EISSN 3089-2724
Vol. 2, No. 2, July 2026
https://librarium.id/index.php/synctech
SYNCTECH System Information and Computer Technology

Klasifikasi Diabetes Menggunakan Algoritma K-Nearest Neighbor dengan Preprocessing dan Min-Max Scaling pada Pima Indians Diabetes Dataset

Nurdilla [1*], Roberto Kaban [2]
[1,2] Jurusan Rekayasa Perangkat Lunak, Institut Teknologi dan Bisnis Indonesia, Indonesia
[1] nurdilla331@gmail.com, [2] roberto.kaban@yahoo.com

Diajukan: 17/06/2026 | Direvisi: 02/07/2026 | Diterima: 06/07/2026 | Diterbitkan: 31/07/2026

Abstract

Diabetes mellitus is a chronic metabolic disease characterized by high blood glucose levels, which can lead to various complications if not detected early. The use of machine learning technology is increasingly prevalent in the healthcare sector due to its ability to assist in disease analysis and classification based on patient data. This study aims to apply the K-Nearest Neighbor (KNN) algorithm to classify diabetes using the Pima Indians Diabetes Dataset. The dataset comprises 768 patient records, featuring eight predictor attributes and one target attribute. The research methodology involves several stages: data preprocessing (replacing invalid values with the median of the respective attributes), data normalization using the Min-Max Scaling method, splitting the dataset into training and testing sets at an 80:20 ratio, and implementing the KNN algorithm with K=5. Model performance was evaluated using a confusion matrix, yielding metrics for Accuracy, Precision, Recall, and F1-Score. The results show that the model achieved an Accuracy of 74.68%, Precision of 66.00%, Recall of 60.00%, and an F1-Score of 62.86%. These findings demonstrate that the combination of preprocessing, Min-Max Scaling normalization, and the KNN algorithm yields effective performance in classifying diabetes data based on patient health characteristics. This study is expected to serve as a reference for future research on developing machine learning applications to support more effective diabetes identification.

Keywords**:** *Classification, Diabetes Mellitus, K-Nearest Neighbor, Machine Learning, Python.*

Abstrak

Diabetes melitus merupakan penyakit metabolik kronis yang ditandai dengan tingginya kadar glukosa dalam darah dan berpotensi menyebabkan berbagai komplikasi apabila tidak dideteksi sejak dini. Pemanfaatan teknologi machine learning semakin berkembang dalam bidang kesehatan karena mampu membantu proses analisis dan klasifikasi penyakit berdasarkan data pasien. Tujuan dari penelitian ini yaitu menerapkan algoritma K-Nearest Neighbor (KNN) untuk mengklasifikasikan penyakit diabetes menggunakan Pima Indians Diabetes Dataset. Dataset yang digunakan terdiri atas 768 data pasien dengan 8 atribut prediktor dan 1atribut target. Tahapan penelitian meliputi proses preprocessing dengan mengganti nilai yang tidak valid menggunakan median pada atribut terkait, normalisasi data menggunakan metode Min-Max Scaling, pembagian dataset menjadi data training dan data testing dengan perbandingan 80:20, serta penerapan algoritma KNN menggunakan nilai K = 5. Evaluasi performa model dilakukan menggunakan Confusion Matrix yang menghasilkan nilai Accuracy, Precision, Recall, dan F1-Score. Berdasarkan hasil pengujian, model memperoleh Accuracy sebesar 74,68%, Precision sebesar 66,00%, Recall sebesar 60,00%, dan F1-Score sebesar 62,86%. Temuan penelitian menunjukkan bahwa kombinasi tahapan preprocessing, normalisasi Min-Max Scaling, dan algoritma KNN mampu memberikan performa yang cukup baik dalam mengklasifikasikan data diabetes berdasarkan karakteristik kesehatan pasien. Penelitian ini diharapkan dapat memberikan kontribusi sebagai referensi bagi penelitian selanjutnya dalam pengembangan aplikasi machine learning untuk mendukung proses identifikasi diabetes secara lebih efektif.

Kata Kunci: Diabetes Melitus, Klasifikasi, K-Nearest Neighbor, Machine Learning, Python.

1. PENDAHULUAN

Salah satu penyakit kronis yang banyak menjadi perhatian dalam bidang kesehatan adalah diabetes melitus. Penyakit ini ditandai dengan meningkatnya kadar glukosa dalam darah yang disebabkan oleh ketidakmampuan tubuh memproduksi insulin dalam jumlah yang cukup, berkurangnya efektivitas kerja insulin, atau kombinasi keduanya. Jumlah penderita diabetes terus mengalami peningkatan di berbagai negara sehingga penyakit ini menjadi salah satu masalah kesehatan yang mendapatkan perhatian serius [1]. Kondisi diabetes yang tidak ditangani secara tepat dapat menimbulkan berbagai komplikasi, seperti penyakit kardiovaskular, gangguan fungsi ginjal, kerusakan saraf, hingga stroke. Oleh karena itu, upaya identifikasi dini terhadap risiko diabetes sangat diperlukan untuk mendukung penanganan pasien secara lebih efektif serta mengurangi kemungkinan terjadinya komplikasi lanjutan [2].

Kemajuan teknologi di bidang kecerdasan buatan telah mendorong pemanfaatan machine learning dalam berbagai sektor, termasuk layanan kesehatan. Metode machine learning memungkinkan proses pengolahan dan analisis data medis dilakukan secara otomatis sehingga dapat membantu tenaga kesehatan dalam memperoleh informasi yang mendukung proses pengambilan keputusan [3]. Berbagai algoritma machine learning telah diterapkan untuk menyelesaikan permasalahan klasifikasi penyakit, salah satunya adalah algoritma *K-Nearest Neighbor* (*KNN*) [4].

K-Nearest Neighbor merupakan metode klasifikasi berbasis kedekatan jarak yang menentukan kategori suatu data berdasarkan sejumlah tetangga terdekat yang telah diketahui kelasnya. Algoritma ini banyak digunakan karena memiliki konsep yang sederhana, mudah diimplementasikan, dan mampu menghasilkan performa yang kompetitif pada berbagai kasus klasifikasi. Dalam bidang kesehatan, KNN telah diterapkan pada sejumlah penelitian untuk mengidentifikasi berbagai jenis penyakit berdasarkan karakteristik data pasien [5].

Beberapa penelitian terdahulu menunjukkan bahwa algoritma KNN memiliki kemampuan yang cukup baik dalam melakukan klasifikasi penyakit diabetes. Berdasarkan hasil penelitian yang dilaporkan dalam [6] metode KNN mampu menghasilkan tingkat klasifikasi yang memadai berdasarkan atribut kesehatan pasien. Temuan tersebut menunjukkan bahwa pendekatan KNN dapat digunakan sebagai salah satu alternatif dalam mendukung proses identifikasi penyakit diabetes.

Penelitian lain yang dilakukan oleh [7] mengungkapkan bahwa performa KNN dipengaruhi oleh beberapa faktor, terutama pemilihan nilai K yang digunakan selama proses klasifikasi. Penentuan nilai K yang sesuai dapat meningkatkan kemampuan model dalam mengenali pola data sehingga menghasilkan tingkat akurasi yang lebih baik. Penelitian tersebut juga menekankan pentingnya tahapan preprocessing dan normalisasi data dalam meningkatkan kualitas hasil klasifikasi.

Selain menggunakan algoritma KNN, beberapa penelitian klasifikasi diabetes juga memanfaatkan Pima Indians Diabetes Dataset sebagai sumber data penelitian. Menurut [8], dataset tersebut merupakan salah satu dataset yang paling banyak digunakan dalam penelitian *machine learning* untuk klasifikasi diabetes karena memiliki atribut kesehatan yang relevan serta sering digunakan dalam evaluasi model klasifikasi. Atribut seperti kadar glukosa darah, tekanan darah, indeks massa tubuh (BMI), usia, dan riwayat kesehatan keluarga dinilai memiliki kontribusi penting dalam proses identifikasi diabetes.

Walaupun memiliki beberapa kelebihan, algoritma KNN juga mempunyai keterbatasan. Salah satu faktor yang sering memengaruhi hasil klasifikasi adalah sensitivitas terhadap pemilihan nilai K. Penggunaan nilai K yang terlalu rendah dapat membuat model lebih sensitif terhadap variasi data atau *noise*, sementara nilai K yang terlalu tinggi dapat menyebabkan batas antar kelas menjadi kurang jelas sehingga kemampuan model dalam melakukan klasifikasi dapat menurun. Selain itu, kualitas data serta proses preprocessing yang diterapkan juga berpengaruh terhadap performa model yang dihasilkan [6].

Meskipun algoritma K-Nearest Neighbor telah banyak diterapkan pada klasifikasi diabetes, hasil yang diperoleh pada berbagai penelitian masih menunjukkan variasi performa. Perbedaan tersebut dipengaruhi oleh kualitas data, teknik preprocessing, metode normalisasi, serta strategi pembagian data yang digunakan sebelum proses klasifikasi dilakukan. Selain itu, beberapa penelitian masih menunjukkan nilai Recall yang belum optimal sehingga terdapat kasus diabetes yang belum berhasil dikenali secara akurat. Berdasarkan kondisi tersebut, penelitian ini menerapkan proses preprocessing melalui penggantian nilai yang tidak valid menggunakan median, normalisasi dengan metode Min-Max Scaling, serta penerapan algoritma K-Nearest Neighbor menggunakan nilai K = 5 pada Pima Indians Diabetes Dataset untuk mengevaluasi kinerja model menggunakan metrik Accuracy, Precision, Recall, dan F1-Score.

Dataset yang digunakan diperoleh dari Kaggle dan terdiri atas berbagai atribut kesehatan pasien yang relevan dengan proses identifikasi diabetes. Data tersebut dipilih karena telah banyak digunakan dalam penelitian sebelumnya sehingga memungkinkan hasil penelitian untuk dibandingkan dengan studi terkait [8].

Tujuan penelitian ini adalah menerapkan algoritma K-Nearest Neighbor dalam proses klasifikasi penyakit diabetes serta mengevaluasi performa model menggunakan metrik *Accuracy, Precision, Recall*, dan F1-Score. Hasil penelitian diharapkan dapat memberikan gambaran mengenai kemampuan algoritma

KNN dalam mengidentifikasi penyakit diabetes berdasarkan data kesehatan pasien serta menjadi referensi bagi penelitian selanjutnya di bidang machine learning dan kesehatan [9].

2. METODE PENELITIAN/ALGORITMA

2.1. Dataset Penelitian

Sebagai sumber data penelitian, digunakan Pima Indians Diabetes Dataset yang diperoleh dari repositori Kaggle [10]. Dataset tersebut merupakan salah satu dataset terbuka yang sering digunakan dalam penelitian machine learning, khususnya untuk klasifikasi penyakit diabetes, karena menyediakan atribut kesehatan yang relevan dan telah banyak dijadikan acuan dalam berbagai penelitian sejenis [8].

Data yang digunakan dalam penelitian ini berisi 768 rekam data pasien dengan delapan atribut sebagai variabel masukan dan satu atribut sebagai variabel keluaran (Outcome). Atribut prediktor meliputi Pregnancies, Glucose, BloodPressure, SkinThickness, Insulin, BMI, DiabetesPedigreeFunction, dan Age. Adapun variabel Outcome digunakan sebagai label klasifikasi untuk menentukan apakah seorang pasien termasuk kategori diabetes atau tidak diabetes.

Tabel 1. Deskripsi Atribut Pima Indians Diabetes Dataset

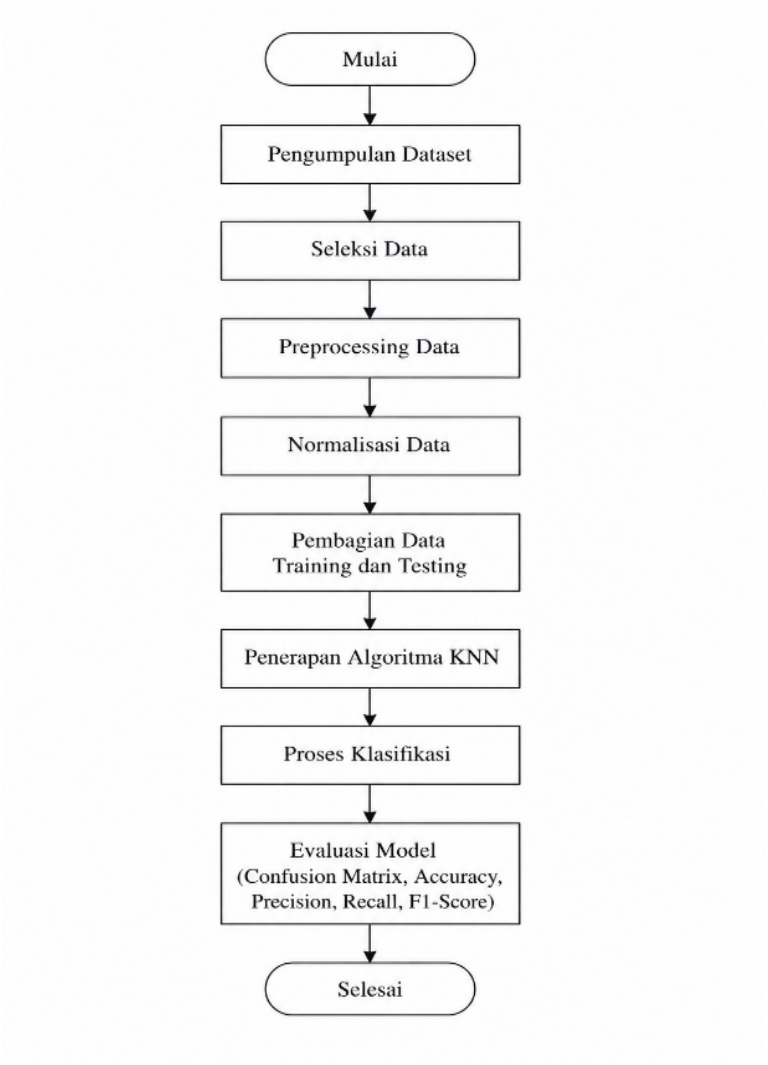
No	Atribut	Keterangan
1	Pregnancies	Jumlah kehamilan
2	Glucose	Kadar glukosa darah
3	BloodPressure	Tekanan darah diastolik
4	SkinThickness	Ketebalan lipatan kulit trisep
5	Insulin	Kadar insulin serum
6	BMI	Body Mass Index
7	DiabetesPedigreeFunction	Faktor riwayat keturunan diabetes
8	Age	Usia pasien
9	Outcome	Status diabetes (0 = tidak diabetes, 1 = diabetes)

Berdasarkan Tabel 1, dataset terdiri atas delapan atribut prediktor dan satu atribut target yang digunakan sebagai label klasifikasi. Atribut Outcome digunakan untuk menentukan apakah pasien termasuk kategori diabetes atau tidak diabetes.

Pemilihan dataset ini dilakukan karena dataset Pima Indians Diabetes memiliki karakteristik data kesehatan yang lengkap sehingga dapat digunakan untuk proses pelatihan dan pengujian model klasifikasi penyakit diabetes. Selain itu, dataset ini juga banyak digunakan pada penelitian machine learning di bidang kesehatan sehingga memudahkan proses evaluasi dan perbandingan hasil penelitian [2].

2.2. Tahapan Penelitian

Penelitian ini dilaksanakan melalui serangkaian tahapan yang terstruktur dan saling berkaitan untuk membangun model klasifikasi penyakit diabetes menggunakan algoritma *K-Nearest Neighbor* (KNN). Rangkaian tahapan penelitian tersebut ditunjukkan pada gambar 1.



Gambar 1. Diagram Alur Penelitian

Berdasarkan Gambar 1, penelitian diawali dengan pengumpulan dataset Pima Indians Diabetes yang diperoleh dari Kaggle. Setelah data diperoleh, dilakukan seleksi data untuk memastikan atribut yang digunakan sesuai dengan kebutuhan penelitian, yaitu atribut yang berkaitan dengan kondisi kesehatan pasien dan status diabetes.

Tahap berikutnya adalah preprocessing data yang bertujuan meningkatkan kualitas dataset sebelum digunakan dalam proses klasifikasi. Pada tahap ini dilakukan pemeriksaan data, penanganan nilai yang tidak valid, serta verifikasi kelengkapan data. Proses preprocessing diperlukan agar kualitas data menjadi lebih baik sehingga model machine learning dapat menghasilkan performa klasifikasi yang lebih optimal [11].

Setelah preprocessing selesai dilakukan, tahap selanjutnya adalah normalisasi data. Normalisasi bertujuan menyeragamkan rentang nilai setiap atribut sehingga tidak terjadi dominasi atribut tertentu dalam proses perhitungan jarak pada algoritma KNN. Langkah ini penting karena KNN merupakan metode yang bergantung pada perhitungan jarak, sehingga perbedaan skala antar atribut dapat memengaruhi hasil klasifikasi yang diperoleh [12].

Dataset yang telah melalui tahap normalisasi kemudian dialokasikan ke dalam dua kelompok data, yaitu data training sebesar 80% dan data testing sebesar 20% menggunakan metode train-test split. Pembagian data dilakukan menggunakan random_state = 42 agar proses pemisahan data dapat direplikasi

pada penelitian selanjutnya. Pada tahap ini data training digunakan sebagai dasar pembelajaran model, sedangkan data testing digunakan untuk mengevaluasi kemampuan model terhadap data yang belum pernah dipelajari sebelumnya.

Tahap berikutnya adalah penerapan algoritma *K-Nearest Neighbor* (*KNN*). Pada proses ini, model melakukan klasifikasi dengan menghitung jarak antara data uji dan data latih menggunakan metode *Euclidean Distance*. Selanjutnya, penentuan kelas dilakukan berdasarkan mayoritas tetangga terdekat yang dimiliki oleh data tersebut [6].

Hasil klasifikasi yang diperoleh kemudian dievaluasi menggunakan *Confusion Matrix* untuk mengetahui jumlah prediksi yang benar maupun salah. Berdasarkan *Confusion Matrix* tersebut dihitung nilai Accuracy, Precision, Recall, dan F1-Score untuk mengukur performa model klasifikasi yang dihasilkan [13].

2.3. Algoritma *K-Nearest Neighbor* (*KNN*)

Algoritma K-Nearest Neighbor (*KNN*) merupakan salah satu teknik klasifikasi yang menentukan kategori suatu data berdasarkan tingkat kedekatannya dengan data lain. Proses klasifikasi dilakukan dengan mencari sejumlah data yang memiliki jarak paling dekat terhadap data yang akan diprediksi, kemudian kelas data tersebut ditentukan berdasarkan kelas yang paling banyak muncul pada tetangga terdekatnya. Menurut [7], algoritma KNN mampu melakukan klasifikasi data dengan cara membandingkan karakteristik data baru terhadap data yang telah diketahui kelasnya sehingga dapat menghasilkan prediksi berdasarkan tingkat kedekatan antar data.

KNN termasuk metode *supervised learning* yang cukup sederhana karena tidak memerlukan proses pembentukan model yang kompleks. Meskipun demikian, performa algoritma KNN sangat dipengaruhi oleh pemilihan nilai K serta karakteristik data yang digunakan dalam proses klasifikasi.

Pada penelitian ini digunakan nilai K = 5 sebagai jumlah tetangga terdekat yang dijadikan dasar dalam menentukan kelas suatu data. Pemilihan satu konfigurasi nilai K dilakukan sesuai dengan ruang lingkup penelitian yang berfokus pada implementasi dan evaluasi performa algoritma K-Nearest Neighbor tanpa melakukan proses optimasi hyperparameter. Oleh karena itu, hasil yang diperoleh merepresentasikan kinerja model berdasarkan konfigurasi yang digunakan dan dapat menjadi dasar bagi penelitian selanjutnya untuk mengevaluasi pengaruh variasi nilai K melalui proses optimasi hyperparameter. Untuk menghitung kedekatan antar data, algoritma KNN menggunakan metode *Euclidean Distance*. Metode tersebut digunakan untuk mengukur jarak antara data testing dan data training berdasarkan nilai atribut yang dimiliki masing-masing data. Rumus *Euclidean Distance* yang digunakan ditunjukkan pada Persamaan (1).

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \qquad (1)$$

Keterangan:

$d(x,y)$= jarak antar data

x_i= data training

y_i= data testing

n=jumlah atribut

Semakin kecil nilai jarak yang diperoleh, maka semakin tinggi tingkat kemiripan antara data uji dan data latih. Setelah seluruh jarak dihitung, penentuan kelas dilakukan berdasarkan kelas yang paling dominan dari sejumlah tetangga terdekat yang diperoleh dari hasil perhitungan tersebut [9].

2.4. Implementasi Menggunakan *Python*

Pada penelitian ini, seluruh tahapan pengolahan data hingga proses pembangunan model klasifikasi dikerjakan dengan memanfaatkan bahasa pemrograman *Python*. *Library* yang dimanfaatkan meliputi Pandas untuk mengelola dan memproses dataset, serta Scikit-learn untuk melakukan normalisasi data, pembagian dataset menjadi data training dan testing, penerapan algoritma *K-Nearest Neighbor* (KNN), dan evaluasi hasil klasifikasi. Penggunaan *Python* dipilih karena menyediakan berbagai pustaka yang mendukung proses pengolahan data, machine learning, dan evaluasi model secara efisien.

2.5. Evaluasi Model

Setelah proses klasifikasi selesai dilakukan, tahap berikutnya adalah mengevaluasi hasil yang diperoleh untuk mengetahui tingkat efektivitas algoritma *K-Nearest Neighbor* (*KNN*). Evaluasi dilakukan menggunakan Confusion Matrix yang menggambarkan hubungan antara hasil prediksi model dan kondisi aktual data melalui empat kategori, yaitu True Positive (TP), True Negative (TN), False Positive (FP), dan False Negative (FN). Keempat nilai tersebut kemudian dimanfaatkan untuk menghitung *Accuracy, Precision, Recall, dan F1-Score* sebagai indikator kinerja model klasifikasi [3].

Accuracy

Accuracy menunjukkan tingkat ketepatan model dengan membandingkan jumlah prediksi yang benar terhadap keseluruhan data yang diuji [3].

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (2)$$

Precision

Precision menggambarkan kemampuan model dalam memberikan prediksi positif yang tepat. Nilai ini menunjukkan seberapa banyak data yang diprediksi positif benar-benar termasuk ke dalam kelas positif [3].

$$\text{Precision} = \frac{TP}{TP+FP} \quad (3)$$

Recall

Recall digunakan untuk menilai kemampuan model dalam mengenali seluruh data yang sebenarnya berada pada kelas positif. Semakin tinggi nilai *Recall*, semakin banyak data positif yang berhasil teridentifikasi oleh model [3].

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

F1-Score

F1-Score merupakan metrik yang mengombinasikan nilai *Precision* dan *Recall* ke dalam satu ukuran evaluasi. Nilai ini digunakan untuk memberikan gambaran yang lebih seimbang mengenai performa model klasifikasi, terutama ketika distribusi kelas tidak seimbang [3].

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (5)$$

Keterangan:

TP = True Positive

TN = True Negative

FP = False Positive

FN = False Negative

Nilai *Accuracy, Precision, Recall,* dan *F1-Score* yang semakin tinggi menunjukkan bahwa model memiliki kemampuan yang semakin baik dalam melakukan klasifikasi data diabetes secara tepat dan konsisten.

3. HASIL PENELITIAN DAN PEMBAHASAN

3.1. Hasil Preprocessing Data

Sebelum diterapkan metode *K-Nearest Neighbor* (*KNN*), dataset terlebih dahulu melalui tahap preprocessing untuk meningkatkan kualitas data. Tahap ini dilakukan dengan memeriksa keberadaan nilai yang tidak sesuai pada beberapa atribut kesehatan yang digunakan dalam penelitian. *Preprocessing* merupakan tahapan penting dalam machine learning karena kualitas data yang baik dapat meningkatkan performa model klasifikasi yang dihasilkan [14].

Hasil pemeriksaan menunjukkan bahwa terdapat beberapa atribut yang memiliki nilai 0, yaitu atribut Glucose sebanyak 5 data, BloodPressure sebanyak 35 data, SkinThickness sebanyak 227 data, Insulin sebanyak 374 data, dan BMI sebanyak 11 data. Nilai 0 pada atribut tersebut dianggap tidak merepresentasikan kondisi medis yang sebenarnya sehingga dilakukan penggantian menggunakan nilai median masing-masing atribut.

Tabel 2. Jumlah Nilai 0 Sebelum Preprocessing

Atribut	Jumlah Nilai 0
Glucose	5
BloodPressure	35
SkinThickness	227
Insulin	374
BMI	11

Setelah dilakukan penggantian nilai menggunakan median, seluruh nilai 0 pada atribut tersebut berhasil diatasi sehingga dataset siap digunakan pada tahap normalisasi dan klasifikasi. Hasil jumlah nilai 0 setelah preprocessing ditunjukkan pada Tabel 3.

Tabel 3. Jumlah Nilai 0 Setelah Preprocessing

Atribut	Jumlah Nilai 0
Glucose	0
BloodPressure	0
SkinThickness	0
Insulin	0
BMI	0

Berdasarkan Tabel 3, proses preprocessing berhasil menghilangkan seluruh nilai 0 pada atribut *Glucose, BloodPressure, SkinThickness, Insulin,* dan *BMI*. Hasil tersebut menunjukkan bahwa kualitas dataset telah menjadi lebih baik sehingga seluruh atribut dapat dimanfaatkan secara optimal pada tahap normalisasi maupun penerapan metode KNN. Perbaikan kualitas data ini penting karena keberadaan nilai yang tidak sesuai dapat memengaruhi proses perhitungan jarak dan menurunkan performa model klasifikasi [15].

Untuk memberikan gambaran yang lebih jelas mengenai proses preprocessing yang telah dilakukan, penelitian ini juga menampilkan sebagian data sebelum dan sesudah penggantian nilai yang tidak valid menggunakan median. Penyajian data tersebut bertujuan memperlihatkan perubahan nilai pada atribut yang sebelumnya memiliki nilai 0 sehingga kualitas data menjadi lebih representatif untuk digunakan pada proses normalisasi dan klasifikasi. Perubahan data sebelum dan sesudah preprocessing disajikan pada tabel 4 dan tabel 5.

Tabel 4. Data Sebelum Preprocessing

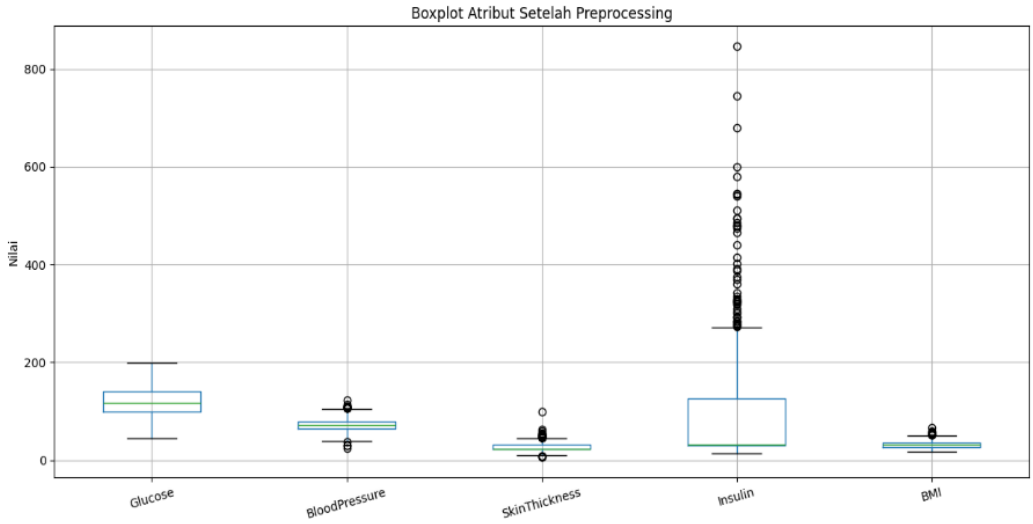
Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DPF	Age	Outcome
6	148	72	35	0	33.6	0.627	50	1
1	85	66	29	0	26.6	0.351	31	0
8	183	64	0	0	23.3	0.672	32	1
5	116	74	0	0	25.6	0.201	30	0
10	115	0	0	0	35.3	0.134	29	0

Tabel 5. Data Setelah Preprocessing

Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DPF	Age	Outcome
6	148	72	35	30.5	33.6	0.627	50	1
1	85	66	29	30.5	26.6	0.351	31	0
8	183	64	23	30.5	23.3	0.672	32	1
5	116	74	23	30.5	25.6	0.201	30	0
10	115	72	23	30.5	35.3	0.134	29	0

Berdasarkan tabel 4 dan tabel 5 dapat dilihat bahwa proses preprocessing berhasil mengganti seluruh nilai 0 pada atribut *Glucose, BloodPressure, SkinThickness, Insulin*, dan *BMI* menggunakan nilai median masing-masing atribut. Perubahan tersebut dilakukan karena nilai 0 pada atribut tersebut tidak merepresentasikan kondisi fisiologis yang valid sehingga dapat memengaruhi proses pembelajaran model. Penggunaan median sebagai metode imputasi dipilih karena lebih tahan terhadap pengaruh nilai ekstrem (outlier) dibandingkan nilai rata-rata (mean), sehingga karakteristik distribusi data dapat dipertahankan setelah proses imputasi dilakukan. Setelah preprocessing, seluruh atribut telah memiliki nilai yang lebih representatif sehingga dataset lebih siap digunakan pada tahap normalisasi dan klasifikasi.

Untuk memberikan gambaran mengenai distribusi data setelah proses preprocessing, penelitian ini menampilkan visualisasi menggunakan boxplot. Visualisasi tersebut digunakan untuk mengamati penyebaran data, nilai tengah, serta keberadaan nilai ekstrem (outlier) pada setiap atribut yang digunakan dalam penelitian. Hasil visualisasi tersebut disajikan pada Gambar 2.



Gambar 2. Visualisasi Sebaran Data Atribut Setelah Tahap Preprocessing

Berdasarkan gambar 2, setiap atribut menunjukkan karakteristik sebaran data yang berbeda sesuai dengan kondisi kesehatan pasien pada dataset. Setelah proses preprocessing dilakukan melalui penggantian nilai yang tidak valid menggunakan median, seluruh atribut telah terbebas dari nilai 0 yang sebelumnya tidak merepresentasikan kondisi fisiologis sehingga data menjadi lebih siap untuk digunakan pada tahap normalisasi dan proses klasifikasi. Visualisasi boxplot juga menunjukkan bahwa beberapa atribut masih memiliki nilai ekstrem (outlier), terutama pada atribut Insulin. Keberadaan nilai ekstrem tersebut menunjukkan bahwa variasi data pada dataset masih tetap dipertahankan setelah proses preprocessing, sehingga karakteristik asli dataset tidak berubah. Dengan demikian, proses preprocessing berhasil meningkatkan kualitas data tanpa mengubah distribusi utama yang dimiliki dataset.

3.2. Hasil Normalisasi dan Pembagian Data

Setelah tahap preprocessing selesai dilakukan, data selanjutnya dipersiapkan untuk tahap klasifikasi melalui normalisasi menggunakan metode Min-Max Scaling. Metode ini mengubah nilai setiap atribut ke rentang yang sama, yaitu antara 0 dan 1, sehingga tidak ada atribut yang memiliki pengaruh lebih besar akibat perbedaan skala nilai. Dengan demikian, proses pengukuran jarak pada algoritma *K-Nearest Neighbor* (*KNN*) dapat dilakukan secara lebih seimbang. Data yang telah dinormalisasi kemudian

dipisahkan menjadi dua kelompok, yaitu data training dan data testing. Pada penelitian ini digunakan skenario pembagian data sebesar 80% untuk data training dan 20% untuk data testing. Pembagian tersebut dilakukan agar model dapat mempelajari pola dari sebagian data yang tersedia dan selanjutnya diuji menggunakan data yang belum pernah digunakan pada tahap pelatihan.

Tabel 6. Hasil Pembagian Dataset

Jenis Data	Jumlah Data
Data Training	614
Data Testing	154
Total Data	768

Berdasarkan tabel 6, jumlah data training lebih besar dibandingkan data testing karena data training digunakan untuk membentuk pola pembelajaran pada model KNN. Sementara itu, data testing dimanfaatkan untuk mengevaluasi kemampuan model dalam melakukan klasifikasi terhadap data yang belum pernah digunakan selama proses pelatihan sehingga hasil pengujian yang diperoleh dapat menggambarkan kinerja model secara lebih akurat.

3.3. Hasil Klasifikasi Menggunakan Algoritma KNN

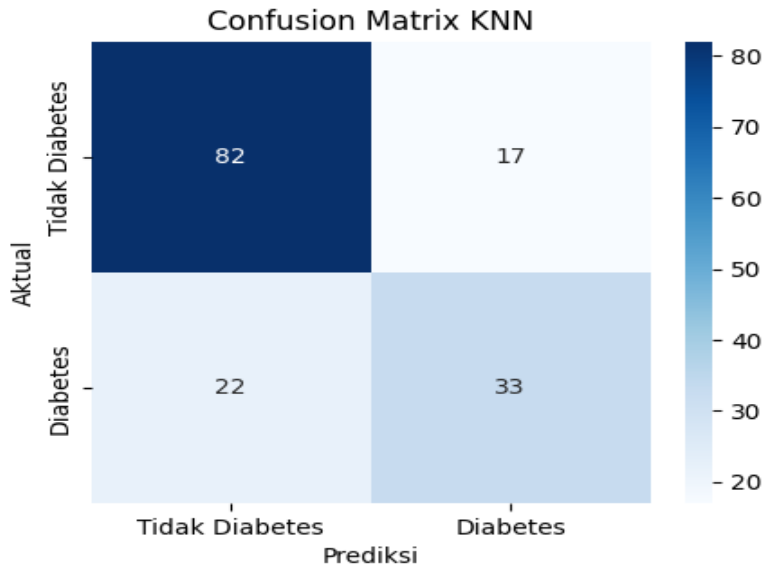
Proses klasifikasi dilakukan menggunakan algoritma *K-Nearest Neighbor* (*KNN*) dengan nilai K = 5 menggunakan data training sebagai data pembelajaran dan data testing sebagai data pengujian. Pada tahap ini model menentukan kelas suatu data berdasarkan kedekatan jarak terhadap sejumlah tetangga terdekat yang telah diketahui kelasnya [6].

Hasil klasifikasi kemudian dievaluasi menggunakan Confusion Matrix untuk mengetahui jumlah prediksi yang benar maupun salah yang dihasilkan oleh model.

Tabel 7. Confusion Matrix

	Prediksi Tidak Diabetes	Prediksi Diabetes
Aktual Tidak Diabetes	82	17
Aktual Diabetes	22	33

Dari tabel 7 dapat diketahui bahwa model berhasil mengidentifikasi 82 data pada kelas tidak diabetes dan 33 data pada kelas diabetes secara tepat. Di sisi lain, terdapat 17 data yang diprediksi sebagai diabetes meskipun sebenarnya termasuk kategori tidak diabetes, serta 22 data diabetes yang tidak berhasil dikenali oleh model. Temuan tersebut menunjukkan bahwa model KNN telah mampu memberikan hasil klasifikasi yang cukup baik, walaupun masih ditemukan sejumlah kesalahan prediksi terutama pada identifikasi pasien diabetes. Jumlah False Negative (FN) yang lebih tinggi dibandingkan False Positive (FP) menunjukkan bahwa model masih mengalami kesulitan dalam mengenali sebagian pasien yang sebenarnya menderita diabetes. Kondisi ini dapat disebabkan oleh adanya kemiripan karakteristik antara data pasien diabetes dan non-diabetes pada beberapa atribut kesehatan sehingga kedekatan jarak antar data dari kedua kelas menjadi semakin sulit dibedakan oleh algoritma K-Nearest Neighbor. Selain itu, distribusi kelas pada Pima Indians Diabetes Dataset yang tidak seimbang, di mana jumlah data non-diabetes lebih banyak dibandingkan data diabetes, dapat menyebabkan model lebih cenderung mengenali kelas mayoritas. Hal tersebut menyebabkan kemampuan model dalam mengenali kelas diabetes menjadi berkurang sehingga jumlah False Negative lebih tinggi dibandingkan False Positive. Kondisi tersebut berdampak pada nilai Recall yang belum optimal [16]. Visualisasi hasil klasifikasi berdasarkan Confusion Matrix dapat dilihat pada gambar 3.



Gambar 3. Confusion Matrix Hasil Klasifikasi KNN

3.4. Evaluasi Model

Evaluasi model dilakukan menggunakan Accuracy, Precision, Recall, dan F1-Score. Penggunaan beberapa metrik evaluasi diperlukan agar performa model dapat dianalisis secara lebih komprehensif dan tidak hanya bergantung pada satu ukuran evaluasi [3].

Tabel 8. Hasil Evaluasi Model KNN

Metrik	Nilai
Accuracy	74,68%
Precision	66,00%
Recall	60,00%
F1-Score	62,86%

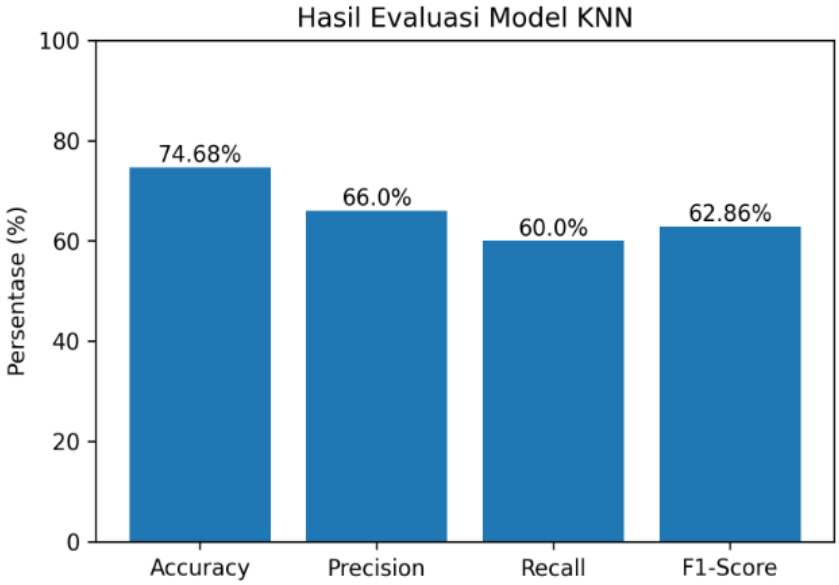
Hasil pengujian menunjukkan bahwa algoritma KNN mampu menghasilkan nilai *accuracy* sebesar 74,68%. Nilai tersebut menunjukkan bahwa sebagian besar data testing berhasil diklasifikasikan dengan benar oleh model.

Nilai *precision* sebesar 66,00% menunjukkan bahwa dari seluruh data yang diprediksi sebagai diabetes, sebanyak 66,00% merupakan prediksi yang benar. Sementara itu, nilai recall sebesar 60,00% menunjukkan bahwa model mampu mengidentifikasi 60,00% pasien yang benar-benar menderita diabetes. Nilai Recall yang lebih rendah dibandingkan Accuracy menunjukkan model masih belum mampu mengenali seluruh pasien diabetes secara optimal. Kondisi tersebut mengindikasikan bahwa masih terdapat sejumlah data positif yang diprediksi sebagai kelas negatif sehingga kemampuan model dalam mendeteksi pasien diabetes belum maksimal. Pada konteks klasifikasi penyakit, kondisi tersebut perlu menjadi perhatian karena kesalahan dalam bentuk False Negative dapat menyebabkan pasien yang sebenarnya menderita diabetes diprediksi sebagai tidak diabetes. Oleh karena itu, peningkatan kemampuan model dalam mendeteksi kelas positif menjadi salah satu aspek yang perlu dipertimbangkan pada penelitian selanjutnya, misalnya melalui optimasi parameter, penerapan validasi silang (cross-validation), atau penanganan ketidakseimbangan kelas [17].

Nilai *F1-Score* sebesar 62,86% menunjukkan keseimbangan antara precision dan recall dalam proses klasifikasi. Semakin tinggi nilai F1-Score, maka semakin baik kemampuan model dalam mempertahankan keseimbangan antara ketepatan dan sensitivitas klasifikasi [3].

Secara keseluruhan, hasil penelitian menunjukkan bahwa algoritma *K-Nearest Neighbor* (KNN) mampu digunakan untuk melakukan klasifikasi penyakit diabetes menggunakan Pima Indians Diabetes Dataset dengan tingkat akurasi sebesar 74,68%.

Temuan penelitian ini konsisten dengan penelitian yang dilakukan oleh [6], yang menunjukkan bahwa algoritma K-Nearest Neighbor (KNN) dapat diterapkan untuk melakukan klasifikasi penyakit diabetes. Perbedaan performa model yang diperoleh dapat dipengaruhi oleh tahapan preprocessing, metode normalisasi, pembagian data training dan testing, serta nilai K yang digunakan dalam proses klasifikasi. Selain itu, karakteristik dataset, seperti distribusi kelas yang tidak seimbang dan kemiripan karakteristik antar kelas, juga dapat memengaruhi kemampuan model dalam membedakan pasien diabetes dan non-diabetes. Hal ini menunjukkan bahwa kualitas data dan parameter yang digunakan memiliki peran penting dalam menentukan performa model KNN. Perbandingan nilai Accuracy, Precision, Recall, dan F1-Score ditampilkan pada gambar 4.



Gambar 4. Perbandingan Nilai Metrik Evaluasi Model KNN

Berdasarkan Gambar 4, metrik evaluasi tertinggi diperoleh pada *Accuracy* dengan nilai 74,68%, diikuti oleh *Precision* sebesar 66,00%, *F1-Score* sebesar 62,86%, dan *Recall* sebesar 60,00%. Tingginya nilai *Accuracy* menunjukkan bahwa model mampu memberikan prediksi yang benar pada sebagian besar data pengujian. Namun, nilai Recall yang masih lebih rendah mengindikasikan bahwa masih terdapat sejumlah kasus diabetes yang belum berhasil dikenali oleh model.

4. KESIMPULAN

Penelitian ini berhasil menerapkan algoritma *K-Nearest Neighbor* (KNN) untuk melakukan klasifikasi penyakit diabetes menggunakan Pima Indians Diabetes Dataset. Berdasarkan hasil pengujian yang telah dilakukan, model memperoleh nilai *Accuracy* sebesar 74,68%, *Precision* sebesar 66,00%, *Recall* sebesar 60,00%, dan *F1-Score* sebesar 62,86%. Nilai tersebut menunjukkan bahwa algoritma KNN mampu memberikan performa yang cukup baik dalam membedakan data pasien diabetes dan tidak diabetes berdasarkan atribut kesehatan yang tersedia pada dataset.

Hasil penelitian membuktikan bahwa metode *K-Nearest Neighbor* dapat dimanfaatkan sebagai salah satu teknik machine learning untuk mendukung proses klasifikasi penyakit diabetes. Meskipun demikian, hasil evaluasi menunjukkan bahwa kemampuan model dalam mengenali seluruh kasus diabetes masih dapat ditingkatkan, sehingga penelitian selanjutnya dapat mempertimbangkan optimasi parameter, penerapan cross-validation, maupun pengembangan metode klasifikasi untuk meningkatkan performa model.

Penelitian ini memiliki beberapa keterbatasan yang perlu diperhatikan dalam menginterpretasikan hasil yang diperoleh. Penelitian ini menggunakan satu konfigurasi parameter, yaitu K = 5, sesuai dengan

ruang lingkup penelitian, sehingga perbandingan performa terhadap variasi nilai K belum menjadi fokus pembahasan. Selain itu, penelitian ini menggunakan metode pembagian data train-test split sehingga proses validasi silang (cross-validation) belum diterapkan. Penelitian ini juga menggunakan Pima Indians Diabetes Dataset sebagai satu-satunya sumber data sehingga hasil yang diperoleh masih terbatas pada karakteristik dataset tersebut dan belum dapat digeneralisasikan pada dataset lain yang memiliki karakteristik berbeda.

DAFTAR PUSTAKA

[1] M. A. Hama Saeed, "Diabetes type 2 classification using machine learning algorithms with up-sampling technique," *Journal of Electrical Systems and Inf Technol*, vol. 10, no. 1, p. 8, Feb. 2023, doi: 10.1186/s43067-023-00074-5.

[2] M. Y. Shams, Z. Tarek, and A. M. Elshewey, "A novel RFE-GRU model for diabetes classification using PIMA Indian dataset," *Sci Rep*, vol. 15, no. 1, p. 982, Jan. 2025, doi: 10.1038/s41598-024-82420-9.

[3] H. H. Rashidi, S. Albahra, S. Robertson, N. K. Tran, and B. Hu, "Common statistical concepts in the supervised Machine Learning arena," *Front. Oncol.*, vol. 13, p. 1130229, Feb. 2023, doi: 10.3389/fonc.2023.1130229.

[4] R. G. Wardhana, G. Wang, and F. Sibuea, "PENERAPAN MACHINE LEARNING DALAM PREDIKSI TINGKAT KASUS PENYAKIT DI INDONESIA," *JOISM*, vol. 5, no. 1, pp. 40–45, Jul. 2023, doi: 10.24076/joism.2023v5i1.1136.

[5] A. K. Ermy Pily, Oktavianda, F. Aprilia, Rahmaddeni, and L. Efrizoni, "Komparasi Algoritma K-Nearest Neighbors dan Naïve Bayes dalam Klasifikasi Penyakit Diabetes Gestasional," *ijcs*, vol. 13, no. 1, Feb. 2024, doi: 10.33022/ijcs.v13i1.3714.

[6] L. N. Mukarromah, Z. Fatah, and I. Yunita, "KLASIFIKASI PENYAKIT DIABETES MENGGUNAKAN METODE K-NEAREST NEIGHBORS (KNN)," vol. 1, no. 4, 2024.

[7] F. D. Musa, D. Purwanto, S. Amri, A. Fadlurohman, and A. Fitriyanan, "Klasifikasi Dataset Diabetes menggunakan Algoritma K-Nearest Neighbors," 2024.

[8] H. M. Saleh, "A Comprehensive Review of Data Mining Techniques for Diabetes Diagnosis Using the Pima Indian Diabetes Dataset," *EDRAAK*, vol. 2024, pp. 39–42, Apr. 2024, doi: 10.70470/EDRAAK/2024/006.

[9] Novian Ikhsan, "Klasifikasi penyakit diabetes menggunakan metode K-Nearest Neighbors (KNN) berdasarkan data klinis," *infotech*, vol. 7, no. 1, pp. 19–26, Jun. 2026, doi: 10.37373/infotech.v7i1.2110.

[10] "Pima Indians Diabetes Database." Accessed: Jun. 02, 2026. [Online]. Available: https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database

[11] A. T. Akbar, H. Prapcoyo, and R. Husaini, "SMOTE and K-Means Preprocessing for Classification by Logistic Regression on Pima Indian Diabetes Dataset," vol. 20, no. 2, 2023.

[12] D. S. F. Azzahrah and A. Alamsyah, "Comparison of Probabilistic Neural Network (PNN) and k-Nearest Neighbor (k-NN) Algorithms for Diabetes Classification," *Recursive J. of Informatics*, vol. 1, no. 2, pp. 73–82, Sep. 2023, doi: 10.15294/rji.v1i2.66078.

[13] Muhammad Randy Fachrezi, Hafiz Aryanda, Alwi Syahputra, and Risma Riansyah, "Sistem Klasifikasi Diabetes Mellitus Menggunakan Algoritma K-Nearest Neighbor (KNN) Berbasis Web," *JUIKTI*, vol. 2, no. 1, pp. 119–130, Jan. 2026, doi: 10.64803/juikti.v2i1.116.

[14] T. Emmanuel, T. Maupong, D. Mpoeleng, T. Semong, B. Mphago, and O. Tabona, "A survey on missing data in machine learning," *J Big Data*, vol. 8, no. 1, p. 140, Oct. 2021, doi: 10.1186/s40537-021-00516-9.

[15] M. Phongying and S. Hiriote, "Diabetes Classification Using Machine Learning Techniques," *Computation*, vol. 11, no. 5, p. 96, May 2023, doi: 10.3390/computation11050096.

[16] A. I. ElSeddawy, F. K. Karim, A. M. Hussein, and D. S. Khafaga, "Predictive Analysis of Diabetes-Risk with Class Imbalance," *Computational Intelligence and Neuroscience*, vol. 2022, pp. 1–16, Oct. 2022, doi: 10.1155/2022/3078025.

[17] M. Sandeep Kumar, M. Zubair Khan, S. Rajendran, A. Noor, A. Stephen Dass, and J. Prabhu, "Imbalanced Classification in Diabetics Using Ensembled Machine Learning," *Computers, Materials & Continua*, vol. 72, no. 3, pp. 4397–4409, 2022, doi: 10.32604/cmc.2022.025865.