Implementasi Data Mining dalam Melakukan Prediksi Penyakit Diabetes Menggunakan Metode Random Forest dan Xgboost

Muhammad Salsabil, Nuril Lutvi Azizah dan Ade Eviyanti

Informatika, Fakultas Sains dan Teknologi Universistas Muhammdiyah Sidoarjo Kampus 2
JL. Raya Gelam No.250, Pagerwaja, Gelam, Kec.Candi, Kabupaten Sidoarjo, Jawa Timur 61271
E-mail: 191080200129@umsida.ac.id, nurillutviazizah@umsida.ac.id, adeeviyanti@umsida.ac.id

Abstrak

Penyakit diabetes telah menjadi isu global dalam bidang kesehatan. Penelitian ini berfokus pada implementasi dua teknik data mining yaitu Random Forest dan XGBoost untuk memproyeksikan perkembangan penyakit diabetes. Kedua metode ini memanfaatkan dataset klinis dan biokimia yang terkait dengan diabetes. Setelah fase preprocessing, dilakukan evaluasi performa menggunakan metrik evaluasi seperti akurasi, presisi, recall dan f1 score. Dataset yang digunakan sebanyak 768 entri dan 9 indikator yang diperoleh dari platform Kaggle. Dalam Penelitian ini data diolah melalui tahap preprocessing diantaranya handling missing value, handling outlier dan normalisasi data, dan didapatkan data yang akan diolah sebesar 688. Setelah didapat data hasil preprocessing, dilakukan tahapan pelatihan dan pengujian dengan Cross Validation dan dilakukan pengujian untuk mengetahui parameter-parameter terbaik yang akan digunakan, lalu dilakukan evaluasi kinerja model Random Forest dan XGBoost menggunakan metrik akurasi, presisi, recall, dan F1-score. Hasil evaluasi model menunjukkan performa yang baik dalam penelitian ini, didapatkan hasil akurasi keseluruhan dalam penggunaan random forest sebesar 74% dan penggunaan XGBoost sebesar 76%.

Kata kunci : *Data Mining, Prediksi Penyakit Diabetes, Random Forest, XGBoost, Evaluasi Model, Kaggle.*

Pendahuluan

Diabetes mellitus adalah salah satu penyakit yang menjadi masalah kesehatan global yang signifikan. Penyakit ini ditandai oleh peningkatan kadar gula darah yang disebabkan oleh masalah dalam produksi atau penggunaan hormon insulin dalam tubuh[1]. Dalam rangka menghadapi tantangan ini, pengembangan metode klasifikasi yang akurat dan efisien dalam mendiagnosis penyakit diabetes menjadi sangat penting.

Dalam beberapa tahun terakhir, penggunaan metode pembelajaran mesin untuk klasifikasi penyakit telah menjadi fokus penelitian yang meningkat[2]. Metode ini memungkinkan para peneliti untuk menganalisis dataset yang besar dan kompleks dengan lebih efisien, sehingga dapat menghasilkan model yang dapat memprediksi dan mengklasifikasikan penyakit dengan tingkat akurasi yang tinggi[3].

Dalam penelitian ini, akan memfokuskan pada penggunaan dua metode klasifikasi yang populer, yaitu Random Forest dan XGBoost, untuk melakukan klasifikasi penyakit diabetes. Random Forest adalah teknik dalam machine learning yang menggabungkan beberapa pohon keputusan untuk membuat prediksi. Setiap pohon dibangun secara independen menggunakan subset acak dari data pelatihan, dan kemudian prediksi dari setiap pohon digunakan untuk membuat prediksi akhir. Salah satu keunggulan utamanya adalah kemampuannya dalam menangani dataset yang besar dan kompleks, sambil mengurangi risiko overfitting yang sering terjadi pada model yang kompleks. Dengan menggunakan Random Forest, kita dapat memperoleh prediksi yang lebih stabil dan terpercaya, serta memiliki kemampuan untuk mengevaluasi pentingnya setiap fitur dalam prediksi yang dibuat[4]. XGBoost, singkatan dari Extreme Gradient Boosting, merupakan metode yang sangat efektif dalam machine learning, terutama untuk tugas klasifikasi dan regresi. Dengan menggunakan teknik ensemble learning, XGBoost secara bertahap membangun model prediksi dengan menambahkan model pohon keputusan secara berurutan. Setiap model berusaha untuk memperbaiki kesalahan prediksi yang dilakukan oleh model sebelumnya. Selain itu, XGBoost melakukan optimisasi ob-

DOI : http://dx.doi.org/10.32409/jikstik.23.1.3507

jektif yang agresif untuk meminimalkan kesalahan prediksi dan menggunakan regularisasi untuk mengontrol kompleksitas model serta mencegah overfitting. Selain itu, XGBoost memiliki fitur untuk mengevaluasi pentingnya setiap fitur dalam pembuatan prediksi. Dengan kinerja yang tinggi dan fleksibilitasnya, XGBoost telah menjadi salah satu pilihan utama dalam berbagai kompetisi data dan aplikasi di berbagai industri[5].

Tujuan dari penelitian ini adalah melakukan analisis kinerja metode Random Forest dan XGBoost dalam klasifikasi penyakit diabetes. Menggunakan dataset yang terdiri dari berbagai fitur klinis dan biokimia yang relevan dengan diabetes, dan melatih model menggunakan kedua metode tersebut[6]. Selanjutnya, pada penelitian ini akan mengevaluasi kinerja model menggunakan metrik yang umum digunakan seperti akurasi, presisi, recall, dan F1-score. Diharapkan melalui penelitian ini, dapat diperoleh pemahaman yang lebih baik tentang efektivitas dan kelebihan masing-masing metode dalam klasifikasi penyakit diabetes, serta memberikan wawasan baru dalam penggunaan metode Random Forest dan XGBoost untuk klasifikasi penyakit diabetes. Selain itu, hasil penelitian ini dapat menjadi dasar untuk pengembangan metode klasifikasi yang lebih baik dan akurat dalam bidang medis. Beberapa penelitian mengenai prediksi penyakit diabetes melitus telah banyak dilakukan dengan berbagai metode, untuk menguji tingkat presisi dan ketepatan dalam memprediksi penyakit diabetes, beberapa penelitian terdahulu antara lain:

Penelitian berjudul "Peningkatan Kinerja Akurasi Prediksi Penyakit Diabetes Mellitus Menggunakan Metode Grid Seacrh pada Algoritma Logistic Regression". Penulis Muhamad Ichsan Gunawan, Dedy Sugiarto, Is Mardianto pada tahun 2020, Penelitian ini bertujuan untuk meningkatkan akurasi prediksi penyakit Diabetes Mellitus menggunakan metode Regresi Logistik dengan menerapkan teknik Grid Search. Metode penelitian melibatkan penggunaan dataset Pima Indians Diabetes Database. Hasilnya, model Regresi Logistik memiliki rata-rata akurasi sekitar 79%. Ketika diuji dengan data baru, model ini menunjukkan akurasi sebesar 83,33%[7].

Penelitian Berjudul "Komparasi Metode Klasifikasi Data Mining Decision Tree dan Naive Bayes Untuk Prediksi Penyakit Diabetes", Penulis Baiq Andriska Candra Permana, Intan Komala Dewi, pada tahun 2021. Tujuan penelitian yaitu untuk dapat mengetahui secara dini seseorang mengalami diabetes. Algoritma klasifikasi yang digunakan decision tree dan naive bayes dengan menggunakan cross validation, Data yang digunakan berasal dari kaggle terdiri atas 520 data pasien dan 17 atribut, hasil yang didapat yaitu algoritma klasifikasi decision tree lebih baik dalam prediksi penyakit diabetes dengan nilai akurasi 95,58% dab nilai AUC 0,981 lebih tinggi dibandingkan naïve bayes dengan akurasi 87,69% dan nilai AUC 0,947[8].

Penelitian Berjudul "Prediksi Penyakit Diabetes Menggunakan Algoritma Support Vector Machine (SVM)". Penulis Hovi Sohibul Wafa, Asep Id Hadiana, Fajri Rakhmat Umbara, pada tahun 2022. Penelitian ini bertujuan untuk menentukan apakah penderita/pasien dapat terkena penyakit diabetes atau tidak dengan menerapkan teknik data mining dan klasifikasi menggunakan algoritma SVM Radial Basis Function berbasis Forward Selection. Database yang digunakan dalam penelitian ini adalah data dengan jenis wanita dengan keturunan indian pima yang memiliki 8 atribut dan 1 label, 8 atribut. Hasil penelitian bahwa model support vector machine (rbf) mampu memberikan hasil akurasi yang diperoleh sebesar 91.2% untuk accuracy, 93.0% untuk precision, 94.3% untuk recall, dan 93.7% untuk f1-scorer, dari hasil evaluasi confusion matrix[9].

Penelitian Berjudul "Implementasi Algoritma K-Nearest Neighbor (KNN) Untuk Memprediksi Pasien Terkena Penyakit Diabetes Pada Puskesmas Manyampa Kabupaten Bulukumba. Nama Penulis M. Syukri Mustafa, I Wayan Simpen, pada tahun 2019. Penelitian ini dimaksudkan untuk melakukan pengujian terhadap kemungkinan seorang pasien baru pada puskesmas Manyampa dapat terkena penyakit diabetes militus atau tidak menggunakan analisis data mining. Data training yang digunakan dalam penelitian ini sebanyak 200 data pasien yang telah melakukan pemeriksaan dalam 2 tahun terakhir. Hasil akurasi yang dipeloleh dari pengujian tersebut sebesar 68,30%. Hasil pengujian data dari sistem ini yang menggunakan 104 data pasien pada puskesmas Manyampa memperoleh hasil prediksi yang benasr sebanyak 71 dan salah atau ragu-ragu sebesar 33 dengan tingkat akurasi sebesar 68.3%[10].

Penelitian Berjudul "Implementasi Data Mining Untuk Prediksi Penyakit Diabetes Dengan Algoritma C4.5. Nama Penulis Sanni Ucha Putri, Eka Irawan, Fitri Rizky. Database yang digunakan adalah 49 data pasien penyakit diabetes dari RSUD Dr. Djasamen Saragih Pematangsiantar, pada tahun 2021. Penelitian ini bertujuan untuk membuat model prediksi menggunakan Data Mining Algoritma C4.5 yang menghasilkan sebuah pohon keputusan serta pengujian yang dilakukan dengan menggunakan Rapidminner agar pencegahan terhadap penyakit diabetes dapat dilakukan segera mungkin. Algoritma yang digunakan Decision tree dan algoritma C4.5. Hasil yang didapat Implementasi Data Mining menggunakan Algoritma C4.5 prediksi positif mencapai 90,00% dari total 36 prediksi. Algoritma C4.5 dalam RapidMiner menghasilkan hasil yang konsisten dengan perhitungan manual dengan akurasi 90,00%[11].

Pada penelitian Implementasi Data Mining Dalam Melakukan Prediksi Penyakit Diabetes ini, peneliti menggunakan database yang berasal dari kaggle, sebanyak 768 data dengan mencakup 9 in-

dikator penyakit diabetes. Penelitian ini menggunakan 2 metode yaitu Random forest dan XGBoost, penggunaan dua metode tersebut untuk menganalisa prediksi dalam penyakit diabetes. Penelitian ini melalui beberapa tahapan-tahapan preprocessing untuk mengolah data-data awal sebelum di ujikan dengan kedua pemodelan topik antara random forest dan XGBoost, dilakukan pengujian menggunakan cross validation 5. Menggunakan matrix evaluasi akurasi, presisi, recall dan f1 score. Kedua metode ini efektif dalam mengatasi overfitting dan cocok untuk data besar.

Metode Penelitian

Pada Tahapan ini menggambarkan bagaimana pola alur penelitian yang dilakukan selama pengerjaan pebelitian ini, digambarkan secara rinci mengenai metode dan proses yang digunakan dalam melakukakan prediksi penyakit diabetes menggunakan metode random forest dan Xgboost, berikut tahapan-tahapannya:

1. Desain Sistem

   Perancangan perangkat lunak diawali dengan tahap preprocessing. Jika data telah melalui tahap Preprocessing, maka dilakukan modelling menggunakan metode Random Forest dan XGBoost[12]. Untuk mendapatkan parameter terbaik, peneliti menggunakan teknik Grid Search Cross Validation dengan cross validation 5 untuk melakukan tuning pada saat pemodelan. Berikut desain sistem alur program yang dibuat,

Gambar 1: Desain Sistem

Pada Gambar 1, merupakan alur perancangan sistem pada penelitian ini, berjalan melalui beberapa proses mulai dari pengumpulan data, preprocessing data awal, processing dan pemodelan metode random forest dan Xgboost, hingga tahap analisis hasil supaya menciptakan hasil yang akurat.

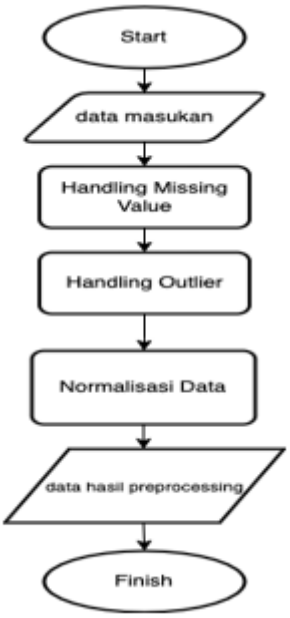
2. Data

   Data yang digunakan dalam penelitian ini berasal dari Kaggle dan terdiri dari 768 entri. Dataset ini mencakup 9 indikator yang terkait dengan penyakit diabetes yang digunakan dalam penelitian ini, Meliputi:

   (a) Pregnacies
   (b) Glucose
   (c) Blood Presure
   (d) Skin Thickness
   (e) Insulin
   (f) BMI
   (g) Diabetes pedigree function
   (h) Age outcome

3. Preprocessing

   Data mentah tidak dapat langsung digunakan oleh sistem dan memerlukan tahap preprocessing. Preprocessing dilakukan untuk memodifikasi data dan meningkatkan kualitasnya sebelum dilakukan pemodelan[6]. Pada penelitian ini, preprocessing dilakukan untuk membersihkan data sebelum proses pemodelan dimulai.



Gambar 2: Tahap Preprocessing

Pada Gambar 2 dijelaskan tahapan-tahapan dalam preprocessing data sebelum dilakukan pemodelan topik random forest dan xgboost,

beberapa tahapan penting diantaranya handling missing value, handling outlier dan normalisasi data. Setelah data berhasil diolah baru dilakukan pemodelan dengan kedua topik tersebut.

Handling Missing Value

Pada tahapan awal preprocessing, data masukan dilakukan handling missing value terlebih dahulu. Missing value dapat terjadi karena kesalahan penginputan data atau memang data tersebut tidak ada. Karena algoritma machine learning tidak dapat memproses data yang terdapat missing value, maka sebelum dilakukan modelling harus dilakukan handling missing value terlebih dahulu. Peneliti menggunakan Teknik imputasi mean. Sehingga data yang terdapat missing value diisi dengan nilai rata-rata dari kolom tersebut.

Handling Outlier

Selanjutnya, tahapan berikutnya adalah melakukan penanganan outlier menggunakan teknik Z-score. Z-score digunakan untuk membantu mengidentifikasi apakah suatu data termasuk dalam kategori outlier atau bukan. Data outlier merujuk pada data yang memiliki nilai yang sangat jauh dari rata-rata. Aturan umum yang digunakan adalah jika nilai Z-score kurang dari -3 atau lebih dari +3, maka data tersebut dianggap sebagai nilai ekstrem. Oleh karena itu, data yang melebihi batas bawah atau batas atas tersebut akan dihapus dari dataset.

Normalisasi Data

Normalisasi data adalah salah satu teknik penting dalam tahap preprocessing. Hal ini penting karena seringkali data memiliki rentang nilai yang berbeda antar variabelnya. Dalam penelitian ini, peneliti menggunakan metode min-max scaler untuk melakukan normalisasi data. Metode ini akan menyesuaikan nilai-nilai data ke dalam rentang yang ditentukan, biasanya 0 hingga 1, sehingga memungkinkan perbandingan yang lebih adil antar variabel.

4. Processing

   Pada sub bab ini akan dijelaskan mengenai proses pengolahan data sehingga mendapatkan hasil yang diharapkan seperti yang telah dijelaskan sebelumnya.

   Klasifikasi dengan Random Forest dan XGBoost

   Langkah pertama dari tahap klasifikasi dengan Random Forest dan XGBoost adalah membuka data yang telah dilakukan ekstraksi fitur ke dalam jupyter notebook menggunakan library pandas. Ketika data sudah berhasil diload, maka dilakukan pembagian data antara data X dan data y dimana data X merupakan kolom fitur dan data y merupakan kolom target[13].

   Setelah melakukan pembagian data X dan y, kemudian dilakukan pembagian data train dan data test pada data X menggunakan modul scikit-learn yaitu train_test_split. Besaran pembagian data yaitu 80% untuk data train dan 20% untuk data test.

   Untuk mendapatkan parameter yang paling optimal pada kasus ini, peneliti menggunakan Teknik GridSearchCV yang mana Teknik ini dapat mencari parameter optimal dari algoritma yang digunakan untuk kasus yang sedang dianalisa[14]. GridSearchCV adalah bagian dari modul scikit-learn yang bertujuan secara otomatis dan sistematis melakukan validasi beberapa model dan setiap hyperparameter. Ketika proses running GridSearchCV sudah selesai, maka akan didapatkan model beserta score test dan score train.

5. Tahap Evaluasi

   Tahapan ini digunakan untuk mengukur performa dari model machine learning yang telah dibuat. terdapat tiga metrik evaluasi yang dapat digunakan yaitu precision, recall dan confusion matrix[15]. Seperti yang telah dijelaskan pada sub bab sebelumnya, data akan dibagi menjadi data train dan data test dengan perbandingan[16].

   Agar mendapatkan hasil terbaik, beberapa perbandingan data train dan data test nantinya akan dilakukan percobaan pada masing-masing perbandingan[17]. Pada penelitian ini, peneliti menggunakan Randomized Search Cross Validation dengan cross validation adalah 5. Sehingga dataset dibagi menjadi 5 bagian data sama banyak. Jika bagian 1 menjadi data test maka bagian 2 hingga 5 menjadi data train. Sedangkan jika bagian 2 menjadi data test maka bagian 1, 3, 4 dan 5 menjadi data train. Begitu seterusnya hingga bagian 5 menjadi data test.

Hasil dan Pembahasan

Data

Penelitian ini berfokus pada analisis data yang terdiri dari 768 entri. Dataset ini mencakup 9 indika-

tor yang memiliki keterkaitan dengan penyakit diabetes. Dalam penelitian ini, 9 kategori yang mewakili indikator-indikator tersebut diidentifikasi. Fokus penggunaan 9 indikator ini adalah untuk membatasi ruang lingkup masalah dalam prediksi penyakit diabetes, lihat Tabel 1. Indikator-indikator tersebut meliputi:

1. Pregnancies
2. Glucose
3. Blood Pressure
4. Skin Thickness
5. Insulin
6. BMI
7. Diabetes Pedigree Function
8. Age
9. Outcome

Tabel 1: Indikator Data

Attribut	Keterangan
Pregnancies	Jumlah Kehamilan
Glucose	Konsentrasi glukosa plasma 2 jam setelah uji toleransi glukosa oral
Blood Presure	Tekanan darah diastolik
Skin Thickness	Ketebalan lipatan kulit trisep
Insulin	Insulin serum 2 jam
BMI	Indikator untuk menentukan kategori berat badan
Diabetes Pedigree Function	Fungsi silsilah diabetes
Age	umur
Outcome	kelas

Dengan mempertimbangkan 9 indikator ini, penelitian ini bertujuan untuk lebih memahami dan merumuskan prediksi penyakit diabetes dengan kerangka kerja yang kokoh.

Preprocessing

1. Handling Mising Value

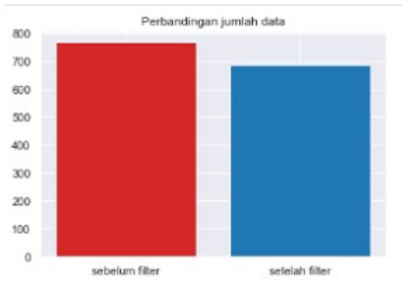
   Output yang dihasilkan dari langkah ini adalah data yang telah mengalami penyaringan dan perubahan.

   Pengisian data dilakukan secara komprehensif sehingga tidak ada nilai yang terlewatkan. Presentasi visual dalam Gambar 3 mencerminkan keseragaman warna yang sama. Pengamatan ini mengindikasikan bahwa pengisian nilai telah berhasil dilakukan dengan sukses.

Gambar 3: Hasil *Missing Value*

2. Handling Outlier

   Tahap berikutnya melibatkan penanganan outlier menggunakan metode Z-score. Z-score digunakan untuk membantu mengidentifikasi apakah suatu data termasuk dalam kategori outlier atau tidak. Data outlier merujuk pada data yang memiliki nilai yang signifikan dari rata-rata. Pedoman umum adalah jika nilai Z-score lebih kecil dari -3 atau lebih besar dari +3, data dianggap sebagai nilai ekstrem. Karena itu, data yang melampaui batas-batas ini akan dihilangkan dari dataset.



Gambar 4: Proses *Outlier*

Setelah menerapkan metode Z-score untuk mengatasi outlier, Gambar 4 menunjukkan bahwa dari jumlah data awal sebanyak 768, jumlah data yang tersisa setelah proses tersebut adalah 688.

3. Normalisasi Data

   Normalisasi data adalah langkah kunci dalam preprocessing, mengingat variasi nilai antar variabel[18]. Dalam penelitian ini, metode yang diterapkan adalah skala min-max. Pendekatan ini mengatur nilai data dalam rentang 0 hingga 1 untuk perbandingan yang lebih adil[19].

```
from sklearn.preprocessing import MinMaxScaler
# Inisialisasi MinMaxScaler
scaler = MinMaxScaler()

# Fit scaler ke data dan transformasi data
normalized_data = scaler.fit_transform(X)

print("Data awal:\n", X)
print("Data yang sudah dinormalisasi:\n", normalized_data)
```

```
Data awal:
      Pregnancies  Glucose  BloodPressure  SkinThickness  Insulin   BMI  \
0             6.0    148.0           72.0           35.0      0.0  33.6
1             1.0     85.0           66.0           29.0      0.0  26.6
2             8.0    183.0           64.0            0.0      0.0  23.3
3             1.0     89.0           66.0           23.0     94.0  28.1
5             5.0    116.0           74.0            0.0      0.0  25.6
..            ...      ...            ...            ...      ...   ...
763          10.0    101.0           76.0           48.0    180.0  32.9
764           2.0    122.0           70.0           27.0      0.0  36.8
765           5.0    121.0           72.0           23.0    112.0  26.2
766           1.0    126.0           60.0            0.0      0.0  30.1
767           1.0     93.0           70.0           31.0      0.0  30.4

     DiabetesPedigreeFunction   Age
0                       0.627  50.0
1                       0.351  31.0
2                       0.672  32.0
3                       0.167  21.0
5                       0.201  30.0
..                        ...   ...
763                     0.171  63.0
764                     0.340  27.0
765                     0.245  30.0
766                     0.349  47.0
767                     0.315  23.0

[688 rows x 8 columns]
Data yang sudah dinormalisasi:
 [[0.46153846 0.67096774 0.48979592 ... 0.41847826 0.39696312 0.61702128]
 [0.07692308 0.26451613 0.42857143 ... 0.22826087 0.19739696 0.21276596]
 [0.61538462 0.89677419 0.40816327 ... 0.13858696 0.42950108 0.23404255]
 ...
 [0.38461538 0.49677419 0.48979592 ... 0.2173913  0.12075199 0.19148936]
 [0.07692308 0.52903226 0.36734694 ... 0.32336957 0.19595083 0.55319149]
 [0.07692308 0.31612903 0.46938776 ... 0.33152174 0.17136659 0.04255319]]
```

Gambar 5: Proses Normalisasi Data

Dalam rangka penelitian ini, metode yang digunakan melibatkan penerapan skala min-max. Pendekatan ini mengelola nilai-nilai data agar berada dalam kisaran 0 hingga 1, untuk memastikan kesetaraan perbandingan. Gambar 5 menunjukkan hasil prediksi penyakit diabetes yang dihasilkan setelah menjalankan metode skala min-max.

Processing

Cross-validation adalah pendekatan yang membagi dataset menjadi beberapa bagian untuk tujuan pelatihan dan pengujian model. Dalam konteks Random Forest, strategi ini membantu mengukur sejauh mana model mampu menggeneralisasi data yang belum pernah dilihatnya[20]. Model diperlakukan sebagai pelatihan dan diuji pada setiap bagian dataset, sambil menghitung metrik evaluasi yang relevan[21]. Hasil dari setiap tahapan diambil untuk memberikan gambaran yang lebih akurat tentang performa keseluruhan model Berikut merupakan parameter-parameter terbaik dalam random forest dan XGBoost.

Pada Tabel 2, menjelaskan parameter-parameter yang memberikan performa terbaik untuk model Random Forest dan XGBoost sesuai dengan metrix evaluasi yang telah ditentukan.

Tabel 2: Parameter Random Forest dan XGBoost

Algoritma	Parameter
Random Forest	Fitting 5 folds for each of 10 candidates, totalling 50 fits [Parallel(n_jobs=-1)]: Using backend LokyBackend with 8 concurrent workers. [Parallel(n_jobs=-1)]: Done 34 tasks | elapsed: 10.8s [Parallel(n_jobs=-1)]: Done 50 out of 50 | elapsed: 12.7s finished 0.9646569646569647 0.7713273195876289 0.7439613526570048 `model_rf.best_params_` {'algo__max_depth': 29, 'algo__max_features': 0.9051404267682241, 'algo__min_samples_leaf': 3, 'algo__n_estimators': 110}
XGBoost	Fitting 5 folds for each of 10 candidates, totalling 50 fits [Parallel(n_jobs=-1)]: Using backend LokyBackend with 8 concurrent workers. [Parallel(n_jobs=-1)]: Done 34 tasks | elapsed: 49.5s [Parallel(n_jobs=-1)]: Done 50 out of 50 | elapsed: 49.8s finished 0.7879417879417879 0.7754295532646047 0.7294685990338164 `model_xgb.best_params_` {'algo__colsample_bytree': 0.293530703818476, 'algo__gamma': 10, 'algo__learning_rate': 0.3556191493125588 4, 'algo__max_depth': 2, 'algo__n_estimators': 160, 'algo__reg_alpha': 1.5611961512098822, 'algo__reg_lambda': 0.04563169215430241, 'algo__subsample': 0.32688076857370457}

Evaluasi

Setelah melalui serangkaian langkah-langkah penting dalam preprocessing dan tahap klasifikasi data, langkah selanjutnya yang dijalankan adalah tahap evaluasi. Pada tahap ini, peneliti melanjutkan dengan menggunakan confusion matrix sebagai instrumen untuk melakukan evaluasi mendalam terhadap performa model yang telah dibangun.

Tabel 3: *Clasification Report*

Algoritma	*Clasification Report*			
Random Forest		precision	recall	f1-score	support
	0.0	0.75	0.88	0.81	127
	1.0	0.74	0.53	0.61	80
	accuracy			0.74	207
	macro avg	0.74	0.70	0.71	207
	weighted avg	0.74	0.74	0.73	207
XGBoost		precision	recall	f1-score	support
	0.0	0.75	0.91	0.83	127
	1.0	0.79	0.53	0.63	80
	accuracy			0.76	207
	macro avg	0.77	0.72	0.73	207
	weighted avg	0.77	0.76	0.75	207

Berdasarkan Tabel 3, berikut hasilnya:
Random Forest :

1. Akurasi keseluruhan: 74%
2. Rata-rata (macro avg): Presisi 0.74, Recall 0.70, F1-Score 0.71
3. Rata-rata berbobot (weighted avg): Presisi 0.74, Recall 0.74, F1-Score 0.73

XGBoost :

1. Akurasi keseluruhan: 76%
2. Rata-rata (macro avg): Presisi 0.77, Recall 0.72, F1-Score 0.73
3. Rata-rata berbobot (weighted avg): Presisi 0.77, Recall 0.76, F1-Score 0.75

Hasil ini memberikan gambaran tentang seberapa baik kinerja kedua model dalam melakukan klasifikasi, dengan mempertimbangkan presisi, recall, dan F1-score dari masing-masing model.

Random Forest:

1. Akurasi Keseluruhan: Model Random Forest memiliki akurasi keseluruhan sebesar 74%. Akurasi mengukur seberapa banyak prediksi model yang benar dari seluruh instance dalam data uji.
2. Rata-rata:
   (a) Presisi: 0.74, yang berarti dari semua kelas, 74% prediksi positif yang benar.
   (b) Recall: 0.70, yang berarti dari semua instance yang benar-benar positif, model mampu memprediksi 70% di antaranya.
   (c) F1-Score: 0.71, yang merupakan rata-rata harmonik dari presisi dan recall. F1-score memberikan keseimbangan antara kedua metrik tersebut.
3. Rata-rata Berbobot (Weighted Avg):
   (a) Presisi: 0.74, yang menunjukkan presisi secara rata-rata dengan mempertimbangkan kelas-kelas yang berbeda.
   (b) Recall: 0.74, yang menunjukkan recall secara rata-rata dengan mempertimbangkan kelas-kelas yang berbeda.
   (c) F1-Score: 0.73, yang menunjukkan F1-score secara rata-rata dengan mempertimbangkan kelas-kelas yang berbeda.

XGBoost:

1. Akurasi Keseluruhan:
   (a) Model XGBoost memiliki akurasi keseluruhan sebesar 76%.
2. Rata-rata (Macro Avg):
   (a) Presisi: 0.77, yang berarti dari semua kelas, 77% prediksi positif yang benar.
   (b) Recall: 0.72, yang berarti dari semua instance yang benar-benar positif, model mampu memprediksi 72% di antaranya.
   (c) F1-Score: 0.73, yang merupakan rata-rata harmonik dari presisi dan recall.
3. Rata-rata Berbobot (Weighted Avg):
   (a) Presisi: 0.77, yang menunjukkan presisi secara rata-rata dengan mempertimbangkan kelas-kelas yang berbeda.
   (b) Recall: 0.76, yang menunjukkan recall secara rata-rata dengan mempertimbangkan kelas-kelas yang berbeda.
   (c) F1-Score: 0.75, yang menunjukkan F1-score secara rata-rata dengan mempertimbangkan kelas-kelas yang berbeda.

Penutup

Dalam Penelitian Implementasi Data Mining Dalam Melakukan Prediksi Penyakit Diabetes Menggunakan Metode Random Forest Dan Xgboost, Penerapan teknik data mining melalui Random Forest dan XGBoost dalam meramalkan penyakit diabetes keduanya memberikan hasil prediksi yang akurat dan konsisten berdasarkan analisis dataset klinis dan biokimia yang didapat dari situs kaggle yang berjumlah 768 dan 9 indikator. Data diolah melalui tahap preprocessing diantaranya handling missing value, handling outlier dan normalisasi data, dan didapatkan data yang akan diolah sebesar 688. Setelah didapat data hasil preprocessing, data dilakukan tahapan pelatihan dan pengujian dengan Cross Validation dan dilakukan pengujian untuk mengetahui parameter-parameter terbaik yang akan digunakan, lalu dilakukan evaluasi kinerja model Random Forest dan XGBoost menggunakan metrik akurasi, presisi, recall, dan F1-score. Hasil evaluasi model menunjukkan performa yang baik dalam penelitian ini, didapatkan hasil akurasi keseluruhan dalam penggunaan random forest sebesar 74% dan penggunaan XGBoost sebesar 76%. Kesimpulan ini menyoroti peran penting data mining dalam mengatasi tantangan kesehatan global, penelitian ini berpotensi mendukung upaya deteksi dini dan pengelolaan penyakit khususnya diabetes.

Daftar Pustaka

[1] T. Hidayat, S. S. Anelia, R. I. Pratiwi, N. Salsabila dan D. S. Prasvita, "Perbandingan Akurasi Klasifikasi Penyakit Diabetes Menggunakan Algoritma Adaboost- Random Forest Dan Adaboost- Decision Tree Dengan Imputasi Median dan KNN", Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)Jakarta-Indonesia,e-ISBN 978-623-93343-3-8, pp. 616–623, April 2021.

[2] G. Abdurrahman, H. Oktavianto dan M. Sintawati, "Optimasi Algoritma XGBoost Classifier Menggunakan Hyperparameter Gridesearch dan Random Search Pada Klasifikasi Penyakit Diabetes", INFORMAL Informatics J., vol. 7, no. 3, p. 193, doi: 10.19184/isj.v7i3.35441, 2022.

[3] A. Fauzi dan A. H. Yunial, " Optimasi Algoritma Klasifikasi Naive Bayes, Decision Tree, K-Nearest Neighbor, dan Random Forest menggunakan Algoritma Particle Swarm Optimization pada Diabetes Dataset", J. Edukasi dan Penelit. Inform., vol. 8, no. 3, pp. 470–481, 2022.

[4] F. ANISHA, Dodi Vionanda, Nonong amalita, and Zilrahmi, "Application of Random Forest for The Classification Diabetes Mellitus Disease in RSUP Dr. M. Jamil Padang", UNP J. Stat. Data Sci., vol. 1, no. 2, pp. 45–52, doi: 10.24036/ujsds/vol1-iss2/30, 2023.

[5] N. N. Pandika Pinata, I. M. Sukarsa dan N. K. Dwi Rusjayanthi, "Prediksi Kecelakaan Lalu Lintas di Bali dengan XGBoost pada Python", J. Ilm. Merpati (Menara Penelit. Akad. Teknol. Informasi), vol. 8, no. 3, p. 188, doi: 10.24843/jim.2020.v08.i03.p04, 2020.

[6] Gde Agung Brahmana Suryanegara, Adiwijaya dan Mahendra Dwifebri Purbolaksono, "Peningkatan Hasil Klasifikasi pada Algoritma Random Forest untuk Deteksi Pasien Penderita Diabetes Menggunakan Metode Normalisasi", J. RESTI (Rekayasa Sist. dan Teknol. Informasi), vol. 5, no. 1, pp. 114–122, doi: 10.29207/resti.v5i1.2880, 2021.

[7] Muhamad Ichsan Gunawan, Dedy Sugiarto dan Is Mardianto, "Peningkatan Kinerja Akurasi Prediksi Penyakit Diabetes Mellitus Menggunakan Metode Grid Seacrh pada Algoritma Logistic Regression", J. Edukasi dan Penelit. Inform., vol. 6, no. 3, pp. 280–284, 2020.

[8] B. A. Candra Permana dan I. K. Dewi Patwari, "Komparasi Metode Klasifikasi Data Mining Decision Tree dan Naïve Bayes Untuk Prediksi Penyakit Diabetes", Infotek J. Inform. dan Teknol., vol. 4, no. 1, pp. 63–69, doi: 10.29408/jit.v4i1.2994, 2021.

[9] H. S. Wafa Hofi, Asep Id Hadiana dan F. Rakhmat Umbara, "Prediksi Penyakit Diabetes Menggunakan Algoritma Support Vector Machine (SVM)", Informatics Digit. Expert, vol. 4, no. 1, pp. 40–45, doi: 10.36423/index.v4i1.895, 2022.

[10] M. Syukri Mustafa and I. Wayan Simpen, "Implementation of the K-Nearest Neighbor (KNN) Algorithm to Predict Patients Affected by Diabetes at the Manyampa Health Center, Bulukumba Regency", Pros. Semin. Ilm. Sist. Indormasi dan Teknol. Inf., vol. VIII, no. 1, pp. 1–10, 2019.

[11] S. Ucha Putri, E. Irawan dan F. Rizky, "Implementasi Data Mining Untuk Prediksi Penyakit Diabetes Dengan Algoritma C4.5", Kesatria - Jurnal Penerapan Sistem Informasi (Komputer dan Manajemen) , Januari, vol. 2, no. 1, pp. 39–46, 2021.

[12] E. C. P. Witjaksana, R. R. Saedudin dan V. P. Widartha, "Perbandingan Akurasi Algoritma Random Forest dan Algoritma Artificial Neural Network untuk Klasifikasi Penyakit Diabetes", e-Proceeding Eng., vol. 8, no. 5, pp. 9765–9772, 2021.

[13] M. D. Purbolaksono, M. Irvan Tantowi, A. Imam Hidayat dan A. Adiwijaya, "Perbandingan Support Vector Machine dan Modified Balanced Random Forest dalam Deteksi Pasien Penyakit Diabetes", J. RESTI (Rekayasa Sist. dan Teknol. Informasi), vol. 5, no. 2, pp. 393–399, doi: 10.29207/resti.v5i2.3008, 2021.

[14] Z. Munawar, N. I. Putri dan D. Z. Musadad, "Meningkatkan Rekomendasi Menggunakan Algoritma Perbedaan Topik", J. Sist. Inf., vol. 01, no. 02, pp. 17–26, 2020.

[15] A. E. Pramadhani dan T. Setiadi, "Penerapan Data Mining untuk Klasifikasi Penyakit ISPA dengan Algoritma Desicion Tree", J. Sarj. Tek. Inform. e-ISSN 2338-5197, vol. 2, no. 1, pp. 831–839, 2014.

[16] N. Chamidah, U. Salamah, "Pengaruh Normalisasi Data pada Jaringan Syaraf Tiruan Backpropagasi Gradient Descent Adaptive Gain (BPGDAG) untuk Klasifikasi", J. Itsmart, vol. 1, no. 1, pp. 28–33, 2012.

[17] W. Apriliah, I. Kurniawan, M. Baydhowi dan T. Haryati, "Prediksi Kemungkinan Diabetes pada Tahap Awal Menggunakan Algoritma Klasifikasi Random Forest", SISTEMASI: Jurnal Sistem Informasi , vol. 10, no. 1, pp. 163–171, 2021.

[18] R. Bonetto and V. Latzko, "Machine learning", in book Comput. Commun. Networks From Theory to Pract., pp. 135–167, doi: 10.1016/B978-0-12-820488-7.00021-9, 2021.

[19] N. L. Rachmawati dan M. Lentari, "Penerapan Metode Min-Max untuk Minimasi Stockout dan Overstock Persediaan Bahan Baku", J. INTECH Tek. Ind. Univ. Serang Raya, vol. 8, no. 2, pp. 143–148, doi: 10.30656/intech.v8i2.4735, 2022.

[20] H. Azis, P. Purnawansyah, F. Fattah dan I. P. Putri, "Performa Klasifikasi K-NN dan Cross Validation Pada Data Pasien Pengidap Penyakit Jantung", Ilk. J. Ilm., vol. 12, no. 2, pp. 81–86a, doi: 10.33096/ilkom.v12i2.507.81-86, 2020.

[21] Iqbal Fathur Rahman, "Implementasi Metode SVM, MLP dan XGBOOST pada Data Ekpresi Gen (Studi Kasus: Klasifikasi Data Ekspresi Gen Skeletal Muscle NGT, IGT dan Diabetes Melitus Tipe-2 GSE18732) ", Skripsi, S1 Program Studi Statistika Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Islam Indonesia- Yogyakarta , 2020.