Perbandingan RFE dan SelectKbest untuk Klasifikasi Penyakit Diabetes dengan Random Forest

Gede Brandon Abelio Ogaden[a1], Ida Bagus Gede Dwidasmara[a2]

Program Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam,
Universitas Udayana
Jalan Raya Kampus UNUD, Bukit Jimbaran, Kuta Selatan, Badung, Bali, Indonesia
[1]brandon.ogaden@gmail.com
[2]dwidasmara@unud.ac.id

Abstract

Diabetes is a condition that happens in our metabolic system characterized by high level of blood sugar or known as hyperglycemia. Hyperglycemia can either be caused by auto immune insulin destruction problems or insulin resistance in the body. According to World Health Organization, nearly 350 million people suffers from diabetes. Several unwanted side effects can occur from diabetes such as blindness, amputation, and kidney failures if they aren't aware of the disease. Sadly, not many people know the dangers of diabetes. Therefore, a machine that can accurately and efficiently classify diabetes from its symptoms is our top priorities. On this research SelectKBest feature selection when paired with Random Forest Algorithm is fairly accurate at classifying and predicting diabetes with accuracy and recall value of 0.72 each.

Keywords: *Diabetes, Machine Learning, Recursive Feature Elimination, KBest Feature Selection, Classification, Random Forest*

1. Pendahuluan

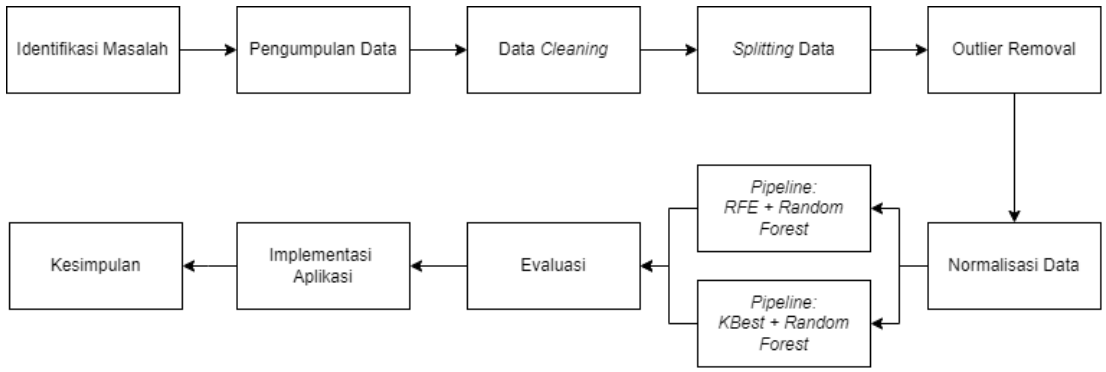
Diabetes adalah penyakit metabolik yang terjadi akibat pengaruh kondisi hiperglikemia di dalam tubuh. Hiperglikemia ditandai dengan kadar gula darah di dalam tubuh lebih tinggi dari kadar normal. Hiperglikemia dapat terjadi oleh satu dua hal, diantaranya gejala autoimun dimana sistem kekebalan tubuh menghancurkan sel penghasil insulin (diabetes tipe 1) atau gejala resistensi insulin dimana insulin tidak dapat digunakan secara normal (diabetes tipe 2) [1].

Penyakit diabetes merupakan salah satu penyakit kronis mematikan yang penderitanya terbanyak di seluruh kalangan dunia. WHO (*World Health Organization*) melaporkan bahwa penderita penyakit diabetes didunia mendekati jumlah 350 juta orang [2]. Bahaya yang berakar dari penyakit diabetes meliputi kebutaan, gagal ginjal, hingga stroke dan serangan jantung. Klasifikasi penyakit diabetes sangat penting untuk penanganan yang efektif, dengan mengidentifikasi gejala-gejala yang muncul.

Akurasi mesin yang baik memerlukan data yang baik juga. Diperlukannya data yang bersih, lengkap, dan berbobot merupakan hal yang krusial dalam pembuatan mesin klasifikasi penyakit diabetes. Pengolahan data yang baik harus dapat diterapkan untuk meminimalisasi terjadinya kesalahan dalam proses klasifikasi sehingga mendapatkan hasil yang maksimal.

Dalam penelitian ini, akan dilakukan proses klasifikasi penyakit diabetes menggunakan algoritma *Random Forest.* Pada penelitian ini peneliti bertujuan untuk membandingkan hasil dari dua algoritma *feature selection* yaitu RFE dan SelectKbest yang dimana hasilnya akan dijalankan melalui *Random Forest* dan hasilnya akan dievaluasi menggunakan *metrics Confusion Matrix.*

2. Metode Penelitian



Gambar 1. Alur Penelitian

Penelitian ini dilaksanakan dengan identifikasi permasalahan dilanjutkan dengan proses pengumpulan *dataset* yang akan digunakan. *Dataset* akan diolah dalam proses *pre-processing* menggunakan *library pandas.* Tahapan *preprocessing* terdiri dari *data cleaning, splitting* data*, outlier removal,* dan normalisasi data. Selanjutnya data akan masuk ke dalam dua *pipeline* seleksi fitur dan *machine learning model*. Akan dilakukan dua cara seleksi fitur yang berbeda yaitu RFE (*Recursive Feature Elimination)* dan *SelectKbest*. Model mesin klasifikasi dengan algoritma *Random Forest* akan menganalisis data dan mencari hubungan antara data dengan *outcome* penyakit diabetes. Nilai yang telah diprediksi dari kedua kasus percobaan akan dievaluasi dan dianalisis.

2.1. Identifikasi Masalah

Peneliti mengidentifikasikan sebuah permasalahan nyata yang ada di dunia melalui beberapa kajian tulisan seperti artikel dan penelitian yang telah dilaksanakan sebelum sebelumnya. Masalah yang diangkat oleh peneliti yaitu banyaknya korban jiwa yang disebabkan oleh penyakit diabetes. Pasalnya efek samping seperti kebutaan, amputasi, hingga gagal ginjal diakibatkan karena masyarakat dunia memiliki kesadaran yang kurang akan bahayanya penyakit diabetes ini [2]. Diperlukannya sebuah mesin yang akurat dan efisien dalam mengklasifikasikan penyakit diabetes menjadi sebuah kepentingan.

2.2. Pengumpulan Data

Data pada penelitian ini merupakan data tipe sekunder yang diambil dari halaman *website kaggle*. *Dataset* berjumlah 70.692 data dengan 17 atribut fitur dan 1 atribut target. *Dataset* berisi informasi *record* kesehatan pasien yang diambil oleh *Behavioral Risk Factor Surveillance System* (BRFSS) melalui sebuah survey tahunan pada tahun 2015. Data mentah yang digunakan dalam penelitian ini dapat dilihat pada Tabel 1, dan keterangan dari setiap atribut dapat dilihat pada Tabel 2.

Tabel 1. *Dataset*

Age Category	**Sex**	**High Cholesterol**	**Cholesterol Check**	**BMI**	**Smoker**	**Heart Disease**	**Physical Activity**	**Fruits**
4.0	1.0	0.0	1.0	26.0	0.0	0.0	1.0	0.0
12.0	1.0	1.0	1.0	26.0	1.0	0.0	0.0	1.0
13.0	1.0	0.0	1.0	26.0	0.0	0.0	1.0	1.0
11.0	1.0	1.0	1.0	28.0	1.0	0.0	1.0	1.0
8.0	0.0	0.0	1.0	29.0	1.0	0.0	1.0	1.0
Veggies	**Heavy**	**General**	**Mental**	**Physical**	**Difficult**	**Stroke**	**HighBP**	**Diabetes**

p-ISSN: 2986-3929
e-ISSN: 3032-1948

Age Category	Sex	High Cholesterol	Cholesterol Check	BMI	Smoker	Heart Disease	Physical Activity	Fruits
	Alcohol	Health	Health	Health	Walking			
1.0	0.0	3.0	5.0	30.0	0.0	0.0	1.0	0.0
0.0	0.0	3.0	0.0	0.0	0.0	1.0	1.0	0.0
1.0	0.0	1.0	0.0	10.0	0.0	0.0	0.0	0.0
1.0	0.0	3.0	0.0	3.0	0.0	0.0	1.0	0.0
1.0	0.0	2.0	0.0	0.0	0.0	0.0	0.0	0.0

Tabel 2. Keterangan *Dataset*

Atribut	Jenis Data	Keterangan Data
Age Category	Numerikal	Kategori Umur dari pasien, terdiri dari 13 level kategori. Level 1= 18-24, dst
Sex	Numerikal	Gender pasien. 1: laki laki, 0: perempuan
High Cholesterol	Numerikal	Tingkat kolesterol pasien. 1: kolesterol tinggi, 0: tidak kolesterol tinggi
Cholesterol Check	Numerikal	Pasien yang telah melakukan cek kolesterol dalam jangka waktu 5 tahun terakhir. 1: telah cek kolesterol, 0: belum cek kolesterol
BMI	Numerikal	*Body Mass Index*. Indeks lemak tubuh yang dihitung berdasarkan tinggi dan berat badan.
Smoker	Numerikal	Riwayat merokok pasien, setidaknya 100 batang rokok dalam seumur hidup. 1: iya, 0: tidak
Heart Disease	Numerikal	Riwayat penyakit jantung *coronary heart disease* (CHD) atau *myocardial infarction* (MI). 1: iya, 0: tidak
Physical Activity	Numerikal	Apakah pasien melakukan aktivitas fisik dalam jangka waktu 30 hari terakhir. 1: iya, 0: tidak
Fruits	Numerikal	Apakah pasien mengonsumsi 1 atau lebih buah buahan per harinya. 1: iya, 0: tidak
Veggies	Numerikal	Apakah pasien mengonsumsi 1 atau lebih sayur sayuran per harinya. 1: iya, 0: tidak
Heavy Alcohol	Numerikal	Apakah pasien mengonsumsi alkohol dalam jumlah banyak tiap minggunya. 14 botol per minggu untuk pria, 7 botol per minggu untuk wanita. 1: iya, 0: tidak
General Health	Numerikal	Tingkat kesehatan secara general menurut pasien. Skala 1-5, 1 = *excellent,* 2 = *very good,* 3 = *good,* 4 = *fair,* 5 = *poor*
Mental Health	Numerikal	Jumlah hari dengan kondisi mental buruk. Skala 1 – 30 hari
Physical Health	Numerikal	Jumlah hari dengan penyakit fisik atau kecelakaan. Skala 1 – 30 hari
Difficult Walking	Numerikal	Apakah pasien memiliki kesulitan dalam berjalan atau menaiki anak tangga. 1: iya, 0: tidak
Stroke	Numerikal	Apakah pasien memiliki riwayat penyakit stroke. 1: iya, 0: tidak
HighBP	Numerikal	Tingkat tekanan darah pasien. 1: tekanan darah tinggi, 0: tekanan darah normal

p-ISSN: 2986-3929
e-ISSN: 3032-1948

Atribut	Jenis Data	Keterangan Data
Diabetes	Numerikal	*Outcome* dari survey kesehatan pasien. 1: memiliki diabetes, 0: tidak memiliki diabetes

2.3. Preprocessing

Preprocessing data merujuk pada pengolahan data yang dilakukan sebelum proses utama dilakukan. *Preprocessing* dilakukan untuk mengatasi masalah dan membuat data menjadi mudah untuk diproses atau digunakan. Tujuan dari dilaksanakannya *pre-processing* adalah untuk meningkatkan kualitas data, seperti kelengkapan, konsistensi, dan ketepatan waktu untuk meningkatkan hasil [3]. Pada penelitian ini beberapa tahapan *pre-processing* diantaranya *data cleaning, splitting* data*, outlier removal,* dan normalisasi data*.*

a. *Data Cleaning*

Data cleaning merupakan sebuah proses pembersihan data dari data kosong atau data yang tidak berisikan nilai. Nilai yang kosong dapat diisikan dengan beberapa cara diantaranya dengan menggunakan nilai mean atau rata rata dan median atau nilai tengah. Proses *data cleaning* juga menghilangkan data duplikat yang ada dalam *dataset*.

b. *Splitting* Data

Data splitting merujuk pada pembagian *dataset* menjadi dua bagian yaitu *data training* dan *data testing*. *Data training* akan digunakan sebagai bahan dalam melatih model. *Data training* akan menjadi kunci akan besar kecilnya akurasi dari model. Sedangkan *data testing* akan berperan dalam proses evaluasi sistem. Pada penelitian ini pembagian data dilakukan dengan *data training* sebesar 80% dan *data testing* sebesar sisanya yaitu 20%. Penggalan kode *splitting* data dijabarkan pada Tabel 3.

Tabel 3. Penggalan Kode *Splitting* Data

Baris	Potongan Kode
1	x = df.drop(['Diabetes'],axis = 1)
2	y = df['Diabetes']
3	X_train, X_test, y_train, y_test = train_test_split(x, y, test_size=0.2,
4	random_state=42)
5	X_train['diabetes'] = y_train
	df = X_train.copy()

c. *Outlier Removal*

Outlier removal merupakan sebuah metode penghilangan *data point* yang abnormal. Proses ini menghilangkan data yang tidak sesuai dengan pola, data yang dapat merusak hasil prediksi model. Penggalan kode *outlier removal* dapat dilihat pada Tabel 3.

Tabel 4. Penggalan Kode *Outlier Removal*

Baris	Potongan Kode
1	quantile = QuantileTransformer(output_distribution='normal')
2	df_quantile = quantile.fit_transform(df)
3	df_quantile_hasil = DataFrame(df_quantile)

d. *Min-Max Normalization*

Normalisasi pada data dilakukan untuk menyamakan skala nilai pada atribut ke dalam suatu rentang yang sama. *Min-Max Normalization* merupakan suatu metode transformasi linear dengan mengambil nilai minimum dan maksimum setiap atribut untuk menghasilkan sebuah keseimbangan antar data [4].

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

Keterangan:

data(x) = data hasil setelah dinormalisasi
x = data yang akan dinormalisasikan
x_{min} = nilai terkecil pada data
x_{max} = nilai terbesar pada data

Tabel 5. Penggalan Kode Normalisasi

Baris	Potongan Kode
1	scaler = MinMaxScaler()
2	df_normalization = scaler.fit_transform(df_quantile_hasil)
3	df_normalization_hasil = DataFrame(df_normalization.round(2))

Dengan dilakukannya normalisasi berakhirlah proses *preprocessing*. Hasil akhir proses *preprocessing* dapat dilihat Tabel 6.

Tabel 6. Hasil *Min-Max Normalization*

Age Category	Sex	High Cholesterol	Cholesterol Check	BMI	Smoker	Heart Disease	Physical Activity	Fruits
0.58	1.0	0.0	1.0	0.43	0.0	0.0	1.0	0.0
0.58	0.0	1.0	1.0	0.47	1.0	1.0	0.0	0.0
0.47	0.0	0.0	1.0	0.47	1.0	0.0	1.0	0.0
0.61	0.0	1.0	1.0	0.43	0.0	0.0	1.0	0.0
0.39	0.0	1.0	1.0	0.45	1.0	0.0	1.0	0.0
Veggies	**Heavy Alcohol**	**General Health**	**Mental Health**	**Physical Health**	**Difficult Walking**	**Stroke**	**HighBP**	**Diabetes**
1.0	0.0	0.59	0.00	0.55	0.0	0.0	1.0	1.0
0.0	0.0	1.00	0.00	1.00	1.0	0.0	1.0	1.0
0.0	0.0	0.59	0.61	0.61	1.0	0.0	1.0	0.0
0.0	0.0	0.51	0.00	0.00	0.0	0.0	0.0	1.0
0.0	1.0	0.51	0.00	0.53	1.0	0.0	1.0	0.0

2.4. Feature Selection

a. RFE (*Recursive Feature Elimination*)
RFE atau *Recursive Featuer Elimination* merupakan sebuah metode penyeleksian fitur yang dapat mengetahui fitur fitur utama dalam sebuah *dataset*. Proses dilakukan dengan melakukan algoritma rekursif untuk menghilangkan atribut yang paling tidak memiliki pengaruh terhadap target. Pengulangan akan terus dilakukan hingga jumlah fitur yang

diinginkan telah tercapai. Penggalan kode pengaplikasian seleksi fitur RFE dapat dilihat pada Tabel 7.

Tabel 7. Penggalan Kode RFE

Baris	Potongan Kode
1	n = 6 #Jumlah fitur yang ingin di ekstraksi
2	rfe = RFE(estimator=RandomForestClassifier(), n_features_to_select=n)

b. *SelectKBest Feature Selection*
SelectKBest merupakan sebuah metode seleksi fitur berbasis filter. *SelectKBest* mengandalkan statistik, mengukur dan menghitung skor dari fitur pada dataset berdasarkan hubungannya dengan variabel target. Penggalan kode pengaplikasian seleksi fitur *KBest* dapat dilihat pada Tabel 8.

Tabel 8. Penggalan Kode *KBest*

Baris	Potongan Kode
1	n = 6 #Jumlah fitur yang ingin di ekstraksi
2	kbest = SelectKBest(score_func=chi2, k=n)

2.5. Random Forest

Random forest adalah sekumpulan *decision tree* yang dikembangkan untuk mengklasifikasi berdasarkan pemilihan atribut yang dipilih secara acak untuk setiap node [5]. Algoritma *random forest* menggabungkan pendapat dari banyak *tree* untuk membuat prediksi yang lebih akurat. *Random forest* adalah metode klasifikasi yng tingkat keakuratannya sangat tinggi, biasa digunakan dalam model prediksi, dan dapat menangani input data yang sangat besar tanpa *overfitting* [6]. Penggalan kode *random forest* dapat dilihat pada Tabel 9.

Tabel 9. Penggalan Kode *Random Forest*

Baris	Potongan Kode
1	model = RandomForestClassifier()
2	model2 = RandomForestClassifier()
3	pipe = Pipeline([('Feature Selection', rfe), ('Model', model)])
4	pipe2 = Pipeline([('Feature Selection', kbest), ('Model', model2)])

2.6. Evaluasi

Pada penelitian ini, dalam proses evaluasi diterapkan metode *confusion matrix* sebagai sarana dalam mengukur ketepatan hasil klasifikasi yang dihasilkan oleh model *random forest* yang telah dipadankan dengan masing masing seleksi fitur.

Tabel 10. *Confusion Matrix*

	Prediksi Diabetes	Prediksi Tidak Diabetes
Benar	TP	FN
Salah	FP	TN

Dengan *confusion matrix,* dapat ditarik perhitungan yang dapat mengukur performa model yang telah dibuat. Perhitungan tersebut antara lain akurasi, *recall, precision* dan *F1 score.*

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

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

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

$$F1\ score = \frac{2*precision*recall}{precision+recall} \quad (5)$$

Keterangan:

TP = *True Positives* (jumlah model memprediksi pasien memiliki penyakit diabetes dan benar)
TN = *True Negatives* (jumlah model memprediksi pasien tidak memiliki penyakit diabetes dan benar)
FP = *False Positives* (jumlah model memprediksi pasien memiliki penyakit diabetes tapi salah)
FN = *False Negatives* (jumlah model memprediksi pasien tidak memiliki penyakit diabetes tapi salah)

3. Hasil dan Pembahasan

3.1. Implementasi *Machine Learning*

Penelitian ini membandingkan antara dua algoritma *feature selection,* yaitu *Recursive Feature Elimination* dan *SelectKbest.* Masing masing algoritma seleksi fitur akan dijalankan untuk n jumlah fitur dari range 6 sampai 10 fitur. Hasil pemilihan fitur yang dilakukan oleh masing masing metode dapat terlihat pada Tabel 11.

Tabel 11. Hasil Seleksi Fitur RFE dan KBest

Metode	Age Category	Sex	High Cholesterol	Cholesterol Check	BMI	Smoker	Heart Disease	Physical Activity
RFE 6 fitur	✓				✓			
KBest 6 fitur			✓				✓	
RFE 7 fitur	✓		✓		✓			
KBest 7 fitur			✓				✓	
RFE 8 fitur	✓		✓		✓			
KBest 8 fitur			✓				✓	
RFE 9 fitur	✓		✓		✓	✓		
KBest 9 fitur			✓				✓	✓
RFE 10 fitur	✓		✓		✓	✓		
KBest 10 fitur			✓			✓	✓	✓

p-ISSN: 2986-3929
e-ISSN: 3032-1948

Fruits	Veggies	Heavy Alcohol	General Health	Mental Health	Physical Health	Difficult Walking	Stroke	HighBP
			✓	✓	✓			✓
					✓	✓	✓	✓
			✓	✓	✓			✓
			✓		✓	✓	✓	✓
✓			✓	✓	✓			✓
		✓	✓		✓	✓	✓	✓
✓			✓	✓	✓			✓
		✓	✓		✓	✓	✓	✓
✓			✓	✓	✓	✓		✓
		✓	✓		✓	✓	✓	✓

Pada Tabel 11 terlihat bahwa masing masing algoritma memilih fitur yang berbeda beda yang dianggap lebih penting.

Setelah seleksi fitur dilaksanakan, atribut yang terpilih akan masuk ke dalam model *random forest* yang akan memprediksi penyakit diabetes pasien. Tabel 12 berikut menampilkan *metrics* dari performa model.

Tabel 12. *Metrics* performa

	akurasi	***precision***	***recall***	***F1-score***
RFE 6 fitur	0.71	0.71	0.71	0.71
KBest 6 fitur	0.70	0.70	0.70	0.70
RFE 7 fitur	0.71	0.71	0.71	0.71
KBest 7 fitur	0.72	0.72	0.72	0.72
RFE 8 fitur	0.71	0.71	0.71	0.71
KBest 8 fitur	0.72	0.72	0.72	0.72
RFE 9 fitur	0.71	0.71	0.71	0.71
KBest 9 fitur	0.71	0.72	0.71	0.71
RFE 10 fitur	0.71	0.71	0.71	0.71
KBest 10 fitur	0.71	0.71	0.71	0.71

Pada Tabel 12, dapat dilihat bahwa RFE memiliki akurasi yang konstan pada nilai 0.71 untuk setiap n jumlah fiturnya. Sedangkan KBest memiliki performa akurasi yang naik turun dimana akurasi terbaik didapatkan jika Kbest menyeleksi 8 fitur dengan akurasi sebesar 0.72.

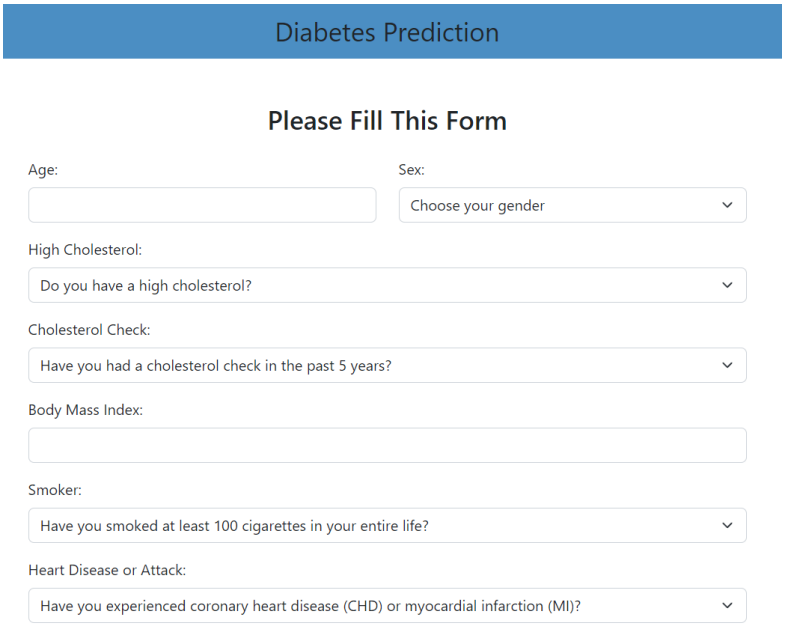
3.2. Implementasi Aplikasi

Aplikasi prediksi penyakit diabetes mengambil inputan *user* dari serangkaian form pertanyaan mengenai data kesehatan pasien yang memiliki korelasi terhadap munculnya penyakit diabetes. Setiap jawaban akan mengisi kolom sesuai pada dataset, sebagai referensi lihat pada Tabel 2.

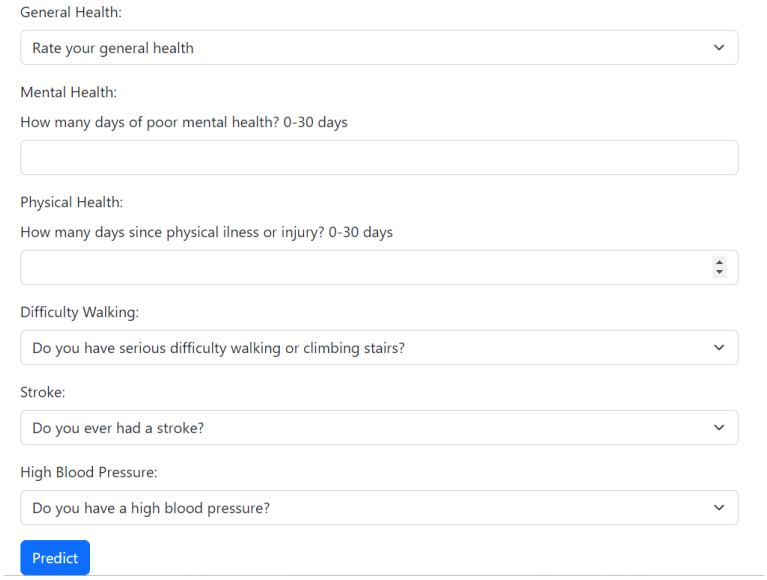
p-ISSN: 2986-3929
e-ISSN: 3032-1948

Jawaban terhadap form akan diteruskan ke dalam model *machine* prediksi diabetes yang telah dibuat, kemudian hasil prediksi akan diteruskan kembali kepada *user*.

Backend aplikasi diimplementasikan dengan bahasa pemrograman *Python* dan *framework Flask* untuk menyimpan dan menjalankan model *machine learning.* Sedangkan *interface* diimplementasikan dengan bahasa pemrograman HTML dengan framework *Bootstrap.* Tampilan formulir aplikasi dapat dilihat pada Gambar 2 dan Gambar 3. Serta tampilan untuk hasil prediksi mesin dapat dilihat pada Gambar 4.

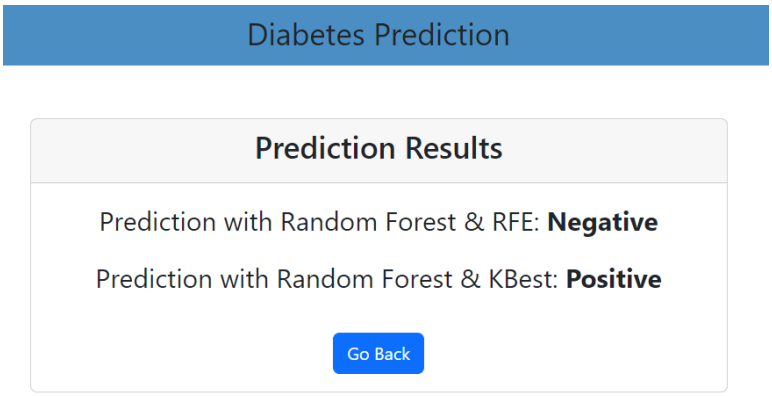


Gambar 2. Formulir Pertanyaan (1)



Gambar 3. Formulir Pertanyaan (2)

p-ISSN: 2986-3929
e-ISSN: 3032-1948



Gambar 4. Hasil Prediksi Aplikasi

4. Kesimpulan

Dari hasil penelitian *selectKBest* merupakan sebuah metode seleksi fitur yang cukup baik digunakan dalam memilih dan melakukan *ranking* terhadap seberapa pentingnya setiap atribut. Untuk klasifikasi penyakit diabetes dengan *random forest* metode *selectKBest* memiliki nilai akurasi terbesar dengan nilai 0.72 dan *recall* sebesar 0.72 dengan jumlah total fitur 8. Sedangkan metode seleksi fitur RFE (*Recursive Feature Elimination*) memiliki nilai akurasi dan *recall* yang tidak berubah pada jumlah fitur 6 sampai dengan 10 dengan nilai masing masing 0.71 dan 0.71.

Daftar Pustaka

[1] "Diabetes Fact Sheets." Accessed: May 10, 2024. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/diabetes
[2] G. Satya Nugraha, M. Nurkholis Abdillah, and M. Innuddin, "Komparasi Akurasi Metode Correlated Naive Bayes Classifier Dan Naive Bayes Classifier Untuk Diagnosis Penyakit Diabetes," *Jurnal Nasional Informatika dan Teknologi Jaringan*.
[3] Gde Agung Brahmana Suryanegara, Adiwijaya, and Mahendra Dwifebri Purbolaksono, "Peningkatan Hasil Klasifikasi pada Algoritma Random Forest untuk Deteksi Pasien Penderita Diabetes Menggunakan Metode Normalisasi," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 1, pp. 114–122, Feb. 2021, doi: 10.29207/resti. v5i1.2880.
[4] P. P. Pratiwi *et al.*, "Akurasi Klasifikasi Kualitas Wine Menggunakan Algoritma Random Forest dengan Min-Max Normalization," *JNATIA*, vol. 2, no. 2, 2024, doi: 10.29207/resti.
[5] I Made Krisna Dwipa Jaya and I Gusti Agung Gede Arya Kadyanan, "Perbandingan Random Forest, Decision Tree, Gradient Boosting, Logistic Regression untuk Klasifikasi Penyakit Jantung," *JNATIA*, vol. 2, pp. 61–69, 2023.
[6] T. N. Nuklianggraita, A. Adiwijaya, and A. Aditsania, "On the Feature Selection of Microarray Data for Cancer Detection based on Random Forest Classifier," *Jurnal Infotel*, vol. 12, no. 3, pp. 89–96, Aug. 2020, doi: 10.20895/infotel. v12i3.485.