**Journal of Applied Informatics and Computing (JAIC)**
Vol.8, No.2, December 2024, pp. 598~604
e-ISSN: 2548-6861 

# Comparison of Machine Learning Models for Heart Disease Classification with Web-Based Implementation

**Angga Ramda Ramadhan [1]*, Nandang Saefulloh [2]*, Nisa Utami [3]*, Muji Diana [4]*, Abiyyu Aji Prasetyo Utomo [5]*, Yusuf Eka Wicaksana [6]***
* Teknik Informatika, Univcrsitas Buana Perjuangan Karawang
if20.anggaramadhan@mhs.ubpkarawang.ac.id [1] if20.nandangsaefulloh@mhs.ubpkarawang.ac.id [2] if20.nisautami@mhs.ubpkarawang.ac.id [3] if20.mujidiana@mhs.ubpkarawang.ac.id [4] if20.abiyyuutomo@mhs.ubpkarawang.ac.id [5] yusuf.eka.@ubpkarawang.ac.id [6]

**Article Info**

***Article history:***

Received 2024-10-25
Revised 2024-11-05
Accepted 2024-11-06

***Keyword:***

*Classification,*
*Comparison,*
*Heart Disease,*
*Implementation,*
*Machine Learning*

**ABSTRACT**

Heart disease has become one of the most concerning diseases in Indonesia according to research published in 2018 by the Health Ministry of Indonesia. Based on said research, 15 out of 1000 Indonesians suffer from heart disease. Furthermore, according to data published by the Health Ministry of Indonesia, 3 million premature deaths (under 60 years old) occurred in 2013 due to heart disease. Therefore, this research aims to develop a web-based system designed to aid health workers in screening for heart diseases and producing early diagnosis. In developing this system, 5 models were evaluated based on performance and the model with the best metrics was selected to be used in the final system. These models were: Logistic Regression, Decision Tree, Random Forest, Naïve Bayes, and K-Nearest Neighbours. SMOTE and ADASYN was also used to deal with imbalanced data, and the resulting balanced data was used as additional training scenarios in order to compare the result with algorithms trained using imbalanced data. Cross validation, accuracy, precision, recall, f1-score, and ROC with AUC were set as evaluation metrics. Results show that Random Forest trained with data balanced using ADASYN achieved the highest AUC score of 0.920. Meanwhile, Logistic Regression scored lowest with an AUC score of 0.500. These results indicate that Random Forest is the most suitable for this system Therefore, Random Forest was selected as the algorithm to be used in the final system. Furthermore, this system has been tested successfully using the black-box method and is ready to be implemented.



## I. PENDAHULUAN

Penyakit jantung menjadi penyakit yang berbahaya dengan rating penyebab kematian yang relatif tinggi [1], [2]. Menurut World Health Organization terdapat lebih dari 7 juta orang meninggal akibat penyakit ini pada tahun 2002. Hal ini juga di prediksi meningkat hingga 11 juta orang pada tahun 2020 [3]. Selain itu di Indonesia penyakit ini ada pada fase yang cukup mengkhawatirkan. Hal ini terjadi karena menurut data dari RISKESDAS (Riset Kesehatan Dasar). Sekitar 15 dari 1000 penduduk Indonesia terkena penyakit jantung pada tahun 2018, juga terdapat penigkatan sebesar 1.5% dari tahun 2013 sampai 2018 [4], [5]. Penyakit serangan jantung terjadi ketika saluran arteri tersumbat akibat timbunan lemak. Hal ini mengakibatkan asupan darah ke jantung berkurang. Dampak yang akan dirasakan jika aliran darah berkurang pada penderita seperti nyeri dada yang merupakan peringatan bahwa seseorang tersebut dapat terkena serangan jantung [6], [7]. Adapun permasalahan terkait penyakit jantung timbul karena tingkat kematian yang cukup tinggi sebesar 12 juta orang secara global [8]. Sedangkan di Indonesia 3 juta orang pada usia dibawah 60 tahun meninggal akibat penyakit ini, yang diantaranya 37% pada rentang 20-40 tahun mengidap penyakit *cardiovascular*, di mana hal ini seharusnya dapat dicegah [9]. Maka dari itu dibutuhkannya sebuah media yang bisa digunakan tenaga medis dalam mengambil keputusan untuk mencegah terjadinya serangan jantung [10] Dengan menggunakan teknik data mining dapat dijadikan solusi untuk

memecahkan masalah ini dengan menganalisis pola yang ada pada data [11].

Adapun penelitian terkait penyakit serangan jantung dengan membandingkan berbagai algoritma dilakukan oleh [12] Pada prosesnya algoritma yang dibandingkan meliputi logistic regression, naïve bayes, dan neural network untuk membuat model klasifikasi. Pada proses pembagian data yang dilakukan dengan 80% data sampel, dilanjutkan dengan pengujian menggunakan ROC dan AUC, confusion matrix, dan classification report. Adapun nilai tertinggi dalam proses evaluasi didapat oleh algoritma Naïve bayes dengan akurasi sebesar 84.3%. Sedangkan nilai terendah didapat oleh metode neural network dengan nilai akurasi sebesar 78.2%. Berikutnya penelitian yang dilakukan oleh [13] yang melakukan perbandingan metode support vector machine serta naïve bayes untuk mengklasifikasikan peluang penyakit serangan jantung. Dalam prosesnya digunakan 2 skenario dalam proses training, di mana skenario pertama akan mengguna 20% data training. Sedangkan dalam skenario kedua akan menggunakan data sebanyak 40% untuk dilakukan training. Hasil nilai evaluasi tertinggi didapatkan oleh algoritma support vector machine menggunakan skenario pertama. Hasilnya meliputi akurasi sebesar 87%, precision 92%, dan recall 87%. Selanjutnya penelitian lain dilakukan oleh [14] dengan membandingkan kinerja algoritma c 4.5 serta extreme learning dalam mendiagnosa 3 penyakit jantung koroner. Pada prosesnya menggunakan data yang didapat dari Kaggle dengan nama dataset "Cleveland dataset (VA Medical Center, Long Beach and Cleveland Clinic Foundation: Robert Detrano, MD, PhD) Irvine: The University of California Irvine; 1988". Dimana dataset ini memiliki jumlah pasien sebanyak 303 dengan 14 atribut yang digunakan pada data training dan data testing. Kapasitas data training dalam penelitian ini sebesar 80% dengan 20% pada data testing. Selanjutnya merupakan tahap normalisasi data dengan tujuan membuat data memliki rentang nilai yang sama. Selanjutnya pada tahap evaluasi model didapatkan akurasi sebesar 73.33% oleh metode ELM. Sedangkan untuk algoritma c4.5 menghasilkan akurasi dengan jumlah 93.33%. Dimana hal ini menunujukan performa yang lebih baik ketika menggunakan metode c4.5.

Berdasarkan penelitian terkait, tujuan dari penelitian kali ini yaitu membuat model klasifikasi pada penyakit serangan jantung. Dengan membandingkan performa dari beberapa algoritma seperti *random forest*, *k-nearest neighbors*, *decision tree*, *logistic regression*, dan *naïve bayes*. Dengan menambahkan juga beberapa pengujian dengan menggunakan data yang seimbang serta data yang tidak seimbang. Adapun beberapa algoritma dari penelitian ini dipilih untuk mencari metode manakah yang lebih relevan dalam kasus dan data yang digunakan. Metode yang digunakan untuk menyeimbangkan data antara lain SMOTE dan ADASYN. Selain itu pada penelitian ini Evaluasi yang akan diterapkan yaitu, *cross validation score* terhadap nilai akurasi, *precision*, *recall*, dan f1-*score*, serta ROC dan AUC. Selanjutnya hasil terbaik dari setiap pengujian akan dijadikan model yang nantinya digunakan pada sistem pengklasifikasian penyakit jantung berbasis web.

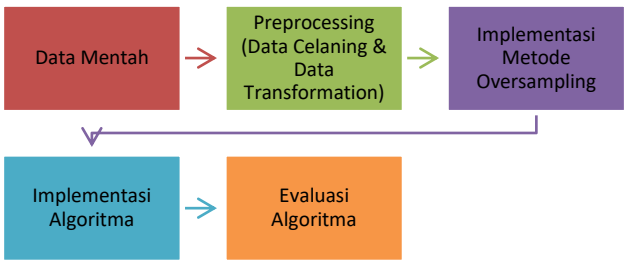
## II. METODE PENELITIAN

Pada penelitian ini data yang digunakan didapat dari website *Kaggle* yang berjudul "*Indicators of Heart Disease (2022 UPDATE)*". Dimana data ini memiliki jumlah baris sebesar 445.132 data dengan total kolom sebanyak 39. Penelitian kali ini dilakukan dengan mencoba membandingkan setiap algoritma, untuk mendapatkan performa terbaik dari satu algoritma. Adapun perbandingan meliputi hasil dari nilai *recall*, *precision*, *f1-core*, akurasi serta ROC dan AUC. Dimana selanjutnya hasil terbaik dari salah satu algoritma akan dijadikan model dalam sistem pengklasifikasian penyakit serangan jantung. Berikutnya dapat dilihat hasil ilustrasi alur penelitian pada Gambar 1.


Gambar 1. Alur Penelitian

### A. *Modelling*

Pada tahap pemodelan diawalai dengan tahap pra pemrosesan. Selanjutnya merupakan tahap implementasi algoritma. Dilanjutkan dengan tahapan oversampling SMOTE dan ADASYN. Dan terakhir merupakan tahap evaluasi algoritma. Adapun berikutnya dapat dilihat detail dari proses modelling pada Gambar 2.




Gambar 2. Alur Modelling

Tahapan pada Gambar 2 setelah mendapatkan data mentah diawali dengan tahap *preprocessing*. Adapun pada tahap *preprocessing* terdapat data *cleaning* dan data *transformation*. Data *cleaning* meliputi *pengecekan missing value*, pembersihan *missing value*, pengecekan *outlier*, dan pengecekan duplikat data. Selanjutnya merupakan tahapan implementasi metode oversampling SMOTE dan ADASYN. Tahapan ini menghasilkan data yang seimbang, oleh karena itu data yang akan digunakan dalam setiap implementasi algoritma meliputi data seimbang dan data yang tidak seimbang. Berikutnya merupakan tahap evaluasi algoritma dimana pada proses evaluasi menggunakan data yang seimbang dan tidak seimbang. Data yang seimbang didapat dari proses *oversampling* dengan SMOTE dan ADASYN. Nilai yang akan dievaluasi meliputi nilai akurasi, *recall*,

*precision, f1-score*, akurasi serta ROC dan AUC. Terakhir setelah tahap evaluasi, selanjutnya dilakukan analisis dari setiap hasil evaluasi untuk mendapatkan model terbaik dari satu algoritma.

### *B. Cross Validation Score*

Metode cross validation score digunakan untuk memperkirakan suatu model yang akan menggeneralisasi pada data yang sebelumnya belum terlihat. Dataset yang ada dibagi menjadi subset yang tumpang tindih dan bergeser, sehingga model tersebut kemudian dilatih pada subset besar dan diuji pada subset kecil [15].

### *C. Synthetic Minority Oversampling Technique* (SMOTE)

Merupakan sebuah metode yang digunakan untuk mengambil sampel ulang pada tingkat data yang popular. SMOTE dapat secara efektif mengatasi masalah overfitting yang disebabkan oleh oversampling acak. SMOTE menghasilkan contoh sintesis dari minoritas lingkungannya [16].

### *D. Adaptive Synthetic* (ADASYN)

Merupakan versi perbaikan dari metode SMOTE, metode ini biasanya digunakan untuk mencegah overfitting. Yang terjadi Ketika replika dari instance minoritas ditambahkan pada dataset utama. Metode ADASYN bekerja menggunakan distribusi identitas sebagai kriteria yang bekerja secara otomatis dalam memilah jumlah sampel sintetis yang sesuai dari setiap contoh pada data minoritas [17].

### *E. Naïve Bayes*

Naïve bayes berakar pada sebuah metode bernama teorema bayes. Dimana metode ini menggunakan probabilitas dan statistik yang ditemukan oleh ilmuwan inggris bernama Thomas Bayes. Sedangkan dalam kasus klasifikasi naïve bayes memiliki asumsi yang sangat kuat (naif) dari independensi masing-masing kondisi/kejadian. Adapun persamaan pada metode naïve bayes seperti pada persamaan di bawah ini.

$$P(H|X) = \frac{P(X|H).P(H)}{P(X)}$$

Dimana pada persamaan di atas nilai X merupakan data dengan kelas yang belum diketahui. H merupakan hipotesis data suatu kelas spesifik, P(H|X) adalah probabilitas hipotesis H berdasarkan kondisi X (posteriori probabilitas). P(H) merupakan probabilitas dari hipotesis H (prior probabilitas), P(X|H) merupakan probabilitas X berdasarkan kondisi hipotesis H. Sedangkan P(X) merupakan probabilitas X [18].

### *F. Logistic Regression*

Adalah algoritma yang cukup popular dalam dunia statistic, analisis data, dan pembelajaran mesin. Regresi logistik digunakan dalam memprediksi probabilitas terhadap suatu peristiwa berdasarkan fitur-fitur yang relevan.Fungsi yang digunakan oleh algoritma ini mirip dengan regresi linear. Dimana perbedaannya algoritma ini menyelesaikan masalah klasifikasi dalam bentuk keluaran probabilitas atau kelas-kelas diskret seperti “1 atau 0” dan “ya atau tidak” [19].

### *G. Decision Tree*

Decision tree merupakan salah satu metode yang popular digunakan dalam menyelesaikan masalah klasifikasi. Dalam prosesnya decision tree membuat pohon keputusan berdasarkan set data pada input berlabel. Adapun kelebihan dari algoritma ini yaitu dapat diimplementasikan dengan data kontinu dan nilai diskrit.Algoritma ini membagi data training menjadi bantuan perolehan informasi. Dimana atribut yang memilliki frekuensi tertinggi dapat dipertimbangkan untuk memisahkan data dari informasi yang tersedia pada dataset. Adapun pada proses menghitung nilai gain memerlukan nilai entropy, dimana untuk mencari nilai entropy dapat menggunakan persamaan berikut.

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

Dimana pada persamaan di atas S merupakan himpunan kasus, A merupakan atribut, n merupakan jumlah patisi S, dan Pi merupakan proporsi dari Si terhadap S. Sedangkan dalam menghitung nilai information gain dapat dilihat pada persamaan di bawah ini.

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

Pada persamaan di atas diketahui bahwa S merupakan himpunan kasus, A merupakan atribut, n merupakan jumlah patisi S. Selanjutnya Pi merupakan proporsi dari Si terhadap S, A|Si| merupakan jumlah kasus pada partisi ke I, dan |S| merupakan jumlah kasus dalam S [20]

### *H. Random Forest*

Random forest merupakan sebuah metode klasifikasi yang dalam prosenya terjadi penggabungan data dengan dengan decision tree. Adapun keuntungan yang bisa didapat dengan menggunakan algoritma ini yaitu. Dapat menghasilkan performa yang baik dalam jumlah data yang besar [21]. Selain itu random forest bekerja dengan menggunakan subset dari suatu variable pada setiap pohon, yang kemudian dicari nilai ambang batas terbaik dalam proses memisahkan data [22].

### *I. K-Nearest Neighbors*

Algoritma knn adalah sebuah metode yang dapat menyelesaikan kasus klasifikasi maupun prediksi. KNN melakukan klasifikasi dalam suatu data berdasarkan nilai k yang ditetapkan sebelumnya. Adapun nilai ganjil digunakan ketika kasus yang diselesaikan yaitu klasifikasi. Sedangkan untuk prediksi dapat menggunakan nilai genap dan ganjil.Selanjutnya perhitungan jarak yang sering digunakan pada metode KNN adalah Euclidean distance, dimana persamaannya dapat dilihat pada persamaan di bawah ini.

$$d(X_i,\ X_j = \sqrt{\sum_{r}^{n} = 1(a_r(x_i) - a_r\ x_j))}\quad ^2$$

Dimana pada persamaan, "d(Xi, Xj) merupakan jarak euclidean, (xi) merupakan record ke-1 (baris), (xj) merupakan record ke-j (kolom), (ar) merupakan data ke-r, dan yang terakhir i, j merupakan 1,2,3…n [23].

### *J. Implementasi Sistem*

Pada tahap ini akan dilakukan pengimplementasian model terbaik ke dalam aplikasi berbasis web agar model dapat digunakan untuk mengklasifikasikan pasien penyakit jantung.

## III. HASIL DAN PEMBAHASAN

### *A. Hasil Modelling*

Tahap awal dari hasil modelling diawali dengan preprocessing, yang meliputi data cleaning dan data transformation. Dimana setelah penghapusan missing value data yang sebelumnya berjumlah 445.132 baris, menjadi 246.022 baris. Berikutnya terdapat juga duplikat data sebanyak 9 baris yang langsung dihapus, sehingga data yang diolah menjadi 246.013 baris. Selanjutnya merupakan tahap transformasi data dimana pada tahap ini diawali dengan penghapusan variable yang tidak akan digunakan dengan alasan variable tersebut tidak relevan. Berikutnya dilakukan transformasi data dengan merubah isian dari variabel "AgeCategory". Dimana sebelumnya isian dari variable ini merupakan numeric lalu diubah menjadi kategori, yang mana dapat dilihat perubahannya pada Tabel 1.

TABEL I
HASIL TRANSFORMASI VARIABLE AGE CATEGORY

| **AgeCategoryCat** | |
|---|---|
| **AgeCategory** | **Hasil Kategori** |
| 25-39 | Adults |
| 40-59 | Middle-Aged-adults |
| 60-80+ | Other-Adults |

Hasil pada Tabel I merupakan hasil pengkategorian variable dimana sebelumnya variable ini memiliki data rentang umur. Yang diubah menjadi 3 kategori meliputi kelas Adults, Middle-Aged-Adults, dan Other-Adults. Selanjuntnya melakukan tahap perubahan isian dari variable BMI, dimana awalnya variabel ini berisikan numeric. Lalu akan diubah menjadi kategori seperti Tabel II.

TABEL II
HASIL TRANSFORMASI VARIABEL BMI

| **BMICat** | |
|---|---|
| **BMI** | **Hasil Kategori** |
| <18.50 | Underweight |
| >=18.50 | Healthy Weight |
| >=25.00 | Overweight |
| >=30 | Obese |

Pada Tabel II diketahui bahwa awalnya variabel BMI memiliki isian *numeric* yang selanjutnya diubah menjadi kategori dengan 4 kelas. Yang antara lain *Underweight*, *Healthyweight*, *Overweight*, dan *Obese*. Berikutnya merupakan perubahan pada variable "*PhysicalHealthDays*", menjadi kategori. Dimana hasilnya dapat dilihat pada Tabel III.

TABEL III
HASIL TRANSFORMASI VARIABEL PHYSICAL HEALTH DAYS

| **PhysicalHealthDaysCat** | |
|---|---|
| **PhysicalHealthDays** | **Hasil Kategori** |
| <10.0 | Under 10 Days |
| >=10.0 | Under 20 Days |
| >=20.0 | Under 30 days |

Pada Tabel III dilakukan perubahan data pada variabel "*PhysicalHealthDays*" menjadi kategori. Dimana data diubah dari *numeric* menjadi 4 kelas yaitu *Under 10 Day*s, *Under 20 Days*, dan *Under 30 Days*. Dilanjutkan dengan perubahan pada variabel "*MentalHealthDays*" dari *numeric* menjadi kategori. Berikut Tabel IV mendeskripsikan hasil dari perubahan data.

TABEL IV
HASIL TRANSFORMASI VARIABEL MENTAL HEALTH DAYS

| **MentalHealthDaysCat** | |
|---|---|
| **MentalHealthDays** | **Hasil Kategori** |
| <10.0 | Under 10 Days |
| >=10.0 | Under 20 Days |
| >=20.0 | Under 30 days |

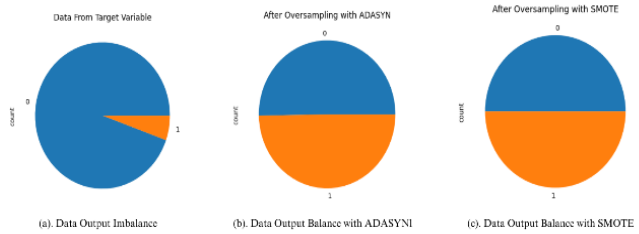
Berdasarkan Tabel IV data yang ada pada variabel "*MentalHealthDays*" diubah menjadi 4 kategori. Meliputi *Under 10 Days*, *Under 20 Days*, dan *Under 30 Days*. Selanjutnya dilakukan perubahan isian pada variabel "*SleepHours*" dari *numeric* menjadi kategori. Hasil perubahan data dapat dilihat pada Tabel V.

TABEL V
HASIL TRANSFORMASI VARIABEL SLEEP HOURS

| **SleepHoursCat** | |
|---|---|
| **SleepHours** | **Hasil Kategori** |
| <8.0 | Under 8 Hour |
| >=8.0 | Under 16 Hour |
| >=16.0 | Under 24 Hour |

Dapat dilihat pada Tabel V, perubahan dilakukan untuk mengkategorikan isian menjadi 3 kelas. Seperti *Under 8 Hours*, *Under 16 Hours*, dan *Under 24 Hours*. Selanjutnya setelah data seragam sebagi kategori, seluruh data diubah menjadi kategori dalam bentuk *numerik* agar dapat diolah oleh komputer. Dalam prosenya dilakukan menggunakan *library* "*label encoder*" dari *python*.

Selanjutnya merupakan tahap penyeimbangan data, hal ini dilakukan kaarna sebelumnya data yang diolah memiliki hasil output yang tidak seimbang. Adapun Metode yang digunakan untuk membuat data menjadi seimbang dengan menggunakan SMOTE dan ADASYN. Hasil dari penyeimbangan data dapat dilihat pada Gambar 3.




Gambar 1 Perbandingan Data Seimbang dan Data Tidak Seimbang

Pada Gambar 3 dapat dilihat bahwa data yang ada pada point (a) merupakan data awal, yang memiliki nilai tidak seimbang. Sedangkan pada point (b) dan (c) merupakan data yang sudah seimbang dengan metode SMOTE dan ADASYN. Adapun data yang tidak seimbang dimaksudkan dengan jumlah data yang sangat berbeda jumlahnya, seperti pada Gambar 3 di mana data kelas "0" atau "tidak memiliki penyakit jantung" memili nilai mayoritas > 70% data, hal ini dapat mengakibatkan model yang dibuat menjadi bias karena data dilatih dengan data yang menyimpan record lebih banyak pada kelas tertentu. Berdasarkan hal ini maka metode SMOTE dan ADASYN akan digunakan untuk membuat datanya menjadi seimbang dengan membuat data sintetis pada kelas minoritas hingga data menjadi seimbang. Selanjutnya pada tahap implementasi algoritma ketiga jenis data ini akan diuji untuk mengetahui performa dengan metode apakah yang mendapatkan hasil terbaik. Untuk kemudian diimplementasikan pada sistem pengklasifikasian penyakit serangan jantung berbasis web.

### *B. Hasil Cross Validation score*

Pengujian *cross validation* kali ini dilakukan menggunakan data yang seimbang dan tidak seimbang. Adapun skema pengujian dilakukan mengunakan data yang tidak seimbang, seimbang menggunakan SMOTE, dan seimbang menggunakan ADASYN. Adapun dalam pengujian *cross validation* ditentukan nilai CV = 10, serta dalam mendapatkan nilai *accuracy*, *precision*, *recall* dan f1-*score*, dilakukan penyesuaian pada parameter "*scoring*" yang terdapat pada pustaka "cross_val_score". Selanjutnya hasil tertinggi akan ditandai dengan arsiran berwarna abu-abu. Berikut pada Tabel VI merupakan hasil dari cross validation setiap algoritma.

TABEL VI
HASIL PENGUJIAN CROSS VALIDATION

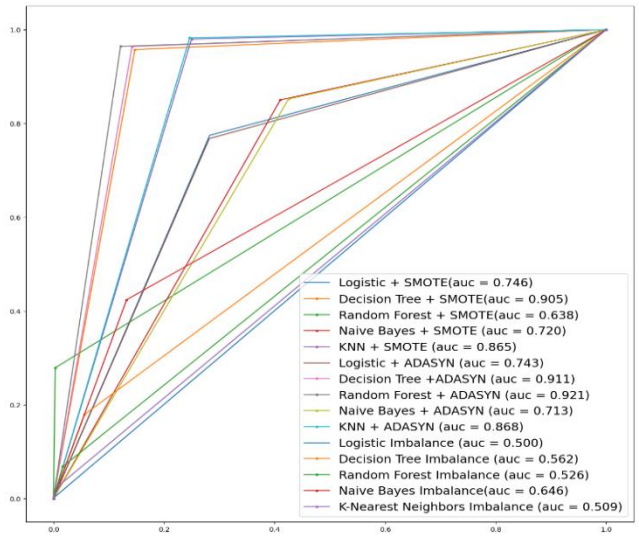
| CV=10 | Accuracy (%) | Precision (%) | Recall (%) | F1-Score (%) |
|---|---|---|---|---|
| Naïve Bayes Imbalance | 84.34 | 56.15 | 65.11 | 57.30 |
| Logistic Regression Imbalance | 94.50 | 64.81 | 50.24 | 49.11 |
| Decision Tree Imbalance | 89.93 | 54.80 | 55.62 | 55.15 |
| Random Forest Imbalance | 93.37 | 57.15 | 52.58 | 53.31 |
| K-NN Imbalance | 93.87 | 56.98 | 51.47 | 51.65 |
| Naïve Bayes + SMOTE | 72.08 | 73.62 | 72.09 | 71.62 |
| Logistic Regression + SMOTE | 74.56 | 74.63 | 74.57 | 74.55 |
| Decision Tree + SMOTE | 89.90 | 90.37 | 89.91 | 89.87 |
| Random Forest + SMOTE | 91.22 | 91.52 | 91.23 | 91.21 |
| K-NN + SMOTE | 86.33 | 88.35 | 86.35 | 86.15 |
| Naïve Bayes + ADASYN | 71.20 | 73.00 | 71.29 | 70.68 |
| Logistic Regression + ADASYN | 74.00 | 74.05 | 74.01 | 73.99 |
| Decision Tree + ADASYN | 90.49 | 90.92 | 90.52 | 90.47 |
| Random Forest + ADASYN | 91.65 | 91.95 | 91.68 | 91.64 |
| K-NN + ADASYN | 86.31 | 88.38 | 86.38 | 86.14 |

Pengujian *cross validation* pada Tabel VI didapatkan dengan data yang seimbang dan tidak seimbang. Adapun pengujian pada Tabel VI meliputi nilai akurasi, *precision*, *recall*, dan *f1-score*. Berdasarkan hasil yang didapat pada Tabel VI didapat hasil tertiggi pada nilai akurasi oleh metode *logistic regression* dengan nilai 94.50%. Sementara itu haril *precision* tertinggi sebesar 91.95%, *recall* 91.68, dan 91.64% pada nilai *f1-score*. Didapatkan oleh algoritma *random forest* dengan data yang seimbang menggunakan metode ADASYN. Hasil tertinggi pada *cross validation* serupa dengan *classification report*. Dimana akurasi tertinggi didapat oleh *logistic regression*, sedangkan nilai *precision*, *recall*, dan *f1-score* tertinggi didapat oleh algoritma *random forest*. Hal ini menjadi bias mengingat output akhir harus mendapatkan algoritma yang terbaik. Maka dari itu akan dilakukan evaluasi ROC dan AUC untuk menentukan algoritma dengan metode apa yang terbaik.

### *C. Hasil ROC dan AUC*

Evaluasi terhadap model dilakukan dengan mempertimbangkan perbandingan performa pada data seimbang dan tidak seimbang. Untuk mencapai keseimbangan pada dataset, *metode Synthetic Minority Over-sampling Technique* (SMOTE) dan *Adaptive Synthetic Sampling* (ADASYN) digunakan sebagai teknik oversampling yang efektif dalam menyeimbangkan proporsi kelas.

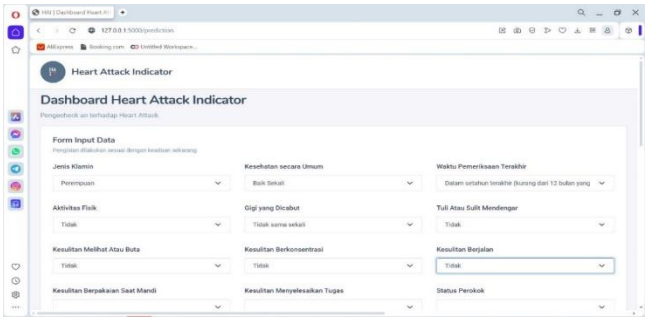
Pengukuran performa model dilakukan melalui analisis Receiver Operating Characteristic (ROC) dan perhitungan Area Under the Curve (AUC), yang memberikan gambaran mengenai kemampuan model dalam membedakan kelas positif dan negatif. Ilustrasi dari kurva ROC dan nilai AUC dapat dilihat pada Gambar 4, yang menunjukkan perbandingan kinerja model pada data sebelum dan sesudah proses penyeimbangan data.




Gambar 2 Hasil Evaluasi ROC dan AUC

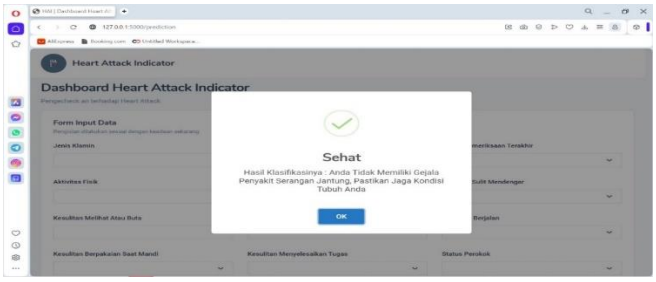
Hasil pada Gambar 4 menunjukan plot ROC dan AUC yang didapat menggunakan data testing. Dimana skema pengujian dibagi menjadi 3, pengujian pada data *imbalance*, pengujian dengan data *balance* + SMOTE, serta pengujian dengan data *balance* + ADASYN. Hasil pada gambar 5 menunjukan bahwa nilai tertinggi didapat oleh algoritma *random forest*, dengan data yang seimbang menggunakan metode ADASYN. Dengan nilai AUC sebesar 0.921, hal ini menandakan bahwa algoritma ini sangat baik dalam membedakan kelas pada kasus klasifikasi. Sedangkan nilai terendah didapat oleh algoritma *logistic regression* dengan nilai AUC sebesar 0.500. Hal ini juga membuktikan bahwa dipengujian sebelumnya memang akurasi yang didapat oleh *logistic regression* cukup tinggi. Namun tetap bahwa dipengujian sebelumnya nilai *precision*, *recall*, dan *f1-score* tetinggi didapat oleh *random forest*. Berdasarkan hal ini diketahui bahwa nilai akurasi belum cukup membuktikan bahwa model tersebut sudah optimal. Sehingga membutuhkan pengujian lainnya untuk dapat memastikan performa terbaik dari setiap algoritma. Diketahui juga bahwa berdasarkan *graphic* ROC dan AUC pada Gambar 4, serta pengujian sebelumnya. Mendapatkan hasil bahwa algoritma terbaik dari penelitian ini adalah *random forest*.

Berdasarkan hasil evaluasi model diketahui bahwa algoritma random forest ditambahkan dengan metode ADASYN mendapatkan nilai AUC tertinggi sebesar 0.920. Maka dari itu model ini akan digunakan pada sistem yang di rancang, adapun ilustrasi hasil sistem dapat dilihat pada Gambar 5.



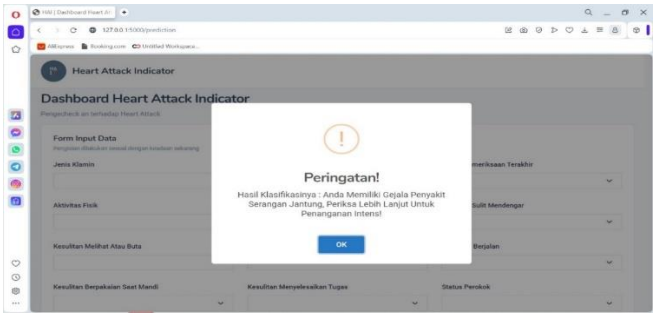

Gambar 3 Hasil Utama Sistem

Dapat dillihat pada Gambar 5 merupakan halaman utama ketika mengakses sistem. Pada halaman utama terdapat formulir yang dapat langsung diisi oleh pengguna. Bila sudah mengisi formulir pengguna dapat langsung menekan tombol klasifikasi agar pengguna bisa langsung melihat hasil klasifikasi dari sistem seperti pada Gambar 6.




Gambar 4 Hsil Bila Tidak Terindikasi Terkena Penyakit Serangan Jantung

Dapat dilihat pada Gambar 6 terdapat hasil dari pengguna yang sudah mengisi formulir dan menekan tombol klasifikasi. Pada Gambar merupakan hasil apabila pasien tidak terindikasi terkena penyakit serangan jantung. Namun bila pengguna terindikasi terkena penyakit serangan jantung maka sistem akan menampilkan notifikasi yang berbeda seperti pada Gambar 7.




Gambar 5 Hasil Bila Terindikasi Terkena Penyakit Serangan Jantung

Berdasarkan Gambar 7 diketahui bahwa ilustrasi tersebut menunjukan notifikasi yang akan muncul bila pengguna terindikasi penyakit serangan jantung.

## IV. KESIMPULAN

Berdasarkan hasil penelitian kali ini diketahui bahwa algoritma *random forest* menjadi algoritma yang memiliki performa terbaik. Adapun hal ini didukung dengan hasil nilai AUC yang didapat sebesar 0.920 dimana berdasarkan nilai ini model sudah dapat membedakan kelas dengan sangat baik. Diketahui juga bahwa berdasarkan hasil pengujian akurasi tidak dapat langsung menentukan model tersebut memilik performa yang terbaik. Hal ini didukung dengan hasil terendah yang didapatkan oleh metode *logistic regression*. Walaupun mendapatkan nilai akurasi tertinggi, tetapi nilai AUC nya sangat rendah sebesar 0.503. Walaupun demikian, hasil yang didapat oleh algoritma *logistic regression* pun tidak menandakan bahwa algoritma tersebut buruk, dalam hal ini bisa saja data yang digunakan tidak cocok untuk di olah oleh algoritma tersebut. Adapun model yang dibuat juga sudah dapat diimplementasikan ke dalam aplikasi berbasis web.

## DAFTAR PUSTAKA


[1] A. Rahmat, M. Syafiih, and M. Faid, "Implementasi Klasifikasi Potensi Penyakit Jantung dengan Menggunakan Metode C4.5 Berbasis Website ( Studi Kasus Kaggle.Com )," *INFOTECH journal*, vol. 9, no. 2, pp. 393–400, Jul. 2023, doi: 10.31949/infotech.v9i2.6295.

[2] T. A. Munandar and A. Q. Munir, "Implementasi K-Nearest Neighbor Untuk Prototype Sistem Pakar Identifikasi Dini Penyakit Jantung K-Nearest Neighbor for Prototype Expert System for Early Identification of Heart Disease," *Jurnal Teknologi Informasi*, vol. XVII, no. 2, pp. 44–50, 2022.

[3] B. Hirwono, A. Hermawan, and D. Avianto, "Implementasi Metode Naïve Bayes untuk Klasifikasi Penderita Penyakit Jantung," *Jurnal Teknologi Informasi dan Komunikasi)*, vol. 7, no. 3, pp. 451–457, 2023, doi: 10.35870/jti.

[4] A. Riani, Y. Susianto, and N. Rahman, "Implementasi Data Mining Untuk Memprediksi Penyakit Jantung Mengunakan Metode Naive Bayes," *Journal of Innovation Information Technology and Application (JINITA)*, vol. 1, no. 01, pp. 25–34, Dec. 2019, doi: 10.35970/jinita.v1i01.64.

[5] J. Waruwu and A. Dharma, "Perbandingan Algoritma Klasifikasi Pada Pasien Penyakit Jantung Comparison Of Classification Algorithms In Heart Disease Patients," *Journal of Information Technology and Computer Science (INTECOMS)*, vol. 7, no. 5, 2024, [Online]. Available: https://www.kaggle.com/datasets/mexw

[6] I. Optimasi *et al.*, "Implementasi Optimasi Hyperparameter GridSearchCV Pada Sistem Prediksi Serangan Jantung Menggunakan SVM," *Online) Teknologi: Jurnal Ilmiah Sistem Informasi*, vol. 13, no. 1, pp. 8–15, 2023, doi: 10.26594/teknologi.v13i1.3098.

[7] D. Sitanggang, N. Nicholas, V. Wilson, A. R. A. Sinaga, and A. D. Simanjuntak, "Implementasi Data Mining untuk Memprediksi Penyakit Jantung Menggunakan Metode K-Nearest Neighbor dan Logistic Regression," *Jurnal Teknik Informasi dan Komputer (Tekinkom)*, vol. 5, no. 2, p. 493, Dec. 2022, doi: 10.37600/tekinkom.v5i2.698.

[8] D. P. Utomo and M. Mesran, "Analisis Komparasi Metode Klasifikasi Data Mining dan Reduksi Atribut Pada Data Set Penyakit Jantung," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 4, no. 2, p. 437, Apr. 2020, doi: 10.30865/mib.v4i2.2080.

[9] H. Azis, P. Purnawansyah, F. Fattah, and I. P. Putri, "Performa Klasifikasi K-NN dan Cross Validation Pada Data Pasien Pengidap Penyakit Jantung," *ILKOM Jurnal Ilmiah*, vol. 12, no. 2, pp. 81–86, Aug. 2020, doi: 10.33096/ilkom.v12i2.507.81-86.

[10] A. Sari, A. Sihananto, and D. Prasetya, "Implementasi Metode K-NN dalam Klasterisasi Kasus Kesehatan Jantung," *ALINIER JURNAL*, vol. 3, no. 2, pp. 95–99, 2022, [Online]. Available: www.elektro.itn.ac.id

[11] Sahar, "Analisis Perbandingan Metode K-Nearest Neighbor dan Naïve Bayes Classiffier pada Data Set Penyakit Jantung," *Indonesian Journal of Data and Science (IJODAS)*, vol. 1, no. 3, pp. 79–86, 2020.

[12] R. Pranandito, "Pperbandingan Pprediksi Peyakit Serangan Jantung Menggunakan Model Machine Learning," *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 8, no. 4, pp. 1228–1237, 2023, doi: 10.29100/jipi.v8i4.4165.

[13] M. G. Pradana, P. H. Saputro, and D. P. Wijaya, "Komparasi Metode Support Vector Machine dan Naive Bayes Dalam Klasifikasi Peluang Penyakit Serangan Jantung," *Indonesian Journal of Business Intelligence (IJUBI)*, vol. 5, no. 2, p. 87, Dec. 2022, doi: 10.21927/ijubi.v5i2.2659.

[14] J. Pangaribuan, C. Tedja, and S. Wibowo, "Perbandingan Metode Algoritma C4.5 dan Extreme Learning Machine untuk Mendiagnosis Penyakit Jantng Korner," *Informatics Engineering Research And Technology*, vol. 1, no. 1, pp. 1–7, 2019.

[15] I. Kusuma and N. Cahyono, "Analisis Sentimen Masyarakat Terhadap Penggunaan E-Commerce Menggunakan Algoritma K-Nearest Neighbor," *Jurnal Pengembangan IT (JPIT)*, vol. 8, no. 3, pp. 302–307, 2023.

[16] B. Rachmat, A. Suwarisman, I. Afriyanti, A. Wahyudi, and D. Saputra, "Analisis Sentimen Complaindan Bukan Complainpada Twitter Telkomsel dengan SMOTEdan Naïve Bayes," *Jurnal Teknologi Informasi dan Komunikasi)*, vol. 7, no. 1, pp. 107–113, 2023, doi: 10.35870/jti.

[17] J. Al Amien, Yoze Rizki, and Mukhlis Ali Rahman Nasution, "Implementasi Adasyn Untuk Imbalance Data Pada Dataset UNSW-NB15 Adasyn Implementation For Data Imbalance on UNSW-NB15 Dataset," *Jurnal CoSciTech (Computer Science and Information Technology)*, vol. 3, no. 3, pp. 242–248, Dec. 2022, doi: 10.37859/coscitech.v3i3.4339.

[18] A. Watratan, A. B, and D. Moeis, "Implementasi Algoritma Naive Bayes Untuk Memprediksi Tingkat Penyebaran Covid-19 Di Indonesia," *Journal of Applied Computer Science and Technology (JACOST)*, vol. 1, no. 1, pp. 7–14, 2020, [Online]. Available: http://journal.isas.or.id/index.php/JACOST

[19] Y. A'yunan, U. Indahyanti, and S. Busono, "Implementasi Data Mining Dalam Klasifikasi Diagnosa Kanker Payudara Menggunakan Algoritma Logistic Regresion," *Jurnal TEKINKOM*, vol. 6, no. 2, pp. 400–407, 2023, doi: 10.37600/tekinkom.v6i2.948.

[20] Hozairi, Anwari, and S. Alim, "Implementasi Orange Data Mining untuk Klasifikasi Kelulusan Mahasiswa dengan Model K-Nearest Neighbor, Decision Tree serta Naive Bayes," *Jur nal Ilmiah NERO*, vol. 6, no. 2, pp. 133–144, 2021.

[21] G. Mursianto, I. Falih, M. Irfan, T. Sakinah, and D. Sandya, "Perbandingan Metode Klasifikasi Random Forest dan XGBoost Serta Implementasi Teknik SMOTE pada Kasus Prediksi Hujan," *Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)*, pp. 41–50, 2021.

[22] D. P. Sinambela, H. Naparin, M. Zulfadhilah, and N. Hidayah, "Implementasi Algoritma Decision Tree dan Random Forest dalam Prediksi Perdarahan Pascasalin," *Jurnal Informasi dan Teknologi*, vol. 5, no. 3, pp. 58–64, Sep. 2023, doi: 10.60083/jidt.v5i3.393.

[23] A. Khairi, A. Fais Ghozali, and A. Darul Nur Hidayah, "Implementasi K-Nearest Neighbor (KNN) untuk Klasifikasi Masyarakat Pra Sejahtera Desa Sapikerep Kecamatan Sukapura," *TRILOGI: Jurnal Ilmu Teknologi, Kesehatan, dan Humaniora*, vol. 2, no. 3, pp. 319–323, 2021.