



Pengembangan Sistem Prediksi Risiko Diabetes Menggunakan Algoritma *Support Vector Machine (SVM)*

Alfon Setiawan¹, Suhirman²

Informatika, Fakultas Sains dan Teknologi, Universitas Teknologi Yogyakarta¹
Program Pascasarjana, Teknologi Informasi, Universitas Teknologi Yogyakarta²

Alfonaja123@gmail.com¹ Suhirman@uty.ac.id²

Abstract

Diabetes mellitus is a chronic metabolic disorder characterized by elevated blood glucose levels due to impaired insulin production or utilization. Early detection of diabetes is essential to prevent long-term complications. This study focuses on developing a web-based diabetes risk prediction system using the Support Vector Machine (SVM) algorithm. The dataset used is the Pima Indians Diabetes dataset obtained from Kaggle consists 768 records and eight medical parameters including glucose level, blood pressure, body mass index (BMI), insulin, and age. The system was developed using Python and Flask without applying StandardScaler or SMOTE to maintain the originality of data distribution. Testing and Validation were carried out using separate training and testing datasets. The results show that the SVM algorithm achieved high accuracy in classifying diabetes risk and successfully provided early prediction output through an interactive web interface. This study concludes that the developed system can serve as an effective decision-support tool for healthcare professionals and give contribution to machine learning application in digital health diagnosis.

Keyword: Diabetes Mellitus, Support Vector Machine, Prediction System, Machine Learning, Health Informatics

Abstrak

Diabetes melitus merupakan gangguan metabolismik kronis yang ditandai dengan meningkatnya kadar glukosa darah akibat gangguan pada produksi atau fungsi insulin. Deteksi dini penyakit ini sangat penting untuk mencegah komplikasi jangka panjang. Penelitian ini berfokus pada pengembangan sistem prediksi risiko diabetes berbasis web menggunakan algoritma *Support Vector Machine (SVM)*. Dataset yang digunakan adalah *Pima Indians Diabetes* yang diperoleh dari Kaggle, terdiri dari 768 data dengan delapan parameter medis seperti kadar glukosa, tekanan darah, indeks massa tubuh (BMI), insulin, dan usia. Sistem dikembangkan menggunakan bahasa pemrograman *Python* dan *framework Flask* tanpa penerapan *StandardScaler* maupun *SMOTE* untuk menjaga keaslian distribusi data. Pengujian dilakukan dengan memisahkan data latih dan data uji untuk *Validasi* performa model. Hasil penelitian menunjukkan bahwa algoritma SVM mampu mencapai tingkat akurasi tinggi dalam klasifikasi risiko diabetes serta memberikan hasil prediksi secara interaktif melalui antarmuka web. Penelitian ini menyimpulkan bahwa sistem yang dikembangkan dapat menjadi alat bantu pengambilan keputusan yang efektif bagi tenaga medis dan berkontribusi pada penerapan *machine learning* dalam diagnosis kesehatan digital.

Katakunci: Diabetes Melitus, *Support Vector Machine*, Sistem Prediksi, Pembelajaran Mesin, Informatika Kesehatan

© 2025 Author
Creative Commons Attribution 4.0 International License



1. Pendahuluan

Diabetes melitus merupakan penyakit metabolisme kronis yang ditandai oleh peningkatan kadar glukosa darah akibat gangguan sekresi maupun kerja hormon insulin [1]. Kondisi ini dapat menimbulkan berbagai komplikasi jangka panjang, seperti gangguan kardiovaskular, kerusakan ginjal, gangguan penglihatan, hingga masalah pada sistem saraf [2]. Tingginya prevalensi diabetes secara global menuntut adanya pendekatan inovatif untuk mendukung deteksi dini serta manajemen penyakit secara lebih efektif.

Perkembangan teknologi kecerdasan buatan (*Artificial Intelligence/AI*) membuka peluang baru dalam bidang kesehatan, terutama dalam diagnosis dan prediksi penyakit berbasis data medis. Salah satu cabang AI yang banyak digunakan adalah *machine learning*, yaitu metode yang memungkinkan komputer belajar dari data dan melakukan prediksi tanpa pemrograman eksplisit [3]. Dalam dunia medis, *machine learning* membantu mengidentifikasi pola kompleks dalam data pasien yang sulit dianalisis menggunakan metode konvensional.

Salah satu algoritma *machine learning* yang terbukti efektif untuk klasifikasi medis adalah *Support Vector Machine* (SVM). Algoritma ini bekerja dengan menentukan *hyperplane* optimal yang memisahkan kelas data dengan margin maksimal, serta memiliki kemampuan yang baik dalam menangani data berdimensi tinggi dan kompleks [4]. Berbagai penelitian telah menunjukkan kinerja SVM yang unggul dalam prediksi diabetes. Misalnya, pendekatan ensemble learning berbasis SVM mampu mencapai akurasi di atas 80% [5], sementara penelitian lain menunjukkan bahwa SVM yang dikombinasikan dengan *feature selection* dapat meningkatkan akurasi model [6]. Selain itu, juga membuktikan bahwa SVM tetap efektif meskipun tanpa proses normalisasi data yang kompleks [7].

Meskipun demikian, sebagian besar penelitian terdahulu masih berfokus pada analisis model secara teoretis dan belum banyak yang mengembangkan implementasi sistem berbasis web yang dapat digunakan secara langsung oleh pengguna. Penelitian sebelumnya yang mengembangkan sistem pakar berbasis web untuk diagnosis hipertensi menunjukkan bahwa integrasi model komputasional ke dalam aplikasi web mampu meningkatkan aksesibilitas dan kemudahan penggunaan sistem deteksi penyakit [8]. Selain itu, penelitian Prastiko dan Wiranata menunjukkan bahwa algoritma SVM dapat diimplementasikan secara efektif dalam aplikasi berbasis web untuk analisis data secara real-time, sehingga relevan untuk dikembangkan pada konteks prediksi medis [9].

Penelitian terbaru oleh juga menunjukkan bahwa SVM yang dioptimalkan menggunakan *Grid Search* dan dianalisis dengan metode SHAP mampu menghasilkan performa prediksi diabetes yang lebih akurat dan interpretatif, sehingga menegaskan bahwa SVM masih menjadi algoritma yang kompetitif untuk klasifikasi medis modern [11]. Di sisi lain, penelitian membuktikan bahwa SVM menghasilkan akurasi lebih tinggi dibandingkan K-NN dalam analisis sentimen berbasis Twitter, sehingga memperkuat bahwa SVM konsisten memberikan kinerja klasifikasi yang baik pada berbagai domain [11].

Meskipun penelitian terkait penggunaan SVM cukup banyak, sebagian besar masih berfokus pada aspek teoretis dan pengujian model tanpa mengintegrasikannya ke dalam sistem berbasis web yang dapat diakses langsung oleh pengguna. Penelitian Supriatna et al. juga menunjukkan bahwa integrasi model *machine learning* ke dalam platform digital dapat meningkatkan aksesibilitas dan kemudahan penggunaan, seperti pada pengembangan sistem prediksi cuaca berbasis LSTM dan GRU di platform web [12].

Berdasarkan celah penelitian tersebut, penelitian ini bertujuan untuk mengembangkan sistem prediksi risiko diabetes berbasis web menggunakan algoritma SVM tanpa penerapan *StandardScaler* maupun *SMOTE*. Sistem dibangun menggunakan bahasa pemrograman *Python* dan *framework Flask* dengan memanfaatkan dataset Pima Indians Diabetes yang berisi delapan parameter medis utama seperti kadar glukosa, tekanan darah, indeks massa tubuh (BMI), kadar insulin, dan usia.

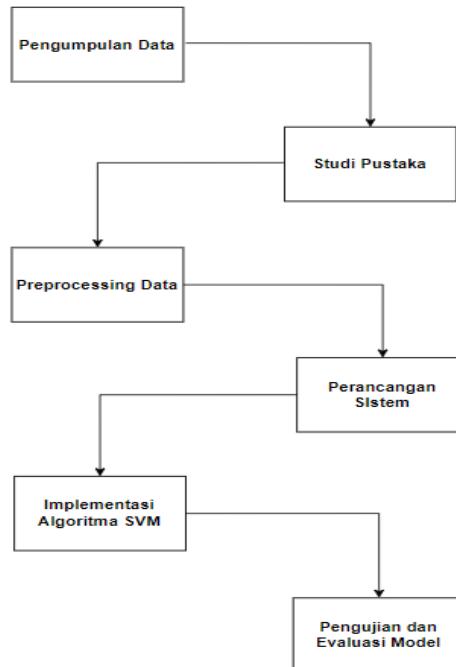
Penelitian ini diharapkan mampu menghasilkan sistem prediksi yang akurat, efisien, serta mudah diakses oleh pengguna umum maupun tenaga medis. Selain itu, penelitian ini diharapkan memberikan kontribusi terhadap pengembangan teknologi *machine learning* di bidang *health informatics* guna mendukung deteksi dini dan pengambilan keputusan berbasis data dalam layanan kesehatan digital.

2. Metode Penelitian

Penelitian ini menggunakan metode pendekatan kuantitatif dengan teknik eksperimen komputasional (*computational experiment*). Metode ini dipilih karena penelitian berfokus pada penerapan algoritma *machine*

learning untuk menghasilkan sistem prediksi berbasis data medis yang terukur dan dapat diuji secara objektif. Pendekatan kuantitatif digunakan untuk menganalisis data numerik pada dataset *Pima Indians Diabetes*, sedangkan eksperimen komputasional diterapkan untuk membangun, melatih, dan menguji model prediksi menggunakan algoritma *Support Vector Machine* (SVM).

Metode penelitian ini juga menggabungkan unsur rekayasa perangkat lunak (*software engineering*), karena hasil akhirnya berupa sistem berbasis web yang dapat digunakan oleh pengguna untuk melakukan prediksi risiko diabetes. Setiap tahap penelitian dilakukan secara terstruktur mulai dari pengumpulan data, pra-pemrosesan, pelatihan model, pengujian performa, hingga implementasi model ke dalam aplikasi web interaktif berbasis Python–Flask. Secara garis besar, langkah-langkah dalam penelitian ini terdiri atas:



Gambar 1. Rancangan Penelitian

2.1 Desain Penelitian

Penelitian ini menggunakan metode eksperimen dengan pendekatan kuantitatif yang bertujuan untuk membangun, menguji, dan mengimplementasikan sistem prediksi risiko diabetes berbasis algoritma *Support Vector Machine* (SVM). Pendekatan kuantitatif dipilih karena seluruh variabel dalam dataset bersifat numerik dan dapat dianalisis menggunakan teknik statistik serta evaluasi model klasifikasi. Pemilihan metode eksperimen dilakukan untuk mengamati secara langsung bagaimana perubahan komposisi data dan parameter model memengaruhi performa prediksi yang dihasilkan.

Desain penelitian terdiri dari beberapa tahapan utama yang saling berhubungan. Tahap pertama adalah analisis masalah serta studi literatur untuk mengidentifikasi kebutuhan sistem, memahami karakteristik penyakit diabetes, serta meninjau algoritma dan teknologi yang relevan, termasuk penelitian terdahulu terkait implementasi SVM pada data kesehatan. Tahap kedua adalah pengumpulan dan pemahaman dataset *Pima Indians Diabetes*, termasuk pemeriksaan struktur data, atribut medis yang tersedia, serta karakter distribusi data. Pada tahap ini dilakukan pula pengecekan nilai ekstrem maupun ketidakseimbangan kelas untuk memastikan kesiapan data sebelum pelatihan model.

Tahap ketiga adalah proses pelatihan model SVM. Data dibagi menggunakan proporsi 80% sebagai data training dan 20% sebagai data testing untuk mengukur kemampuan generalisasi model secara objektif. Model SVM dilatih tanpa menggunakan teknik normalisasi seperti *StandardScaler* maupun metode penyeimbangan kelas seperti *SMOTE* agar nilai medis tetap berada pada skala aslinya. Pemilihan SVM didasarkan pada kemampuan algoritma ini dalam melakukan klasifikasi data non-linear melalui penggunaan *kernel*, serta performanya yang stabil pada

dataset berdimensi kecil hingga menengah. Setelah pelatihan, model dievaluasi menggunakan metrik akurasi untuk menilai efektivitasnya dalam melakukan prediksi.

Tahap keempat adalah implementasi sistem berbasis web. Model yang telah dilatih diintegrasikan ke dalam aplikasi berbasis Flask, memungkinkan sistem untuk menerima masukan berupa parameter medis seperti kadar glukosa, tekanan darah, BMI, dan usia. Sistem kemudian memproses input tersebut menggunakan model SVM dan menampilkan hasil prediksi secara langsung kepada pengguna. Implementasi ini dirancang agar sederhana, mudah diakses, dan mampu menggambarkan pemanfaatan algoritma *machine learning* pada aplikasi kesehatan digital.

Secara keseluruhan, desain penelitian ini memastikan bahwa setiap tahapan berjalan secara terstruktur, mulai dari analisis konsep, pemrosesan data, pelatihan model, evaluasi performa, hingga implementasi aplikasi. Dengan demikian, penelitian ini tidak hanya menghasilkan model prediksi, tetapi juga menunjukkan bagaimana model tersebut dapat digunakan dalam platform berbasis web untuk mendukung proses deteksi dini diabetes secara praktis.

2.2 Dataset Penelitian

Penelitian ini menggunakan dataset Pima Indians Diabetes yang diperoleh dari situs Kaggle (<https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database>). Dataset ini sering digunakan dalam penelitian kesehatan karena berisi data medis nyata dari pasien perempuan keturunan Indian Pima yang berusia di atas 21 tahun. Data ini dikumpulkan oleh *National Institute of Diabetes and Digestive and Kidney Diseases* dan dirancang untuk memprediksi kemungkinan seseorang menderita diabetes berdasarkan sejumlah parameter biologis. Dataset ini terdiri atas 768 baris data (records) dan 9 atribut, yang terdiri dari 8 variabel independen (fitur input) dan 1 variabel dependen (target). Semua variabel dalam dataset bersifat numerik sehingga dapat langsung digunakan oleh algoritma *machine learning* tanpa proses kategorisasi tambahan. Tabel 1 berikut menunjukkan daftar variabel yang digunakan beserta penjelasannya:

Tabel 1. Deskripsi Atribut pada Pima Indians Diabetes Dataset

No	Nama Atribut	Tipe Data	Deskripsi
1	<i>Pregnancies</i>	Numerik	Jumlah kehamilan pasien
2	<i>Glucose</i>	Numerik	Konsentrasi glukosa plasma
3	<i>BloodPressure</i>	Numerik	Tekanan darah diastolik
4	<i>SkinThickness</i>	Numerik	Ketebalan lipatan kulit triceps
5	<i>Insulin</i>	Numerik	Kadar insulin serum
6	<i>BMI(Body Mass Index)</i>	Numerik	Indeks massa tubuh
7	<i>DiabetesPedigreeFunction</i>	Numerik	Indeks riwayat keluarga terhadap diabetes
8	<i>Age</i>	Numerik	Usia pasien
9	<i>Outcome</i>	Numerik Kategorikal	Status diabetes (1 = diabetes, 0 = tidak diabetes)

2.3 Pemisahan Data

Data yang telah diperoleh dibagi menjadi dua bagian, yaitu data pelatihan dan data pengujian. Pembagian dilakukan dengan rasio 80% data untuk pelatihan dan 20% untuk pengujian menggunakan fungsi `train_test_split()` dari pustaka *scikit-learn*. Pemisahan ini bertujuan untuk menguji kemampuan model dalam melakukan generalisasi terhadap data baru yang belum pernah dilihat sebelumnya. Dengan demikian, akurasi yang diperoleh menggambarkan kinerja model yang sebenarnya.

2.4 Algoritma *Support Vector Machine* (SVM)

Algoritma *Support Vector Machine* digunakan sebagai metode klasifikasi untuk menentukan apakah seseorang berisiko menderita diabetes. Prinsip dasar SVM adalah mencari bidang pemisah atau *hyperplane* yang dapat memisahkan dua kelas data dengan jarak maksimal antara keduanya. Data yang berada paling dekat dengan *hyperplane* disebut *support vector* dan berperan penting dalam menentukan posisi batas pemisah.

Dalam penelitian ini, model SVM menggunakan kernel *Radial Basis Function* (RBF) karena *kernel* ini mampu menangani data yang bersifat *non-linear*. Parameter utama yang digunakan adalah $C = 1$ sebagai pengatur regularisasi kesalahan dan $\gamma = \text{'scale'}$ yang menyesuaikan lebar kernel berdasarkan distribusi data. Model dilatih dengan menggunakan fungsi `fit()` dan diuji dengan fungsi `predict()` dari pustaka *scikit-learn*.

2.5 Implementasi Sistem

Model prediksi yang telah dibangun diintegrasikan ke dalam sistem berbasis web menggunakan *framework Flask*. Arsitektur sistem terdiri dari dua bagian utama, yaitu sisi server dan sisi pengguna.

1. Sisi server (*back-end*) mengelola logika aplikasi, proses pelatihan model, prediksi, serta autentikasi pengguna. File utama app.py berfungsi sebagai pengatur rute aplikasi seperti */login*, */register*, */predict*, */train*, dan */test*. Proses prediksi dilakukan dengan membaca input pengguna, mengubahnya menjadi *DataFrame*, lalu memprosesnya menggunakan model SVM yang telah dilatih.
2. Sisi pengguna (*front-end*) dibangun menggunakan HTML5, CSS, *Bootstrap 5*, dan *JavaScript* dengan dukungan AJAX agar hasil prediksi dapat ditampilkan tanpa memuat ulang halaman. Tampilan yang dikembangkan meliputi halaman *login*, registrasi, dashboard utama, halaman prediksi, dan tampilan data pelatihan maupun pengujian. Pengguna yang telah masuk ke sistem dapat menginput data medis dan memperoleh hasil prediksi berupa “Diabetes” atau “Tidak Diabetes”.

2.6 Proses Pelatihan dan Prediksi

Tahapan pelatihan dan prediksi terdiri dari beberapa langkah utama. Pertama, dataset dibaca menggunakan pustaka pandas dan dipisahkan menjadi data input (X) dan output (y). Kedua, model SVM dilatih dengan data pelatihan untuk mengenali pola hubungan antarvariabel. Ketiga, model diuji menggunakan data pengujian untuk menilai tingkat akurasi klasifikasi. Keempat, model yang telah terlatih diintegrasikan ke aplikasi Flask. Ketika pengguna memasukkan delapan parameter medis melalui form pada halaman prediksi, data akan dikirim ke server menggunakan metode POST. Server kemudian memproses data tersebut melalui model dan menghasilkan keluaran berupa prediksi risiko diabetes.

2.7 Evaluasi Model dan Sistem

Evaluasi dilakukan untuk mengukur performa model dan efektivitas sistem. Nilai akurasi dihitung dengan fungsi *accuracy_score()* berdasarkan perbandingan antara hasil prediksi dan data sebenarnya. Selain akurasi, sistem juga dinilai berdasarkan kecepatan pemrosesan, konsistensi hasil, dan kemudahan penggunaan.

Berdasarkan hasil pengujian, model SVM mampu memberikan tingkat akurasi sekitar 70–80 persen pada data uji. Nilai ini menunjukkan bahwa model dapat melakukan klasifikasi dengan cukup baik meskipun tanpa proses normalisasi atau penyeimbangan data tambahan. Sistem juga dapat memberikan hasil prediksi secara real-time dengan waktu pemrosesan singkat, sehingga layak digunakan sebagai alat bantu diagnosis awal.

2.8 Alur Penelitian

Tahapan penelitian secara keseluruhan terdiri atas 6 langkah utama, yaitu:

1. Melakukan pengumpulan dataset.
2. Melakukan studi pustaka
3. Melakukan pra-pemrosesan data atau preprocessing data.
4. Melakukan perancangan sistem
5. Mengimplementasikan ke *algoritma Support Vector Machine*
6. Melakukan pengujian antarmuka dan evaluasi model

Setiap tahap dilakukan secara berurutan untuk memastikan hasil yang diperoleh akurat dan dapat dipertanggungjawabkan secara ilmiah. Hasil akhir dari penelitian ini berupa sistem prediksi berbasis web yang memanfaatkan algoritma SVM untuk membantu deteksi dini risiko diabetes secara efisien dan interaktif.

3. Hasil dan Pembahasan

Penelitian ini bertujuan untuk mengembangkan sistem prediksi penyakit diabetes menggunakan algoritma *Support Vector Machine* (SVM) berbasis web dengan *framework Flask*. Aplikasi ini tidak hanya berfungsi sebagai alat klasifikasi, tetapi juga sebagai media pembelajaran yang interaktif dalam memahami bagaimana model *machine learning* bekerja dalam konteks medis.

3.1 Deskripsi Dataset

Dataset yang digunakan merupakan *Pima Indians Diabetes* Dataset. yang diperoleh dari Kaggle, berisi 768 data pasien dengan 8 variabel prediktor dan satu kolom target Outcome. Kolom Outcome memiliki dua nilai, yaitu 1 untuk pasien yang terdeteksi diabetes dan 0 untuk pasien yang tidak diabetes.1. Spesifikasi Dataset dibagi menjadi dua bagian, yaitu 80% untuk data latih dan 20% untuk data uji, menggunakan metode `train_test_split` tanpa dilakukan proses scaling maupun oversampling. Pemilihan ini dilakukan untuk menguji kemampuan algoritma SVM dalam menangani data mentah secara langsung. Gambar 2 berikut adalah contoh dari 14 sampel data pertama pada *Pima Indians Diabetes* Dataset.

Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
6	98	58	33	190	34.0	0.430	43	0
2	112	75	32	0	35.7	0.148	21	0
2	108	64	0	0	30.8	0.158	21	0
8	107	80	0	0	24.6	0.856	34	0
7	136	90	0	0	29.9	0.210	50	0
6	103	72	32	190	37.7	0.324	55	0
1	71	48	18	76	20.4	0.323	22	0
0	117	0	0	0	33.8	0.932	44	0
4	154	72	29	126	31.3	0.338	37	0
5	147	78	0	0	33.7	0.218	65	0
10	111	70	27	0	27.5	0.141	40	1
7	179	95	31	0	34.2	0.164	60	0
4	148	60	27	318	30.9	0.150	29	1
5	96	74	18	67	33.6	0.997	43	0

Gambar 2. Dataset Penelitian

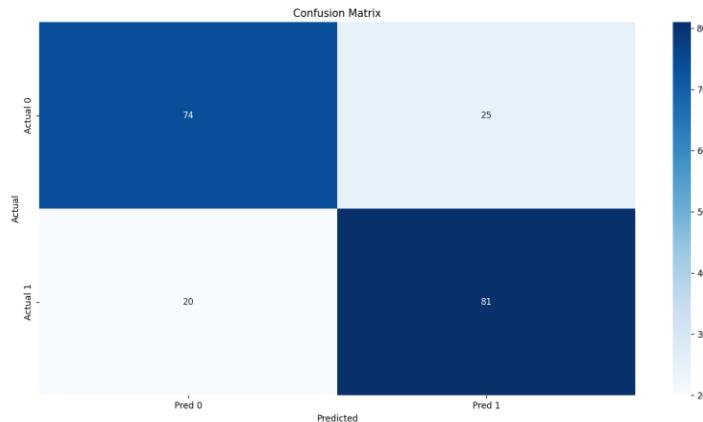
3.2 Pra-pemrosesan data

Pada langkah ini mencakup pemilihan data dan pemilihan atribut- atribut atau *field* yang sudah pengolahan data awal atau *preprocessing* terpilih yang akan digunakan nantinya untuk proses klasifikasi. Pada dataset yang bertujuan untuk membersihkan data agar siap diolah. Dalam proses ini ada 8 fitur yang akan diproses yaitu *pregnancies*, *Glucose*, *BloodPressure*, *SkinThickness*, *Insulin*, *BMI*, *DiabetesPedigreeFunction*, dan *Age*. Data tersebut diolah menggunakan bahasa pemrograman python dengan menggunakan metode *Support Vector Machine* untuk mendapatkan hasil yang optimal

3.3. Pelatihan dan Evaluasi model

Algoritma *Support Vector Machine* (SVM) digunakan sebagai model utama dengan parameter kernel *Radial Basis Function* (RBF). Pemilihan kernel ini didasarkan pada kemampuannya menangani data non-linear dengan baik. Model dilatih menggunakan training set, kemudian diuji menggunakan testing set. Evaluasi dilakukan dengan menghitung akurasi prediksi, yaitu rasio antara jumlah prediksi benar terhadap total data uji. Berdasarkan hasil pengujian, model menghasilkan akurasi sebesar 76,62%. Nilai ini menunjukkan bahwa model memiliki kemampuan klasifikasi yang cukup baik meskipun tidak dilakukan proses scaling maupun resampling data.

Performa model juga diuji menggunakan *confusion matrix* yang menggambarkan jumlah prediksi benar dan salah untuk masing-masing kelas. Pada gambar 3 menunjukkan bahwa model mampu mengenali pasien yang tidak menderita diabetes dengan lebih baik dibandingkan pasien yang positif diabetes, yang dapat disebabkan oleh ketidakseimbangan jumlah data antar kelas.



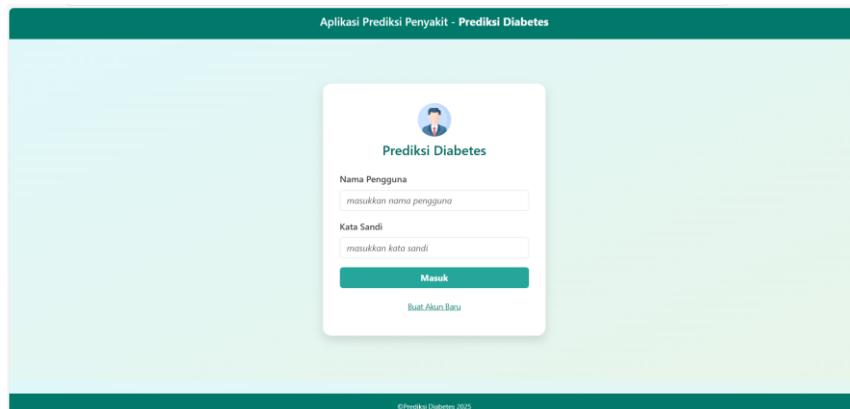
Gambar 3. Confusion Matrix

3.4 Implementasi Sistem

Sistem ini diimplementasikan dalam bentuk aplikasi web menggunakan framework Flask dengan bahasa pemrograman Python. Struktur utama aplikasi mencakup :

1. Login.html untuk autentikasi pengguna

Halaman login pada aplikasi prediksi diabetes bias diakses . Pada tampilan login pengguna harus memasukkan nama pengguna dan password dan jika benar maka akan masuk ke dalam aplikasi tersebut. Halaman login dapat dilihat pada Gambar 4.



Gambar 4. Halaman Login

2. Index.html sebagai dashboard utama

Halaman dashboard pada aplikasi prediksi diabetes dapat diakses setelah pengguna berhasil melakukan proses *login*. Pada halaman ini terdapat ucapan selamat datang, informasi tingkat akurasi model, serta beberapa menu utama seperti Data Train, Data Test, Prediksi, dan Keluar. Halaman dashboard ditunjukkan pada Gambar 5.



Gambar 5. Halaman Dashboard

3. train.html dan test.html untuk menampilkan data latih dan uji

Halaman Data Train dan Data Test berfungsi menampilkan sebagian data yang digunakan dalam proses pelatihan dan pengujian model. Saat pengguna memilih menu tersebut, sistem akan menampilkan 20 data acak dari masing-masing dataset dalam bentuk tabel. Fitur ini membantu pengguna memahami data yang digunakan oleh model sebelum proses prediksi dilakukan. halaman Data Train dapat dilihat pada Gambar 6 dan Data Test pada Gambar 7.

Data Training									
Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome	
60	2	84	0	0	0.0	0.304	21	0	
618	9	112	82	24	0	28.2	1282	50	1
346	1	139	46	19	83	28.7	0.654	22	0
294	0	161	50	0	0	21.9	0.254	65	0
231	6	134	80	37	370	46.2	0.238	46	1
340	1	130	70	13	105	25.9	0.472	22	0
535	4	132	0	0	0	32.9	0.302	23	1
306	10	161	68	23	132	25.5	0.326	47	1
718	1	108	60	46	178	35.5	0.415	24	0
90	1	80	55	0	0	19.1	0.258	21	0
377	1	87	60	37	75	37.2	0.509	22	0

Gambar 6. Halaman Data Train

Data Testing									
Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome	
668	6	98	58	33	190	34.0	0.430	43	0
324	2	112	75	32	0	35.7	0.148	21	0
624	2	108	64	0	0	30.8	0.158	21	0
690	8	107	80	0	0	24.6	0.856	34	0
473	7	136	90	0	0	29.9	0.210	50	0
204	6	103	72	32	190	37.7	0.324	55	0
97	1	71	48	18	76	20.4	0.323	22	0
336	0	117	0	0	0	33.8	0.932	44	0
568	4	154	72	29	126	31.3	0.338	37	0
148	5	147	78	0	0	33.7	0.218	65	0
667	10	111	70	27	0	27.5	0.141	40	1
212	7	179	95	31	0	34.2	0.164	60	0
100	4	148	60	27	318	30.0	0.160	20	1

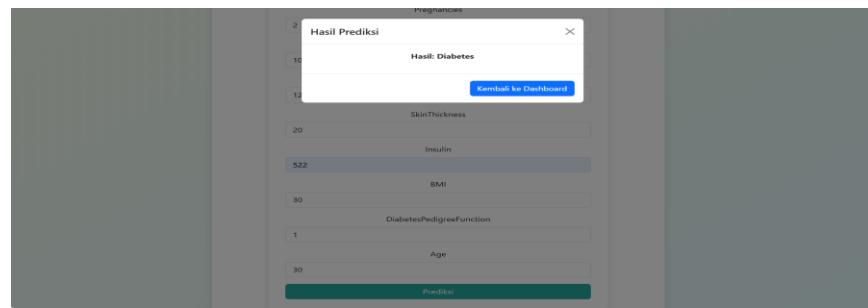
Gambar 7. Halaman Data Test

4. predict.html untuk melakukan prediksi dengan input manual

Halaman prediksi dapat lihat setelah pengguna mengklik button prediksi dan aplikasi akan menampilkan prediksi dan pengguna akan diminta menginput data baru untuk apakah pengguna tersebut terdiagnosa diabetes atau tidak oleh sistem. Pengguna diminta memasukan data-data seperti *Pregnancies* (kehamilan), *Glucose* (glukosa), *Bloodpressure* (tekanan darah), *SkinThickness* (ketebalan kulit), *Insulin*, *BMI* (*Body*

Mass Index), DPF (Diabetes Pedigree Function), Age (umur). Setelah pengguna memasukan data baru aplikasi akan memprediksi pengguna tersebut didiagnosis diabetes atau tidak, setelah itu pengguna dapat menginput ulang data atau kembali ke halaman dashboard. Halaman prediksi dapat dilihat pada Gambar 8 dan hasil prediksi pada Gambar 9.

Gambar 8. Halaman Prediksi



Gambar 9. Halaman Hasil Prediksi

3.5 Pembahasan hasil

Pembahasan hasil penelitian ini menggunakan *black box testing* yang berguna untuk pengujian atau mengamati hasil *input* dan *output* dari perangkat lunak sudah berjalan sesuai dengan keinginan atau belum. *Black box testing* digunakan untuk pengujian *login*, pengujian *dashboard*, pengujian fitur deteksi. Pengujian *black box testing* dapat dilihat sebagai berikut.

1. Pengujian Halaman *Login*

Pada halaman login penulis melakukan beberapa pengujian aktivitas *login* dan *register*. Tahap pengujian dapat dilihat pada Tabel 2.

Tabel 2. Pengujian Halaman *Login*

No	Aktivitas Pengujian	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1.	<i>Username</i> dan <i>password</i> tidak diisi	<i>Username (null)</i> dan <i>password (null)</i>	Sistem tidak dapat melanjutkan ke halaman <i>dashboard</i> dan memunculkan notifikasi "Masukan <i>Username</i> atau <i>Password</i> "	Sesuai harapan	Valid
2.	<i>Username</i> benar dan <i>password</i> salah	<i>Username (admin)</i> dan <i>password (false)</i>	Sistem tidak dapat melanjutkan ke halaman <i>dashboard</i> dan memunculkan notifikasi "Username atau <i>Password</i> salah"	Sesuai harapan	Valid
3.	<i>Username</i> salah dan <i>password</i> benar	<i>Username (false)</i> dan <i>password (admin)</i>	Sistem tidak dapat melanjutkan ke halaman <i>dashboard</i> dan memunculkan notifikasi "Username atau <i>Password</i> salah"	Sesuai harapan	Valid
4.	<i>Username</i> benar dan <i>password</i> benar	<i>Username (admin)</i> dan <i>password (admin)</i>	Sistem dapat melanjutkan ke halaman <i>dashboard</i>	Sesuai harapan	Valid

2. Pengujian Halaman *Dashboard*

Pada halaman dashboard penulis melakukan pengujian terhadap beberapa fitur yang tersedia. Tahap pengujian fitur dapat dilihat pada Tabel 3.

Tabel 3. Pengujian Halaman *Dashboard*

No	Aktivitas Pengujian	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1.	Masuk ke fitur lihat data <i>train</i>	Klik button <i>Train</i>	Sistem dapat memperlihat data <i>train</i> pada dataset <i>diabetes.csv</i> yang telah dilatih dengan model SVM	Sesuai harapan	Valid
2.	Masuk ke fitur lihat data <i>test</i>	Klik button <i>Test</i>	Sistem dapat memperlihat data <i>test</i> pada dataset <i>diabetes.csv</i> yang telah diuji dengan model SVM	Sesuai harapan	Valid
3.	Masuk ke fitur deteksi diabetes	Klik button deteksi	Sistem meminta pengguna menginput data baru untuk prediksi apakah terdiagnosa diabetes atau tidak terdeteksi diabetes, setelah itu sistem memberikan pilihan untuk input ulang atau kembali ke halaman <i>dashboard</i>	Sesuai harapan	Valid
4.	Masuk ke fitur keluar	Klik button keluar	Sistem kembali ke halaman login dan meminta pengguna memasukan <i>username</i> dan <i>password</i> kembali	Sesuai harapan	Valid

3. Pengujian Fitur Deteksi Diabetes

Pada fitur deteksi diabetes penulis mencoba melakukan input data baru untuk mendeteksi diagnosis diabetes. Tahap pengujian fitur deteksi diabetes dapat dilihat pada Tabel 4.

Tabel 4. Pengujian Fitur Deteksi Diabetes

No	Aktivitas Pengujian	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1.	Data tidak ada atau dikosongkan	Data(<i>null</i>)	Sistem tidak memprediksi apapun dan memberikan notifikasi "Data Wajib diisi"	Sesuai harapan	Valid
2.	Data hanya diisi beberapa	Data(<i>null</i>)	Sistem tidak memprediksi apapun dan memberikan notifikasi "Data Wajib diisi"	Sesuai harapan	Valid
3.	Data diisi semua tetapi tidak sesuai	Data(<i>False</i>)	Sistem tidak memprediksi apapun dan pada data yang diisi dengan tidak sesuai sistem akan memunculkan notifikasi"	Sesuai harapan	Valid
4.	Data diisi semua dan sesuai	Data(<i>True</i>)	Sistem akan memprediksi apakah data yang sudah masukan terdiagnosa diabetes atau tidak, jika terdeteksi diabetes sistem akan memunculkan notifikasi "Diabetes" dan jika tidak sistem akan memunculkan notifikasi "Tidak Diabetes"	Sesuai harapan	Valid

3.6 Pembahasan

Hasil penelitian menunjukkan bahwa algoritma *Support Vector Machine* (SVM) yang dilatih tanpa proses *preprocessing* tambahan seperti normalisasi atau *resampling* masih mampu mencapai akurasi sebesar 76,62%. Nilai ini menunjukkan bahwa SVM tetap memiliki kemampuan klasifikasi yang cukup baik meskipun menggunakan data numerik mentah. Namun, evaluasi lebih lanjut memperlihatkan bahwa model cenderung bias terhadap kelas mayoritas, yaitu kategori non-diabetes, sehingga sensitivitas terhadap kelas minoritas belum optimal. Kondisi ini sejalan dengan karakteristik dataset Pima Indians Diabetes yang memang memiliki distribusi kelas tidak seimbang. Oleh karena itu, penelitian selanjutnya disarankan untuk mempertimbangkan teknik

penyeimbangan data seperti *Synthetic Minority Over-sampling Technique* (SMOTE) atau *class weighting* guna meningkatkan kemampuan model dalam mengenali pasien dengan risiko diabetes. Selain itu, analisis terhadap kontribusi fitur menunjukkan bahwa variabel *glucose* dan *BMI* memberikan pengaruh terbesar dalam proses klasifikasi, diikuti oleh *age* dan *insulin*. Pola ini konsisten dengan literatur medis yang menyatakan bahwa kadar glukosa darah dan indeks massa tubuh merupakan indikator utama yang berhubungan erat dengan risiko diabetes. Implementasi model ke dalam aplikasi berbasis web juga memberikan nilai tambah karena memungkinkan pengguna untuk memasukkan data secara langsung dan memperoleh prediksi secara real-time, sehingga sistem tidak hanya berfungsi sebagai model penelitian, tetapi juga sebagai media edukasi dan simulasi. Integrasi antara model SVM dan aplikasi web ini membuktikan bahwa teknologi *machine learning* dapat diadaptasi secara praktis dalam mendukung deteksi dini diabetes melalui platform digital yang sederhana, efisien, dan mudah digunakan.

4. Kesimpulan

Penelitian ini menghasilkan sistem prediksi risiko diabetes berbasis web dengan algoritma *Support Vector Machine* (SVM) menggunakan bahasa pemrograman Python dan *framework* Flask. Model SVM yang dibangun tanpa proses *preprocessing* seperti *StandardScaler* maupun *SMOTE* mampu mencapai tingkat akurasi sebesar 76,62%. Nilai tersebut menunjukkan bahwa SVM masih dapat bekerja cukup baik pada data numerik mentah meskipun tanpa penyesuaian skala atau penyeimbangan kelas. Sistem yang dikembangkan juga telah terintegrasi dengan antarmuka berbasis HTML5 dan *Bootstrap 5*, sehingga memungkinkan pengguna untuk melakukan prediksi secara langsung dan interaktif melalui web. Secara keseluruhan, penelitian ini membuktikan bahwa penerapan *machine learning* dapat diimplementasikan secara efektif dalam sistem prediksi kesehatan digital yang mudah diakses dan digunakan oleh masyarakat umum. Untuk penelitian selanjutnya, disarankan penerapan metode penyeimbangan data seperti *SMOTE* atau *class weighting* agar sensitivitas model terhadap kelas minoritas dapat meningkat dan hasil prediksi menjadi lebih representatif.

Daftar Rujukan

- [1] American Diabetes Association. (2020). Classification and Diagnosis of Diabetes. *Diabetes Care*, 43(Supplement 1), S14–S31. <https://doi.org/10.2337/dc20-S002>
- [2] Han, L., Luo, S., Yu, J., Pan, L., & Chen, S. (2020). Rule Extraction from Support Vector Machines Using Ensemble Learning Approach: Application to Diabetes Diagnosis. *BioMed Research International*, 2020, 1–10. <https://doi.org/10.1155/2020/4817130>
- [3] Sisodia, D., & Sisodia, D.S. (2018). Prediction of Diabetes Using Classification Algorithms. *Procedia Computer Science*, 132, 1578–1585. <https://doi.org/10.1016/j.procs.2018.05.122>
- [4] Ramya, R., & Radha, N. (2016). Diagnosis of Diabetes Mellitus Using Classification Mining Techniques. *International Journal of Innovative Research in Computer and Communication Engineering*, 4(3), 1689–1696.
- [5] Kaur, H., & Kumari, V. (2020). Predictive Modelling and Analytics for Diabetes Using a Machine Learning Approach. *Applied Computing and Informatics*, 18(1), 90–100. <https://doi.org/10.1016/j.aci.2019.11.004>
- [6] Rahman, A., Islam, M. T., & Siddique, N. (2021). Diabetes Prediction by Support Vector Machine and Random Forest Classifiers Using Pima Indians Dataset. *International Journal of Advanced Computer Science and Applications (IJACSA)*, 12(4), 62–70. <https://doi.org/10.14569/IJACSA.2021.0120410>
- [7] Choudhury, A., & Kumar, S. (2022). Machine Learning Approach for Early Detection of Diabetes Mellitus. *Journal of King Saud University - Computer and Information Sciences*, 34(9), 6813–6824. <https://doi.org/10.1016/j.jksuci.2020.04.001>
- [8] Trilaksono, W., Nopriyeni, & Utama, A. P. (2020). Sistem Pakar Diagnosa Penyakit Hipertensi Menggunakan Metode Forward Chaining Berbasis Web. *Jurnal Teknologi dan Sistem Informasi*, 6(3), 210–219.
- [9] Prastiko, A. D., & Wiranata, A. D. (2023). Analisis Sentimen Publik terhadap Fenomena Judi Online di Media Sosial X dengan SVM. *Jurnal Pustaka AI*, 8(2), 140–149. <https://doi.org/10.55382/jurnalpustakaai.v5i2.1180>
- [10] Safii, M., Husain, H., & Marzuki, K. (2025). *Support Vector Machine Optimization for Diabetes Prediction Using Grid Integrated with SHapley Additive exPlanations*. *MATRIK*, 5(2), 119–131. <https://journal.universitasbumigora.ac.id/matrik/article/view/5133>
- [11] Siswanto, D., Zamzami, Z., Nijal, L., & Rajab, S. (2022). Analisa Sentimen Publik Mengenai Perekonomian Indonesia Pada Masa Pandemi Covid-19 di Twitter Menggunakan Metode Klasifikasi K-NN dan SVM. *Jurnal Pustaka AI*, 2(1), 01–09.
- [12] Supriatna, D., Anggai, S., & Tukiyat. (2025). Analisis Prediksi Curah Hujan di Kota Tangerang Menggunakan Metode LSTM dan GRU. *Jurnal Pustaka AI*, 5(2), 119–131. <https://doi.org/10.55382/jurnalpustakaai.v5i2.1068>