

# Studi Kinerja Algoritma K-Nearest Neighbors (KNN) untuk Klasifikasi Pasien Diabetes

Edelweiss Rinjani Bawana<sup>1, a)</sup>, Erni Seniwati\*<sup>2, a)</sup>, Peni Febrian Kristami<sup>3, a)</sup>,  
Arsellina Milka Martin<sup>4, a)</sup>, Ninik Tri Hartanti<sup>5, a)</sup>

<sup>a)</sup>*Fakultas Ilmu Komputer, Program Studi Sistem Informasi*

*Universitas Amikom Yogyakarta, Jl. Padjajaran, Ring Road Utara, Yogyakarta 55283, Indonesia*

## *Author Emails*

<sup>2)</sup> Corresponding author: [erni.s@amikom.ac.id](mailto:erni.s@amikom.ac.id)

<sup>1)</sup>[edelweissrinjb@student.amikom.ac.id](mailto:edelweissrinjb@student.amikom.ac.id), <sup>3)</sup>[penifebriankristami@student.amikom.ac.id](mailto:penifebriankristami@student.amikom.ac.id),

<sup>4)</sup>[arsellinamilka@student.amikom.ac.id](mailto:arsellinamilka@student.amikom.ac.id), <sup>5)</sup>[ninik.t@amikom.ac.id](mailto:ninik.t@amikom.ac.id)

**Abstract.** Diabetes is a chronic disease whose number of sufferers continues to increase every year, including in Indonesia. Early detection and accurate classification are crucial to aid the diagnosis and appropriate treatment process, thus minimizing the risk of complications. This study aims to build a diabetes patient classification model and assess the performance of the K-Nearest Neighbor (KNN) algorithm in classifying diabetes patients based on medical data. The dataset used in this study is the Pima Indians Diabetes Dataset, which contains patient health information such as pregnancies (Pregnancies), glucose levels (Glucose), blood pressure (BloodPressure), insulin levels (Insulin), BMI (BMI), age (Age), and patient diagnostic status (Outcome). The research process includes 6 stages of activities carried out. In this study, the parameter value  $k = 8$  is the optimal  $k$  value. Model performance evaluation using a confusion matrix produced an accuracy of 0.83 or 83%, precision (0.78), recall (0.61), and F1-score (0.69). The model was also implemented interactively using Jupyter Notebook and Streamlit as the user interface, allowing users to conduct consultations by entering medical data and simultaneously obtaining predictions. Test results showed that the KNN algorithm performed quite well in classifying patients with and without diabetes.

**Keywords :** Algorithm, Diabetes, Classification, K-Nearest Neighbors (KNN).

**Abstraksi.** Penyakit diabetes merupakan salah satu penyakit kronis yang jumlah penderitanya terus meningkat setiap tahun termasuk di Indonesia. Deteksi dini dan klasifikasi yang akurat sangat penting untuk membantu proses diagnosis dan penanganan yang tepat sehingga meminimalkan resiko komplikasi. Penelitian ini bertujuan untuk membangun model klasifikasi pasien diabetes serta mengkaji kinerja algoritma *K-Nearest Neighbor* (KNN) dalam melakukan klasifikasi pasien diabetes berdasarkan data medis. Dataset yang digunakan dalam penelitian ini adalah *Pima Indians Diabetes Dataset*, yang berisi informasi kesehatan pasien seperti kehamilan (Pregnancies), tingkat glukosa (Glucose), tekanan darah (Blood Pressure), kadar insulin (Insulin), nilai BMI (BMI), usia (Age) dan status diagnosa pasien (Outcome). Proses penelitian mencakup 6 tahapan kegiatan yang dilakukan. Pada penelitian ini menghasilkan parameter nilai  $k=8$  adalah nilai  $k$  optimal. Evaluasi performa model menggunakan confusion matrix yang menghasilkan akurasi yang menghasilkan 0.83 atau 83%, presisi (0.78), recall (0.61) dan F1-score (0.69). Model juga diimplementasikan secara interaktif menggunakan Jupyter Notebook serta penggunaan Streamlit sebagai userinterface sehingga memungkinkan pengguna dapat melakukan konsultasi dengan memasukkan data medis dan sekaligus mendapatkan hasil prediksi. Hasil pengujian menunjukkan bahwa algoritma KNN mampu memberikan performa yang cukup baik dalam mengklasifikasikan pasien yang terkena diabetes dan tidak terkena diabetes.

**Kata Kunci :** Algoritma, Diabetes, Klasifikasi, *K-Nearest Neighbor* (KNN).

## PENDAHULUAN

Di era digital saat ini, perkembangan teknologi informasi telah memberikan dampak signifikan di berbagai sektor, termasuk bidang kesehatan. Salah satu penerapannya adalah pemanfaatan machine learning dalam deteksi penyakit secara otomatis dan lebih cepat. Teknologi ini memungkinkan komputer untuk mengenali pola pada data medis dan melakukan klasifikasi terhadap kondisi pasien, untuk deteksi dini penyakit. Dalam konteks penyakit tidak menular seperti diabetes, penggunaan teknologi menjadi semakin penting seiring dengan bertambahnya jumlah kasus. Menurut data dari International Diabetes Federation (IDF), pada tahun 2024 terdapat sekitar 20,426,400 kasus diabetes di Indonesia, yang merupakan 11,3% dari total populasi dewasa sebanyak 185 juta orang. Jumlah ini menjadikan Indonesia sebagai salah satu negara dengan beban diabetes terbesar di kawasan Western Pacific. IDF juga memperkirakan jumlah penderita diabetes akan terus meningkat secara global hingga mencapai 254 juta orang di kawasan ini pada tahun 2050. Oleh karena itu, dibutuhkan pendekatan teknologi yang dapat membantu proses deteksi dini. Penelitian ini dilakukan untuk merancang model klasifikasi diabetes menggunakan pendekatan machine learning yang dapat mendukung upaya tersebut. Algoritma yang digunakan adalah K-Nearest Neighbors (K-NN), karena kemampuannya dalam mengenali pola-pola data berdasarkan kemiripan antar individu.

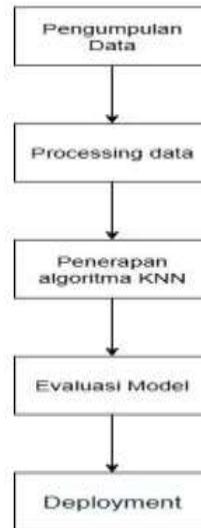
Model klasifikasi dalam penelitian ini dibangun berdasarkan dataset Pima Indians Diabetes yang telah banyak digunakan sebagai acuan dalam pengembangan sistem deteksi penyakit. Dataset ini mengandung beberapa atribut penting seperti jumlah kehamilan, kadar glukosa, tekanan darah, kadar insulin, indeks massa tubuh (BMI), usia, dan status diagnosa pasien (positif atau negatif diabetes). Nilai parameter k algoritma K-Nearest Neighbors kemudian dihitung dengan melakukan uji coba pada berbagai skenario. Ini dilakukan untuk menentukan jumlah tetangga terdekat yang memiliki kinerja terbaik. Untuk mengevaluasi model, metrik seperti akurasi, precision, recall, dan skor F1 digunakan. Metode ini memungkinkan model memprediksi pasien baru berdasarkan kemiripan dengan data sebelumnya. Tujuannya adalah agar proses ini dapat diterapkan pada sistem berbasis perangkat lunak atau aplikasi ponsel untuk skrining awal di fasilitas kesehatan tingkat pertama. Selain itu, penggunaan algoritma yang sederhana namun efektif seperti K-Nearest Neighbors memungkinkan pengembangan teknologi yang murah dan mudah digunakan. Hal ini sangat penting untuk memenuhi kebutuhan deteksi dini di daerah dengan layanan kesehatan yang terbatas.

Studi ini memiliki keterkaitan yang kuat dengan upaya nasional untuk mengatasi masalah diabetes di Indonesia. Menurut data dari IDF (International Diabetes Federation), pada tahun 2024, lebih dari 20 juta orang dewasa di Indonesia akan hidup dengan kondisi diabetes, dan jumlah ini diperkirakan akan terus bertambah tanpa adanya tindakan yang sesuai. Solusi potensial untuk meningkatkan deteksi dini terletak pada pendekatan teknologi, seperti machine learning, terutama melalui digitalisasi dalam pelayanan kesehatan.

Berdasarkan uraian diatas maka dengan menciptakan model yang didasarkan pada data medis seperti yang diusulkan di dalam penelitian ini, proses deteksi diabetes dapat dilakukan secara fleksibel (dapat dilakukan dari mana pun) dan biaya terjangkau. Melalui pendekatan ini, diharapkan dapat mengurangi beban layanan kesehatan akibat penyakit kronis dengan cara yang lebih efektif dalam pencegahan dan deteksi awal. Inti dari penelitian ini adalah kombinasi antara data, algoritma, dan tujuan sosial. Di masa mendatang, pendekatan serupa juga mungkin dapat digunakan untuk mengidentifikasi penyakit lain yang berdampak signifikan terhadap kesehatan masyarakat.

## METODOLOGI PENELITIAN

Pada metodologi penelitian ini dijelaskan bahwa terdapat 5 tahapan yang dilakukan yaitu 1) Pengumpulan data, 2) Processing data, 3) Penerapan algoritma K-Nearest Neighbors, 4) Evaluasi model, dan 5) Deployment. Pada bagian deployment menggunakan bahasa pemrograman Python dan library Streamlit yang ada pada Python digunakan untuk userinterface. Alur penelitian yang terdapat pada metodologi ini dapat dilihat pada Gambar 1.


**GAMBAR 1.** Alur Penelitian

### Pengumpulan Data

Pada penelitian ini, data yang dipakai merupakan data dari kaggle <https://www.kaggle.com/datasets/nancyalaswad90/review> dengan informasi yang tersedia berupa informasi medis. Data tersebut mencakup beberapa variabel penting, seperti kehamilan (Pregnancies), tingkat glukosa (Glucose), tekanan darah (BloodPressure), kadar insulin (Insulin), nilai BMI (BMI), usia (Age) dan status diagnosa pasien (Outcome). Pada diagnosanya sendiri sudah dilabeli dengan angka 1 jika pasien terindikasi diabetes dan 0 jika pasien tidak terindikasi diabetes. Berdasarkan dataset Pima Indians Diabetes ini terdapat fitur-fitur yang digunakan pada dataset dapat dilihat pada Tabel 1.

**TABEL 1.** Tabel Fitur Dataset

Nomor	Nama Fitur	Keterangan Fitur
1.	Pregnancies	Pada dataset ini digunakan sebagai prediksi jumlah kehamilan yang pernah dialami
2.	Glucose	Pada dataset ini digunakan sebagai penyimpanan kadar glukosa plasma 2 jam dalam tes toleransi glukosa oral (mg/dl)
3.	BloodPressure	Digunakan untuk menyimpan data tekanan darah diastolik
4.	Insulin	Penyimpanan data kadar insulin serum
5.	BMI	Digunakan untuk menyimpan data berat badan
6.	Age	Digunakan untuk menyimpan data umur
7.	Outcome	Digunakan sebagai label data yang sudah dinyatakan positif (1) atau negatif (0)

### Processing Data

Tahap Processing Data merupakan tahap penting untuk digunakan dalam proses model machine learning yaitu K-Nearest Neighbors (KNN). Dalam penelitian ini, berikut beberapa langkah processing data yang dilakukan yaitu:

a. Seleksi Fitur

Pada tahap ini, dilakukan pemilihan kolom atau atribut untuk mendukung proses klasifikasi. Beberapa fitur yang dipilih yaitu Pregnancies, Glucose, BloodPressure, BMI, Insulin, dan Age, sedangkan untuk kolom Outcome digunakan sebagai label target. Tujuan dari pemilihan fitur ini adalah untuk menyederhanakan

- struktur model, sehingga membuat model lebih efisien selama proses pemrosesan data.
- b. Eksplorasi Data  
Proses eksplorasi data dilakukan dengan mengecek jumlah data pada masing-masing label Outcome, serta menghitung nilai rata-rata dan standar deviasi dari tiap fitur yang tersedia. Dari hasil eksplorasi, bisa dilihat apakah data tergolong seimbang dan bagaimana persebaran nilainya di tiap fitur.
- c. Normalisasi Data  
Setelah itu, data dinormalisasi menggunakan *StandardScaler*, agar semua fitur berada dalam skala yang sama. Dengan normalisasi, proses penghitungan jarak antar data jadi lebih seimbang dan hasil prediksinya jadi lebih akurat.
- d. Split Data  
Pada tahap berikutnya, data dipisahkan menjadi dua bagian yaitu data latih dan data uji, menggunakan fungsi `train_test_split`. Tujuan dari pembagian ini adalah agar model dapat dilatih menggunakan data latih, lalu diuji kemampuannya menggunakan data uji.

### **Penerapan Algoritma KNN**

Model yang dipakai pada penelitian ini adalah K-Nearst Neighbors. Dengan tujuan utamanya adalah untuk mengklasifikasi apakah pasien masuk ke kategori diabetes atau tidak, berdasarkan data medis yang sudah ada. Cara kerja K-Nearest Neighbors yaitu K-Nearest Neighbors akan memeriksa seberapa mirip data pasien baru dengan data pasien lain yang sudah pernah dicatat sebelumnya. Jadi, keputusan diambil berdasarkan data-data yang memiliki jarak paling dekat untuk membuat prediksi.

### **Evaluasi Model**

Evaluasi model pada penelitian ini dilakukan dengan tujuan untuk mengukur seberapa baik algoritma K-Nearest Neighbors (K-NN) dalam melakukan klasifikasi terhadap data pasien, apakah berdasarkan data pasien maka pasien tersebut tergolong menderita diabetes atau tidak. Pada penelitian ini, tahap evaluasi model dilakukan dengan menggunakan beberapa metrik seperti classification report yang mencakup untuk mengetahui tingkat akurasi (accuracy) serta confusion matrix untuk mengetahui jumlah prediksi yang benar (terkena diabetes) dan salah (tidak terkena diabetes), ketepatan (precision), sensitivitas (recall), dan keseimbangan performa model secara keseluruhan melalui nilai F1-score. Pada tahap evaluasi ini juga untuk dapat melihat performa algoritma K-Nearest Neighbors dalam prediksi data uji.

### **Deployment**

Deployment merupakan tahap yang dijalankan oleh model machine learning yaitu K-Nearest Neighbors, agar bisa diakses langsung oleh pengguna lewat antarmuka aplikasi. Dalam studi ini, proses deployment dilakukan menggunakan bahasa pemrograman Python dan library streamlit, yang dimana model klasifikasi diabetes ditampilkan dalam bentuk web. Dengan adanya antarmuka ini, pengguna dapat memasukkan data yang ada dan mendapatkan langsung hasil prediksi.

Tahap deployment merupakan bagian krusial dalam penelitian ini, karena menentukan apakah model klasifikasi dapat digunakan dalam konteks nyata. Dengan mengimplementasikan model K-Nearest Neighbors ke dalam bahasa pemrograman Python dan menghubungkannya melalui antarmuka web, pengguna dapat memanfaatkan teknologi ini untuk melakukan deteksi dini diabetes. Langkah deployment yang dilakukan juga mempertimbangkan aspek validasi input, keamanan, dan skalabilitas sistem.

## **HASIL DAN PEMBAHASAN**

### **Pengumpulan Dataset**

Pada penelitian ini melibatkan 768 data pasien yang merupakan gabungan dari pasien yang positif diabetes ataupun yang negatif diabetes.

Pregnancies	Glucose	BloodPressure	BMI	Insulin	Age	Outcome
0	0	99	21.0	0	31	0
1	1	85	28.6	0	31	0
2	0	183	23.3	0	32	1
3	1	89	28.1	94	21	0
4	0	137	31.0	168	33	1
...	...	...	...	...	...	...
763	10	101	28.9	180	63	0
764	2	122	28.8	0	27	0
765	5	121	28.2	112	30	0
766	1	126	30.1	0	47	1
767	1	83	30.4	0	23	0

768 rows × 7 columns

**GAMBAR 2.** Tampilan Dataset

Pada Gambar 2 merupakan tampilan dataset yang dihasilkan dengan melakukan import semua library pada Python yang dibutuhkan untuk mendukung pemrosesan data. Dataset dapat dibaca pada Python dengan disimpan dengan file csv.

### Processing Data

Pada bagian ini dilakukan pemisahan antara fitur (x) dan target (y) dengan memilih kolom-kolom yaitu Pregnancies, Glucose, Blood Pressure, BMI, Insulin, dan Age sebagai fitur, sementara untuk kolom Outcome sebagai label.

```
Jumlah Pasien berdasarkan Outcome:
Outcome
0    500
1    268
Name: count, dtype: int64

Nilai Mean Fitur:
Pregnancies      3.845052
Glucose          120.894531
BloodPressure    69.105469
BMI              31.992578
Insulin          79.799479
Age              33.240885
dtype: float64

Standar Deviasi Fitur:
Pregnancies      3.369578
Glucose          31.972618
BloodPressure    19.355807
BMI              7.884160
Insulin          115.244002
Age              11.760232
dtype: float64
```

**GAMBAR 3.** Seleksi Fitur dan Eksplorasi Fitur

Berdasarkan output yang tampil pada Gambar 3, diketahui bahwa ada 500 pasien tidak mengidap diabetes (label 0) dan 268 pasien yang terdiagnosis positif diabetes (label 1). Selanjutnya, dilakukan perhitungan nilai rata-rata dari tiap fitur menggunakan X.mean() untuk mengetahui nilai tengah dari masing-masing variabel. Kemudian dihitung standar deviasinya dengan X.std() untuk melihat seberapa besar penyebaran data terhadap nilai rata-ratanya. Kemudian dilanjutkan dengan normalisasi fitur yang dapat dilihat pada gambar 4.

	Pregnancies	Glucose	BloodPressure	BMI	Insulin	Age
0	0.839947	0.848324	0.140641	0.204013	-0.692891	1.425995
1	-0.844865	-1.123396	-0.160546	-0.684422	-0.692891	-0.190672
2	1.233600	1.843724	-0.263941	-1.103255	-0.692891	-0.105584
3	-0.844865	-0.098208	-0.160546	-0.494043	0.123302	-1.041549
4	-1.141652	0.904056	-1.804687	1.406746	0.705836	-0.020496
5	0.342981	-0.153185	0.253036	-0.811341	-0.692891	-0.275780
6	-0.250952	-1.342478	-0.967710	-0.128977	0.071204	-0.616111
7	1.827813	-0.164482	-3.572597	0.419775	-0.692891	-0.360847
8	-0.547919	2.361884	0.046245	-0.189437	4.021922	1.681259
9	1.233880	0.128489	1.390387	-4.060474	-0.692891	1.788348

**GAMBAR 4.** Hasil Normalisasi Data

Pada proses ini, dilakukan normalisasi fitur dengan tujuan untuk menyamakan skala fitur agar model tidak bingung dengan fitur yang memiliki perbedaan nilai yang spesifik. Rumus perhitungan ini menggunakan Z-score.

### Penerapan Algoritma KNN

Penerapan algoritma K-Nearest Neighbors menggunakan bahasa pemrograman Python yang dapat dilihat pada Gambar 5.

```

# Split data
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.2, random_state=0)

# Latih model KNN
knn = KNeighborsClassifier(n_neighbors=8)
knn.fit(X_train, y_train)
# Prediksi
y_pred = knn.predict(X_test)

# Evaluasi
print("== Evaluasi Model KNN ==")
print("Classification Report:\n", classification_report(y_test, y_pred, zero_division=0))

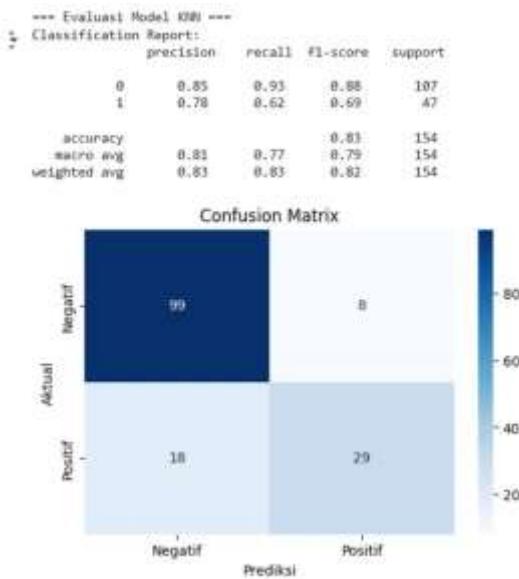
# Confusion matrix
cm = confusion_matrix(y_test, y_pred)
plt.figure(figsize=(6, 4))
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', xticklabels=['Negatif', 'Positif'], yticklabels=['Negatif', 'Positif'])
plt.xlabel('Prediksi')
plt.ylabel('Aktual')
plt.title('Confusion Matrix')
plt.show()

```

**GAMBAR 5.** Latih Model KNN

Dari Gambar 5 dapat dilihat bahwa pada kode “split data” digunakan untuk membagi data menjadi 2, x\_train untuk fitur data latih, x\_test untuk fitur data uji, y\_train untuk label data latih, dan y\_test untuk label data uji. test\_size=0.2 adalah 20% dari data digunakan untuk pengujian, sedangkan 80% digunakan untuk pelatihan. Dan random\_state=0 digunakan agar pembagian data konsisten setiap kali dijalankan. Kemudian pada kode “latih model K-Nearest Neighbors, Prediksi” menggunakan KNeighborsClassifier(n\_neighbors=8) untuk membuat model K-Nearest Neighbors dengan 8 tetangga terdekat dilanjut kode .fit(x\_train, y\_train) untuk mrlatih model dengan data pelatihan. Dan pada kode “prediksi” model K-Nearest Neighbors yang telah dilatih digunakan untuk memprediksi label dari x\_test, yang hasilnya disimpan di y\_pred. Kemudian pada kode “evaluasi” digunakan untuk menampilkan hasil evaluasi model menggunakan precision, recall dan F1-score. Dilanjutkan dengan kode pada “confusion Matrix” yang mana kode ini digunakan untuk membuat confusion matrix dalam bentuk grafik : true positive, true negatives, false positives, false negatives. Sns.heatmap digunakan untuk menampilkan matriks sebagai heatmap (peta warna).

## Evaluasi Model



**GAMBAR 6.** Hasil Evaluasi Model Menggunakan Confusion Matrix

Pada Gambar 6 menunjukkan hasil confusion matrix, terlihat bahwa model K-Nearest Neighbors memprediksi 99 pasien negatif diabetes dan 29 pasien positif diabetes. Namun, terdapat 8 pasien negatif yang diprediksi sebagai positif, serta 18 pasien positif yang diprediksi sebagai negatif. Model ini mencatat akurasi sebesar 83% yang menandakan bahwa model cukup baik dalam memprediksi data. Untuk nilai precision, kelas negatif (0) menunjukkan angka tertinggi yaitu 0.85, sedangkan pada kelas positif (1) nilainya sedikit lebih rendah, yaitu 0.78.

## Deployment

Pada saat melakukan deployment ini menggunakan streamlit, diperlukan beberapa file yang telah dibuat pada proses pengembangan model. Beberapa file tersebut diunggah ke GitHub dan dihubungkan dengan aplikasi streamlit.



**GAMBAR 7.** Hasil Deployment

Pada Gambar 7 diujicobakan dengan memberikan data terhadap fitur yang ada dan sekaligus dapat dilihat hasil prediksi untuk positif atau negatif diabetes.

## KESIMPULAN

Model yang telah dibuat pada penelitian ini kemudian telah di-deploy ke dalam bentuk prediksi berbasis web, menggunakan streamlit dari Python. Hal ini menunjukkan bahwa model dapat diimplementasikan dalam sistem nyata dan digunakan secara langsung oleh pengguna untuk melakukan deteksi dini diabetes secara digital secara fleksibel yang dapat dilakukan dari mana pun. Pada Penelitian ini juga menunjukkan bahwa algoritma K-Nearest Neighbors dapat digunakan untuk melakukan deteksi potensi diabetes berdasarkan data medis pasien. Dengan menggunakan dataset yang didapat dari Kaggle dengan beberapa informasi yang ada, model K-Nearest Neighbors mampu melakukan klasifikasi pasien ke dalam kategori positif atau negatif diabetes. Proses preprocessing data seperti seleksi fitur dan normalisasi data menggunakan Z-score kemudian hasil implementasi model menggunakan nilai  $k = 8$ . Evaluasi model menggunakan metrik akurasi yang menghasilkan 0.83 atau 83%, presisi (0.78), recall (0.61) dan F1-score (0.69). Model juga diimplementasikan secara interaktif menggunakan Jupyter Notebook serta penggunaan Streamlit sebagai user interface sehingga memungkinkan pengguna dapat melakukan konsultasi dengan memasukkan data medis dan sekaligus mendapatkan hasil prediksi. Sistem yang telah dibentuk pada penelitian ini dapat mengalami pengembangan yang bersifat fleksibel dan dapat dikembangkan lebih lanjut dengan integrasi fitur tambahan seperti visualisasi data, riwayat pasien, atau penggabungan dengan model machine learning lainnya untuk meningkatkan akurasi dan interpretabilitas.

## TINJAUAN PUSTAKA

- [1] Asmarani *et al.*, “Implementasi Algoritma K-Nearst Neighbor Untuk Memprediksi Penyakit Diabetes,” *J. Inform. Dan Rekayasa Komputer(JAKAKOM)*, vol. 2, no. 2, pp. 231–239, 2022, doi: 10.33998/jakakom.2022.2.2.110.
- [2] I. M. Karo Karo and H. Hendriyana, “Klasifikasi Penderita Diabetes menggunakan Algoritma Machine Learning dan Z-Score,” *J. Teknol. Terpadu*, vol. 8, no. 2, pp. 94–99, 2022, doi: 10.54914/jtt.v8i2.564.
- [3] M. Saputra, J. P. Sidabuke, R. P. Sinulingga, and R. B. Tamba, “Analisis Metode Algoritma K-Nearest Neighbor (KNN) dan Naive Bayes Untuk Klasifikasi Diabetes Mellitus,” *J. TEKINKOM*, vol. 6, no. 2, pp. 723–729, 2023, doi: 10.37600/tekinkom.v6i2.942.
- [4] N. Azizah, M. Riyad Firdaus, R. Suyaningsih, and F. Indrayatna, “Penerapan Algoritma Klasifikasi K-Nearest Neighbor pada Penyakit Diabetes,” *Pros. Semin. Nas. Stat. Aktuaria*, vol. 2, no. 1, pp. 119–126, 2023, [Online]. Available: [https://prosidingnsa.statistics.unpad.ac.id/?journal=prosidingnsa&page=article&op=view&h\[\]](https://prosidingnsa.statistics.unpad.ac.id/?journal=prosidingnsa&page=article&op=view&h[])344
- [5] L. U. Hasanah, R. Putri Natasya, and V. D. Utami, “Penerapan Algoritma K-Nearest Neighbor (Knn) Untuk Diagnosis Penyakit Diabetes Melitus,” *J. Ilmu Komput.*, vol. 8, no. 1, pp. 86–89, 2024.
- [6] A. Gunawan and I. Fenriana, “Design of Diabetes Prediction Application Using K-Nearest Neighbor Algorithm,” *bit-Tech*, vol. 6, no. 2, pp. 110–117, 2023, doi: 10.32877/bt.v6i2.939.
- [7] H. A. Dwi Fasnuari, H. Yuana, and M. T. Chulkamdi, “Penerapan Algoritma K-Nearest Neighbor Untuk Klasifikasi Penyakit Diabetes Melitus,” *Antivirus J. Ilm. Tek. Inform.*, vol. 16, no. 2, pp. 133–142, 2022, doi: 10.35457/antivirus.v16i2.2445.
- [8] Aswin Ardiansyah, Enos C.O.Telaumbanua, Aron S. Gultom, and Angelita A. S. M. Limbong, “Klasifikasi Penyakit Diabetes Menggunakan Metode SVM Dan KNN,” *J. Penelit. Rumpun Ilmu Tek.*, vol. 3, no. 1, pp. 77–83, Dec. 2023, doi: 10.55606/juprit.v3i1.3151.
- [9] T. A. Fortuna, H. Karuniawati, D. Purnamasari, and D. E. Purlinda, “Faktor – Faktor yang Mempengaruhi Komplikasi pada Pasien Diabetes Mellitus di RSUD Dr. Moewardi,” *Pharmacon J. Farm. Indones.*, vol. 20, no. 1, pp. 27–35, Jul. 2023, doi: 10.23917/pharmacon.v20i1.21877
- [10] Yustika R., & Nugroho A. (2020). Pengantar Machine Learning: Teori dan Implementasi Python. Yogyakarta: Deepublish.