Jurnal Elektronik Ilmu Komputer Udayana
Volume 14, No 4. May 2026

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Klasifikasi Kualitas Air Layak Minum Menggunakan Algoritma Genetika dan Random Forest

Ni Made Ayu Pranasanthi Dewi[a1], I Made Widiartha [a2], I Dewa Made Bayu Atmaja Darmawan [a3], I Gusti Agung Gede Arya Kadyanan [a4]

[a]ProgramStudi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam, UniversitasUdayana, Jimbaran, Kuta Selatan, Badung, Bali, Indoesia

[1]dewi.2208561130@student.unud.ac.id
[2]madewidiartha@unud.ac.id
[3]dewabayu@unud.ac.id
[4]gungde@unud.ac.id

Abstrak

Akses terhadap air minum yang aman merupakan kebutuhan dasar yang krusial bagi kesehatan masyarakat, namun pemantauan kualitas air secara manual melalui laboratorium sering terkendala biaya dan waktu. Penelitian ini bertujuan mengimplementasikan model *artificial intelligence* untuk klasifikasi kualitas air layak minum (*potable*) dan tidak layak minum (*non-potable*) dengan Random Forest melalui seleksi fitur menggunakan Algoritma Genetika. Penggunaan Algoritma Genetika dimaksudkan untuk mengatasi tantangan pemilihan fitur yang relevan. Penelitian menggunakan *Water Quality Dataset* dengan 3.276 data dan 10 atribut. Tahapan pengolahan meliputi penanganan *missing value, Winsorization, Min-Max Normalization*, *Random Oversampling,* seleksi fitur dengan Algoritma Genetika, dan klasifikasi Random Forest. Hasil menunjukkan seleksi fitur menghasilkan subset terbaik dengan fitness 73,76%. Hasil klasifikasi ini kemudian dilakukan validasi *K-Fold Cross Validation*, mendapatkan hasil model dengan seleksi fitur mengalami penurunan akurasi dari 68,57% menjadi 66,58% dan presisi dari 64,86% menjadi 60,00%. Namun, recall meningkat dari 48,45% menjadi 51,66% serta F1-score dari 55,28% menjadi 55,47%, menunjukkan peningkatan kemampuan mengenali kelas negatif meskipun berdampak pada penurunan performa pada kelas positif.

Kata kunci: Kualitas Air, Klasifikasi, Random Forest, Algoritma Genetika, Seleksi Fitur, *Machine Learning*

1. Pendahuluan

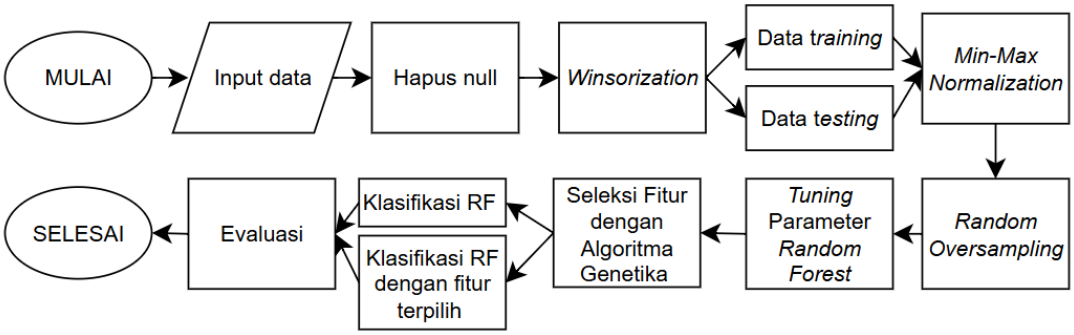
Studi Kualitas Air Minum Rumah Tangga (SKAMRT) pada tahun 2020 melaporkan bahwa hanya 11,9% masyarakat memiliki akses ke air minum yang aman, dan 40,8% menggunakan air tanah sebagai sumber air minum (selain air perpipaan dan depot air minum) [1]. Sebanyak 14,8% rumah tangga di Indonesia masih menggunakan sumur gali untuk keperluan minum, dengan tingkat risiko pencemaran yang tinggi [1]. Hal ini tentunya mengundang perhatian serius dikarenakan satu penyakit yang dapat berkembang pada air adalah diare [2]. Menurut WHO dan UNICEF terjadi sekitar 2 milyar kasus diare dan 1,9 juta diantaranya menyebabkan kematian pada anak balita di seluruh dunia disetiap tahun [1]. Meninjau uraian tersebut maka sangat penting dilakukan pemantauan terhadap kualitas air yang digunakan pada masyarakat untuk memastikan kelayakan dan keamanan konsumsi dari air yang digunakan. Salah satu pendekatan yang efektif untuk pemantauan dan pengelolaan kualitas air adalah penggunaan model kecerdasan buatan (*Artificial Intelegence*), khususnya *Machine Learning* (ML).

Beberapa penelitian menunjukkan bahwa pendekatan machine learning dapat digunakan untuk membantu proses klasifikasi kualitas air layak minum dan tidak layak minum berdasarkan parameter tertentu [3] [4]. Salah satu algoritma yang sering digunakan adalah Random Forest karena memiliki kemampuan dalam menangani data yang kompleks serta mengurangi *overfitting* [5]. Namun demikian, penelitian terdahulu dengan kasus serupa belum menerapkan optimasi seleksi fitur sehingga memungkinkan keberadaan fitur yang redundan dan dapat memengaruhi performa model

[3] [4]. Selain itu, Random Forest memiliki kelemahan dalam memperlakukan semua fitur secara merata, sehingga fitur yang tidak relevan dapat menurunkan performa model [6]. Oleh karena itu, diperlukan seleksi fitur untuk memilih atribut yang paling relevan dan informatif serta mengurangi kompleksitas data [7] [8]. Salah satu metode yang dapat digunakan untuk seleksi fitur adalah Algoritma Genetika yang bekerja berdasarkan prinsip seleksi alam dan mampu mencari solusi optimal secara global [9] [10] [11]. Genetic Algorithm dipilih karena menerapkan pencarian heuristik adaptif yang terinspirasi oleh proses seleksi alam dan genetika serta memiliki keunggulan dalam melakukan global optimization melalui population-based search yang dapat menghindari jebakan local optimum [12]. Genetic Algorithm mampu menangani permasalahan dengan ruang pencarian yang luas dan kompleksitas non-linear [13]. Karakteristik ini diperlukan dalam data penelitian yang digunakan dimana hubungan linear antar fitur tergolong rendah berdaarkan korelasi Pearson, sehingga diperlukan metode yang mampu mengevaluasi kombinasi fitur. Berdasarkan hal tersebut, penelitian ini bertujuan untuk menguji performa Random Forest dengan seleksi fitur menggunakan Algoritma Genetika dan membandingkan hasil sebelum dan sesudah seleksi fitur.

2. Metode Penelitian

Penelitian ini terdiri dari beberapa tahapan meliputi pengumpulan data, *preprocessing* data yang terdiri dari *data cleaning*, *Winsorization* pembagian data, serta normalisasi menggunakan metode *Min-Max*. Kemudian sampling data dilakukan dengan metode *Random Oversampling*, selanjutnya dilakukan seleksi fitur menggunakan Algoritma Genetika serta proses klasifikasi menggunakan Random Forest.
Alur penelitian dapat dilihat pada gambar 1.



Gambar 1. Alur Penelitian

2.1. Pengumpulan Data

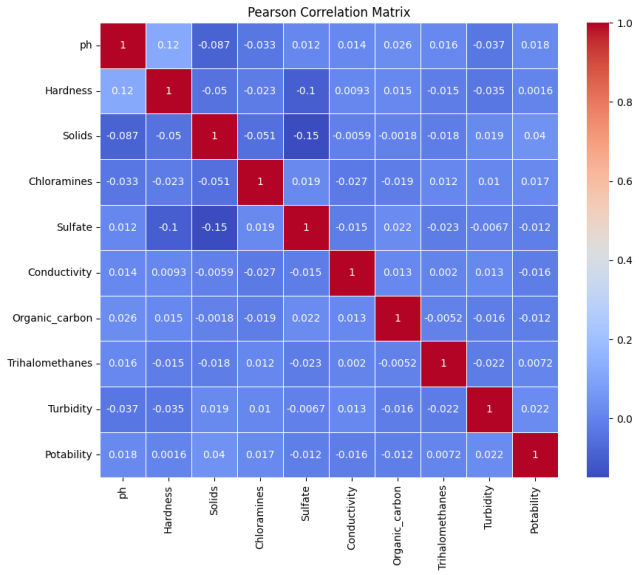
Penelitian ini menggunakan data sekunder berjudul "*Water Quality*" oleh Adiya Kadiwal yang diperoleh dari website *kaggle.com* dengan jumlah 3.276 data dan 10 atribut. Dataset ini diperbarui terakhir kali pada tahun 2021. Terdapat 2 kelas *output* kualitas air dalam dataset ini yaitu nilai 0 yang berarti tidak layak minum(*not potable*) sejumlah 1.954 data dan nilai 1 yang berarti layak minum (*potable*) sejumlah 1322 data. Adapun atribut dan domain dalam dataset yang terdapat dalam dataset ini ditampilkan pada tabel 1.

Atribut	Deskripsi	*Domain*
pH	Derajat keasaman atau kebasaaan dari air.	0-14
Hardness	Kapasitas air untuk mengendapkan sabun untuk menguji adanya kalsium dan garam magnesium.	47.4 - 323 mg/L
Solids	Jumlah padatan yang mampu dilarutkan.	321 - 61200 ppm
Chloramines	Kandungan kloramin dalam air.	0.35 - 13.1 ppm
Sulfate	Tingkat sulfat terlarut dalam air.	129 - 481 mg/L
Conductivity	Tingkat konduktifitas elektrik. Air murni tidak mampu mengalirkan listrik.	181 - 753 µs/cm
Organic Carbon	Tingkat karbon organik dalam air.	2.2 - 28.3 ppm
Trihalomethanes	Tingkat trihalometana dalam air.	0.74 - 124 µg/L
Turbidity	Ukuran sifat pemancaran cahaya dari air yang menunjukkan kekeruhan air.	1.45 - 6.74 NTU

Tabel 1. Atribut, Deskripsi, dan *Domain Dataset*

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Untuk melihat korelasi linear tiap atribut kepada label potability dilakukan dengan matriks korelasi Pearson. Matriks pada gambar 2 menunjukkan bahwa tidak terdapat fitur tunggal yang berhubungan linear kuat terhadap kelas potability maupun pasangan fitur yang memiliki korelasi tinggi secara linear. Maka digunakan Algoritma Genetika yang tidak hanya mempertimbangkan hubungan individu namun mengevaluasi kinerja dari kombinasi fitur.



Gambar 2. Matriks Korelasi Pearson

2.2. *Preprocessing*

Tahap preprocessing data dilakukan untuk mempersiapkan data sebelum digunakan dalam model. Preprocessing data dalam penelitian ini dibagi menjadi 3 bagian:

a. *Cleaning missing value*
Tahap ini dilakukan dengan menghapus baris data yang mengalami missing value agar tidak mengganggu proses perhitungan pada penelitian.

b. *Winsorization*
Tahap ini dilakukan untuk mengurangi pengaruh outlier tanpa menghapus data tersebut. Nilai winsorization ditentukan dengan mendeteksi persentase outlier pada batas bawah dan atas fitur, kemudian nilai ekstrem di luar batas tersebut disesuaikan ke nilai terdekat.

c. *Min-Max Normalization*
Normalisasi Min-Max dilakukan untuk menyamakan skala setiap fitur dengan menggunakan nilai minimum dan maksimum, sehingga memudahkan proses perhitungan dan diterapkan secara berulang pada seluruh fitur dalam dataset. Persamaan metode ini dapat dilihat pada persamaan 1 berikut:

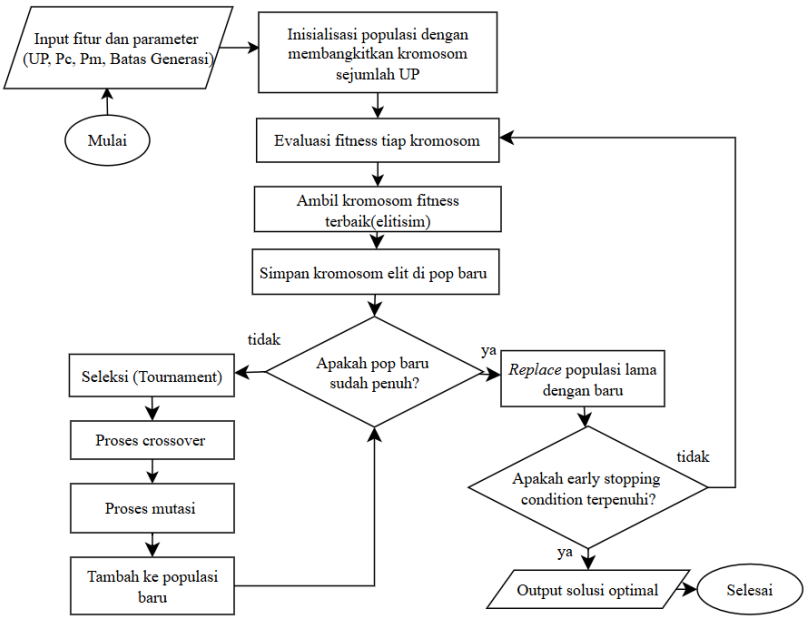
$$v_{norm} = \frac{v - min_A}{max_A - min_A}\left(new_{max_A} - new_{min_A}\right) + new_{min_A} \quad (1)$$

d. *Random Oversampling*
Random Oversampling digunakan untuk mengatasi ketidakseimbangan data dengan menambah jumlah data pada kelas minoritas melalui duplikasi acak, yang diterapkan pada data training agar distribusi kelas menjadi lebih seimbang.

2.3. Pembagian Data

Metode pembagian data dengan rasio 90:10 menggunakan *K-Fold Cross Validation* dilakukan dengan tujuan memvalidasi hasil dari model dengan menggunakan porsi data yang berbeda. *K-Fold Cross Validation* berfungsi untuk menilai kinerja proses sebuah metode algoritma dengan membagi sampel data secara acak dan mengelompokkan data tersebut sebanyak nilai K. Implementasi *K-Fold Cross Validation* akan dilakukan dengan jumlah *fold* 10.

2.4. Seleksi Fitur dengan Algoritma Genetika



Gambar 3. *Flowchart* Algoritma Genetika

Seleksi fitur dengan Algoritma Genetika dilakukan untuk mengetahui subset fitur paling optimal untuk klasifikasi data yang digunakan. Gambar 3 merupakan *flowchart* seleksi fitur dengan Algoritma Genetika yang akan dilakukan. Proses seleksi fitur menggunakan Algoritma Genetika dimulai dengan memasukkan seluruh fitur serta parameter yang digunakan, yaitu *population size*, *crossover rate (CR), mutation rate (MR),* dan batas generasi awal sebesar 1000 generasi. Selanjutnya dilakukan inisialisasi populasi dengan membangkitkan sejumlah kromosom sesuai ukuran populasi, di mana setiap kromosom merepresentasikan kombinasi fitur dalam bentuk bilangan biner 1 untuk fitur yang digunakan, dan 0 untuk fitur yang tidak digunakan. Setiap kromosom kemudian dihitung nilai *fitness*-nya berdasarkan akurasi yang diperoleh dari hasil klasifikasi Random Forest menggunakan fitur terpilih, di mana nilai fitness merepresentasikan kualitas solusi [12]. Perhitungan nilai *fitness* dilakukan dengan persamaan 2 seperti berikut:

$$F_i = fi \quad (1)$$

Keterangan :
Fi : Nilai *fitness* kromosom ke-i.
fi : Performa model (akurasi) kromosom ke-i.

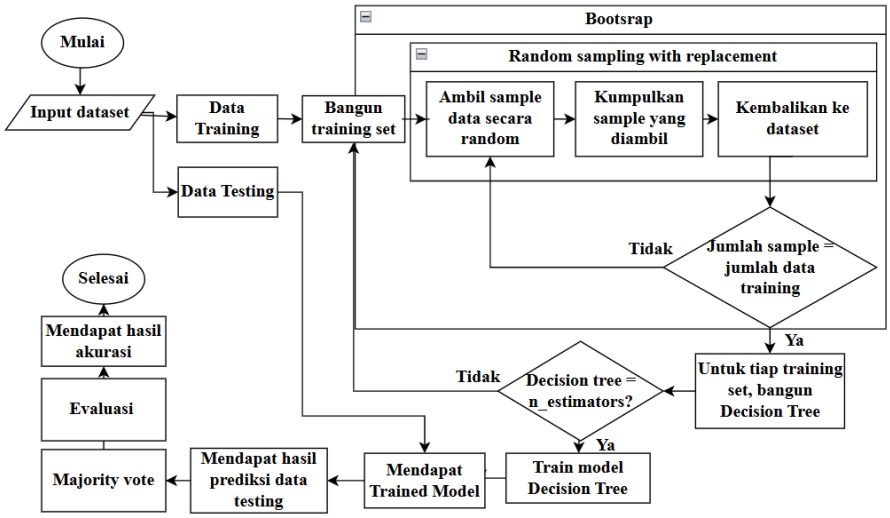
Kromosom dengan nilai *fitness* terbaik akan dipertahankan melalui proses *elitism* ke dalam populasi baru tanpa perubahan. Jika ukuran populasi baru belum terpenuhi, maka dilakukan proses seleksi. Seleksi dilakukan dengan model *tournament selection* yaitu dengan memilih sejumlah k kromosom secara acak dari populasi, kemudian kromosom dengan nilai *fitness* tertinggi dari kelompok tersebut dipilih sebagai *parent* [13]. Nilai yang digunakan untuk *tournament selection* adalah k = 3 [14].

Parent yang terpilih selanjutnya melalui proses crossover menggunakan metode *one-point crossover* dengan probabilitas berdasarkan nilai *crossover rate* serta proses mutasi menggunakan metode *bit-flip*. Proses ini bertujuan untuk memperluas pencarian solusi dan menghindari konvergensi prematur [15]. Hasil dari proses *crossover* dan mutasi akan membentuk populasi baru yang kemudian dievaluasi kembali nilai *fitness*-nya. Proses ini dilakukan secara iteratif hingga mencapai kondisi konvergen, yaitu ketika tidak ditemukan solusi fitur baru dalam 10 generasi berturut-turut, atau hingga mencapai batas maksimum generasi yang telah ditentukan.

p-ISSN: 2301-5373
e-ISSN: 2654-5101

2.5. Klasifikasi dengan Random Forest

Random Forest merupakan metode machine learning berbasis Decision Tree yang diperkenalkan oleh Leo Breiman pada tahun 2001 dan bekerja dengan membangun banyak pohon keputusan dari sampel data yang diambil secara acak [16]. Proses klasifikasi Random Forest digambarkan dalam *flowchart* pada gambar 4.



Gambar 4. *Flowchart* Random Forest

Proses klasifikasi dimulai dengan inisialisasi data training dan data testing serta penentuan parameter yaitu *n_estimators, max_depth, max_features, min_samples_split,* dan *min_samples_leaf.* Selanjutnya dilakukan *bootstrap sampling* pada data training menggunakan teknik *sampling with replacement* sebanyak jumlah *n_estimators* untuk menghasilkan variasi data pada setiap pohon. Setiap sampel *bootstrap* digunakan untuk membangun Decision Tree melalui proses *feature bagging* yaitu pemilihan subset fitur secara acak pada setiap node [16]. Pada setiap node akar, dilakukan perhitungan Gini Impurity untuk mengukur kemurnian data menggunakan persamaan 3 berikut:

$$G = 1 - \sum_{i}^{n} P_i^2 \qquad (3)$$

Keterangan :
G = Gini Impurity
P = Probabilitas kelas dalam split

Selanjutnya dilakukan pemisahan data berdasarkan fitur tertentu dan dihitung nilai Gini pada masing-masing cabang. Nilai *Average Gini* dihitung menggunakan persamaan 4 berikut.

$$Average\ G = \frac{Data\ ruas\ kiri}{Total\ data}(G\ kiri) + \frac{Data\ ruas\ kanan}{Total\ data}(G\ kanan) \qquad (4)$$

Split dengan nilai *Average Gini* terkecil dipilih sebagai *threshold*, dan proses pembentukan tree dilakukan secara rekursif hingga memenuhi kondisi seluruh data dalam node sudah homogen atau sudah mencapai batas maksimum kedalaman (*max_depth*) [16]. Proses ini diulang hingga terbentuk sejumlah Decision Tree sesuai nilai *n_estimators*. Setelah seluruh pohon terbentuk, dilakukan proses prediksi menggunakan data *testing*. Setiap data sampel akan melewati seluruh pohon dari *root* hingga *leaf* untuk mendapatkan hasil prediksi masing-masing pohon. Seluruh hasil prediksi kemudian digabungkan menggunakan metode *majority voting* untuk menentukan kelas akhir, di mana kelas dengan jumlah suara terbanyak menjadi hasil prediksi akhir.

2.6. Evaluasi Model

Pengujian performa sistem dilihat dari hasil prediksi data uji apakah sesuai kelas asalnya atau tidak. Hasil prediksi data uji dapat digambarkan dalam bentuk confusion matrix sebagai berikut.

Tabel 2. *Confusion Matrix*

	Positif Prediksi (1)	Negatif Prediksi (0)
Positif Aktual (1)	*True Positive*(TP)	*False Negative*(FN)
Negatif Aktual (0)	*False Positive*(FP)	*True Negative*(TN)

Confusion matrix digunakan untuk memperoleh nilai akurasi yaitu persentase jumlah data yang dilakukan pada klasifikasi atau prediksi secara benar oleh algoritma [17]. Persamaan untuk menentukan nilai akurasi, presisi, *recall*, dan *f1-score* sebagai berikut.

$$Accurracy = \frac{TP}{TP + TN + FP + FN} x100\% \quad (5)$$

$$Precision = \frac{TP}{TP + FP} \quad (6)$$

$$Recall = \frac{TP}{TP + FN} \quad (7)$$

$$F1 - score = \frac{2(Recall)(Precision)}{(Recall + Precision)} \quad (8)$$

Pengukuran performa klasifikasi diterapkan pada seluruh pendekatan model Random Forest yang digunakan dalam penelitian ini. Hasil akurasi, presisi, *recall*, dan *f1-score* tidak hanya menunjukkan performa sistem, tetapi juga menggambarkan pengaruh penggunaan seleksi fitur Algoritma Genetika dalam meningkatkan kemampuan model dalam mengklasifikasikan kualitas air. Skenario pengujian meliputi penggunaan dataset dengan fitur lengkap serta dataset yang telah melalui proses seleksi fitur, yang dilakukan untuk membandingkan kinerja model pada masing-masing kondisi.

Pengujian dilakukan dengan mengombinasikan parameter Random Forest dan parameter pada Algoritma Genetika. Skenario pengujian parameter Random Forest terdapat pada tabel 3.

Tabel 3. Skenario Pengujian Parameter Algoritma Genetika

Parameter	Deskripsi	Rentang
n_estimators	Jumlah tree yang dibangkitkan.	10, 50, 100, 150, 200, 250, 300, 350
max_depth	Maksimum kedalaman dari sebuah tree.	10, 15, 20, 25, 30
min_samples_split	Minimum sampel yang harus terpenuhi untuk melakukan split.	2
min_samples_leaf	Minimum sampel yang harus terpenuhi untuk membuat leaf node.	1

Parameter terbaik untuk model Random Forest akan digunakan dalam proses penghitungan nilai *fitness* dalam seleksi fitur. Sementara itu, parameter Algoritma Genetika yang digunakan meliputi ukuran populasi (*population size), mutation rate,* dan *crossover rate* dengan skenario pengujian pada tabel 4.

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Tabel 4. Skenario Pengujian Parameter Algoritma Genetika

Parameter	Deskripsi	Rentang
population_size	Jumlah total keseluruhan individu (subset fitur) dalam satu populasi.	20, 40, 60, 80, 100
crossover_rate	Kemungkinan terjadinya crossover..	(0,2), (0,4), (0,6), (0,8), (1,0)
mutation_rate	Kemungkinan terjadinya mutation.	(0,05)

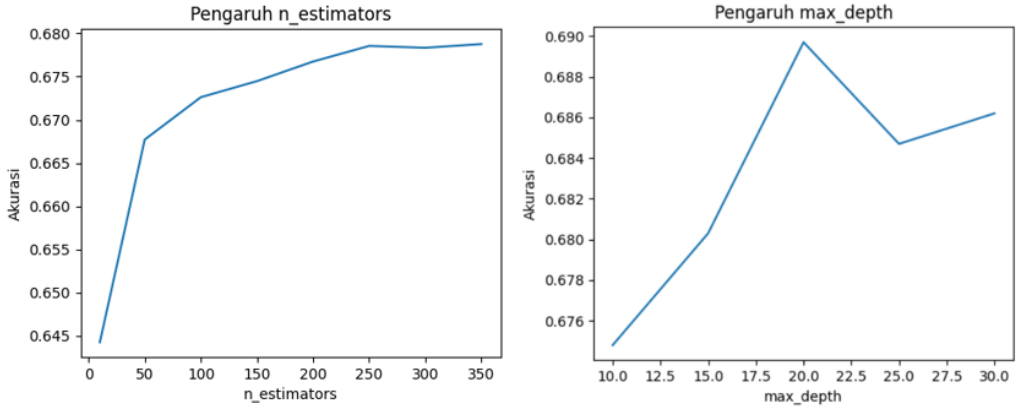
Setelah mendapat fitur terbaik dari Algoritma Genetika, dilakukan perbandingan model dari sebelum seleksi fitur dan setelah seleksi fitur, dengan melakukan pengujian lebih lanjut terhadap klasifikasi. Pengujian dilakukan dengan membandingkan skor klasifikasi pada Random Forest menggunakan fitur terpilih dan menggunakan seluruh fitur. Evaluasi menggunakan *Confusion Matrix* yang terdiri dari akurasi, presisi, *recall,* dan *F1-Score*.

3. Hasil dan Pembahasan

Hasil implementasi system menunjukkan beberapa perolehan skor klasifikasi model Random Forest setelah *tuning parameter*, subset fitur terbaik melalui seleksi fitur Algoritma Genetika, beserta hasil klasifikasi menggunakan fitur terpilih. Skor klasifikasi model Random Forest tanpa seleksi fitur dan dengan seleksi fitur ditampilkan bersama *confusion matrix* untuk melihat karakteristik hasil tersebut.

3.1. Pengujian Parameter Random Forest

Pada evaluasi ini diuji model Random Forest dengan melakukan tuning parameter untuk mendapatkan parameter dengan akurasi tertinggi. Pengujian dilakukan dengan 10-fold cross validation dengan perbandingan skor rata rata akurasi, presisi, *recall*, dan *f1-score*. Dalam pengujian ini didapatkan nilai akurasi rata rata terbaik sebesar 68,97% menggunakan kombinasi parameter yaitu *n_estimators* = 250 dan *max_depth* = 20. Dari pengujian ini didapatkan visualisasi pengaruh tiap parameter Random Forest.



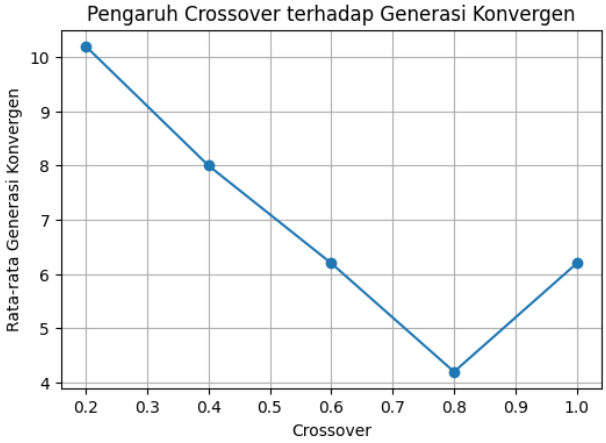
Gambar 5. Pengaruh Parameter Random Forest

Nilai akurasi meningkat seiring bertambahnya jumlah *n_estimators* dari 10 hingga 250 hingga tidak lagi menunjukkan peningkatan akurasi yang menandakan sudah mencapai konvergen. Peningkatan ini menunjukkan bahwa penambahan jumlah pohon dalam Random Forest berbanding lurus dengan peningkatan akurasi. Grafik selanjutnya menunjukkan bahwa akurasi meningkat dari *max_depth* 10 ke 20, kemudian menurun pada nilai 25 sebelum meningkat lagi pada 30. Dikarenakan akurasi masih fluktuatif, menunjukkan bahwa rentang yang diuji belum mencapai konvergen. Nilai *max_depth* yang mampu menghasilkan akurasi tertinggi yaitu 20.

3.2. Pengujian Seleksi Fitur Algoritma Genetika

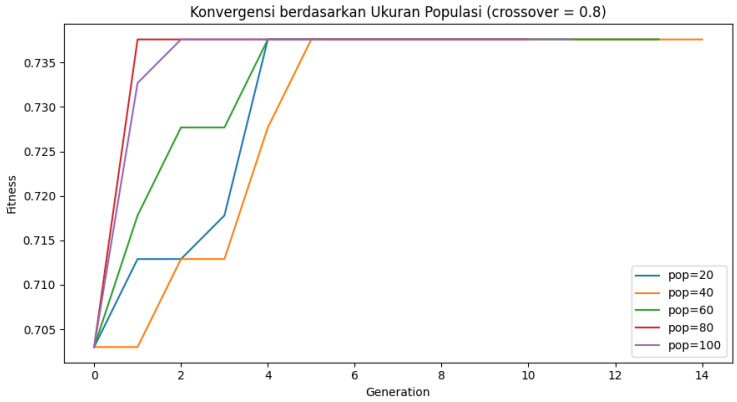
Pada evaluasi ini diuji performa Algoritma Genetika dengan melakukan *tuning* parameter untuk mendapatkan parameter terbaik yang mampu mencapai solusi subset fitur tertinggi dan konvergen lebih awal. ditemukan bahwa seluruh kombinasi parameter mencapai solusi terbaik yaitu sebanyak 5 fitur dengan indeks [0, 1, 3, 4, 7] yaitu *pH, Hardness, Chloramines, Sulfate,* dan *Trihalomethanes*. kombinasi parameter ukuran populasi 80 dengan *crossover rate* 0,8 mampu menemukan solusi

terbaik pada generasi kedua. Gambar 6 menunjukkan pengaruh *crossover rate* dengan rata-rata generasi konvergen.



Gambar 6. Grafik Pengaruh *Crossover Rate*

Pada grafik terlihat bahwa dari nilai 0,2 hingga 0,8 semakin besar crossover rate maka semakin sedikit generasi yang dibutuhkan untuk mencapai konvergen. Generasi konvergen paling kecil dicapai oleh nilai 0,8 yaitu rata-rata sebanyak 4,2 generasi, menunjukkan bahwa performa crossover rate 0,8 mampu mencapai konvergen lebih awal dibanding nilai lainnya.



Gambar 7. Konvergensi Berdasarkan Ukuran Populasi

Grafik selanjutnya menunjukkan ukuran populasi 80 dengan warna garis merah mampu mencapai fitness tertinggi lebih awal dibanding ukuran populasi lainnya. Namun fitness seluruh ukuran populasi pada akhirnya mencapai nilai yang sama, sehingga pada penelitian ini ukuran populasi tidak memengaruhi kualitas fitness namun memengaruhi seberapa cepat nilai fitness tersebut didapatkan.

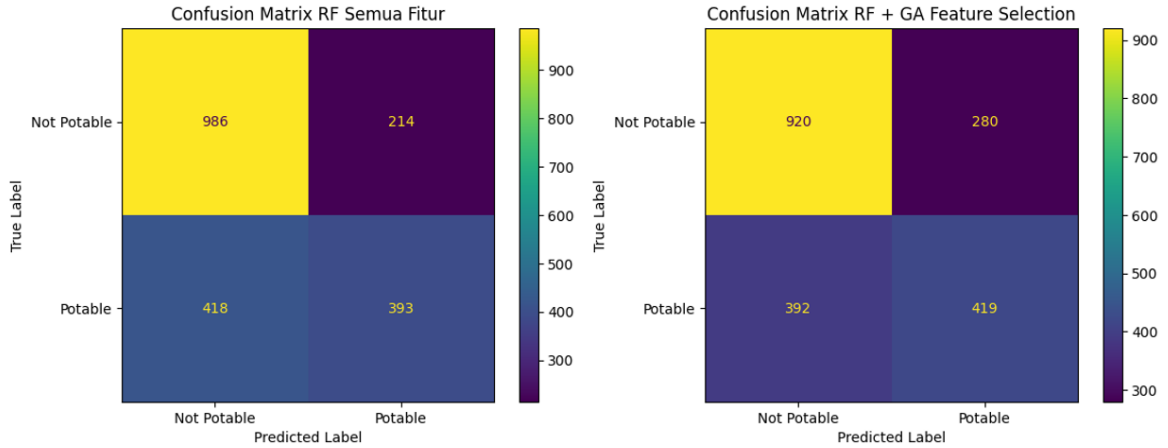
3.3. Pengujian Klasifikasi Random Forest

Pengujian model dilakukan menggunakan dataset fitur yang lengkap hingga yang telah diterapkan seleksi fitur Algoritma Genetika. Pada Random Forest tanpa seleksi fitur diperoleh nilai rata-rata akurasi sebesar 68,57%, presisi sebesar 64,86%, *recall* sebesar 48,45%, dan *f1-score* sebesar 55,28%. Berdasarkan *confusion matrix* pada gambar 3, model ini memiliki nilai *True Negative* sebesar 986 sedangkan *True Positive* hanya sebesar 393. Hal ini menunjukkan bahwa model memprediksi kelas 0 lebih baik dibanding kelas 1. Nilai presisi menunjukkan bahwa data yang diprediksi sebagai kelas 1, sebesar 64,86% benar kelas 1, sedangkan dari seluruh kelas 1 hanya 48,45% yang berhasil diprediksi sebagai kelas 1 oleh model.

Ketika diterapkan seleksi fitur, nilai rata-rata akurasi sebesar 66,58% menunjukkan penurunan 1,17% dan presisi sebesar 60,00% menunjukkan penurunan 2,80%. Pada nilai *recall* sebesar 51,66% menunjukkan peningkatan sebesar 3,01% dan *f1-score* sebesar 55,47% menunjukkan peningkatan 0,72%. Dari *confusion matrix* pada gambar 3 menunjukkan nilai *True Negative* sebesar 920 masih lebih tinggi dibanding *True Positive* sebesar 419. Adanya penurunan akurasi dan presisi ditunjukkan

p-ISSN: 2301-5373
e-ISSN: 2654-5101

dari peningkatan *False Positive* dari 214 menjadi 280, artinya model lebih sering memprediksi kelas 1 pada data yang sebenarnya kelas 0. Meskipun begitu, model ini mengalami peningkatan mendeteksi data kelas 1 ditunjukkan dengan peningkatan *True Positive* dari 393 menjadi 419, serta penurunan *False Negative* dari 423 menjadi 398. Ini menunjukkan model dapat lebih mengenali data kelas 1 dibanding sebelumnya. Penurunan akurasi setelah seleksi fitur menggunakan Algoritma Genetika menunjukkan bahwa fitur yang dihilangkan masih berkontribusi terhadap performa klasifikasi secara keseluruhan, terutama pada performa model dalam mengenali kelas positif meskipun penghapusan fitur membantu meningkatkan sensitivitas model terhadap kelas negatif.



Gambar 8. *Confusion Matrix* Klasifikasi Random Forest

4. Kesimpulan

Berdasarkan hasil penelitian, dapat disimpulkan bahwa penerapan seleksi fitur menggunakan Algoritma Genetika pada model Random Forest berhasil menghasilkan kombinasi fitur optimal yaitu *pH, Hardness, Chloramines, Sulfate,* dan *Trihalomethanes* dengan *fitness* sebesar 73,76%. Kombinasi parameter terbaik yang mampu menemukan solusi dengan cepat adalah *population size* = 80 dan *crossover rate* = 0,8 yang mencapai konvergen pada generasi kedua. Model Random Forest yang digunakan dalam penelitian ini menerapkan parameter terbaik hasil *tuning*, yaitu *n_estimators* = 200 dan *max_depth* = 20.

Meskipun demikian, penggunaan fitur hasil seleksi menunjukkan penurunan akurasi dari 68,57% menjadi 66,58% serta penurunan presisi dari 64,86% menjadi 60,00%. Penurunan ini ditunjukkan oleh peningkatan nilai *False Positive* dari 214 menjadi 280 serta penurunan *True Negative* dari 986 menjadi 920, yang mengindikasikan berkurangnya kemampuan model dalam mengenali kelas positif. Di sisi lain, terjadi peningkatan *recall* dari 48,45% menjadi 51,66% dan *f1-score* dari 55,28% menjadi 55,47% ditunjukkan oleh kenaikan *True Positive* dari 393 menjadi 419 serta penurunan *False Negative* dari 418 menjadi 392, yang menunjukkan bahwa model menjadi lebih baik dalam mengenali kelas negatif. Secara keseluruhan, hasil penelitian menunjukkan bahwa seleksi fitur menggunakan Algoritma Genetika mampu meningkatkan kemampuan model dalam mendeteksi kelas negatif, namun memberikan dampak pada penurunan performa model dalam mengenali kelas positif sehingga berpengaruh terhadap nilai keseluruhan.

Referensi

[1] Kemenkes Republik Indonesia, “Laporan Kinerja Direktorat Jenderal Pencegahan dan Pengendalian Penyakit,” Dirjen P2P, Jakarta, 2022.

[2] S. Manyullei, E. Nathalinri, H. A. Alfrial, I. Harsil, R. Andriany, A. Rahmadani, N. Jayanti, H. Wandi dan D. Alamsyah, “Penyuluhan Kesehatan Tentang Kualitas Air Layak Minum Dan Bahaya Konsumsi Air Mentah Bagi Kesehatan Anak Di Kelurahan Ma’rang Kabupaten Pangkep,” *JURNAL ALTIFANI,* vol. 4, no. 3, pp. 258-264, 2024.

[3] J. Li, “Prediction of Water's Safety for Consumption by Machine Learning,” *Proceedings of the International Conference on Mathematics and Machine Learning,* pp. 234-239, 2023.

[4] S. Kaloria, R. K. Saxena dan D. Bairwa, “Water potability prediction using big data analytics and

visualization tools," *IET Conference Proceedings CP912,* vol. 2024, no. 38, pp. 89-95, 2024.

[5] H. Tantyoko, D. K. Sari dan A. R. Wijaya, "Prediksi Potensial Gempa Bumi Indonesia Menggunakan Metode Random Forest Dan Feature Selection," *IDEALIS: Indonesia Journal Information System,* vol. 6, no. 2, pp. 83-89, 2023.

[6] I. Nawawi, "Optimisasi Pemilihan Fitur Untuk Prediksi Gagal Jantung: Fusion Random Forest Dan Particle Swarm Optimization.," *INTI Nusa Mandiri,* vol. 18, no. 2, pp. 122-128, 2024.

[7] G. H. Martono, R. Rismayati dan I. Karor, "Analisis Seleksi Fitur Menggunakan Metode ANOVA F-test dan Algoritma Random Forest Untuk Deteksi Diabetes," *CORISINDO,* vol. 1, pp. 133-142, 2025.

[8] C. Bielza dan P. Larrañaga, Data-driven computational neuroscience: machine learning and statistical models., Cambridge University Press., 2020.

[9] A. Tohari dan Y. P. Astuti, "Penerapan Algoritma Genetika Dalam Menentukan Rute Terpendek PT. Pos Cabang Lamongan," *MATHunesa: Jurnal Ilmiah Matematika,* vol. 11, no. 3, pp. 458-467, 2023.

[10] D. R. Syaputra, M. A. Ansya dan M. H. Dwinanda, "Prediksi Harga Emas Menggunakan Algoritma Genetik Pada Platform Pegadaian," *Jurnal Riset Informatika dan Teknologi Informasi (JRITI),* vol. 1, no. 2, pp. 53-56, 2023.

[11] T. Sutrisno, J. Setiawan dan D. Herwindiati, "Algoritma Genetika Dengan Roulette Wheel Selection dan Arithmetic Crossover Untuk Pengelompokan," *Jurnal Ilmu Komputer dan Sistem Informasi,* vol. 7, no. 1, pp. 58-64, 2019.

[12] L. A. Pramanda, A. Nugraha, C. Tapala, Z. Ramadhani dan A. Wardhana, "Implementasi Algoritma Genetika Untuk Optimalisasi Pid Control Pada Plant Separator Industri Migas," *Prosiding Seminar Nasional Teknologi Energi dan Mineral,* vol. 5, no. 1, pp. 609-621, 2025.

[13] F. Hamzah, R. Tsany, D. Zulfika dan T. Brian, "Desain Optimal Mini Turbin Angin untuk Penerangan Perahu Nelayan Menggunakan Algoritma Genetika dan Optimasi Partikel Swarm," *MASTER PPNS,* vol. 10, no. 1, pp. 82-93, 2025.

[14] R. Hartono dan A. Zein, "Penerapan Algoritma Genetika dan Jaringan Syaraf Tiruan Dalam Penjadwalan Mata Kuliah Studi Kasus: Prodi Sistem Informasi Universitas Pamulang," *urnal Ilmu Komputer,* vol. 6, no. 3, pp. 7-10, 2023.

[15] B. Rawat, D. Duwal, S. Phuyal dan A. Pant, "A comparative review between various selection techniques in genetic algorithm for finding optimal solutions," *ternational Journal of Computer Sciences and Engineering,* vol. 10, pp. 15-22, 2022.

[16] Y. Pyrih, M. Klymash, M. Kaidan dan B. & Strykhalvuk, "Investigating the efficiency of tournament selection operator in genetic algorithm for solving TSP," *IEEE 5th International Conference on Advanced Information and Communication Technologies (AICT),* pp. 170-173, 2023.

[17] I. W. Supriana, M. A. Raharja, I. M. S. Bimantara dan D. Bramantya, "Implementasi dua model crossover pada algoritma genetika untuk optimasi penggunaan ruang perkuliahan," *urnal RESISTOR (Rekayasa Sistem Komputer),* vol. 4, no. 2, pp. 167-177, 2021.

[18] N. Muna, F. L. Afriansyah dan A. B. Suprayogy, "Muna, N., Afriansyah, F. L., & Suprayogy, A. B. (2020). Penerapan Algoritma Random Forest Untuk Identifikasi Dehidrasi Berbasis Citra Urine," *Jurnal Informatika Polinema,* vol. 6, no. 3, pp. 49-54, 2020.

[19] D. A. R. A. Adha, A. N. Allanda, D. A. Fatmasari dan S. Narulita, "Performansi Algoritma C4. 5 untuk Prediksi Kelulusan Mahasiswa," *Jurnal Cakrawala Informasi,* vol. 3, no. 2, pp. 9-17, 2023.