Jurnal Masyarakat Informatika, 15(1), 2024
DOI:10.14710/jmasif.xx.x.xxxxx
ISSN: 2777-0648

Analisis Prediksi Penyakit Diabetes Menggunakan Metode Decision Tree: Perbandingan dengan Random Forest dan XGBoost

Juliyana Herman
Jurusan Teknik Informatika, Universitas Halu Oleo, Kendari, Indonesia
* *Corresponding author: julianaoppo87gmail@email.com*

Abstrak

Diabetes merupakan penyakit kronis yang memerlukan perhatian khusus dalam deteksi dini untuk mencegah komplikasi lebih lanjut. Penelitian ini bertujuan mengevaluasi kinerja tiga algoritma marchine learning, yaitu Decision Tree, Random Forest, dan XGBoost, dalam memprediksi penyakit diabetes menggunkan dataset Pima Indians Diabetes Dataset. Proses penelitian melibatkan tahapan praprocessing, seperti penanganan missing value, outlier, dan normalisasi, serta pembagian data untuk pelatihan dan pengujian model.

Hasil penelitian menunjukkan bawah algoritma terbaik dengan akurasi tinggi, efisiensi, dan kemampuan menangani data tidak seimbang. Random Forest unggul dalam generalisasi melalui pendekatan ansambel, meskipun membutuhkan waktu komputasi lebih lama. Decision Tree, meskipun sederhana dan mudah diinterpretasikan, menunjukkan akurasi yang lebih rendah dibanding algoritma lainnya pada data dengan kompleksitas tinggi.

Penelitian ini memberikan kontribusi signifikan dalam pemanfaatan algoritma machine learning untuk mendukung deteksi dini penyakit diabetes. Pengembang lebih lanjut diusulkan dengan menggunakan dataset yang lebih besar dan penerapan teknik optimasi untuk meningkatkan performa model prediksi.

Kata kunci : *Diabetes, decision tree, random forest, xgboost*

Abstract

Diabetes is a chronic disease that requires special attention in early detection to prevent further complications. This study aims to evaluate the performance of three marchine learning algorithms, namely Decision Tree, Random Forest, and XGBoost, in predicting diabetes using the Pima Indians Diabetes Dataset. The research process involves preprocessing stages, such as handling missing values, outliers, and normalization, as well as data division for model training and testing.

The results showed the best algorithm with high accuracy, efficiency, and ability to handle unbalanced data. Random Forest excels in generalization through an ensemble approach, although it requires longer computation time. Decision Tree, although simple and easy to interpret, shows lower accuracy than other algorithms on data with high complexity.

This research makes a significant contribution in the utilization of machine learning algorithms to support early detection of diabetes. Further development is proposed by using larger datasets and applying optimization techniques to improve the performance of the prediction model.

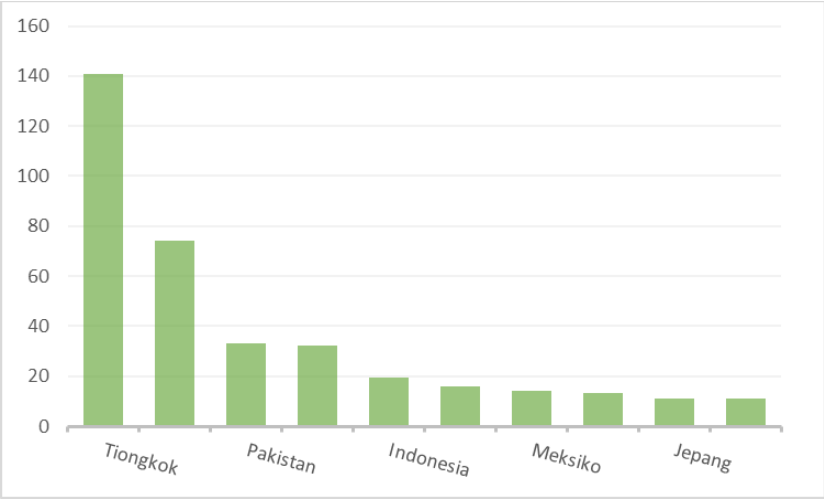
Keywords : *Diabetes, decision tree, random forest, xgboost*

1 Pendahuluan

Received: dd month yyyy ; Revised: dd month yyyy; Accepted: dd month yyyy; Published: dd month yyyy

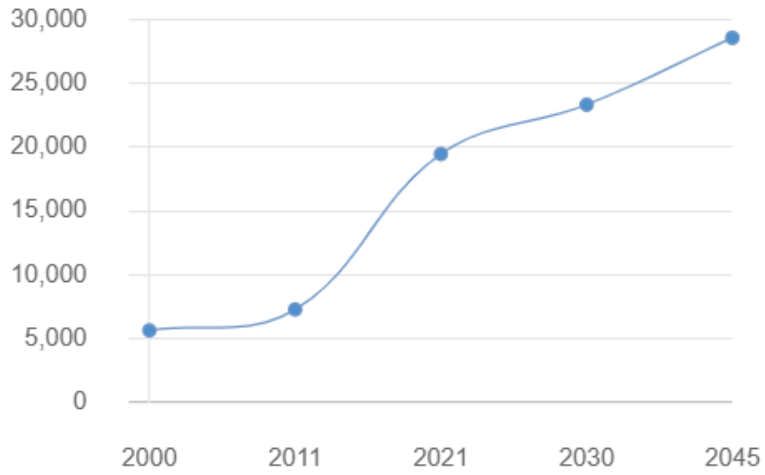
Diabetes adalah kondisi kronis yang terjadi saat pankreas tidak dapat lagi memproduksi insulin, atau tubuh tidak dapat menggunakan insulin secara efektif [1]. Insulin adalah hormon yang dibuat oleh pankreas yang berfungsi sebagai kunci untuk menyalurkan glukosa dari makanan yang kita makan dari aliran darah ke dalam sel-sel tubuh untuk menghasilkan energi [2]. tubuh memecahkan semua makanan karbohidrat menjadi glukosa dalam darah, dan insulin membantu glukosa bergerak ke dalam sel-sel. Bila tubuh tidak memproduksi atau menggunakan insulin secara efektif, hal ini menyebabkan kadar glukosa darah tinggi, yang disebut *hiperglikemia* [3].

Organisasi *Internasional Diabetes Federation* (IDF) mencatatkan penderita penyakit diabetes di dunia pada tahun 2021 sebanyak 537 juta orang dengan rentang usia 20-79 tahun, jumlah data tersebut diperkirakan akan terus bertambah dan sampai dengan tahun 2045 sebanyak 700 juta orang. Seperti yang terlihat pada Gambar 1, Indonesia menduduki peringkat kelima dengan jumlah penderita diabetes sebanyak 19,47 juta orang pada tahun 2021.



Gambar 1 Sepuluh Negara Dengan Kasus Penyakit Diabetes Tertinggi Tahun 2021

Jumlah penderita diabetes di Indonesia terus mengalami peningkatan dari tahun ke tahun. Berdasarkan data dari *International Diabetes Federation* (IDF), jumlah penderita diabetes meningkat sebesar 13,82 juta orang dari tahun 2000 hingga tahun 2021. IDF memprediksi jumlah penderita penyakit diabetes di Indonesia pada tahun 2045 sebanyak 28.57 juta orang. Kondisi ini seharusnya menjadi perhatian yang serius bagi masyarakat Indonesia untuk selalu menjaga kesehatan dengan pola hidup sehat serta mencegah terjadinya penyakit diabetes dengan mengetahui penyebab dan gejala penyakit diabetes sejak dini.



Gambar 2 Peningkatan Jumlah Penderita Penyakit Diabetes di Indonesia

Pada masa industri 4.0 ini, teknologi informasi dan komunikasi bisa digunakan untuk mempermudah dalam memprediksi penyakit diabetes tersebut. Salah satu caranya dengan menggunakan suatu algoritma klasifikasi pada data mining [4]. Adapun metode yang paling banyak digunakan adalah machine learning. Machine learning dapat melakukan pembelajaran dari data sehingga memungkinkan komputer untuk melakukan klasifikasi atau prediksi berdasarkan data yang diberikan [5]. Pendekatan ini memungkinkan analisis data dalam skala besar, yang pada gilirannya menghasilkan model prediktif dan klasifikasi penyakit dengan tingkat akurasi yang tinggi [6].

Dalam penelitian ini, akan difokuskan pada penggunaan tiga metode klasifikasi yang populer, yaitu Decision Tree, Random Forest, dan XGBoost, untuk memprediksi penyakit diabetes. Decision tree merupakan salah satu teknik yang dapat digunakan untuk melakukan klasifikasi terhadap sekumpulan objek atau record. Teknik ini terdiri dari kumpulan decision node, dihubungkan oleh cabang, bergerak ke bawah dari *root node* sampai berakhir di *leaf node* [7]. Random Forest adalah pengembangan dari metode Decision Tree yang menggunakan beberapa Decision Tree, dimana setiap Decision Tree telah dilakukan pelatihan menggunakan sampel individu dan setiap atribut dipecah pada pohon yang dipilih antara atribut subset yang bersifat acak [8]. Random Forest mempunyai proses seleksi fitur dimana mampu mengambil fitur terbaik sehingga dapat meningkatkan performa terhadap model klasifikasi [9]. Algoritma XGBoost adalah mesin yang dapat diskalakan sistem pembelajarannya untuk meningkatkan pohon keputusan. Adapun keunggulan dari model pembelajaran XGBoost memiliki kecepatan, skalabilitas, efisiensi, dan kesederhanaan yang sangat cepat [10].

Penelitian ini bertujuan untuk menganalisis kinerja metode Decision Tree dalam memprediksi penyakit diabetes, dengan membandingkannya dengan metode Random Forest dan XGBoost. Dataset yang digunakan berisi beberapa fitur medis dan biokimia, seperti jumlah pregnancies, glucose, blood pressure, skin thickness, insulin, BMI, diabetes pedigree function, dan age, yang relevan dengan penyakit diabetes. Selanjutnya, model dilatih menggunakan ketiga metode tersebut dan kinerjanya dievaluasi menggunakan metrik seperti akurasi, presisi, recall, dan F1-score. Diharapkan melalui penelitian ini, dapat diperoleh pemahaman yang lebih baik mengenai efektivitas masing-masing metode dalam prediksi penyakit diabetes, serta memberikan wawasan baru dalam penerapan metode Decision Tree, Random Forest, dan XGBoost untuk klasifikasi penyakit diabetes. Hasil penelitian ini juga diharapkan dapat menjadi dasar untuk pengembangan metode klasifikasi yang lebih akurat di bidang medis.

Data mining adalah proses yang menggunakan teknik statistik, matematika, kecerdasan buatan, dan machine learning untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari berbagai database besar [11]. Pada penelitian digunakan data mining untuk menganalisis penyakit diabetes dengan menggunakan Decision Tree, Random Forest, dan XGBoost untuk melakukan komparasi antara ketiga metode ini dalam menentukan metode mana yang lebih optimal. Data yang digunakan dalam penelitian ini berasal dari Institut Nasional Diabetes, Pencernaan, dan Penyakit Ginjal. Dataset ini mencakup informasi medis dari pasien perempuan berusia minimal 21 tahun yang merupakan keturunan Indian Pima. Data ini dapat diakses melalui https://www.kaggle.com/organizations/uciml/datasets.

2 Penelitian Terkait

Beberapa penelitian sebelumnya telah mengkaji penggunaan algoritma mechine learning dalam memprediksi penyakit diabetes. Achmad Afifuddin dan Lukman Hakim (2023) menggunakan algoritma Decision Tree C4.5 untuk membangun model prediksi diabetes, dengan memanfaatkan dataset yang diperoleh dari kaggle [12]. Hasil akurasi dari model prediksi mencapai 96%, dengan nilai precision sebesar 0.99, recall 0.95, dan f1-score 0.97. Tujuan dari penelitian ini adalah mempermudah deteksi mandiri sebelum berkonsultasi dengan dokter.

Umi Kulsum Indah Lestari, Anis Yusrotun Nadhiroh, dan Cahyuni Novia (2021) menerapkan algoritma K-Nearest Neighbor (KNN) untuk mengidentifikasi resiko seseorang menderita diabetes [13]. Pengujian dilakukan menggunakan Pima Indians Diabetes Database sebagai dataset, dan hasilnya menunjukkan akurasi sebesar 96% dengan nilai k optimal sebesar 23. Terdapat penelitian serupa yang menerapkan metode KNN untuk membangun sistem diagnosa diabetes [14]. Dengan nilai k sebesar 3 tingkat akurasi mencapai 85,71%. Penelitian ini bertujuan membantu mendiagnosa diabetes berdasarkan gejala yang diinputkan oleh pasien.

Selain itu, penelitian yang dilakukan oleh Nurlaelatul Maulidah, Riki Supriyadi, Dwi Yuni Utami, Fuad Nur Hasan, Ahmad Fauzi, dan Ade Christian (2021) Menggunakan Metode S*upport Vector Machine* dan *Naive Bayes* untuk memprediksi penyakit diabetes. Dataset yang digunakan adalah dataset Diabetes yang diperoleh dari kaggle. Hasil menunjukkan bahwa metode SVM memiliki akurasi lebih tinggi sebesar 78,04%, dibandingkan dengan metode Naive Bayes yang memiliki akurasi 76,98%. Selisih akurasi antara kedua metode adalah 1,06%, sehingga SVM dianggap lebih efektif dalam klasifikasi diagnosis diabetes [15].

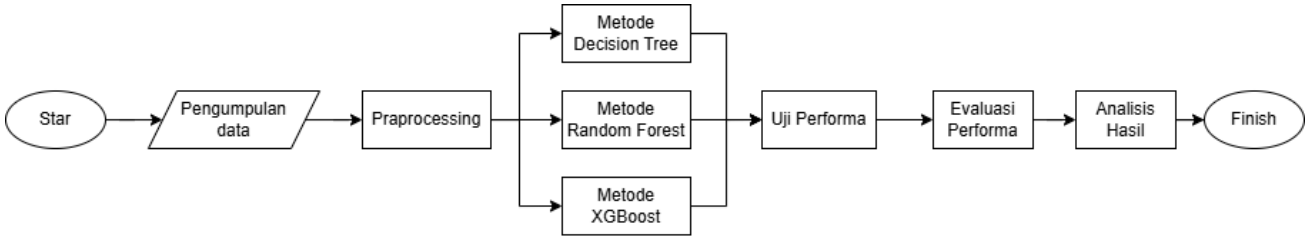
Berdasarkan hasil dari penelitian-penelitian sebelumnya menunjukan bahwa algoritma machine learning, seperti Decision Tree, KNN, SVM, dan Naive Bayes, memiliki potensi dalam membantu deteksi dini dan diagnosis penyakit diabetes. Dengan penerapan teknologi ini, diharapkan proses pengambilan keputusan medis dapat menjadi lebih cepat dan akurat.

3 Metode Penelitian

Pada bagian ini memaparkan bagaimana pola alur pengerjaan penelitian yang dilakukan. Digambarkan secara rinci mengenai metode dan proses yang digunakan dalam melakukan prediksi penyakit diabetes menggunakan metode decision tree, random forest dan xgboost. Berikut ini adalah tahapan-tahapannya:

3.1 Desain Sistem

Berikut desain sistem yang dibuat:



Gambar 3 Desain Sistem

Gambar 3 merupakan alur perancangan sistem pada penelitian ini, berjalan melalui beberapa tahapan mulai dari pengumpulan data, preprocessing data, processing, pemodelan metode (Decision Tree, Random Forest dan XGBoost), uji performa, evaluasi performa menggunakan confusion matrix, hingga analisis hasil.

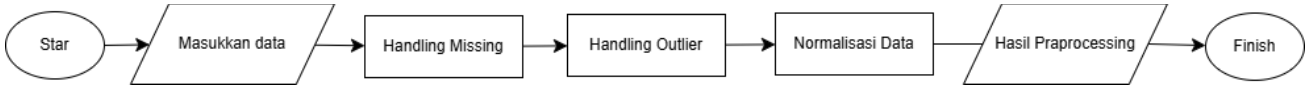
3.2 Data

Data yang digunakan dalam penelitian ini berasal dari Institut Nasional Diabetes dan terdiri dari 768 entri. Data ini dapat diakses melalui https://www.kaggle.com/organizations/uciml/datasets. Dataset ini mencakup informasi medis dari pasien perempuan berusia minimal 21 tahun yang merupakan keturunan Indian Pima. Terdapat 9 indikator terkait dengan penyakit diabetes, yang digunakan dalam penelitian ini, sebagai berikut:

a) Pregnancies
b) Glucose
c) Blood Pressure
d) Skin Thickness
e) Insulin
f) BMI
g) Diabetes pedigree function
h) Age
i) Outcome

3.3 Preprocessing

Dataset tidak dapat langsung digunakan oleh sistem dan memerlukan tahap preprocessing. Tujuan preprocessing dilakukan yakni untuk menghilangkan masalah data dan meningkatkan kualitas data sebelum dilakukan pemodelan [16].



Gambar 4 Tahap Preprocessing

Pada Gambar 2 ditampilkan tahapan-tahapan dalam preprocessing data sebelum pemodelan menggunakan Decision Tree, Random Forest, dan XGBoost. Beberapa tahapan penting yang

dilakukan meliputi penanganan missing value, penanganan outlier, dan normalisasi data. Output yang dihasilkan akan digunakan sebagai input untuk permodelan dari ketiga metode.

3.3.1 Handling Missing Value

Tahap pertama pada preprocessing, data masukan terlebih dahulu akan melalui proses penanganan missing value. Nilai yang hilang dapat terjadi akibat kesalahan penginputan data atau karena tidak ada data yang disimpan untuk partisipan (variabel) tertentu. Algoritma pembelajaran mesin tidak dapat memproses data yang mengandung nilai yang hilang, maka penanganan missing value dilakukan sebelum proses pemodelan [17]. Peneliti menggunakan teknik imputasi mean, di mana nilai missing value pada suatu kolom di ganti dengan nilai rata rata dari kolom tersebut. Berikut rumus dari imputasi mean:

$$x_i = \frac{\sum_{j=1}^{n} x_j}{n} \quad (1)$$

Keterangan:

X_i : Nilai – nilai yang ada (tidak hilang).

3.3.2 Handling Outlier

Tahap selanjutnya adalah data yang sudah dibersihkan dari missing value akan melalui proses headling data outlier. Data outlier merupakan data yang memiliki nilai yang jauh berbeda dari rata-rata. Penanganan outlier dilakukan dengan menerapkan teknik Z-score. Secara umum, data dengan nilai Z-score kurang dari -3 atau lebih dari +3 dianggap sebagai nilai ekstrim [18]. Oleh karena itu, data yang berada diluar batas tersebut akan dihapus dari dataset. Adapun rumus Z-score yang digunakan sebagai berikut:

$$z = \frac{x - \mu}{\sigma} \quad (2)$$

Keterangan:

X : Nilai data uji.

μ (mu) : Rata-rata (mean) dari seluruh data.

σ (sigma) : Deviasi standar dari data.

3.3.3 Normalisasi Data

Tahap terakhir dari praprocessing adalah data yang sudah melewati proses headling missing value dan headling outlier akan dinormalisasikan. Hal ini dilakukan sebab kebanyakan data memiliki rentang nilai yang berbeda antar variabel. Untuk menormalisasikan data, peneliti menggunakan metode min-max scaler. Dimana metode ini akan menyesuaikan nilai-nilai data dalam rentang yang telah ditentukan, biasanya antara 0 hingga 1, sehingga memungkinkan perbandingan yang lebih adil antar variabel [19].

3.4 Processing

Setelah melalui tahap praprocessing, dataset kemudian digunakan sebagai input untuk membangun model prediksi menggunakan tiga metode yang telah ditentukan. Langkah pertama adalah membuka data ke dalam jupyter notebook dengan menggunakan text editor visual studio code. Ketika data berhasil diload, kemudian dilakukan pembagian data menjadi data latih dan data uji menggunakan

modul scikit-learn yaitu train_test_split. Untuk besaran masing-masing yaitu 80% untuk data latih dan 20% untuk data uji.

Setelah itu, data latih dan uji akan digunakan dalam tiga algoritma pembelajaran mesin yakni Decision Tree, Random Forest, dan XGBoost. Untuk data latih digunakan untuk melatih model, sementara data uji digunakan untuk menguji sejauh mana model dapat melakukan prediksi pada data yang belum dilihat sebelumnya. Untuk inisalisasi dari masing-masing metode sebagai berikut:

a) Decision Tree menggunakan kriteria Gini dan membatasi kedalam pohon hingga 5. Parameter random state sama dengan 42.
b) Random Forest menggunakan 100 pohon keputusan (n_estimators=100) dan menggunakan kriteria Gini. Parameter random state sama dengan 42.
c) XGBoost dengan menonaktifkan encoder label lama (use_label_encoder=False) dan menggunakan evaluasi logloss. Parameter random state sama dengan 42.

3.5 Tahap Evaluasi

Tahap evaluasi bertujuan untuk mengevaluasi performa model klasifikasi yang sudah dibangun. Untuk mengevaluasi performa model peneliti menggunakan 5 matrix evaluasi yaitu precision, recall, f1-score, sensitivity, dan specificity. Untuk menghitung sensitivity dan specivity peneliti membagi klasifikasi confusion matriks menjadi 4 elemen utama, yaitu:

1. True Positives (TP): Kondisi dimana model memprediksi kelas positif dengan benar, yaitu data yang sebenarnya sakit diprediksi sakit.
2. False Positives (FP): Kondisi dimana model memprediksi kelas positif, tetapi kenyataannya data tersebut negatif, yaitu data yang sebenarnya tidak sakit diprediksi sakit.
3. True Negative (TN): Kondisi dimana model memprediksi kelas negatif dengan benar, yaitu data yang sebenarnya tidak sakit diprediksi tidak sakit.
4. False Negative (FN): Kondisi dimana model memprediksi kelas negatif, tetapi kenyataanya data tersebut positif, yaitu data yang sebenarnya tidak sakit diprediksi sakit

Dari keempat elemen ini, sensitivity dan specivity dihitung dengan rumus:

$$Sensitivity = \frac{TP}{TP + FN} \quad (3)$$

$$Specivity = \frac{TN}{TN + FP} \quad (4)$$

Keterangan:

TP : Jumlah data positif yang diprediksi dengan benar
FP : Jumlah data negatif yang diprediksi sebagai positif
TN : Jumlah data negatif yang diprediksi dengan benar
FN : Jumlah data positif yang diprediksi sebagai negatif

4 Hasil dan Pembahasan

4.1 Data

Penelitian ini menganalisis dataset yang terdiri dari 768 entri, yang mencakup 9 indikator yang relevan dengan penyakit diabetes. Dalam penelitian ini, 9 filter indikator tersebut telah diidentifikasi. Penggunaan 9 indikator ini bertujuan untuk memperjelas batasan ruang cakupan dalam prediksi penyakit diabetes, seperti yang dijelaskan pada Tabel 1 :

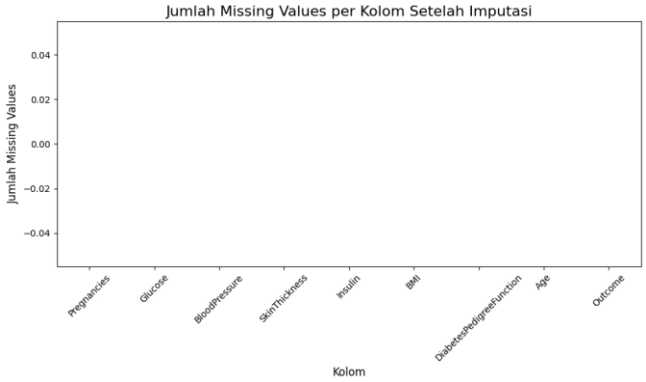
Tabel 1 Indikator Data

Attribut	Keterangan
Pregnancies	Jumlah kehamilan
Glucose	Konsentrasi glukosa plasma 2 jam setelah uji toleransi glukosa oral
BloodPresure	Tekanan darah diastolik
SkinThickness	ketebalan lipatan kulit trisep
Insulin	insulin serum 2 jam
BMI	indikator untuk menentukan kategori berat badan
DiabetesFunction	fungsi silsilah diabetes
Age	Umur
Outcome	Kelas

4.2 Preprocessing

4.2.1 Handling missing value

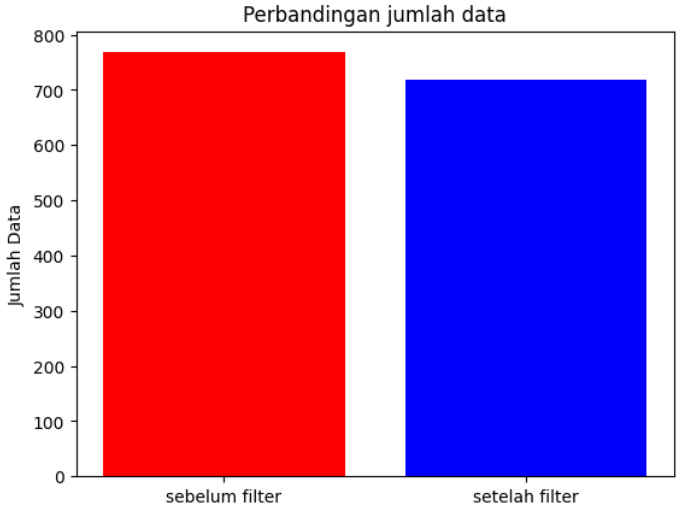
Garfik berikut merupakan data yang telah dibersihkan dari nilai yang hilang. Dapat disimpulkan bahwa semua kolom, termaksud Pregnancies, Glucose, BloodPressure, SkinThickness, Insulin, BMI, DiaabetesPedigreeFunction, Age, dan Outcome, tidak terdapat nilai yang hilang (missing value). Hal itu ditunjukan dengan nilai 0 untuk jumlah missing value disetiap kolomnya. Sehingga dapat diambil kesimpulan bahwa pengisian nilai telah berhasil dilakukan disemua kolom dengan baik.



Gambar 5 Hasil Missing Value

4.2.2 Handling Outlier

Tahap berikutnya adalah penanganan outlier menggunakan metode statistika yakni Z-score. Metode ini digunakan untuk mengidentifikasi data yang tergolong outliet, yaitu data yang memiliki nilai jauh dari rata-rata. Ketentuan yang digunakan adalah jika nilai Z-score lebih kecil dari -3 atau lebih besar dari +3, data dianggap sebagai nilai ekstrem. Oleh karena itu, data yang melebihi batas-batas ini akan otomatis dihapus dari dataset.

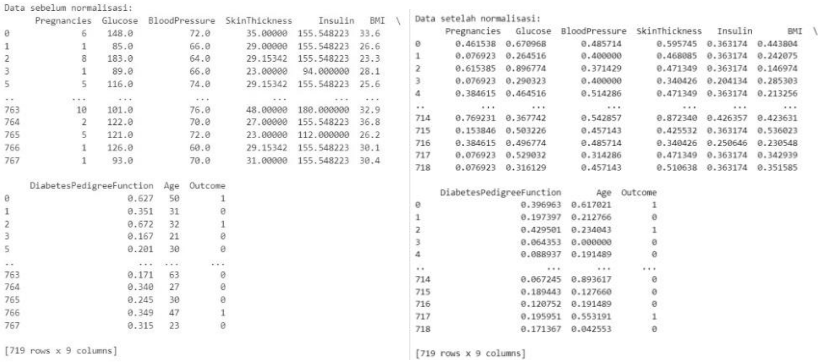


Gambar 6 Hasil Handling Outllier

Pada grafik, terlihat perbandingan jumlah data sebelum dan sesudah proses filtering outlier . Dimana sebelum filtering dataset berjumlah 768, sedangkan setelah proses filtering berkurang menjadi 719. Pengurangan ini mengindikasikan bahwa data yang termasuk kategori outlier telah berhasil dihapus dari dataset.

4.2.3 Normalisasi Data

Normalisasi data adalah langkah utama dalam preprocessing untuk memastikan konsistensi semua variabel berada dalam skala yang konsisten. Dalam penelitian ini, normalisasi dilakukan menggunakan metode skala min-max, yang mengubah setiap nilai data ke dalam rentang 0 hingga 1[19]. Tujuannya untuk mempermudah perbandingan antar variabel yang memiliki rentang nilai berbeda.



Data sebelum normalisasi:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148.0	72.0	35.00000	155.548223	33.6	0.627	50	1
1	1	85.0	66.0	29.00000	155.548223	26.6	0.351	31	0
2	8	183.0	64.0	29.15342	155.548223	23.3	0.672	32	1
3	1	89.0	66.0	23.00000	94.000000	28.1	0.167	21	0
5	5	116.0	74.0	29.15342	155.548223	25.6	0.201	30	0
..	...	...	...	...	...	...	...	...	...
763	10	101.0	76.0	48.00000	180.000000	32.9	0.171	63	0
764	2	122.0	70.0	27.00000	155.548223	36.8	0.340	27	0
765	5	121.0	72.0	23.00000	112.000000	26.2	0.245	30	0
766	1	126.0	60.0	29.15342	155.548223	30.1	0.349	47	1
767	1	93.0	70.0	31.00000	155.548223	30.4	0.315	23	0

[719 rows x 9 columns]

Data setelah normalisasi:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	0.461538	0.670968	0.485714	0.595745	0.363174	0.443804	0.396963	0.617021	1
1	0.076923	0.264516	0.400000	0.468085	0.363174	0.242075	0.197397	0.212766	0
2	0.615385	0.896774	0.371429	0.471349	0.363174	0.146974	0.429501	0.234043	1
3	0.076923	0.290323	0.400000	0.340426	0.204134	0.285303	0.064353	0.000000	0
4	0.384615	0.464516	0.514286	0.471349	0.363174	0.213256	0.088937	0.191489	0
..	...	...	...	...	...	...	...	...	...
714	0.769231	0.367742	0.542857	0.872340	0.426357	0.423631	0.067245	0.893617	0
715	0.153846	0.503226	0.457143	0.425532	0.363174	0.536023	0.189443	0.127660	0
716	0.384615	0.496774	0.485714	0.340426	0.250646	0.230548	0.120752	0.191489	0
717	0.076923	0.529032	0.314286	0.471349	0.363174	0.342939	0.195951	0.553191	1
718	0.076923	0.316129	0.457143	0.510638	0.363174	0.351585	0.171367	0.042553	0

[719 rows x 9 columns]

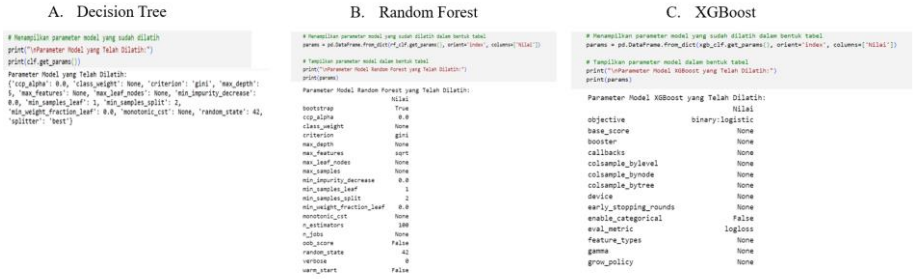
Gambar 7 Hasil Normalisasi Data

Berdasarkan gambar 7, setelah tahap headling missing dan headling outlier, dataset yang dihasilkan berjumlah 719 baris, seperti yang digambarkan pada tabel “Data sebelum normalisasi”. Data ini masih memiliki rentang nilai yang bervariasi pada setiap tabelnya, salah satu contoh adalah nilai pada kolom Glucose mencapai ratusan sedangkan pada kolom BMI hanya berada pada skala puluhan. Untuk mengatasi masalah perbedaan skala antar variabel, peneliti melakukan normalisasi. Untuk hasil normalisasi dapat dilihat pada tabel “Data setelah dinormalisasi”, dimana setiap nilai

dikonversi kedalam rentang 0 hingga 1. Sehingga proses ini memastikan agar semua variabel memiliki skala yang seragam. Dengan demikian, normalisasi data membantu meningkatkan kualitas hasil analisis dan kinerja dari tiga algoritma pembelajar mesin yang digunakan.

4.3 Processing

Dataset yang telah melalui praprocessing diproses dibagi menjadi fitur (X) dan label (y). Selanjutnya, data tersebut dipisahkan menjadi data latih 80% dan data uji 20%. Data ini kemudian digunakan untuk melatih dan menguji model. Pada Tabel 2, menjelaskan parameter-parameter yang digunakan oleh ketiga model yakni model Decision Tree, Random Forest, dan XGBoost sesuai dengan metrix evaluasi yang telah ditentukan.



Gambar 8 Parameter Decision Tree (A), Random Forest (B), dan XGBoost (C)

4.4 Evaluasi

Setelah melalui serangkaian tahapan-tahapan penting dalam preprocessing dan processing, langkah selanjutnya adalah evaluasi model. Pada tahap ini, peneliti melanjutkan dengan menggunakan confusion matrix sebagai instrumen untuk melakukan evaluasi mendalam terhadap performa model yang telah dilatih.

Tabel 2 *Clasification Report*

Algoritma	Clasification Report
Decision Tree	Classification Report (Decision Tree): precision recall f1-score support 0 0.84 0.89 0.86 107 1 0.61 0.51 0.56 37 accuracy 0.79 144 macro avg 0.73 0.70 0.71 144 weighted avg 0.78 0.79 0.79 144
Random Forest	Classification Report (Random Forest): precision recall f1-score support 0 0.86 0.83 0.84 107 1 0.55 0.59 0.57 37 accuracy 0.77 144 macro avg 0.70 0.71 0.71 144 weighted avg 0.78 0.77 0.77 144

	Classification Report (XGBoost):				
		precision	recall	f1-score	support
XGBoost	0	0.88	0.79	0.83	107
	1	0.53	0.68	0.60	37
	accuracy			0.76	144
	macro avg	0.70	0.74	0.71	144
	weighted avg	0.79	0.76	0.77	144

Berdasarkan tabel 2, Berikut pemaparan hasilnya:

A. Decision Tree:
   1. Akurasi Keseluruhan : 79%
   2. Rata-rata (macro avg) : Presisi 0.73, Recall 0.70, F1-Score 0.71
   3. Rata-rata (weighted avg) : Presisi 0.78, Recall 0.79, F1-Score 0.79

B. Random Forest:
   1. Akurasi Keseluruhan : 77%
   2. Rata-rata (macro avg) : Presisi 0.70, Recall 0.71, F1-Score 0.71
   3. Rata-rata (weighted avg) : Presisi 0.78, Recall 0.77, F1-Score 0.77

C. XGBoost:
   1. Akurasi Keseluruhan : 76%
   2. Rata-rata (macro avg) :Presisi 0.70, Recall 0.74, F1-Score 0.71
   3. Rata-rata berbobot (weighted avg) :Presisi 0.79, Recall 0.76, F1-Score 0.77

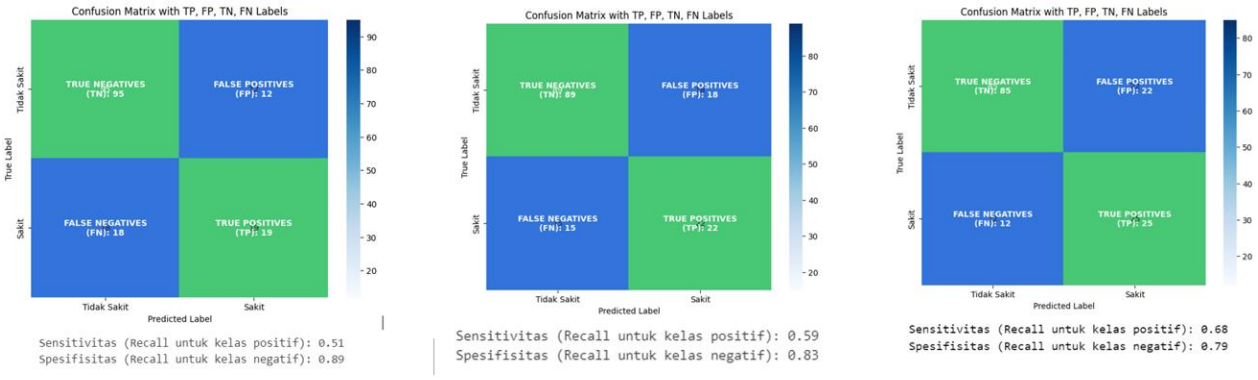
Berdasarkan gambar yang dipaparkan menunjukan seberapa baik kinerja ketiga model dalam melakukan klasifikasi, dengan mempertimbangkan precision, recal, dan F1-Score dari masing-masing model. Berikut analisis lebih dalam dari hasil yang diperoleh:

A. Decision Tree :
   1. Akurasi Keseluruhan: Model Decision Tree memiliki akurasi keseluruhan sebesar 79%. Akurasi mengukur seberapa banyak prediksi model yang benar dari seluruh instance dalam data uji.
   2. Rata-rata (macro avg):
      a) Presisi: 0.73, yang berarti secara rata-rata tanpa memperhitungkan proporsi kelas, 73% prediksi positif adalah benar.
      b) Recall: 0.70, yang berarti secara rata-rata tanpa bobot, model mampu mendeteksi 70% dari semua instance yang benar-benar positif.
      c) F1-Score: 0.71, yang merupakan rata-rata harmonik dari presisi dan recall. F1-score memberikan keseimbangan antara kedua metrik tersebut.
   3. Rata-rata Berbobot (weighted avg):
      a) Presisi: 0.78, yang menunjukkan presisi secara rata-rata dengan mempertimbangkan proporsi setiap kelas dalam data.
      b) Recall: 0.79, yang menunjukkan recall secara rata-rata dengan mempertimbangkan proporsi setiap kelas dalam data.
      c) F1-Score: 0.79, yang menunjukkan F1-score secara rata-rata dengan mempertimbangkan proporsi setiap kelas dalam data.

B. Random Forest:
   1. Akurasi Keseluruhan: Model Random Forest memiliki akurasi keseluruhan sebesar 77%. Akurasi mengukur seberapa banyak prediksi model yang benar dari seluruh instance dalam data uji.
   2. Rata-rata (macro avg):
      a) Presisi: 0.70, yang berarti secara rata-rata tanpa memperhitungkan proporsi kelas, 70% prediksi positif adalah benar.
      b) Recall: 0.71, yang berarti secara rata-rata tanpa bobot, model mampu mendeteksi 71% dari semua instance yang benar-benar positif.
      c) F1-Score: 0.71, yang merupakan rata-rata harmonik dari presisi dan recall. F1-score memberikan keseimbangan antara kedua metrik tersebut.
   3. Rata-rata Berbobot (weighted avg):
      a) Presisi: 0.78, yang menunjukkan presisi secara rata-rata dengan mempertimbangkan proporsi setiap kelas dalam data.
      b) Recall: 0.77, yang menunjukkan recall secara rata-rata dengan mempertimbangkan proporsi setiap kelas dalam data.
      c) F1-Score: 0.77, yang menunjukkan F1-score secara rata-rata dengan mempertimbangkan proporsi setiap kelas dalam data.

C. XGBoost:
   1. Keseluruhan Akurasi: Model XGBoost memiliki akurasi keseluruhan sebesar 76%. Akurasi mengukur seberapa banyak prediksi model yang benar dari seluruh instance dalam data uji.
   2. Rata-rata (macro avg):
      a) Presisi: 0.70, yang berarti secara rata-rata tanpa memperhitungkan proporsi kelas, 70% prediksi positif adalah benar.
      b) Recall: 0.74, yang berarti secara rata-rata tanpa bobot, model mampu mendeteksi 74% dari semua instance yang benar-benar positif.
      c) F1-Score: 0.71, yang merupakan rata-rata harmonik dari presisi dan recall. F1-score memberikan keseimbangan antara kedua metrik tersebut.
   3. Rata-rata Berbobot (weighted avg):
      a) Presisi: 0.79, yang menunjukkan presisi secara rata-rata dengan mempertimbangkan proporsi setiap kelas dalam data.
      b) Recall: 0.76, yang menunjukkan recall secara rata-rata dengan mempertimbangkan proporsi setiap kelas dalam data.
      c) F1-Score: 0.77, yang menunjukkan F1-score secara rata-rata dengan mempertimbangkan proporsi setiap kelas dalam data.

Selain itu, terdapat juga evaluasi sensitivity dan specivity dengan menggunakan cofusion matriks. Proses klasifikasi mencangkup 4 elemen utama yaitu True Positives (TP), False Positives (FP), True Negatives (TN), dan False Negatives (FN).



A. Decision Tree B. Random Forest C. XGBoost

Gambar 9 Nilai Sensitivity dan Specivicity dari masing-masing metode

Berikut analisis dari gambar di atas:

A. Decision Tree
   1. True Positives (TP) : 19
   2. False Positives (FP) : 12
   3. True Negatives (TN) : 95
   4. False Negative (FN) : 18
   5. Nilai Sensitivity : 0.51, yang menunjukan bahwa kemampuan model dalam mendeteksi kelas “Sakit” cukup rendah.
   6. Nilai Specivicity : 0.89, menunjukan bahwa kemampuan model dalam mendeteksi kelas “Tidak Sakit” cukup baik

B. Random Forest
   1. True Positives (TP) : 22
   2. False Positives (FP) : 18
   3. True Negatives (TN) : 89
   4. False Negative (FN) : 15
   5. Nilai Sensitivity : 0.59, yang menunjukan bahwa kemampuan model dalam mendeteksi kelas “Sakit” cukup rendah.
   6. Nilai Specivicity : 0.83, menunjukan bahwa kemampuan model dalam mendeteksi kelas “Tidak Sakit” cukup baik

C. XGBoost
   1. True Positives (TP) : 25
   2. False Positives (FP) : 22
   3. True Negatives (TN) : 85
   4. False Negative (FN) : 12
   5. Nilai Sensitivity : 0.68, yang menunjukan bahwa kemampuan model dalam mendeteksi kelas “Sakit” cukup baik.
   6. Nilai Specivicity : 0.79, menunjukan bahwa kemampuan model dalam mendeteksi kelas “Tidak Sakit” cukup baik

5 Kesimpulan

Penelitian ini mengevaluasi kinerja algoritma Decision Tree, Random Forest, dan XGBoost dalam memprediksi penyakit diabetes menggunakan dataset Pima Indians Diabetes Dataset. Proses pengolahan data melibatkan penanganan missing values, outlier, normalisasi dan pembagian data untuk melatih data dan menguji model. Berdasarkan analisis yang dilakukan, ditemukan bahwa:

1. Decision Tree
   Memberikan keunggulan dalam interpretasi hasil karena sifatnya yang mudah dipahami. Namun, akurasinya lebih rendah dibandingkan algoritma lainnya, terutama pada data dengan kompleksitas tinggi.
2. Random Forest
   Memperlihatkan kemampuan generalisasi yang lebih baik dengan pendekatan ansambel. Akurasi lebih tinggi dibandingkan Decision Tree, namun membutuhkan waktu komputasi lebih lama.
3. XGBoost
   Menunjukkan performa terbaik secara keseluruhan dalam hal akurasi dan efisiensi. Kemampuan menangani data tidak seimbang dan fitur kompleks menjadikannya algoritma unggulan dalam penelitian ini.

Penelitian ini memberikan kontribusi signifikan dalam pemanfaatan teknologi machine learning untuk mendukung deteksi dini penyakit diabetes. Kedepannya, penelitian ini dapat diperluas dengan menggunakan dataset yang lebih besar dan beragam serta mengintegrasikan metode optimasi untuk menigkatkan performa model.

Ucapan Terima Kasih

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada Rizal Adi Saputri, S.T., M.Kom selaku dosen pembimbing yang telah memberikan tugas penelitian ini sebagai bagian dari pembelajaran di mata kuliah yang beliau ajarkan. Dukungan, arahan, serta motivasi beliau sangat membantu dalam menyelesaikan penelitian ini.

Ucapan terima kasih juga disampaikan kepada Universitas Halu Oleo, khususnya Jurusan Teknik Informatika, yang telah memberikan fasilitas dan dukungan selama penelitian berlangsung. Penulis juga menghargai kontribusi rekan sejawat yang memberikan masukan berharga sepanjang proses penelitian ini.

Daftar Pustaka

[1] N. S. Jasmine, S. Wahyuningsih, and M. S. Thadeus, “Analisis faktor tingkat kepatuhan minum obat pasien diabetes melitus di Puskesmas Pancoran Mas periode Maret-April 2019,” *J. Manaj. Kesehat. Indones.*, vol. 8, no. 1, pp. 61–66, 2020.

[2] M. Hayati, Z. Hamzah, and A. Tri, “Hubungan Kadar Insulin Pankreas dan Kadar Glukosa Darah pada Model Tikus Wistar Jantan setelah Diinduksi Bisphenol-A (The Relation of Insulin Pancreas Levels and Blood Glucose Levels to Wistar Rat Models After Bisphenol-A Induction),” *Stomatognatic (J.K.G Unej)* , vol. 17, no. 1, pp. 4–7, 2020.

[3] I. Binanto, N. F. Sianipar, N. M. D. Aprilianti, J. Gein, and P. D. Paska, “Analisis Perbandingan Algoritma Knn, Gaussian Naive Bayes, Random Forest Untuk Data Tidak Seimbang Dan Data Yang Diseimbangkan Dengan Metode Tomek Link Undersampling Pada Dataset Lcms Tanaman Keladi Tikus,” *Pros. Sains Nas. dan Teknol.*, vol. 13, no. 1, p. 156,

2023, doi: 10.36499/psnst.v13i1.9002.
[4] A. Fauzi and A. H. Yunial, "Optimasi Algoritma Klasifikasi Naive Bayes, Decision Tree, K – Nearest Neighbor, dan Random Forest menggunakan Algoritma Particle Swarm Optimization pada Diabetes Dataset," *J. Edukasi dan Penelit. Inform.*, vol. 8, no. 3, p. 470, 2022, doi: 10.26418/jp.v8i3.56656.
[5] L. M. Cendani and A. Wibowo, "Perbandingan Metode Ensemble Learning pada Klasifikasi Penyakit Diabetes," *J. Masy. Inform.*, vol. 13, no. 1, pp. 33–44, 2022, doi: 10.14710/jmasif.13.1.42912.
[6] Y. Yusuf, "Perbandingan Performansi Algoritma Decision Tree C5 . 0 , Cart ," *Seminar*, vol. 2007, no. Snati, pp. 0–3, 2007.
[7] M. Salsabil, N. Lutvi, and A. Eviyanti, "Implementasi Data Mining Dalam Melakukan Prediksi Penyakit Diabetes Menggunakan Metode Random Forest Dan Xgboost," *J. Ilm. Komputasi*, vol. 23, no. 1, pp. 51–58, 2024, doi: 10.32409/jikstik.23.1.3507.
[8] R. Supriyadi, W. Gata, N. Maulidah, and A. Fauzi, "Penerapan Algoritma Random Forest Untuk Menentukan Kualitas Anggur Merah," *E-Bisnis J. Ilm. Ekon. dan Bisnis*, vol. 13, no. 2, pp. 67–75, 2020, doi: 10.51903/e-bisnis.v13i2.247.
[9] R. G. Gunawan, Erik Suanda Handika, and Edi Ismanto, "Pendekatan Machine Learning Dengan Menggunakan Algoritma Xgboost (Extreme Gradient Boosting) Untuk Peningkatan Kinerja Klasifikasi Serangan Syn," *J. CoSciTech (Computer Sci. Inf. Technol.*, vol. 3, no. 3, pp. 453–463, 2022, doi: 10.37859/coscitech.v3i3.4356.
[10] M. S. Mustafa, M. R. Ramadhan, and A. P. Thenata, "Implementasi Data Mining untuk Evaluasi Kinerja Akademik Mahasiswa Menggunakan Algoritma Naive Bayes Classifier," *Creat. Inf. Technol. J.*, vol. 4, no. 2, p. 151, 2018, doi: 10.24076/citec.2017v4i2.106.
[11] A. Afifuddin and L. Hakim, "Deteksi Penyakit Diabetes Mellitus Menggunakan Algoritma Decision Tree Model Arsitektur C4.5," *J. Krisnadana*, vol. 3, no. 1, pp. 25–33, 2023, doi: 10.58982/krisnadana.v3i1.470.
[12] U. I. Lestari, "Penerapan Metode K-Nearest Neighbor Untuk Sistem Pendukung Keputusan Identifikasi Penyakit Diabetes Melitus," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 8, no. 4, pp. 2071–2082, 2021, doi: 10.35957/jatisi.v8i4.1235.
[13] R. A. Ramadhani and R. K. Niswatin, "Sistem Diagnosa Diabetes Menggunakan Metode K-NN," *J. Sains dan Inform.*, vol. 4, no. 2, pp. 98–104, 2018, doi: 10.34128/jsi.v4i2.121.
[14] N. Maulidah, R. Supriyadi, D. Y. Utami, F. N. Hasan, A. Fauzi, and A. Christian, "Prediksi Penyakit Diabetes Melitus Menggunakan Metode Support Vector Machine dan Naive Bayes," *Indones. J. Softw. Eng.*, vol. 7, no. 1, pp. 63–68, 2021, doi: 10.31294/ijse.v7i1.10279.
[15] G. A. B. Suryanegara, Adiwijaya, and M. D. Purbolaksono, "Peningkatan Hasil Klasifikasi pada Algoritma Random Forest untuk Deteksi," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 1, no. 10, pp. 114–122, 2021.
[16] S. Susanti, "Penerapan Data Mining Analisa Penyakit," 2021.
[17] Yeni Melia and Rini Deswita, "Analisis Predeksi Kebangkrutan dengan MenggunakanMetode Altman Z-Score," *J. Akunt. Keuang. dan Bisnis*, vol. 13, no. 1, pp. 71–80, 2020, [Online]. Available: www.idx.co.id
[18] K. Aditya, A. Wisnu, and A. M. A. Rahim, "Analisis Perbandingan Algoritma XGBoost Dan Algoritma Random Forest Untuk Klasifikasi Data Kesehatan Mental," vol. 2, no. 5, pp. 808–818, 2024.
[19] A. Ilham, "Hybrid Metode Boostrap Dan Teknik Imputasi Pada Metode C4-5 Untuk Prediksi Penyakit Ginjal Kronis," *Statistika*, vol. 8, no. 1, pp. 43–51, 2020.