [7]. Decision Tree Classifier ialah metode yang memakai struktur pohon, yang mana setiap node nya melakukan pengujian terhadap suatu fitur, setiap cabang menunjukkan *output* menurut pengujian tersebut, dan setiap node daun menggambarkan label kelas tertentu [8]. Dalam *text mining*, Decision Tree digunakan untuk menemukan pola atau fungsi-fungsi yang dapat menggambarkan dan membedakan antar kelas data, sehingga memungkinkan prediksi kelas data untuk data baru yang belum diklasifikasikan [9]. Sedangkan XGBoost Classifier adalah algoritma pembelajaran mesin yang diterapkan untuk regresi dan klasifikasi pada dataset besar [10]. Algoritma ini menggunakan pohon keputusan dangkal yang dibangun secara berurutan untuk memberikan hasil yang akurat dan metode pelatihan yang sangat *scalable* yang menghindari *overfitting*.

Pada penelitian terdahulu, berbagai algoritma *machine learning* telah diterapkan untuk prediksi penyakit. Studi oleh Permana dan Dewi (2021) melakukan prediksi terhadap pasien diabetes memakai algoritma klasifikasi Decision Tree dan Naive Bayes [11]. Hasilnya menunjukkan bahwa Decision Tree memberikan performa yang lebih baik, dengan akurasi mencapai 95,58% dan nilai AUC sebesar 0,981, yang lebih unggul dibandingkan Naive Bayes yang hanya memiliki akurasi 87,69% dan AUC 0,947. Keunggulan metode Decision Tree juga terlihat pada kemampuannya untuk mengatasi masalah *overfitting*, menangani *missing value* pada atribut data latih, dan meningkatkan efisiensi komputasi secara keseluruhan [12]. Selain itu, penelitian yang dilakukan oleh Nasution, Saedudin, dan Widartha (2021) membandingkan akurasi antara algoritma Naive Bayes dan XGBoost dalam prediksi penyakit diabetes [13]. Hasil penelitian menunjukkan bahwa metode XGBoost memiliki tingkat akurasi yang lebih unggul dibandingkan Naive Bayes. Evaluasi akurasi dilakukan melalui analisis *confusion matrix* dan penghitungan nilai AUC. Dari hasil klasifikasi, Algoritma Naive Bayes mencapai akurasi sebesar 79,68%, sedangkan Algoritma XGBoost menunjukkan performa yang lebih baik dengan akurasi sebesar 90,10%. Kedua temuan ini menunjukkan bahwa kedua algoritma, yaitu Decision Tree dan XGBoost memiliki potensi untuk meningkatkan akurasi dalam prediksi penyakit jantung [11][13], berkat kemampuan keduanya dalam mengolah data dengan kompleksitas tinggi. Oleh karena itu, penelitian ini diharapkan bisa memberikan kontribusi yang signifikan dengan membandingkan akurasi dan efektivitas kedua metode dalam prediksi penyakit jantung.

Pesatnya perkembangan teknologi informasi di berbagai sektor, termasuk bidang medis telah mendorong penerapan sistem prediksi berbasis *machine learning* untuk memprediksi berbagai penyakit [2] [14]. Sistem prediksi ini memungkinkan para praktisi kesehatan untuk memperkirakan kemungkinan terjadinya penyakit berdasarkan data historis, sehingga potensi risiko dapat diidentifikasi lebih dini dan manfaat dari pencegahan dini dapat dioptimalkan. Penelitian ini bertujuan untuk mengembangkan model prediksi penyakit jantung yang lebih akurat melalui perbandingan metode Decision Tree dan XGBoost dalam hal akurasi serta efektivitas. Dataset UCI *Machine Learning* digunakan sebagai sumber data untuk analisis dalam penelitian ini. Melalui pendekatan yang diusulkan, diharapkan sistem prediksi berbasis *machine learning* ini dapat mendukung praktisi kesehatan dalam mengidentifikasi risiko penyakit jantung dengan lebih cepat dan lebih akurat yang memungkinkan pasien untuk menerima intervensi preventif lebih awal. Penelitian ini diharapkan mampu berkontribusi secara signifikan dalam pengembangan metode prediksi penyakit jantung yang efisien, yang dapat diintegrasikan ke dalam layanan kesehatan guna mengurangi angka kematian akibat penyakit jantung secara signifikan.

II. METODE PENELITIAN

Penelitian ini memanfaatkan metode eksperimen untuk mengembangkan sistem prediksi penyakit jantung berdasarkan data yang diambil dari *UCI Machine Learning Repository*. Dataset ini berisi data numerik terkait diagnosis penyakit jantung, yang diproses menggunakan bahasa pemrograman Python dengan bantuan berbagai library. Dua metode pembelajaran mesin, yaitu Decision Tree dan XGBoost, diterapkan untuk prediksi penyakit jantung. Dataset yang diperoleh dipisahkan berdasarkan kriteria yang telah ditetapkan, lalu dilakukan pengujian dengan kedua metode tersebut guna mengevaluasi kinerja masing-masing.

A. Prosedur Penelitian

Metode yang diterapkan dalam penelitian ini terdiri dari langkah-langkah sistematis dan logis yang bertujuan untuk memberikan pedoman serta mempermudah analisis perbandingan antara metode Decision Tree Classifier dan XGBoost Classifier, seperti yang terlihat pada Gambar 1 berikut.



Tabel 1. Diagram Langkah-Langkah Analisis

Pada Gambar 1 terlihat langkah-langkah analisis yang diterapkan dalam penelitian ini dengan tujuan untuk membandingkan efektivitas metode Decision Tree dan XGBoost dalam prediksi penyakit jantung, sehingga dapat membantu mendukung diagnosa dini. Berdasarkan Gambar 1, diagram dimulai dengan tahap persiapan data, yaitu proses preprocessing yang mencakup encoding data kategori, penanganan nilai-nilai null, dan *oversampling*. Selanjutnya, dilakukan penentuan dan perhitungan model klasifikasi, yaitu Decision Tree dan XGBoost. Kemudian, analisis data dilakukan berdasarkan metrik evaluasi seperti akurasi, *precision*, *recall*, dan *F1-score* untuk menilai efektivitas masing-masing model dalam memprediksi penyakit jantung. Langkah terakhir adalah penentuan metode terbaik berdasarkan hasil evaluasi performa kedua model.

B. Decision Tree

Decision Tree ialah metode yang populer dan efektif untuk prediksi dan klasifikasi. Metode ini mengubah fakta menjadi pohon keputusan yang merepresentasikan aturan yang mudah dipahami [15]. Decision Tree merupakan struktur pohon yang menguji atribut untuk memprediksi *output*. Setiap simpul internal menunjukkan pengujian atribut, cabang mewakili hasil pengujian, dan simpul akhir memegang label kelas. Simpul akar (*root node*) terletak di bagian atas pohon. Akar pohon ditentukan berdasarkan *gain* tertinggi dari setiap atribut atau entropi terendah [15]. Proses ini dimulai dengan menghitung entropi menggunakan rumus pada Persamaan (1), lalu menghitung *gain* dengan rumus pada Persamaan (2) [16].

$$Entropy(S) = \sum_{i=0}^n -p_i * \log^2 p_i \quad (1)$$

$$Gain(S, A) = Entropy(S) - \sum_i^n \frac{|S_i|}{|S|} * Entropy(S_i) \quad (2)$$

C. XGBoost

Extreme Gradient Boosting atau XGBoost ialah salah satu metode *supervised learning* yang digunakan untuk klasifikasi maupun regresi [17]. Algoritma ini merupakan sistem *machine learning* pada *tree boosting* yang dapat dioptimalkan untuk membangun pohon dengan skala lebih besar. Sebagai pengembangan dari Gradient Boosting, XGBoost adalah metode ensemble berbasis decision tree yang dirancang untuk mempercepat waktu proses, bahkan pada data berukuran besar [17]. XGBoost bekerja dengan menggabungkan berbagai pengklasifikasi lemah menjadi model yang lebih kuat, melalui pelatihan berurutan menggunakan hasil klasifikasi sebelumnya, yang disebut residuals atau *error*. Rumus XGBoost memperkenalkan regularisasi dalam fungsi objektif untuk mencegah *overfitting*, dengan fungsi objektif yang didefinisikan pada Persamaan (3) [18]:

$$O - \sum_{i=1}^n L(y_i, Fx_i) + \sum_{k=1}^t R(f_k) + C \quad (3)$$

Pada persamaan (3) terdapat beberapa penjelasan sebagai berikut :

- $L(y_i, Fx_i)$: Fungsi kerugian (loss function) yang berfungsi untuk mengukur tingkat akurasi model dalam memprediksi data
- $R(f_k)$: Istilah regularisasi yang berfungsi mencegah *overfitting*, diformulasikan sebagai $\alpha H + \frac{1}{2}n + \sum_{j=1}^H w_j^2$ dengan :
 - α : menyatakan tingkat kompleksitas pada daun.
 - H : menunjukkan jumlah daun dalam model.
 - n : mengindikasikan parameter penalti.
 - w_j^2 : mengacu pada *output* yang dihasilkan oleh setiap simpul daun.
- C : konstanta yang dapat diabaikan secara efektif.

D. Teknik Pengumpulan Dataset

Penelitian ini menggunakan data numerik yang berkaitan dengan diagnosis penyakit jantung, yang diambil dari *UCI Machine Learning Repository*. Dataset publik ini, berjudul "Heart Disease," tersedia secara *online* di laman <http://archive.ics.uci.edu/ml/datasets/Heart+Disease> [19]. Seluruh atribut dalam dataset memiliki nilai numerik dan dapat dianggap sebagai data klinis yang dikumpulkan dari empat lokasi berbeda terkait diagnosis penyakit jantung. Jumlah observasi dari masing-masing lokasi adalah 303 observasi dari Cleveland, 294 dari Hungarian, 123 dari Switzerland, dan 200 dari Long Beach VA, sehingga totalnya mencakup 920 observasi. Dataset yang digunakan berjumlah 920 dengan 14 fitur yang dijelaskan lebih lanjut dalam Tabel 1.

Tabel 1. Atribut Dataset

No	Atribut	Kategori	Deskripsi
1	Umur	-	Menunjukkan usia individu
2	Jenis Kelamin	1 = laki-laki 0 = perempuan	Menunjukkan jenis kelamin individu
3	Cp (Jenis Nyeri Dada)	1 = angina tipikal 2 = angina atipikal 3 = bukan nyeri angina 4 = asimtotik	Mengindikasikan jenis nyeri dada yang dialami oleh individu
4	Trestbps (Tekanan Darah)	-	Menunjukkan nilai tekanan darah dalam satuan mmHg
5	Chol (Kolesterol Serum)	-	Menunjukkan kadar kolesterol serum dalam mg/dl
6	Fbs (Gula Darah Puasa)	1 = jika gula darah puasa > 120mg/dl 0 = lainnya	Membandingkan kadar gula darah puasa seseorang terhadap batas 120 mg/dl
7	Restecg	0 = biasa 1 = memiliki kelainan gelombang ST-T 2 = hipertrofi ventrikel kiri	Menunjukkan hasil elektrokardiogram
8	Thalach (Detak Jantung Maksimum)	-	Menunjukkan detak jantung maksimum yang dicapai individu
9	Exang (Angina yang Dipicu Oleh Latihan)	1 = ya 0 = tidak	Menunjukkan apakah individu mengalami angina yang dipicu oleh latihan

No	Atribut	Kategori	Deskripsi
10	Oldpeak (Depresi ST yang Diinduksi Oleh Latihan)	-	Menunjukkan nilai depresi ST yang dapat berupa bilangan integer atau float
11	Slope (Slope dari Segmen ST Puncak Latihan)	1 = menanjak 2 = datar 3 = miring ke bawah	Menunjukkan bentuk kemiringan segmen ST pada puncak latihan
12	Ca (Jumlah Pembuluh Darah Utama yang Diberi Warna)	0-3 values	Menunjukkan jumlah pembuluh darah yang terlihat melalui fluoroskopi
13	Thal	3 = biasa 6 = cacat tetap 7 = cacat reversible	Menunjukkan tipe thalassemia pada individu
14	Num	0 = tidak ada penyakit 1-4 = tingkat keparahan penyakit	Menunjukkan tipe thalassemia pada individu

E. Confusion Matrix

Confusion matrix merupakan metode pengujian yang digunakan untuk mengetahui kinerja algoritma *machine learning* [20]. Dalam penelitian ini, pengujian sistem dilakukan untuk mengevaluasi akurasi model dengan menggunakan *confusion matrix*. Pengujian ini dilakukan berdasarkan tabel *confusion matrix* yang ditampilkan pada Tabel 2. Tabel ini digunakan untuk membandingkan kinerja antara metode Decision Tree Classifier dan XGBoost Classifier dalam memprediksi penyakit jantung.

Tabel 2 Confusion Matrix

Aktual	Prediksi	
	Sakit Jantung	Tidak Sakit Jantung
Sakit Jantung	TP	FN
Tidak Sakit Jantung	FP	TN

Berdasarkan Tabel 2, TP adalah jumlah data dengan nilai positif yang diprediksi positif. TN adalah jumlah data dengan nilai positif yang diprediksi negatif. FP adalah jumlah data dengan nilai negatif yang diprediksi positif. Sementara itu, FN adalah jumlah data dengan nilai negatif yang diprediksi negatif [21]. *Confusion matrix* mencakup beberapa perhitungan [22]:

a. Presisi

Presisi adalah metrik untuk menghitung proporsi data yang benar-benar positif dari seluruh prediksi yang dinyatakan positif. Perhitungan presisi dapat dilakukan menggunakan Persamaan (4) [21].

$$Presisi = \frac{TP}{FN + TP} \quad (4)$$

b. Recall

Recall ialah metrik untuk menghitung proporsi data yang diprediksi benar positif dari seluruh data yang seharusnya positif. Recall dihitung menggunakan Persamaan (5) [22].

$$Recall = \frac{TP}{TP + FN} \quad (5)$$

c. F1 Score

F1 Score ialah adalah metrik yang digunakan untuk menghitung rata-rata harmonis antara presisi dan recall. Perhitungan F1 Score dilakukan dengan menggunakan Persamaan (6) [22].

$$F1\ Score = 2 \times \frac{Recall \times Presisi}{Recall + Presisi} \quad (6)$$

d. Akurasi

Akurasi ialah metrik untuk mengukur sejauh mana model dapat melakukan klasifikasi dengan benar. Perhitungan akurasi dapat dilakukan menggunakan Persamaan (7) [21].

$$Akurasi = \frac{TN + TP}{TP + TN + FP + FN} \quad (7)$$

III. HASIL DAN PEMBAHASAN

A. Persiapan Data

1. Deskripsi Dataset

Penelitian ini menggunakan dataset penyakit jantung yang diambil dari *UCI Machine Learning Repository*. Dataset ini berisi informasi numerik terkait faktor-faktor yang berpengaruh terhadap penyakit jantung, seperti umur, jenis kelamin, tekanan darah, kolesterol, dan atribut medis lainnya. Informasi lebih lanjut mengenai data tersebut dapat dilihat pada Gambar 2.

	age	sex	cp	trestbps	chol	fbs	restecg	thalch	exang	oldpeak	slope	ca	thal	num
915	54	Female	asymptomatic	127.0	333.0	True	st-t abnormality	154.0	False	0.0	NaN	NaN	NaN	1
916	62	Male	typical angina	NaN	139.0	False	st-t abnormality	NaN	NaN	NaN	NaN	NaN	NaN	0
917	55	Male	asymptomatic	122.0	223.0	True	st-t abnormality	100.0	False	0.0	NaN	NaN	fixed defect	2
918	58	Male	asymptomatic	NaN	385.0	True	lv hypertrophy	NaN	NaN	NaN	NaN	NaN	NaN	0
919	62	Male	atypical angina	120.0	254.0	False	lv hypertrophy	93.0	True	0.0	NaN	NaN	NaN	1

Gambar 2. Data Penyakit Jantung

Dataset yang digunakan berbentuk file CSV yang terdiri dari 920 baris dengan 14 kolom data. Dalam penelitian ini, kolom "age", "sex", "cp", "trestbps", "chol", "fbs", "restecg", "thalach", "exang", "oldpeak", "slope", "ca", dan "thal" akan dijadikan variabel independen (X), sementara kolom "num" akan dijadikan variabel dependen (Y) atau variabel target.

2. Proses *Preprocessing* Data

Tahap awal adalah melakukan *preprocessing* pada data, termasuk pengubahan nilai pada kolom yang berisi data kategori menjadi nilai bilangan bulat menggunakan LabelEncoder. Proses ini penting untuk mengonversi data kategorikal ke format numerik yang dapat dipahami oleh model klasifikasi. Setelah proses encoding, data penyakit jantung menjadi siap untuk digunakan dalam pelatihan model klasifikasi. Perubahan ini dapat dilihat pada Gambar 3, yang memperlihatkan bagaimana nilai pada kolom kategori telah diubah menjadi format numerik.

	age	sex	cp	trestbps	chol	fbs	restecg	thalch	exang	oldpeak	slope	ca	thal	num
0	63	1	3	145.0	233.0	1	0	150.0	0	2.3	0	0.0	0	0
1	67	1	0	160.0	286.0	0	0	108.0	1	1.5	1	3.0	1	2
2	67	1	0	120.0	229.0	0	0	129.0	1	2.6	1	2.0	2	1
3	37	1	2	130.0	250.0	0	1	187.0	0	3.5	0	0.0	1	0
4	41	0	1	130.0	204.0	0	0	172.0	0	1.4	2	0.0	1	0

Gambar 3. Data Penyakit Jantung Setelah Proses Encoding

Pada tahap berikutnya, dilakukan penanganan terhadap nilai-nilai yang hilang (null) dalam dataset melalui proses imputasi. Nilai null ini dapat mengganggu kelengkapan analisis jika tidak diatasi dengan tepat. Beberapa teknik imputasi yang digunakan meliputi pengisian nilai null dengan rata-rata, median, atau metode statistik lainnya. Proses ini penting untuk menjaga kelengkapan dataset dan memastikan bahwa analisis dapat dilakukan tanpa gangguan. Gambar 4 memperlihatkan bagaimana proses imputasi diterapkan pada dataset penyakit jantung, yang membantu memastikan data yang digunakan lebih representatif dan akurat.

```
#inisialisasi IterativeImputer
imputer = IterativeImputer()

#menerapkan imputasi pada dataframe
df = pd.DataFrame(imputer.fit_transform(df), columns=df.columns)
```

Gambar 4. Proses Imputasi

Di samping itu, untuk mengatasi ketidakseimbangan kelas pada variabel target "num", yang dapat mempengaruhi kemampuan model untuk memprediksi kelas minoritas dengan baik, diterapkan teknik *oversampling* yang bertujuan meningkatkan jumlah sampel pada kelas minoritas. Pendekatan ini menggunakan metode seperti RandomOverSampler atau SMOTE (*Synthetic Minority Over-sampling Technique*) guna memperbaiki distribusi data dalam kelas target. Dengan menambahkan data pada kelas yang kurang terwakili, model klasifikasi menjadi lebih seimbang, sehingga mampu menghasilkan prediksi yang lebih akurat. Hasil penerapan *oversampling* ditampilkan pada Gambar 5, yang menunjukkan peningkatan representasi kelas minoritas dalam dataset.

```
#metode oversampling digunakan untuk penanganan ketidakseimbangan
ros = RandomOverSampler()
smote = SMOTE()
x,y = ros.fit_resample(x,y)
```

Gambar 5. Penerapan *Oversampling*

3. Tahap Pembagian Data (*Train-test Split*)

Setelah keseluruhan langkah *preprocessing* dilakukan, data kemudian dibagi ke dalam dua subset utama, yakni set pelatihan (*train set*) dan set pengujian (*test set*). Proses pembagian ini, yang dilakukan melalui *train-test split* dengan rasio 80:20, bertujuan untuk memisahkan 80% data guna melatih model dan 20% data lainnya untuk menguji kinerja model. Langkah ini bertujuan menghindari terjadinya *overfitting*, yaitu ketika model terlalu terlatih pada data pelatihan sehingga kurang mampu melakukan generalisasi pada data baru. Gambar 6 memperlihatkan tahapan pembagian data ini, yang memastikan model dapat dievaluasi secara objektif berdasarkan kemampuan memprediksi data yang belum pernah dilihat sebelumnya.

```
#membagi data menjadi set pelatihan dan pengujian
X_train, X_test, y_train, y_test = train_test_split(x, y, test_size=.2, shuffle=True, stratify=y )
```

Gambar 6. *Train-test Split*

B. Penentuan dan Perhitungan Model Klasifikasi

1. Metode Decision Tree Classifier

Model pertama yang digunakan adalah Decision Tree Classifier. Metode ini dibangun menggunakan dataset yang telah diproses melalui tahap *preprocessing*. Hasil *classification report* Decision Tree Classifier dengan Python terlihat pada Gambar 7.

	precision	recall	f1-score	support
0.0	0.90	0.77	0.83	82
1.0	0.78	0.76	0.77	82
2.0	0.91	0.99	0.95	82
3.0	0.90	0.99	0.94	82
4.0	1.00	1.00	1.00	83
accuracy			0.90	411
macro avg	0.90	0.90	0.90	411
weighted avg	0.90	0.90	0.90	411

Gambar 7. *Classification Report* Decision Tree Classifier

Dari hasil *classification report* pada Gambar 7, dapat dilihat bahwa model Decision Tree memiliki performa yang baik pada sebagian besar kelas, dengan *precision* dan *recall* yang cukup tinggi pada kelas-kelas tertentu, terutama kelas 2.0 dan kelas 4.0.

2. Metode XGBoost Classifier

Model kedua yang digunakan adalah XGBoost Classifier, yang merupakan model *boosting* yang lebih canggih. Metode ini dibangun menggunakan dataset yang telah diproses melalui tahap *preprocessing*. Hasil *classification report* XGBoost Classifier dengan Python terlihat pada Gambar 8.

	precision	recall	f1-score	support
0.0	0.91	0.91	0.91	82
1.0	0.90	0.73	0.81	82
2.0	0.92	0.99	0.95	82
3.0	0.91	1.00	0.95	82
4.0	0.99	1.00	0.99	83
accuracy			0.93	411
macro avg	0.93	0.93	0.92	411
weighted avg	0.93	0.93	0.92	411

Gambar 8. Classification Report XGBoost Classifier

Dari hasil *classification report* pada Gambar 8, dapat dilihat bahwa model XGBoost menunjukkan kinerja yang lebih unggul dibandingkan dengan Decision Tree pada sebagian besar kelas. Metrik yang dianalisis, seperti *precision* dan *recall*, menunjukkan nilai yang lebih tinggi pada XGBoost, dengan *F1-Score* yang lebih optimal untuk setiap kelas, terutama pada kelas 2.0 dan kelas 4.0.

C. Analisis Data dan Penentuan Metode Terbaik

Penelitian ini berfokus pada evaluasi model berdasarkan akurasi yang dicapai pada data uji., karena akurasi merupakan ukuran umum yang menunjukkan kemampuan model dalam menghasilkan prediksi yang tepat [23]. Metrik ini dipilih karena relevansinya dengan tujuan penelitian, yaitu mengidentifikasi model dengan prediksi yang paling akurat dalam mendeteksi kasus penyakit jantung. Evaluasi dilakukan dengan menggunakan data uji (*testing set*) yang telah dipisahkan sebelumnya melalui metode *train-test split*. Data uji ini tidak digunakan selama pelatihan model, sehingga hasil evaluasi mencerminkan kemampuan model dalam memprediksi data yang benar-benar baru. Tabel 2 menunjukkan perbandingan akurasi antara Decision Tree Classifier dan XGBoost Classifier.

TABEL 3. HASIL PREDIKSI

Metode	Akurasi
Decision Tree Classifier	90%
XGBoost Classifier	93%

Berdasarkan Tabel 2, XGBoost Classifier menunjukkan kinerja yang lebih baik dibandingkan Decision Tree Classifier, dengan akurasi mencapai 93%, sedangkan Decision Tree hanya 90%. Selain akurasi, hasil evaluasi berdasarkan metrik *precision*, *recall*, dan *F1-score* juga menunjukkan bahwa XGBoost lebih unggul pada hampir seluruh kelas. Nilai yang lebih tinggi pada XGBoost ini disebabkan oleh kemampuannya dalam menangani *overfitting* dan *imbalance* pada bobot fitur, yang sering menjadi kendala pada Decision Tree. Dengan demikian, XGBoost lebih efektif dalam memprediksi penyakit jantung pada dataset yang digunakan.

IV. KESIMPULAN

Penelitian ini menerapkan metode Decision Tree Classifier dan XGBoost Classifier untuk memprediksi penyakit jantung dengan menggunakan dataset dari *UCI Machine Learning Repository*. Hasil evaluasi menunjukkan bahwa XGBoost menunjukkan performa yang lebih unggul dibandingkan dengan Decision Tree dalam hal akurasi, *precision*, *recall*, dan *F1-score*. XGBoost mencapai akurasi sebesar 93%, sedangkan Decision Tree hanya mencapai 90%. Penelitian ini berfokus pada evaluasi model berdasarkan metrik akurasi karena akurasi merupakan indikator utama keberhasilan prediksi pada dataset yang digunakan. Precision dan recall juga diperhitungkan untuk menilai keseimbangan kemampuan model dalam mengidentifikasi kasus positif dan negatif. Hasilnya, precision dan recall XGBoost lebih tinggi yang menunjukkan bahwa model ini lebih handal dalam mengidentifikasi kasus penyakit jantung tanpa mengorbankan akurasi deteksi terhadap kasus negatif. Aspek lain, seperti waktu pemrosesan dan

penggunaan memori, tidak dievaluasi dalam penelitian ini karena berada di luar cakupan studi. Evaluasi terhadap aspek tersebut dapat menjadi bahan pertimbangan untuk penelitian di masa mendatang. Berdasarkan hasil evaluasi ini, XGBoost Classifier terbukti lebih unggul dalam memprediksi penyakit jantung, sehingga berpotensi menjadi alat yang lebih andal untuk mendukung tenaga medis dalam diagnosis dini penyakit jantung.

UCAPAN TERIMAKASIH

Penulis mengucapkan rasa syukur yang mendalam kepada Allah Subhanahu Wa Ta'ala atas segala rahmat dan petunjuk-Nya, yang memungkinkan penelitian ini dapat diselesaikan dengan baik. Ucapan terima kasih yang tulus juga disampaikan kepada LPPM Telkom University Purwokerto atas dukungan dana penelitian yang memungkinkan pelaksanaan penelitian ini. Penulis juga menyampaikan rasa terima kasih yang sebesar-besarnya kepada Bapak Andre selaku dosen pengampu atas bimbingan dan arahan yang sangat berarti, serta kepada rekan-rekan yang telah memberikan dukungan dan motivasi sepanjang proses penelitian. Terakhir, penulis mengucapkan terima kasih kepada keluarga tercinta yang selalu memberikan doa dan dorongan tanpa henti.

DAFTAR PUSTAKA

- [1] F. Handayani, "Komparasi Support Vector Machine, Logistic Regression Dan Artificial Neural Network Dalam Prediksi Penyakit Jantung," *J. Edukasi dan Penelit. Inform.*, vol. 7, no. 3, p. 329, 2021.
- [2] A. Sepharni, I. E. Hendrawan, and C. Rozikin, "Klasifikasi Penyakit Jantung dengan Menggunakan Algoritma C4.5," *STRING (Satuan Tulisan Ris. dan Inov. Teknol.)*, vol. 7, no. 2, p. 117, 2022.
- [3] Dimsyiar M Al Hafiz, Khoirul Amaly, Javen Jonathan, M Teranggono Rachmatullah, and Rosidi, "Sistem Prediksi Penyakit Jantung Menggunakan Metode Naive Bayes," *J. Rekayasa Elektro Sriwij.*, vol. 2, no. 2, pp. 151–157, 2021.
- [4] D. Arjun Vahlevy, E. Levis Putra Zentrato, R. Fadillah, and R. Jafar Sidiq, "Tinjauan Literatur Sistematis pada Sistem Pakar untuk Diagnosa Penyakit Manusia," *J. Artif. Intel. dan Sist. Penunjang Keputusan*, vol. 1, no. 1, pp. 1–8, 2023.
- [5] A. B. Wibisono and A. Fahrurrozi, "Perbandingan Algoritma Klasifikasi Dalam Pengklasifikasian Data Penyakit Jantung Koroner," *J. Ilm. Teknol. dan Rekayasa*, vol. 24, no. 3, pp. 161–170, 2019.
- [6] D. H. Depari, Y. Widiastiw, and M. M. Santoni, "Perbandingan Model Decision Tree, Naive Bayes dan Random Forest untuk Prediksi Klasifikasi Penyakit Jantung," *Inform. J. Ilmu Komput.*, vol. 18, no. 3, p. 239, 2022.
- [7] Ö. N. ERGÜN and H. O. İLHAN, "Early Stage Diabetes Prediction Using Machine Learning Methods," *Eur. J. Sci. Technol.*, no. 29, pp. 52–57, 2021.
- [8] A. Lailiyah, V. R. S. Nastiti, E. D. Wahyuni, and C. S. K. Aditya, "Klasifikasi Tingkat Kemampuan Adaptasi Siswa dalam Pembelajaran Online Menggunakan Decision Tree," *Techno.Com*, vol. 23, no. 1, pp. 11–19, 2024.
- [9] Yumi, Lukman, and R. Y. Bakti, "Klasifikasi Mahasiswa Berpotensi Putus Studi Menggunakan Algoritma Decision Tree pada Fakultas Teknik Unismuh Makassar," *Arus J. Sains dan Teknol. (AJST)*, vol. 2, no. 2, 2024.
- [10] K. Y. Nazara, "Perancangan Smart Predictive Maintenance untuk Mesin Produksi," *Semin. Nas. Off. Stat.*, vol. 2022, no. 1, pp. 691–702, 2022.
- [11] B. A. C. Permana and I. K. Dewi, "Komparasi Metode Klasifikasi Data Mining Decision Tree dan Naive Bayes Untuk Prediksi Penyakit Diabetes," *Infotek J. Inform. dan Teknol.*, vol. 4, no. 1, pp. 63–69, 2021.
- [12] M. I. Aziz, A. Z. Fanani, and A. Affandy, "Analisis Metode Ensemble Pada Klasifikasi Penyakit Jantung Berbasis Decision Tree," *J. Media Inform. Budidarma*, vol. 7, no. 1, p. 1, 2023.

-
- [13] M. K. Nasution, R. R. Saedudin, and V. P. Widartha, "Perbandingan Akurasi Algoritma Naïve Bayes Dan Algoritma Xgboost Pada Klasifikasi Penyakit Diabetes," *e-Proceeding Eng.*, vol. 8, no. 5, pp. 9765–9772, 2021.
- [14] M. B. Priyantono, A. A. Rachmawan, L. A. P. Budi, and K. C. Kirana, "Sistem Prediksi Gejala Virus Korona dengan Metode Forward Chaining," *JTERA (Jurnal Teknol. Rekayasa)*, vol. 5, no. 1, p. 111, 2020.
- [15] C. Cahyaningtyas, Y. Nataliani, and I. R. Widiyari, "Analisis Sentimen Pada Rating Aplikasi Shopee Menggunakan Metode Decision Tree Berbasis SMOTE," *Aiti*, vol. 18, no. 2, pp. 173–184, 2021.
- [16] N. Nurajjah, D. A. Ningtyas, and M. Wahyudi, "Klasifikasi Siswa Smk Berpotensi Putus Sekolah Menggunakan Algoritma Decision Tree, Support Vector Machine Dan Naive Bayes," *J. Khatulistiwa Inform.*, vol. 7, no. 2, pp. 85–90, 2019.
- [17] A. C. Nugraha and M. I. Irawan, "Komparasi Deteksi Kecurangan pada Data Klaim Asuransi Pelayanan Kesehatan Menggunakan Metode Support Vector Machine (SVM) dan Extreme Gradient Boosting (XGBoost)," *J. Sains dan Seni ITS*, vol. 12, no. 1, 2023.
- [18] K. Aditya, A. Wisnu, and A. M. A. Rahim, "Analisis Perbandingan Algoritma XGBoost Dan Algoritma Random Forest Untuk Klasifikasi Data Kesehatan Mental," *Log. J. Ilmu Komput. dan Pendidik.*, vol. 2, no. 5, pp. 808–818, 2024.
- [19] "Heart Disease," UCI Machine Learning Repository. Accessed: Jan. 28, 2024.
- [20] O. Terrada, B. Cherradi, A. Raihani, and O. Bouattane, "Classification and Prediction of Atherosclerosis Diseases Using Machine Learning Algorithms," *2019 Int. Conf. Optim. Appl. ICOA 2019*, pp. 1–5, 2019.
- [21] B. Santoso, "An Analysis of Spam Email Detection Performance Assessment Using Machine Learning," *J. Online Inform.*, vol. 4, no. 1, p. 53, 2019.
- [22] D. Derisma, "Perbandingan Kinerja Algoritma untuk Prediksi Penyakit Jantung dengan Teknik Data Mining," *J. Appl. Informatics Comput.*, vol. 4, no. 1, pp. 84–88, 2020.
- [23] A. A. Ramadhani, R. A. Saputra, and I. P. Ningrum, "Klasifikasi Tingkat Kepuasan Pengguna Google Classroom dalam Pembelajaran Online Menggunakan Algoritma Naïve Bayes," *JIKO (Jurnal Inform. dan Komputer)*, vol. 8, no. 2, p. 310, 2024.