KONVERGENSI
Volume 21, Nomor 2, Juli 2025: 47-54
e-ISSN: 2721-0936, p-ISSN: 1858-0688

Sistem Pendukung Keputusan Pemeriksaan Kesehatan Menggunakan Decision Tree Classifier

Mohamad Ardi[1,*], Eviana Tjatur Putri[2]
[1]Teknik Informatika, STMIK PPKIA Tarakanita Rahmawati Tarakan
[2]Sistem Informasi, STMIK PPKIA Tarakanita Rahmawati Tarakan

ABSTRACT

Health screening is a crucial process in medical practice, which often involves the analysis of vital signs such as body temperature, pulse rate, and other clinical indicators. This study focuses on the development of a decision support system for health screening using the Decision Tree Classifier method. This system aims to provide an alternative to traditional methods by leveraging computational efficiency and simplicity. Data from 4,210 patient records collected in April and May 2020 were used to train and validate the model. This method produces an accuracy value for the training data of 0.97 or it can be said that the accuracy rate of the training data for this method is 97%, indicating its potential for reliable health diagnostics. This article outlines the methodological framework, including data preprocessing, model training, and validation. In addition, the computational process of the Decision Tree Classifier is detailed to improve reproducibility. This system is designed to assist healthcare professionals in making informed decisions while reducing time and resource constraints. The results show that the Decision Tree Classifier is an effective method for supporting health screening decisions, which offers high accuracy and ease of implementation.

Keywords: *Decision Support System, Decision Tree Classifier, Health Screening*

ABSTRAK

Pemeriksaan kesehatan merupakan proses krusial dalam praktik medis, yang sering kali melibatkan analisis tanda-tanda vital seperti suhu tubuh, denyut nadi, dan indikator klinis lainnya. Penelitian ini berfokus pada pengembangan sistem pendukung keputusan untuk pemeriksaan kesehatan menggunakan metode Decision Tree Classifier. Sistem ini bertujuan untuk memberikan alternatif bagi metode tradisional dengan memanfaatkan efisiensi dan kesederhanaan komputasi. Data dari 4.210 catatan pasien yang dikumpulkan pada bulan April dan Mei 2020 digunakan untuk melatih dan memvalidasi model. Metode ini menghasilkan nilai akurasi untuk data latih adalah 0,97 atau dapat dikatakan bahwa tingkat akurasi data latih untuk metode ini adalah 97%, hal ini menunjukkan potensinya untuk diagnostik kesehatan yang andal. Artikel ini menguraikan kerangka metodologis, termasuk praproses data, pelatihan model, dan validasi. Selain itu, proses komputasi Decision Tree Classifier dirinci untuk meningkatkan reproduktifitas. Sistem ini dirancang untuk membantu para profesional kesehatan dalam membuat keputusan yang tepat sekaligus mengurangi kendala waktu dan sumber daya. Hasil penelitian menunjukkan bahwa Decision Tree Classifier merupakan metode yang efektif untuk mendukung keputusan pemeriksaan kesehatan, yang menawarkan akurasi tinggi dan kemudahan implementasi..

Kata Kunci: Sistem Pendukung Keputusan, Decision Tree Classifier, Pemeriksaan Kesehatan

Corresponding Author E-mail: mohamadardi@ppkia.ac.id *
Received Januari 2025; revised Februari 2025; accepted Maret 2025; published Juli 2025

1. Pendahuluan

Kemajuan teknologi canggih diikuti dengan perkembangan zaman yang semakin modern. Berbagai macam teknologi telah diciptakan untuk membantu kehidupan manusia dalam menjalankan aktivitas di bidang [1].

Sistem Pendukung Keputusan (SPK) semakin penting dalam perawatan kesehatan, menawarkan wawasan berbasis data untuk meningkatkan akurasi diagnostik dan hasil pasien. Metode pemeriksaan kesehatan tradisional sering kali melibatkan pengukuran manual dan penilaian subjektif, yang dapat memakan waktu dan rawan kesalahan. Pohon Keputusan (*Decision Tree*), telah menjadi metode yang efektif untuk mengotomatiskan dan meningkatkan proses diagnostik [2]. Salah satu jenis dari Pohon Keputusan adalah Decision Tree Classifier terkenal karena kesederhanaan, efektivitas, dan interpretabilitasnya saat bekerja dengan data numerik dan kategoris [3].

Decision Tree Classifier beroperasi dengan membuat struktur seperti pohon yang memudahkan pengambilan keputusan dengan membagi kumpulan data menjadi subset sesuai dengan ambang batas fitur [4]. Pohon Keputusan ini telah diterapkan secara luas di berbagai bidang, termasuk perawatan kesehatan yang membantu diagnosis penyakit [5][6].

Pohon Keputusan telah terbukti bermanfaat dalam diagnostik medis dalam penelitian terkini. Misalnya, metode ini telah digunakan untuk mengklasifikasikan diabetes [7], memprediksi penyakit kardiovaskular [8], dan menganalisis tingkat readmisi pasien [9]. Metode ini juga digunakan untuk mengklasifikasikan data stroke [10].

Decision Tree Classifier selain memiliki kelebihan juga memiliki keterbatasan. Masalah seperti overfitting dan sensitivitas terhadap data [11][12]. Teknik seperti pemangkasan dan validasi silang sering digunakan untuk mengurangi tantangan ini, memastikan model yang kuat dan dapat digeneralisasi [13][14].

Tujuan dari penelitian ini adalah untuk membuat SPK pemeriksaan kesehatan dengan memanfaatkan keunggulan Decision Tree Classifier. Dengan memperhatikan tanda-tanda vital tubuh seperti suhu tubuh, dan denyut nadi. sistem ini berupaya untuk menyediakan alternatif yang andal dan efisien untuk metode diagnostik manual. Pendekatan yang diusulkan sejalan dengan semakin meningkatnya penekanan pada solusi perawatan kesehatan berbasis data [15][16].

Kontribusi dari penelitian ini adalah pembuatan SPK yang disesuaikan dengan pemeriksaan kesehatan dan pendalaman terperinci dari proses komputasi Decision Tree Classifier. Upaya ini bertujuan untuk menjembatani kesenjangan pemberian layanan kesehatan antara praktik tradisional dan modern [17][18].

Penelitian ini bertujuan mengembangkan SPK berbasis Decision Tree Classifier untuk mengotomatisasi pemeriksaan kesehatan berdasarkan tanda-tanda vital (suhu tubuh, denyut nadi, usia, dan jenis kelamin). Sistem ini dirancang untuk mengatasi keterbatasan metode manual, seperti subjektivitas dalam penilaian klinis, waktu yang lama untuk analisis data pasien, dan risiko kesalahan manusia (human error). Penanganan masalah seperti overfitting ditangani dengan teknik pemangkasan pohon (pruning) dan validasi silang (cross-validation) untuk memastikan model tidak terlalu kompleks.

2. Tinjauan Pustaka

Penelitian ini didasarkan pada beberapa penelitian sebelumnya [5-10]. Penulis menggunakan topik yang sama dengan penelitian sebelumnya yakni pemeriksaan kesehatan, tetapi dengan objek dan metode penelitian yang berbeda. Objek yang digunakan dalam penelitian ini adalah tanda-tanda vital tubuh yaitu suhu tubuh, dan denyut nadi. Selain itu, objek lainnya adalah usia, dan jenis kelamin. Sedangkan, metode yang digunakan adalah Decision Tree Classifier. Untuk Pengolahan datanya dilakukan dengan bahasa pemrograman Python menggunakan software Jupyter Notebook.

2.1. *Decision Tree Classifier*

Decision Tree Classifier ialah metode yang populer dan efektif untuk prediksi dan klasifikasi. Metode ini mengubah fakta menjadi pohon keputusan yang merepresentasikan aturan yang mudah dipahami [5][19]. *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 [5][19]. Proses ini dimulai dengan menghitung entropi menggunakan rumus pada persamaan (1), lalu menghitung *gain* dengan rumus pada persamaan (2) [5][20].

$$\text{Entropy}\ (S) = \sum_{i=1}^{n} - p\text{i} \times \log_2 p\text{i} \quad (1)$$

$$\text{Gain}\ (S, A) = \text{Entropy}\ (S) - \sum_{i=1}^{n} \frac{|\text{Si}|}{|\text{S}|} \times \text{Entropy}\ (S\text{i}) \quad (2)$$

2.2. *Confusion Matrix*

Confusion Matrix adalah metode yang digunakan untuk menghitung akurasi dalam suatu konsep klasifikasi. Evaluasi dengan confusion matrix menghasilkan beberapa matriks seperti nilai *accuracy*, *precision*, *recall*, dan *f*1-*score*.

Accuracy merupakan persentase ketepatan data yang telah diklasifikasikan dengan benar. Precision

mengacu pada proporsi kasus yang diprediksi positif dan benar positif dalam data. Recall meng-gambarkan proporsi kasus positif yang berhasil diprediksi dengan tepat [21][22]. *F*1-*Score* adalah metrik yang digunakan untuk menghitung rata-rata harmonis antara precision dan recall [5][23]. Adapun rumus *accuracy* dapat dilihat pada persamaan (3) [11][24]. rumus *precision*, *recall*, dan *f*1-*score* dapat dilihat pada persamaan (4), (5), dan (6) [11][25].

$$Accuracy = \frac{TN + TP}{TN + TP + FP + FN} \quad (3)$$

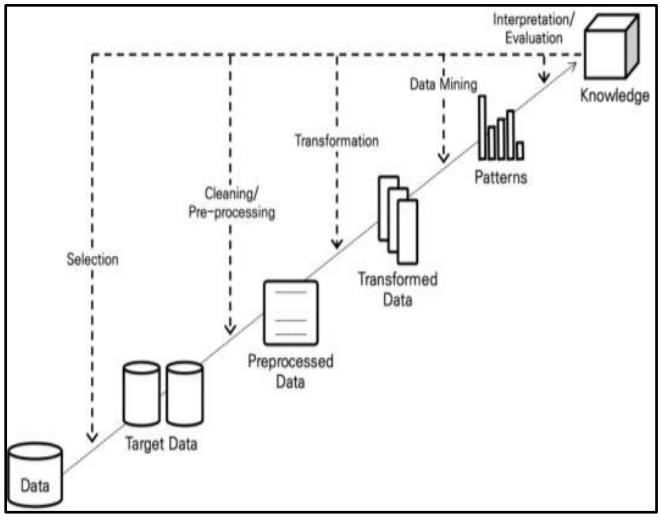
$$Precision = \frac{TP}{TP + FP} \quad (4)$$

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

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

3. Metode

Penelitian ini mengaplikasikan metode Decision Tree Classifier dalam metodologi yang digunakan. Riset ini memanfaatkan metode penelitian eksperimen untuk mencapai tingkat akurasi terbaik, yang diterapkan dalam tahapan mengembangkan SPK pemeriksaan kesehatan berdasarkan data yang diambil dari UPTD Rumah Sakit Umum Daerah Tarakan, sedangkan metode Knowledge Discovery in Database (KDD) dipergunakan dalam analisis data. Detail dari tahapan analisis data tertera pada Gambar 1 [11][26].



Gambar 1. Tahapan KDD

3.1. *Selection*

Tahap *Selection* merupakan tahap pertama dalam metode analisa KDD, dilakukan proses seleksi dari data awal sebanyak 5867, data rekam medis pasien dari UPTD Rumah Sakit Umum Daerah Kota Tarakan, pada tahun 2020. Data tersebut berisi 7 atribut, yakni No, Jenis Kelamin, Usia, Denyut Nadi, Suhu Tubuh, Type Pasien, dan Status Kesehatan. Pelabelan data pada atribut Status Kesehatan terdiri dari Sehat, Tidak Sehat atau Kritis, pelabelan data tersebut berdasarkan data sekunder dari dokumen arsip data pasien yang ada di ruang Rekam Medis dan ruang MCU UPTD Rumah Sakit Umum Daerah Kota Tarakan. Dokumen arsip data pasien tersebut diisi oleh dokter yang menangani pasien sesuai dengan tanggal yang tercantum pada dokumen. Setelah itu, dilakukan proses cleaning terhadap data yang tidak lengkap atau data outlier yang mempengaruhi proses analisis. Hasil dari proses selection diperoleh 4210 data.

3.2. *Preprocessing*

Pada tahap *Preprocessing* mencakup pembersihan data (penghapusan *missing values*), penanganan *missing values* dilakukan dengan data nilai kosong dihapus (*listwise deletion*) selanjutnya dilakukan tahap eliminasi redundansi, menghapus data yang memiliki *value* atribut yang sama. Hasil akhir diperoleh 3691 data. Tabel 1 menunjukkan sebagian dataset dari hasil *Preprocessing* data pasien yang telah dilakukan.

Tabel 1. Hasil *Preprocessing* Data

Jenis Kelamin	Usia (Thn)	Denyut Nadi (BPM)	Suhu Tubuh (°C)	Status Kesehatan
Perempuan	17	100	36	Kritis
Perempuan	17	104	36	Tidak Sehat
Perempuan	17	106	36	Tidak Sehat
Laki-Laki	17	96	36,2	Kritis
Laki-Laki	17	98	36	Sehat
...	...	...	...	...
Laki-Laki	25	89	36	Tidak Sehat
Laki-Laki	25	90	36	Sehat
Laki-Laki	25	90	36,5	Kritis
Perempuan	22	98	36,8	Kritis
Perempuan	22	100	36	Tidak Sehat
...	...	...	...	...
Laki-Laki	69	99	36,3	Kritis
Laki-Laki	70	96	36,3	Kritis
Laki-Laki	71	83	36	Tidak Sehat
Perempuan	33	76	36	Sehat
Perempuan	33	76	36	Tidak Sehat

Berdasarkan *Exploratory Data Analysis* (EDA), distribusi kelas Sehat sebesar 44,1% (1628 data pasien), Tidak Sehat 30,4% (1122 data pasien), Kritis 25,5% (941 data pasien). Statistik deskriptifnya untuk rata-rata suhu tubuh 36,1°C dengan standar deviasi 0,45, hal ini menunjukkan mayoritas suhu tubuh berada di antara 35,65°C – 36,55°C artinya dalam kisaran normal. untuk denyut nadi rata-rata 85 bpm dengan standar deviasi 17, hal ini menunjukkan mayoritas berada di kisaran 68 bpm – 102 bpm artinya denyut nadi berada dalam variasi

sedang hingga tinggi (denyut nadi normal 60 bpm - 100 bpm).

Tahap berikutnya adalah *data reduction*, dengan memilih atribut yang paling relevan tanpa menghilangkan informasi yang penting. Atribut yang dihilangkan adalah No dan Type Pasien.

3.3. *Transformation*

Selanjutnya masuk ketahap *Transformation*, tipe atribut data yang bertipe kategori akan dikonversi menjadi tipe numerik. Klasifikasi data mining hanya dapat dilakukan dengan menggunakan tipe atribut numerik, sehingga tipe atribut kategori harus dikonversi menjadi tipe numerik sebelum melangkah ke tahap selanjutnya dalam data mining. Atribut yang diubah pada Tabel 1 adalah Jenis Kelamin, untuk Perempuan diubah menjadi 0 dan Laki-Laki diubah menjadi 1. Tabel 2 menunjukkan hasil *Transformation* data yang telah dilakukan.

Tabel 2. Hasil *Transformation* Data

Jenis Kelamin	Usia (Tahun)	Denyut Nadi (BPM)	Suhu Tubuh (°C)	Status Kesehatan
0	17	100	36	Kritis
0	17	104	36	Tidak Sehat
0	17	106	36	Tidak Sehat
1	17	96	36,2	Kritis
1	17	98	36	Sehat
...	...	...	...	...
1	25	89	36	Tidak Sehat
1	25	90	36	Sehat
1	25	90	36,5	Kritis
0	22	98	36,8	Kritis
0	22	100	36	Tidak Sehat
...	...	...	...	...
1	69	99	36,3	Kritis
1	70	96	36,3	Kritis
1	71	83	36	Tidak Sehat
0	33	76	36	Sehat
0	33	76	36	Tidak Sehat

3.4. *Data Mining*

Pada tahap *Data Mining*, analisis dan perhitungan akan dilakukan menggunakan metode *Decision Tree Classifier*. Aplikasi *Jupyter Notebook* dengan *library phyton pandas*, serta diolah dengan *library matplotlib* digunakan pada tahap ini, untuk memastikan bahwa data yang akan dianalisis memenuhi syarat.

Dalam tahap ini, dataset akan dibagi menjadi 80% *training* (2953 data) dan 20% *testing* (738 data) dengan stratifikasi untuk mempertahankan distribusi kelas. Implementasi menggunakan bahasa pemrograman phyton yaitu *train_test_split* dari *scikit-learn*. Kemudian, dilanjutkan dengan menerapkan library python untuk metode *Decision Tree Classifier*.

3.5. *Evaluation*

Pada tahap *Evaluation*, akan dilakukan penilaian terhadap aturan yang sudah diperoleh sesuai dengan pola dan tahap yang telah ditentukan sebelumnya. Dalam tahap ini, aplikasi *Jupyter Notebook* akan digunakan untuk menentukan hasil *evaluation* yang diperoleh melalui metode *Decision Tree Classifier*. Hasil evaluasi ini akan ditentukan dengan menggunakan library python *accuracy_score*, *confusion_matrix*, dan *classification_report*.

4.1. Proses Perhitungan *Decision Tree Classifier*

Pengolahan data dengan metode *Decision Tree Classifier* dilakukan dengan bahasa pemrograman python menggunakan *software Jupyter Notebook*. Klasifikasi data terdiri dari beberapa Langkah sebagai berikut.

4.1.1. *Import Library*

Pada tahap awal implementasi, proses diawali dengan pemanggilan dan pemuatan pustaka (*library*) yang dibutuhkan dalam lingkungan Jupyter Notebook. Seluruh pustaka yang digunakan dipastikan telah terinstal dan dapat dijalankan tanpa kesalahan sintaksis maupun dependensi. Salah satu pustaka yang digunakan adalah *sklearn.metrics*, yang dimanfaatkan untuk menghitung nilai *f-measure* atau *F*1-*score* sebagai metrik evaluasi kinerja model. Gambar 2 menunjukkan potongan program untuk mengimpor *library* dalam metode *Decision Tree Classifier*.

```
#Impor pustaka
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn import datasets
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix
from sklearn.metrics import classification_report
```

Gambar 2. Kode *import library*

4.1.2. Unggah Data Pasien

Data penelitian diunggah ke dalam lingkungan Jupyter Notebook dengan memanfaatkan pustaka *pandas* untuk proses pembacaan dan pengelolaan data, sedangkan pustaka *NumPy* digunakan dalam pembentukan dan pengolahan struktur data berupa array atau matriks. Dataset "datapasien80%.csv" digunakan sebagai data pelatihan (*training data*). Cuplikan kode program yang menggambarkan proses pemuatan data pasien sebagai data pelatihan ditunjukkan pada Gambar 3.

```
#Mengambil data dari file csv
datapasien = pd.read_csv('datapasien80%.csv')
df = pd.DataFrame(datapasien, columns=['JK','Usia','BPM','Suhu','Status'])
X=np.asarray(datapasien)
x_train=X[:,:-1]
y_train=X[:,-1]
```

Gambar 3. Kode unggah data pasien

4.1.3. Memanggil Metode

Metode *Decision Tree Classifier* diimplementasikan menggunakan pustaka *scikit-learn* yang tersedia dalam lingkungan pemrograman Python. Pustaka ini dimanfaatkan untuk membangun model klasifikasi serta melakukan proses prediksi berdasarkan data pelatihan yang telah disiapkan sebelumnya. Implementasi model dilakukan melalui pemanggilan fungsi *Decision Tree Classifier* pada lingkungan Jupyter Notebook. Cuplikan kode program yang menunjukkan proses pemanggilan dan penerapan metode *Decision Tree Classifier* disajikan pada Gambar 4.

```
#Memanggil Decision Tree
dtc = DecisionTreeClassifier()
#Fit model dengan data latih dan target/label
dtc.fit(x_train,y_train)
```

Gambar 4. Kode memanggil metode

Pada penggunaan *class DecisionTreeClassifier* pada Gambar 4, parameter yang digunakan adalah paramater *default* yang dimiliki oleh class tersebut, antara lain parameter criterion = 'gini', splitter = 'best', dan max_depth = 'none'.

4.1.4. Akurasi Data *Training*

Untuk memperoleh tingkat presisi yang tinggi pada data uji, maka data training harus dipersiapkan dan diuji tingkat akurasi terlebih dahulu, menggunakan perhitungan akurasi skor. Pada penelitian ini data training yang telah dipersiapkan, diuji menggunakan metode *dtc.predict*. Perhitungan nilai akurasi data training menggunakan dtc.predict dapat dilihat pada Gambar 5. Dari perhitungan tersebut diperoleh nilai akurasi skor sebesar 97% yang memiliki arti bahwa data training tersebut memiliki akurasi yang baik, untuk memprediksi data uji pada metode ini.

```
#Akurasi data training
y_pred =dtc.predict(x_train)
print(accuracy_score(y_train, y_pred))

0.9673396674584323
```

Gambar 5. Akurasi data *training*

4.1.5. Unggah Data Validasi

Kinerja dan tingkat keakuratan metode *Decision Tree Classifier* dievaluasi menggunakan data uji validasi. Dataset "datapasien20%.csv" digunakan sebagai data uji (*testing data*) untuk menilai kemampuan model dalam melakukan klasifikasi terhadap data yang belum pernah dilatih sebelumnya. Cuplikan kode program yang menggambarkan proses pemuatan data validasi sebagai data uji pada metode *Decision Tree Classifier* ditampilkan pada Gambar 6.

```
#Mengambil data dari file csv
datavalidasi = pd.read_csv('datapasien20%.csv')
df2 = pd.DataFrame(datavalidasi, columns=['JK','Usia','BPM','Suhu','Status'])
X2 = np.asarray(datavalidasi)
x_val=X2[:,:-1]
y_val=X2[:,-1]
```

Gambar 6. Kode unggah data validasi

4.1.6. Akurasi Data Uji

Tahap selanjutnya adalah melakukan perhitungan tingkat akurasi terhadap data uji (*testing data*) guna mengetahui kemampuan metode Decision Tree Classifier dalam melakukan klasifikasi secara tepat. Perhitungan akurasi dilakukan berdasarkan hasil prediksi model terhadap data uji yang digunakan. Cuplikan kode program untuk proses perhitungan nilai akurasi data uji ditampilkan pada Gambar 7. Berdasarkan hasil pengujian tersebut, diperoleh nilai akurasi sebesar 0,67, yang menunjukkan bahwa tingkat keakuratan metode ini dalam mengklasifikasikan data uji mencapai 67%.

```
#Mengambil data dari file csv
datavalidasi = pd.read_csv('datapasien20%.csv')
df2 = pd.DataFrame(datavalidasi, columns=['JK','Usia','BPM','Suhu','Status'])
X2 = np.asarray(datavalidasi)
x_val=X2[:,:-1]
y_val=X2[:,-1]
```

Gambar 7. Kode akurasi data uji

4.1.7. Klasifikasi dan Prediksi *Report*

Evaluasi kinerja model dilakukan melalui analisis laporan klasifikasi yang mencakup metrik *precision*, *recall*, dan *f-measure* atau *F1-score*. *Precision* merepresentasikan tingkat ketepatan sistem dalam memberikan respons yang relevan terhadap data yang diklasifikasikan, sedangkan *recall* menunjukkan kemampuan sistem dalam menemukan kembali seluruh data yang seharusnya terklasifikasi dengan benar. Adapun *F1-score* digunakan sebagai ukuran keseimbangan antara *precision* dan *recall*. Selain itu, hasil prediksi model juga dianalisis berdasarkan keluaran yang dihasilkan saat eksekusi sel atau keseluruhan kode program. Cuplikan kode program yang menampilkan proses perhitungan laporan klasifikasi dan hasil prediksi menggunakan metode Decision Tree Classifier disajikan pada Gambar 8.

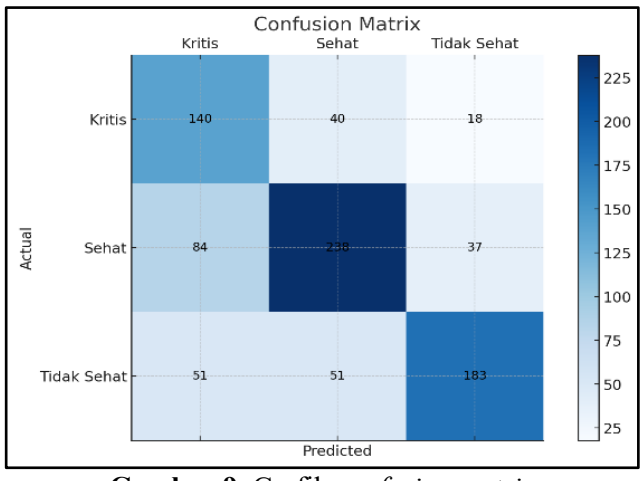
Pada Gambar 8, nilai *precision* untuk status Kritis adalah 0,51, status Sehat adalah 0,72, dan status Tidak

Sehat adalah 0,77. Sementara nilai *recall* untuk status Kritis adalah 0,71, status Sehat adalah 0,66, dan status Tidak Sehat adalah 0,64. Nilai *akurasi f1-score* untuk metode ini adalah 0,67. Selanjutnya, dilakukan prediksi dengan data pasien baru yang menghasilkan nilai prediksi "Tidak Sehat".

```
print(confusion_matrix(y_val, y_pred))

[[140  40  18]
 [ 84 238  37]
 [ 51  51 183]]

print(classification_report(y_val, y_pred))

              precision    recall  f1-score   support

      Kritis       0.51      0.71      0.59       198
       Sehat       0.72      0.66      0.69       359
 Tidak Sehat       0.77      0.64      0.70       285

    accuracy                           0.67       842
   macro avg       0.67      0.67      0.66       842
weighted avg       0.69      0.67      0.67       842


data=[[0,56,89,36]]
prediksi=dtc.predict(data)
print(prediksi)

['Tidak Sehat']
```

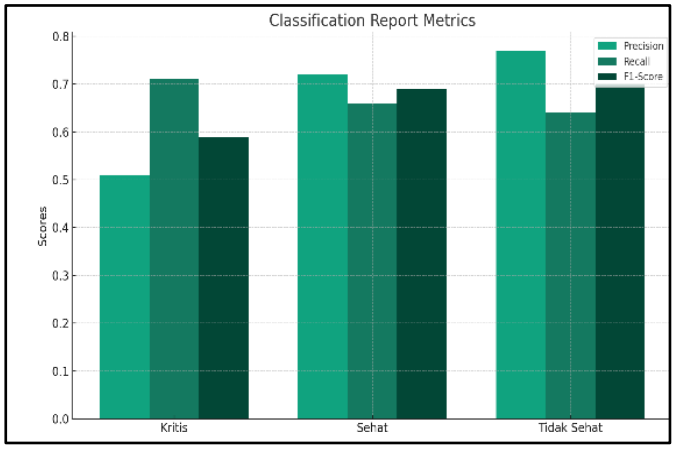
Gambar 8. Klasifikasi dan prediksi *report*

Pada hasil *confusion matrix* di atas diperoleh informasi sebagai berikut: Baris pertama, 140 merupakan prediksi benar sebagai Kritis, 40 merupakan salah prediksi sebagai Sehat (*False Positive* untuk Sehat), dan 18 merupakan salah prediksi sebagai Tidak Sehat (*False Positive* untuk Tidak Sehat). Baris kedua, 84 merupakan salah prediksi sebagai Kritis (*False Positive* untuk Kritis), 238 merupakan prediksi benar sebagai Sehat, dan 37 merupakan Salah prediksi sebagai Tidak Sehat (*False Positive* untuk Tidak Sehat). Baris ketiga, 51 merupakan salah prediksi sebagai Kritis (*False Positive* untuk Kritis), 51 merupakan Salah prediksi sebagai Sehat (*False Positive* untuk Sehat), dan 183 merupakan prediksi benar sebagai Tidak Sehat.



Gambar 9. Grafik *confusion matrix*

Hasil dari *confusion matrix* dan *classification report* pada Gambar 8 dibuat visualisasi datanya untuk mengomunikasikan informasi secara jelas dan menyampaikan pesan secara lebih efektif melalui representasi grafik. Grafik *confusion matrix* dapat dilihat pada Gambar 9 di atas dan grafik *classification report* pada Gambar 10 di bawah.



Gambar 10. Grafik *classification report*

4.2. Manfaat Sistem

Sistem yang diusulkan menawarkan berbagai manfaat yang berpotensi meningkatkan efisiensi dan keandalan dalam proses pemeriksaan kesehatan. Dengan tingkat akurasi data pelatihan mencapai 97%, sistem mampu memberikan hasil diagnostik yang andal, sehingga dapat mengurangi risiko kesalahan klasifikasi dan meningkatkan kualitas hasil perawatan pasien. Otomatisasi proses klasifikasi juga memungkinkan penghematan waktu secara signifikan dalam penilaian kondisi kesehatan, sehingga pengambilan keputusan klinis dapat dilakukan dengan lebih cepat dan tepat.

Selain meningkatkan efisiensi waktu, sistem ini berkontribusi terhadap optimalisasi sumber daya dengan meminimalkan kebutuhan input manual dan mengurangi beban kerja tenaga medis. Hal ini memungkinkan tenaga kesehatan untuk lebih fokus pada tugas-tugas klinis yang bersifat kritis. Penggunaan metode *Decision Tree Classifier* memberikan keunggulan dari sisi interpretabilitas, karena aturan keputusan yang dihasilkan bersifat transparan dan mudah dipahami oleh praktisi kesehatan, sehingga dapat meningkatkan kepercayaan terhadap rekomendasi yang diberikan oleh sistem.

Dari aspek implementasi, sistem memiliki tingkat skalabilitas yang baik dan mampu menangani kumpulan data dalam skala besar, sehingga dapat diterapkan pada berbagai lingkungan layanan kesehatan, mulai dari klinik kecil hingga rumah sakit besar. Standardisasi proses diagnostik yang diterapkan juga berperan dalam mengurangi kesalahan manusia akibat kelelahan, bias, atau keterbatasan keahlian. Selain itu, model yang digunakan bersifat adaptif karena dapat dilatih ulang

menggunakan data terbaru agar tetap relevan seiring perkembangan praktik medis dan perubahan karakteristik pasien. Dari sisi ekonomi, otomatisasi proses diagnostik membantu menekan biaya operasional, sehingga layanan kesehatan berbasis teknologi menjadi lebih efisien dan terjangkau.

5. Penutup

Berdasarkan hasil analisa yang dilakukan oleh penulis, penulis menarik kesimpulan bahwa penelitian ini menunjukkan keefektifan *Decision Tree Classifier* dalam mengembangkan sistem pendukung keputusan untuk pemeriksaan kesehatan. Sistem ini mencapai tingkat akurasi yang luar biasa sebesar 97% dari data pelatihan, yang menyoroti potensinya untuk menggantikan metode diagnostik manual tradisional. Kemampuan interpretasi dan efisiensi komputasinya menjadikannya metode praktis bagi praktisi perawatan kesehatan. studi ini membenarkan perlunya eksplorasi lebih lanjut.

Penelitian di masa mendatang dapat mengintegrasikan fitur tambahan, seperti hasil uji laboratorium atau data pencitraan, untuk meningkatkan kinerja prediktif. Menggunakan teknik ensemble seperti *Gradient Boosting* atau *Random Forests* dapat membantu meningkatkan akurasi dan mengurangi overfitting. Secara keseluruhan, riset ini menegaskan bahwa sistem yang dikembangkan tidak hanya mengurangi kendala waktu dan sumber daya, tetapi juga menyediakan solusi untuk meningkatkan pemeriksaan kesehatan.

6. Daftar Pustaka

[1] M. Ardi and E. D. Bintari, "Systematic Literature Review: Risiko Privasi dan Keamanan Data Pribadi Dalam Penggunaan Artificial Intelligence (AI)", *Informasi Interaktif*, vol. 9, no. 1, pp. 23–28, 2024.

[2] Y. Jumaryadi et al., *Sistem Pendukung Keputusan*. Yogyakarta: Penamuda Media, 2024.

[3] B. Sunarko et al., "Penerapan Stacking Ensemble Learning untuk Klasifikasi Efek Kesehatan Akibat Pencemaran Udara", *EDUKOMPUTIKA*, vol. 10, no. 1, pp. 55–63, 2023.

[4] I. Setiawan, R. F. A. Cahyani, and I. Sadida, "Exploring Complex Decision Trees: Unveiling Data Patterns And Optimal Predictive Power", *IFTECH*, vol. 5, no. 2, pp. 112–123, 2023.

[5] S. Sausan, D. M. Pratiwi, and L. Mufidah, "Metode Decision Tree Classifier dan XGBoost Classifier Dalam Memprediksi Penyakit Jantung", *CENTIVE*, vol. 4, no. 1, pp. 991–1000, 2024.

[6] R. E. Nugroho, W. Y. Pamungkas, and J. H. Jaman, "Pendeteksi Penyakit Hepatitis Menggunakan CART Decision Tree", *JITET*, vol. 12, no. 3S1, pp. 3690–3696, 2024.

[7] M. M. J. Samodro, M. K. Biddinika, and A. Fadlil, "Klasifikasi Penyakit Diabetes dengan Algoritma Decision Tree dan Naïve Bayes", *RESISTOR*, vol. 6, no. 2, pp. 113–118, 2023.

[8] Y. Jumaryadi, "Klasifikasi Penyakit Kardiovaskular Menggunakan Algoritma Decision Tree C4.5", *FIFO*, vol. 17, no. 1, pp. 1–6, 2025.

[9] M. Yusa, E. Utami, and E. T. Luthfi, "Evaluasi Performa Algoritma Klasifikasi Decision Tree ID3, C4.5, dan CART Pada Dataset Readmisi Pasien Diabetes", *InfoSys Journal*, vol. 4, no. 1, pp. 23–34, 2016.

[10] C. M. Sidiq, A. Faqih, and G. Dwilestari, "Algoritma Decision Tree C4.5 Digunakan Untuk Mengklasifikasikan Data Stroke", *JATI*, vol. 8, no. 2, pp. 1869–1874, 2024.

[11] W. A. Firmansyach, U. Hayati, and Y. A. Wijaya, "Analisa Terjadinya Overfitting dan Underfitting Pada Algoritma Naive Bayes dan Decision Tree Dengan Teknik Cross Validation", *JATI*, vol. 7, no. 1, pp. 262–269, 2023.

[12] A. Hidayahtulloh and D. Prasetyo, "Penerapan Algoritma Decision Tree Untuk Prediksi Kelulusan Mahasiswa Berdasarkan Data Akademik", *TEKNOBIS*, vol. 2, no. 4, pp. 615–619, 2024.

[13] C. N. Syahputri and M. S. Hasibuan, "Optimasi Klasifikasi Decision Tree Dengan Teknik Pruning Untuk Mengurangi Overfitting", *JSiI*, vol. 11, no. 2, pp. 87–96, 2024.

[14] H. A. Damayanti and U. P. Sanjaya, "Perbandingan Model Pembelajaran Mesin Berbasis Smote Meningkatkan Identifikasi Siswa Berisiko di Sekolah Menengah Pertama", *JSiI*, vol. 12, no. 1, pp. 119–127, 2025.

[15] F. R. A'la and Z. Fatah, "Perbandingan Algoritma Decision Tree dan Deep Learning dalam Prediksi Masalah Kesehatan berdasarkan Kebiasaan Gaya Hidup", *MUTIARA*, vol. 2, no. 10, pp. 1034–1046, 2024.

[16] E. Karyadiputra and A. Setiawan, "Sistem pendukung keputusan berbasis decision tree algorithm untuk prediksi penyakit diabetes", *TEKNOSAINS*, vol. 17, no. 3, pp. 294–301, 2023.

[17] B. D. Putro, "Persepsi dan Perilaku Pengobatan Tradisional Sebagai Alternatif Upaya Mereduksi Penyakit Tidak Menular", *Sunari Penjor*, vol. 4, no. 2, pp. 87–94, 2021.

[18] A. D. Astuti et al., "Aksesibilitas Layanan Kesehatan Modern dan Tradisional di Masyarakat Pantai Putra Serdang, Pantai Labu, Deli Serdang", *Medika*, vol. 3, no. 1, pp. 78–85, 2025.

[19] C. Cahyaningtyas, Y. Nataliani, and I. R. Widiasari, "Analisis Sentimen Pada Rating Aplikasi Shopee Menggunakan Metode Decision Tree Berbasis SMOTE", *Aiti*, vol. 18, no. 2, pp. 173–184, 2021.

[20] N. Nurajijah, 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.
[21] V. I. Lim, F. Fitria, and M. Hafid, "Implementasi Text Summarization pada Ulasan Aplikasi Mobile JKN Menggunakan TF-IDF dan Cosine Similarity", *KONVERGENSI*, vol. 21, no. 1, pp. 9–17, 2025.
[22] V. S. Ginting, K. Kusrini, and E. Taufiq, "Implementasi Algoritma C4.5 untuk Memprediksi Keter-lambatan Pembayaran Sumbangan Pembangunan Pendidikan Sekolah Menggunakan Python", *Inspiration: Jurnal Teknologi Informasi dan Komunikasi*, vol. 10, no. 1, pp. 36–44, 2020.
[23] 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.
[24] A. H. Wicaksono et al., "Klasifikasi Siswa Slow Learner Untuk Mendukung Sekolah Algoritma Naïve Bayes Classification of Slow Learner Students To Support Schools in Improving Student Understanding Using the Naïve Bayes", vol. 9, no. 3, pp. 589–596, 2022.
[25] M. Artur, "Review the performance of the Bernoulli Naïve Bayes Classifier in Intrusion Detection Systems using Recursive Feature Elimination with Cross-validated selection of the best number of features", *Procedia Comput. Sci.*, vol. 190, no. 2019, pp. 564–570, 2021.
[26] M. R. Sujono, A. Bahtiar, and B. Irawan, "Analisis Model Machine Learning Untuk Jenis Aspal di Jawa Barat Menggunakan Algoritma Decision Tree dan Random Forest", *JATI*, vol. 7, no. 6, pp. 3886–3891, 2023.