e-ISBN 978-623-93343-3-8
Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)
Jakarta-Indonesia, 20 April 2021

PERBANDINGAN AKURASI KLASIFIKASI PENYAKIT DIABETES MENGGUNAKAN ALGORITMA ADABOOST-RANDOM FOREST DAN ADABOOST- DECISION TREE DENGAN IMPUTASI MEDIAN DAN KNN

Taufik Hidayat[1], Syifa Sabrina Anelia[2], Rizki Indah Pratiwi[3], Nadya Salsabila[4], Desta Sandya Prasvita[5]
Informatika / Fakultas Ilmu Komputer
Universitas Pembangunan Nasional Veteran Jakarta
Jl. Rs. Fatmawati, Pondok Labu, Jakarta Selatan 12450
taufikh@upnvj.ac.id[1], syifaanelia@gmail.com[2], rizkiindah@upnvj.ac.id[3], nadyasal93@gmail.com[4],
desta.sandya@upnvj.ac.id[5]

Abstrak. Diabetes merupakan penyakit yang ditandai dengan tingginya kadar gula (glukosa) darah. Pada penelitian ini dilakukan pendekatan dalam memprediksi penyakit menggunakan metode *Adaboost-Random Forest* dan *Adaboost-Decision Tree* dengan imputasi Median dan KNN, sehingga terdapat 4 tahapan uji klasifikasi. Tujuan penelitian ini yaitu membandingkan hasil akurasi prediksi dari algoritma yang digunakan, sehingga didapatkan hasil akurasi prediksi terbaik. Data yang digunakan yaitu *Pima Indians Diabetes Dataset* (PIDD) yang bersumber dari Kaggle. Hasil penelitian ini didapatkan model terbaik yaitu klasifikasi menggunakan algoritma *Adaboost-Random Forest* dengan imputasi median, model tersebut menghasilkan nilai akurasi terbesar 0.7786458 pada uji ke-8. Sedangkan hasil untuk model yang menggunakan algoritma *Adaboost-Decision Tree* dengan imputasi median menghasilkan nilai akurasi sebesar 0.7721354 pada nilai *max_depth* = 7, model klasifikasi menggunakan algoritma *Adaboost-Random Forest* dengan imputasi KNN menghasilkan nilai akurasi 0.77083 pada uji ke-5, sedangkan model dari algoritma *Adaboost-Decision Tree* dengan imputasi KNN menghasilkan nilai akurasi sebesar 0.74609375 pada nilai *max_depth* = 7.

Kata Kunci: diabetes, *random forest*, *decision tree*, akurasi.

1 Pendahuluan

Diabetes mellitus merupakan suatu penyakit yang ditandai dengan meningkatnya kandungan glukosa dalam darah yang melebihi batas normal, hal tersebut dapat disebabkan oleh kurangnya produksi insulin didalam tubuh dimana penyakit ini memiliki karakteristik yang dapat menyerang manusia dari berbagai usia dan bersifat menahun (kronis) [1]. Menurut WHO pada tahun 2014 memperkirakan sebanyak 422 juta orang secara global dengan kategori dewasa yang memiliki usia diatas 18 tahun menderita diabetes, dimana diperkirakan wilayah Asia Tenggara dan Pasifik Barat memiliki jumlah terbesar yaitu sekitar 50% kasus diabetes dunia. *International Diabetes Federation* (IDF) Atlas 2017 mencatat bahwa penyakit diabetes di Indonesia mengalami peningkatan. Hal tersebut didukung oleh Kementrian Kesehatan pada tahun 2018 bahwa Indonesia berada pada peringkat keenam di dunia dengan jumlah penderita diabetes kategori usia 20-79 tahun sekitar 10,3 juta orang, dimana negara-negara seperti Tiongkok, India, Amerika Serikat, Brazil, dan Meksiko memiliki penderita lebih banyak sehingga berada pada peringkat diatasnya [2].

Penderita Diabetes mengakibatkan penyakit *cardiovascular* yaitu terjadinya gangguan pada jantung dan pembuluh darah dengan 2 hingga 4 kali lebih banyak dibanding dengan bukan penderita diabetes. Dampak dari penyakit

Diabetes Mellitus yaitu adanya kerusakan *vascular* mikro seperti *retinopati* yaitu gangguan pada mata dan *neuropati* yaitu kerusakan jaringan saraf [3]. Dengan adanya dampak yang diakibatkan oleh penyakit diabetes, maka dapat dilakukan prediksi awal.

Prediksi telah banyak dilakukan dalam bidang ilmu *computer science*. Algoritma yang biasa digunakan yaitu *Naive Bayes*, *Decision Tree*, SVM, dan lain-lain. Menurut penelitian Noviandi, yang melakukan prediksi penyakit diabetes dengan menggunakan algoritma *Decision Tree* C4.5. Hasil dari penelitian tersebut yaitu model prediksi memiliki akurasi 70,32% dengan menghasilkan 9 *rule*, dengan jumlah *class* tidak sebanyak 4 *rule* dan 5 *rule class* iya untuk melakukan prediksi penyakit *Diabetes Mellitus* [4].

Zehra dkk dalam penelitiannya melakukan perbandingan akurasi klasifikasi pada beberapa teknik *data mining* dengan menggunakan *preprocessing* dan *non-preprocessing* pada Pima Indians Diabetes Databases (PPID). Hasil yang didapatkan yaitu akurasi pada data yang dilakukan *preprocessing* lebih baik dibandingkan dengan *non-preprocessing* data. Hal ini menunjukkan betapa pentingnya melakukan *pre-processing* dalam melakukan teknik data mining [5].

Berdasarkan uraian diatas, maka penelitian ini akan membandingkan akurasi prediksi yang didapatkan dari model yang diuji (*Adaboost-Random Forest* dengan imputasi median, *Adaboost-Decision Tree* dengan imputasi median, *Adaboost-Random Forest* dengan imputasi KNN, dan *Adaboost-Decision Tree* dengan imputasi KNN) yaitu prediksi apakah pasien menderita penyakit diabetes terhadap wanita yang telah melahirkan dengan beberapa faktor lainnya.

2 Tinjauan Pustaka

2.1 Algoritma *Adaboost*

Algoritma *Adaptive Boosting* atau yang biasa dikenal sebagai *Adaboost*, diperkenalkan oleh Freund dan Schapire pertama kali pada 1997. Algoritma *Adaboost* merupakan salah satu algoritma *machine learning* yang digunakan untuk seleksi fitur dan melatih *classifiers*. Algoritma *Adaboost* dipakai bersamaan dengan banyak algoritma lain untuk meningkatkan kinerja klasifikasi dari sebuah algoritma pembelajaran yang sederhana, seperti digunakan untuk melakukan *boosting* kinerja *simple perceptron*. Hal tersebut dilakukan dengan melakukan kombinasi pada sekumpulan fungsi klasifikasi lemah untuk membentuk sebuah *classifier* yang lebih kuat yang kemudian diistilahkan dengan *weak learner*.

Dalam beberapa hal, Algoritma *Adaboost* kurang rentan terhadap masalah *overfitting*, jika dibandingkan dengan algoritma pembelajaran pada umumnya. *Boosting* mengacu pada kumpulan algoritma yang dapat mengkonversi *weak learners* untuk *strong learners*. Prinsip utama dari *boosting* adalah menyesuaikan urutan *weak learners* hanya sedikit lebih baik daripada tebakan acak, sementara *strong learners* dekat dengan kinerja sempurna seperti pohon keputusan kecil.

2.2 Algoritma *Random Forest*

Algoritma *Random Forest* merupakan suatu algoritma yang menerapkan suatu gabungan dari pohon keputusan, algoritma ini biasanya digunakan tujuan untuk meningkatkan hasil performa model yang diterapkan. Dimana masing-masing pohon keputusan akan mengamati suatu kondisi yang berbeda dan dilatih dengan suatu data yang bersifat acak, sehingga nantinya setiap pohon akan menunjukan probabilitas lalu menghasilkan model prediksi dengan hasil yang sesuai.

2.3 Algoritma *Decision Tree*

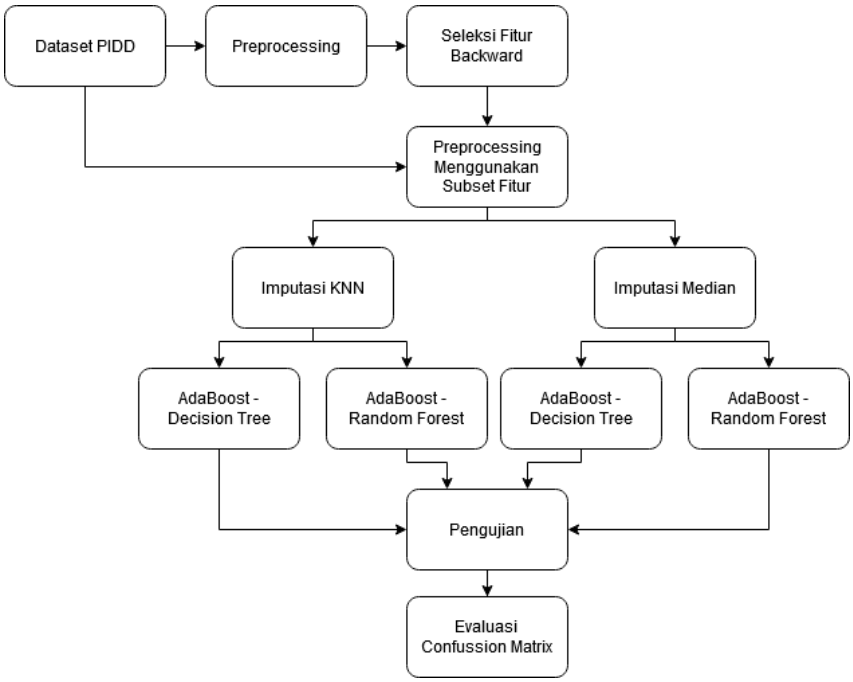
Algoritma Decision Tree merupakan salah satu algoritma klasifikasi yang berfungsi untuk membuat pohon keputusan. Algoritma *Decision Tree* ini sering digunakan karena memiliki kelebihan seperti dapat menghasilkan pohon keputusan yang mudah diinterpretasikan, memiliki tingkat akurasi yang dapat diterima, efisien dalam menangani atribut bertipe

diskrit dan dapat menangani atribut bertipe diskrit dan numerik.

3 Metodologi Penelitian

3.1 Kerangka Pikir

Untuk melaksanakan kegiatan penelitian, maka disusun beberapa langkah-langkah dari tahapan yang akan dijalankan untuk mencapai tujuan penelitian yang dilakukan. Berikut merupakan langkah-langkah yang dijalankan dapat diilustrasikan pada gambar 1.



Gambar. 1. Kerangka Pikir.

3.2 Dataset

Dataset yang digunakan dalam penelitian ini yaitu *Pima Indians Diabetes Dataset* (PPID) yang didapatkan dari *link* (*https://www.kaggle.com/uciml/pima-indians-diabetes-database*). Dataset ini sepenuhnya berasal dari *National Institute of Diabetes and Digestive and Kidney Diseases*. Data berasal dari pasien wanita setidaknya 21 tahun dari India Pima Heritage. Data terdiri dari 768 *record* dengan sebanyak 268 pasien terdeteksi sakit dan 500 pasien terdeteksi sehat. Informasi mengenai variabel pada dataset sebagai berikut.

Tabel 1. Informasi Variabel pada Dataset

Column	**Keterangan**
Pregnancies	Jumlah wanita melahirkan
Glucose	Kadar gula darah
BloodPressure	Tekanan darah
SkinThickness	Triceps skin fold thickness (mm)
Insulin	2-Hour serum insulin (mu U/ml)
BMI	Body mass indeks
DiabetesPedigreeFunction	Riwayat diabetes mellitus pada kerabat keturunan yang memiliki hubungan genetik dengan subjek
Age	Umur
Outcome	Class (0 atau 1)

3.3 *Preprocessing*

Akan dilakukan dua tahap *preprocessing*, yang pertama akan ditujukan untuk seleksi fitur *backward*. Pada tahap praproses ini ada beberapa proses yang dilakukan, diantaranya melakukan pemisahan antara variabel sumber dan variabel target lalu dilakukan penggantian (*replace*) data yang bernilai 0 (nol) menggunakan metode knn. Selain penggantian juga dilakukan normalisasi menggunakan normalisasi *MinMax*. Tahap praproses ini hanya untuk mendapatkan subset fitur dari proses seleksi fitur *backward*.

3.4 Seleksi Fitur *Backward*

Tahap ini yaitu melakukan seleksi fitur menggunakan metode *backward selection*. dimana pada program yang dibuat, proses seleksi fitur dijalankan setelah didahului oleh oleh proses klasifikasi. Hal tersebut terjadi karena diperlukan proses klasifikasi terlebih dulu agar dapat menghasilkan fitur mana saja yang paling berpengaruh.

3.5 *Preprocessing* Menggunakan Subset Fitur

Pada tahapan *preprocessing* yang kedua ini, akan digunakan data utuh sebelum dilakukan praproses yang pertama tetapi hanya berdasarkan subset fitur yang sudah didapatkan pada seleksi fitur *backward* sebelumnya. Pada praproses ini ditujukan untuk proses *training* dan *testing* dengan menggunakan atribut yang telah ditentukan. Praproses kali ini setelah dilakukan pemisahan antara variabel sumber dan variabel target, akan dilakukan dua imputasi berbeda untuk masing-masing algoritma. Selanjutnya akan dilakukan normalisasi dengan menggunakan metode *MinMax*.

3.6 Imputasi

Proses imputasi yang digunakan dalam penelitian ini yaitu menggunakan 2 metode yang berbeda, diantaranya dengan menggunakan metode KNN dan Median. Dimana *imputer* KNN serta Median yang digunakan dalam proses imputasi dapat melengkapi *missing values* yang terdapat didalam dataset yang digunakan. Setelah itu data akan dinormalisasi,

e-ISBN 978-623-93343-3-8
Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)
Jakarta-Indonesia, 20 April 2021

dimana proses tersebut berfungsi untuk melakukan perubahan fitur dengan menskalakan setiap fitur yang ada ke dalam suatu rentang tertentu.

3.7 Klasifikasi

Tahapan ini akan dilakukan proses untuk membangun model yang akan diterapkan dengan menggunakan 2 jenis algoritma yang berbeda. Pada penelitian ini algoritma yang akan digunakan diantaranya yaitu, *Adaboost- Random Forest* dan *Adaboost-Decision Tree* yang akan diproses sesuai dengan pengujian yang dilakukan. Kedua algoritma yang digunakan merupakan kombinasi dari algoritma *Adaboost*, *Random Forest*, dan *Decision Tree*. *Adaboost-Random Forest* merupakan gabungan antara *Adaptive Boosting* dan *Random Forest*, sedangkan *Adaboost-Decision Tree* adalah gabungan antara *Adaptive Boosting* dan *Decision Tree*. Data yang digunakan dalam proses ini akan dipisah (*split*) menjadi data *training* dan data *testing* dengan menggunakan metode *K-Fold Cross Validation*. Selanjutnya sesuai dengan masing-masing algoritma yang digunakan, akan dilakukan proses *training* menggunakan data *training* sehingga menghasilkan suatu model prediksi.

3.8 Evaluasi

Pada tahap evaluasi akan dilakukan proses evaluasi dengan memanfaatkan data *testing* yang telah didapat sebelumnya untuk menguji model prediksi yang telah terbentuk. Sehingga kita dapat melihat seberapa bagus performa model yang telah melewati proses *training* sebelumnya.

4 Hasil dan Pembahasan

4.1 Hasil Evaluasi Menggunakan *Adaboost-Random Forest*

Setelah data melewati proses imputasi, kemudian akan diproses untuk dilakukan pembagian dataset menjadi data *training* dan data *testing* dengan menggunakan *K-Fold Cross Validation*. Dimana pada penelitian ini nilai *K* yang digunakan yaitu 21. Setelah itu dilakukan proses *training* pada data *training* untuk membentuk model dengan menggunakan algoritma *Adaboost-Random Forest*. Selanjutnya dilakukan proses evaluasi pada model prediksi yang sudah terbentuk dengan menggunakan data *testing* yang sudah ada. Hasil evaluasi akan menunjukan performa algoritma dengan menghasilkan nilai akurasi, *sensitivity*, *specificity*, dan *error rate*.

4.1.1 *Adaboost-Random Forest* dengan Imputasi Median

Berikut merupakan hasil evaluasi dari pengujian dengan menggunakan algoritma *Adaboost-Random Forest* dengan imputasi Median. Hasil evaluasi akan ditunjukan melalui nilai akurasi, *sensitivity*, *specificity*, dan *error rate* pada tabel 2 dibawah ini.

Tabel 2. Hasil Evaluasi Model *Adaboost-Random Forest* dengan Imputasi Median

Uji	Akurasi	*Sensitivity*	*Specificity*	*Error Rate*
1	0,772135416666666	1,0	0,706867671691792	0,227864583333333
2	0,765625	1,0	0,699499165275459	0,234375
3	0,766327083333334	1,0	0,700167504187604	0,233072916666666

4	0,764322916666666	1,0	0,698333333333334	0,235677083333333
5	0,768229166666666	1,0	0,702838063439065	0,231770833333333
6	0,766927083333334	1,0	0,700668896321070	0,233072916666666
7	0,768229166666666	1,0	0,701842546063651	0,231770833333333
8	0,778645833333334	1,0	0,714765100671141	0,221354166666666
9	0,768229166666666	1,0	0,702341137123745	0,231770833333333
10	0,766927083333333	1,0	0,700167504187604	0,233072916666666

4.1.2 *Adaboost-Random Forest* dengan Imputasi KNN

Berikut merupakan hasil evaluasi dari pengujian dengan menggunakan algoritma *Adaboost-Random Forest* dengan imputasi KNN. Hasil evaluasi akan ditunjukan melalui nilai akurasi, *sensitivity*, *specificity*, dan *error rate* pada tabel 3 dibawah ini.

Tabel 3. Hasil Evaluasi Model *Adaboost-Random Forest* dengan Imputasi KNN

Uji	**Akurasi**	***Sensitivity***	***Specificity***	***Error Rate***
1	0,766927083333333	1,0	0,701168614357262	0,233072916666666
2	0,76171875	1,0	0,696517412935323	0,23828125
3	0,760416666666666	1,0	0,696369636963696	0,239583333333334
4	0,763020833333333	1,0	0,699173553719008	0,236979166666666
5	0,776833333333333	1,0	0,706666666666666	0,229166666666666
6	0,760416666666666	1,0	0,694352159468438	0,239583333333333
7	0,770833333333333	1,0	0,707154742096505	0,229166666666665
8	0,764322916666666	1,0	0,699335548172757	0,235677083333333
9	0,764322916666666	1,0	0,698835270542429	0,235677083333333
10	0,76171875	1,0	0,697019567549668	0,23828125

4.2 Hasil Pengujian Menggunakan *Adaboost-Decision Tree*

Setelah data melewati proses imputasi, kemudian akan diproses untuk dilakukan pembagian dataset menjadi data *training* dan data *testing* dengan menggunakan *K-Fold Cross Validation*. Dimana pada penelitian ini nilai *K* yang digunakan yaitu 21. Setelah itu dilakukan proses *training* pada data *training* untuk membentuk model prediksi dengan menggunakan algoritma *Adaboost-Decision Tree*. Selanjutnya dilakukan proses evaluasi pada model prediksi yang sudah terbentuk dengan menggunakan data *testing* yang sudah ada. Hasil evaluasi akan menunjukan performa algoritma dengan menghasilkan nilai akurasi, *sensitivity*, *specificity*, dan *error rate*.

4.2.1 *Adaboost-Decision Tree* dengan Imputasi Median

Berikut merupakan hasil evaluasi dari pengujian dengan menggunakan algoritma *Adaboost-Decision Tree* dengan imputasi Median. Hasil evaluasi akan ditunjukan melalui nilai akurasi, *sensitivity*, *specificity*, dan *error rate* pada tabel 4 dibawah ini.

Tabel 4. Hasil Evaluasi Model *Adaboost- Decision Tree* dengan Imputasi Median

Max Depth	Akurasi	*Sensitivity*	*Specificity*	*Error Rate*
1	0,725260416666666	1,0	0,657467532467532	0,274739583333333
2	0,727864583333334	1,0	0,657937806873977	0,272135416666666
3	0,717447916666666	1,0	0,696003262642740	0,282552083333333
4	0,790685416666666	1,0	0,671074388165269	0,259119583333333
5	0,75390625	1,0	0,685	0,24609375
6	0,752604166666666	1,0	0,683860232945091	0,247395833333334
7	0,772135416666666	1,0	0,708333333333334	0,227869583333334
8	0,765625	1,0	0,699499165275459	0,234375
9	0,768229166666666	1,0	0,702341137123745	0,231770833333333
10	0,766927083333333	1,0	0,700167504187604	0,233072916666666

4.2.2 *Adaboost-Decision Tree* dengan Imputasi KNN

Berikut merupakan hasil evaluasi dari pengujian dengan menggunakan algoritma *Adaboost-Decision Tree* dengan imputasi KNN. Hasil evaluasi akan ditunjukan melalui nilai akurasi, *sensitivity*, *specificity*, dan *error rate* pada tabel 5 dibawah ini.

Tabel 5. Hasil Evaluasi Model *Adaboost- Decision Tree* dengan Imputasi KNN

Max Depth	Akurasi	*Sensitivity*	*Specificity*	*Error Rate*
1	0,717447916666666	1,0	0,647727272727272	0,282552083333333
2	0,720052083333333	1,0	0,696381578947368	0,279947916666666
3	0,717447916666666	1,0	0,647159471544715	0,282552083333333
4	0,731770833333333	1,0	0,663398692810457	0,268229166666666
5	0,733072916666666	1,0	0,661716171617161	0,266927083333333
6	0,740885416666666	1,0	0,671617161716171	0,259114583333333

7	0,79609375	1,0	0,678217821782178	0,25390625
8	0,74609375	1,0	0,676616915422885	0,25390625
9	0,7921875	1,0	0,672185930463576	0,2578125
10	0,739583333333333	1,0	0,672131147540983	0,260416666666666

5 Penutup

5.1 Kesimpulan

Berdasarkan hasil pengujian yang telah dilakukan pada dataset *Pima Indians Diabetes Dataset* (PPID) dengan menggunakan kombinasi 2 algoritma dan 2 metode imputasi yang berbeda. Dimana kombinasi tersebut diantaranya algoritma *Adaboost-Decision Tree* dengan imputasi Median, *Adaboost-Random Forest* dengan imputasi KNN, *Adaboost-Decision Tree* dengan imputasi Median, *Adaboost-Decision Tree* dengan imputasi KNN. Menunjukan bahwa hasil akurasi tertinggi dari kombinasi keseluruhan algoritma yang telah disebutkan sebelumnya, yaitu terletak pada model *Adaboost- Random Forest* dengan imputasi Median, di mana model tersebut menghasilkan nilai akurasi terbesar 0.7786458 pada uji ke-8. Sedangkan hasil untuk model yang menggunakan algoritma *Adaboost-Decision Tree* dengan imputasi median menghasilkan nilai akurasi sebesar 0.7721354 pada nilai *max_depth* = 7. Untuk model klasifikasi menggunakan algoritma *Adaboost-Random Forest* dengan imputasi KNN menghasilkan nilai akurasi 0.77083 pada uji ke-5, sedangkan model yang menggunakan algoritma *Adaboost-Decision Tree* dengan imputasi KNN menghasilkan nilai akurasi sebesar 0.74609375 pada nilai *max_depth* = 7.

5.2 Saran

Adapun penelitian yang telah dilakukan tidak terlepas dari kekurangan yang dapat diperbaiki sehingga dapat dijadikan pengembangan pada penelitian yang akan dilakukan pada masa yang akan datang. Oleh karena itu penulis memberikan saran yang dapat dilakukan untuk pengembangan selanjutnya, yaitu dengan melakukan tahapan *preprocessing* yang lebih baik dan berbagai model dengan teknik *data mining* lainnya untuk memprediksi penyakit diabetes, sehingga didapatkan akurasi prediksi yang lebih baik.

Referensi

[1] Mahendra, B. *et al*. (2008) *Care Yourself : Diabetes Mellitus*. 1st edn. Jakarta: Penebar Plus.
[2] Ulfa, N. M., Lubada, E. I. and Darmawan, R. (2020) *Buku Ajar Farmasi Klinis dan Komunitas : Medication Picture dan Pill Count Pada Kepatuhan Minum Obat Penderita Diabetes Mellitus dan Hipertensi*. 1st edn. Gresik: Graniti.
[3] "Retinopati Diabetik" [Daring]. Tersedia pada: https://www.alodokter.com/retinopati- diabetik. [Diakses: 4-Juni-2020].
[4] Noviandi, N. (2018). Implementasi Algoritma Decision Tree C4. 5 Untuk Prediksi Penyakit Diabetes. *Indonesian of Health Information Management Journal (INOHIM)*, *6*(01), 1-5.
[5] Sudarto. (2016). Analisis Penanganan Ketidakseimbangan Kelas dengan menggunakan Density Based Feature Selection (DBFS) dan Adaptive Boosting (Adaboost) [Master's thesis, Universitas Sumatera Utara]. http://repositori.usu.ac.id/handle/123456789/1891.
[6] Zehra, A., Asmawaty, T., & Aznan, M. (2014). *A Comparative Study on The Pre- Processing and Mining of Pima Indian Diabetes Dataset*. Technical report. 80, 98, 99, 102, 106, 138, 141, 142.