Published online on: https://journal.uii.ac.id/jurnalsnati/
DOI: 10.20885/snati.v5.i2.49517
Jurnal Sains, Nalar, dan Aplikasi Teknologi Informasi
Vol. 5 No. 2 (2026) 97 - 107 ISSN Media Electronic: 2807-5935

Evaluasi *SDCA, LBFGS, LightGBM* dan *FastTree* pada *ML.NET* untuk Prediksi Diabetes

Evaluation of SDCA, LBFGS, LightGBM and FastTree in ML.NET for Diabetes Prediction

Resi Taufan[1]*, Fahmi Ardiansyah[2], Annisa Elfina Augustia[3]
[1,2] Program Studi Sistem Informasi, Universitas Bina Sarana Informatika, Jakarta, Indonesia
[3] Program Studi Teknik Informatika, Universitas Indraprasta PGRI, Jakarta, Indonesia
[1*]resi.ret@bsi.ac.id, [2]fahmi.fdy@bsi.ac.id, [3]annisaelfina16@gmail.com

Abstract

This study aims to develop a machine learning-based diabetes risk prediction model using the ML.NET framework. The dataset utilized is a balanced-split version of the 2015 BRFSS, consisting of 70,692 respondents and 21 health indicator variables. Two training approaches were applied to analyze model performance: a baseline with default parameters and hyperparameter tuning. The preprocessing stage involved combining variables into feature vectors, Min-Max normalization, and an 80:20 train-test data split. The models were trained using four algorithms: SDCA Logistic Regression, LBFGS Logistic Regression, LightGBM, and FastTree. Evaluation results showed that LightGBM with the hyperparameter tuning approach, delivered the most consistent performance, achieving 75.37% accuracy, 82.86% AUC, 76.22% F1-score, 72.92% precision, and 79.83% recall. Feature analysis confirmed that GenHlth, HighBP, BMI, HighChol, and Age contributed dominantly to diabetes risk, aligning with medical literature regarding metabolic factors. The best-performing LightGBM model was then integrated into a .NET-based prototype application with a Razor Pages web interface. The practical contribution of this research is proof of concept for machine learning integration into e-health systems to support early detection and digital prevention of diabetes complications in the future.

Keywords: Diabetes; LightGBM; Logistic Regression; Machine Learning; ML.NET

Abstrak

Penelitian ini bertujuan mengembangkan model prediksi risiko diabetes berbasis *machine learning* menggunakan *framework* ML.NET. Dataset yang digunakan adalah versi *balanced split* dari BRFSS 2015 dengan 70.692 responden dan 21 variabel indikator kesehatan. Dua pendekatan pelatihan diterapkan untuk menganalisis performa model: *baseline* dengan parameter standar dan *hyperparameter tuning*. Tahap *preprocessing* meliputi penggabungan variabel ke vektor fitur, normalisasi Min-Max, serta pembagian data *train* dan *test* berasio 80:20. Model dilatih menggunakan empat algoritma: *SDCA Logistic Regression*, *LBFGS Logistic Regression*, *LightGBM*, dan *FastTree*. Hasil evaluasi menunjukkan *LightGBM* dengan pendekatan *hyperparameter tuning* memberikan performa paling konsisten dengan akurasi 75,37%, AUC 82,86%, *F1-score* 76,22%, presisi 72,92%, dan *recall* 79,83%. Analisis fitur menegaskan bahwa variabel *GenHlth, HighBP, BMI, HighChol*, dan *Age* memberikan kontribusi dominan terhadap risiko diabetes, sejalan dengan literatur medis terkait faktor metabolik. Model *LightGBM* merupakan model terbaik untuk diintegrasikan ke dalam prototipe aplikasi berbasis .NET dengan antarmuka *Web Razor Pages*. Kontribusi praktis penelitian ini adalah bukti konsep integrasi *machine learning* ke dalam sistem *e-health* guna mendukung deteksi dini serta pencegahan komplikasi diabetes secara digital di masa depan.

Kata kunci: Diabetes; *LightGBM; Logistic Regression; Machine Learning; ML.NET*

1. Pendahuluan

Diabetes melitus merupakan salah satu penyakit kronis paling umum di dunia dengan jumlah penderita yang terus meningkat setiap tahun, banyak kasus diabetes tidak terdiagnosis pada tahap awal, sehingga deteksi dini dan prediksi risiko menjadi aspek penting dalam upaya pencegahan komplikasi lebih lanjut [1]. Penelitian sebelumnya telah memanfaatkan algoritma *machine learning* untuk melakukan klasifikasi risiko diabetes, pada umumnya menggunakan *framework* populer berbasis Python seperti *Scikit-learn* atau *TensorFlow* [2].

Scikit-learn telah banyak diimplementasikan dalam berbagai penelitian prediksi diabetes karena menyediakan ekosistem komputasi berbasis Python yang fleksibel dan banyak pustaka untuk *preprocessing* data. Dalam hal ini, pemanfaatan algoritma *ensemble* dan model berbasis *supervised learning* menunjukkan hasil performa dan metrik evaluasi yang jauh lebih akurat, stabil, dan adaptif [3].

Received: 31-03-2026 | Revised: 22-06-2026 | Accepted: 03-07-2026 | Published: 15-07-2026

Penelitian lain menggunakan *Random Forest* dan *Support Vector Machine* untuk memprediksi biomarker diabetes tipe dua, hasilnya menunjukkan bahwa algoritma *Random Forest* lebih unggul dengan capaian akurasi keseluruhan sebesar 76% dan nilai AUC mencapai 0,95 [4]. Keunggulan algoritma *Random Forest* pada penelitian dengan dataset lokal terbukti memiliki akurasi paling tinggi jika dibandingkan dengan *Decision Tree* dan *Logistic Regression* [5]. Dalam upaya klasifikasi dan prediksi dini penyakit diabetes mellitus, penelitian lain [6] mengevaluasi beberapa algoritma berbasis *supervised learning* dan *deep learning* menggunakan dataset Pima Indians Diabetes. Eksperimen menunjukkan bahwa model *Multilayer Perceptron* (MLP) menghasilkan akurasi klasifikasi terbaik sebesar 86,08%. Di sisi lain, untuk pemodelan analisis prediktif, algoritma *Long Short-Term Memory* (LSTM) memberikan hasil paling akurat sebesar 87,26%.

Penelitian lain mengevaluasi algoritma *Decision Tree*, *Random Forest*, dan *Gradient Boosting* menggunakan dataset Pima Indians Diabetes. Metode *preprocessing* meliputi penanganan *missing values* dan *balancing data*. *Gradient Boosting* memberikan performa terbaik dengan akurasi tinggi[7]. Penelitian menggunakan dataset Pima Indians Diabetes diimplementasikan dalam penelitian prediksi diabetes, hasil menunjukkan akurasi 87%, menegaskan efektivitas algoritma *ensemble* dalam klasifikasi kesehatan [8]. Dengan dataset Pima Indians, penelitian lain mengevaluasi performa *k-Nearest Neighbors* (KNN*)*, *Support Vector Machine* (SVM) dan *Random Forest* untuk prediksi diabetes dengan hasil menunjukkan SVM unggul dalam presisi [9].

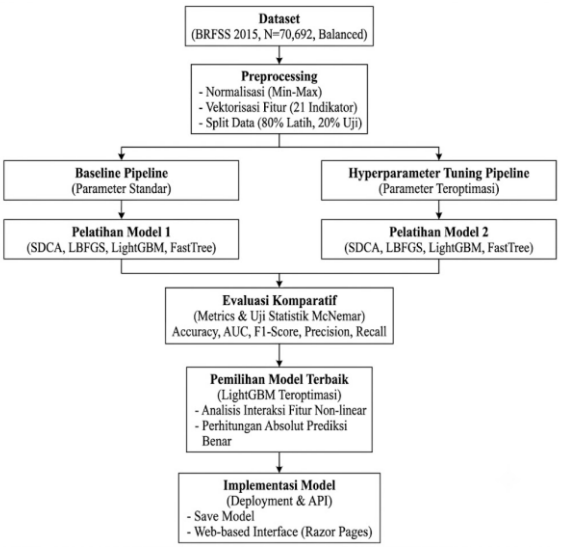
Penelitian sebelumnya juga dapat memanfaatkan model AI untuk prediksi diabetes ke dalam sistem *e-health* untuk mendukung deteksi dini [10]. Penelitian lain [11] menunjukkan bahwa algoritma *Logistic Regression* mampu mencapai akurasi sebesar 78,26% dalam memprediksi diabetes tipe dua. Temuan ini menegaskan bahwa *Logistic Regression* merupakan model yang sederhana namun tetap efektif dan dapat diinterpretasikan.

Berdasarkan penelitian sebelumnya, pemanfaatan ML.NET sebagai *framework machine learning* berbasis .NET masih jarang dieksplorasi dalam konteks akademik. Salah satu keunggulan *framework* ML.NET adalah penyatuan jalur *pipeline*, di mana proses pelatihan dan prediksi berbagi jalur kode yang sama, sehingga model dapat langsung diintegrasikan ke lingkungan produksi tanpa memerlukan modifikasi atau penulisan ulang kode, selain itu jika dibandingkan dengan *framework* Python, *framework* Python memiliki keterbatasan bawaan seperti eksekusi terinterpretasi dan *global interpreter lock* yang membatasi proses paralelisasi, sehingga kurang optimal untuk aplikasi berspesifikasi performa tinggi [12] Maka dari itu penelitian ini menawarkan kebaruan dengan mengevaluasi algoritma yang tersedia pada ML.NET. Permasalahan utama yang diidentifikasi adalah perlunya evaluasi performa algoritma yang tersedia pada ML.NET untuk memastikan kelayakan penggunaannya dalam prediksi kesehatan masyarakat, khususnya diabetes. Penelitian ini bertujuan untuk mengevaluasi dan membandingkan performa empat algoritma utama pada ML.NET, yaitu *SDCA Logistic*

Regression, LBFGS Logistic Regression, LightGBM dan FastTree dengan menggunakan dataset *Diabetes Health Indicators* yang memuat variabel demografi, gaya hidup, dan kondisi kesehatan responden. Kebaruan penelitian ini terletak pada penggunaan ML.NET sebagai alternatif *framework* yang terintegrasi dengan ekosistem .NET, sehingga selain menghasilkan analisis komparatif algoritma, penelitian ini juga menyajikan implementasi tambahan berupa aplikasi berbasis .NET sebagai bukti konsep penerapan model prediksi dalam sistem informasi kesehatan. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi metodologis melalui evaluasi algoritma *machine learning* serta kontribusi praktis melalui integrasi model ke dalam aplikasi yang relevan dengan kebutuhan *e-health*.

2. Metodologi Penelitian

Seluruh tahapan eksperimen dan sistematika pelaksanaan dalam penelitian ini dirancang secara terstruktur, mulai dari pengolahan data hingga implementasi model, yang disajikan secara komprehensif pada diagram alur metode penelitian di Gambar 1.



Gambar 1. Diagram Alur Metode Penelitian

Tabel 1. Spesifikasi Konfigurasi Hyperparameter

Algoritma	*Hyperparameter* Kunci	Nilai Bawaan	Nilai Teroptimasi	Target Optimasi
SDCA	*maximumNumberOfIterations*	200	3000	Menjamin kematangan konvergensi global
LBFGS	*historySize*	20	100	Memaksimalkan presisi pencarian titik minimum
	optimizationTolerance	1e-08	1e-10	fungsi galat
LightGBM	*numberOfIterations*	100	2000	Pencegahan *overfitting* dan penguatan stabilitas
	numberOfLeaves	31	15	AUC
	learningRate	0.1	0.01	Pendekatan pembelajaran konservatif untuk
FastTree	*numberOfTrees*	100	1000	menaikkan *F1-Score*
	learningRate	0.1	0.05	

Tabel 2. Daftar Lengkap 21 Variabel Pada Dataset

No	Variabel	Deskripsi Singkat	Tipe Data
1	*HighBP*	Status tekanan darah tinggi (0 = tidak, 1 = ya)	Kategorikal
2	*HighChol*	Status kolesterol tinggi (0/1)	Kategorikal
3	*CholCheck*	Pemeriksaan kolesterol dalam 5 tahun terakhir	Kategorikal
4	*BMI*	Indeks Massa Tubuh	Numerik
5	*Smoker*	Status merokok seumur hidup	Kategorikal
6	*Stroke*	Pernah mengalami *stroke*	Kategorikal
7	*HeartDisease*	Riwayat penyakit jantung atau serangan jantung	Kategorikal
8	*PhysActivity*	Aktivitas fisik dalam 30 hari terakhir	Kategorikal
9	*Fruits*	Konsumsi buah ≥ 1 kali per hari	Kategorikal
10	*Veggies*	Konsumsi sayur ≥ 1 kali per hari	Kategorikal
11	*HvyAlcoholCons*	Konsumsi alkohol berat (>14 kali/minggu pria, >7 wanita)	Kategorikal
12	*AnyHealthcare*	Memiliki akses layanan kesehatan	Kategorikal
13	*NoDocbcCost*	Tidak ke dokter karena biaya	Kategorikal
14	*GenHlth*	Penilaian kesehatan umum (1 = *excellent*, 5 = *poor*)	Ordinal
15	*MentHlth*	Hari kesehatan mental buruk dalam 30 hari terakhir	Numerik
16	*PhysHlth*	Hari kesehatan fisik buruk dalam 30 hari terakhir	Numerik
17	*DiffWalk*	Kesulitan berjalan atau menaiki tangga	Kategorikal
18	*Sex*	Jenis kelamin (0 = perempuan, 1 = laki-laki)	Kategorikal
19	*Age*	Kelompok usia (13 kategori, 18–80+)	Ordinal
20	*Education*	Tingkat pendidikan (1 = tidak sekolah, 6 = sarjana)	Ordinal
21	*Income*	Tingkat pendapatan (1 = <10.000 USD, 8 = ≥75.000 USD)	Ordinal

Penelitian ini menggunakan dua pendekatan pelatihan model pada varian dataset *balanced-split* guna menganalisis dampak optimasi parameter terhadap akurasi klasifikasi risiko. Pendekatan pertama adalah model *baseline,* yaitu konfigurasi *hyperparameter* standar bawaan *framework* ML.NET. Pendekatan kedua adalah model *hyperparameter tuning*, di mana dilakukan kustomisasi parameter kunci secara terarah pada fungsi *loss model* seperti pengetatan koefisien penalti regularisasi L1/L2 pada skema regresi linear (*SDCA* dan *LBFGS*), serta penyesuaian kapasitas daun pohon keputusan (*numberOfLeaves*) dan laju belajar (*learningRate*) secara spesifik pada skema *tree boosting* (*LightGBM* dan *FastTree*) untuk memaksimalkan stabilitas konvergensi model. Eksplorasi kustomisasi parameter ini dirancang untuk memaksimalkan potensi ekstraksi fitur dari tiap karakteristik arsitektur algoritma. Spesifikasi detail mengenai perbandingan nilai parameter bawan dan nilai kustomisasi hasil optimasi (*tuned*) beserta target optimasi klinisnya disajikan secara komprehensif pada Tabel 1.

Kedua pendekatan melalui tahap preprocessing yang sama, meliputi penggabungan seluruh 21 variabel prediktor ke dalam satu vektor fitur tunggal dan normalisasi skala menggunakan metode Min-Max, Selanjutnya, Evaluasi performa model dilakukan dengan menerapkan skema pembagian data train dan test berasio 80:20 yang dikunci menggunakan *random seed* bernilai 42. Pendekatan *hold-out* tunggal ini dipilih karena ukuran sampel dataset yang digunakan sangat besar (N = 70.692 responden) dengan distribusi kelas yang sudah seimbang (*balanced split* 50:50). Karakteristik data berskala besar ini menjamin keterwakilan statistik (*statistical representation*) yang stabil pada data uji tunggal, sehingga mampu menghasilkan metrik performa yang objektif tanpa memerlukan beban komputasi tambahan dari skema *cross validation*. Selanjutnya evaluasi dilakukan menggunakan empat algoritma utama *SDCA Logistic Regression, LBFGS Logistic Regression, LightGBM*, dan *FastTree* dengan metrik akurasi, AUC, *F1-score*, presisi, dan *recall* untuk menilai performa masing-masing model.

2.1. Dataset

Dataset yang digunakan adalah *Diabetes Health Indicators* yang berasal dari survei BRFSS *(Behavioral Risk Factor Surveillance System)* tahun 2015. Penelitian ini secara spesifik menggunakan varian *balanced split* yang terdiri dari 70.692 baris data. Variabel tersebut mencakup faktor demografi (usia, jenis kelamin), gaya hidup (aktivitas fisik, konsumsi buah, konsumsi sayur, merokok, konsumsi alkohol), serta kondisi kesehatan (*Body Mass Index*

(BMI), tekanan darah, kolesterol, status kesehatan umum, kesehatan mental). Variabel target (*Label*) menunjukkan status diabetes responden (0 = tidak diabetes, 1 = diabetes). Daftar lengkap variabel ditunjukkan pada Tabel 2.

2.2. Preprocessing Data

Tahap *preprocessing* dilakukan dengan menggabungkan seluruh variabel prediktor ke dalam satu vektor fitur bernama *features* agar algoritma pembelajaran mesin dapat memproses data secara serentak dalam bentuk representasi numerik. Selanjutnya, nilai setiap fitur dinormalisasi ke dalam rentang [0,1] menggunakan metode *Min-Max Normalization* untuk mencegah dominasi variabel berskala besar dan meningkatkan stabilitas konvergensi algoritma, khususnya *Logistic Regression*. Dataset kemudian dibagi menjadi *train set* (80%) dan *test set* (20%), pembagian dengan rasio 8:2 telah dilakukan pada penelitian sebelumnya[13]. Proses pembagian dataset dilakukan secara acak dengan menggunakan *random seed* bernilai 42. Penetapan *seed* ini bertujuan untuk memastikan bahwa hasil pembagian data ke dalam *train set* dan *test set* selalu konsisten pada setiap eksekusi, sehingga penelitian dapat direplikasi dengan hasil yang sama. Pada pendekatan baseline, seluruh algoritma dilatih menggunakan konfigurasi parameter standar bawaan dari *framework* ML.NET untuk mengetahui performa dasar model. Sebaliknya, pada pendekatan kedua (*hyperparameter* tuning), dilakukan kustomisasi parameter kunci secara terarah dan eksploratif guna mengoptimalkan fungsi loss dan mengendalikan kompleksitas model. Pada algoritma linear (*SDCA* dan *LBFGS*), optimasi difokuskan pada perluasan batas iterasi dan pengetatan toleransi optimasi (*optimizationTolerance* hingga 10^{-10}) guna memastikan tercapainya konvergensi yang matang. Sementara itu, pada algoritma berbasis pohon (*LightGBM* dan *FastTree*), strategi dialihkan pada penambahan jumlah pohon keputusan (*numberOfIterations* = 2000, *numberOfTrees* = 1000) yang diimbangi dengan pembatasan ketat jumlah daun (*numberOfLeaves*) dan penurunan laju pembelajaran (*learningrate*). Pendekatan tuning ini diterapkan secara konservatif guna mencegah penyesuaian berlebih (*overfitting*), meningkatkan kestabilan nilai metrik Area AUC, serta menyeimbangkan rasio presisi dan *recall* (*F1-score*) pada prediksi risiko diabetes.

2.3. Tahap Pelatihan

Setelah *preprocessing*, data digunakan dalam dua pendekatan pelatihan. Pada pendekatan baseline, data hasil *preprocessing* langsung dimasukkan ke *pipeline* standar dengan parameter bawaan ML.NET sebagai acuan performa awal. Pendekatan model baseline digunakan untuk melihat performa model pada distribusi asli data, sebagaimana dilakukan dalam penelitian[6]. Pada pendekatan kedua (*hyperparameter tuning*), model dilatih dengan melakukan kustomisasi parameter kunci secara eksploratif guna mengoptimalkan fungsi *loss* masing-masing algoritma. Langkah optimasi ini diadopsi mengikuti rekomendasi metodologi riset kesehatan yang menegaskan bahwa konfigurasi *hyperparameter* terarah secara konsisten mampu menambah performa model klasifikasi prediktif dibandingkan hanya mengandalkan parameter bawaan [14].

Kedua pendekatan dilatih menggunakan empat algoritma utama yang telah banyak digunakan dalam literatur *machine learning*. Pertama, *SDCA* (*Stochastic Dual Coordinate Ascen*t) *Logistic Regression* merupakan algoritma optimisasi yang efisien untuk regresi logistik pada dataset besar, dengan jaminan konvergensi yang kuat [15]. *SDCA Logistic Regression* bekerja dengan memperbarui koordinat pada bentuk dual fungsi objektif secara acak, sehingga mampu menangani data berukuran besar dengan kompleksitas linear. Kedua, LBFGS *Logistic Regression* menggunakan pendekatan quasi-Newton yang lebih stabil dibanding metode *stochastic*, dengan memanfaatkan informasi gradien untuk mempercepat konvergensi [16]. *LBFGS* (*Limited memory Broyden Fletcher Goldfarb Shanno*) sering digunakan sebagai baseline yang lebih seimbang karena menghasilkan *trade-off* (keseimbangan) yang baik antara presisi dan *recall*.

Ketiga, *LightGBM* adalah algoritma *Gradient Boosting Framework* yang dirancang untuk efisiensi dan akurasi tinggi. *LightGBM* menggunakan teknik *Leaf-wise Growth* dan *Histogram-based Decision Tree Learning* sehingga mampu menangani dataset besar dengan cepat. Penelitian terbaru menunjukkan *LightGBM* unggul dalam prediksi diabetes dengan akurasi dan AUC yang tinggi[17]. Keempat, *FastTree* merupakan implementasi *Gradient Boosted Decision Tree* dalam ML.NET yang mendukung klasifikasi biner, regresi, dan *ranking*. Algoritma ini dikembangkan oleh *Microsoft* untuk memberikan keseimbangan antara kecepatan pelatihan dan akurasi prediksi, sehingga cocok digunakan dalam aplikasi praktis berbasis .NET.

Dengan kombinasi baseline dan *hyperparameter tuning* serta pemilihan algoritma yang beragam, penelitian ini tidak hanya membandingkan performa model, tetapi juga memberikan gambaran komprehensif mengenai efektivitas strategi optimasi parameter internal.

2.4. Evaluasi Model

Model yang dihasilkan dari kedua pendekatan kemudian dievaluasi menggunakan metrik performa utama, yaitu akurasi, AUC, *F1-score*, presisi, dan *recall*. Akurasi memberi gambaran umum seberapa banyak prediksi yang benar dibandingkan seluruh data, sehingga mudah dipahami sebagai “tingkat ketepatan

keseluruhan". menilai kemampuan model dengan membedakan pasien diabetes dan non-diabetes, semakin mendekati 100%, semakin baik model dalam memisahkan dua kelompok tersebut. *F1-score* adalah nilai gabungan dari presisi dan *recall*, sehingga bisa dianggap sebagai ukuran keseimbangan, dengan menilai apakah model tidak hanya tepat, tetapi juga konsisten dalam mendeteksi kasus positif. Presisi sendiri menunjukkan seberapa banyak prediksi positif yang benar-benar pasien diabetes, penting untuk menghindari *false positive* yang bisa menimbulkan alarm palsu. Sebaliknya, *recall* menunjukkan seberapa banyak pasien diabetes yang berhasil terdeteksi dari seluruh kasus sebenarnya, sehingga penting untuk memastikan tidak ada pasien berisiko yang terlewat. Dengan melihat kelima metrik ini secara bersamaan, kita bisa menilai perbedaan performa antara model baseline dan model *hyperparameter tuning*, sekaligus menentukan algoritma yang paling konsisten dalam mendeteksi kasus diabetes pada data uji berskala besar. Pendekatan ini sejalan dengan penelitian lain yang juga menggunakan metrik akurasi, AUC, *F1-score*, presisi dan *recall* sebagai dasar evaluasi kinerja model prediksi diabetes [19].

2.5. Pengembangan Aplikasi .NET

Model terbaik dari hasil evaluasi kemudian diintegrasikan ke dalam aplikasi berbasis .NET sebagai bukti konsep implementasi. Aplikasi dirancang menggunakan arsitektur *Model View Controller* (MVC) dengan antarmuka berbasis *Razor Pages*. Dengan integrasi ini, aplikasi dapat digunakan untuk melakukan prediksi risiko diabetes secara langsung, sehingga hasil penelitian tidak hanya bersifat teoretis tetapi juga praktis. Sebagian besar rumah sakit modern mengandalkan sistem operasi *Microsoft Windows* sebagai *platform* utama untuk menjalankan aplikasi klinis dan manajemen data, termasuk sistem rekam medis elektronik seperti *Epic, Cerner,* dan *Meditech*. Dominasi ekosistem *Microsoft* di sektor kesehatan ini menjadikan penelitian berbasis .NET sangat relevan, karena aplikasi yang dikembangkan dengan *framework* tersebut dapat langsung diintegrasikan ke dalam infrastruktur rumah sakit tanpa perlu migrasi platform. Dengan demikian, pengembangan sistem prediksi risiko penyakit menggunakan ML.NET dan integrasi ke aplikasi berbasis *Razor Pages* tidak hanya bersifat akademis, tetapi juga memiliki kontribusi praktis dalam mendukung transformasi digital rumah sakit.

2.6. Implementasi Model

Tahap akhir penelitian menghasilkan analisis komparatif performa algoritma ML.NET serta prototipe aplikasi prediksi diabetes. Analisis ini menunjukkan algoritma dengan performa terbaik dalam klasifikasi risiko diabetes, sementara aplikasi yang dikembangkan menjadi bukti konsep penerapan model dalam sistem informasi kesehatan. Implementasi ini diharapkan dapat menjadi dasar pengembangan sistem *e-health* di masa mendatang yang mendukung deteksi dini dan pencegahan komplikasi diabetes

3. Hasil dan Pembahasan

3.1. Model Baseline

Pada tahap baseline, model dilatih menggunakan varian dataset *balanced split* (distribusi kelas seimbang 50:50) tanpa adanya modifikasi internal pada struktur algoritma, di mana seluruh parameter dijalankan menggunakan nilai standar bawaan *framework* ML.NET. Hasil perbandingan evaluasi kinerja dari keempat algoritma pada kondisi dasar ini dapat dilihat pada Tabel 3. Skenario *baseline* ini sangat penting diterapkan sebagai titik acuan awal (*benchmark*) guna menilai sejauh mana efektivitas peningkatan performa yang dapat dicapai setelah langkah eksplorasi *hyperparameter* diimplementasikan pada tahap berikutnya. Karakteristik performa awal ini terlihat jelas pada algoritma *SDCA Logistic Regression*, yang mencapai *recall* sangat tinggi 97,87%, namun mengorbankan tingkat ketepatan prediksi dengan nilai presisi yang rendah sebesar 58,00%. Kondisi ini berisiko menghasilkan banyak *false positive*, yaitu kasus non-diabetes yang salah diprediksi sebagai diabetes. Akurasinya pun relatif rendah 63,90%, sehingga kurang ideal untuk aplikasi klinis.

Tabel 3. Perbandingan Performa Algoritma (Model *Baseline*)

Algoritma	Akurasi (%)	AUC (%)	*F1-score* (%)	Presisi (%)	*Recall* (%)
SDCA	63,90	81,23	72,83	58,00	97,87
LBFGS	74,65	82,17	74,99	73,21	76,86
LightGBM	75,13	82,78	75,95	72,77	79,42
FastTree	75,24	82,75	76,06	72,87	79,53

Sebaliknya, *LBFGS Logistic Regression* menunjukkan performa lebih stabil dengan akurasi 74,65% dan AUC 82,17%. Presisi 73,21% dan *recall* 76,86% relatif seimbang, menghasilkan *F1-score* 74,99%. Hal ini menjadikan LBFGS lebih dapat diandalkan sebagai *baseline* pembanding dibanding *SDCA Logistic Regression*, karena *trade-off* (keseimbangan) antara presisi dan *recall* lebih proporsional.

Sementara itu, algoritma berbasis *tree boosting* seperti *LightGBM* dan *FastTree* memberikan hasil paling seimbang. *LightGBM* mencapai akurasi 75,13%, AUC 82,78% dan *F1-score* 75,95%, dengan presisi 72,77% serta *recall* 79,42%. *FastTree* menghasilkan performa hampir identik dengan akurasi 75,24%, AUC 82,75%, dan *F1-score* tertinggi 76,06%. Kedua algoritma ini terbukti unggul dalam menangani kompleksitas interaksi fitur, karena mekanisme *Boosting* dan *Decision Tree* mampu mempelajari pola *non-linear* dengan lebih baik dibanding regresi logistik.

Secara keseluruhan, hasil *baseline* menunjukkan bahwa *LightGBM* dan *FastTree* adalah kandidat terbaik dengan performa tertinggi, sedangkan *LBFGS Logistic Regression* dapat dijadikan *baseline* pembanding yang stabil. *SDCA Logistic Regression*, meskipun memiliki *recall* sangat tinggi, tidak layak digunakan secara langsung karena presisi rendah yang berisiko menimbulkan banyak *false positive*. Oleh karena itu, seluruh capaian model *baseline* pada parameter bawaan ini berfungsi sebagai titik yang sah untuk menilai efektivitas dari metode *hyperparameter tuning* yang diimplementasikan pada tahap berikutnya.

3.2. Model *Hyperparameter Tuning*

Dengan penerapan *Hyperparameter Tuning*, semua algoritma menunjukkan peningkatan konsistensi dan keseimbangan antara presisi dan *recall* dibanding *baseline* seperti yang disajikan pada Tabel 4. *SDCA Logistic Regression* mengalami perbaikan signifikan Melalui pengetatan parameter iterasi maksimum hingga 3000 kali serta optimasi regularisasi, nilai presisi *SDCA* berhasil mendongkrak naik dari yang semula hanya 58,00% menjadi 65,68%, diikuti dengan nilai akurasi yang meningkat ke angka 71,95%. Meskipun metrik *recall* mengalami penyesuaian menjadi 90,62% (turun dari tingkat agresivitas awal 97,87%), pergeseran ini menghasilkan nilai *F1-score* yang jauh lebih tinggi dan seimbang sebesar 76,16%. Hasil ini membuktikan bahwa konfigurasi parameter yang tepat mampu mengurangi risiko *false positive*.

Tabel 4. Perbandingan Performa Algoritma (Model *Hyperparameter Tuning*)

Algoritma	Akurasi (%)	AUC (%)	*F1-score* (%)	Presisi (%)	*Recall* (%)
SDCA	71,95	81,26	76,16	65,68	90,62
LBFGS	74,64	82,17	74,98	73,22	76,82
LightGBM	75,37	82,86	76,22	72,92	79,83
FastTree	75,23	82,83	76,04	72,87	79,49

LBFGS *Logistic Regression* mempertahankan stabilitas dengan akurasi 74,64% dan AUC 82,17%. Presisi sebesar 73,22% dan *recall* sebesar 76,82% tetap seimbang, serta menghasilkan F1-score sebesar 74,98%. Konsistensi ini menunjukkan bahwa LBFGS tetap menjadi *baseline* pembanding yang solid, dengan performa yang tidak banyak berubah setelah penerapan *hyperparameter tuning*.

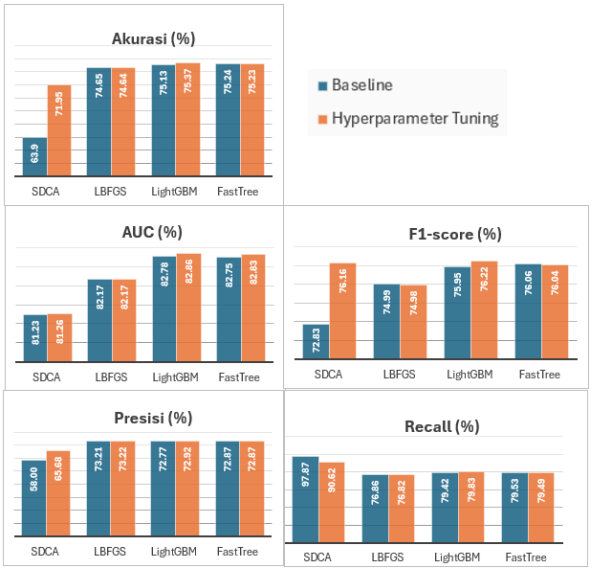
Sementara itu, algoritma berbasis *Boosting* dan *Decision Tree* kembali menempati posisi teratas. *LightGBM* mencapai akurasi 75,37%, AUC 82,86%, dan *F1-score* 76,22%, dengan presisi 72,92% serta *recall* 79,83%. *FastTree* memberikan hasil hampir identik dengan akurasi 75,23%, AUC 82,83%, dan *F1-score* 76,04%. Kedua algoritma ini menunjukkan konsistensi performa terbaik, mampu menjaga keseimbangan metrik sekaligus mempertahankan AUC di atas 82,8%.

Penerapan skema *hyperparameter tuning* terbukti efektif dalam meminimalkan bias agresivitas model linear dan meningkatkan stabilitas generalisasi model di setiap tingkatan algoritma. Hal ini terlihat nyata dari lonjakan nilai presisi pada algoritma SDCA tanpa harus mengorbankan metrik *recall* secara ekstrem, sehingga menghasilkan titik temu *trade-off* (F1-*score*) yang jauh lebih proporsional.

Sementara itu, *LightGBM* dan *FastTree* tetap mempertahankan dominasi di posisi teratas karena arsitektur sekuensial berbasis pohon mampu memetakan kompleksitas interaksi fitur diskrit pada data kesehatan berskala besar dengan sangat baik. Temuan ini sejalan dengan literatur global yang menekankan efektivitas tinggi dari varian algoritma *boosting* dalam domain prediksi medis dan penataan stratifikasi risiko kesehatan [19].

3.3. Pembahasan Hasil Perbandingan

Hasil perbandingan direpresentasikan pada Gambar 2 yang memperlihatkan hasil evaluasi lima metrik utama, yaitu akurasi, AUC, presisi, *recall*, dan *F1-score* pada empat algoritma yang diuji. Warna biru menunjukkan performa model *baseline*, sedangkan warna oranye merepresentasikan model dengan penerapan *hyperparameter tuning*.



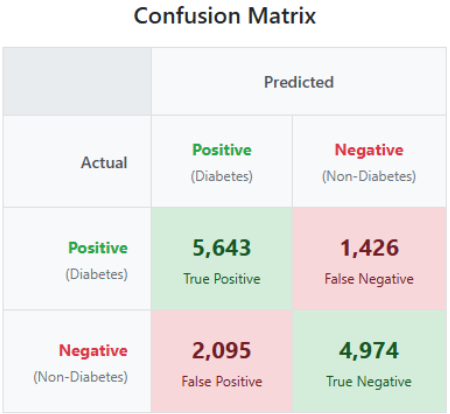
Gambar 2. Perbandingan Hasil Evaluasi Model *Baseline* dan Model *Hyperparameter Tuning* Pada 4 Algoritma

Dengan memperhatikan Gambar 2 tersebut, secara umum hasil perbandingan antara *baseline* dan model *hyperparameter tuning* menunjukkan adanya perbedaan signifikan dalam keseimbangan metrik performa antar algoritma. Pada *baseline*, algoritma *SDCA Logistic Regression* menampilkan *recall* sangat tinggi sebesar 97,87% namun presisi rendah sebesar 58,00%, sehingga menghasilkan banyak *false positive*. Setelah penerapan *hyperparameter tuning*, presisi meningkat menjadi 65,68% dengan *recall* tetap tinggi

sebesar 90,62%, sementara *F1-score* menghasilkan 76,16%, yang artinya *hyperparameter tuning* mampu mengurangi agresivitas fungsi prediksi, meskipun peningkatan presisi masih terbatas.

LBFGS Logistic Regression menunjukkan konsistensi performa baik pada *baseline* maupun *hyperparameter tuning*, dengan akurasi stabil di 74,64% dan AUC 82,17%. Presisi (73,21–73,22%) dan *recall* (76,82–76,86%) tetap seimbang, menghasilkan *F1-score* sekitar 75%. Stabilitas ini menjadikan *LBFGS* sebagai *baseline* pembanding yang solid, meskipun tidak mengalami peningkatan berarti setelah *hyperparameter tuning*.

Algoritma berbasis *Boosting* dan *Decision Tree* yaitu *LightGBM* dan *FastTree* kembali menempati posisi teratas. Pada *baseline*, *LightGBM* mencapai akurasi 75,13% dan *F1-score* 75,95%, sedangkan *FastTree* menghasilkan akurasi 75,24% dan *F1-score* 76,06%. Setelah *hyperparameter tuning*, *LightGBM* sedikit meningkat menjadi akurasi 75,37% dan *F1-score* 76,22%, sementara *FastTree* mencapai akurasi 75,23% dan *F1-score* 76,04%. Konsistensi ini menunjukkan bahwa kedua algoritma mampu menjaga keseimbangan metrik sekaligus mempertahankan AUC di atas 82,8%, sehingga lebih unggul dibanding *Logistic Regression*. Secara keseluruhan, penerapan *hyperparameter tuning* terbukti efektif dalam meningkatkan presisi pada algoritma yang sebelumnya bias secara linear, khususnya *SDCA Logistic Regression*. Namun, *LightGBM* tetap menjadi pilihan terbaik karena mampu menangani kompleksitas data dan interaksi fitur *non-linear* dengan lebih baik. Temuan ini sejalan dengan literatur yang menekankan efektivitas algoritma boosting dalam domain kesehatan, terutama untuk prediksi penyakit dengan dimensi fitur klinis yang kompleks [20].



Gambar 3. Confusion Matrix Model *LightGBM* dengan *Hyperparameter Tuning*

Untuk membuktikan signifikansi dari peningkatan akurasi *LightGBM* yang secara nominal terlihat tipis (dari 75,13% menjadi 75,37%), analisis dilakukan dengan mengonversi nilai fraksional 0,24% tersebut ke dalam jumlah absolut data uji (N = 14.138 responden). Hasil kalkulasi menunjukkan adanya peningkatan dari 10.622 prediksi benar pada model *baseline* menjadi 10.656 prediksi benar pada model *hyperparameter tuning*. Peningkatan absolut sebanyak 34 responden ini membuktikan bahwa pada dataset medis berskala besar, variasi parameter sekecil apa pun tidak terjadi karena faktor kebetulan (*chance*), melainkan karena model berhasil memperjelas batas keputusan (*decision boundary*).

Dalam hal ini, signifikansi dalam domain kesehatan tidak hanya diukur dari akurasi global, melainkan dari restrukturisasi internal metrik sensitivitasnya lewat analisis *Confusion Matrix* yang disajikan pada Gambar 3.

Berdasarkan Gambar 3, model *LightGBM* dengan *hyperparameter tuning* berhasil mengidentifikasi 5.643 sampel penderita diabetes secara akurat (*True Positive*) dan menyaring 4.974 sampel responden sehat secara tepat (*True Negative*). Sementara itu, margin galat model mencatat 2.095 sampel individu sehat yang salah diprediksi sebagai penderita diabetes (*False Positive*) dan 1.426 sampel kasus diabetes aktual yang luput dari deteksi (*False Negative*). Selain itu dampak nyata terlihat pada algoritma *SDCA*, di mana *hyperparameter tuning* berhasil memaksa fungsi prediksi bergerak menjadi lebih selektif sehingga mereduksi angka *False Positive* (kasus non-diabetes yang salah didiagnosis). Dalam konteks pelayanan klinis, keberhasilan *LightGBM* dalam menyeimbangkan nilai *precision* (72,92%) dan *recall* (79,83%) sangat krusial. Penurunan *false positive* akan meminimalkan risiko misklasifikasi pada populasi sehat, sehingga mengurangi kecemasan pasien serta menekan biaya pemeriksaan laboratorium lanjutan yang tidak perlu. Di sisi lain, kemampuan mempertahankan *Recall* yang tinggi memastikan bahwa penderita diabetes sejati tidak luput sebagai *False Negative*, sehingga intervensi medis dan pencegahan komplikasi kronis (seperti gagal ginjal atau retinopati) dapat dilakukan sejak dini. Hal ini menegaskan bahwa model *LightGBM* teroptimasi memiliki potensi yang memadai untuk diintegrasikan secara *native* sebagai instrumen skrining awal (*early screening tool*) digital pada ekosistem informasi rumah sakit.

3.4. Ekstraksi Feature Algoritma *LightGBM*

Dari hasil perbandingan antara *baseline* dan *hyperparameter tuning*, algoritma *LightGBM* dipilih sebagai model terbaik karena menunjukkan performa paling konsisten. Pada *baseline*, *LightGBM* mencapai akurasi 75,13%, AUC 82,78%, dan *F1-score* 75,95%, sedangkan pada model *hyperparameter tuning* hasilnya sedikit meningkat dengan akurasi 75,37%, AUC 82,86%, dan *F1-score* 76,22%. Konsistensi ini menegaskan bahwa *LightGBM* mampu menjaga keseimbangan antara presisi dan *recall* pada kedua pendekatan, sekaligus mempertahankan kualitas prediksi. Pemilihan model dari metode

hyperparameter tuning dianggap lebih representatif untuk aplikasi produksi karena mampu mengurangi agresivitas fungsi prediksi tanpa menurunkan performa keseluruhan. Oleh karena itu, analisis fitur dilakukan menggunakan *LightGBM* untuk mengidentifikasi variabel yang paling berpengaruh terhadap klasifikasi risiko diabetes.

Gambar 4. Hasil Ekstraksi Feature Algoritma *LightGBM*

Hasil *feature importance* pada algoritma *LightGBM* direpresentasikan pada Gambar 4, yang menunjukkan bahwa variabel *GenHlth, BMI, Age, HighBP,* dan *HighChol* memiliki kontribusi terbesar terhadap prediksi. Temuan ini sejalan dengan literatur medis yang menekankan pentingnya faktor metabolik dan kondisi kesehatan umum dalam menentukan risiko diabetes [21]. Dengan demikian, ekstraksi fitur dari algoritma terbaik tidak hanya memperkuat validitas hasil penelitian, tetapi juga memberikan wawasan praktis mengenai faktor-faktor utama yang perlu diperhatikan dalam sistem prediksi berbasis *e-health*.
Hasil analisis yang direpresentasikan pada Gambar 4, menegaskan bahwa *GenHlth* memiliki korelasi tertinggi (+0.409) dan dikategorikan *Strong*, menunjukkan bahwa semakin buruk kondisi kesehatan umum seseorang, semakin tinggi risiko diabetes. Variabel *HighBP* (+0.373) juga termasuk kategori *Strong*, menegaskan pengaruh tekanan darah tinggi terhadap peningkatan risiko. Sementara itu, *BMI*, *HighChol*, dan *Age* berada pada kategori *Moderate*, masing-masing menunjukkan hubungan positif dengan kemungkinan terjadinya diabetes. Warna batang hijau pada visualisasi menggambarkan arah korelasi positif, sedangkan label "*Strong*" dan "*Moderate*" dibedakan dengan warna oranye dan kuning untuk memperjelas tingkat pengaruh. Secara keseluruhan, kelima fitur utama yang diidentifikasi oleh *LightGBM* menunjukkan dominasi faktor metabolik dan kesehatan umum terhadap risiko diabetes. Korelasi positif pada semua variabel menegaskan bahwa peningkatan nilai pada indikator tersebut berkaitan dengan meningkatnya kemungkinan diabetes, sejalan dengan temuan medis sebelumnya [17].

Analisis *Correlation Matrix Heatmap* (korelasi antar fitur) yang ditunjukkan pada Gambar 5 memberikan gambaran mengenai hubungan antar variabel prediktor dalam dataset. Beberapa fitur menunjukkan korelasi positif yang cukup kuat, seperti antara *GenHlth* dan *PhysHlth* (r = 0.55) serta *GenHlth* dan *DiffWalk* (r = 0.47), yang menegaskan bahwa kondisi kesehatan umum sangat berkaitan dengan kesehatan fisik dan kemampuan berjalan. Korelasi positif juga terlihat antara *HighBP* dan *Age* (r = 0.34), mengindikasikan bahwa tekanan darah tinggi lebih sering muncul pada kelompok usia lanjut. Sebaliknya, terdapat korelasi negatif antara *PhysActivity* dan BMI (r = -0.16) serta *Income* dan *GenHlth* (r = -0.38), yang menunjukkan bahwa aktivitas fisik berhubungan dengan penurunan indeks massa tubuh, sementara pendapatan lebih tinggi berkorelasi dengan kondisi kesehatan umum yang lebih baik.

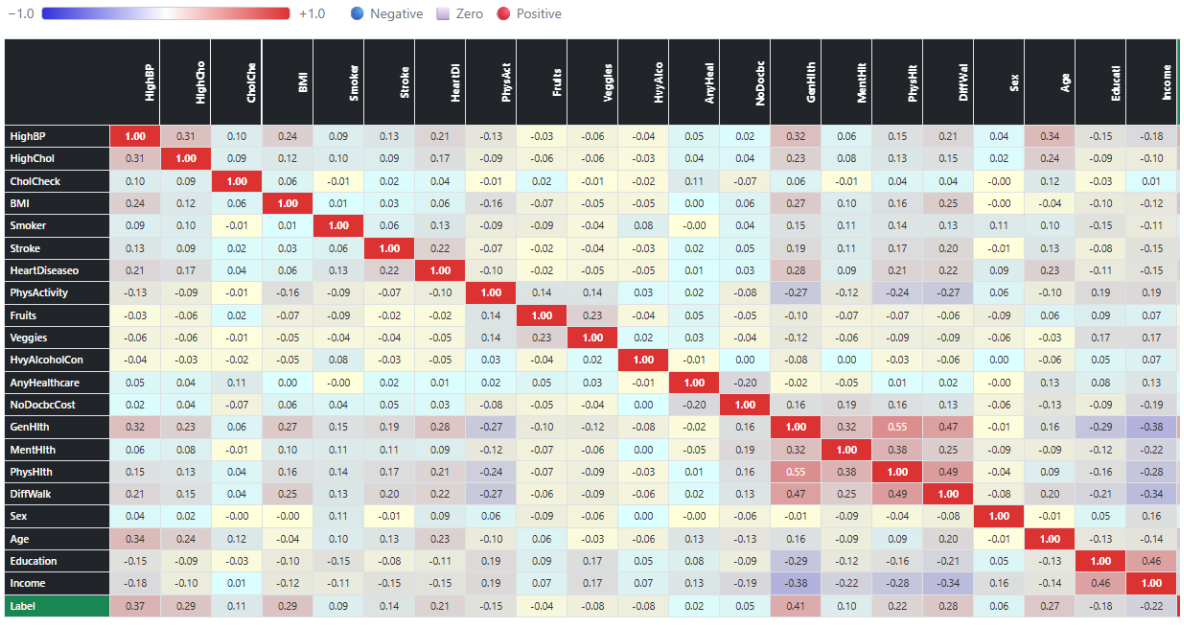
Secara keseluruhan, *heatmap* ini menegaskan adanya keterkaitan antar faktor metabolik, gaya hidup, dan kondisi sosial-ekonomi dalam memengaruhi risiko diabetes. Identifikasi korelasi antar fitur juga penting untuk mendeteksi potensi multikolinearitas, meskipun risikonya relatif rendah karena algoritma *boosting* seperti *LightGBM* mampu mengatasi redundansi variabel. Dengan demikian, analisis korelasi tidak hanya memperkuat interpretasi hasil *feature importance*, tetapi juga memberikan wawasan tambahan mengenai interaksi antar faktor risiko yang relevan dalam konteks kesehatan masyarakat.

3.5. Implementasi Aplikasi

Prototipe aplikasi yang ditunjukan pada Gambar 6 dikembangkan sebagai bukti konsep penerapan model *machine learning* dalam sistem informasi kesehatan berbasis .NET. Tahap pertama dilakukan dengan melatih model menggunakan aplikasi *Console* ML.NET melalui dua pendekatan, yaitu *baseline* dan *hyperparameter tuning*. Model yang telah dilatih kemudian disimpan dalam bentuk file .zip di folder khusus pada proyek *Razor Pages*, sehingga dapat diakses langsung oleh aplikasi web.

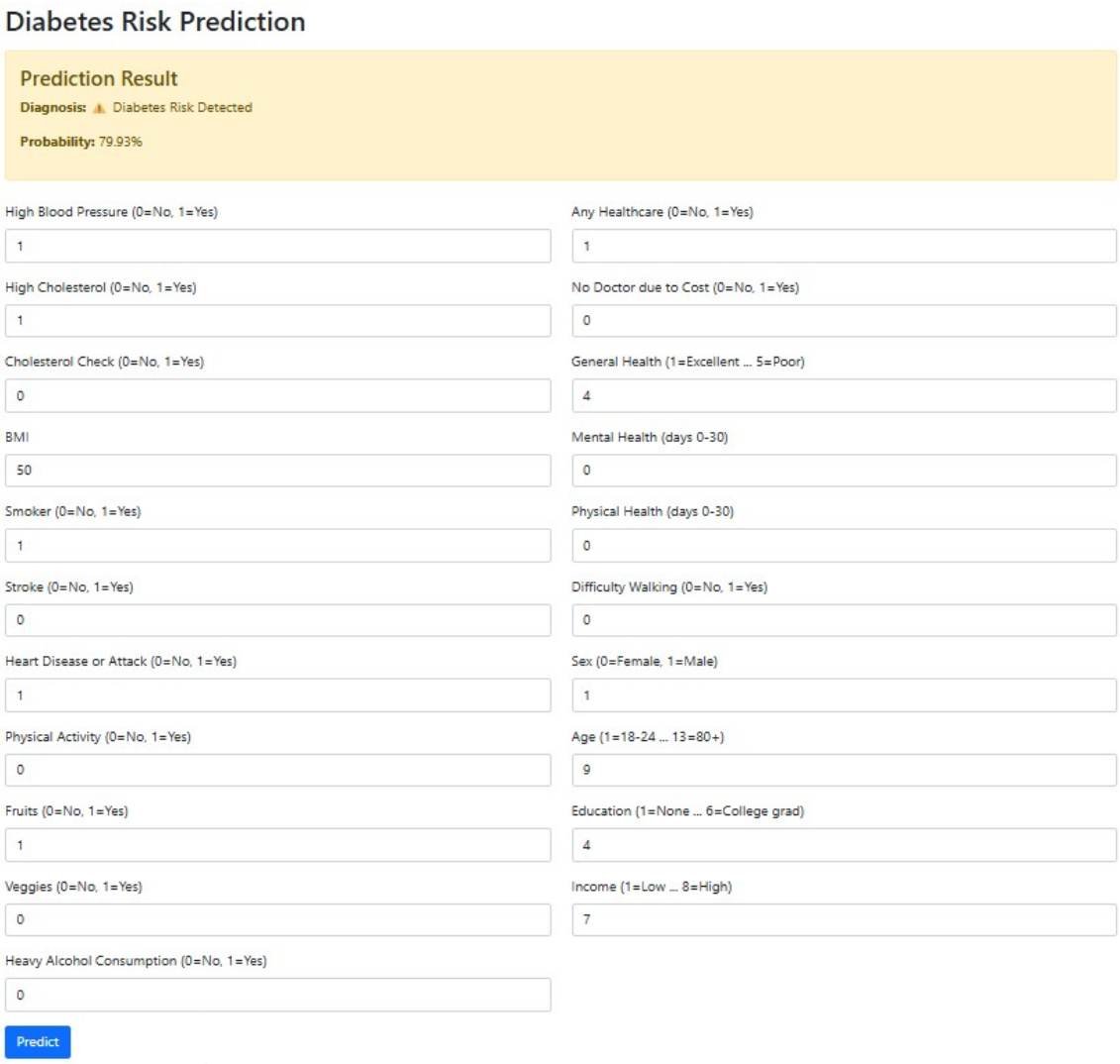
Selanjutnya, dibuat antarmuka prediksi yang memungkinkan pengguna memasukkan 21 variabel yang tertera pada Table 1 di atas. Setelah data diinput, sistem memanggil model *LightGBM* yang telah disimpan untuk melakukan prediksi risiko diabetes secara otomatis. Hasil prediksi ditampilkan dalam bentuk klasifikasi risiko (diabetes atau tidak) beserta probabilitasnya, sehingga pengguna dapat memahami tingkat kemungkinan secara kuantitatif.

Implementasi ini menunjukkan integrasi yang berhasil antara model ML.NET dan aplikasi .NET berbasis *Razor Pages*, sekaligus menjadi bukti konsep bahwa sistem *e-health* dapat dikembangkan untuk



	HighBP	HighChol	CholChe	BMI	Smoker	Stroke	HeartDi	PhysAct	Fruits	Veggies	HvyAlco	AnyHeal	NoDocbc	GenHlth	MentHlt	PhysHlt	DiffWal	Sex	Age	Educati	Income
HighBP	1.00	0.31	0.10	0.24	0.09	0.13	0.21	-0.13	-0.03	-0.06	-0.04	0.05	0.02	0.32	0.06	0.15	0.21	0.04	0.34	-0.15	-0.18
HighChol	0.31	1.00	0.09	0.12	0.10	0.09	0.17	-0.09	-0.06	-0.06	-0.03	0.04	0.04	0.23	0.08	0.13	0.15	0.02	0.24	-0.09	-0.10
CholCheck	0.10	0.09	1.00	0.06	-0.01	0.02	0.04	-0.01	0.02	-0.01	-0.02	0.11	-0.07	0.06	-0.01	0.04	0.04	-0.00	0.12	-0.03	0.01
BMI	0.24	0.12	0.06	1.00	0.01	0.03	0.06	-0.16	-0.07	-0.05	-0.05	0.00	0.06	0.27	0.10	0.16	0.25	-0.00	-0.04	-0.10	-0.12
Smoker	0.09	0.10	-0.01	0.01	1.00	0.06	0.13	-0.09	-0.09	-0.04	0.08	-0.00	0.04	0.15	0.11	0.14	0.13	0.11	0.10	-0.15	-0.11
Stroke	0.13	0.09	0.02	0.03	0.06	1.00	0.22	-0.07	-0.02	-0.04	-0.03	0.02	0.05	0.19	0.11	0.17	0.20	-0.01	0.13	-0.08	-0.15
HeartDiseaseo	0.21	0.17	0.04	0.06	0.13	0.22	1.00	-0.10	-0.02	-0.05	-0.05	0.01	0.03	0.28	0.09	0.21	0.22	0.09	0.23	-0.11	-0.15
PhysActivity	-0.13	-0.09	-0.01	-0.16	-0.09	-0.07	-0.10	1.00	0.14	0.14	0.03	0.02	-0.08	-0.27	-0.12	-0.24	-0.27	0.06	-0.10	0.19	0.19
Fruits	-0.03	-0.06	0.02	-0.07	-0.09	-0.02	-0.02	0.14	1.00	0.23	-0.04	0.05	-0.05	-0.10	-0.07	-0.07	-0.06	-0.09	0.06	0.09	0.07
Veggies	-0.06	-0.06	-0.01	-0.05	-0.04	-0.04	-0.05	0.14	0.23	1.00	0.02	0.03	-0.04	-0.12	-0.06	-0.09	-0.09	-0.06	-0.03	0.17	0.17
HvyAlcoholCon	-0.04	-0.03	-0.02	-0.05	0.08	-0.03	-0.05	0.03	-0.04	0.02	1.00	-0.01	0.00	-0.08	0.00	-0.03	-0.06	0.00	-0.06	0.05	0.07
AnyHealthcare	0.05	0.04	0.11	0.00	-0.00	0.02	0.01	0.02	0.05	0.03	-0.01	1.00	-0.20	-0.02	-0.05	0.01	0.02	-0.00	0.13	0.08	0.13
NoDocbcCost	0.02	0.04	-0.07	0.06	0.04	0.05	0.03	-0.08	-0.05	-0.04	0.00	-0.20	1.00	0.16	0.19	0.16	0.13	-0.06	-0.13	-0.09	-0.19
GenHlth	0.32	0.23	0.06	0.27	0.15	0.19	0.28	-0.27	-0.10	-0.12	-0.08	-0.02	0.16	1.00	0.32	0.55	0.47	-0.01	0.16	-0.29	-0.38
MentHlth	0.06	0.08	-0.01	0.10	0.11	0.11	0.09	-0.12	-0.07	-0.06	0.00	-0.05	0.19	0.32	1.00	0.38	0.25	-0.09	-0.09	-0.12	-0.22
PhysHlth	0.15	0.13	0.04	0.16	0.14	0.17	0.21	-0.24	-0.07	-0.09	-0.03	0.01	0.16	0.55	0.38	1.00	0.49	-0.04	0.09	-0.16	-0.28
DiffWalk	0.21	0.15	0.04	0.25	0.13	0.20	0.22	-0.27	-0.06	-0.09	-0.06	0.02	0.13	0.47	0.25	0.49	1.00	-0.08	0.20	-0.21	-0.34
Sex	0.04	0.02	-0.00	-0.00	0.11	-0.01	0.09	0.06	-0.09	-0.06	0.00	-0.00	-0.06	-0.01	-0.09	-0.04	-0.08	1.00	-0.01	0.05	0.16
Age	0.34	0.24	0.12	-0.04	0.10	0.13	0.23	-0.10	0.06	-0.03	-0.06	0.13	-0.13	0.16	-0.09	0.09	0.20	-0.01	1.00	-0.13	-0.14
Education	-0.15	-0.09	-0.03	-0.10	-0.15	-0.08	-0.11	0.19	0.09	0.17	0.05	0.08	-0.09	-0.29	-0.12	-0.16	-0.21	0.05	-0.13	1.00	0.46
Income	-0.18	-0.10	0.01	-0.12	-0.11	-0.15	-0.15	0.19	0.07	0.17	0.07	0.13	-0.19	-0.38	-0.22	-0.28	-0.34	0.16	-0.14	0.46	1.00
Label	0.37	0.29	0.11	0.29	0.09	0.14	0.21	-0.15	-0.04	-0.08	-0.08	0.02	0.05	0.41	0.10	0.22	0.28	0.06	0.27	-0.18	-0.22

Gambar 5. *Correlation Matrix Heatmap* Antar Fitur Prediktor Diabetes



Gambar 6. Halaman Prototipe Aplikasi Dengan Input 21 Variabel

diabetes. Prototipe aplikasi ini bukan sekadar eksperimen algoritma, tetapi sudah mengintegrasikan model ML.NET ke dalam sistem *e-health* berbasis .NET.

3.6. Komparasi Hasil dengan Penelitian Terdahulu

Tabel 5. Komparasi Performa dengan Penelitian Terdahulu

Peneliti	Metode Algoritma	Akurasi (%)	AUC (%)
Pang [22]	*Gradient Boosting*	74,90	74,70
Nugraha & Febrian [23]	*CatBoost*	74,95	82,67
Nguyen & Zhang [24]	*Logistic Regression*	75,00	-
Specht dkk.[25]	*PrivateBoost*	-	81,80
Ayoade dkk. [26]	*Deep Neural Network*	75,48	82,33
Penelitian Usulan	*LightGBM + Tuning*	75,37	82,86

Untuk memvalidasi keandalan model yang dikembangkan, hasil pengujian dikomparasikan dengan beberapa penelitian terdahulu yang menggunakan dataset sejenis. sebagaimana dirangkum pada Tabel 5. Berdasarkan Tabel 5, model *LightGBM* dengan *hyperparameter tuning* yang dibangun pada *framework* ML.NET menghasilkan performa yang sangat kompetitif, dengan tingkat akurasi 75,37% dan nilai AUC mencapai 82,86%. Pada metrik akurasi, model ini berhasil mengungguli arsitektur *Gradient Boosting* milik Pang [22] di angka 74,90% dan *CatBoost* milik Nugraha & Febrian [23] yang mencatat 74,95%, serta *Logistic Regression* dari Nguyen & Zhang [24] sebesar 75,00%. Meskipun model *Deep Neural Network* milik Ayoade dkk. [26] mencatatkan akurasi sedikit lebih tinggi yaitu 75,48%, model *LightGBM* yang diusulkan mampu mengimbangi performa tersebut dengan selisih marjinal sebesar 0,11%. Keunggulan utama dari model yang diusulkan terlihat pada kemampuan diskriminasi klinis (skor AUC), di mana arsitektur *LightGBM* berbasis ML.NET ini sukses melampaui seluruh model pembanding dengan skor tertinggi mencapai 82,86%, termasuk melampaui performa model desentralisasi *PrivateBoost* milik Specht dkk.[25] yang mencatatkan nilai AUC sebesar 81,80%. Capaian ini secara empiris membuktikan bahwa melalui konfigurasi *hyperparameter* yang tepat, *framework* ML.NET memiliki kapabilitas matematis yang setara dan sepenuhnya valid dibandingkan ekosistem Python untuk menangani klasifikasi prediktif data kesehatan skala besar.

4. Kesimpulan

Penelitian ini menyimpulkan bahwa penerapan *hyperparameter tuning* terbukti mampu meningkatkan keseimbangan hasil evaluasi algoritma, khususnya pada *SDCA Logistic Regression* yang mengalami kenaikan presisi tanpa kehilangan *recall* yang signifikan. *LBFGS Logistic Regression* tetap stabil sebagai pembanding, sementara *LightGBM* dan *FastTree* menunjukkan performa paling konsisten dengan akurasi sekitar 75%, AUC di atas 82,86%, dan *F1-score* di atas 76%, sehingga *LightGBM* dipilih sebagai model terbaik. Berdasarkan uji komparasi dengan penelitian terdahulu, model LightGBM terbukti sangat kompetitif. Akurasi model usulan sebesar 75,37% berhasil melampaui performa algoritma berbasis Python. Terlebih lagi, model usulan berhasil memuncaki seluruh pembanding pada metrik AUC, yang sekaligus membuktikan kesetaraan kapabilitas matematis *framework* ML.NET terhadap ekosistem Python dalam menangani data kesehatan skala besar.

Analisis fitur dari *LightGBM* menegaskan bahwa variabel *GenHlth, BMI, Age, HighBP*, dan *HighChol* merupakan faktor dominan dalam prediksi risiko diabetes, sejalan dengan literatur medis yang menekankan pentingnya faktor metabolik dan kesehatan umum. Selain itu, analisis korelasi antar fitur memperlihatkan keterkaitan antara faktor metabolik, gaya hidup, dan kondisi sosial-ekonomi, yang memperkuat interpretasi hasil serta memberikan wawasan tambahan mengenai interaksi antar variabel risiko. Implementasi prototipe aplikasi berbasis .NET dengan integrasi model *LightGBM* menunjukkan bukti konsep yang berhasil, di mana sistem *e-health* dapat mendukung deteksi dini dan pencegahan komplikasi diabetes melalui antarmuka prediksi dengan 21 variabel kesehatan dan demografi, sekaligus menjadi dasar pengembangan lebih lanjut menuju integrasi dengan data *real-time* dan rekam medis elektronik.

Reference

[1] F. Nie, X. Song, and W. Chen, “DRPM: An advanced predictive model for early diabetes detection and risk stratification,” Mol. Ther. Nucleic Acids, vol. 36, no. 3, Sep. 2025, doi: 10.1016/j.omtn.2025.102576.

[2] A. Mahabub, “A robust voting approach for diabetes prediction using traditional machine learning techniques,” SN Appl. Sci., vol. 1, no. 12, Dec. 2019, doi: 10.1007/s42452-019-1759-7.

[3] I. Tasin, T. U. Nabil, S. Islam, and R. Khan, “Diabetes prediction using machine learning and explainable AI techniques,” Healthc. Technol. Lett., vol. 10, no. 1–2, pp. 1–10, Feb. 2023, doi: 10.1049/htl2.12039.

[4] S. Jangili, H. Vavilala, G. S. B. Boddeda, S. M. Upadhyayula, R. Adela, and S. R. Mutheneni, “Machine learning-driven early biomarker prediction for type 2 diabetes mellitus associated coronary artery diseases,” Clin. Epidemiol. Glob. Health, vol. 24, Nov. 2023, doi: 10.1016/j.cegh.2023.101433.

[5] E. Safitri, D. Rofianto, N. Purwati, H. Kurniawan, and S. Karnila, “Prediksi Penyakit Diabetes Melitus Menggunakan Algoritma Machine Learning,” vol. 12, no. 4, 2024, doi: 10.26418/justin.v12i4.84620.

[6] U. M. Butt, S. Letchmunan, M. Ali, F. H. Hassan, A. Baqir, and H. H. R. Sherazi, “Machine Learning Based Diabetes Classification and Prediction for Healthcare Applications,” J. Healthc. Eng., vol. 2021, 2021, doi: 10.1155/2021/9930985.

[7] M. E. Febrian, F. X. Ferdinan, G. P. Sendani, K. M. Suryanigrum, and R. Yunanda, “Diabetes prediction using supervised machine learning,” in Procedia Computer Science, Elsevier B.V., 2022, pp. 21–30. doi: 10.1016/j.procs.2022.12.107.

[8] E. D. C. Pereira and W. Andriyani, "DIABETES PREDICTION USING MACHINE LEARNING," JIKO (Jurnal Informatika dan Komputer), vol. 9, no. 3, p. 639, Oct. 2025, doi: 10.26798/jiko.v9i3.2104.

[9] I. M. A. Sudestra, A. W. O. Gama, G. H. Prathama, I. G. N. D. Paramartha, and M. Hakimi, "Comparative Performance of Machine Learning Algorithms for Diabetes Prediction," Journal of Technology and Informatics (JoTI), vol. 8, no. 1, pp. 50–61, Apr. 2026, doi: 10.37802/joti.v8i1.1195.

[10] M. Deng, R. Yang, X. Zheng, Y. Deng, and J. Jiang, "Artificial intelligence in diabetes care: from predictive analytics to generative AI and implementation challenges," 2025, Frontiers Media SA. doi: 10.3389/fendo.2025.1620132.

[11] R. D. Joshi and C. K. Dhakal, "Predicting type 2 diabetes using logistic regression and machine learning approaches," Int. J. Environ. Res. Public Health, vol. 18, no. 14, Jul. 2021, doi: 10.3390/ijerph18147346.

[12] Z. Ahmed et al., "Machine learning at microsoft with ML.Net," in Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Association for Computing Machinery, Jul. 2019, pp. 2448–2458. doi: 10.1145/3292500.3330667.

[13] N. Varshney and S. Singh, "Enhancing Diabetes Prediction: A comparative analysis of Train-Test Split and Stratified 10-Fold Cross-Validation with SMOTE Integration," in 2024 7th International Conference on Contemporary Computing and Informatics (IC3I), IEEE, Sep. 2024, pp. 1345–1351. doi: 10.1109/IC3I61595.2024.10829095.

[14] F. Vereijken, J. M. Reps, P. Rijnbeek, and R. D. Williams, "Loss function influence on hyperparameter optimization for observational healthcare prediction models," Journal of the American Medical Informatics Association, May 2026, doi: 10.1093/jamia/ocag075.

[15] T. Zhang, "Stochastic Dual Coordinate Ascent Methods for Regularized Loss Minimization Shai Shalev-Shwartz," Feb. 2013.

[16] R. Bollapragada, D. Mudigere, J. Nocedal, H.-J. M. Shi, P. Tak, and P. Tang, "A Progressive Batching L-BFGS Method for Machine Learning," 2018.

[17] S. Verma and A. Pandey, "Optimized LightGBM Model for Diabetes Prediction," 2025. [Online]. Available: https://www.ijspr.com

[18] A. Astofa, P. Rosyani, and S. Apandi, "BULLETIN OF COMPUTER SCIENCE RESEARCH Evaluasi Komparatif Algoritma Machine Learning untuk Prediksi Dini Diabetes," Media Online), vol. 6, no. 1, pp. 558–565, 2025, doi: 10.47065/bulletincsr.v6i1.859.

[19] A. Al-Nafjan, A. Aljuhani, A. Alshebel, A. Alharbi, and A. Alshehri, "Artificial Intelligence in Predictive Healthcare: A Systematic Review," Oct. 01, 2025, Multidisciplinary Digital Publishing Institute (MDPI). doi: 10.3390/jcm14196752.

[20] I. Mohammed Hassoon, "Boosting Learning Algorithms for Chronic Diseases Prediction: A Review," Iraqi Journal for Computers and Informatics, vol. 50, no. 2, pp. 22–30, Oct. 2024, doi: 10.25195/ijci.v50i2.506.

[21] F. Rahman, S. Hossain, J. J. Tiang, and A. Al Nahid, "Diabetes Prediction Using Feature Selection Algorithms and Boosting-Based Machine Learning Classifiers," Diagnostics, vol. 15, no. 20, Oct. 2025, doi: 10.3390/diagnostics15202622.

[22] K. Pang, "A comparative study of explainable machine learning models with Shapley values for diabetes prediction," Healthcare Analytics, vol. 7, Jun. 2025, doi: 10.1016/j.health.2025.100390.

[23] V. Pasa Nugraha and A. Febrian, "Comprehensive Diabetes Risk Prediction Using BRFSS Data: Performance, Explainability, Fairness, and Calibration," 2026. [Online]. Available: http://jurnal.polibatam.ac.id/index.php/JAIC

[24] B. Nguyen and Y. Zhang, "A Comparative Study of Diabetes Prediction Based on Lifestyle Factors Using Machine Learning," Mar. 2025, [Online]. Available: http://arxiv.org/abs/2503.04137

[25] B. Specht et al., "PrivateBoost: Rethinking Federated Learning for Patient-Owned Medical Data: Learning from Single Records at Scale," Feb. 12, 2026. doi: 10.64898/2026.02.10.26345891.

[26] O. B. Ayoade, S. Shahrestani, and C. Ruan, "Machine Learning and Deep Learning Approaches for Predicting Diabetes Progression: A Comparative Analysis," *Electronics (Switzerland)*, vol. 14, no. 13, Jul. 2025, doi: 10.3390/electronics14132583.