

Prediksi Penyakit Jantung Menggunakan Algoritma *Random Forest* Dan *Synthetic Minority Oversampling Technique (SMOTE)*

Salma Ngarifatul Khofiyah¹, Adam Prayogo Kuncoro²

^{1,2}Program Studi Teknik Informatika, Universitas Amikom Purwokerto
Banyumas, Jawa Tengah, Indonesia

E-mail : adam@amikompurwokerto.ac.id

Abstrak

Penyakit jantung merupakan salah satu penyebab utama kematian di dunia, sehingga diagnosis dini menjadi sangat penting. Penelitian ini mengembangkan sistem prediksi jantung berbasis web dengan menggunakan algoritma *Random Forest* dan kerangka kerja *Streamlit*. Dataset yang digunakan terdiri dari 299 sampel dengan 13 atribut, diambil dari *Kaggle*. Tahapan penelitian mencakup pengumpulan data, pra-pemrosesan, teknik *SMOTE* untuk menangani ketidakseimbangan data, pemodelan, evaluasi, dan implementasi sistem. Model yang dihasilkan menunjukkan akurasi 86,58% pada data uji, menunjukkan praktik kerja yang efektif dalam mengklasifikasikan risiko penyakit jantung. Sistem ini dirancang dengan antarmuka interaktif yang memudahkan pengguna untuk menganalisis data dan memperoleh prediksi. Penelitian ini memberikan solusi teknologi yang dapat membantu tenaga medis dalam membuat diagnosis awal secara cepat dan akurat, sehingga mempercepat pengambilan keputusan klinis.

Kata kunci : Prediksi Penyakit Jantung, *Random Forest*, *Machine Learning*, *Streamlit*

Heart Disease Prediction Using Random Forest Algorithm And Synthetic Minority Oversampling Technique (SMOTE)

Abstract

Heart disease is one of the leading causes of death in the world, so early diagnosis is very important. This research develops a web-based heart prediction system using Random Forest algorithm and Streamlit framework. The dataset used consists of 299 samples with 13 attributes, taken from Kaggle. The research stages include data collection, pre-processing, SMOTE technique to handle data imbalance, modeling, evaluation, and system implementation. The resulting model showed 86.58% accuracy on the test data, demonstrating effective working practices in classifying heart disease risk. The system is designed with an interactive interface that makes it easy for users to analyze data and derive predictions. This research provides a technological solution that can assist medical personnel in making an initial diagnosis quickly and accurately, thus accelerating clinical decision-making.

Keywords : *Heart Disease Prediction, Random Forest, Machine Learning, Streamlit*

1. Pendahuluan

Penyakit jantung adalah salah satu penyebab utama kematian di dunia. Menurut data dari *World Health Organization (WHO)*, lebih dari 17 juta orang meninggal setiap tahun akibat penyakit ini [1,2]. Penyakit ini juga berdampak besar pada kualitas hidup, biaya kesehatan, dan ekonomi global [3,4]. Faktor utama penyebab penyakit jantung meliputi gaya hidup tidak sehat, kebiasaan merokok, pola makan yang buruk, tekanan darah tinggi, kolesterol tinggi, dan obesitas [5,2].

Dalam beberapa tahun terakhir, kemajuan dalam teknologi kecerdasan telah memberikan kontribusi yang signifikan terhadap diagnosis beberapa penyakit, termasuk penyakit jantung. Teknologi ini memungkinkan prediksi penyakit dengan akurasi tinggi melalui analisis data kesehatan [1,4]. Salah satu

metode yang menonjol adalah *Random Forest*, sebuah algoritma *machine learning* berbasis *ensemble* yang mampu menangani data kompleks dan mengurangi risiko *overfitting* [1,6].

Berbagai penelitian sebelumnya menunjukkan bahwa penggunaan algoritma *Random Forest* untuk klasifikasi penyakit jantung menghasilkan akurasi yang sangat baik. Sebagai contoh, penelitian yang dilakukan di Indonesia menggunakan dataset dari *Kaggle* menunjukkan tingkat akurasi mencapai 87,7% pada data pengujian dan 92,63% pada data pelatihan [1,7]. Penelitian lain membandingkan algoritma ini dengan metode lain seperti *Support Vector Machine* dan *Naïve Bayes*, di mana *Random Forest* terbukti lebih unggul dalam hal akurasi dan stabilitas [3,6].

Tujuan dari penelitian ini adalah untuk mengembangkan sistem prediksi jantung berbasis web dengan menggunakan algoritma *Random Forest*. Sistem ini diharapkan dapat membantu tenaga medis dalam melakukan diagnosis awal, sehingga penanganan penyakit dapat dilakukan lebih cepat dan tepat. Dengan memanfaatkan *framework Streamlit*, penelitian ini juga bertujuan untuk menyajikan antarmuka pengguna yang interaktif dan mudah digunakan [7].

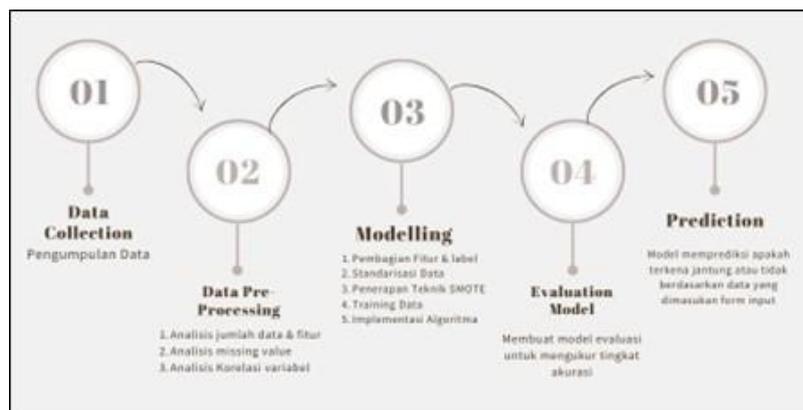
2. Metodologi

2.2 Data Pre-Processing

Langkah pertama dalam proses penelitian adalah preprocessing, yang sering dikenal sebagai persiapan data. Karena sistem tidak dapat menggunakan data dengan cara yang bersih, maka dilakukanlah prapemrosesan. Akibatnya, ada beberapa langkah yang harus diambil untuk sedikit memodifikasi data untuk meningkatkan kualitas data yang digunakan [7]. Langkah pertama dalam persiapan data adalah analisis jumlah data dan fitur, yaitu dilakukan pemilihan data atribut atau variable yang relevan yang akan digunakan dalam penelitian [2].

Langkah kedua dalam persiapan data adalah analisis *missing value* yang bertujuan untuk memahami sejauh mana nilai kosong memengaruhi integritas dataset dan membantu dalam pengambilan keputusan tentang penanganan nilai-nilai yang hilang [8].

Langkah selanjutnya adalah melakukan analisis korelasi variabel, yaitu ketika data yang akan dianalisis memiliki label, target, atau kelas dengan tujuan untuk mengetahui hubungan kualitas antara variabel yang bebas dan yang menjadi target/labelnya (tidak terkena penyakit jantung dan terkena penyakit jantung) [9].



Gambar 1. Tahapan Penelitian.

2.3 Modelling

Setelah persiapan data, dilanjutkan dengan proses pemodelan. Pada tahap ini, berbagai algoritme *machine learning* diterapkan pada dataset pelatihan untuk membuat model prediktif [10]. Langkah pertama pada tahap pemodelan yaitu pembagian fitur dan label. Di mana saya dapat menemukan variabel independen yang memiliki hubungan kuat dengan variabel dependen [11].

Langkah kedua pada tahap pemodelan yaitu standarisasi data yaitu memusatkan variabel pada nol dan menstandarisasi varians pada 1 sehingga akan memiliki properti distribusi normal [12].

Langkah ketiga dalam proses pemodelan adalah penerapan teknik SMOTE (*Synthetic Minority Oversampling Technique*). Tidak ada perbedaan yang signifikan dalam jumlah titik data antara pasien dengan jantung dan yang tidak. Jumlah data yang tidak konsisten dapat mempengaruhi hasil model machine learning. Oleh karena itu, meskipun akurasi model cukup tinggi, model tidak dapat menentukan apakah seorang pasien menderita jantung atau tidak [7].

Fokus utama pada tahap ini adalah pelatihan data. Data yang akan digunakan akan dibagi menjadi dua kelompok: data pelatihan dan data pengujian [13]. Analisis data digunakan untuk memvalidasi model yang telah dilatih dengan mengevaluasi akurasi prediktifnya, sedangkan pelatihan data digunakan untuk melatih model [11].

Terakhir pada tahap pemodelan yaitu implementasi algoritma, dimana dalam penelitian ini algoritma *machine learning* yang diterapkan untuk membuat model dengan menggunakan algoritma *Random Forest*. Salah satu algoritma pembelajaran mesin yang paling efektif adalah *random forest*, yang menggabungkan beberapa teknik keputusan untuk meningkatkan akurasi prediksi. Algoritma ini menggunakan beberapa atribut kunci dan subset data secara akurat untuk mempelajari setiap topik, lalu menggunakan hasil prediksi untuk menentukan klasifikasi akhir [14]. Kelebihan dari *random forest* adalah dapat menghindari *missing values* dan *outlier* serta dapat digunakan untuk data yang berukuran besar, permasalahan *nonlinier* dan permasalahan regresi [15].

2.4 Evaluation Model

Tahap evaluasi ini merupakan hasil dari model yang digunakan, seperti menentukan keakuratan data yang sedang diperiksa [4]. Akurasi adalah metrik evaluasi yang paling sering digunakan, yang menunjukkan bahwa model tertentu yang baik dapat mengklasifikasikan data secara akurat [16].

2.5 Prediction

Setelah tahap evaluasi dan pemodelan, langkah selanjutnya adalah tahap prediksi menggunakan model penyebaran. dimana model tersebut diperlukan dalam bentuk acar untuk diimplementasikan dalam sistem. Kerangka kerja yang digunakan dalam pengembangan sistem adalah streamlit [7].

3. Hasil dan Pembahasan

3.1 Data Collection

| age | anaemia | creatinine phosphokinase | diabetes | ejection fraction | high blood pressure | platelets | serum creatinine | serum sodium | sex | smoking | time | DEATH |
|------|---------|--------------------------|----------|-------------------|---------------------|-----------|------------------|--------------|-----|---------|------|-------|
| 75.0 | 0 | 582 | 0 | 20 | 1 | 265000.00 | 1.9 | 130 | 1 | 0 | 4 | |
| 55.0 | 0 | 7861 | 0 | 38 | 0 | 263358.03 | 1.1 | 136 | 1 | 0 | 6 | |
| 65.0 | 0 | 146 | 0 | 20 | 0 | 162000.00 | 1.3 | 129 | 1 | 1 | 7 | |
| 50.0 | 1 | 111 | 0 | 20 | 0 | 210000.00 | 1.9 | 137 | 1 | 0 | 7 | |
| 65.0 | 1 | 160 | 1 | 20 | 0 | 327000.00 | 2.7 | 116 | 0 | 0 | 8 | |

Gambar 2. Jumlah Dalam Dataset

Pada tahap data collection gambar 2, dataset yang digunakan mempunyai 13 atribut. Atribut-atribut ini meliputi:

1. *Age*: Usia pasien dalam tahun, yang merupakan salah satu faktor risiko utama penyakit jantung.
2. *Anaemia*: Indikator kondisi anemia (1 jika pasien mengalami anemia, 0 jika tidak), yang dapat memengaruhi fungsi jantung akibat kurangnya oksigen dalam darah.
3. *Creatinine Phosphokinase (CPK)*: Tingkat enzim CPK dalam darah, yang menunjukkan adanya kerusakan otot, termasuk otot jantung.
4. *Diabetes*: Indikator apakah pasien memiliki riwayat diabetes (1 untuk ya, 0 untuk tidak), yang sering dikaitkan dengan risiko tinggi penyakit kardiovaskular.

5. *Ejection Fraction*: Persentase darah yang dipompa keluar dari ventrikel kiri pada setiap kontraksi, digunakan untuk mengukur fungsi jantung.
6. *High Blood Pressure*: Indikator hipertensi (1 jika tekanan darah tinggi, 0 jika normal), yang merupakan salah satu penyebab utama penyakit jantung.
7. *Platelets*: Jumlah trombosit dalam darah (dalam kiloplatelet per mikroliter), yang dapat memengaruhi pembentukan bekuan darah.
8. *Serum Creatinine*: Kadar kreatinin dalam serum darah (mg/dL), digunakan untuk mengukur fungsi ginjal, yang berkaitan erat dengan kesehatan kardiovaskular.
9. *Serum Sodium*: Kadar natrium dalam darah (mEq/L), yang mencerminkan keseimbangan elektrolit tubuh dan dapat memengaruhi fungsi jantung.
10. *Sex*: Jenis kelamin pasien (1 untuk pria, 0 untuk wanita), yang menjadi salah satu faktor risiko biologis.
11. *Smoking*: Indikator kebiasaan merokok pasien (1 jika merokok, 0 jika tidak), yang dikenal sebagai faktor risiko utama penyakit jantung.
12. *Time*: Waktu dalam hari sejak pasien pertama kali diperiksa hingga akhir periode penelitian.
13. *Death Event*: Indikator hasil (1 jika pasien menunjukkan terindikasi penyakit jantung sejak periode penelitian, 0 jika tidak terindikasi penyakit jantung), yang digunakan sebagai label target untuk analisis prediksi.

3.2 Data Pre-Processing

```

age                float64
anaemia            int64
creatinine_phosphokinase  int64
diabetes           int64
ejection_fraction int64
high_blood_pressure int64
platelets          float64
serum_creatinine  float64
serum_sodium      int64
sex               int64
smoking           int64
time              int64
DEATH_EVENT       int64
dtype: object
    
```

Gambar 3. Jumlah Data & Fitur

Pada gambar 3, menunjukkan data dengan Kolom atau fitur yang ada sebanyak 13 kolom, yang terdiri dari tipe data int64 dan float64.

```

# Check missing values
print(data_jantung.isnull().sum())

age                0
anaemia            0
creatinine_phosphokinase  0
diabetes           0
ejection_fraction 0
high_blood_pressure 0
platelets          0
serum_creatinine  0
serum_sodium      0
sex               0
smoking           0
time              0
DEATH_EVENT       0
dtype: int64
    
```

Gambar 4. Analisis Missing Value

Pada gambar 4, adalah hasil analisis missing value. Dari hasil analisis tidak ditemukan missing value dalam dataset ini. Setiap fitur, termasuk *age*, *anaemia*, *creatinine_phosphokinase*, *diabetes*, *ejection_fraction*, *high_blood_pressure*, *platelets*, *serum_creatinine*, *serum_sodium*, *sex*, *smoking*, *time*

,*death_event* semuanya memiliki 0 missing value. Artinya, dataset ini sudah bersih dan lengkap, sehingga tidak memerlukan penanganan khusus untuk missing value dan dapat langsung digunakan dalam proses analisis atau pemodelan.

```
data_jantung['DEATH_EVENT'].value_counts()

DEATH_EVENT
0      203
1       96
Name: count, dtype: int64
```

Gambar 5. Analisis Variabel

Hasil analisis variabel ditunjukkan pada gambar 5, di mana 0 menunjukkan bahwa tidak ada penyakit jantung yang diketahui dan 1 menunjukkan adanya penyakit jantung.

3.3 Modelling

```
# memisahkan data dan label
x = data_jantung.drop(columns='DEATH_EVENT', axis=1)
y = data_jantung['DEATH_EVENT']
```

Gambar 6. Pembagian Fitur & Label

Pada gambar 6, merupakan tahap pertama dalam *modelling*. Selanjutnya pembagian fitur dan label, dimana dalam tahap ini data dipecah menjadi dua bagian utama yaitu fitur (*independent variables*) kolom lain selain target dan label (*dependent variables*) variabel dependen yaitu target.

```
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()

[11]

> scaler.fit(x)

[12]
...
StandardScaler
StandardScaler()

standarized_data = scaler.transform(x)

[13]
```

Gambar 7. Standarisasi Data

Setelah pembagian fitur dan label, selanjutnya pada gambar 7, yaitu standarisasi data. Standarisasi data bertujuan untuk membuat semua fitur dalam skala yang sama dan menghilangkan bias yang diakibatkan karena perbedaan skala antar fitur.

```
from imblearn.over_sampling import SMOTE
smote = SMOTE()
X_resampled, y_resampled = smote.fit_resample(standarized_data, y)
```

Gambar 8. Penerapan Teknik SMOTE

Langkah selanjutnya dalam pemodelan adalah penerapan teknik SMOTE (Synthetic Minority Over-sampling Technique), yang dapat dilihat pada gambar 8. SMOTE adalah teknik oversampling yang digunakan untuk membuat data sintetis dari data sampel minoritas yang dapat digunakan untuk menyesuaikan proporsi kelas dataset. SMOTE akan membantu model agar tidak bias terhadap kelas mayoritas dan memberikan prediksi yang lebih akurat untuk kelas minoritas.

```
x_train, X_test, y_train, y_test = train_test_split(X_resampled, y_resampled, test_size=0.2, random_state=42)
```

Gambar 9. Training Data

Pada gambar 9, dalam tahap *modelling* yaitu *training* data. Analisis dataset ini menggunakan fungsi `train_test_split` dari `scikit-learn`, yang bertujuan untuk memisahkan data menjadi 80% data latih dan 20% data uji, memastikan bahwa hasil evaluasi model akurat, dan mengidentifikasi *overfitting*. Dataset yang telah di-resampling sebelumnya (`X_resampled` untuk fitur dan `y_resampled` untuk label) digunakan untuk mengukur ketidakkonsistenan data, seperti pada prediksi penyakit jantung, di mana jumlah pasien tanpa penyakit jantung secara konsisten lebih tinggi daripada jumlah pasien dengan penyakit jantung. Parameter `random_state = 42` digunakan untuk memberikan hasil analisis data yang konsisten.

```
from sklearn.ensemble import RandomForestClassifier

classifier = RandomForestClassifier(n_estimators=100, random_state=42)

classifier.fit(x_train, y_train)
```

RandomForestClassifier
RandomForestClassifier(random_state=42)

Gambar 10. Random Forest

Implementasi algoritma Random Forest Classifier dari paket `scikit-learn` ditunjukkan pada Gambar 10 untuk membuat model prediksi penyakit jantung. Algoritma ini merupakan metode *ensemble learning* yang menggabungkan prediksi dari beberapa *decision tree* untuk meningkatkan akurasi dan mengurangi risiko *overfitting*. Parameter `n_estimators=100` menunjukkan bahwa model menggunakan 100 pohon keputusan dalam proses pembelajaran, sementara `random_state=42` memastikan replikasi hasil dengan mengatur *seed* untuk pembagian data dan pelatihan model. Model ini dilatih menggunakan data latih (`x_train` untuk fitur seperti tekanan darah, dan `y_train` untuk label risiko penyakit) dengan tujuan mengidentifikasi pola dan hubungan yang relevan.

3.4 Evaluation Model

```
print('Akurasi Data Training Adalah = ', training_data_accuracy)
```

Akurasi Data Training Adalah = 1.0

Gambar 11. Akurasi Data Latih

Pada gambar 11 menunjukkan hasil evaluasi tersebut menunjukkan bahwa model *Random Forest Classifier* memiliki akurasi data pelatihan (*train data*) sebesar 1.0 (100%). Hasil tersebut menandakan bahwa model mampu memprediksi data pelatihan dengan sempurna tanpa kesalahan.

```
print('Akurasi Data Testing Adalah = ', test_data_accuracy)
Akurasi Data Testing Adalah = 0.8658536585365854
```

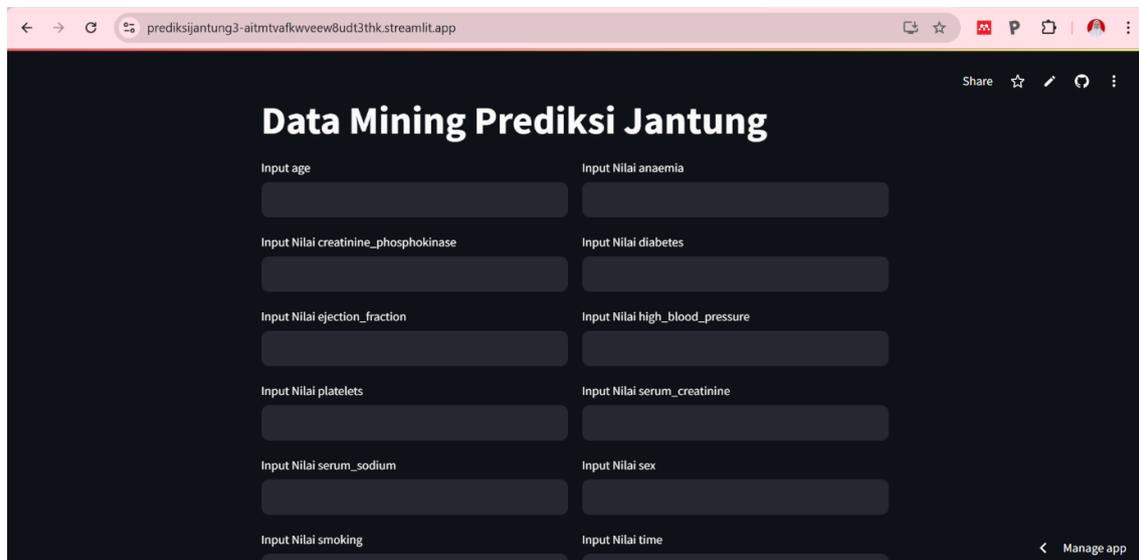
Gambar 12. Akurasi Data Uji

Seperti yang dapat dilihat pada gambar 12 hasil evaluasi, akurasi model pada data test (data pengujian) adalah 0.8658 (86.58%), yang menunjukkan bahwa model Random Forest Classifier memiliki kinerja yang baik ketika memprediksi masalah berdasarkan data yang belum pernah diperiksa. Jika dibandingkan dengan akurasi pada data training (100%), akurasi ini lebih tinggi, mengindikasikan bahwa model berhasil mengurangi kemungkinan terjadinya overfitting dengan tetap mempertahankan performa yang baik.

3.5 Prediction

Setelah menyelesaikan tahap evaluasi, langkah selanjutnya adalah membuat prediksi menggunakan model penyebaran. Para peneliti menggunakan ekstensi pickle, yang merupakan format serialisasi Python yang memungkinkan model diekspor sebagai file eksternal. Format ini berguna karena memudahkan untuk mengintegrasikan model dengan sistem lain dan memperkuat struktur dan parameter model sehingga dapat digunakan kembali tanpa memerlukan pelatihan ulang untuk diterapkan pada sistem di kemudian hari.

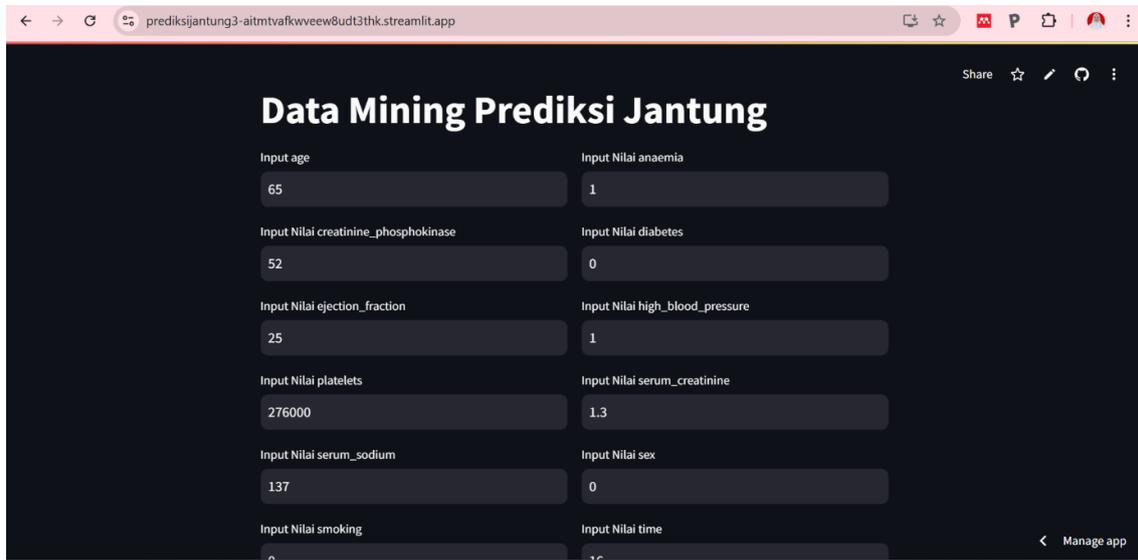
Ketika mengembangkan sistem berbasis web, para peneliti menggunakan kerangka kerja Streamlit. Hal ini disebabkan oleh kemampuannya untuk membuat antarmuka pengguna yang responsif dan interaktif dengan cepat tanpa memerlukan pengembangan web yang rumit.



Gambar 13. Tampilan Interfaces Sistem

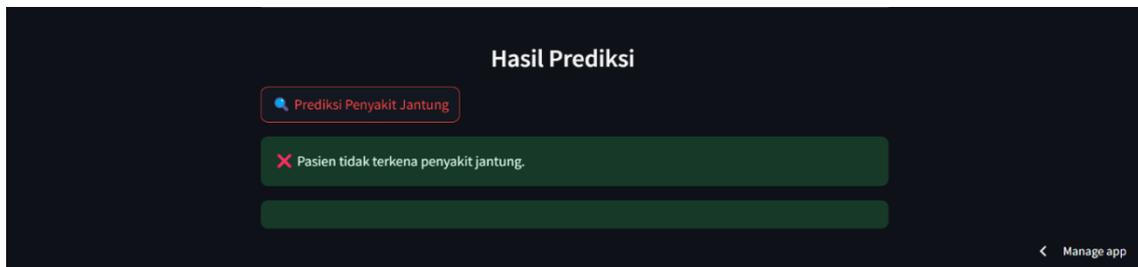
Pada gambar 13, tampilan *interface* uji coba sistem menunjukkan desain antarmuka yang sederhana, dirancang untuk mempermudah pengguna dalam berinteraksi dengan sistem prediksi penyakit jantung berbasis web. Antarmuka ini mencakup berbagai elemen input, yang memungkinkan pengguna memasukkan data pasien berdasarkan indikator yang telah ditentukan, seperti usia, tekanan darah, kadar kolesterol, riwayat merokok, dan parameter kesehatan lainnya.

Kesederhanaan desain antarmuka ini tidak hanya bertujuan meningkatkan efisiensi pengguna tetapi juga memastikan sistem dapat digunakan oleh tenaga medis dengan berbagai tingkat keahlian teknis. Hal ini penting untuk menjadikan sistem ini sebagai alat bantu yang efektif dalam mendukung pengambilan keputusan medis berbasis data.



Gambar 14. Uji Coba Sistem

Pada gambar 14, peneliti memasukkan data tersebut ke dalam sistem sebagai bagian dari proses uji coba untuk memastikan bahwa model bekerja sesuai dengan yang diharapkan. Data yang dimasukkan mencerminkan parameter kesehatan pasien, yang selanjutnya diproses oleh model *machine learning* yang telah *dideploy*. Setelah semua data indikator dimasukkan, pengguna dapat menekan tombol “Prediksi Penyakit Jantung”, yang menginisiasi proses analisis oleh model *Random Forest*.



Gambar 15. Hasil Prediksi Tidak Terkena Jantung



Gambar 16. Hasil Prediksi Terkena Jantung

Hasil analisis ditampilkan pada gambar 15 dan 16, di mana sistem memberikan output berupa hasil prediksi risiko penyakit jantung. Jika model mendeteksi bahwa data pasien menunjukkan risiko tinggi penyakit jantung, hasil prediksi akan menampilkan informasi bahwa pasien "Pasien Tidak Terkena Penyakit

Jantung". Sebaliknya, jika data pasien menunjukkan tidak ada risiko signifikan, sistem akan memberikan hasil bahwa pasien "Pasien Terkena Penyakit Jantung".

Proses ini menunjukkan bagaimana sistem dapat mendukung pengguna, terutama tenaga medis, dalam membuat keputusan yang lebih cepat dan berbasis data. Dengan pendekatan berbasis web, sistem ini dapat diakses dengan mudah di berbagai perangkat, memberikan *fleksibilitas* untuk digunakan dalam berbagai situasi klinis.

4. Kesimpulan

Penelitian ini berhasil mengembangkan sistem prediksi penyakit jantung berbasis web menggunakan algoritma Random Forest dan framework Streamlit. Dengan akurasi 86,58% pada data uji, sistem ini dapat memprediksi risiko penyakit jantung dan menunjukkan kinerja yang baik dalam membantu diagnosis. Dengan dataset yang terdiri dari 299 sampel dan 13 atribut, model ini dapat menangani ketidakseimbangan data melalui penerapan teknik SMOTE, sehingga menghasilkan prediksi yang lebih akurat, terutama untuk kelas minoritas.

Antarmuka sistem yang dirancang sederhana dan interaktif memberikan kemudahan bagi tenaga medis dalam menganalisis data pasien, mendukung pengambilan keputusan yang cepat dan tepat. Meskipun demikian, terdapat beberapa keterbatasan, seperti ukuran dataset yang kecil dan atribut yang bersifat statis, yang dapat memengaruhi generalisasi hasil. Oleh karena itu, diperlukan penelitian lanjutan untuk mengatasi keterbatasan ini guna meningkatkan keandalan dan fleksibilitas sistem dalam berbagai skenario dunia nyata.

Ucapan Terima Kasih

Penulis mengucapkan terima kasih kepada bapak Adam Prayogo Kuncuro, M.Kom. yang telah memberi dukungan terhadap penelitian ini.

Daftar Pustaka

- [1] S. A. Putri, N. Selayanti, and M. Kristanaya, "Penerapan Machine Learning Algoritma Random Forest Untuk Prediksi Penyakit Jantung," vol. 2024, no. Senada, 2024.
- [2] R. Y. Donny Maulana, "IMPLEMENTASI ALGORITMA NAÏVE BAYES UNTUK KLASIFIKASI PENDERITA PENYAKIT JANTUNG DI INDONESIA MENGGUNAKAN RAPID MINER," *SIGMA – J. Teknol. Pelita Bangsa*, vol. 84, no. 10, pp. 1511–1518, 2019, doi: 10.1134/s0320972519100129.
- [3] Firdlous A.D., "Komparasi Algoritma Klasifikasi Data Mining untuk Memprediksi Penyakit Jantung," *J. Ilmu-ilmu Inform. dan ManajemenSTMIK*, vol. 16, no. 1, pp. 79–84, 2022, [Online]. Available: <http://ejournal.stmik-sumedang.ac.id/index.php/infomans/article/view/412>
- [4] F. Muzakki, I. Ubaydillah, N. R. Assyiami, and S. Soleha, "Penerapan Algoritma C4.5 Untuk Prediksi Penyakit Jantung Menggunakan Rapidminer," *J. Komput. Antart.*, vol. 2, no. 2, pp. 71–79, 2024, doi: 10.70052/jka.v2i2.304.
- [5] H. M. Nawawi, J. J. Purnama, and A. B. Hikmah, "Komparasi Algoritma Neural Network Dan Naïve Bayes Untuk Memprediksi Penyakit Jantung," *J. Pilar Nusa Mandiri*, vol. 15, no. 2, pp. 189–194, 2019, doi: 10.33480/pilar.v15i2.669.
- [6] S. P. R. Yulianto, A. Z. Fanani, A. Affandy, and M. I. Aziz, "Analisis Metode Smoote pada Klasifikasi Penyakit Jantung Berbasis Random Forest Tree," *J. Media Inform. Budidarma*, vol. 8, no. 3, p. 1460, 2024, doi: 10.30865/mib.v8i3.7712.
- [7] A. Putranto, N. L. Azizah, and A. I. Ratna Ika, "Sistem Prediksi Penyakit Jantung Berbasis Web Menggunakan Metode SVM dan Framework Streamlit," *J. Penerapan Sist. Inf. (Komputer Manajemen)*, vol. 4, no. 2, pp. 442–452, 2023, [Online]. Available: <https://archive.ics.uci.edu/ml/datasets/heart+disease>
- [8] A. R. Raharja, Jayadi, A. Pramudianto, and Y. Muchsam, "Penerapan Algoritma Decision Tree dalam Klasifikasi Data 'Framingham' Untuk Menunjukkan Risiko Seseorang Terkena Penyakit Jantung dalam 10 Tahun Mendatang," *Technol. J.*, vol. 1, no. 1, 2024, doi: 10.62872/cwggzp962.
- [9] A. Athohiroh, R. Ayu, and S. Maharani, "Penerapan Metode Naive Bayes Untuk Memprediksi Penyakit Jantung," *J. Tek.*, vol. 3, no. 1, p. 8, 2023, doi: 10.54314/teknisi.v3i1.1252.

-
- [10] A. Ariawan, "Optimasi Prediksi Gagal Jantung dengan Teknik Ensemble Bagging Pada Neural Network," vol. 5, no. 3, pp. 1000–1009, 2024.
- [11] A. Wijayadhi, M. Makmun Effendi, and S. Budi Rahardjo, "Prediksi Penyakit Jantung Dengan Algoritma Regresi Linier," *Bull. Inf. Technol.*, vol. 4, no. 1, pp. 15–28, 2023, doi: 10.47065/bit.v4i1.463.
- [12] "Bagaimana, Kapan dan Mengapa Anda Harus Menormalkan / Standarisasi / Mengubah Skala Data Anda?" [Online]. Available: <https://ichi.pro/id/bagaimana-kapan-dan-mengapa-anda-harus-menormalkan-standarisasi-mengubah-skala-data-anda-69304631376127>
- [13] 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, doi: 10.36706/jres.v2i2.29.
- [14] N. H. Alfajr and S. Defiyanti, "METODE RANDOM FOREST DAN PENERAPAN PRINCIPAL COMPONENT ANALYSIS (PCA)," vol. 12, no. 3, 2024.
- [15] M. Anshori, N. Rikatsih, and M. S. Haris, "Prediksi Pasien Dengan Penyakit Kardiovaskular Menggunakan Random Forest," *TEKTRIKA - J. Penelit. dan Pengemb. Telekomun. Kendali, Komputer, Elektr. dan Elektron.*, vol. 7, no. 2, p. 58, 2023, doi: 10.25124/tektrika.v7i2.5279.
- [16] A. Handika Permana, F. Rakhmat Umbara, and F. Kasyidi, "Klasifikasi Penyakit Jantung Tipe Kardiovaskular Menggunakan Adaptive Synthetic Sampling dan Algoritma Extreme Gradient Boosting," *Build. Informatics, Technol. Sci.*, vol. 6, no. 1, pp. 499–508, 2024, doi: 10.47065/bits.v6i1.5421.
-