

Optimasi Decision Tree Menggunakan Teknik Boosting pada Prediksi Penyakit Diabetes

Pedi Nopedi¹, Aries Saifudin²

Teknik Informatika, Universitas Pamulang, Jl. Raya Puspittek No.46, Buaran, Serpong, Kota Tangerang Selatan, Banten Indonesia 15310

e-mail: ¹pedinopedi30@gmail.com, ²aries.saifudin@unpam.ac.id

Submitted Date: September 10th, 2023

Reviewed Date: September 19th, 2023

Revised Date: September 23rd, 2023

Accepted Date: September 30th, 2023

Abstract

Early detection of diabetes is very important to reduce the consequences caused by the disease. Diabetes is influenced by many factors, so to make a diagnosis requires a complex analysis. The dataset used to analyze the prediction of diabetes is using machine learning algorithm. The machine learning algorithm is used to classify someone with diabetes or not based on the factors that have been set as input. The results of the diagnosis/prediction that are not perfect are caused by many misclassifications. To reduce classification errors, it is proposed to apply decision tree and boosting techniques. The classification algorithm used in this study is Random Forest. The experimental results show that decision tree and boosting techniques and a combination of the two can reduce misclassification in diabetes prediction.

Keywords: diabetes; expert system; decision tree and boosting

Abstrak

Mendeteksi lebih dini terkait adanya penyakit diabetes sangat penting untuk mengurangi risiko yang diakibatkan oleh penyakit tersebut. Penyakit diabetes dipengaruhi banyak faktor, sehingga untuk melakukan diagnosa membutuhkan analisa yang kompleks. Dataset yang digunakan untuk menganalisa prediksi penyakit diabetes menggunakan algoritma machine learning. Algoritma machine learning digunakan untuk mengklasifikasikan seseorang terkena penyakit diabetes atau tidak berdasarkan faktor-faktor yang telah ditetapkan sebagai masukan. Hasil diagnosa/prediksi yang belum sempurna disebabkan oleh kesalahan klasifikasi yang masih banyak. Untuk mengurangi kesalahan klasifikasi, diusulkan penerapan teknik decision tree dan boosting. Algoritma klasifikasi yang digunakan pada penelitian ini adalah Random Forest. Hasil eksperimen menunjukkan bahwa teknik decision tree dan boosting serta perpaduan ke duanya dapat mengurangi kesalahan klasifikasi pada prediksi penyakit diabetes.

Kata kunci: diabetes; sistem cerdas; decision tree dan boosting

1. Pendahuluan

Diabetes merupakan penyakit tidak menular yang cukup serius dimana insulin tidak dapat diproduksi secara maksimal oleh pancreas (Safitri & Nurhayati, 2019). Insulin merupakan hormone yang mengatur glukosa. Insulin yang tidak bekerja dengan adekuat akan membuat kadar glukosa dalam darah tinggi. Kadar glukosa darah normal adalah 70-110 mg/dL pada saat berpuasa (Fatimah, 2015). Penyakit Diabetes Mellitus merupakan ranking keenam penyebab kematian di Dunia, hal

ini diungkapkan oleh dunia World Health Organization (WHO) (Wicaksono, 2015). Di Indonesia sendiri prevalensi diabetes menempati urutan ketujuh tertinggi di dunia setelah China, India, USA, Brazil, Rusia dan Mexico (Megawat, Agustini, & Krismayanti, 2020). Menurut laporan International Diabetes Federation (IDF) tahun 2020 Indonesia menempati urutan ke 7 dari 10 negara dengan jumlah pasien tertinggi di Dunia dengan jumlah lebih dari 10,8 juta jiwa (International Diabetes Federation, 2020). Banyak komplikasi

terjadi jika diabetes tetap tidak diobati dan tidak teridentifikasi. Diagnosa diabetes perlu dilakukan untuk mendeteksi penyakit secara dini, sehingga risiko akibat dari penyakit diabetes bisa diantisipasi lebih dini. Penyakit diabetes dipengaruhi banyak faktor, sehingga untuk melakukan diagnosa membutuhkan analisa yang kompleks.

Bagi orang awam, setidaknya harus mengenali tahapan awal gejala penyakit diabetes yaitu frekwensi buang air kecil meningkat hal ini disebabkan oleh kondisi gula darah yang tinggi sehingga ginjal harus bekerja keras untuk menyaring agar tubuh tidak kelebihan glukosa, gejala awal selanjutnya yaitu mudah merasa lapar dan haus hal ini disebabkan oleh hormon insulin bekerja kurang maksimal yang mengakibatkan terhambatnya pengolahan makanan menjadi energi, gejala awal berikutnya yaitu mudah merasa lelah dan mengantuk hal ini disebabkan oleh kurangnya penyerapan energi yang dibutuhkan oleh tubuh yang diperoleh dari makanan, gejala awal berikutnya yang umum sering terjadi yaitu merasa gatal yang berlebihan di area kemaluan hal ini disebabkan oleh infeksi jamur yang muncul di area yang mengandung gula tinggi. Pada tahapan selanjutnya penderita penyakit diabetes akan mengalami beberapa gejala, dalam kondisi ini penderita sudah mulai mengalami komplikasi dengan penyakit lain contoh beberapa gejala lanjutan yaitu penglihatan memudar, kerontokan pada rambut, otot-otot terasa kaku serta luka yang tidak kunjung sembuh (Kemenkes, 2019). Untuk mendeteksi penyakit diabetes saat ini harus dilakukan pemeriksaan oleh dokter melalui cek darah. Metode yang ada untuk deteksi diabetes adalah dengan menggunakan tes laboratorium seperti glukosa darah dan toleransi glukosa oral. Namun, cara konvensional dokter (Kemenkes, 2019) membutuhkan waktu lama, data yang dihasilkan dari hasil cek darah cukup kompleks (Mujumdar & Vaidehi, 2019).

Saat ini penyakit diabetes sering diketahui terlambat atau sudah masuk ke dalam tahap komplikasi sehingga proses untuk penyembuhan jadi lama. Orang-orang baru menyadari terkena penyakit diabetes setelah terjadinya komplikasi dan hasil laboratorium keluar, sehingga pencegahan pun terlambat jika sudah masuk ke dalam tahap ini. Untuk mengklasifikasi gejala diabetes secara cepat dan akurat, maka diperlukan data data yang valid dan metode yang handal agar dapat meminimalisir

kesalahan kesalahan yang akan ditimbulkan. Lalu data tersebut harus diproses agar bisa diterjemahkan menjadi diagnosa.

Metode yang diusulkan yaitu menggunakan metode Random Forest dan Boosting. Random Forest pertama kali dikenalkan oleh Breiman, Random Forest merupakan pengembangan dari Decision Tree yang digunakan untuk penggolongan yang terintegrasi. Dan dapat digunakan untuk mengevaluasi kontribusi setiap fitur dalam klasifikasi dengan menghitung gini dan permutasi (Wei, Zhao, Feng, Li, & Sun, 2018). Random Forest merupakan suatu prediksi klasifikasi yang kemudian dievaluasi dengan ukuran untuk menentukan mana yang harus menghasilkan pemisahan. Keacakan ini memberikan pohon yang berbeda, yang biasanya mencapai kinerja prediksi yang lebih baik yang kemudian digabungkan (Drummond & Resende, 2018). Boosting adalah metode ensemble yang sederhana dan sangat kuat dalam penerapan prosedur untuk algoritma machine learning varians tinggi (Lefebvre-Ulrikson, Da Costa, Rigutti, & Blum, 2016).

Berdasarkan latar belakang prediksi penyakit diabetes di atas, maka penelitian ini akan diterapkan algoritma Machine Learning dengan teknik ensemble boosting menggunakan dataset yang diunduh dari UCI Machine Learning Repository. Untuk mendiagnosa pasien terkena penyakit diabetes atau tidak, maka penelitian ini menerapkan metode Random Forest yang diharapkan dapat meningkatkan kinerja nilai akurasi dengan menggunakan teknik Ensemble pada Bagging yang dapat menangani masalah prediksi penyakit diabetes.

2. Metodologi

2.1 Metode Penelitian

Metode yang digunakan dalam penelitian ini adalah metode eksperimen. Eksperimen biasanya dilakukan dalam pengembangan, evaluasi dan pemecahan masalah proyek. Eksperimen biasanya dilakukan dalam pengembangan, evaluasi dan pemecahan masalah proyek. Kemudian data yang telah dipilih tersebut diimpor menggunakan aplikasi rapid miner untuk mengetahui Algoritma mana yang dapat meningkatkan kinerja yang terbaik dalam mencari nilai akurasi dan AUC paling tinggi. Setelah mengetahui Algoritma mana yang memiliki nilai akurasi dan AUC tertinggi

maka Algoritma tersebut akan digunakan pada penelitian ini. karena penelitian ini merupakan kegiatan yang telah direncanakan untuk memperoleh data agar dapat menjawab masalah dan menguji suatu hipotesis.

2.2 Metode Pengumpulan Data

Data yang digunakan dalam penelitian ini adalah data sekunder yang diperoleh dari UCI Dataset Machine Learning Repository. Data yang digunakan merupakan hasil pemeriksaan terhadap 520 pasien. Dataset yang digunakan terdiri dari 16 atribut dan 1 atribut tujuan. Atribut yang digunakan adalah age, gender, polyuria, polydipsia, sudden weight loss, weakness, polyphagia, genital thrush, visual blurring, itching, irritability, delayed healing, partial paresis, muscle stiffness, alopecia, obesity, dan class.

2.3 Metode Usulan

Metode yang diusulkan untuk penelitian ini adalah dengan cara menerapkan teknik ensemble boosting dengan pengklasifikasikan Random Forest menggunakan Machine Learning. Metode diawali dengan memasukkan dataset Prediksi Penyakit Diabetes. Kemudian data dibagi menjadi dua yaitu data latih dan data uji. Data uji akan melakukan test perulangan menggunakan 10-fold cross validation menggunakan teknik ensemble boosting. kemudian data tersebut akan diterapkan menggunakan pengklasifikasian Random Forest. Data uji dan data latih kemudian akan diuji dan divalidasi terlebih dahulu. kemudian akan didapatkan nilai akurasi dan Area Under Curve (AUC) yang terbaik dari dataset tersebut.

2.4 Perancangan

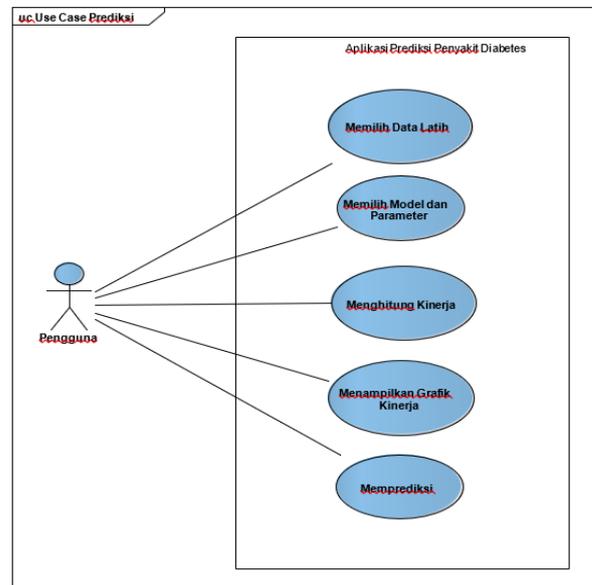
2.4.1 Perancangan Sistem

Perancangan sistem adalah merancang atau mendesain suatu sistem yang baik yang isinya adalah langkah-langkah operasi dalam proses pengolahan data dan proses prosedur-prosedur untuk mendukung operasi sistem. Tujuan dari perancangan sistem adalah untuk memenuhi kebutuhan para pemakai sistem serta memberikan gambaran yang jelas dan rancang bangun yang lengkap.

2.4.1.1 Use Case Diagram

Use case diagram digunakan untuk menggambarkan semua kasus antara aktor atau

pelaku dengan perangkat lunak untuk melakukan pekerjaan tertentu (Vani, Kumari, Priya, & Harika, 2015). Seperti gambar berikut yang menggambarkan use case diagram dari aplikasi prediksi penyakit diabetes.

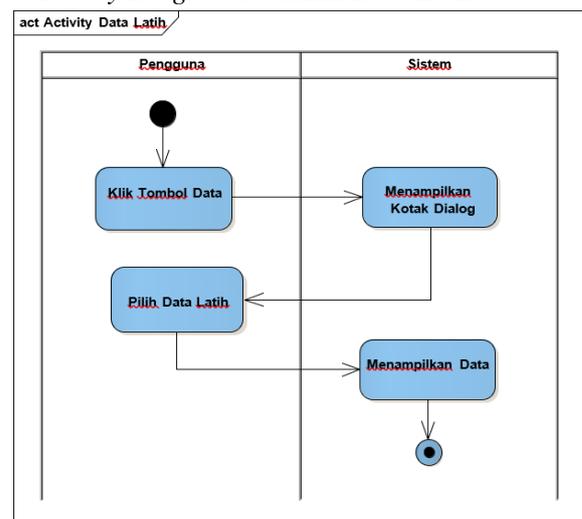


Gambar 1 Use Case Aplikasi Prediksi Penyakit Diabetes

2.4.1.2 Activity Diagram

Activity diagram digunakan untuk memodelkan aspek dinamis dari sistem berupa langkah-langkah yang berurutan. Berikut ini activity diagramnya:

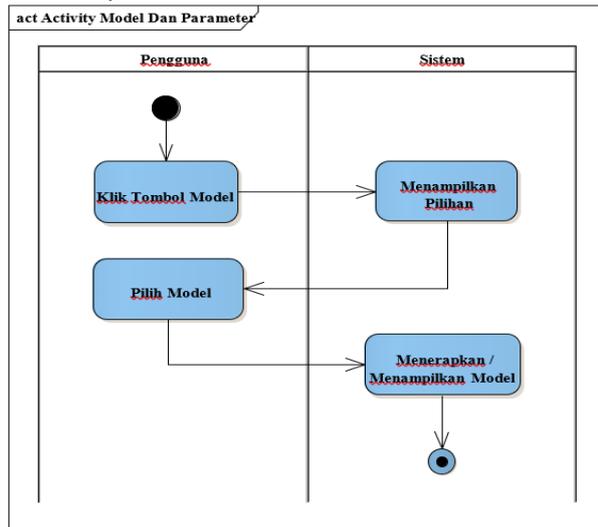
a. Activity Diagram Memilih Data Latih



Gambar 2 Activity Data Latih

Gambar activity data latih di atas menjelaskan pada pengguna bahwa si pengguna akan mengklik data latih terlebih dahulu, kemudian akan muncul tampilan kotak dialog dimana dataset yang akan diuji di simpan. Setelah itu pengguna pilih data latih dan sistem akan menampilkan data latih terpilih.

b. Activity Model dan Parameter

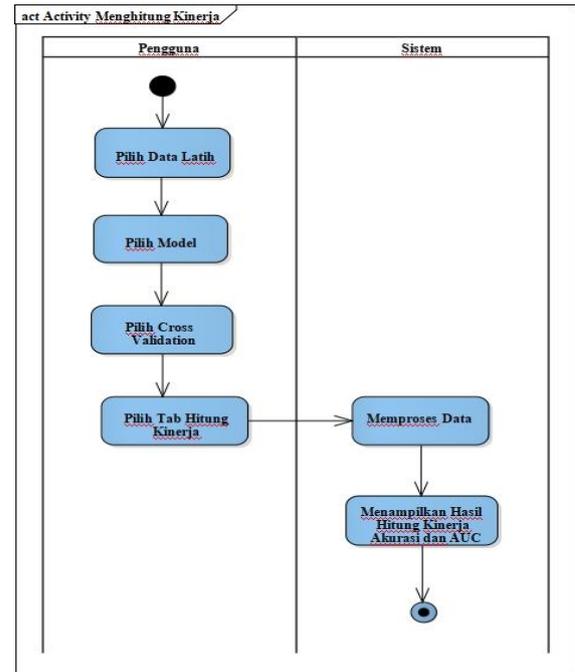


Gambar 3 Activity Model dan Parameter

Berdasarkan gambar activity model dan parameter di atas menjelaskan bahwa si pengguna mengklik tombol model kemudian akan muncul pilihan dari model yang mau di gunakan. Kemudian pilih model tersebut. Setelah selesai pilih model, lalu sistem akan menampilkan model yang akan dipilih.

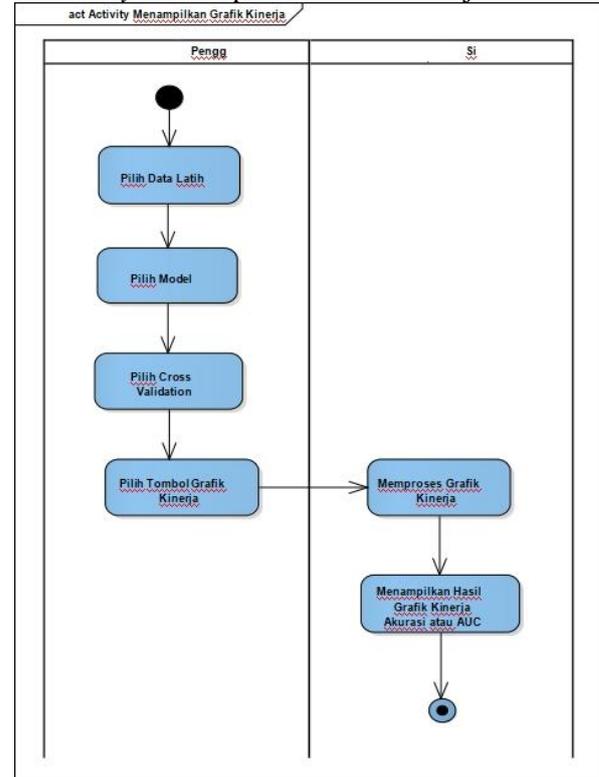
c. Activity Menghitung Kinerja

Activity menghitung kinerja pada Gambar 4 menjelaskan pada pengguna untuk memilih data latih terlebih dahulu, kemudian memilih model dan cross validation. Setelah itu pengguna diharuskan mengklik hitung kinerja, dan sistem akan memproses untuk menampilkan hasil dari akurasi dan AUC dari data yang telah pengguna pilih tersebut.



Gambar 4 Activity Menghitung Kinerja

d. Activity Menampilkan Grafik Kinerja

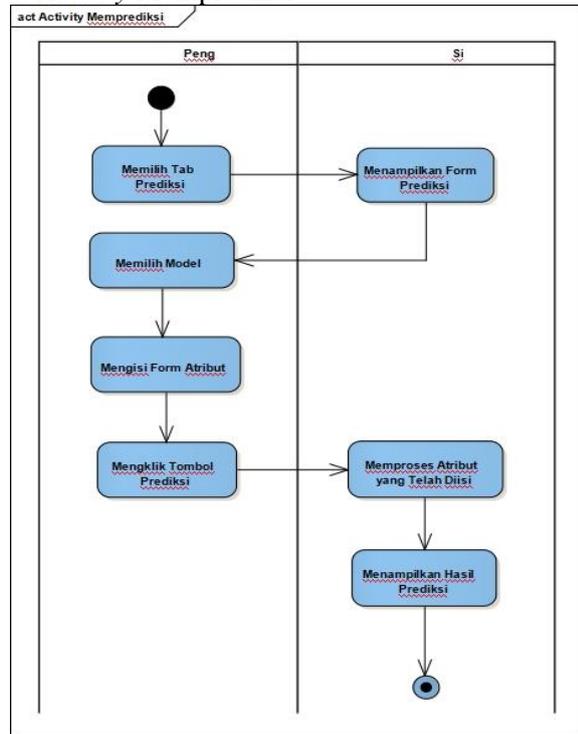


Gambar 5 Activity Grafik Kinerja

Gambar activity menghitung grafik kinerja diatas menjelaskan pada pengguna untuk memilih

data latih terlebih dahulu, kemudian memilih model dan cross validation. Setelah itu pengguna mengklik grafik kinerja. Lalu memilih grafik akurasi atau AUC dan sistem akan memproses untuk menampilkan hasil dari grafik kinerja akurasi dan AUC dari data yang telah pengguna pilih tersebut.

e. Activity Memprediksi



Gambar 6 Activity Prediksi

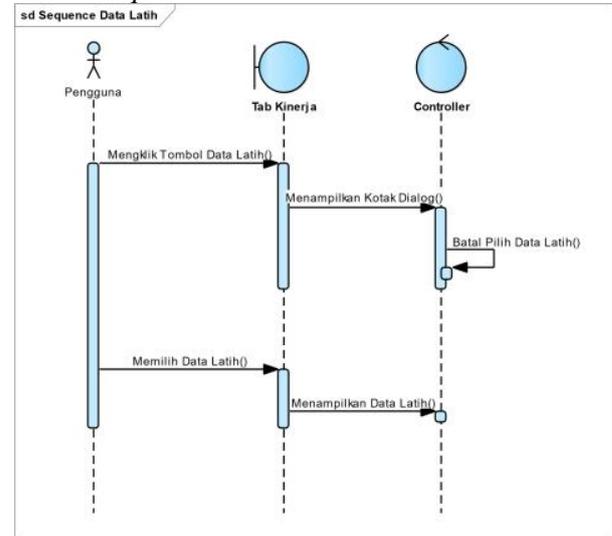
Gambar activity memprediksi diatas menjelaskan bahwa pada pengguna untuk memilih tab prediksi setelah itu akan tampil form prediksi, kemudian memilih model dan mengisi atribut. Setelah itu pengguna mengklik tombol prediksi. Dan sistem akan memproses atribut yang telah diisi dan juga sistem akan menampilkan hasil dari prediksi tersebut.

2.4.1.3 Sequence Diagram

Sequence diagram biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respon dari sebuah event untuk menghasilkan output tertentu. Diawali dari apa yang men-trigger aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan output apa yang dihasilkan (Vani, Kumari, Priya, & Harika, 2015). Seperti gambar di bawah

ini yang menggambarkan sequence diagram dari aplikasi prediksi penyakit diabetes:

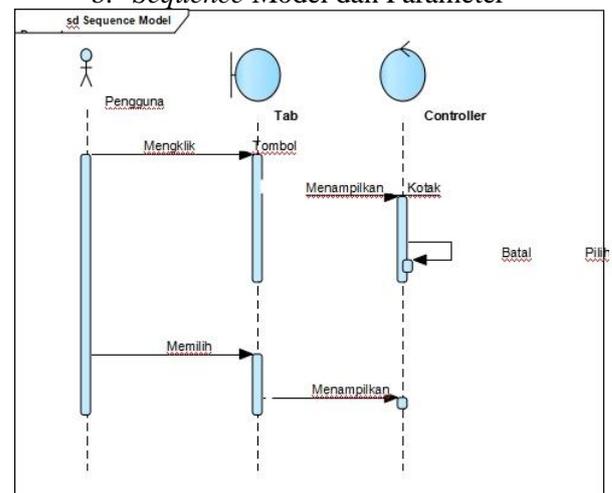
a. Sequence Data Latih



Gambar 7 Sequence Data Latih

Gambar sequence data latih diatas menjelaskan pertama-tama pengguna akan mengklik tombol data latih kemudian sistem akan menampilkan kotak dialog. Dan jika membatalkan untuk pilih data latih maka akan kembali ke menu kotak dialog. Setelah itu pengguna memilih data latih kembali dan sistem akan menampilkan data latih tersebut.

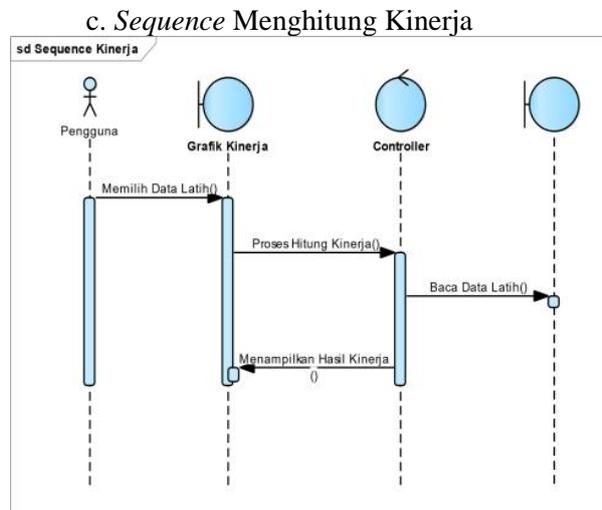
b. Sequence Model dan Parameter



Gambar 8 Sequence Model dan Parameter

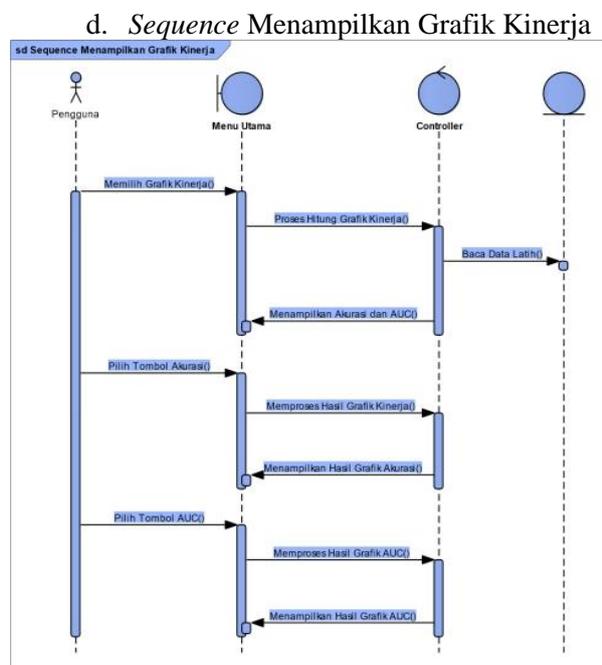
Gambar sequence model dan parameter diatas menjelaskan bahwa pengguna akan mengklik tombol model kemudian kemudian sistem akan

memproses dan dan menampilkan kotak dialog. Lalu pengguna memilih model yang ingin diuji terlebih dahulu, kemudian sistem akan menerapkan model tersebut.



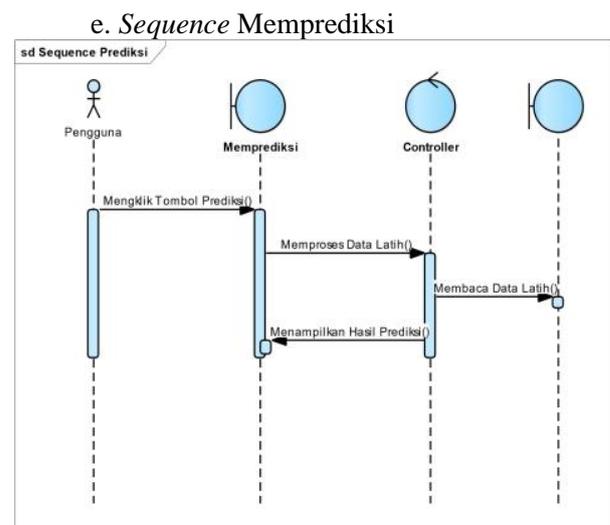
Gambar 9 Sequence Menghitung Kinerja

Gambar sequence menghitung kinerja diatas menjelaskan bahwa setelah pengguna memilih data latih kemudian akan dilakukan proses hitung kinerja. Lalu sistem akan membaca data latih untuk menampilkan hasil dari kinerja yaitu berupa akurasi , AUC dan Confusion matrix.



Gambar 10 Sequence Menghitung Kinerja

Gambar sequence menampilkan grafik kinerja diatas menjelaskan bahwa pengguna memilih grafik kinerja kemudian akan dilakukan proses hitung grafik kinerja. Lalu sistem akan membaca data latih untuk menampilkan hasil dari akurasi dan AUC. Kemudian pengguna mengklik tombol akurasi dan dan sistem akan memproses lalu akan menampilkan hasil grafik akurasi. Dan pengguna akan mengklik tombol AUC juga, kemudian hasil grafik akan di proses dan sistem akan menampilkan hasil grafik AUC.



Gambar 11 Memprediksi Penyakit Diabetes

Gambar sequence memprediksi diatas menjelaskan bahwa pengguna akan mengklik tombol prediksi, kemudian akan memproses data latih, lalu data latih akan terbaca. Dan sistem akan menampilkan hasil prediksi.

2.4.2 Desain Aplikasi

Aplikasi ini terdiri dari form kinerja dan form prediksi. Form kinerja akan menampilkan hasil kinerja dari setiap metode yang dipilih, hasil tersebut berupa nilai akurasi, AUC serta confusion matrix. Sedangkan form prediksi akan menampilkan prediksi dari data baru yang berupa nilai dari atribut yang ada. Setelah mengisi nilai atribut sistem akan memprediksi apakah data tersebut merupakan data yang teridentifikasi penyakit Diabetes atau tidak.

2.4.2.1 Desain Form Kinerja

Pada form kinerja, terdapat tombol pilih data latih untuk memilih kemudian memproses dataset tersebut. Dataset harus berjenis integer atau berupa

angka untuk dapat diproses. Hasil yang terdapat pada form kinerja berupa confusion matrix serta nilai dan juga grafik kinerja dari akurasi dan AUC. Seperti tampilan gambar di bawah ini yang menggambarkan desain dari form kinerja.

Gambar 12 Desain form kinerja

2.4.2.2 Desain Form Prediksi

Pada desain form prediksi terdapat tabel atribut serta kondisi dari atribut tersebut yang harus diisi untuk menampilkan hasil dari prediksi setelah menekan tombol prediksi. Seperti tampilan gambar di bawah ini yang menggambarkan desain dari form prediksi.

Gambar 13 Desain form Prediksi

3. Hasil dan Pembahasan

3.1 Persiapan Data

Dalam penelitian ini dataset yang telah diuji coba adalah data sekunder yang terdapat di website <http://archive.ics.uci.edu/ml/> atau UCI Machine Learning Repository. Dataset ini merupakan hasil pemeriksaan yang dilakukan terhadap 520 pasien, dimana jumlah pasien laki-laki sebanyak 328

pasien. Sedangkan jumlah pasien perempuan sebanyak 192 pasien. Dari 328 pasien laki laki dimana 147 pasien positif terdeteksi terkena penyakit diabetes dan 181 pasien lainnya negatif penyakit diabetes, sedangkan untuk pasien perempuan dari 192 pasien dimana 173 pasien positif terdeteksi terkena penyakit diabetes dan 19 pasien lainnya negatif penyakit diabetes. Dataset yang akan digunakan terdapat 16 atribut prediksi dan 1 atribut tujuan.

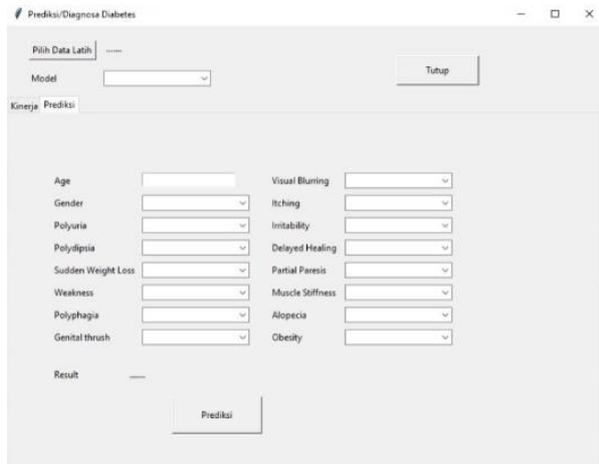
3.2 Tampilan Awal Aplikasi (Tab Kinerja)

Pada saat membuka aplikasi prediksi penyakit diabetes ini yang pertama muncul adalah tab kinerja. Tab kinerja terdiri dari pilih data latih (dataset yang akan diuji) yaitu data latih yang ingin diprediksi hasil kinerjanya. Kemudian setelah pilih data latih terdapat pilihan model yaitu Algoritma yang akan digunakan, lalu terdapat pilihan jumlah fold cross validation yang akan digunakan.

Gambar 14 Tampilan Awal Aplikasi Kinerja Penyakit Diabetes

3.3 Tampilan Tab Prediksi

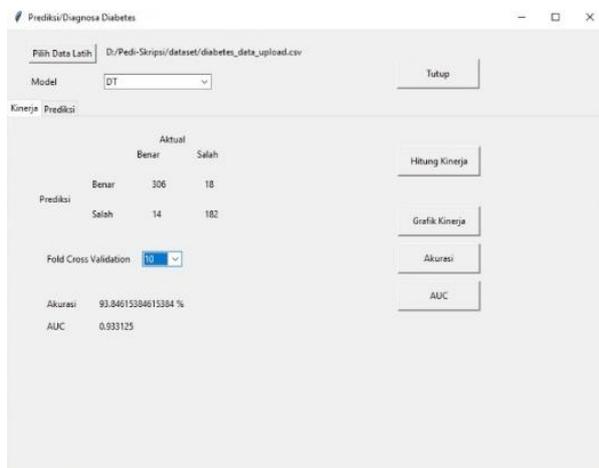
Untuk menampilkan hasil dari prediksi perlu dilakukan pengisian untuk data baru, data yang perlu diisi adalah nilai atribut yang terdapat pada dataset yang akan diuji. Setelah terisi didapatkan hasil apakah data yang di masukkan positif atau negative terkena Penyakit Diabetes. Berikut adalah tampilan dari tab prediksi.



Gambar 15 Tampilan Tab Prediksi Penyakit Diabetes

3.4 Implementasi Metode Hasil Kinerja Penyakit Diabetes Menggunakan Algoritma Decision Tree

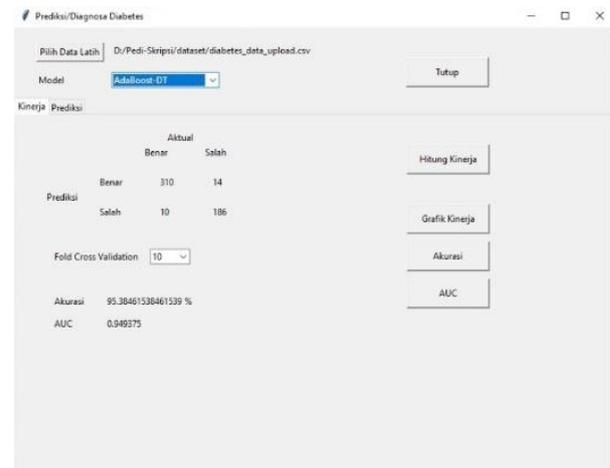
Pada penelitian ini eksperimen pertama menerapkan Algoritma decision tree. Hasil dari eksperimen pertama yang menggunakan Algoritma decision tree didapatkan akurasi sebesar 93,8461% dan AUC (Area Under Curve) sebesar 0,9331.



Gambar 16 Hasil Kinerja Menggunakan Algoritma Decision Tree

3.5 Hasil Kinerja Penyakit Diabetes Menggunakan Algoritma Decision Tree Dengan Mengoptimasi Teknik Adaboost (Boosting)

Hasil dari eksperimen yang menggabungkan menggunakan teknik adaboost didapatkan akurasi sebesar 95,3846 % dan AUC (Area Under Curve) sebesar 0,9494.

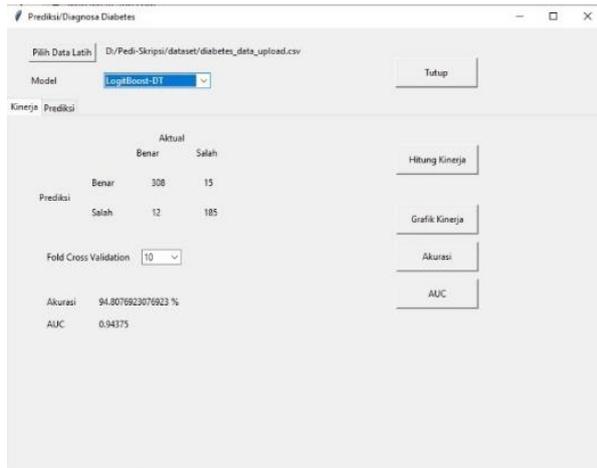


Gambar 17 Hasil Kinerja Menggunakan Algoritma Decision Tree Dengan Mengoptimasi Teknik Adaboost.

Pada gambar aplikasi di atas dataset penyakit diabetes dihubungkan dengan memasukkan model decision tree dengan mengoptimasi teknik adaboost kemudian akan menghasilkan data dari confusion matrix. Data yang mempunyai prediksi benar terkena penyakit diabetes ada 310 orang, sedangkan yang masih hasil prediksi benar-salah ada 14 orang. Sedangkan data yang mempunyai prediksi salah-benar tidak terkena penyakit diabetes ada 10 orang, sedangkan yang masih hasil prediksi masih salah ada 186 orang.

3.6 Hasil Kinerja Penyakit Diabetes Menggunakan Algoritma Decision Tree Dengan Mengoptimasi Teknik Logitboost (Boosting)

Hasil dari eksperimen yang menggabungkan menggunakan teknik logitboost didapatkan akurasi sebesar 94,8077 % dan AUC(Area Under Curve) sebesar 0,9437.



Gambar 18 Hasil Kinerja Menggunakan Algoritma Decision Tree Dengan Mongoptimasi Teknik Logitboost.

Data yang mempunyai prediksi benar terkena penyakit diabetes ada 308 orang, sedangkan yang masih hasil prediksi benar-salah ada 15 orang. Sedangkan data yang mempunyai prediksi salah-benar tidak terkena penyakit diabetes ada 12 orang, sedangkan yang masih hasil prediksi masih salah ada 185 orang.

Tabel 1 Perbandingan Hasil Kinerja

Model	Model Kinerja	
	Akurasi	AUC
Decision Tree	93,8461%	0,9331
Adaboost + Decision Tree	95,3846%	0,9494
Logitboost + Decision Tree	94,8077%	0,9437

4. Kesimpulan

Berdasarkan pengujian yang telah dilakukan dalam penelitian ini dalam penerapan boosting untuk meningkatkan ketepatan dalam memprediksi penyakit diabetes dengan menggunakan decision tree. Mekan kesimpulan dari penelitian ini adalah:

- Hasil dari kinerja algoritma decision tree dapat ditingkatkan atau dapat diperbaiki dengan menggunakan dataset yang telah diuji dalam penelitian ini. nilai kinerja dalam penelitian ini dapat menghasilkan akurasi sebesar 93,8461 %. Dan nilai AUC (Area Under Curve) sebesar 0,9331.
- Dengan menggunakan dataset yang telah dipilih dalam website UCI Machine Learning Repository pada algoritma decision tree yang dioptimasi dengan teknik adaboost dapat meningkatkan hasil akurasi

sebesar 95,3846 %. Dan nilai AUC (Area Under Curve) sebesar 0,9494. Dan dengan teknik logitboost menghasilkan akurasi sebesar 94,8077 %. Dan nilai AUC (Area Under Curve) sebesar 0,9437.

5. Saran

Berdasarkan hasil penelitian yang jauh dari kata sempurna, oleh karena itu dibutuhkan saran agar dapat lebih akurat dalam membuat aplikasi prediksi penyakit diabetes yaitu:

- Data yang diteliti harus lebih banyak agar menghasilkan prediksi yang lebih baik
- Penelitian ini dapat dikembangkan kembali dengan metode pengklasifikasian yang berbeda seperti KNearest Neighbor, Naïve Bayes, Neural Network dan lain-lain. Yang digabungkan dengan algoritma lain sehingga dihasilkan kinerja yang lebih akurat.
- Penelitian ini juga dapat dilakukan dengan bekerja sama dengan instansi kesehatan seperti Rumah Sakit agar data yang dihasilkan lebih actual.

References

- Drummond, A., & Resende, P. (2018). A survey of random forest based methods for intrusion detection systems. *ACM Computing Surveys*, 51.
- Fatimah, R. (2015). Diabetes Melitus Tipe 2. *juke.kedokteran.unila, IV*, 1.
- International Diabetes Federation. (2020). *Naik 6,2 Persen Selama Pandemi Pasien Diabetes Indonesia Peringkat 7 di Dunia*. Jakarta: Kompas.Com.
- Kemendes. (2019). Tanda dan Gejala Diabetes. *Direktorat Pencegahan dan Pengendalian Penyakit Tidak Menular*, 1-3.
- Lefebvre-Ulrikson, W., Da Costa, G., Rigutti, L., & Blum, I. (2016). Data Mining. *In Atom Probe Tomography: Put Theory Into Practice*, 1.
- Megawat, F., Agustini, N., & Krismayanti, N. (2020). Studi Retrospektif Terapi Antidiabetik pada Penderita Diabetes Melitus Rawat Inap di Rumah Sakit Umum Ari Canti. *Jurnal Ilmiah Medicamento, VI (1)*, 28-32.
- Mujumdar, A., & Vaidehi, V. (2019). Diabetes Prediction using Machine Learning Algorithms. *International Conference on Recent Trends in Advanced Computing*, 292-299.
- Safitri, & Nurhayati. (2019). Pengaruh Pemberian Sari Pati Benguang Terhadap Kadar Glukosa Darah Pada Penderita Diabetes Mellitus Tipe 2. *Jurnal Ners, 1*.

- Vani, Kumari, Priya, & Harika. (2015). An Effective Language for Object Oriented Design-UML (Unified Modeling Language). *Int. Reserch J. Eng. Technol, II(5)*, 1212-1218.
- Wei, G., Zhao, J., Feng, Y., Li, G., & Sun, X. (2018, October). An Effective Gas Sensor Array Optimization Method Base on Random Forest. *Proceedings of IEEE Sensors*, 1-4.
- Wicaksono, A. (2015). Pengaruh pemberian ekstrak jahe merah (zingiber officinale) terhadap kadar glukosa darah puasa dan postprandial pada tikus diabetes. *Jurnal Majority, IV (7)*, 97-102.

