ALGORITMA GENETIKA UNTUK OPTIMASI PARAMETER NILAI K PADA METODE K-NEAREST NEIGHBOR

Oleh
Ardina Surya Gracya[1], Ariesta Damayanti[2], Rudy Cahyadi[3]
[1,2]Informatika, Universitas Teknologi Digital Indonesia, Indonesia
[3]Teknologi Permainan, Polimedia, Jakarta, Indonesia
Email : [1] ardinasurya2299@gmail.com, [2] ariesta@utdi.ac.id, [3]masrudyc@gmail.com

Article History:
Received: 01-12-2024
Revised: 18-12-2024
Accepted: 02-01-2025

Keywords:
Genetic Algorithm, Classification, Drugs, K-Nearest Neighbor, Optimization

Abstrak: *Classification is a systematic grouping of objects into certain categories based on the common characteristics they have. One of the algorithms that is often used in classification is the K-Nearest Neighbor (KNN) algorithm, because the algorithm is easy to understand and apply, can be used on data that has many classes, and is effectively used for large data. However, this algorithm also has a weaknesses in making biased K parameters, resulting in reduced level of accuracy. Therefore, in this study, optimization of K parameters was carried out using a genetic algorithm.The data used in this study is a drug dataset sourced from the Kaggle platform. The total number of these datasets is 200 data records with 150 data used for training data and 50 data used for data testing. The overall data will be classified into five categories, namely drug category A, drug B, drug C, drug X, and drug Y. The classification is based on five criteria, namely gender, age, blood pressure, cholesterol, and sodium potassium. The best accuracy results obtained from the optimized KNN classification is 85% with parameter K=1. The accuracy is the same as the KNN search results without optimization by using the K parameter from a value range of 1 to 5 which produces the best accuracy when the parameter is at K = 1.*

PENDAHULUAN

Klasifikasi merupakan salah satu metode yang ada pada data mining. Metode ini digunakan untuk menemukan pola yang akan membedakan kategori atau kelas data, dengan tujuan agar dapat memperkirakan label yang belum diketahui dari suatu objek kelas data. [1]. Salah satu algoritma yang sering digunakan dalam klasifikasi adalah metode K-Nearest Neighbor (KNN). KNN juga merupakan teknik klasifikasi paling dasar dan sederhana karena bisa dengan efektif melakukan pelatihan data dengan jumlah yang banyak dalam waktu yang relatif cepat. Di sisi lain metode KNN memiliki kelemahan yaitu dalam penentuan parameter K yang masih bias, sebab belum ada rumus untuk menentukan parameter tersebut. Sehingga, tingkat akurasi dari kinerja KNN yang dihasilkan tidak optimal jika dalam menentukan parameter K kurang tepat [2]. Oleh karenanya, dibutuhkan suatu metode optimasi untuk mengoptimalkan penentuan parameter K.

Beberapa penelitian telah dilakukan untuk mengoptimalkan nilai parameter K dengan

cara menggabungkan algoritma KNN dengan beberapa algoritma, diantaranya penelitian [3] menggabungkan algoritma genetika dengan KNN dan terbukti bahwa algoritma genetika mampu membantu metode KNN menghasilkan nilai akurasi lebih tinggi dengan membantu dalam penentuan nilai parameter K yang bias. Algoritma ini dapat melakukan pencarian dengan sangat efektif untuk menyelesaikan permasalahan optimasi dengan mekanisme evolusi. Berdasarkan beberapa pokok permasalahan tersebut, maka dilakukan penelitian untuk mengoptimasi parameter K pada KNN menggunakan algorima genetika. Selanjutnya, digunakan confusion matrix untuk membandingkan hasil klasifikasi penggabungan motode KNN dan algoritma genetika dengan hasil klasifikasi yang seharusnya.

Penggunaan algoritma genetika pada optimasi nilai K pada algoritma KNN dilakukan dengan menggunakan skema pengkodean *binary encoding* dengan Proses persilangan atau rekombinasi menggunakan metode *one-cut point crossover.* Proses mutasi menggunakan metode *random mutation* dengan proses seleksi menggunkan metode *elitism*. Kualitas hasil klasifikasi dihitung menggunakan metode evaluasi *confusion matrix*. Sedangkan dataset yang digunakan yaitu dataset obat (*drug*) sejumlah 200 *record* data yang diperoleh dari *platform* Kaggle. Dimana dari data tersebut diambil sebanyak 75% untuk data latih dan sebanyak 25% untuk data uji. Untuk output adalah parameter K paling optimal dari algoritma genetika, hasil klasifikasi dari KNN, dan kualitas hasil klasifikasi dari *confusion matrix*.

LANDASAN TEORI

Algoritma *particle swarm optimization* (PSO) digunakan untuk mengoptimalkan parameter nilai K pada KNN dalam kasus pengendalian hama tanaman jeruk [4]. Penelitian tersebut membuktikan bahwa metode PSO-KNN mampu meningkatkan akurasi mencapai 96,25%. Akurasi tersebut lebih tinggi dibandingkan dengan hanya menggunakan metode KNN saja, yaitu sebesar 90%, sehingga PSO dapat memperbaiki kekurangan KNN Algoritma genetika untuk mengoptimalkan parameter nilai K pada MKNN dimana nilai fitness diperoleh dari perhitungan rata-rata validitas semua data latih terhadap nilai K. Hasil yang didapat dari proses optimasi pada kasus deteksi penyakit pada kucing ini menghasilkan tingkat akurasi hingga 100% ketika nilai K=1 [5]. Algoritma genetika dimanfaatkan untuk mengoptimalkan penentuan parameter nilai K pada KNN sehingga akurasi pada dataset iris meningkat hingga mencapai akurasi tertinggi sebesar 99%. Akan tetapi, untuk menghasilkan nilai K yang optimal algoritma genetika membutuhkan waktu yang cukup lama sehingga proses klasifikasi berjalan cukup lambat.[3]

Data Mining

Data mining merupakan proses menggali atau menambang (*mining*) data berukuran besar pada data *warehouse* menggunakan kecerdasan buatan,matematika dan statistik sehingga menghasilkan pengetahuan baru [6]. Teknologi data mining diharapkan bisa menjadi penghubung antara data dengan penggunanya.

Secara garis besar kegunaan data mining dibagi menjadi dua, yaitu kegunaan deskriptif yang berfungsi untuk mencari pola tertentu dari suatu data sehingga dapat digunakan untuk menemukan karakteristik yang mudah dipahami oleh manusia. Sedangkan untuk kegunaan prediktif berfungsi untuk menemukan model pengetahuan sehingga dapat digunakan untuk melakukan prediksi terhadap suatu data.

ISSN 2798-3471 (Cetak)
ISSN 2798-3641 (Online)

Metode Klasifikasi Data Mining

Metode klasifikasi merupakan teknik yang didasarkan pada atribut dari kelompok yang sudah didefinisikan. Sehingga didapatkan suatu aturan yang digunakan untuk melakukan klasifikasi pada data baru dengan cara memanipulasi data yang sudah ada dan sudah diklasifikasi [7].

Metode ini termasuk ke dalam kelompok *supervised learning* yang setiap *item* datanya memiliki label atau kelas yang dipengaruhi atribut. Tipe data yang cocok digunakan pada metode klasifikasi yaiu biner atau nominal sedangkan untuk tipe data ordinal kurang cocok sebab pada metode ini menggunakan pendekatan secara implisit [7].

K-Nearest Neighbor

Algoritma KNN didasarkan pada pembelajaran dengan analogi yaitu membandingkan data uji yang diberikan dengan data latih yang serupa. Dimana data latih dideskripsikan oleh n-atribut yang kemudian setiap *record* pada data latih disimpan dalam n-dimensi. Sehingga, ketika diberikan suatu *record* data yang belum diketahui maka KNN akan mencari pola untuk K data latih yang paling dekat dengan *record* yang belum diketahui [8].

Menurut [9] ada beberapa hal menarik pada algoritma KNN yaitu mudah diimplementasikan hanya menggunakan cara yang sederhana dengan menentukan satu parameter K dan algoritma KNN bekerja secara lokal dengan hanya memperhitungkan sejauh K data. Namun, disisi lain KNN juga memiliki kelemahan yaitu sangat sensitif terhadap *noise* ataupun *outlier* pada data. Selain itu, pada algoritma ini kesulitan menentukan parameter K dalam proses pelatihan. Parameter K yang optimal hanya bisa ditemukan secara empiris berdasarkan beberapa kali percobaan terhadap pola-pola representatif dengan jumlah yang memadai [9].

Algoritma Genetika

Algoritma genetika memiliki prinsip utama untuk meniru proses seleksi alam dimana setiap individu akan bersaing untuk bertahan hidup dalam melakukan reproduksi untuk menghasilkan keturunan. Pada algoritma genetika hanya individu-individu yang "*fit*" yang akan memiliki peluang untuk hidup dan sebaliknya individu yang kurang "*fit*" akan mati (prinsip *survival of the fittest*).

Pada proses seleksi alam akan "dilahirkan" individu baru yang lebih "*fit*" dari *parent*-nya melalui proses persilangan (*crossover*) dan mutasi. Pada algoritma genetika proses seleksi dan reproduksi (persilangan dan mutasi) akan terus berulang sampai dihasilkan individu baru yang "*fit*" [10].

Confusion Matrix

Confusion matrix digunakan untuk melakukan evaluasi kinerja pada metode klasifikasi dengan menganalisis tingkat akurasi dari *classifier* dalam mengenali *tuple* dari kelas yang berbeda. Ada beberapa istilah yang digunakan dalam *confusion matrix* yaitu TP (*True Positive*) dan TN (*True Negative*) memberikan informasi jika *classifier* benar sedangkan FP (*False Positive*) dan FN (*False Negative*) memberikan informasi ketika *classifier* salah [8].

Confusion matrix merupakan salah metode yang digunakan untuk mengukur performa dari suatu model klasifikasi yang telah dibuat, dimana *output* dapat berupa dua kelas atau banyak kelas. Dari *confusion matrix* maka dapat dihitung nilai akurasi yang merepresentasikan seberapa akurat model klasifikasi yang telah dibuat dalam melakukan pengklasifikasian secara benar menggunakan persamaan berikut [11].

$$\text{Akurasi } (2\ kelas) = \frac{\text{TP+TN}}{\text{TP+FP+FN+TN}} . 100\% \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

$$\text{Akurasi}(> 2 kelas) = \frac{\text{TP}}{\text{Total Data}} . 100\% \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3)$$

METODE PENELITIAN

Dalam penelitian ini digunakan dataset obat yang bersumber dari platform Kaggle, dataset tersebut berisi 200 *record* data dan memiliki 5 parameter yaitu usia, jenis kelamin, tekanan darah, kadar kolesterol, dan natrium-kalium. Kemudian parameter tersebut diklasifikasikan ke dalam 5 kategori yaitu obat A, obat B, obat C, obat X, dan obat Y.

Kebutuhan *Input*

Beberapa kebutuhan *input* yang diperlukan dalam perancangan aplikasi ini yaitu:

a. Jumlah populasi, merepresentasikan kemungkinan solusi K optimal pada KNN.
b. *Crossover* probability, merepresentasikan jumlah anak hasil persilangan individu dalam satu generasi.
c. Mutation probability, merepresentasikan laju mutasi dalam satu generasi
d. Dataset obat yang akan digunakan sebagai data latih dan data uji.

Kebutuhan Proses

Pada penelitian ini kebutuhan proses yang diperlukan dalam perancangan aplikasi yaitu:

a. Sistem melakukan proses pencarian nilai parameter K optimal dengan menggunakan algoritma genetika
b. Nilai K yang diperoleh dari proses algoritma genetika tersebut digunakan dalam proses klasifikasi dengan KNN
c. Kualitas hasil klasifikasi ditentukan dengan menggunakan metode evaluasi *confusion matrix*.

Kebutuhan *Output*

Dalam pembuatan aplikasi ini diperlukan beberapa kebutuhan *output*, yaitu:

a. Menampilkan hasil pencarian parameter *k* optimal
b. Menampilkan hasil klasifikasi
c. Menampilkan hasil kualitas klasifikasi.

Sumber Data

Data sekunder untuk mendukung penelitian ini bersumber dari Kaggle.com yaitu dataset pemberian obat pada pasien. Dataset tersebut berupa excel yang akan melakukan klasifikasi ke dalam 5 kategori yaitu obat A, obat B, obat C, obat X dan obat Y dengan atribut-atribut berikut:

a. Usia (age)
b. Jenis kelamin (sex)
c. Tekanan darah (*Blood Pressure*)
d. Kadar kolesterol (Cholesterol level)
e. Natrium Kalium (Na-Ka)

Preprocessing Data

a. Label Encoding

Data awal yang digunakan pada penelitian ini masih memiliki label dalam bentuk huruf. Sehingga label dalam bentuk huruf ini harus diubah dalam bentuk numerik supaya

ISSN 2798-3471 (Cetak)
ISSN 2798-3641 (Online)

dapat diproses oleh sistem untuk dilakukan pelatihan. Sehingga, dibutuhkan pembuat label encoding yang melakukan proses transformasi label huruf menjadi bentuk numerik. Pada tabel 1 disajikan pedoman untuk melakukan tranformasi nilai yang dibutuhkan pada penelitian ini.

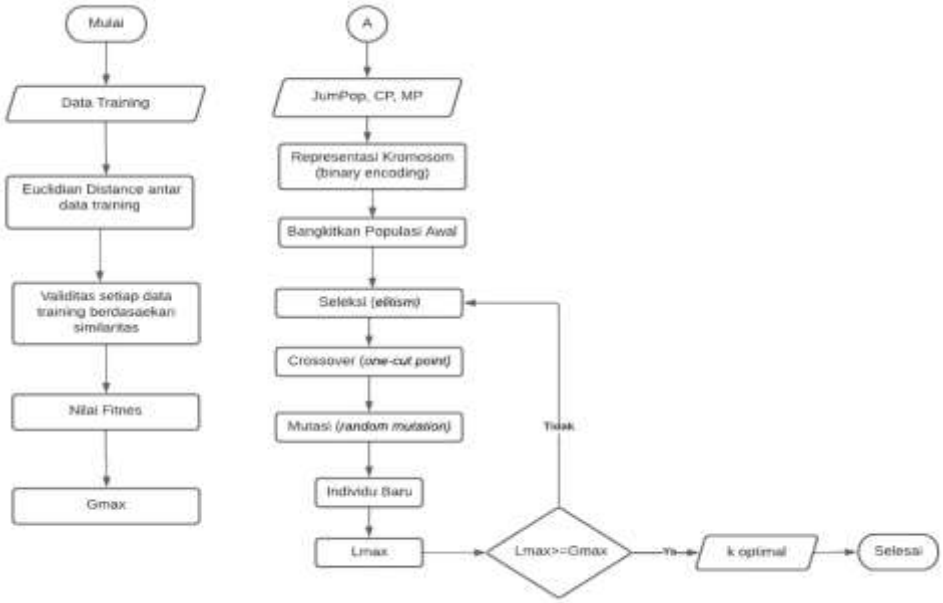
Tabel 1 Pedoman Transformasi Nilai

	Jenis Kelamin		**Takanan Darah (mmHg)**			**Kolesterol Total (mg/dL)**	
Nilai Asli (Kategorikal)	F (Wanita)	M (Pria)	Low (>90/60)	Normal (90/60-139/89)	High (140/90-≥180/110	Normal (<200-239)	High (≥240)
Nilai Transformasi (Numerik)	0	1	0	1	2	1	2

**sumber : Indonesia Heart Association dan NCEP ATP 2001*

Analisis dan Rancangan Sistem

Flowchart pada penelitian ini disajikan pada gambar 1.



Gambar 1. Flowchart Proses Pelatihan

Berdasarkan flowchart pada gambar 1 maka langkah-langkah proses pelatihan yang dilakukan, yaitu:

1. **Memasukkan Data Latih**

   Data latih diperoleh dari pembagian dataset (*split dataset)* obat. Pada penelitian persentase data yang digunakan adalah sebesar 75% data untuk data latih dan 25% untuk data uji.

2. *Euclidian Distance* Antar Data Latih

Perhitungan *eulidian distance* antar data latih dilakukan menggunakan rumus *Euclidian Distance* sehingga menghasilkan perhitungan dengan menggunakan persentase data latih sebesar 150 data disajikan pada tabel 2.

Tabel 2 Tabel Euclidean Distance Antar Data Latih

DATA	1	2	3	4	5	...	150
1	0	56.03274	31.16289	52.39095	35.43976		27.34671
2	56.03274	0	25.08174	24.09754	21.51391		29.20281
3	31.16289	25.08174	0	26.32913	5.543299		4.731342
4	52.39095	24.09754	26.32913	0	21.48986		27.61949
5	35.43976	21.51391	5.543299	21.48986	0		8.224101
...							
150	27.34671	29.20281	4.731342	27.61949	8.224101		0

$$(d1, d2) = \sqrt{\sum(18-18)^2 + (0-0)^2 + (2-2)^2 + (0-0)^2 + (8.75-8.75)^2} = 0$$

Berdasarkan hasil *euclidean distance* antar data latih pada tabel 2 maka selanjutnya dapat digunakan untuk menentukan nilai validitas.

3. Mencari Validitas Setiap Data Latih Berdasarkan Similaritas

Nilai validitas diperoleh dari hasil penjumlahan seluruh nilai similaritas. Sehingga diperoleh nilai validitas yang ditampilkan pada tabel 3.

Tabel 3 Tabel Nilai Validitas

	DATA 1	DATA 2	DATA 3	DATA 4	DATA 5	...	DATA 150
DATA 1	1	0	0	1.00	0.00		0
DATA 2	1	0	0.00	0.50	0.00		0
DATA 3	0.67	0	033	0.33	0.33		0
DATA 4	0.50	0.25	0.25	0.25	0.25		0
DATA 5	0.4	0.2	0.2	0.4	0.4		0
...							
DATA 150	0.43	0.29	0.29	0.29	0.29		0

4. Mencari nilai fitness

Hasil dari perhitungan nilai fitness dapat dilihat pada tabel 4.

Tabel 4 Perhitungan Nilai Fitness

	DATA 1	DATA 2	DATA 3	DATA 4	DATA 5	...	DATA 150	FITNESS
DATA 1	1	0	0	1.00	0.00		0	0.28
DATA 2	1	0	0.00	0.50	0.00		0	0.36
DATA 3	0.67	0	0.33	0.33	0.33		0	0.29
DATA 4	0.50	0.25	0.25	0.25	0.25		0	0.26
DATA 5	0.4	0.2	0.2	0.4	0.4		0	0.26
...								

ISSN 2798-3471 (Cetak)
ISSN 2798-3641 (Online)

DATA 150	0.43	0.29	0.29	0.29	0.29		0	0.23

5. Menentukan *Global maximum* (Gmax)

Penentukan *global maximum* didasarkan pada kromosom yang memiliki nilai fitness paling tinggi.

6. Representasi kromosom

Setiap gen yang terdapat dalam kromosom direpresentasikan ke dalam bentuk biner melalui pengkodean *binary encoding*. Pada tabel 5 merupakan penerapan pengkodean menggunakan *binary encoding*.

Tabel 5 Pengkodean Binary Enoding

Kromosom	Binary Encoding
2	00010
8	01000
15	01111
26	11010

7. Masukkan yang harus diinputkan:

a. Jumlah Populasi (JumPop) : 4
b. *Crossover Probability* (CP) : 0.8
c. *Mutation Probability* (MP) :0.2

8. Menginputkan dataset

Setelah pengguna menginputkan dataset maka secara otomatis sistem akan melakukan pembagian dataset yang telah diinputkan menjadi data latih yang akan digunakan pada proses pelatihan dan data uji yang akan digunakan pada proses pengujian.

9. Inisialisasi populasi awal

Cara membangkitkan populasi awal secara acak dengan sebelumya telah ditentukan jumlah individu pada populasi di langkah 7(a) dimana jumlah populasi awal yang telah dimasukkan adalah 4, sehingga secara acak akan dibangkitkan kromosom sebanyak 4 buah. Kemudian secara otomatis kromosom tersebut diuubah ke dalam bentuk biner supaya bisa dilakukan proses genetika. Tabel 6 menyajikan proses pembangkitan populasi awal.

Tabel 6 Proses membangkitkan populasi awal

Kromosom	Binary Encoding	Fitness
1	0001	0.28
5	0101	0.256
6	0110	0.2333
7	0111	0.2286

10.Seleksi (*elitism*)

Seleksi dilakukan dengan cara mencari individu yang memiliki nilai fitness terbaik dalam populasi untuk menjadi generasi selanjutnya seperti ditunjukkan pada tabel 7.

Tabel 7 Proses seleksi (elitism)

Parent	Kode Biner	Fitness
1	0001	0.28
5	0101	0.256

11.*Crossover* (*one-cut-point*)

Crossover dilakukan dengan cara memilih induk sesuai hasil seleksi yang telah dilakukan pada langkah ke-10. Kemudian menentukan apakah akan terjadi *crossover* atau tidak dengan cara membangkitkan bilangan acak dengan rentang nilai [0;1] jika nilai acak kurang dari nilai probabilitas *crossover* yang telah ditentukan pada langkah ke-7(b) maka akan dilakukan *crossover*, namun jika tidak maka tidak dilakukan *crossover* dan langsung ke langkah selanjutnya. Setelah ditentukan bahwa akan terjadi *crossover* maka selanjutnya akan memasangkan kromosom terpilih menjadi *parent* kemudian menentukan titik potong *crossover* secara acak pada rentang nilai sesuai jumlah gen. Sehingga jika terdapat 4 gen maka akan digunakan rentang nilai [1;4] Selanjutnya menukar gen-gen antar dua induk kromosom untuk menghasikan *offspring* (keturunan).

Tabel 8 Proses Crossover

Parent	Nilai Acak [0;1]	Nilai Acak [1;4]	Kode Biner Parent	Kode Biner Child
1	0.624 < 0.8	2	**00** 01	01 01
5			01 **01**	**00 01**

Berdasarkan tabel 8 nilai acak [0;1] digunakan untuk menentukan terjadinya *crossover*. Jika nilai acak < CP (0.8) maka individu yang terpilih akan melakukan *crossover*. Dan nilai acak [1;4] digunakan untuk menentukan titik potong *crossover*.

12.Mutasi (*random mutation*)

Metode ini mengambil secara acak satu *parent* dari sebuah populasi. Kemudian menambah atau mengurangi nilai gen terpilih dengan bilangan random yang kecil. Nilai bilangan random ditentukan secara acak pada rentang nilai [1;0]. Misalkan : *Mutation Probability* = 0.8 dan Jumlah Gen = 4.

Tabel 9 Tabel Proses Mutasi

Nilai Acak	MP	Status	Child Awal		Child Mutasi	
			1	2	1	2
0.91266	>0.1	bukan	0101	0001	0111	0011
0.98474	>0.1	bukan				
0.07729	<0.1	mutan				
0.54644	>0.1	bukan				

13.Populasi baru

Populasi baru merupakan gabungan dari individu hasil seleksi yang kemudian menjadi *parent* dan *child* dari hasil reproduksi (*crossover* dan mutasi) *parent*. Sehingga hasil dari populasi baru bisa dilihat pada tabel 10.

ISSN 2798-3471 (Cetak)
ISSN 2798-3641 (Online)

Tabel 10 Tabel Populasi Baru

		Biner	Decoding	Fitness
Populasi Baru	Parent	0001	1	0.28
		0101	5	0.256
	Child	0111	7	0.2286
		0011	3	0.2933

Hasil dan Pembahasan

Setalah dilakukan uji coba sistem dengan menggunakan dataset drug.csv, dimana dataset tersebut dibagi menjadi data latih sebesar 75% dan data uji sebesar 25%. Pada proses pelatihan menggunakan nilai CP sebesar 0.8, nilai MP sebesar 0.2, dan jumlah populasi sebanyak 4 individu. Dengan nilai-nilai tersebut diperoleh nilai K=1 dengan proses pada algoritma genetika disajikan pada tabel 11.Setalah dilakukan uji coba sistem dengan menggunakan dataset drug.csv, dimana dataset tersebut dibagi menjadi data latih sebesar 75% dan data uji sebesar 25%. Pada proses pelatihan menggunakan nilai CP sebesar 0.8, nilai MP sebesar 0.2, dan jumlah populasi sebanyak 4 individu. Dengan nilai-nilai tersebut diperoleh nilai K=1 dengan proses pada algoritma genetika disajikan pada tabel 11

Tabel 11. Proses Algoritma Genetika

No	Usia	Jenis Kelamin	Tekanan Darah	Kolesterol	Natrium Kalium	Klasifikasi Obat (Aktual)	Klasifikasi Obat (Model)
1	23	0	2	2	25.355	DrugY	DrugY
2	47	1	0	2	10.114	DrugC	DrugC
3	61	0	0	2	18.043	DrugY	DrugY
4	49	0	1	2	16.275	DrugY	DrugY
5	41	1	0	2	11.037	DrugC	DrugX
6	43	1	0	1	19.368	DrugY	DrugY
7	47	0	0	2	11.767	DrugC	DrugC
8	74	0	0	2	20.942	DrugY	DrugY
9	50	0	1	2	12.703	DrugX	DrugX
10	23	1	0	2	7.298	DrugC	DrugC
...							
50	23	1	2	2	8.011	DrugA	DrugX

Berdasarkan tabel 11 maka diketahui bahwa proses algoritma genetika berulang hingga 76 generasi. Perulangan pada algorima genetika akan berhenti ketika solusi yang didapatkan sudah optimal (*Gmax* ≥ *Lmax*). Hal tersebut dapat diketahui pada hasil mutasi di generasi ke-76 menghasilkan *child* [00000, 11101] jika dirubah ke dalam bentuk desimal maka child yang dihasilkan adalah 1 dan 29. Dari kedua hasil mutasi tersebut oleh sistem dicek apakah ada salah satunya yang memenuhi *Gmax*. Sehingga didapatkan solusi optimal K=1. Selanjutnya hasil parameter K yang didapatkan dari proses algoritma genetika digunakan untuk melakukan klasifikasi obat menggunakan KNN. Hasil proses pengujian tersebut disajikan pada tabel 12

Tabel 12. Proses Pengujian Klasifisikasi Obat

Generasi	Populasi	Fitness	Seleksi	Crossover	Mutasi
1	89,147,130,64	0.309, 0.312, 0.319, 0.316	130, 64	10000000, 01000010	10000010, 01001110
2	130, 64, 130, 78	0.319, 0.316, 0.319, 0.319	130, 130	10000010, 10000010	00000001, 10010010
3	130, 130, 1, 146	0.319, 0.319, 0.267, 0.314	130, 130	10000010, 10000010	10000010, 10000011
4	130, 130, 130, 131	0.319, 0.319, 0.319, 0.321	131, 130	10000011, 10000010	00010011, 00100000
5	131, 130,19, 32	0.321, 0.319,0.308, 0.339	32, 131	00100011, 10000000	00100011, 10000000
6	32, 131,35, 128	0.339, 0.321, 0.339, 0.319	32, 35	00100000, 00100011	00000000, 00100111

Dari hasil pengujian pada tabel 12 maka selanjutnya dilakukan evaluasi hasil klasifikasi berupa akurasi menggunakan persamaan 3.

$$\text{Akurasi} = \frac{\text{TP}}{\text{Total Data}} . 100\% = \frac{42}{50} . 100\% = 84\%$$

Sehingga didapatkan nilai akurasi dari kinerja KNN dengan optimasi menggunakan algortima genetika K=1 menghasilkan akurasi terbaik sebesar 84%. Untuk mengetahui apakah hasil optimasi parameter K yang diperoleh dari proses algoritma genetika merupakan parameter K yang paling optimal. Maka dibuatlah pengujian dengan mencari klasifikasi menggunakan KNN tanpa optimasi dengan algoritma genetika menggunakan nilai K=1 hingga K=5, kemudian dicari hasil evaluasi dari pengujian tersebut. Adapun hasil evaluasinya pada tabel 13.

Tabel 13 Hasil Evaluasi KNN Tanpa Optimasi

K	Akurasi
1	0.84
2	0.84
3	0.75
4	0.75
5	0.75

Berdasarkan tabel 13 di atas hasil evaluasi KNN tanpa optimasi menggunakan algoritma genetika menunjukkan bahwa parameter K yang menghasil nilai K dengan akurasi terbaik adalah pada K=1 dan K=2. Namun pada proses optimasi parameter K menggunakan algortima gentika dihasilkan nilai K paling optimal adalah 1. Hal ini karena pada sistem akan mencari kromosom terbaik berdasarkan nilai fitness yang paling tinggi. Sehingga terpilihlah kromosom ke-1 yang menjadi parameter K terbaik. Maka, dapat disimpulkan bahwa hasil pencarian parameter K menggunakan algortima genetika berhasil mendapatkan parameter K paling optimal dengan akurasi paling tinggi.

KESIMPULAN

Berdasarkan penelitian yang telah dilakukan dan pengujian yang telah dijalankan, maka dapat ditarik kesimpulan sebagai berikut:

ISSN 2798-3471 (Cetak)
ISSN 2798-3641 (Online)

1. Penerapan algoritma genetika dalam optimasi untuk penentuan parameter K pada KNN dilakukan dengan menentukan jumlah populasi sebanyak 4 individu, nilai CP sebesar 0.8, nilai MP sebesar 0.2, dan menggunakan *binary encoding* untuk representasi kromosom. Sehingga menghasilkan nilai K=1.
2. Hasil parameter K yang ter-optimasi digunakan untuk melakukan klasifikasi penentuan obat pada tahapan KNN. Kemudian hasilnya dilakukan evaluasi dan menghasilkan akurasi sebesar 84%. Akurasi tersebut sama dengan akurasi yang dihasilkan pada KNN tanpa optimasi menggunakan parameter K dengan rentang K=1 sampai K=5 menghasilkan akurasi paling tinggi ketika K=1. Sehingga, algoritma genetika dapat digunakan untuk melakukan optimasi pada parameter K untuk kasus klasifikasi penetuan obat pada penelitian ini.

Berdasarkan hasil penelitian yang telah dilakukan, maka diharapkan dalam pengembangan penelitian selanjutnya yaitu:

1. Menggunakan dataset dengan *record* data yang lebih besar dan memiliki banyak atribut yang saling mempengaruhi
2. Menggunakan algortima optimasi lainnya.

DAFTAR PUSTAKA

[1] Anjar Wanto, D. H. (2020). *Data Mining: Algoritma dan Implementasi.* Medan: Yayasan Kita Menulis.

[2] Zadlyka, T. (2021). *Optimasi Metode K-Nearest Neighbor (KNN) Menggunakan ParticleSwarm Optimization (PSO) untuk Diagnosis Penyakit Hati.* Palembang : Fakultas Ilmu Komputer Universitas Sriwijaya.

[3] Ibnu, D.L, Ardian, D.P (2017). *Optimasi K-Nearest Neighbour dengan Algoritma Genetika* :Jurnal Teknik Informatika.

[4] Kukuh, W.M, Yuita, A.S., & Achmad, A*. Optimasi K-Nearest Neighbour Menggunakan Particle Swarm Optimization pada Sistem Pakar untuk Monitoring Pengendalian Hama pada Tanaman Jeruk*.Malang:Fakultas Ilmu Komputer Universitas Brawijaya.

[5] Fitri, D.A , Dian, E.R., & Agus, W.W. *Deteksi Penyakit Kucing dengan Menggunakan Modified K-Nearest Neighbor Teroptimasi (Studi Kasus: Puskeswan Klinik Hewan dan Satwa Sehat Kota Kediri)*. Malang:Fakultas Ilmu Komputer Universitas Brawijaya.

[6] Jollyta, D., Ramadan, W., & Zarlis, M. (2020). *Konsep Data Mining dan Penerapan.* Sleman: Deepublish.

[7] Novriansyah, D., & Nurcahyo, G. W. (2015). *Algoritma Data Mining dan Pengujian.* Sleman: Deepublish.

[8] Han, J., Kamber, M., & Pei, J. (2012). *Data Mining Concepts and Techniques.* United States of America: Morgan Kaufmann.

[9] Suyanto. (2017). *Data Mining untuk klasifikasi dan klasterisasi data.* Bandung: Informatika.

[10] Arkeman, Y., Herdiyeni, Y., Hermadi, I., & Laxmi, G. F. (2014). *Algoritma Genetika Tujuan Jamak (Multi-Objective Genetic Algorithms): Teori dan Aplikasinya untuk Bisnis dan Agroindustri.* Bogor: IPB Press.

[11] Anggreany, M. S. (2022, Maret Senin). *Confusion Matrix*. Retrieved from Binus: https://socs.binus.ac.id/2020/11/01/confusion-matrix/

HALAMAN INI SENGAJA DIKOSONGKAN

ISSN 2798-3471 (Cetak)
ISSN 2798-3641 (Online)