

Prediksi Klasifikasi Diabetes Berbasis Faktor Risiko Kesehatan Menggunakan Gradient Boosted Trees

Ihya Bahrul Alam ¹, Hasbi Firmansyah ², Wahyu Asriyani ³

¹Informatika, Universitas Pancasakti Tegal, Indonesia

²Informatika, Universitas Pancasakti Tegal, Indonesia

³Pendidikan Bahasa dan Sastra Indonesia, Universitas Pancasakti Tegal, Indonesia

Email: ¹ihyaalam795@gmail.com, ²hasbifirmansyah@upstegal.ac.id,

³asriyani1409@gmail.com

Intisari

Diabetes melitus merupakan penyakit tidak menular dengan beban kesehatan yang tinggi dan memerlukan pendekatan deteksi dini berbasis faktor risiko. Penelitian ini menyusun model klasifikasi status diabetes menggunakan indikator kesehatan dan demografi yang relatif mudah diperoleh (misalnya hipertensi, kolesterol, BMI, kebiasaan merokok, aktivitas fisik, serta variabel terkait layanan kesehatan). Data bersumber dari Diabetes Health Indicators (hasil cleaning survei CDC BRFSS 2015) yang tersedia publik, dengan total 253.680 instance dan label biner (0/1). Pemodelan dilakukan menggunakan algoritma Gradient Boosted Trees pada RapidMiner/Altair AI Studio, dan kinerja dievaluasi menggunakan accuracy, weighted mean precision, weighted mean recall, serta classification error. Pada pengujian data uji 20% (2.000 observasi), model mencapai accuracy 71,25% dengan classification error 28,75%. Confusion matrix menunjukkan TN=1.212, TP=213, FP=431, dan FN=144, menandakan kesalahan dominan berupa false positive yang berdampak pada rendahnya presisi kelas positif. Nilai weighted mean precision dan weighted mean recall masing-masing sebesar 61,23% dan 66,72%, sehingga performa keseluruhan tergolong cukup, tetapi ketepatan identifikasi kelas “true” masih perlu ditingkatkan melalui optimasi parameter dan strategi penanganan ketidakseimbangan kelas.

Kata kunci: diabetes melitus, faktor risiko, gradient boosted trees, klasifikasi, pembelajaran mesin, rapidminer.

Abstract

Diabetes mellitus is a major non-communicable disease, and early screening based on observable risk factors can support timely interventions. This study develops a binary classification model to predict diabetes status using health and demographic indicators commonly collected in population surveys (e.g., hypertension, cholesterol, BMI, smoking behavior, physical activity, and healthcare-related variables). The data come from the publicly available Diabetes Health Indicators dataset, derived from the CDC BRFSS 2015 survey, comprising 253,680 instances with a 0/1 target label. A Gradient Boosted Trees (GBT) classifier is implemented in RapidMiner/Altair AI Studio and evaluated using accuracy, weighted mean precision, weighted mean recall, and classification error. Using a 20% test split (2,000 observations), the model achieves 71.25% accuracy with a 28.75% classification error. The confusion matrix (TN=1,212; TP=213; FP=431; FN=144) indicates that false positives dominate the errors, which substantially lowers the positive-class

Prediksi Klasifikasi Diabetes Berbasis Faktor Risiko Kesehatan Menggunakan Gradient Boosted Trees (Ihya Bahrul Alam, Hasbi Firmansyah, Wahyu Asriyani)

precision. The weighted mean precision and weighted mean recall are 61.23% and 66.72%, respectively, suggesting acceptable overall performance but highlighting the need to improve the reliability of positive predictions through hyperparameter tuning and class-imbalance mitigation.

Keywords: *classification, data mining, diabetes mellitus, gradient boosted trees, machine learning, RapidMiner.*

PENDAHULUAN

Diabetes melitus merupakan salah satu penyakit tidak menular yang berkontribusi besar terhadap morbiditas, mortalitas, dan beban pembiayaan kesehatan. Kompleksitas diabetes tidak hanya terletak pada kondisi hiperglikemia, tetapi juga pada berbagai komplikasi jangka panjang seperti penyakit kardiovaskular, gangguan ginjal, neuropati, dan penurunan kualitas hidup. Dalam konteks kesehatan masyarakat, tantangan yang sering muncul adalah deteksi dini dan pengendalian faktor risiko—karena banyak individu berada pada spektrum risiko (misalnya prediabetes) sebelum berkembang menjadi kondisi yang lebih berat [1], [2], [3].

Berbagai studi epidemiologi menunjukkan bahwa diabetes berkaitan erat dengan faktor risiko yang dapat diamati melalui indikator kesehatan dan perilaku, seperti indeks massa tubuh (BMI), hipertensi, kolesterol tinggi, aktivitas fisik, kebiasaan merokok, pola konsumsi, serta akses layanan kesehatan. Indikator-indikator tersebut relatif mudah diperoleh melalui survei populasi, sehingga berpotensi digunakan untuk membangun sistem skrining berbasis data yang mendukung keputusan awal (*early warning*) secara lebih cepat dan terukur [4], [5].

Pemanfaatan pembelajaran mesin (*machine learning*) menjadi relevan karena pendekatan ini mampu menangkap hubungan nonlinier dan interaksi antarvariabel yang kompleks, yang sering kali sulit dimodelkan secara memadai dengan pendekatan konvensional. Pada permasalahan klasifikasi diabetes, tujuan utama pemodelan adalah mengelompokkan individu ke dalam kategori status diabetes berdasarkan kombinasi faktor risiko kesehatan. Kualitas model sangat ditentukan oleh pemilihan algoritma, prosedur evaluasi, dan kesesuaian metrik performa dengan karakteristik data [6], [7], [8], [9].

Penelitian ini menggunakan dataset Diabetes Health Indicators yang tersedia secara publik melalui Kaggle. Dataset tersebut merupakan hasil *cleaning* dari data survei CDC BRFSS 2015 (*Behavioral Risk Factor Surveillance System*) dan disediakan dalam beberapa berkas untuk kebutuhan pemodelan (misalnya versi target biner maupun versi 3-kelas) [10].

Dataset BRFSS sendiri adalah survei kesehatan berbasis telepon yang mengumpulkan data populasi dewasa (≥ 18 tahun) di Amerika Serikat terkait perilaku berisiko, kondisi kesehatan kronis, serta penggunaan layanan kesehatan preventif [10].

Secara operasional, penelitian ini memanfaatkan variabel-variabel indikator kesehatan sebagai prediktor untuk membangun model klasifikasi status diabetes. Pada versi dataset biner yang umum digunakan, target label direpresentasikan sebagai dua kelas (misalnya “tidak diabetes” vs “prediabetes/diabetes”), sementara fitur mencakup kumpulan indikator kesehatan dan demografi yang ringkas untuk kebutuhan *machine learning* [1], [2], [5].

Tersedianya varian dataset—termasuk versi berukuran besar (ratusan ribu respon) dan versi seimbang—menjadikan dataset ini relevan untuk eksperimen klasifikasi sekaligus pembahasan metodologi evaluasi [11], [12].

Metode yang digunakan dalam penelitian ini adalah Gradient Boosted Trees (GBT) untuk tugas klasifikasi. GBT merupakan metode ensemble yang

membangun model secara bertahap melalui penambahan pohon keputusan (decision trees) yang tiap tahapnya berfokus pada perbaikan kesalahan model sebelumnya. Pendekatan ini dikenal efektif untuk data tabular dan mampu memodelkan pola nonlinier serta interaksi fitur dengan baik. Dalam implementasi menggunakan RapidMiner, GBT umumnya dijalankan melalui operator Gradient Boosted Trees (H2O) yang mengeksekusi algoritma boosting pada engine H2O [8], [7], [13], [3].

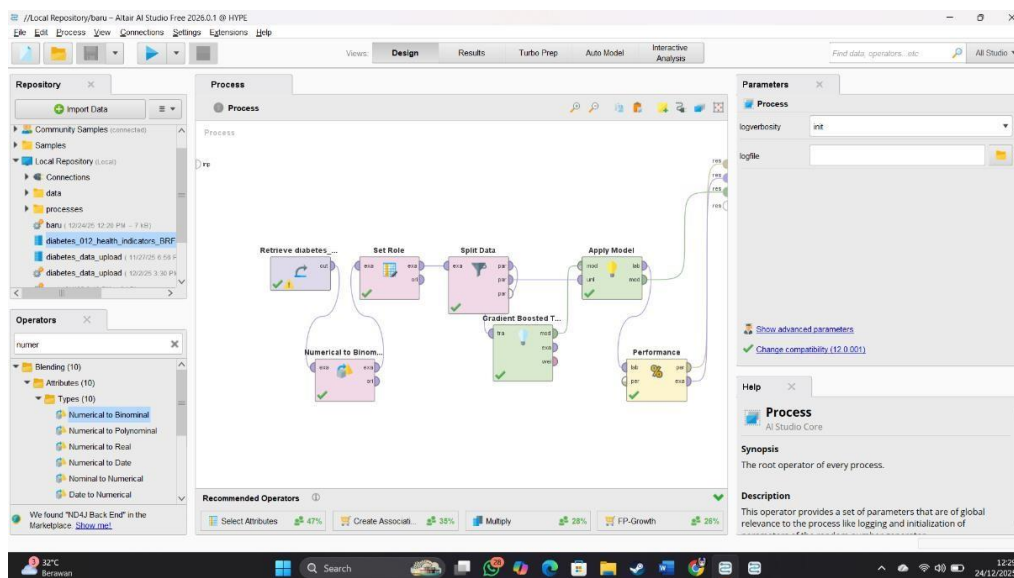
Untuk menilai kinerja model, penelitian ini menggunakan metrik evaluasi yang sesuai dengan kebutuhan klasifikasi, yaitu accuracy, weighted mean precision, weighted mean recall, dan classification error. Penggunaan metrik berbobot (weighted) penting terutama ketika distribusi kelas tidak sepenuhnya seimbang, karena penilaian performa menjadi lebih representatif terhadap keseluruhan data dan tidak hanya didominasi oleh kelas mayoritas. Selain itu, classification error digunakan sebagai ukuran komplementer untuk menggambarkan proporsi kesalahan prediksi model [14], [12], [11].

Dengan demikian, fokus penelitian ini adalah menyusun dan mengevaluasi model klasifikasi diabetes berbasis faktor risiko kesehatan menggunakan Gradient Boosted Trees, serta mendokumentasikan kinerja model melalui metrik evaluasi yang telah ditetapkan. Diharapkan hasil penelitian dapat memberikan kontribusi akademik berupa bukti empiris performa GBT pada data indikator kesehatan, sekaligus kontribusi praktis berupa kerangka pemodelan yang dapat direplikasi untuk studi skrining risiko berbasis data survei kesehatan populasi [1], [8].

METODE PENELITIAN

1. Desain Penelitian

Penelitian ini menggunakan pendekatan kuantitatif dengan metode data mining untuk membangun model klasifikasi status diabetes berdasarkan faktor risiko kesehatan. Tahapan utama meliputi pengumpulan data, prapemrosesan, pemodelan menggunakan Gradient Boosted Trees, serta evaluasi kinerja model menggunakan metrik accuracy, weighted mean precision, weighted mean recall, dan classification error [8], [13], [3], [14], [12].



Gambar 1. Alur Pemodelan

2. Sumber dan Karakteristik Data

Data yang digunakan berasal dari dataset publik Diabetes Health Indicators yang tersedia di platform Kaggle, yang merupakan hasil pengolahan (cleaning dan seleksi variabel) dari survei CDC BRFSS 2015 (Behavioral Risk Factor Surveillance System) [10].

Dataset berisi indikator kesehatan dan demografi (misalnya tekanan darah tinggi, kolesterol, BMI, kebiasaan merokok, aktivitas fisik, dan variabel terkait layanan kesehatan) yang digunakan sebagai atribut prediktor. Label/target klasifikasi pada penelitian ini mengikuti berkas biner yang umum digunakan, yaitu status diabetes dalam dua kelas (0/1) [4], [5], [1], [2].

Tabel 1. Ringkasan dataset

Komponen	Keterangan
Nama dataset	CDC Diabetes Health Indicators
Repository	UCI Machine Learning Repository
Bidang	Health and Medicine
Tipe fitur	Tabular, Multivariate
Tipe fitur	Categorical, Integer
Jumlah instance	253.680

3. Prapemrosesan Data

Prapemrosesan dilakukan untuk memastikan data siap digunakan oleh algoritma klasifikasi. Langkah-langkah yang dilakukan adalah:

- a. Import data ke RapidMiner dari file CSV dan verifikasi struktur atribut serta tipe data.

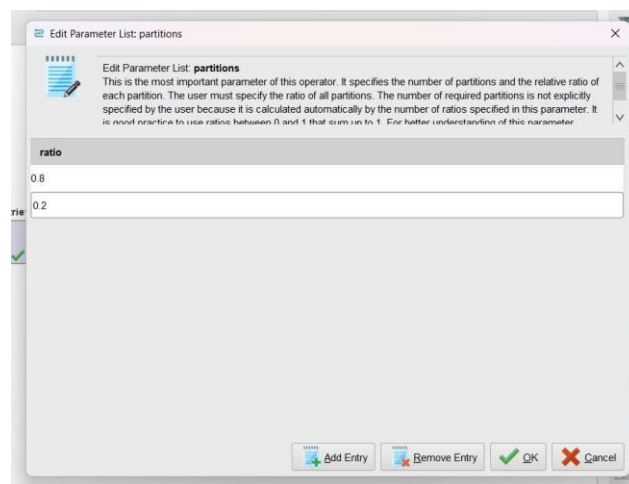
Prediksi Klasifikasi Diabetes Berbasis Faktor Risiko Kesehatan Menggunakan Gradient Boosted Trees (Ihya Bahrul Alam, Hasbi Firmansyah, Wahyu Asriyani)

- b. Penetapan peran atribut (Set Role) dengan menetapkan atribut target (mis. Diabetes_binary) sebagai label, sedangkan atribut lainnya sebagai regular attributes.
- c. Penanganan data hilang (jika ada) menggunakan operator Replace Missing Values dengan strategi pengisian yang sesuai tipe atribut (mis. nilai modus untuk biner/kategorikal dan rata-rata/median untuk numerik).
- d. Validasi tipe label: jika label terbaca sebagai numerik murni, label dikonversi menjadi binominal agar RapidMiner menjalankan mode classification secara benar.

Langkah prapemrosesan ini bertujuan meminimalkan bias akibat kualitas data serta menjaga konsistensi input model [14], [12], [11].

4. Pembagian Data dan Skema Validasi

Untuk memperoleh evaluasi performa yang lebih stabil dibanding satu kali pembagian data, penelitian ini menggunakan k-fold cross-validation (umumnya 10-fold) dengan shuffle aktif. Pada skema ini, data dibagi menjadi k subset (fold); setiap iterasi menggunakan k-1 fold sebagai data latih dan 1 fold sebagai data uji, lalu metrik kinerja dirata-ratakan dari seluruh iterasi. Skema ini mengurangi ketergantungan hasil pada satu pembagian data tertentu dan meningkatkan reliabilitas estimasi performa model [9], [14], [12].



Gambar 2.

Untuk memperoleh estimasi performa model yang lebih stabil dan representatif, penelitian ini menerapkan k-fold cross-validation (umumnya 10-fold) dengan opsi shuffle aktif. Pendekatan ini dipilih karena evaluasi berbasis satu kali pembagian data (hold-out) cenderung sensitif terhadap komposisi data latih dan data uji, sehingga nilai performa dapat berubah apabila pembagian data diulang dengan urutan atau pengacakan yang berbeda. Dengan cross-validation, evaluasi tidak bergantung pada satu split tertentu, melainkan memanfaatkan beberapa skenario pembagian data yang lebih menyeluruh [9].

Pada skema k-fold cross-validation, seluruh dataset dibagi menjadi k subset (fold) berukuran relatif sama. Proses pengujian dilakukan sebanyak k iterasi, di mana pada setiap iterasi satu fold digunakan sebagai data uji, sedangkan k-1 fold lainnya digunakan sebagai data latih. Dengan mekanisme ini, setiap data akan berperan sebagai data uji tepat satu kali dan sebagai data latih pada iterasi lainnya, sehingga evaluasi kinerja model mencerminkan perilaku model pada berbagai kombinasi data dan tidak terfokus pada subset tertentu saja [9].

Pengaktifan shuffle bertujuan untuk mengacak urutan data sebelum pembentukan fold, sehingga distribusi data pada tiap fold lebih merata dan mengurangi risiko adanya pola urutan data yang mempengaruhi hasil evaluasi. Pada setiap iterasi, model dilatih menggunakan data latih dan kemudian dievaluasi pada data uji untuk memperoleh nilai metrik seperti accuracy, weighted mean precision, weighted mean recall, dan classification error. Seluruh nilai metrik dari tiap fold kemudian dirata-ratakan sebagai performa akhir yang dilaporkan, sehingga hasil evaluasi menjadi lebih reliabel dan lebih dapat digeneralisasi [14], [12], [9].

5. Pemodelan Menggunakan Gradient Boosted Trees

Model klasifikasi dibangun menggunakan algoritma Gradient Boosted Trees (GBT), yaitu metode ensemble boosting yang membangun sejumlah pohon keputusan secara bertahap untuk memperbaiki kesalahan prediksi pada iterasi sebelumnya. Implementasi dilakukan di RapidMiner menggunakan operator Gradient Boosted Trees (umumnya tersedia sebagai varian berbasis H2O) [8], [13], [7], [3].

Parameter model ditetapkan pada nilai awal (baseline) dan dapat ditingkatkan melalui proses tuning. Parameter yang umumnya digunakan dalam penelitian ini meliputi [8]:

- a. Number of trees: jumlah pohon dalam ensemble.
- b. Max depth: kedalaman maksimum tiap pohon.
- c. Learning rate: laju pembelajaran pada boosting.

Penyesuaian parameter dilakukan untuk memperoleh kombinasi yang menghasilkan metrik evaluasi terbaik pada skema validasi yang digunakan [8], [9].

6. Metode Evaluasi Kinerja Model

Evaluasi kinerja model dalam penelitian ini dilakukan secara sistematis selama proses cross-validation, yaitu dengan menghitung nilai performa pada setiap iterasi (fold) pengujian. Pada setiap fold, model yang telah dilatih pada data latih diuji menggunakan data uji, kemudian hasil prediksi dibandingkan dengan label aktual untuk memperoleh ukuran kinerja. Untuk memastikan penilaian performa bersifat komprehensif dan merepresentasikan kemampuan model secara keseluruhan, penelitian ini menggunakan empat metrik evaluasi

utama, yaitu accuracy, weighted mean precision, weighted mean recall, dan classification error, yang dijelaskan sebagai berikut [14], [12], [11], [9] :

a. Accuracy

Mengukur proporsi prediksi yang benar terhadap seluruh data uji.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

Keterangan:

Accuracy : proporsi prediksi model yang benar terhadap seluruh jumlah data.

TP : jumlah data positif yang diprediksi positif dengan benar.

TN : jumlah data negatif yang diprediksi negatif dengan benar.

FP : jumlah data negatif yang diprediksi positif (kesalahan tipe I).

FN : jumlah data positif yang diprediksi negatif (kesalahan tipe II).

TP + TN + FP + FN : total seluruh data/prediksi (total sampel).

Accuracy memberikan gambaran umum performa, namun dapat bias bila distribusi kelas tidak seimbang [14], [12].

b. Weighted Mean Precision

Mengukur ketepatan prediksi per kelas, lalu dirata-ratakan berdasarkan proporsi (support) masing-masing kelas.

$$\text{Precision}_i = \frac{TP_i}{TP_i + FP_i} \quad (2)$$

Keterangan:

Precision_i : nilai presisi untuk kelas ke- *i* yaitu seberapa banyak prediksi kelas *i* yang benar

TP_i : jumlah data yang sebenarnya kelas *i* dan diprediksi sebagai kelas *i*.

FP_i : jumlah data yang sebenarnya bukan kelas *i* tetapi diprediksi sebagai kelas *i*.

TP_i + FP_i : total data yang diprediksi sebagai kelas *i*.

$$\text{Weighted Precision} = \sum_{i=1}^c \left(\frac{n_i}{N} \right) \times \text{Precision}_i \quad (3)$$

Keterangan:

Weighted Precision : rata-rata presisi semua kelas yang dibobotkan berdasarkan jumlah data pada tiap kelas.

C : jumlah total kelas (jumlah kategori dalam klasifikasi multi-class).

n_i : jumlah data (support) pada kelas ke-*i* (banyaknya sampel yang benar-benar termasuk kelas *i*).

N : jumlah seluruh data, dengan $N = \sum_{i=1}^c n_i$

$\frac{n_i}{N}$: bobot untuk kelas *i* (proporsi data kelas *i* terhadap total data).

Precision_i : nilai presisi pada kelas *i* (sesuai rumus pertama) [14], [12], [11].

c. Weighted Mean Recall

Mengukur kemampuan model menangkap anggota kelas yang sebenarnya, lalu dirata-ratakan berdasarkan proporsi kelas.

$$\text{Recall}_i = \frac{TP_i}{TP_i + FN_i} \quad (4)$$

Recall_i : nilai recall untuk kelas ke- i , yaitu kemampuan model menemukan data yang benar-benar termasuk kelas i .

TP_i : jumlah data yang sebenarnya kelas i dan diprediksi sebagai kelas i (prediksi benar).

FN_i : jumlah data yang sebenarnya kelas i tetapi diprediksi sebagai kelas lain (gagal terdeteksi sebagai kelas i).

$TP_i + FN_i$: total jumlah data yang sebenarnya termasuk kelas i (total aktual kelas i).

$$\text{Weighted Recall} = \sum_{i=1}^c \left(\frac{n_i}{N} \right) \times \text{Recall}_i \quad (5)$$

Weighted Recall : nilai rata-rata recall dari seluruh kelas yang dibobotkan berdasarkan jumlah data pada masing-masing kelas, sehingga kelas dengan jumlah data lebih banyak memberi kontribusi lebih besar terhadap nilai akhir.

C : jumlah total kelas.

n_i : jumlah sampel (support) pada kelas ke- i (banyaknya data yang benar-benar termasuk kelas i).

N : jumlah total sampel, dengan

$$N = \sum_{i=1}^c n_i \quad (6)$$

$\frac{n_i}{N}$: bobot kelas ke- i (proporsi data kelas i terhadap total data).

Recall_i : nilai recall untuk kelas ke- i , dengan

$$\text{Recall}_i = \frac{TP_i}{TP_i + FN_i}$$

TP_i : jumlah data kelas i yang diprediksi benar sebagai kelas i .

FN_i : jumlah data kelas i yang salah diprediksi sebagai kelas lain [14], [12], [11].

d. Classification Error

Mengukur proporsi kesalahan prediksi (kebalikan dari accuracy).

$$\text{Classification Error} = 1 - \text{Accuracy}$$

Pada penelitian ini, penggunaan metrik berbobot (weighted) dimaksudkan agar evaluasi tetap representatif terhadap seluruh kelas, khususnya bila terdapat perbedaan jumlah contoh antar kelas. Nilai akhir

metrik dilaporkan sebagai rata-rata (dan bila diperlukan deviasi standar) dari seluruh fold validasi [14], [12], [11], [9].

7. Implementasi di RapidMiner

Implementasi proses dilakukan dengan alur operator berikut:

- a. Retrieve/Read CSV → memuat dataset,
- b. Set Role → menetapkan label,
- c. (Opsional) Replace Missing Values,
- d. Cross Validation:
 - 1) Subproses Training: Gradient Boosted Trees,
 - 2) Subproses Testing: Apply Model → Performance (Classification).

Keluaran utama berupa nilai metrik accuracy, weighted mean precision, weighted mean recall, dan classification error, serta model terlatih (bila diperlukan untuk analisis lanjutan) [8], [9].

8. Ringkasan Alur Penelitian

Secara ringkas, penelitian ini mengolah dataset indikator kesehatan, melakukan prapemrosesan, melatih model klasifikasi menggunakan Gradient Boosted Trees, kemudian mengevaluasi performa model menggunakan empat metrik utama (accuracy, weighted mean precision, weighted mean recall, classification error) dalam skema validasi k-fold cross-validation. Output penelitian berupa performa model dan interpretasi hasil sebagai dasar pembahasan efektivitas GBT dalam klasifikasi diabetes berbasis faktor risiko kesehatan [8], [13], [14], [12], [9], [10].

HASIL DAN PEMBAHASAN

Berdasarkan hasil pengujian pada data uji (20% dari total data), tabel confusion matrix menunjukkan empat komponen utama yang menjadi dasar semua metrik evaluasi. Model menghasilkan

Tabel 2. Tabel confusion matrix

TN	FP	FN	TP	Total Observasi Uji
1.212	431	144	213	2.000

Artinya, model benar mengklasifikasikan 1.212 data sebagai false (pred. false dan true false), serta 213 data sebagai true (pred. true dan true true). Di sisi lain, model melakukan kesalahan dengan memprediksi true padahal sebenarnya false sebanyak 431 kasus (false positive), dan memprediksi false padahal sebenarnya true sebanyak 144 kasus (false negative).

Selain itu, tabel juga menunjukkan bahwa distribusi kelas pada data uji tidak seimbang: jumlah kasus true false (kelas false) adalah 1.643 (1.212 + 431), sedangkan jumlah kasus true true (kelas true) adalah 357 (144 + 213). Kondisi ini

penting karena pada data yang tidak seimbang, metrik seperti accuracy dapat terlihat cukup tinggi meskipun performa pada kelas minoritas belum optimal.

1. Hasil Evaluasi Accuracy

Accuracy yang diperoleh adalah 71,25%.

Dari tabel:

Tabel 3. Tabel accuracy

	True False	True True	Class precision
Pred. False	1212	144	89.39%
Pred. True	431	213	33.07%
Class Recall	73.77%	59.66%	

Accuracy dihitung dari semua prediksi yang benar, yaitu $TN + TP = 1.212 + 213 = 1.425$. Karena total data uji 2.000, maka $accuracy = 1.425/2.000 = 71,25\%$.

Maknanya: model mampu memberikan prediksi yang benar pada sekitar 7 dari 10 data uji. Namun, karena kelas false mendominasi (1.643 dari 2.000), accuracy yang cukup tinggi ini sebagian besar “ditopang” oleh kemampuan model dalam mengklasifikasikan kelas false dengan benar (TN besar). Dengan kata lain, accuracy yang baik belum otomatis berarti model sudah kuat dalam mengenali kelas true.

2. Hasil Evaluasi Classification Error

Classification error adalah 28,75%.

Dari tabel:

Tabel 4. Tabel evaluasi classification Error

	True False	True True	Class precision
Pred. False	1212	144	89.39%
Pred. True	431	213	33.07%
Class Recall	73.77%	59.66%	

Jumlah prediksi yang salah adalah $FP + FN = 431 + 144 = 575$. Karena total data uji 2.000, maka $error = 575/2.000 = 28,75\%$.

Maknanya: sekitar 3 dari 10 prediksi model masih salah. Yang menarik, kesalahan lebih banyak berasal dari false positive (431) dibanding false negative (144). Ini menunjukkan karakter model yang cenderung “lebih sering” memprediksi kelas true pada data yang sebenarnya false (over-detection), sehingga meningkatkan jumlah alarm yang tidak tepat.

3. Hasil Evaluasi Precision

a. Precision Kelas False

Precision untuk kelas false adalah 89,38%.

Dari tabel:

Tabel 5. precision false

	True False	True True	Class precision
Pred. False	1212	144	89.39%
Pred. True	431	213	33.07%
Class Recall	73.77%	59.66%	

Ketika model memprediksi false (baris pred. false), totalnya $1.212 + 144 = 1.356$ prediksi. Dari jumlah itu, yang benar-benar false adalah 1.212. Jadi $\text{precision}(\text{false}) = 1.212/1.356 = 89,38\%$.

Maknanya: setiap kali model mengatakan “false”, prediksi itu umumnya benar. Ini menandakan model cukup kuat dalam menjaga agar prediksi false tidak banyak “terkontaminasi” oleh kasus true (FN relatif lebih kecil dibanding TN dalam baris pred. false).

b. Precision Kelas True

Precision untuk kelas true adalah 33,07%

Dari tabel:

Tabel 6. Pprecision true

	True False	True True	Class precision
Pred. False	1212	144	89.39%
Pred. True	431	213	33.07%
Class Recall	73.77%	59.66%	

Ketika model memprediksi true (baris pred. true), totalnya $431 + 213 = 644$ prediksi. Dari jumlah itu, yang benar-benar true adalah 213. Jadi $\text{precision}(\text{true}) = 213/644 = 33,07\%$.

Maknanya: ini menunjukkan persoalan utama model Anda: ketika model menyatakan “true”, hanya sekitar 33 dari 100 yang benar-benar true, sedangkan sisanya adalah false positive. Ini konsisten dengan angka $FP = 431$ yang jauh lebih besar daripada $TP = 213$. Dengan kata lain, model cukup sering “menuduh true” pada kasus yang sebenarnya false.

4. Hasil Evaluasi Recall

a. Recall kelas false

Recall untuk kelas false adalah 73,77%.

Dari tabel:

Tabel 7. Recall false

	True False	True True	Class precision
Pred. False	1212	144	89.39%
Pred. True	431	213	33.07%
Class Recall	73.77%	59.66%	

total data yang benar-benar false adalah $1.212 + 431 = 1.643$. Dari jumlah itu, yang berhasil ditangkap sebagai false (TN) adalah 1.212. Jadi $\text{recall}(\text{false}) = 1.212/1.643 = 73,77\%$.

Maknanya: model berhasil mendeteksi sekitar 74% dari semua kasus false. Sisanya (sekitar 26%) “tergeser” menjadi prediksi true (FP), terlihat dari FP yang cukup besar (431).

b. Recall kelas true

Recall untuk kelas true adalah 59,66%.

Dari tabel:

Tabel 8. Recall true

	True False	True True	Class precision
Pred. False	1212	144	89.39%
Pred. True	431	213	33.07%
Class Recall	73.77%	59.66%	

total data yang benar-benar true adalah $144 + 213 = 357$. Dari jumlah itu, yang berhasil ditangkap sebagai true (TP) adalah 213. Jadi $\text{recall}(\text{true}) = 213/357 = 59,66\%$.

Maknanya: model mampu menangkap sekitar 60% kasus true, tetapi masih melewatkan sekitar 40% kasus true sebagai false negative (FN = 144). Jika kelas true merepresentasikan kondisi risiko (misalnya diabetes/prediabetes), maka FN ini perlu menjadi perhatian karena berarti ada individu berisiko yang tidak terdeteksi oleh model.

5. Hasil Evaluasi Weighted Mean Precision dan Weighted Mean Recall

Pada output Anda tertulis weights: 1,1, yang berarti rata-rata dihitung setara antar dua kelas (kelas false dan true diberi bobot sama). Karena itu:

- Weighted mean precision 61,23% berasal dari rata-rata precision kelas false (89,38%) dan precision kelas true (33,07%). Nilai ini “turun” terutama karena precision kelas true sangat rendah, akibat FP yang tinggi.
- Weighted mean recall 66,72% berasal dari rata-rata recall kelas false (73,77%) dan recall kelas true (59,66%). Nilai ini menunjukkan kemampuan model menangkap anggota kelas secara rata-rata berada pada tingkat sedang.

Maknanya: dua metrik ini memberi gambaran lebih adil antar kelas dibanding accuracy. Hasil Anda menunjukkan bahwa performa kelas mayoritas (false) cukup kuat, namun performa kelas minoritas (true) masih menjadi titik lemah—khususnya dari sisi precision (banyak false positive).

Secara keseluruhan, model Gradient Boosted Trees menghasilkan akurasi 71,25% dengan classification error 28,75%. Confusion matrix mengindikasikan bahwa model lebih baik dalam mengklasifikasikan kelas false (TN = 1.212, precision 89,38%, recall 73,77%) dibanding kelas true (TP = 213,

precision 33,07%, recall 59,66%). Jumlah false positive yang relatif tinggi (FP = 431) menyebabkan rendahnya precision pada kelas true dan menurunkan weighted mean precision menjadi 61,23%. Sementara itu, keberadaan false negative (FN = 144) menunjukkan masih terdapat kasus true yang tidak terdeteksi, sehingga recall kelas true berada pada 59,66% dan weighted mean recall sebesar 66,72%. Temuan ini menegaskan bahwa peningkatan performa terutama diperlukan untuk mengurangi false positive dan meningkatkan ketepatan identifikasi kelas true.

KESIMPULAN

Penelitian ini bertujuan membangun model klasifikasi status diabetes berbasis faktor risiko kesehatan menggunakan algoritma Gradient Boosted Trees (GBT) pada Altair AI Studio/RapidMiner. Proses pemodelan dilakukan dengan alur: pemanggilan data dari repository, penetapan label, pembagian data menggunakan skema hold-out 80% data latih dan 20% data uji, pelatihan model GBT, penerapan model pada data uji, dan evaluasi kinerja menggunakan metrik accuracy, weighted mean precision, weighted mean recall, serta classification error. Hasil evaluasi menunjukkan bahwa model yang dibangun memiliki kemampuan prediksi yang cukup baik pada tingkat keseluruhan, namun masih terdapat kelemahan signifikan pada ketepatan prediksi kelas positif (true).

Berdasarkan hasil pengujian pada data uji sebanyak 2.000 observasi, model menghasilkan accuracy sebesar 71,25% dan classification error sebesar 28,75%. Nilai ini menunjukkan bahwa dari seluruh data uji, model mampu mengklasifikasikan secara benar sebanyak 1.425 kasus dan masih melakukan kesalahan pada 575 kasus. Secara umum, akurasi yang diperoleh menandakan model cukup efektif dalam memetakan pola faktor risiko kesehatan terhadap status diabetes, namun angka error yang masih mendekati sepertiga data uji mengindikasikan bahwa performa model belum optimal untuk digunakan tanpa perbaikan lebih lanjut.

Analisis lebih mendalam melalui confusion matrix memperlihatkan bahwa performa model tidak merata antar kelas. Model menghasilkan True Negative (TN) = 1.212 dan True Positive (TP) = 213, sedangkan kesalahan prediksi terdiri atas False Positive (FP) = 431 dan False Negative (FN) = 144. Komposisi ini menunjukkan bahwa kesalahan paling dominan adalah false positive, yaitu model cukup sering memprediksi kelas true pada data yang sebenarnya false. Hal ini berdampak langsung pada rendahnya ketepatan prediksi kelas true.

Dari sisi metrik berbasis kelas, model memiliki precision kelas false sebesar 89,38% dan recall kelas false sebesar 73,77%, yang menunjukkan bahwa model relatif kuat dalam mengidentifikasi kelas mayoritas (false). Sebaliknya, untuk kelas true diperoleh precision sebesar 33,07% dan recall sebesar 59,66%. Ini berarti ketika model memprediksi “true”, hanya sekitar sepertiga prediksi yang benar, sementara dua pertiganya merupakan alarm palsu. Selain itu, recall kelas true yang berada di kisaran 60% menunjukkan bahwa masih terdapat sekitar 40% kasus true yang belum berhasil ditangkap oleh model (terlihat dari FN = 144), yang menjadi

catatan penting apabila penelitian diarahkan untuk mendukung skrining atau deteksi dini.

Nilai *weighted mean precision* sebesar 61,23% dan *weighted mean recall* sebesar 66,72% (dengan bobot setara antar kelas) memperkuat temuan bahwa performa model secara rata-rata antar kelas berada pada tingkat sedang. Nilai rata-rata *precision* yang lebih rendah dibanding *recall* mengindikasikan bahwa masalah utama model terletak pada ketepatan ketika menetapkan kelas true (banyak false positive), bukan semata-mata kemampuan menangkap kasus true. Dengan demikian, meskipun model memiliki akurasi keseluruhan yang cukup, kualitas prediksi pada kelas true masih perlu ditingkatkan agar model lebih selektif dan efisien.

Secara keseluruhan, penelitian ini menyimpulkan bahwa Gradient Boosted Trees mampu memberikan performa klasifikasi yang cukup baik dalam memodelkan status diabetes berdasarkan indikator risiko kesehatan, dengan akurasi 71,25%. Namun, model masih menunjukkan kecenderungan menghasilkan false positive yang tinggi, yang menyebabkan *precision* kelas true rendah (33,07%). Untuk pengembangan lanjutan, disarankan dilakukan optimasi hyperparameter (misalnya meningkatkan jumlah pohon ketika *learning rate* rendah), mempertimbangkan penanganan ketidakseimbangan kelas (*sampling/weighting*), serta mengganti atau menambah prosedur evaluasi dengan *k-fold cross-validation* agar hasil lebih stabil dan reliabel untuk kebutuhan publikasi ilmiah. Dengan perbaikan tersebut, diharapkan model dapat menghasilkan keseimbangan yang lebih baik antara ketepatan dan kemampuan deteksi, khususnya pada kelas true yang menjadi fokus utama dalam konteks risiko diabetes.

SARAN

Berdasarkan hasil penelitian, disarankan agar evaluasi model pada penelitian selanjutnya tidak hanya mengandalkan pembagian data hold-out (misalnya 80/20), karena metode ini sangat dipengaruhi oleh satu kali pembagian data dan berpotensi menghasilkan performa yang berbeda apabila dilakukan pengacakan ulang. Untuk memperoleh estimasi performa yang lebih stabil dan lebih dapat dipertanggungjawabkan dalam konteks publikasi ilmiah, penelitian berikutnya sebaiknya menggunakan stratified *k-fold cross-validation* sehingga setiap data memiliki kesempatan menjadi data uji dan nilai performa dapat dilaporkan dalam bentuk rata-rata serta variasinya.

Selain itu, performa model pada kelas true masih perlu ditingkatkan karena hasil menunjukkan *precision* kelas true yang rendah dan jumlah false positive yang tinggi. Oleh sebab itu, penelitian lanjutan sebaiknya melakukan optimasi hyperparameter secara sistematis, terutama pada kombinasi *learning rate* dan jumlah pohon. Konfigurasi *learning rate* yang kecil umumnya memerlukan jumlah pohon yang lebih banyak agar model mampu mempelajari pola data secara memadai. Penyesuaian parameter seperti *number of trees*, *maximal depth*, *min rows*, dan *learning rate* melalui prosedur pencarian parameter (*grid* atau

evolutionary search) berpotensi menurunkan false positive sekaligus memperbaiki ketepatan prediksi ketika model menetapkan kelas true.

Penelitian berikutnya juga disarankan untuk memberi perhatian khusus pada isu ketidakseimbangan kelas, karena distribusi kelas yang timpang dapat menyebabkan model cenderung bias dalam menghasilkan prediksi tertentu dan menurunkan kualitas prediksi pada kelas minoritas. Upaya penanganan imbalance seperti oversampling (misalnya SMOTE), undersampling, atau pendekatan cost-sensitive/class weighting dapat dipertimbangkan untuk membantu model belajar lebih baik pada kelas true, sehingga kinerja precision dan recall menjadi lebih seimbang.

Di samping itu, apabila model menghasilkan probabilitas prediksi, penyesuaian ambang keputusan (decision threshold) dapat menjadi strategi untuk mengendalikan trade-off antara precision dan recall. Ambang yang lebih selektif dapat membantu menekan false positive sehingga prediksi true menjadi lebih “bersih”, meskipun konsekuensinya perlu dianalisis karena dapat mempengaruhi jumlah kasus true yang terdeteksi. Oleh karena itu, pemilihan threshold sebaiknya didasarkan pada tujuan penelitian—apakah lebih menekankan minimnya alarm palsu atau meminimalkan kasus terlewat.

Selanjutnya, agar kualitas pelaporan ilmiah semakin kuat, penelitian lanjutan dapat menambahkan metrik evaluasi pendukung yang lebih informatif dalam situasi data tidak seimbang, seperti F1-score atau ROC-AUC, di samping accuracy, weighted precision/recall, dan classification error. Penambahan metrik ini akan memperkaya interpretasi karena memberikan gambaran yang lebih komprehensif tentang keseimbangan ketepatan dan kemampuan deteksi model.

Terakhir, disarankan agar penelitian berikutnya menegaskan konsistensi definisi label yang digunakan (biner atau multi-kelas) serta menyertakan analisis kontribusi faktor risiko (misalnya feature importance) untuk menunjukkan variabel mana yang paling berpengaruh dalam prediksi. Dengan demikian, penelitian tidak hanya menyajikan performa model, tetapi juga memberikan nilai tambah berupa insight yang relevan untuk konteks kesehatan, terutama dalam memahami indikator risiko yang paling berkaitan dengan status diabetes.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada semua pihak yang telah memberikan dukungan dan kontribusi dalam pelaksanaan penelitian ini. Ucapan terima kasih secara khusus disampaikan kepada Program Studi S1 Informatika Universitas Pancasakti Tegal yang telah memberikan fasilitas dan lingkungan akademik yang mendukung proses penelitian. Penulis juga mengucapkan terima kasih kepada dosen pembimbing dan rekan-rekan yang telah memberikan arahan, masukan, serta diskusi konstruktif selama proses penyusunan dan penyelesaian penelitian ini. Selain itu, penulis menyampaikan apresiasi kepada penyedia dataset CDC Diabetes Health Indicators yang telah menyediakan data sehingga penelitian ini dapat terlaksana dengan baik. Semoga hasil penelitian ini dapat memberikan

manfaat dan kontribusi bagi pengembangan ilmu pengetahuan, khususnya di bidang data mining dan informatika kesehatan.

DAFTAR PUSTAKA

- [1] K. D. K. Wardhani and M. Akbar, "Diabetes Risk Prediction Using Extreme Gradient Boosting (XGBoost)," *JOIN (Jurnal Online Informatika)*, vol. 7, no. 2, pp. 244–250, 2022, doi: 10.15575/join.v7i2.970. [Online]. Available: <https://join.if.uinsgd.ac.id/index.php/join/article/view/970>
- [2] K. D. K. W. Wardani and M. Akbar, "Diabetes Risk Prediction Using Feature Importance Extreme Gradient Boosting (XGBoost)," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 7, no. 4, pp. 824–831, 2023, doi: 10.29207/resti.v7i4.4651.
- [3] D. D. Rufo, T. G. Debelee, A. Ibenthal, and W. G. Negera, "Diagnosis of Diabetes Mellitus Using Gradient Boosting Machine (LightGBM)," *Diagnostics*, vol. 11, no. 9, p. 1714, 2021, doi: 10.3390/diagnostics11091714.
- [4] A. O. Nilei, A. S. Oke, and A. G. Ariwayo, "Machine Learning Analysis of Health and Lifestyle Factors in Understanding Diabetes," *Journal of Complementary and Alternative Medical Research*, vol. 25, no. 8, pp. 57–70, 2024, doi: 10.9734/jocamr/2024/v25i8560. [Online]. Available: <https://journaljocamr.com/index.php/JOCAMR/article/view/560>
- [5] B. S. Ahamed, M. S. Arya, S. K. B. Sangeetha, and N. V. Auxilia Osvin, "Diabetes Mellitus Disease Prediction and Type Classification Using Machine Learning," *Applied Computational Intelligence and Soft Computing*, vol. 2022, p. 7899364, 2022, doi: 10.1155/2022/7899364.
- [6] Y. B. D. Setianto, "Comparison of Extreme Gradient Boosting Algorithm and Artificial Neural Network on Diabetes Prediction," *Proxies: Jurnal Informatika*, vol. 5, no. 1, pp. 13–27, 2024, doi: 10.24167/proxies.v5i1.12443.
- [7] S. M. Ganie, P. K. D. Pramanik, M. B. Malik, S. Mallik, and H. Qin, "An Ensemble Learning Approach for Diabetes Prediction Using Boosting Techniques," *Frontiers in Genetics*, vol. 14, p. 1252159, 2023, doi: 10.3389/fgene.2023.1252159.
- [8] M. Kyrychek, "Gradient Boosting as a Tool for Solving Classification Problems in Data-Constrained Environments," *Technologies and Engineering*, vol. 26, no. 2, pp. 37–47, 2025, doi: 10.30857/2786-5371.2025.2.3.
- [9] E. M. Hameed and H. Joshi, "Performance Comparison of Machine Learning Techniques in Prediction of Diabetes Risk," in *AIP Conference Proceedings*, vol. 3051, no. 1, p. 040002, 2024, doi: 10.1063/5.0191611.

- [10] B. Yang, "Diabetes Risk Prediction Model Using Machine Learning," *Transactions on Computer Science and Intelligent Systems Research*, vol. 9, pp. 553–558, 2025, doi: 10.62051/nzr6tw29. [Online]. Available: <https://wepub.org/index.php/TCSISR/article/view/5524>
- [11] I. M. Nur, D. Rosadi, and Abdurakhman, "Multi-Class Imbalance Classification of Diabetes Cases Using Light Gradient Boosting Machine (LGBM)," in *ITM Web of Conferences*, vol. 67, p. 01012, 2024, doi: 10.1051/itmconf/20246701012.
- [12] G. Mary, "Random Oversampling-Based Diabetes Classification via Machine Learning Algorithms," *International Journal of Computational Intelligence Systems*, 2024.
- [13] S. Chettri, "An Ensemble of Light Gradient Boosting Machine and Adaptive Boosting for Type-2 Diabetes Prediction," *International Journal of Computational Intelligence Systems*, vol. 16, p. 14, 2023, doi: 10.1007/s44196-023-00184-y.
- [14] M. H. Saeed, "Diabetes Type 2 Classification Using Machine Learning Algorithms with Up-Sampling Technique," *Journal of Electrical Systems and Information Technology*, vol. 10, p. 8, 2023, doi: 10.1186/s43067-023-00074-5..