STUDI KOMPARATIF: EVALUASI PERFORMA ALGORITMA ARTIFICIAL NEURAL NETWORK DENGAN ALGORITMA MACHINE LEARNING DALAM KLASIFIKASI PENYAKIT DIABETES

Jojor Putri Pasaribu, Zulfahmi Indra, Said Iskandar Al Idrus, Hamidah Nasution, Debi Yandra Niksa
Ilmu Komputer, Universitas Negeri Medan
Jalan WillemPasar V Iskandar Medan, Indonesia
jojorputripasaribu@gmail.com

ABSTRAK

Diabetes adalah penyakit kronis yang ditandai dengan tingginya kadar gula darah dan memerlukan deteksi dini untuk pengelolaan optimal. Karena gejalanya berkembang lambat, diagnosis sering terlambat. Untuk mengatasi hal ini, penelitian terkini memanfaatkan Machine Learning (ML) dan Deep Learning (DL) dalam deteksi dini diabetes. Penelitian ini membandingkan performa algoritma DL (ANN) dan ML (SVM, KNN, Decision Tree, dan Random Forest) berdasarkan akurasi, presisi, recall, dan F1-score. Data yang digunakan berasal dari 486 rekam medis pasien Puskesmas Sri Padang, terdiri atas atribut numerik berupa BMI, kadar gula, tekanan darah, kolesterol, asam urat dan kategorikal berupa gejala klinis. Hasil menunjukkan ANN memiliki akurasi 83,87%, KNN 91,12%, Decision Tree 95,16%, dan SVM 82,25%. Random Forest menunjukkan performa terbaik dengan akurasi 97,58%, precision 95,83%, recall 100%, dan F1-score 97,87%. Dengan teknik ensemble learning, Random Forest terbukti paling andal dalam klasifikasi diabetes.

Kata kunci: *Artificial Neural Network*, diabetes, *machine learning*, perbandingan algoritma, *random forest*

1. PENDAHULUAN

Diabetes merupakan penyakit jangka panjang yang terjadi saat pankreas tidak lagi menghasilkan insulin atau ketika tubuh tidak mampu menggunakan insulin yang diproduksi secara efektif. Menurut laporan *International Diabetes Federation* (IDF) tahun 2021, Indonesia sudah berada di peringkat ke-5 dengan 19,47 juta penderita diabetes dari total populasi 179,72 juta, yang menunjukkan prevalensi diabetes sebesar 10,6% [1].

Diabetes adalah penyakit kronis yang dapat menyerang semua kelompok usia dan berpotensi menimbulkan komplikasi serius. Diagnosis yang tepat sejak dini sangat penting untuk menunjang pengelolaan penyakit ini. Namun, karena gejalanya berkembang secara perlahan, hal ini sering kali menyebabkan keterlambatan dalam proses deteksi [2]. Diabetes sering kali tidak terdeteksi atau terlambat terdiagnosis pada tahap yang sudah lanjut. Akibatnya, kondisi ini dapat menyebabkan berbagai komplikasi serius, seperti kerusakan organ, stroke, dan penyakit jantung. Berdasarkan data dari *International Diabetes Federation* (IDF), 10,5% dari populasi dewasa berusia 20 hingga 79 tahun didiagnosis mengidap diabetes, dan hampir setengah dari mereka tidak menyadari bahwa mereka mengidap penyakit tersebut [3]. Permasalahan selanjutnya karena kurangnya pengetahuan dasar mengenai diabetes menyebabkan banyak orang tidak menyadari bahwa mereka mengidap penyakit pada tahap awal. Saat ini, metode deteksi diabetes umumnya masih bergantung pada pemeriksaan laboratorium, yang membutuhkan waktu relatif lama [4]. Di sisi lain, dalam periode pengumpulan data di Rumah Sakit Islam Siti Khadijah Palembang, jumlah pasien yang menjalani pemeriksaan kesehatan, termasuk penderita diabetes melitus, memengaruhi proses klasifikasi data. Hal ini dapat menimbulkan kesulitan bagi pihak rumah sakit dalam mengelola dan menganalisis data tersebut secara efektif [5]. Beberapa penelitian terbaru telah menggunakan teknologi pembelajaran mesin *(machine learning)* untuk mendukung deteksi penyakit, terutama dalam mengidentifikasi diabetes secara akurat berdasarkan data kesehatan pasien. Pendekatan ini membantu individu mengambil langkah-langkah pencegahan guna mengelola dan menangani kondisi tersebut sejak dini [6]. Akan tetapi Seiring dengan kemajuan teknologi, algoritma ml telah mengalami perkembangan pesat, yang melahirkan sebuah cabang baru dalam bidang ini yang dikenal sebagai *deep learnin*g (dl). DL merupakan pengembangan lebih lanjut dari ml dengan kemampuan yang lebih unggul dalam mengenali pola, mengolah data, serta mengekstraksi informasi dari data tersebut. Salah satu keunggulan utama algoritma dl adalah kemampuannya untuk menangani berbagai bentuk data seperti teks, angka, gambar, audio, dan video, yang masih menjadi tantangan signifikan bagi banyak algoritma ml tradisional [7]. Dalam penelitiannya Clemente menyebutkan bahwa dalam memprediksi potensi *bug* pada *software* algoritma jenis dl memiliki performa lebih baik dibanding algoritma ml [8], hal serupa juga disebutkan oleh Sujatha dalam penelitiannya tentang deteksi penyakit daun pada tanaman, dari total 3 algoritma ml dan 3 algoritma dl secara berturut – turut algoritma dl memberikan performa terbaik dengan menduduki 3 peringkat teratas dari daftar perbandingan performa [9].

Berdasarkan pemaparan masalah sebelumnya, dapat disimpulkan bahwa langkah preventif dalam menangani penyakit diabetes, melalui diagnosis dini terhadap gejala yang muncul, sangatlah penting. Salah

satu pendekatan yang telah dilakukan adalah dengan memanfaatkan algoritma ml sebagai alat bantu diagnosis. Namun, dengan perkembangan teknologi, algoritma DL muncul sebagai metode yang lebih mutakhir dengan sejumlah keunggulan signifikan dibandingkan algoritma ML. Oleh karena itu, dalam penelitian ini akan melakukan perbandingan performa algoritma dl yaitu ANN dengan ML Pengujian ini diharapkan dapat memberikan wawasan lebih dalam mengenai efektivitas ANN dalam mendukung diagnosis medis, khususnya pada kasus diabetes.

2. TINJAUAN PUSTAKA

Beberapa penelitian terbaru memanfaatkan machine learning untuk mendeteksi diabetes secara akurat dari data kesehatan pasien, membantu individu mengambil langkah pencegahan sejak dini [6]. Adapun penelitian yang membandingkan efisiensi model klasifikasi diabetes menggunakan empat teknik pembelajaran mesin: Decision Tree, Random Forest, SVM, dan KNN. Hasil penelitian ini menunjukkan bahwa algoritma-algoritma tersebut efektif dalam melakukan klasifikasi diabetes [10]. Selanjutnya penelitian lain menggunakan model *ensemble stacking* yang menggabungkan berbagai algoritma seperti KNN, *Support Vector Machine* (SVM), *Decision Tree*, dan *Random Forest*, hasilnya model *ensemble* ini mencapai akurasi tertinggi sebesar 88,5%, lebih tinggi dibanding performa algoritma-algoritma tersebut secara terpisah [6].

Dengan kemajuan teknologi, algoritma ML berkembang pesat dan melahirkan deep learning, yang unggul dalam menangani berbagai jenis data seperti teks, angka, gambar, audio, dan video yang masih menjadi tantangan signifikan bagi banyak algoritma ml tradisional [7]. Dalam penelitian Penyakit Jantung Menggunakan Metode *Artificial Neural Network* (ANN) menunjukkan bahwa metode ANN efektif dalam mengklasifikasikan penyakit jantung dengan akurasi 73,77%. Hasil evaluasi dengan presisi 80,43%, recall 84,09%, dan f1-score 82,22% mengindikasikan bahwa ANN memiliki potensi yang baik dalam mendukung diagnosis penyakit jantung [11]. Selanjutnya penelitian tentang prediksi penyakit jantung menggunakan metode *Deep Neural Network* (DNN) menunjukkan bahwa metode ini efektif dan akurat dalam memprediksi penyakit jantung. Selain itu, hasil prediksi divisualisasikan menggunakan Tableau untuk mempermudah interpretasi data [12].

2.1. Penyakit Diabetes

Diabetes merupakan penyakit jangka panjang yang ditandai dengan tingginya kadar gula dalam darah. Glukosa, yang merupakan sumber energi utama bagi tubuh, tidak dapat dimanfaatkan dengan optimal oleh tubuh pada penderita diabetes [13]. Diabetes terbagi menjadi beberapa jenis, yaitu diabetes tipe 1, diabetes tipe 2, diabetes gestasional, dan prediabetes. Diabetes tipe 1 terjadi ketika sistem kekebalan tubuh menyerang sel-sel pankreas yang bertanggung-jawab untuk memproduksi insulin, sehingga produksi insulin berkurang atau berhenti. Diabetes tipe 2 disebabkan oleh resistensi insulin, di mana tubuh tidak dapat menggunakan insulin secara optimal, sehingga gula darah menumpuk. Diabetes gestasional terjadi pada ibu hamil akibat perubahan hormon selama kehamilan, sedangkan Prediabetes adalah suatu kondisi di mana kadar gula darah berada di atas normal, tetapi belum mencapai level yang cukup untuk dikategorikan sebagai diabetes tipe 2. Gejala diabetes tipe 1 dan diabetes gestasional umumnya muncul secara tiba-tiba, sedangkan gejala prediabetes dan diabetes tipe 2 berkembang secara perlahan, sehingga penderita diabetes tipe 2 sering tidak menyadari penyakitnya selama bertahun-tahun [13].

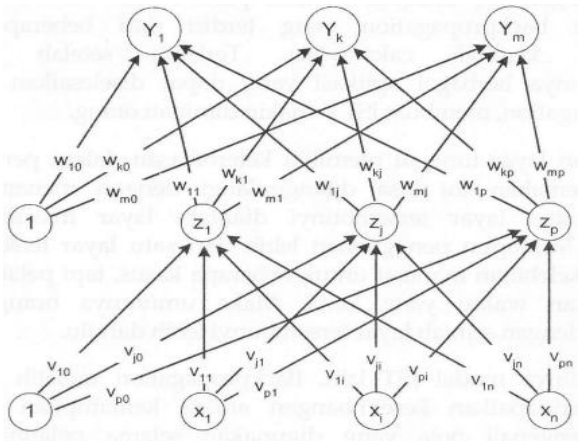
Menurut [14] terdapat beberapa gejala diabetes yang perlu diwaspadai antara lain yaitu Polydipsia yaitu rasa haus yang berlebihan, Polyuria yaitu sering buang air kecil dalam jumlah banyak, Polyphagia yaitu rasa lapar yang terus-menerus, Penurunan berat badan secara tiba-tiba, Kelelahan atau kelemahan yang tidak biasa, Pandangan kabur atau *visual blurring*, kesemutan di tangan dan kaki, luka yang sulit sembuh atau sering mengalami infeksi, acanthosis nigricans, yaitu munculnya area kulit yang berwarna gelap seperti di lipatan leher, ketiak atau selangkangan. Apabila seseorang mengalami beberapa gejala yang telah disebutkan sebelumnya, disarankan untuk segera melakukan konsultasi medis guna mendapatkan diagnosis yang akurat. Deteksi dini dan penanganan yang cepat berperan penting dalam mencegah kemungkinan komplikasi yang lebih serius. Oleh karena itu, pemeriksaan kadar gula darah secara berkala serta konsultasi dengan tenaga medis profesional perlu dilakukan sebagai upaya pencegahan untuk menjaga kesehatan.

2.2. *Deep Learning*

Deep Learning (DL) adalah metode yang dikembangkan dari ml dengan berbagai keunggulan, seperti kemampuan menangani data yang lebih besar dan lebih kompleks serta fitur ekstraksi yang dapat dilakukan secara otomatis. Pada *machine learning*, proses ekstraksi fitur biasanya dilakukan secara terpisah dari tahap pembangunan model, sedangkan dl menyatukan kedua proses tersebut. Metode ini juga dikenal sebagai *deep neural network (*DNN), yaitu jaringan neural berlapis-lapis yang mengadopsi prinsip kerja jaringan saraf di otak manusia [15]. Salah satu keunggulan utama *deep learning* adalah kemampuannya dalam meniru cara kerja jaringan saraf manusia, sehingga model dapat belajar dari data tanpa harus dirancang secara eksplisit. Selain itu, DL juga banyak digunakan dalam pengenalan wajah, analisis sentimen, dan sistem rekomendasi, seperti yang diterapkan oleh platform streaming dan e-*commerce* untuk memberikan saran yang lebih personal kepada pengguna [16].

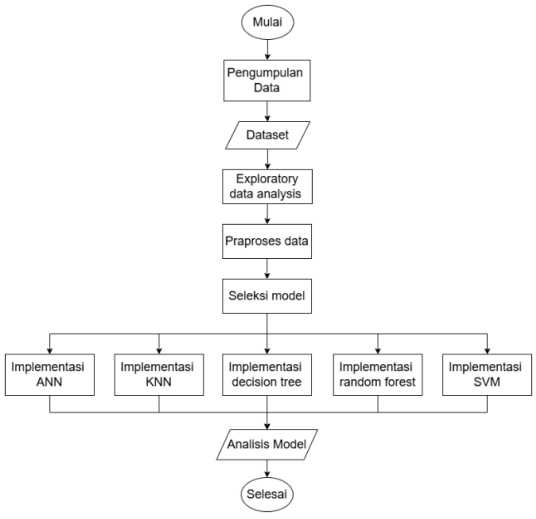
2.3. *Artificial Neural Network* (ANN)

ANN merupakan model komputasi yang terinspirasi dari struktur saraf otak manusia, yang mampu belajar dari pengalaman dan menyelesaikan masalah kompleks dengan efisien. Model ini menawarkan pendekatan sederhana untuk solusi berbasis mesin dan memiliki ketahanan lebih baik terhadap beban tinggi dibandingkan metode tradisional [17]. Backpropagation adalah algoritma dalam jaringan saraf tiruan yang bekerja melalui dua tahap: training untuk memperbarui bobot secara iteratif hingga error minimal, dan testing untuk menguji bobot terbaik yang telah diperoleh. Arsitekturnya terdiri dari lapisan input, tersembunyi, dan output [18]. Pada Gambar 1, setiap lingkaran mewakili neuron dalam lapisan jaringan, dengan panah masuk dan keluar yang menunjukkan arah koneksi sesuai jenis lapisannya. Panah tersebut merepresentasikan bobot yang mencerminkan pentingnya koneksi antar neuron. Lapisan terbawah adalah input, di tengah adalah lapisan tersembunyi, dan paling atas adalah lapisan output.



Gambar 1. Arsitektur *backpropagation* [18]

3. METODE PENELITIAN



Gambar 2. Alur Penelitian

3.1. Pengumpulan Data

Dataset ini diperoleh dari dataset rekam medis di puskesmas sri padang. Pada tahap ini, data dibaca dan diintegrasikan ke dalam sistem agar dapat diolah lebih lanjut dalam tahap berikutnya.

3.2. *Exploratory Data Analysis*

Dalam penelitian ini, EDA dimulai dengan penilaian kualitas data, mencakup identifikasi missing value, duplikat, dan outlier. Missing value diperiksa untuk memastikan tidak ada data kosong yang mengganggu analisis, data duplikat dihapus agar tidak menimbulkan bias, sementara outlier dideteksi untuk mengenali data menyimpang yang bisa memengaruhi hasil. Setelah itu, dilakukan eksplorasi distribusi data, di mana fitur numerik dianalisis dengan histogram dan kurva KDE, sedangkan fitur kategorikal dengan diagram batang. Analisis ini membantu mengidentifikasi pola distribusi dan potensi ketidakseimbangan data secara menyeluruh.

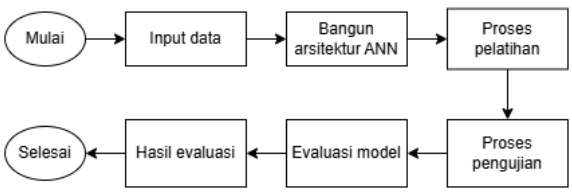
3.3. *Praproses Data*

Setelah data dimuat, dilakukan praprosesuntuk meningkatkan kualitas data sebelum digunakan dalam model *machine learning*. Tahapan ini mencakup penanganan missing value dengan menghapus data yang tidak lengkap, normalisasi atau standarisasi seperti Min-Max atau Z-score untuk menyamakan skala data, serta pembersihan data dari duplikasi, nilai tidak valid, dan outlier. Penelitian ini menggunakan algoritma supervised learning, sehingga data dibagi menjadi data latih dan data uji (*data splitting*). Data latih digunakan untuk membangun model, sedangkan data uji untuk mengevaluasi kinerjanya. Pembagian data mengacu pada penelitian sebelumnya, yaitu 70% untuk pelatihan dan 30% untuk pengujian [19].

3.4. Seleksi Model

Pada tahap seleksi model, beberapa algoritma klasifikasi seperti ANN, KNN, decision tree, random forest, dan SVM akan diimplementasikan dan dievaluasi untuk menentukan model terbaik dalam klasifikasi penyakit diabetes. Setiap model dilatih dengan data yang telah diproses dan diuji menggunakan metrik evaluasi seperti akurasi, precision, recall, dan F1-score. Pemilihan model terbaik didasarkan pada performa keseluruhan terhadap data uji, terutama dalam mengklasifikasikan penyakit diabetes secara akurat.

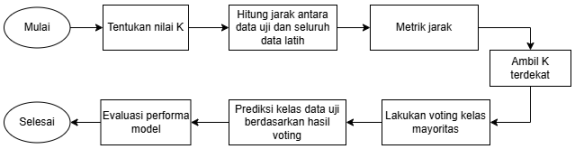
3.5. Implementasi ANN



Gambar 3. Alur implementasi model ANN

Pada tahap implementasi ANN, model dibangun menggunakan data yang telah melalui praproses dan eksplorasi. Jaringan terdiri dari *input layer*, satu atau lebih *hidden layer*, dan *output layer*. Pelatihan dilakukan dengan data training, sedangkan data testing digunakan untuk evaluasi. Parameter seperti epoch, *batch size*, dan *learning rate* disesuaikan untuk hasil optimal. Kinerja model dievaluasi menggunakan metrik akurasi, *precision*, *recall*, dan F1-*score* dalam mengklasifikasikan data menjadi diabetes dan tidak diabetes.

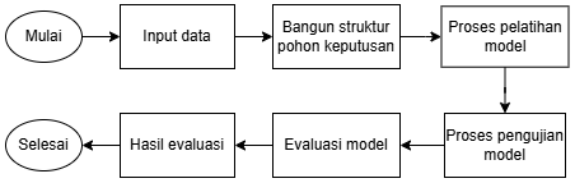
3.6. Implementasi KNN



Gambar 4. Alur implementasi KNN

Pada implementasi KNN, langkah awal adalah menentukan nilai k, yaitu jumlah tetangga terdekat yang digunakan dalam klasifikasi. Setelah k ditentukan, dilakukan perhitungan jarak misalnya dengan *Minkowski distance* antara data uji dan seluruh data latih. Kemudian, k data latih terdekat dipilih, dan kelas data uji diprediksi berdasarkan mayoritas kelas dari tetangga tersebut. Proses ini diterapkan pada seluruh data uji untuk memperoleh hasil prediksi.

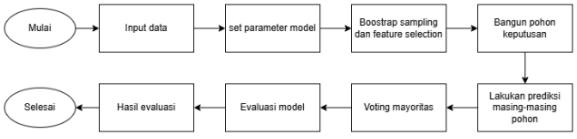
3.7. Implementasi *Decision Tree*



Gambar 5. Alur implementasi *decision tree*

Model dilatih menggunakan data latih yang telah dipraproses, lalu diuji dengan data uji untuk mengukur kemampuannya mengklasifikasikan data baru. Kinerjanya dievaluasi menggunakan metrik akurasi, presisi, recall, dan F1-*score* untuk menilai efektivitas dalam klasifikasi penyakit diabetes.

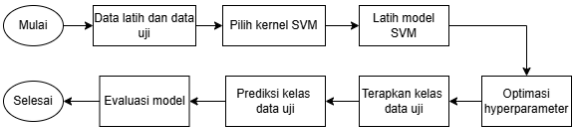
3.8. Implementasi *Random Forest*



Gambar 6. Alur implementasi *random forest*

Implementasi algoritma random forest dimulai dengan membangun sebuah model yang terdiri dari banyak pohon keputusan, yang disebut dengan estimator. Pada model ini, setiap pohon keputusan dibangun dengan menggunakan subset acak dari data latih dan fitur yang berbeda, sehingga memperkenalkan keberagaman di antara pohon-pohon tersebut. Selama proses pelatihan, model random forest melibatkan pembagian data latih menjadi banyak subset, dan masing-masing subset ini digunakan untuk membangun pohon keputusan yang berbeda.

3.9. Implementasi SVM



Gambar 7. Alur implementasi SVM

SVM menggunakan fungsi kernel seperti linear, polynomial, atau RBF untuk menangani data yang tidak terpisah secara linear, sesuai dengan karakteristik data. Model dilatih menggunakan data latih, lalu digunakan untuk memprediksi data uji dengan mengklasifikasikannya berdasarkan posisi relatif terhadap hyperplane yang telah ditentukan.

3.10. Evaluasi Model

Setelah model algoritma dibangun, langkah selanjutnya adalah evaluasi kinerja masing-masing menggunakan metrik akurasi, presisi, recall, dan F1-score. Evaluasi ini bertujuan menilai kemampuan model dalam menggeneralisasi pola pada data uji dan membandingkan efektivitas antar algoritma.

3.11. Analisis Model

Setelah evaluasi, hasil dianalisis untuk mengidentifikasi keunggulan dan kelemahan tiap model berdasarkan metrik kinerja, interpretasi hasil, serta faktor yang memengaruhi performa seperti kualitas data, parameter, dan kompleksitas algoritma. Analisis ini membantu menentukan model terbaik untuk studi kasus, dan hasilnya dirangkum dalam laporan evaluasi sebagai dasar kesimpulan efektivitas metode yang digunakan.

4. HASIL DAN PEMBAHASAN

4.1. Data assement

Hasil analisis menunjukkan bahwa sebagian besar fitur dalam dataset tidak memiliki nilai yang hilang. Namun, ditemukan dua kolom yang memiliki missing value seperti null atau NaN *(Not a Number)*, yaitu pada kolom kolestrol total sebanyak 9 nilai yang hilang dan kadar asam urat sebanyak 3 nilai yang hilang hal ini dapat dilihat dari Gambar 4.1. Meskipun demikian, jumlah nilai hilang tersebut tergolong rendah dibandingkan keseluruhan data.

```
Missing values per fitur:
Age                        0
Gender                     0
Berat Badan (kg)           0
Tinggi Badan (cm)          0
Tinggi Badan (m)           0
BMI                        0
Polyuria                   0
Polydipsia                 0
Polyphagia                 0
Sudden weight loss         0
Weakness                   0
Visual blurring            0
Itching                    0
Obesity                    0
Tangan/kaki nyeri          0
pusing                     0
sakit kepala               0
Tangan/kaki Kebas          0
Delayed healing            0
Tangan/kaki mati rasa      0
Tengkuk/leher sakit        0
Gula Darah (mg/dL)         0
Sistole (mm)               0
Diastole (Hg)              0
Kolestrol Total            9
Kadar Asam Urat            3
Diabetes                   0
```

Gambar 8. Cek *missing value*

```
np.int64(0)
```

Gambar 9. *Duplicate value*

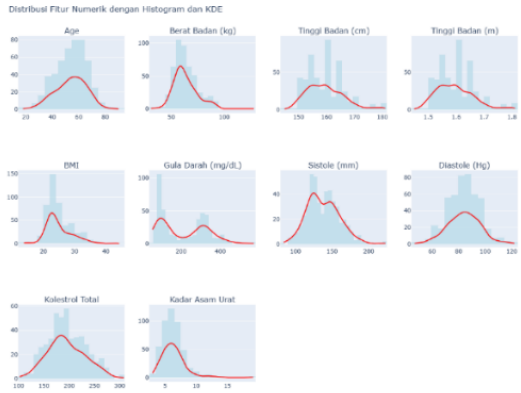
Langkah selanjutnya dalam dalam proses *data assessment* adalah melakukan identifikasi terhadap keberadaan data duplikat *(duplicate value).* Berdasarkan hasil analisis, tidak ditemukan adanya baris data yang bersifat duplikat dalam dataset yang digunakan. Hal ini menunjukkan bahwa setiap entri data bersifat unik dan merepresentasikan informasi yang berbeda.

	nama_kolom	persentase_outlier
0	Age	0.82
1	Berat Badan (kg)	2.06
2	Tinggi Badan (cm)	2.67
3	Tinggi Badan (m)	2.67
4	BMI	2.67
5	Gula Darah (mg/dL)	0.00
6	Sistole (mm)	1.03
7	Diastole (Hg)	1.44
8	Kolestrol Total	0.00
9	Kadar Asam Urat	0.00

Gambar 10. Tabel *persentase outlier*

Tahap terakhir yaitu analisis outlier, proses identifikasi outlier dilakukan menggunakan metode *Interquartile Range* (IQR), di mana nilai-nilai pencilan ditentukan berdasarkan batas bawah (Q1 - 1.5 × IQR) dan batas atas (Q3 + 1.5 × IQR). Setiap nilai yang berada di luar rentang ini dianggap sebagai outlier.

4.2. Ekplorasi data



Gambar 11. Distribusi *fitur numerik*

Masing-masing atribut numerik. Berdasarkan visualisasi, sebagian besar fitur numerik menunjukkan pola distribusi yang tidak normal, baik dalam bentuk distribusi miring (*skewed*) maupun multimodal yaitu memiliki lebih dari satu puncak. Sebagai contoh, fitur BMI, gula darah, dan tinggi badan dalam satuan cm maupun meter memperlihatkan penyimpangan signifikan dari distribusi normal, yang ditunjukkan oleh bentuk kurva yang asimetris atau bertumpuk di beberapa titik. Ketidaksesuaian terhadap distribusi normal ini menjadi pertimbangan penting dalam pemilihan metode praproses data.

Gambar 12. Distribusi fitur kategorikal

Dari visualisasi dapat diamati bahwa sebagian besar fitur menunjukkan ketidakseimbangan jumlah antara kategori “yes” dan “no”, di mana kategori “no” umumnya lebih dominan, seperti pada fitur polyuria, polyphagia, *visual blurring*, dan *itching*. Sementara itu, fitur seperti delayed healing, tangan/kaki mati rasa,

serta tengkuk/leher sakit memiliki distribusi kategori yang lebih seimbang.

Selain itu, ditemukan juga adanya anomali kategori pada beberapa kolom, seperti pada fitur *sudden weight loss* dan pusing, yang menunjukkan adanya variasi penulisan pada nilai "yes" dan "no" misalnya perbedaan huruf kapital atau spasi berlebih.

4.3. Data cleaning

```
Dimensi data sebelum penerapan teknik droping : (486, 27)
Dimensi data setelah penerapan teknik droping : (474, 27)
```

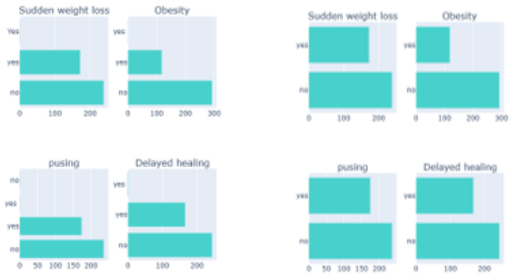
Gambar 13. *Hasil droping missing value*

Pada baris ke-27 ditemukan nilai hilang pada kolom kolestrol total, sehingga seluruh baris tersebut dieliminasi dari dataset. Penerapan teknik ini menyebabkan perubahan dimensi data dari semula 486 baris dan 27 kolom menjadi 474 baris dan 27 kolom.

```
Dimensi data sebelum menghapus outlier : (474, 27)
Dimensi data setelah menghapus outlier : (412, 27)
```

Gambar 14. Hasil *droping* mengatasi *outlier*

Setelah proses penghapusan *outlier* dilakukan, dimensi data mengalami pengurangan dari 474 baris dan 27 kolom menjadi 412 baris dan 27 kolom.



Gambar 15. Sebelum dan sesudah mengatasi data tidak konsisten

4.4. Fitur engineering

Hal ini terlihat pada gambar 16 Oleh karena itu, dibentuk kolom baru bernama Kategori_BMI yang merupakan hasil pengelompokan nilai BMI ke dalam empat kategori, yaitu *underweight*, *normal*, *overweight*, dan *obesity.*

BMI	Polyuria	Polydipsia	Polyphagia	Sudden weight loss	...	Delayed healing	Tangan/kaki mati rasa	Tengkuk/leher sakit	Gula Darah (mg/dL)	Sistole (mm)	Diastole (Hg)	Kolestrol Total	Kadar Asam Urat	Diabetes	Kategori_BMI
26.666667	yes	no	no	no	...	yes	no	yes	410	115	68	228.0	3.0	yes	Overweight
24.973985	yes	no	no	no	...	yes	no	no	308	116	68	136.0	4.0	yes	Overweight
26.446281	no	no	yes	no	...	yes	yes	no	322	112	60	178.0	4.2	yes	Overweight
29.296875	yes	no	yes	no	...	no	yes	no	266	129	84	172.0	7.4	yes	Obesitas
27.209608	yes	no	no	no	...	yes	no	yes	290	120	81	190.0	3.8	yes	Overweight

Gambar 16. Contoh *fitur engineering*

4.5. Normalisasi data

Untuk fitur numerik seperti gula darah, tekanan darah sistolik, tekanan darah diastolik, kolesterol, dan asam urat dilakukan proses normalisasi menggunakan metode Min-Max *Normalization*, dengan rumus:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (1)$$

Berikut adalah contoh hasil dari normalisasi fitur numerik dan konversi fitur kategorik ke numerik dari keseluruhan data:

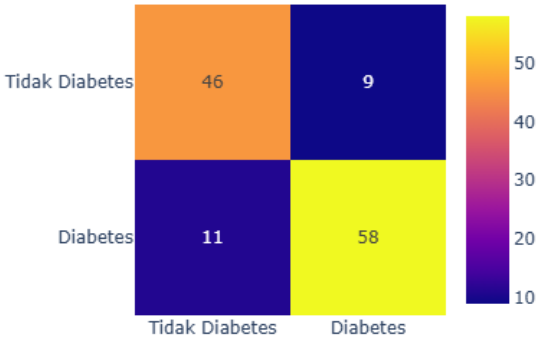
	Kategori_BMI	Polyuria	Polydipsia	Polyphagia	Tangan/kaki nyeri	Gula Darah (mg/dL)	Sistole (mm)	Diastole (Hg)	Kolestrol Total	Kadar Asam Urat	Diabetes
0	Overweight	yes	no	no	yes	0.080021	0.040930	0.051450	0.056792	0.023719	yes
1	Overweight	yes	no	no	no	0.060564	0.041286	0.039757	0.033876	0.031625	yes
2	Overweight	no	no	yes	yes	0.063317	0.039862	0.035079	0.044338	0.033206	yes
3	Obesitas	yes	no	yes	no	0.092305	0.045912	0.049111	0.042843	0.058506	yes
5	Overweight	yes	no	no	yes	0.057025	0.042709	0.047357	0.047327	0.030044	yes

Gambar 17. Sampel hasil data setelah normalisasi fitur numerik

	Kategori_BMI	Polyuria	Polydipsia	Polyphagia	Tangan/kaki nyeri	Gula Darah (mg/dL)	Sistole (mm)	Diastole (Hg)	Kolestrol Total	Kadar Asam Urat	Diabetes
0	2	1	0	0	1	0.080021	0.040930	0.051450	0.056792	0.023719	1
1	2	1	0	0	0	0.060564	0.041286	0.039757	0.033876	0.031625	1
2	2	0	0	1	1	0.063317	0.039862	0.035079	0.044338	0.033206	1
3	1	1	0	1	0	0.092305	0.045912	0.049111	0.042843	0.058506	1
5	2	1	0	0	1	0.057025	0.042709	0.047357	0.047327	0.030044	1

Gambar 18. Sampel hasil data setelah konversi fitur kategorik ke numerik

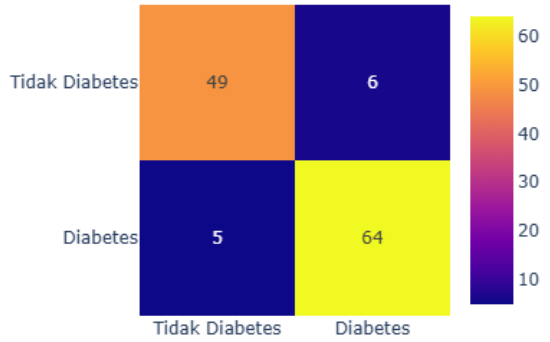
4.6. Implementasi ANN



Gambar 19. Confusion matrix ANN

Model ANN dilatih selama 200 epoch dengan waktu pelatihan cepat, sekitar 4–5 milidetik per epoch. Akurasi pelatihan mencapai 87,68% dengan loss 30,59%, sedangkan akurasi pengujian sebesar 83,87%, mengalami penurunan sekitar 3,81%. Berdasarkan confusion matrix, terdapat 46 data yang benar diprediksi sebagai tidak diabetes (True Negative), 9 data tidak diabetes yang salah diprediksi sebagai diabetes (False Positive), 11 data diabetes yang salah diprediksi sebagai tidak diabetes (False Negative), dan 58 data diabetes yang diprediksi dengan benar (True Positive). Hasil ini menunjukkan bahwa model ANN cukup mampu membedakan data positif dan negatif, meskipun masih terdapat beberapa kesalahan klasifikasi. Secara keseluruhan, performa model sudah baik, namun masih memiliki ruang untuk ditingkatkan.

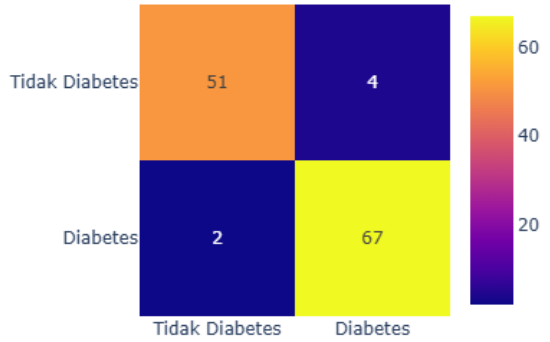
4.7. Implementasi KNN



Gambar 20. Confusion matrix KNN

Model KNN dikembangkan dengan nilai k = 5 dan menggunakan jarak Minkowski untuk mengukur kedekatan antar data. Berdasarkan confusion matrix, model berhasil memprediksi 49 data tidak diabetes dengan benar (True Negative), 6 data tidak diabetes salah diprediksi sebagai diabetes (False Positive), 5 data diabetes salah diprediksi sebagai tidak diabetes (False Negative), dan 64 data diabetes diprediksi dengan benar (True Positive). Hasil ini menunjukkan bahwa KNN mampu mengklasifikasikan data dengan tingkat akurasi yang tinggi dan membedakan pasien diabetes dan non-diabetes secara andal.

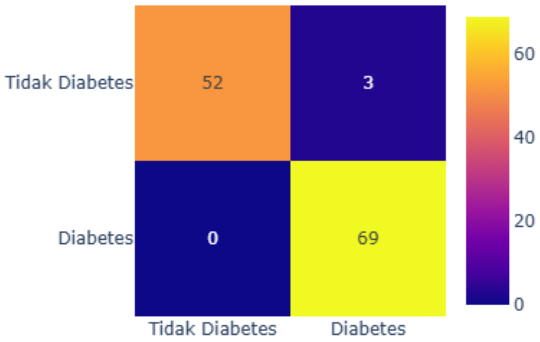
4.8. Implementasi decision tree



Gambar 21. Confusion matrix decision tree

Model Decision tree dibangun menggunakan kriteria pemisahan *Gini impurity* dan metode *splitter* "best" untuk menghasilkan pemisahan optimal di setiap node. Berdasarkan confusion matrix, diperoleh 51 data tidak diabetes yang diprediksi benar (True Negative), 4 data tidak diabetes salah diprediksi sebagai diabetes (False Positive), 2 data diabetes salah diprediksi sebagai tidak diabetes (False Negative), dan 67 data diabetes diprediksi dengan benar (True Positive). Hasil ini menunjukkan bahwa model Decision Tree memiliki performa klasifikasi yang sangat baik dalam membedakan antara pasien diabetes dan non-diabetes.

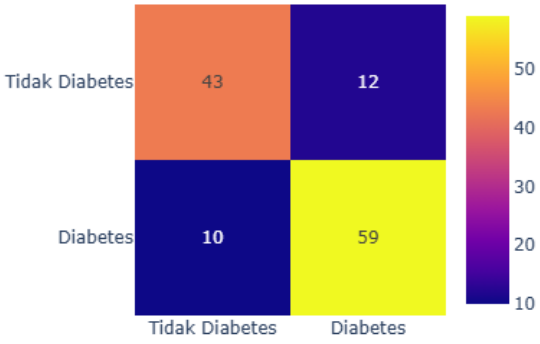
4.9. Implementasi random forest



Gambar 22. Confusion matrix random forest

Model Random Forest dibangun menggunakan 100 pohon keputusan dengan kriteria pemisahan Gini impurity, dan hasil akhir ditentukan melalui voting mayoritas. Berdasarkan confusion matrix, diperoleh 52 data tidak diabetes yang diprediksi benar (True Negative), 3 data tidak diabetes salah diprediksi sebagai diabetes (False Positive), tidak ada kesalahan prediksi untuk pasien diabetes (False Negative = 0), dan 69 data diabetes berhasil diprediksi dengan benar (True Positive). Hasil ini menunjukkan bahwa Random Forest sangat andal dalam klasifikasi penyakit diabetes dengan tingkat kesalahan yang sangat rendah.

4.10. Implementasi SVM



Gambar 23. Confusion matrix SVM

Model SVM dibangun menggunakan kernel Radial Basis Function (RBF) dengan toleransi 0,001 sebagai batas konvergensi. Berdasarkan confusion matrix, diperoleh 43 data tidak diabetes yang diprediksi benar (True Negative), 12 data tidak diabetes salah diprediksi sebagai diabetes (False Positive), 10 data diabetes salah diprediksi sebagai tidak diabetes (False Negative), dan 59 data diabetes diprediksi dengan benar (True Positive). Hasil ini menunjukkan bahwa meskipun SVM mampu mengenali sebagian besar pola dengan baik, masih terdapat ruang perbaikan, seperti optimasi parameter, seleksi fitur, atau penerapan algoritma ensemble untuk meningkatkan presisi klasifikasi.

4.11. Evaluasi Model

Berdasarkan evaluasi, ANN mencatat 58 true positive dengan 9 false positive dan 11 false negative, menunjukkan performa yang cukup baik meskipun masih ada kesalahan klasifikasi. KNN lebih unggul dengan 64 true positive, 6 false positive, dan 5 false negative. Decision tree menunjukkan hasil lebih baik lagi dengan 67 true positive, 4 false positive, dan 2 false negative, mencerminkan akurasi tinggi. Random forest menjadi yang terbaik, mencatat 69 true positive, hanya 3 false positive, dan tanpa false negative, menunjukkan keandalan tinggi. Sebaliknya, SVM menghasilkan 59 true positive dengan 12 false positive dan 10 false negative, menjadikannya model dengan kesalahan klasifikasi terbanyak.

Tabel 1. Confusion matrix algoritma

Algoritma	TP	FP	FN	TN
ANN	58	9	11	46
KNN	64	6	5	49
Decision tree	67	4	2	51
Random forest	69	3	0	52
SVM	59	12	10	43

4.12. Analisis Model

Pada tahap analisis model, lima algoritma dievaluasi. ANN mencatat akurasi 83,87% dengan precision tinggi namun recall sedikit lebih rendah, menunjukkan potensi melewatkan data positif. KNN tampil lebih baik dengan akurasi 91,12% dan keseimbangan antara *precision* dan *recall*, meski tetap ada risiko kesalahan pada data dengan fitur kurang jelas. *Decision tree* menunjukkan akurasi 95,16% dan generalisasi yang baik, meskipun rentan *overfittin*g. *Random forest* menjadi model terbaik dengan akurasi 97,58% dan recall 100%, mampu mendeteksi seluruh data positif tanpa kesalahan. Sementara itu, SVM mencatat akurasi 82,25% dengan performa paling rendah, terutama dalam membedakan data yang tidak terpisah secara linear.

Tabel 2. Perbandingan pada masing-masing algoritma

Algoritma	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
ANN	83.87	86.57	84.06	85.29
KNN	91.13	91.43	92.75	92.09
Decision tree	95.16	94.37	97.10	95.71
Random forest	97.58	95.83	100.0	97.87
SVM	82.26	83.10	85.56	84.28

5. KESIMPULAN DAN SARAN

Penelitian ini berhsil membandingkan performa algoritma machine learning dan deep learning dalam klasifikasi penyakit diabetes. Hasil evaluasi menunjukkan bahwa random forest unggul dengan akurasi 97,58%, precision 95,83%, recall 100%, dan F1-score 97,87%, mengungguli ANN, KNN, decision tree, dan SVM. Meskipun deep learning sering dianggap lebih unggul, hasil ini menunjukkan bahwa pada kasus sederhana seperti klasifikasi diabetes berbasis data klinis, algoritma machine learning seperti random forest dapat memberikan hasil lebih baik. Oleh karena itu, pemilihan algoritma harus disesuaikan dengan kompleksitas masalah dan karakteristik data. Disarankan dilakukan optimasi lebih lanjut seperti tuning hyperparameter untuk meningkatkan performa model, terutama pada algoritma dengan tingkat kesalahan klasifikasi tinggi. Penelitian selanjutnya dapat mengeksplorasi metode ensemble lain, deep learning yang lebih kompleks, atau algoritma terbaru, serta menguji model pada dataset yang lebih besar dan beragam guna memastikan stabilitas dan konsistensinya.

DAFTAR PUSTAKA

[1] D. Ratnasari, "Comparison of Performance of Four Distance Metric Algorithms in K-Nearest Neighbor Method on Diabetes Patient Data," *Indones. J. Data Sci.*, vol. 4, no. 2, pp. 97–108, 2023.

[2] O. M. Haq, A. Ridwan, and T. G. Pratama, "Analisis Perbandingan Kinerja Algoritma Naïve Bayes Dan KNN Untuk Memprediksi Penyakit Diabetes," *J. Ilm. Komput.*, vol. 21, no. 1, pp. 193–201, 2025.

[3] A. Mulyani, S. Khoerunisa, and D. Kurniadi, "Comparison of KNN and SVM Algorithms Performance Using SMOTE to Classify Diabetes," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 14, no. 1, pp. 25–34, 2025.

[4] N. H. Setyawan and N. Wakhidah, "Analisis perbandingan metode logistic regression, random forest, gradient boosting untuk prediksi diabetes," *JIPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, vol. 10, no. 1, pp. 150–162, 2025.

[5] H. Apriyani, "Perbandingan Metode Naïve Bayes Dan Support Vector Machine Dalam Klasifikasi Penyakit Diabetes Melitus," *J. Inf. Technol. Ampera*, vol. 1, no. 3, pp. 133–143, 2020.

[6] M. Bhandarkar, V. S. Bendre, Y. V. Bellary, and A. K. Bhole, "Ensemble stacking classifier model for prediction of diabetes," *Int. J. Informatics Commun. Technol.*, vol. 13, no. 3, pp. 499–508, 2024, doi: 10.11591/ijict.v13i3.pp499-508.

[7] C. C. Aggarwal, *Neural Networks and Deep Learning*, 2nd ed. Springer, 2023.

[8] C. J. Clemente, F. Jaafar, and Y. Malik, "Is Predicting Software Security Bugs using Deep Learning Better than the Traditional Machine Learning Algorithms ?," *2018 IEEE Int. Conf. Softw. Qual. Reliab. Secur.*, pp. 95–102, 2018, doi: 10.1109/QRS.2018.00023.

[9] R. Sujatha, J. Moy, N. Z. Jhanjhi, and S. Nawaz, "Microprocessors and Microsystems Performance of deep learning vs machine learning in plant leaf disease detection," *Microprocess. Microsyst.*, vol. 80, no. November

2020, p. 103615, 2021, doi: 10.1016/j.micpro.2020.103615.
[10] Methaporn & Sasiprapa, "Diabetes Classification Using Machine Learning Techniques," *MDPI (Multidisciplinary Digit. Publ. Institute)*, 2023.
[11] D. Galih, M. Luthfi, and M. Farhan, "Klasifikasi Penyakit Jantung Menggunakan Metode Artificial Neural Network," *Indones. J. Data Sci.*, vol. 3, no. 2, pp. 55–60, 2022.
[12] P. Irpanudin, Reka, R. N., Pratama, "PREDIKSI PENYAKIT JANTUNG MENGGUNAKAN METODE DEEP NEURAL NETWORK DENGAN MEMANFAATKAN INTERNET OF THINGS," 2023.
[13] D. Care and S. S. Suppl, "Classification and Diagnosis of Diabetes : Standards of Medical Care in Diabetes d 2019," in *Classification and Diagnosis of Diabetes: Standards ofMedical Care in Diabetes*, vol. 42, no. January, 2019, pp. 13–28. doi: 10.2337/dc19-S002.
[14] F. Siallagan, "Prediksi penyakit diabetes mellitus menggunakan algoritma C4.5," *J. responsif*, vol. 3, no. 1, pp. 44–52, 2021.
[15] I. Goodfellow, *Deep Learning*. 2016.
[16] J. Schmidhuber, *Deep learning in neural networks : An overview*, vol. 61. 2015.
[17] D. Anderson and G. Mcneill, *Artificial neural networks technology*, vol. 0082. 2017.
[18] D. A. Sani and M. Z. Sarwani, "Koreksi jawaban esai berdasarkan persamaan makna menggunakan fasttext dan algoritma backpropagation," *J. Nas. Pendidik. Tek. Inform.*, vol. 11, pp. 102–111, 2022.
[19] S. Siridion, "Klasifikasi diagnosa penyakit diabetes melitus berdasarkan komparasi algoritma supervised learning," *Mutiara Multidiscip. Sci. J.*, vol. 2, no. 3, pp. 1006–1014, 2024.