Optimasi Prediksi Prediabetes dengan Metode Fitur Selection dan Imbalance Learning

Optimization of Prediabetes Prediction with Feature Selection and Imbalance Learning Methods

Samsul Arifin[1], Imam Tahyudin[2]
[1,2] Jurusan Informatika, Universitas Amikom Purwokerto
E-mail: [1]samsulchin21@gmail.com, [2]imam.tahyudin@amikompurwokerto.ac.id,

Abstrak

Diabetes adalah salah satu tantangan kesehatan global yang terus meningkat, dengan deteksi dini pradiabetes menjadi kunci untuk pencegahan. Data yang digunakan diambil dari Diabetes Health Indicators Dataset dan dipersiapkan melalui tahap feature engineering, analisis korelasi, dan penanganan missing value. Selanjutnya, model dibangun menggunakan tiga algoritma utama, yaitu Random Forest, XGBoost, dan Logistic Regression. Penelitian ini menggabungkan analisis korelasi variabel dan metode imbalance learning untuk mengoptimalkan prediksi pradiabetes menggunakan algoritma machine learning. Untuk menangani ketidakseimbangan data, teknik SMOTE diterapkan guna menghasilkan data sintetik pada kelas minoritas. Hasil penelitian menunjukkan model Random Forest memberikan kinerja terbaik dengan akurasi 97,57%, mengungguli XGBoost dan Logistic Regression. Penerapan analisis korelasi variabel dan imbalance learning terbukti efektif dalam meningkatkan kinerja prediksi dengan identifikasi fitur penting. Penelitian ini menunjukkan bahwa pendekatan yang diterapkan dapat membantu deteksi dini pradiabetes secara lebih akurat dan tepat.

Kata kunci: Diabetes, Deteksi Prediabetes, Machine Learning, Random Forest

Abstract

Diabetes is an increasing global health challenge, with early detection of prediabetes being key to prevention. The data used in this study was sourced from the Diabetes Health Indicators Dataset and prepared through feature engineering, correlation analysis, and handling of missing values. Subsequently, models were developed using three main algorithms: Random Forest, XGBoost, and Logistic Regression. This research combines variable correlation analysis and imbalance learning methods to optimize prediabetes prediction using machine learning algorithms. To address data imbalance, the SMOTE technique was applied to generate synthetic data for the minority class. The results indicate that the Random Forest model performed best, achieving an accuracy of 97.57%, surpassing both XGBoost and Logistic Regression. The implementation of variable correlation analysis and imbalance learning has proven effective in enhancing prediction performance through the identification of important features. This study demonstrates that the applied approach can assist in the more accurate and precise early detection of prediabetes.

Keywords: Diabetes, Prediabetes Detection, Machine Learning, Random Forest

1. PENDAHULUAN

Diabetes merupakan tantangan kesehatan global yang signifikan, dengan prevalensi yang terus meningkat setiap tahunnya. Menurut laporan terbaru dari WHO, diperkirakan lebih dari 10% populasi dunia akan terdiagnosis diabetes pada tahun 2045. Kondisi ini mendorong perlunya pengembangan pendekatan prediksi yang lebih akurat dan tepat waktu guna mendukung identifikasi dini risiko diabetes. Salah satu metode yang saat ini banyak digunakan adalah pembelajaran mesin. Namun, pengembangan model prediksi ini menghadapi tantangan utama

berupa keterpaduan data antara kelas mayoritas (non-diabetes) dan kelas minoritas (diabetes). Kesesuaian ini membuat model cenderung lebih akurat dalam memprediksi kelas mayoritas tetapi kurang efektif pada kelas minoritas, yang dalam konteks medis justru lebih kritis. Oleh karena itu, diperlukan penerapan pendekatan ketidakseimbangan pembelajaran untuk menangani masalah ini, seperti teknik oversampling atau undersampling, guna memperbaiki distribusi kelas sehingga prediksi menjadi lebih seimbang dan akurat. [1].

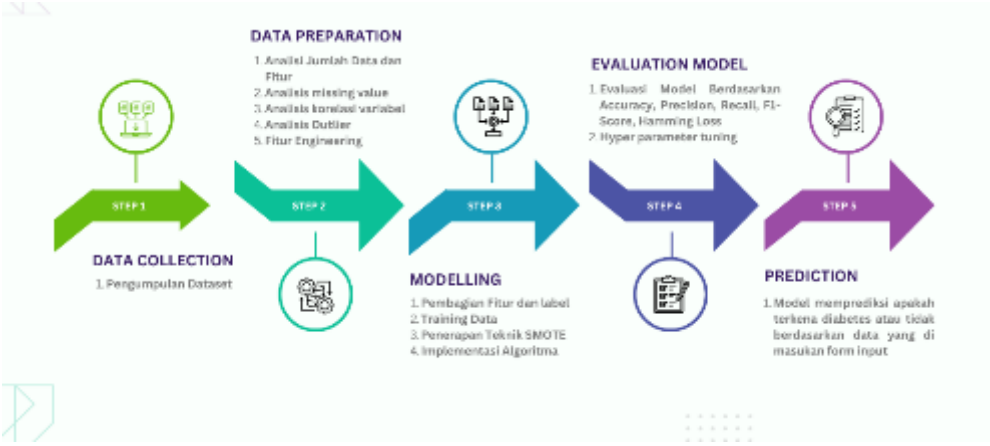
Penelitian sebelumnya telah menunjukkan bahwa algoritma seperti Random Forest, XGBoost, dan Logistic Regression memberikan hasil yang baik dalam prediksi penyakit seperti diabetes. Sebagai contoh, penelitian oleh [2] menunjukkan bahwa XGBoost unggul dalam menangani dataset besar dengan performa prediksi yang tinggi. Namun, penelitian tersebut belum secara spesifik menangani masalah ketidakseimbangan data, yang dapat memengaruhi keakuratan prediksi pada kelas minoritas [3]. Selain itu, penelitian lain oleh [4] menyoroti pentingnya analisis korelasi variabel *(correlation analysis)* untuk memilih fitur yang relevan dan signifikan, yang terbukti dapat meningkatkan akurasi model. Meski demikian, penelitian-penelitian sebelumnya masih memiliki keterbatasan dalam mengintegrasikan pendekatan *imbalance learning* dengan analisis korelasi variabel untuk mengoptimalkan prediksi diabetes [5] [6].

Dalam penelitian ini, diusulkan pendekatan yang lebih komprehensif dengan menggabungkan metode *imbalance learning* dan analisis korelasi variabel untuk meningkatkan kinerja algoritma machine learning, seperti *Random Forest*, *XGBoost*, dan *Logistic Regression*. Teknik penyeimbangan data diterapkan untuk mengurangi bias pada kelas mayoritas, sementara analisis korelasi digunakan untuk mengidentifikasi fitur-fitur penting yang berkontribusi signifikan terhadap hasil prediksi. Dengan pendekatan ini, penelitian bertujuan untuk memberikan kontribusi yang lebih nyata terhadap pengembangan model prediksi berbasis *machine learning* yang lebih akurat dan robust dalam menangani ketidakseimbangan data di konteks medis. Selain itu, penelitian ini juga memperkuat metode yang sebelumnya telah digunakan dengan menerapkan strategi yang lebih holistik untuk mengatasi masalah ketidakseimbangan data sekaligus meningkatkan akurasi prediksi melalui seleksi fitur yang signifikan.

2. METODE PENELITIAN

2.1 Tahapan Penelitian

Berikut ini tahapan-tahapan penelitian yang dilakukan penulis dalam proses prediksi prediabetes menggunakan model *machine learning*. Tahap tersebut meliputi *Data Collection*, *Data Preparation*, *Modelling*, *Evaluation Model* dan *Prediction*. Berikut adalah penjelasan masing-masing tahapan:



Gambar 1 Tahapan Penelitian

2.2 Data Collection

Tahap pertama dalam penelitian ini adalah pengumpulan data, yang mencakup pengumpulan dataset kesehatan yang relevan. Data tersebut mencakup variabel-variabel penting

seperti kadar glukosa darah, tekanan darah, indeks massa tubuh (BMI), usia, serta faktor risiko lainnya yang berhubungan dengan prediabetes. Dataset yang digunakan dalam penelitian ini diperoleh dari *Kaggle*, sebuah platform populer untuk berbagi dataset dan proyek data science. Salah satu dataset yang relevan untuk prediksi prediabetes adalah *Diabetes Health Indicators Dataset*, dengan Alamat URL : https://www.kaggle.com/datasets/alexteboul/diabetes-health-indicators-dataset. Jumlah dataset terdiri 1535 data yang terbagi menjadi 9 fitur. Fitur tersebut *Pregnancies, Glucose, BloodPressure, SkinThickness, Insulin, BMI, DiabetesPedigreeFunction, Age,* dan *Status.* Dataset ini menyediakan berbagai indikator kesehatan yang dapat digunakan untuk memodelkan risiko prediabetes secara akurat [7].

Dataset ini mencakup rentang usia individu antara 21 hingga 75 tahun, yang mencakup populasi dewasa hingga lanjut usia. Sebagian besar data berasal dari populasi di Amerika Serikat, karena dataset ini didasarkan pada studi kesehatan berbasis populasi di wilayah tersebut. Selain itu, dataset memiliki representasi gender yang cukup berimbang, sehingga memberikan gambaran yang lebih akurat dalam analisis risiko prediabetes. Salah satu tantangan dalam dataset ini adalah adanya distribusi kelas yang tidak seimbang, di mana jumlah sampel pada kelas mayoritas (non-diabetes) jauh lebih besar dibandingkan kelas minoritas (diabetes/prediabetes). Ketidakseimbangan ini dapat menyebabkan bias pada model prediksi, di mana model cenderung lebih akurat dalam memprediksi kelas mayoritas tetapi kurang efektif pada kelas minoritas. Oleh karena itu, pendekatan *imbalance learning* akan diterapkan untuk memperbaiki distribusi kelas, sehingga model dapat memberikan prediksi yang lebih akurat dan adil untuk kedua kelas.

2.3 Data Preparation

Tahap *Data Preparation* atau persiapan data merupakan langkah penting dalam memastikan data siap untuk dianalisis atau digunakan dalam model *machine learning*. Langkah pertama dalam persiapan data adalah analisis jumlah data dan fitur, yaitu memeriksa kelengkapan data dan relevansi variabel yang digunakan. Pada tahap ini, penulis mengevaluasi apakah jumlah data mencukupi untuk analisis dan apakah variabel yang ada relevan serta memberikan informasi yang cukup untuk mendukung tujuan penelitian [8].

Langkah kedua adalah analisis *missing value*, di mana penulis menganalisis nilai yang hilang dalam dataset. Data yang tidak lengkap dapat menyebabkan bias dalam analisis, sehingga metode imputasi sering digunakan untuk mengisi nilai yang hilang berdasarkan data yang tersedia, atau dalam beberapa kasus, entri data yang hilang dapat dihapus jika tidak signifikan terhadap keseluruhan dataset. Selanjutnya, penulis melakukan analisis korelasi variabel untuk memahami hubungan antara variabel independen dengan variabel target *(prediabetes* atau *non-prediabetes)*. Analisis ini penting untuk mengetahui fitur mana yang memiliki pengaruh paling besar terhadap prediksi dan mana yang dapat diabaikan atau dikurangi pentingnya dalam model. Variabel yang memiliki korelasi rendah dengan variabel target mungkin dihilangkan atau disederhanakan. Analisis *outlier* juga menjadi bagian penting dalam proses ini. *Outlier* adalah data yang sangat berbeda dari pola umum dalam dataset, dan dapat mempengaruhi hasil analisis jika tidak ditangani dengan benar. Penulis akan mengidentifikasi dan menentukan apakah *outlier* harus dihapus atau ditransformasikan untuk mengurangi dampak negatifnya terhadap model [9].

Terakhir, dilakukan *feature engineering*, yaitu proses menciptakan fitur baru atau mentransformasikan fitur yang ada agar model dapat lebih mudah memahami data. Misalnya, fitur seperti BMI bisa diubah menjadi kategori tertentu untuk meningkatkan interpretabilitas atau interaksi antar fitur tertentu bisa ditambahkan untuk memberikan konteks yang lebih jelas bagi model. Proses *feature engineering* ini sangat penting dalam mengoptimalkan kinerja model dan membantu model menangkap pola yang lebih kompleks dari data [10].

2.4 Modelling

Tahap *Modelling* atau pembuatan model adalah bagian kunci dalam proses machine learning, di mana data yang sudah dipersiapkan digunakan untuk membangun model prediktif. Langkah pertama dalam proses ini adalah pembagian fitur dan label. Dataset dibagi menjadi dua

komponen utama: fitur atau variabel independen, yang mencakup semua faktor atau variabel yang dianggap mempengaruhi hasil, seperti kadar glukosa darah, tekanan darah, dan indeks massa tubuh (BMI); serta label atau variabel dependen, yaitu variabel target yang ingin diprediksi, seperti apakah seseorang mengalami prediabetes atau tidak. Setelah itu, dilakukan pembagian dataset ke dalam *training* data dan *testing* data. Data training digunakan untuk melatih model machine learning, di mana model belajar dari pola-pola dalam data untuk dapat melakukan prediksi. Data *testing*, yang belum pernah dilihat oleh model, nantinya akan digunakan untuk mengevaluasi kinerja model dan seberapa baik model tersebut dapat melakukan generalisasi pada data baru. Untuk menangani ketidakseimbangan data *(imbalance data)*, di mana jumlah sampel kelas minoritas (prediabetes) jauh lebih sedikit dibandingkan dengan kelas mayoritas (non-prediabetes), penulis menerapkan teknik *SMOTE (Synthetic Minority Over-sampling Technique)*. *SMOTE* adalah metode oversampling yang digunakan untuk membuat data sintetik dari sampel kelas minoritas guna menyeimbangkan proporsi kelas dalam dataset. Hal ini membantu model untuk tidak bias terhadap kelas mayoritas dan memberikan prediksi yang lebih akurat untuk kelas minoritas [11].

Selanjutnya, pada tahap implementasi algoritma, berbagai algoritma *machine learning* diterapkan untuk membangun model. Algoritma yang umum digunakan dalam prediksi prediabetes meliputi *Random Forest, XGBoost,* dan *Logistic Regression* [12]. Random Forest adalah algoritma berbasis pohon keputusan yang membangun beberapa pohon dan menggabungkan hasilnya untuk meningkatkan akurasi prediksi [13]. XGBoost, yang merupakan varian dari Gradient Boosting, terkenal karena efisiensi dan kinerja yang tinggi dalam mempelajari pola kompleks [14]. Logistic Regression digunakan untuk masalah klasifikasi biner, di mana model memprediksi kemungkinan terjadinya suatu peristiwa, dalam hal ini adalah apakah seseorang menderita prediabetes atau tidak [15]. Algoritma-algoritma ini dipilih dan dioptimalkan untuk menemukan model yang paling baik dalam memprediksi kondisi prediabetes berdasarkan fitur yang telah disediakan.

2.5 Evaluation Model

Tahap *Evaluation Model* atau evaluasi model merupakan langkah krusial untuk menilai seberapa baik model yang telah dibangun mampu memprediksi prediabetes dengan akurat. Evaluasi dilakukan menggunakan berbagai metrik kinerja seperti *Accuracy*, *Precision*, *Recall*, *F1-Score*, dan *Hamming Loss*. Metrik *Accuracy* mengukur persentase prediksi yang benar dari keseluruhan prediksi yang dibuat oleh model. Namun, karena data yang tidak seimbang (kelas prediabetes lebih sedikit dari non-prediabetes), *Precision*, yang mengukur berapa banyak dari prediksi positif yang benar-benar tepat, dan *Recall,* yang mengukur kemampuan model untuk mendeteksi semua kasus positif, menjadi sangat penting. *F1-Score* digunakan untuk memberikan keseimbangan antara *Precision* dan *Recall*, terutama ketika data tidak seimbang. *Hamming Loss* mengukur frekuensi kesalahan prediksi pada keseluruhan dataset. langkah awal pada tahap *evaluation model* adalah *Hyperparameter Tuning*, yaitu pengaturan parameter pada model untuk meningkatkan performanya. *Hyperparameter Tuning* digunakan untuk menemukan model dengan kinerja terbaik [4]. Setelah *Hyperparameter Tuning,* langkah terakhir mengevaluasi model dengan berdasarkan nilai *Accuracy, Precision, Recall, F1-Score,* dan *Hamming Loss*. Setelah mendapatkan model terbaik model akan di simpan kemudian akan di implementasikan ke sistem prediksi prediabetes.

2.6 Prediction

Tahap terakhir adalah *Prediction.* Model yang telah dioptimalkan dan dievaluasi kemudian digunakan untuk memprediksi apakah seseorang kemungkinan besar mengalami prediabetes atau tidak. Prediksi ini dilakukan berdasarkan data baru yang dimasukkan, baik melalui form input secara real-time maupun melalui dataset lain yang belum digunakan untuk pelatihan. Model ini mampu memberikan prediksi yang lebih akurat berkat pengoptimalan seleksi Fitur dan penerapan *Imbalance Learning* [16].

3. HASIL DAN PEMBAHASAN

Pada tahap *Data Collection*, dataset yang digunakan mempunyai 9 fitur. Gambar 2 menunjukkan jumlah Fitur antara lain *Pregnancies, Glucose, BloodPressure, SkinThickness, Insulin, BMI, DiabetesPedigreeFunction, Age*, dan *Status.*

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Status
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1

Gambar 2 Fitur Dataset

```
Informasi fitur:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1535 entries, 0 to 1534
Data columns (total 9 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   Pregnancies               1535 non-null   int64
 1   Glucose                   1535 non-null   int64
 2   BloodPressure             1535 non-null   int64
 3   SkinThickness             1535 non-null   int64
 4   Insulin                   1535 non-null   int64
 5   BMI                       1535 non-null   float64
 6   DiabetesPedigreeFunction  1535 non-null   float64
 7   Age                       1535 non-null   int64
 8   Status                    1535 non-null   int64
dtypes: float64(2), int64(7)
memory usage: 108.1 KB
None
```

Gambar 3 Jumlah data

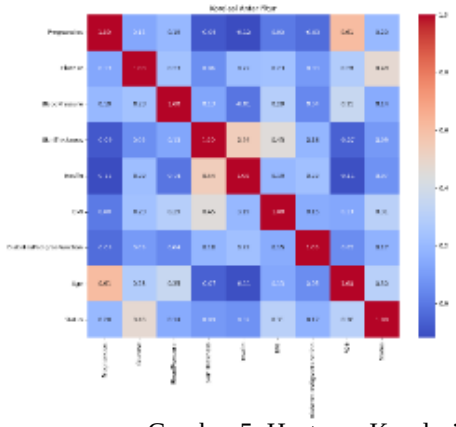
Berdasarkan Gambar 3, data yang digunakan berjumlah 1535 data. Kolom atau fitur yang ada sebanyak 9 kolom, yang terdiri dari tipe data int64 dan float64.

```
# Deteksi Missing Value
missing_values = data.isnull().sum()
print("\nJumlah Missing Value pada Setiap Fitur:")
print(missing_values)
```

```
Jumlah Missing Value pada Setiap Fitur:
Pregnancies                 0
Glucose                     0
BloodPressure               0
SkinThickness               0
Insulin                     0
BMI                         0
DiabetesPedigreeFunction    0
Age                         0
Status                      0
dtype: int64
```

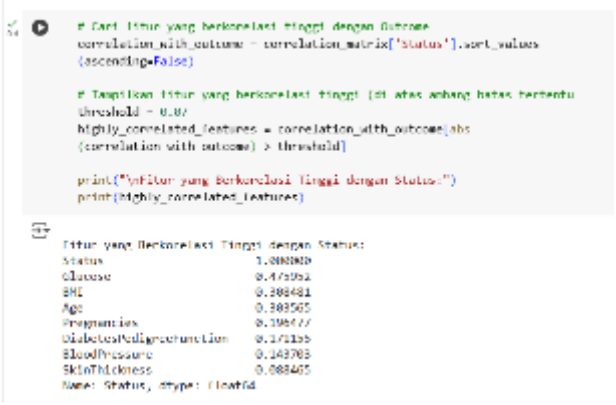
Gambar 4. Deteksi *Missing Value*

Berdasarkan Gambar 4, hasil analisis *missing value* tidak ditemukan *missing value* dalam dataset ini. Setiap fitur, termasuk *Pregnancies*, *Glucose*, *BloodPressure*, *SkinThickness*, *Insulin*, *BMI*, *DiabetesPedigreeFunction*, *Age*, dan *Status*, semuanya memiliki 0 missing value. Artinya, dataset ini sudah bersih dan lengkap, sehingga tidak memerlukan penanganan khusus untuk missing value dan dapat langsung digunakan dalam proses analisis atau pemodelan.



Gambar 5. Heatmap Korelasi

Berdasarkan *heatmap* korelasi yang ditampilkan dalam gambar 5, fitur yang berkorelasi tinggi terhadap class target yaitu *Status* adalah fitur yang berwarna gelap. Secara keseluruhan, dari hasil analisis korelasi ini, *Glucose* memiliki korelasi tertinggi dengan *Outcome* sebesar 0.475952, yang berarti bahwa kadar glukosa merupakan fitur yang paling penting dalam memprediksi apakah seseorang berpotensi mengalami prediabetes atau diabetes. Sedangkan *BMI* memiliki Korelasi sebesar 0.308481 dan *Age* memiliki korelasi sebesar 0.303565 yang lebih rendah tetapi tetap signifikan, yang juga menunjukkan bahwa mereka memiliki kontribusi dalam model prediksi namun tidak sekuat *Glucose.*

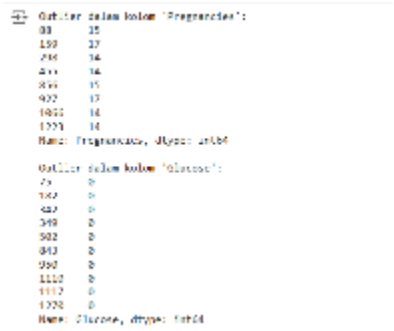


```
# Cari fitur yang berkorelasi tinggi dengan Outcome
correlation_with_outcome = correlation_matrix['Status'].sort_values
(ascending=False)

# Tampilkan fitur yang berkorelasi tinggi (di atas ambang batas tertentu
threshold = 0.07
highly_correlated_features = correlation_with_outcome[abs
(correlation_with_outcome) > threshold]

print("\nFitur yang Berkorelasi Tinggi dengan Status:")
print(highly_correlated_features)
```

```
Fitur yang Berkorelasi Tinggi dengan Status:
Status                      1.000000
Glucose                     0.475952
BMI                         0.308481
Age                         0.303565
Pregnancies                 0.196617
DiabetesPedigreeFunction    0.171195
BloodPressure               0.143703
SkinThickness               0.088405
Name: Status, dtype: float64
```

Gambar 6. Seleksi Fitur

Hasil seleksi fitur pada Gambar 6 menunjukkan bahwa fitur-fitur yang memiliki korelasi tinggi dengan variabel target Status meliputi *Glucose*, *BMI*, *Age*, *Pregnancies*, *DiabetesPedigreeFunction*, *BloodPressure* dan *SkinThickness*, dengan nilai korelasi di atas ambang batas 0.07. Fitur dengan korelasi tertinggi adalah Glucose, diikuti oleh BMI dan Age. Pemilihan fitur-fitur ini didasarkan pada korelasinya yang cukup kuat dengan Status, sehingga diharapkan dapat meningkatkan akurasi dan interpretabilitas model prediktif yang dibangun.



```
Outlier dalam kolom 'Pregnancies':
88     15
159    17
298    14
455    14
456    15
502    17
1001   14
1291   14
Name: Pregnancies, dtype: int64

Outlier dalam kolom 'Glucose':
75     0
182    0
342    0
349    0
502    0
843    0
958    0
1110   0
1112   0
1228   0
Name: Glucose, dtype: int64
```

Gambar 7 Analisis Outlier

Gambar 7 adalah hasil analisis Outlier. Outlier adalah data atau nilai yang sangat berbeda dari sebagian besar data lainnya dalam sebuah dataset. Outlier bisa menyebabkan model statistik atau machine learning menjadi tidak akurat karena mereka bisa mempengaruhi hasil secara tidak proporsional.

Deskripsi Statistik Data Setelah Penanganan Outlier:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Status
0	6.0	148	72	35	0.0	33.6	0.62700	50.0	1
1	1.0	85	66	29	0.0	26.6	0.35100	31.0	0
2	8.0	183	64	0	0.0	23.3	0.67200	32.0	1
3	1.0	89	66	23	94.0	28.1	0.16700	21.0	0
4	0.0	137	40	35	168.0	43.1	1.20875	33.0	1

Gambar 8. Hasil Penanganan *Outlier*

Gambar 8 adaah hasil Penanganan *outlier* dengan *IQR (Interquartile Range)* yang dilakukan dengan cara menghitung jarak antara kuartil pertama (Q1) dan kuartil ketiga (Q3), yang mencakup 50% data tengah. Untuk mendeteksi outlier, nilai-nilai yang terletak di bawah Q1 - 1.5 IQR atau di atas Q3 + 1.5 IQR dianggap sebagai outlier. Setelah outlier terdeteksi, penanganannya dapat dilakukan dengan menghapus outlier tersebut atau mengubahnya (misalnya dengan metode imputasi) agar tidak memengaruhi hasil analisis secara signifikan.

BMI_Category	Glucose_Category	Age_Category	Insulin_Glucose_Ratio	BMI_Age_Interaction
Obese	Prediabetes	Middle-aged	0.000000	1680.0
Overweight	Normal	Middle-aged	0.000000	824.6
Normal	Prediabetes	Middle-aged	0.000000	745.6
Overweight	Normal	Young	1.056180	590.1
Obese	Normal	Middle-aged	1.226277	1422.3

Gambar 9. Feature Engineering

Berdasarkan Gambar 9, Feature engineering adalah proses mengubah dari data mentah menjadi fitur yang lebih bermakna untuk meningkatkan kinerja model machine learning. Berdasarkan gambar yang ditampilkan, hasil *feature engineering* meliputi beberapa fitur baru yang dibuat untuk membantu model memahami pola dalam data. Misalnya, *BMI_Category* dibuat untuk mengklasifikasikan individu berdasarkan kategori indeks massa tubuh seperti *"Obese", "Overweight", atau "Normal"*, sehingga mempermudah model dalam memahami risiko kesehatan terkait obesitas. *Glucose_Category* digunakan untuk mengelompokkan individu berdasarkan kadar glukosa, misalnya, "Normal" atau "Prediabetes", yang penting untuk mendeteksi risiko diabetes. Selain itu, *Age_Category* membagi usia menjadi rentang yang lebih mudah dipahami, seperti *"Young" dan "Middle-aged"*. Ada juga fitur baru seperti *Insulin_Glucose_Ratio*, yang menghitung rasio antara insulin dan glukosa, memberikan informasi tambahan mengenai keseimbangan hormon dan gula darah. Terakhir, *BMI_Age_Interaction* menggabungkan efek dari BMI dan usia, menangkap interaksi antara kedua faktor tersebut yang mungkin mempengaruhi kondisi kesehatan secara lebih kompleks. Secara keseluruhan, fitur-fitur ini dirancang untuk meningkatkan kualitas prediksi model dengan memberikan informasi yang lebih spesifik dan relevan.

```
[12]  from sklearn.model_selection import train_test_split
      from sklearn.ensemble import RandomForestClassifier
      from sklearn.metrics import accuracy_score, classification_report,
      confusion_matrix

      # Memilih fitur yang relevan
      features = ["Glucose", "BMI", "Age", "Pregnancies", "DiabetesPedigreeFunction",
      "BloodPressure", "SkinThickness"]

      X = data[features]
      y = data['Status']
```

Gambar 10 Pembagian Fitur dab label

Berdasarkan Gambar 10 Pada tahap pertama pemodelan, dilakukan pembagian fitur (features) dan label (label). Fitur merupakan variabel independen yang digunakan sebagai input dalam model prediksi, sedangkan label adalah variabel dependen yang menjadi target prediksi.

Dalam penelitian ini, fitur yang dipilih adalah Glucose, BMI, Age, Pregnancies, DiabetesPedigreeFunction, BloodPressure, SkinThickness, yang diidentifikasi sebagai faktor penting dalam memprediksi hasil status prediksi. Sementara itu, fitur Status sendiri dijadikan label, yang mewakili status prediksi, seperti status diabetes dan nondiabetes. Proses ini merupakan langkah awal yang krusial dalam pemodelan untuk memastikan bahwa model dapat dilatih dengan data yang relevan dan akurat.

```
# Split data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42) # Added this line to split the data
```

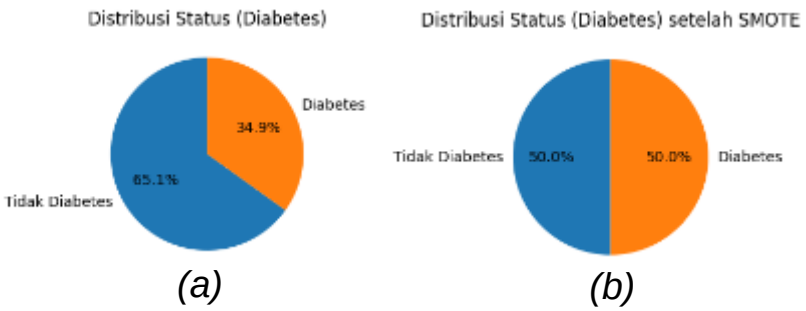
Gambar 11. Training Data

Berdasarkan Gambar 11, langkah selanjutnya dataset dibagi menjadi dua bagian: *Training Set* dan *Testing Set*. *Training set* digunakan untuk melatih model, sementara *testing set* digunakan untuk menguji kinerja model setelah pelatihan. Pada gambar, pembagian ini dilakukan menggunakan fungsi *train_test_split* dari pustaka *scikit-learn* dengan proporsi *test_size=0.2*, yang berarti 20% dari data digunakan untuk pengujian, sedangkan 80% digunakan untuk pelatihan. Parameter random_state=42 digunakan agar pembagian data bersifat *reproducible*, atau dengan kata lain, setiap kali kode dijalankan, pembagian data akan menghasilkan hasil yang sama.

```
# Inisialisasi SMOTE
smote = SMOTE(random_state=42)

# Terapkan SMOTE pada data training
X_train_smote, y_train_smote = smote.fit_resample(X_train, y_train)
```

Gambar 12. Penerapan *SMOTE*

Gamabr 12 adalah proses penerapan *SMOTE (Synthetic Minority Over-sampling Technique)* pada data pelatihan. *SMOTE* adalah teknik yang digunakan untuk menangani masalah *imbalance* pada dataset. Dalam dataset yang tidak seimbang, kelas mayoritas memiliki lebih banyak sampel dibandingkan dengan kelas minoritas, yang dapat menyebabkan model menjadi bias terhadap kelas mayoritas. Untuk mengatasi hal ini, *SMOTE* menghasilkan sampel sintetis untuk kelas minoritas, sehingga distribusi antara kelas mayoritas dan minoritas menjadi lebih seimbang.

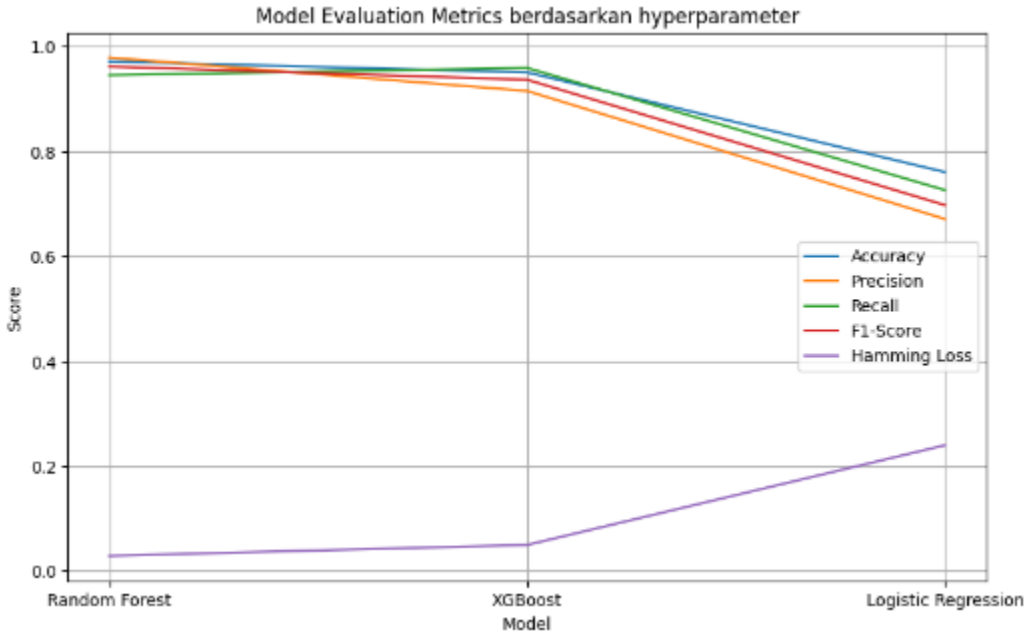


Gambar 13. Distribusi Data (a) Sebelum *SMOTE* dan (b) Setelah *SMOTE*

Berdasarkan Pada Gambar 13 (a), distribusi *Outcome* sebelum diterapkan SMOTE menunjukkan ketidakseimbangan antara kelas Tidak Diabetes dan Diabetes. Kelas Tidak Diabetes mendominasi dengan persentase sebesar 65.1%, sedangkan kelas Diabetes hanya 34.9%. Ketidakseimbangan ini dapat mempengaruhi performa model dalam memprediksi kelas Diabetes, karena model cenderung lebih akurat untuk kelas mayoritas. Setelah diterapkan *SMOTE*, seperti yang terlihat pada Gambar 13 (b), distribusi data menjadi seimbang dengan proporsi 50.0% untuk masing-masing kelas, baik Tidak Diabetes maupun Diabetes. Penerapan *SMOTE* ini membantu dalam menambah sampel sintetis pada kelas minoritas, sehingga model dapat belajar secara lebih merata dari kedua kelas dan tidak bias terhadap kelas mayoritas. Perbandingan antara kedua grafik ini menunjukkan bahwa *SMOTE* berhasil menangani masalah class imbalance dengan

menghasilkan data pelatihan yang lebih seimbang, yang diharapkan dapat meningkatkan performa prediksi model terhadap kelas minoritas (Diabetes).

Tahap selanjutnya adalah penerapan model *machine learning* dengan menggunakan tiga algoritma utama: *Random Forest, XGBoost,* dan *Logistic Regression*. Masing-masing algoritma memiliki karakteristik dan pendekatan berbeda dalam memproses data dan melakukan prediksi.



Gambar 14. Model Evaluation

Berdasarkan Gambar 14. menunjukkan perbandingan metrik evaluasi untuk tiga model yaitu *Random Forest, XGBoost,* dan *Logistic Regression* berdasarkan nilai hyperparameter terbaik yang diperoleh setelah proses tuning. *Random Forest* merupakan model terbaik dalam evaluasi ini karena memiliki nilai tertinggi dalam *Accuracy*, *Precision*, dan *F1-Score*, serta nilai *Hamming Loss* terendah. *XGBoost* juga menunjukkan performa yang baik, terutama pada metrik *Recall*. *Logistic Regression* memiliki performa terendah pada hampir semua metrik, menunjukkan bahwa model ini kurang cocok dibandingkan kedua model lainnya dalam dataset ini.

	Model	Accuracy	Precision	Recall	F1-score	Hamming Loss
0	Random Forest	0.938111	0.935780	0.894737	0.914798	0.061889
1	XGBoost	0.951140	0.923077	0.947368	0.935065	0.048860
2	Logistic Regression	0.762215	0.735632	0.561404	0.636816	0.237785

Gambar 15. Sebelum *SMOTE*

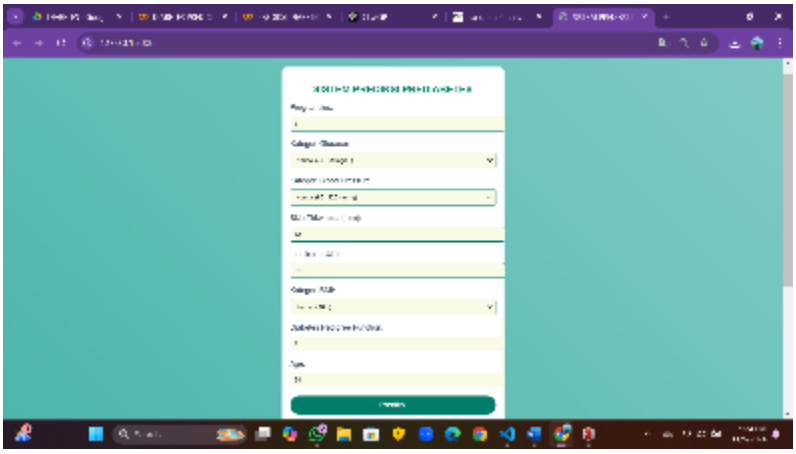
	Model	Accuracy	Precision	Recall	F1-Score	Hamming Loss
0	Random Forest	0.971354	0.978723	0.945205	0.961672	0.028646
1	XGBoost	0.950521	0.915033	0.958904	0.936455	0.049479
2	Logistic Regression	0.760417	0.670886	0.726027	0.697368	0.239583

Gambar 16. Setelah *SMOTE*

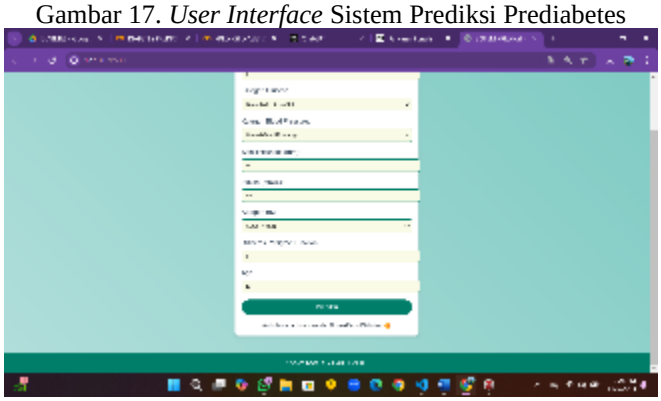
Berdasarkan Gambar 15 dan 16, Hasil evaluasi model sebelum dan sesudah diterapkannya seleksi fitur dan teknik penanganan ketidakseimbangan data (imbalance learning) menunjukkan adanya perubahan performa yang signifikan. Pada algoritma Random Forest, terjadi peningkatan pada semua metrik evaluasi utama, yaitu *Accuracy*, *Precision, Recall,* dan *F1-Score*, sementara nilai *Hamming Loss* menurun drastis. Penurunan *Hamming Loss* ini menandakan bahwa model menjadi lebih akurat dengan tingkat kesalahan yang lebih rendah. Hal ini sesuai

dengan tujuan penelitian untuk meningkatkan kinerja prediksi prediabetes menggunakan pendekatan seleksi fitur dan *imbalance learning*.

Pada evaluasi Model, Random Forest menjadi model dengan akurasi terbaik sebesar 97,13%, sebagaimana ditunjukkan pada Gambar 16. Jika dibandingkan dengan penelitian sebelumnya yang dilakukan oleeh [17] di mana algoritma yang sama hanya mencapai akurasi tertinggi sebesar 74,56%, penelitian ini menunjukkan peningkatan kinerja yang signifikan sebesar 22,57%. Peningkatan ini mengindikasikan bahwa penerapan seleksi fitur yang relevan dan teknik penyeimbangan data memberikan dampak positif yang besar terhadap kemampuan model untuk memprediksi data dengan lebih akurat, terutama pada kelas minoritas (prediabetes/diabetes). Setelah tahap Evaluasi dan di dapatkan model terbaik selanjutnya disimpan dan langkah terakhir model akan di implementasikan untuk membangun sistem prediksi prediabetes berbasis web. Gambar 17 adalah tampilan *User Interface* sistem prediksi prediabetes.

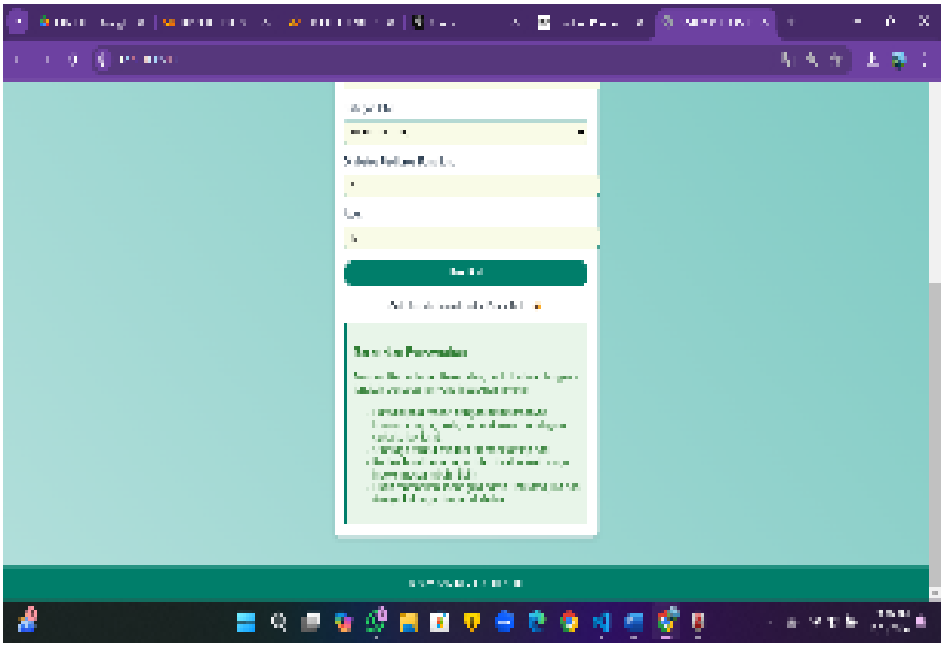


Gambar 17. *User Interface* Sistem Prediksi Prediabetes



Gambar 18. Hasil Prediksi Normal Non Diabetes

Gambar 18 Menunjukkan Ketika pengguna memasukkan data yang tergolong dalam kondisi "Normal," sistem akan menampilkan hasil prediksi dengan pesan "Anda berada dalam kondisi Normal/non-Diabetes" di bawah tombol Prediksi. Hal ini berarti bahwa kombinasi input data yang dimasukkan pengguna, seperti kategori Glucose, Blood Pressure, BMI, dan lainnya, tidak menunjukkan faktor risiko yang signifikan untuk prediabetes atau diabetes.



Gambar 19. Hasil prediksi terkena Prediabetes

Gambar 19 menunjukkan hasil bahwa pengguna berkemungkinan terkena diabetes. Ini berarti, berdasarkan form input model prediksi memperkirakan adanya risiko pengguna untuk terkena Prediabetes. Selain itu, sistem juga memberikan saran dan pencegahan yang dapat dilakukan pengguna untuk mengurangi risiko, seperti memperbaiki pola makan, meningkatkan aktivitas fisik, serta melakukan pemeriksaan kesehatan secara rutin sesuai rekomendasi dari Kementerian Kesehatan. Sistem ini memberikan prediksi serta langkah-langkah yang berguna untuk membantu pengguna memahami risikonya dan melakukan tindakan pencegahan yang tepat.

4. KESIMPULAN DAN SARAN

Penelitian ini secara jelas menunjukkan bahwa penerapan seleksi fitur dan teknik *imbalance learning* dapat secara signifikan meningkatkan kinerja prediksi algoritma *machine learning*, khususnya pada algoritma *Random Forest*. Dengan peningkatan akurasi sebesar 22,57% dibandingkan penelitian sebelumnya, model yang dikembangkan berhasil mengatasi permasalahan ketidakseimbangan data secara lebih efektif. Hasil ini menegaskan pentingnya pendekatan seleksi fitur yang relevan dan penyeimbangan data untuk meningkatkan kemampuan model dalam memprediksi risiko prediabetes secara akurat, terutama pada kelas minoritas.

Selain memberikan kontribusi dalam pengembangan model prediksi berbasis *machine learning*, penelitian ini juga menghadirkan sistem berbasis web yang menawarkan kemudahan akses bagi pengguna untuk mengetahui risiko kesehatan mereka. Sistem ini tidak hanya memberikan hasil prediksi, tetapi juga menyediakan rekomendasi langkah pencegahan yang dapat diambil pengguna, seperti perbaikan pola makan dan peningkatan aktivitas fisik. Hal ini menunjukkan potensi aplikasi nyata dari model dalam mendukung deteksi dini dan pengelolaan risiko kesehatan. Keberhasilan penelitian ini membuka peluang untuk pengembangan lebih lanjut, seperti penerapan algoritma yang lebih kompleks, termasuk *Deep Learning*, guna membandingkan performa dengan pendekatan yang lebih canggih. Selain itu, integrasi data real-time dari perangkat *wearable* dapat menjadi langkah inovatif untuk meningkatkan responsivitas dan akurasi sistem. Pengujian model pada populasi yang lebih luas dengan karakteristik demografi yang beragam juga diperlukan untuk memastikan generalisasi dan validitas model. Dengan hasil yang dicapai, penelitian ini diharapkan menjadi referensi penting dalam pengembangan sistem prediksi kesehatan berbasis data yang lebih efektif dan aplikatif, khususnya dalam mendukung upaya pencegahan penyakit kronis secara dini.

DAFTAR PUSTAKA

[1] D. C. P. Buani, "Deteksi Dini Penyakit Diabetes dengan Menggunakan Algoritma Random Forest," *Evolusi*, vol. 12, no. 1, Jun. 2024, doi: 10.31294/evolusi.v12i1.21005.

[2] W. Li, Y. Peng, and K. Peng, "Diabetes prediction model based on GA-XGBoost and stacking ensemble algorithm," *PLoS ONE*, vol. 19, no. 9, p. e0311222, Sep. 2024, doi: 10.1371/journal.pone.0311222.

[3] Md. M. Hassan, Z. J. Peya, S. Mollick, Md. A.-M. Billah, Md. M. Hasan Shakil, and A. U. Dulla, "Diabetes Prediction in Healthcare at Early Stage Using Machine Learning Approach," in *2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, Kharagpur, India: IEEE, Jul. 2021, pp. 01–05. doi: 10.1109/ICCCNT51525.2021.9579869.

[4] H. Kaur and V. Kumari, "Predictive modelling and analytics for diabetes using a machine learning approach," *ACI*, vol. 18, no. 1/2, pp. 90–100, Mar. 2022, doi: 10.1016/j.aci.2018.12.004.

[5] J.-C. Hsu, Y.-Y. Yang, S.-L. Chuang, L.-Y. Lin, and T. H.-H. Chen, "Prediabetes as a risk factor for new-onset atrial fibrillation: the propensity-score matching cohort analyzed using the Cox regression model coupled with the random survival forest," *Cardiovasc Diabetol*, vol. 22, no. 1, p. 35, Feb. 2023, doi: 10.1186/s12933-023-01767-x.

[6] P. Talari *et al.*, "Hybrid feature selection and classification technique for early prediction and severity of diabetes type 2," *PLoS ONE*, vol. 19, no. 1, p. e0292100, Jan. 2024, doi: 10.1371/journal.pone.0292100.

[7] R. B. Prasetyo, "Prediksi Dini Penyakit Diabetes Pada Ibu Hamil Dengan Algoritma Random Forest," vol. 02, 2024.

[8] A. Fauzi and A. H. Yunial, "Optimasi Algoritma Klasifikasi Naive Bayes, Decision Tree, K – Nearest Neighbor, dan Random Forest menggunakan Algoritma Particle Swarm Optimization pada Diabetes Dataset," *JEPIN*, vol. 8, no. 3, p. 470, Dec. 2022, doi: 10.26418/jp.v8i3.56656.

[9] A. K. Febianto and C. A. Sugianto, "Optimalisasi Algortima Klasifikasi Ensemble Menggunakan Algortima Genetika Untuk Prediksi Resiko Diabetes," vol. 5, no. 2, 2024.

[10] "Implementasi Data Mining Dalam Melakukan Prediksi Penyakit Diabetes Menggunakan Metode Random Forest Dan Xgboost," *jikstik*, vol. 23, no. 1, Mar. 2024, doi: 10.32409/jikstik.23.1.3507.

[11] K. Abnoosian, R. Farnoosh, and M. H. Behzadi, "Prediction of diabetes disease using an ensemble of machine learning multi-classifier models," *BMC Bioinformatics*, vol. 24, no. 1, p. 337, Sep. 2023, doi: 10.1186/s12859-023-05465-z.

[12] S. Gündoğdu, "Efficient prediction of early-stage diabetes using XGBoost classifier with random forest feature selection technique," *Multimed Tools Appl*, vol. 82, no. 22, pp. 34163–34181, Sep. 2023, doi: 10.1007/s11042-023-15165-8.

[13] A. A. Abokhzam, N. K. Gupta, and D. K. Bose, "Efficient diabetes mellitus prediction with grid based random forest classifier in association with natural language processing," *Int J Speech Technol*, vol. 24, no. 3, pp. 601–614, Sep. 2021, doi: 10.1007/s10772-021-09825-z.

[14] H. E. Massari, Z. Sabouri, S. Mhammedi, and N. Gherabi, "Diabetes Prediction Using Machine Learning Algorithms and Ontology," *JICTS*, May 2022, doi: 10.13052/jicts2245-800X.10212.

[15] Y.-M. Han *et al.*, "Risk prediction of diabetes and pre-diabetes based on physical examination data," *MBE*, vol. 19, no. 4, pp. 3597–3608, 2022, doi: 10.3934/mbe.2022166.

[16] N. Sneha and T. Gangil, "Analysis of diabetes mellitus for early prediction using optimal features selection," *J Big Data*, vol. 6, no. 1, p. 13, Dec. 2019, doi: 10.1186/s40537-019-0175-6.
[17] Z. Susanti, P. Sirait, and E. S. Panjaitan, "Peningkatan Kinerja Random Forest Melalui Seleksi Fitur Secara Pca Untuk Mendeteksi Penyakit Diabetes Tahap Awal," vol. 4, no. 3, 2023.