Application of K-Nearest Neighbor (KNN) Algorithm to Predict Drinking Water Quality

Penerapan Algoritma K-Nearest Neighbor (KNN) untuk Memprediksi Kualitas Air Minum

Thomas Brian[1], Evi Nafiatus Sholikhah[2], Alief Nur Aisyi Maulidhia[3], Sekarsari Wibowo[4]

[1,2,3]Prodi Teknik Kelistrikan Kapal, Jurusan Teknik Kelistrikan Kapal, Politeknik Perkapalan Negeri Surabaya

[4]Prodi Teknik Pengolahan Limbah, Jurusan Teknik Permesinan Kapal, Politeknik Perkapalan Negeri Surabaya

E-mail: *[1]thomasbrian@ppns.ac.id, [2]evinafiatus@ppns.ac.id, [3]aliefnur@ppns.ac.id, [4]sekar@ppns.ac.id

Abstract – *The increasing need for quality drinking water requires the development of a reliable method to determine water potability. This study aims to apply the K-Nearest Neighbors (KNN) algorithm in predicting drinking water quality based on the Water Quality dataset from Kaggle. The dataset includes 3,276 data with 9 parameters, such as pH, hardness, and organic carbon content, as well as one target attribute that indicates consumption eligibility. This study will apply the KNN algorithm with various values (k), and evaluate the model performance using accuracy and Jaccard Similarity metrics. The results of the study show that the KNN algorithm in predicting drinking water quality achieves the best accuracy of 58% at a value of (k) = 2, these results indicate that this method is quite good although further development with other methods is needed to improve accuracy. This study contributes to the implementation of machine learning technology in water resource management.*

Keywords *— K-Nearest Neighbors, water quality prediction, data normalization, model accuracy, Jaccard Similarity*

Abstrak – Peningkatan kebutuhan akan air minum berkualitas menuntut pengembangan metode yang andal untuk menentukan potabilitas air. Penelitian ini bertujuan untuk menerapkan algoritma *K-Nearest Neighbors* (KNN) dalam memprediksi kualitas air minum berdasarkan *dataset Water Quality* dari *Kaggle*. Dataset mencakup 3.276 data dengan 9 parameter, seperti pH, kekerasan, dan kandungan karbon organik, serta satu atribut target yang menunjukkan kelayakan konsumsi. Penelitian ini akan menerapkan algoritma KNN dengan berbagai nilai (*k*), dan mengevaluasi kinerja model menggunakan metrik akurasi dan *Jaccard Similarity*. Hasil penelitian menunjukkan bahwa algoritma KNN dalam memprediksi kualitas air minum mencapai akurasi terbaik sebesar 58% pada nilai (*k*) = 2, hasil ini menunjukkan bahwa metode ini cukup baik meskipun perlu pengembangan lebih lanjut dengan metode lain untuk meningkatkan akurasi. Penelitian ini memberikan kontribusi pada implementasi teknologi pembelajaran mesin dalam pengelolaan sumber daya air.

Kata Kunci — *K-Nearest Neighbors*, prediksi kualitas air, normalisasi data, akurasi model, *Jaccard Similarity*

1. PENDAHULUAN

Kebutuhan akan air minum yang berkualitas semakin meningkat seiring dengan pertumbuhan populasi dan urbanisasi, sementara ketersediaannya semakin terancam oleh pencemaran lingkungan.

Kualitas air yang buruk dapat berdampak serius terhadap kesehatan manusia, menyebabkan berbagai penyakit seperti diare, kolera, dan tifus. Oleh karena itu, pemantauan kualitas air secara efektif menjadi sangat krusial untuk memastikan ketersediaan air layak konsumsi. Metode konvensional dalam pengujian kualitas air sering kali memerlukan waktu dan biaya yang besar, sehingga diperlukan pendekatan yang lebih efisien dan berbasis data. Dalam hal ini, penerapan algoritma *K-Nearest Neighbors* (KNN) menawarkan solusi berbasis pembelajaran mesin untuk memprediksi kualitas air dengan lebih cepat dan akurat. KNN bekerja dengan membandingkan parameter fisik dan kimia air, seperti *pH*, kadar karbon organik, dan tingkat kekeruhan, dengan data historis untuk menentukan kelayakan air minum. Dengan pendekatan ini, sistem pemantauan kualitas air dapat dikembangkan menjadi lebih otomatis, efisien, dan akurat, membantu pengelola sumber daya air serta masyarakat dalam mengambil keputusan yang lebih tepat terhadap konsumsi air bersih. Menurut laporan Organisasi Kesehatan Dunia (WHO) pada tahun 2021, konsumsi air yang terkontaminasi menjadi penyebab utama dari berbagai penyakit menular seperti diare, kolera, dan tifus, yang mempengaruhi jutaan orang setiap tahunnya, terutama di negara berkembang [1]. Oleh karena itu, pemantauan kualitas air secara efektif menjadi hal yang sangat krusial untuk memastikan ketersediaan air layak konsumsi. Dalam dua dekade terakhir, kemajuan teknologi telah memungkinkan diterapkannya metode berbasis data untuk memantau dan memprediksi kualitas air. Salah satu pendekatan yang berkembang pesat adalah penggunaan pembelajaran mesin (*machine learning*). Dengan memanfaatkan pembelajaran mesin, prediksi kualitas air dapat dilakukan secara lebih cepat dan akurat dibandingkan metode konvensional. Selain itu, pendekatan ini mampu mengolah data dalam jumlah besar dan menghasilkan analisis yang lebih mendalam terhadap berbagai parameter yang memengaruhi kualitas air [2].

Algoritma *K-Nearest Neighbors* (KNN) merupakan salah satu algoritma pembelajaran mesin yang populer karena kesederhanaannya dan kemampuan adaptasi yang baik terhadap berbagai jenis dataset. KNN bekerja berdasarkan prinsip pencarian jarak terdekat antara data baru dengan data yang telah ada untuk menentukan kelas atau nilai prediksi. Dalam konteks prediksi kualitas air, KNN dapat digunakan untuk mengklasifikasikan apakah air layak konsumsi atau tidak berdasarkan parameter fisik dan kimia seperti pH, kekerasan, kandungan karbon organik, dan padatan terlarut [3]. Salah satu keunggulan KNN adalah kemampuannya untuk beradaptasi dengan baik terhadap *dataset* kecil hingga sedang, terutama jika fitur dalam *dataset* tersebut telah dinormalisasi. Namun, tantangan utama dari algoritma ini adalah efisiensinya yang menurun ketika digunakan pada *dataset* besar atau dengan jumlah fitur yang sangat banyak. Hal ini disebabkan oleh sifat algoritma KNN yang berbasis jarak, sehingga memerlukan komputasi yang tinggi untuk setiap prediksi yang dilakukan. Oleh karena itu, pemilihan parameter seperti nilai (*k*) dan normalisasi data menjadi aspek penting dalam implementasi algoritma ini [4], [5].

Penelitian ini bertujuan untuk mengaplikasikan algoritma KNN dalam memprediksi kualitas air minum menggunakan *dataset Water Quality* yang diperoleh dari *platform Kaggle*. *Dataset* ini mencakup 3.276 data dengan sembilan parameter masukan yang meliputi pH, kekerasan air, padatan terlarut, karbon organik, serta satu atribut target yang menunjukkan potabilitas air, yaitu apakah air tersebut layak minum atau tidak. Analisis dilakukan melalui beberapa tahapan penting, mulai dari *preprocessing* data, normalisasi, pembagian data menjadi pelatihan dan pengujian, hingga evaluasi kinerja model menggunakan metrik akurasi dan *Jaccard Similarity* [6], [7].

Pada tahap *preprocessing*, data dinormalisasi menggunakan metode *Z-Score Normalization* untuk memastikan bahwa semua parameter memiliki skala yang seragam. Normalisasi data penting dalam algoritma berbasis jarak seperti KNN karena fitur dengan rentang nilai yang besar dapat mendominasi perhitungan jarak jika tidak dilakukan normalisasi. Selain itu, *dataset* dibagi menjadi dua subset utama, yaitu data pelatihan (80%) dan data pengujian (20%). Pembagian data ini bertujuan untuk melatih model dengan sebagian besar data yang tersedia dan menguji performa model pada data yang tidak terlihat sebelumnya [8], [9]. Algoritma KNN kemudian diterapkan dengan berbagai nilai (*k*) untuk menentukan jumlah tetangga terdekat yang optimal dalam prediksi kualitas air. Proses evaluasi dilakukan untuk mengukur performa model dalam memprediksi potabilitas air berdasarkan parameter fisik dan kimia. Dalam penelitian ini, metrik akurasi digunakan untuk mengevaluasi seberapa baik model memprediksi data dengan benar, sedangkan *Jaccard Similarity* digunakan untuk mengukur kesamaan antara prediksi model dan data aktual.

Studi ini diharapkan dapat memberikan kontribusi signifikan pada pengelolaan sumber daya air, khususnya dalam mendukung pengambilan keputusan berbasis data untuk memastikan kualitas air yang aman dan layak konsumsi. Dengan penerapan teknologi pembelajaran mesin seperti KNN, sistem

pemantauan kualitas air dapat dikembangkan menjadi lebih efisien, *real-time*, dan dapat diakses oleh berbagai pihak, mulai dari pengelola lingkungan hingga pembuat kebijakan.

2. METODE PENELITIAN

Algoritma *K-Nearest Neighbors* (KNN) adalah salah satu metode pembelajaran mesin yang digunakan untuk klasifikasi dan regresi. Prinsip dasar dari KNN adalah mencari (*k*) tetangga terdekat (*nearest neighbors*) dari data baru yang ingin diprediksi, dan kemudian menggunakan informasi tersebut untuk menentukan kelas atau nilai yang sesuai. KNN sering digunakan dalam berbagai aplikasi seperti pengenalan pola, sistem rekomendasi, dan analisis data berbasis klaster. KNN termasuk metode *supervised learning* (pembelajaran terawasi). Dalam *supervised learning*, model dilatih menggunakan data yang sudah diberi label atau target yang diketahui, dan tujuan model adalah untuk memprediksi label atau target tersebut pada data baru yang belum diketahui. KNN bekerja dengan sangat baik untuk *dataset* kecil dan sedang, tetapi menjadi kurang efisien saat digunakan untuk *dataset* besar atau dengan banyak fitur.

Pada penelitian ini menggunakan *dataset Water Quality* yang diperoleh dari situs *kaggle.com*, yang merupakan situs yang menyediakan beragam dataset yang bisa digunakan secara gratis [10]. Metode penelitian pada penerapan Algoritma KNN pada prediksi Kualitas Air Minum terdiri dari:

a. Persiapan Data: Mengumpulkan dan memproses data (normalisasi, penanganan nilai hilang).
b. Data *Pre-Processing*: Proses memisahkan data menjadi data training dan *testing*.
c. Penerapan Algoritma KNN: Menentukan nilai (*k*).
d. Evaluasi Model: Menilai kinerja model menggunakan metrik yang sesuai.

2.1. Persiapan Data

Pada penelitian ini menggunakan *dataset Water Quality* yang memuat 3.276 data dengan 9 atribut *input* dan 1 atribut *output* atau target. Target yang dimaksud menunjukkan air minum layak diminum bernilai "1" dan tidak layak minum bernilai "0". Deskripsi dari masing-masing fitur dijelaskan pada Tabel 1.

Tabel 1. Deskripsi Fitur

No	Atribut	Deskripsi
1	*ph*	*pH* air (0 hingga 14).
2	*Hardness*	Kapasitas air untuk mengendapkan sabun dalam *mg/L*.
3	*Solids*	Total padatan terlarut dalam *ppm*.
4	*Chloramines*	Jumlah Kloramina dalam *ppm*.
5	*Sulfate*	Jumlah Sulfat yang terlarut dalam *mg/L*.
6	*Conductivity*	Konduktivitas listrik air dalam $\mu S/cm$.
7	*Organic_carbon*	Jumlah karbon organik dalam *ppm*.
8	*Trihalomethanes*	Jumlah Trihalometana dalam $\mu g/L$.
9	*Turbidity*	Ukuran sifat air yang memancarkan cahaya dalam NTU.
10	*Potability*	Menunjukkan apakah air aman untuk dikonsumsi manusia.

2.2. Data Pre-Processing

Pada data *Pre-Processing* dilakukan tahap Normalisasi Data dan Pembagian Data.

2.2.1. Normalisasi Data

Normalisasi data adalah proses yang penting untuk memastikan bahwa fitur dalam *dataset* memiliki skala yang seragam. Ini membantu algoritma yang bergantung pada pengukuran jarak (seperti KNN) bekerja dengan baik, serta meningkatkan performa model secara keseluruhan. Teknik umum normalisasi termasuk *Min-Max Scaling* dan *Z-Score Normalization*. Teknik *Z-Score Normalization* dipilih karena memastikan bahwa semua fitur memiliki kontribusi yang seimbang dalam perhitungan jarak KNN, lebih tahan terhadap *outlier* dan lebih stabil dalam menangani variasi skala parameter air minum, dengan Persamaan 1 sebagai berikut:

 doi: https://doi.org/10.32503/jtecs.v5i1.6715

$$z = \frac{X_i - \mu}{\sigma} \quad \text{..... (1)}$$

Keterangan,
z = Nilai data ke-*i* yang akan dinormalisasikan
X_i = Nilai rata-rata
σ = Standard deviasi

2.2.2. Pembagian Data

Pembagian *dataset* adalah proses membagi data yang tersedia menjadi beberapa subset yang digunakan untuk tujuan yang berbeda dalam pelatihan dan evaluasi model pembelajaran mesin. Pembagian *dataset* yang baik penting untuk memastikan bahwa model yang dibangun tidak *overfitting* (terlalu sesuai dengan data pelatihan) dan dapat menggeneralisasi dengan baik pada data baru yang tidak terlihat sebelumnya. Metode ini adalah cara pembagian *dataset* yang paling sederhana, dimana *dataset* dibagi menjadi dua bagian. Data Pelatihan (*Training Set*) yang digunakan untuk melatih model. Pemilihan rasio 80:20 dalam pembagian *dataset* menjadi 80% data pelatihan dan 20% data pengujian didasarkan pada keseimbangan yang optimal antara pelatihan model dan evaluasi performa. Rasio ini sudah menjadi praktik umum dalam pembelajaran mesin karena memberikan jumlah data pelatihan yang cukup agar model dapat mempelajari pola-pola penting, sementara tetap menyediakan sampel yang memadai untuk menguji akurasi model secara independen. Dalam *dataset* dengan 3.276 sampel, rasio ini memastikan bahwa model dapat belajar dengan baik tanpa *overfitting*, sekaligus memberikan evaluasi yang representatif. Rasio 80:20 juga banyak digunakan dalam literatur dan penelitian terkait, seperti yang dijelaskan oleh *Hastie et al.* dan *Gholamy et al.* [3], yang merekomendasikan pembagian ini sebagai titik keseimbangan antara bias dan varians, serta memberikan performa yang stabil pada model pembelajaran mesin.

2.3. Penerapan Algoritma KNN

Berikut akan dijelaskan tahapan dari penerapan Algoritma KNN:

a. Menentukan nilai (*k*)
b. Mencari jarak dari data uji ke setiap data latih, yaitu dengan perhitungan *Eucledian Distance*, Persamaan 2.

$$Eucledian = \sqrt{\sum_{i=2}^{n}(p_i - q_i)^2} \quad \text{..... (2)}$$

Keterangan,
p_i = Data *Training*
q_i = Data *Testing*

c. Menentukan tetangga terdekat berdasarkan jarak terdekat ke (*k*) dengan mengurutkan data yang sudah dihitung jaraknya
d. Periksa kelas dengan jarak terdekat
e. Menentukan nilai prediksi untuk data yang baru dari mayoritas sederhana kelas tetangga terdekat

2.4. Evaluasi Model

Setelah melalui tahap pelatihan, model dievaluasi untuk mengukur performa prediksi dengan menggunakan metrik akurasi dan *Jaccard Similarity*. Akurasi digunakan untuk menilai seberapa banyak prediksi model yang sesuai dengan data aktual secara keseluruhan, memberikan gambaran umum mengenai performa model. Di sisi lain, *Jaccard Similarity* digunakan untuk mengevaluasi tingkat kesesuaian prediksi pada klasifikasi biner, terutama dalam membandingkan kemiripan antara himpunan data prediksi dan data aktual. Metrik ini menghitung rasio antara *intersection* dan *union* dari dua himpunan, sehingga memberikan wawasan tambahan tentang kemampuan model dalam mengenali kelas target yang relevan. Penggunaan *Jaccard* penting pada kasus klasifikasi biner, karena lebih sensitif

terhadap ketidakseimbangan data dan dapat membantu mengidentifikasi bagaimana model menangani elemen-elemen penting seperti *true positives* dalam menentukan potabilitas air. Dengan menggabungkan kedua metrik tersebut, analisis evaluasi model menjadi lebih komprehensif. *Jaccard Similarity* antara A dan B ditunjukkan pada persamaan 3:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \quad \text{.....(3)}$$

Keterangan,
A = Himpunan A
B = Himpunan B
$\cap$ = *Intersection*
$\cup$ = *Union*

3. HASIL DAN PEMBAHASAN

3.1. Persiapan Data

Dataset Water Quality yang berjenis .csv diimport menggunakan IDE *Jupyter Notebook* dengan 3.276 data, 9 atribut *input* dan 1 atribut *output*. Gambar 1 menunjukkan cuplikan *dataset Water Quality* yang digunakan dalam penelitian, terdiri dari 3.276 baris dan 10 kolom. Kolom-kolomnya mencakup parameter fisik dan kimia air seperti *ph* (tingkat keasaman), *Hardness* (kapasitas mengendapkan sabun), *Solids* (total padatan terlarut), *Chloramines* (kloramina), *Sulfate* (sulfat), *Conductivity* (konduktivitas listrik), *Organic_carbon* (karbon organik), *Trihalomethanes* (trihalometana), dan *Turbidity* (kekeruhan), serta kolom target *Potability* yang menunjukkan kelayakan air minum (0: tidak layak, 1: layak). Beberapa kolom memiliki nilai kosong (*NaN*), seperti *ph*, *Sulfate*, dan *Trihalomethanes*, sehingga memerlukan langkah data *preprocessing* seperti imputasi atau penghapusan data. *Dataset* ini mencakup data numerik dengan skala berbeda, yang memerlukan normalisasi agar algoritma berbasis jarak seperti KNN dapat bekerja secara optimal.

	ph	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity	Potability
0	NaN	204.890455	20791.318981	7.300212	368.516441	564.308654	10.379783	86.990970	2.963135	0
1	3.716080	129.422921	18630.057858	6.635246	NaN	592.885359	15.180013	56.329076	4.500656	0
2	8.099124	224.236259	19909.541732	9.275884	NaN	418.606213	16.868637	66.420093	3.055934	0
3	8.316766	214.373394	22018.417441	8.059332	356.886136	363.266516	18.436524	100.341674	4.628771	0
4	9.092223	181.101509	17978.986339	6.546600	310.135738	398.410813	11.558279	31.997993	4.075075	0
...	...	...	...	...	...	...	...	...	...	...
3271	4.668102	193.681735	47580.991603	7.166639	359.948574	526.424171	13.894419	66.687695	4.435821	1
3272	7.808856	193.553212	17329.802160	8.061362	NaN	392.449580	19.903225	NaN	2.798243	1
3273	9.419510	175.762646	33155.578218	7.350233	NaN	432.044783	11.039070	69.845400	3.298875	1
3274	5.126763	230.603758	11983.869376	6.303357	NaN	402.883113	11.168946	77.488213	4.708658	1
3275	7.874671	195.102299	17404.177061	7.509306	NaN	327.459760	16.140368	78.698446	2.309149	1

3276 rows × 10 columns

Gambar 1. Tampilan *Dataset Water Quality* setelah diimpor pada *Jupyter Notebook*

Gambar 2 menunjukkan struktur *dataset Water Quality* dengan total 3.276 baris dan 10 kolom. Setiap kolom dijelaskan dengan jumlah data *non-null*.

 doi: https://doi.org/10.32503/jtecs.v5i1.6715

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3276 entries, 0 to 3275
Data columns (total 10 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   ph               2785 non-null   float64
 1   Hardness         3276 non-null   float64
 2   Solids           3276 non-null   float64
 3   Chloramines      3276 non-null   float64
 4   Sulfate          2495 non-null   float64
 5   Conductivity     3276 non-null   float64
 6   Organic_carbon   3276 non-null   float64
 7   Trihalomethanes  3114 non-null   float64
 8   Turbidity        3276 non-null   float64
 9   Potability       3276 non-null   int64
dtypes: float64(9), int64(1)
memory usage: 256.1 KB
```

Gambar 2. Tipe *Dataset*

Gambar 3 menunjukkan bahwa beberapa kolom memiliki data hilang, seperti *pH* (491 data hilang), *Sulfate* (781 data hilang), dan *Trihalomethanes* (162 data hilang), sementara kolom lainnya, seperti *Hardness*, *Solids*, dan *Potability*, memiliki data lengkap. Sebagian besar kolom bertipe data *float64* (numerik desimal), kecuali *Potability*, yang bertipe *int64* (bilangan bulat kategori biner).

```
ph               491
Hardness           0
Solids             0
Chloramines        0
Sulfate          781
Conductivity       0
Organic_carbon     0
Trihalomethanes  162
Turbidity          0
Potability         0
dtype: int64
```

Gambar 3. Data Kosong

3.2. Data Pre-Processing

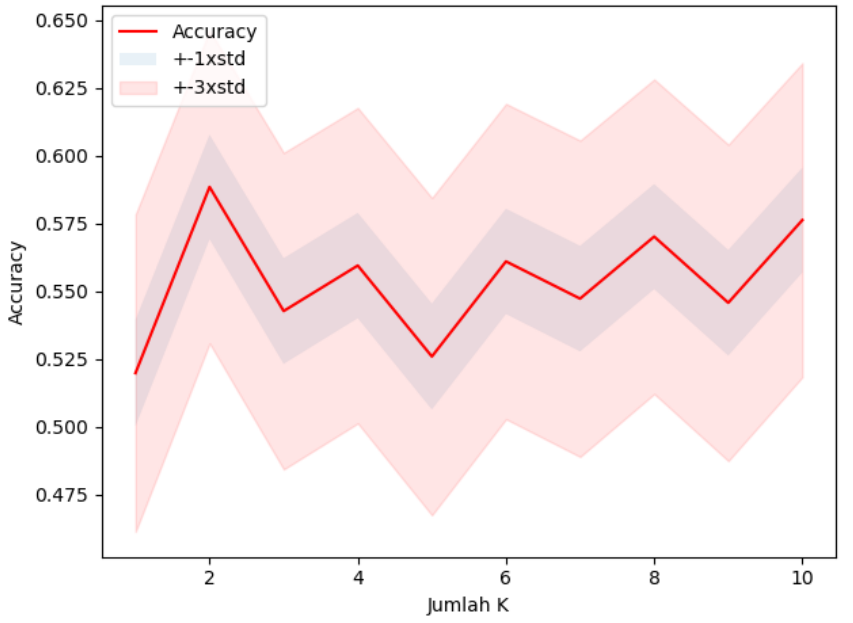
Pada tahap Data *Pre-Processing* dilakukan proses Normalisasi untuk mengatasi fitur yang terdapat data kosong sebelum proses *training* model dilakukan. Selanjutnya *dataset* dibagi menjadi data *Training* dan data *Testing*. Pada Gambar 4 menunjukkan hasil normalisasi data kosong menggunakan metode *KNNImputer* dari *library scikit-learn*. Metode ini merupakan pendekatan imputasi berbasis algoritma KNN, yang menggantikan nilai yang hilang dengan rata-rata nilai dari sejumlah tetangga terdekat (*k*) yang ditentukan. Pendekatan ini dipilih karena kemampuannya untuk mempertimbangkan hubungan antar fitur dalam *dataset*, sehingga menghasilkan imputasi yang lebih kontekstual dibandingkan metode sederhana seperti imputasi dengan rata-rata atau *median*.

	ph	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity	Potability
0	6.655223	204.890455	20791.318981	7.300212	368.516441	564.308654	10.379783	86.990970	2.963135	0.0
1	3.716080	129.422921	18630.057858	6.635246	351.285226	592.885359	15.180013	56.329076	4.500656	0.0
2	8.099124	224.236259	19909.541732	9.275884	347.323743	418.606213	16.868637	66.420093	3.055934	0.0
3	8.316766	214.373394	22018.417441	8.059332	356.886136	363.266516	18.436524	100.341674	4.628771	0.0
4	9.092223	181.101509	17978.986339	6.546600	310.135738	398.410813	11.558279	31.997993	4.075075	0.0
...	...	...	...	...	...	...	...	...	...	...
3271	4.668102	193.681735	47580.991603	7.166639	359.948574	526.424171	13.894419	66.687695	4.435821	1.0
3272	7.808856	193.553212	17329.802160	8.061362	368.086095	392.449580	19.903225	56.689055	2.798243	1.0
3273	9.419510	175.762646	33155.578218	7.350233	316.571962	432.044783	11.039070	69.845400	3.298875	1.0
3274	5.126763	230.603758	11983.869376	6.303357	334.293598	402.883113	11.168946	77.488213	4.708658	1.0
3275	7.874671	195.102299	17404.177061	7.509306	333.330087	327.459760	16.140368	78.698446	2.309149	1.0

Gambar 4. *Dataset* Setelah Normalisasi

3.3. Penerapan Algoritma KNN

Pada Gambar 5 menunjukkan hubungan antara jumlah tetangga terdekat (*k*) dalam algoritma *K-Nearest Neighbors* (KNN) dengan akurasi model, di mana garis merah merepresentasikan tren akurasi, sedangkan area berwarna menunjukkan rentang standar deviasi (STD) dengan interval ±1 STD (abu-abu) dan ±3 STD (merah muda). Nilai (*k*) yang digunakan bervariasi mulai dari nilai 2, 4, 6, 8, dan 10.



Gambar 5. Nilai Akurasi Terhadap Jumlah (*k*)

3.4. Evaluasi Model

Pada tahap evaluasi model setelah dilakukan ujicoba terhadap *dataset*. Maka dapat diperoleh nilai (*k*) = 2 dengan akurasi terbaik model 58%. Pada penilaian menggunakan *Jaccard Score* mendapatkan nilai 33%.

 doi: https://doi.org/10.32503/jtecs.v5i1.6715

4. KESIMPULAN

Kesimpulan pada penelitian ini metode KNN dapat mengklasifikasikan dengan baik sebesar 58% dengan nilai (*k*) = 2. Akurasi model tampak berfluktuasi, dengan nilai tertinggi adalah 58% pada (*k*) = 2, kemudian mengalami penurunan sebelum kembali meningkat secara tidak stabil pada beberapa nilai (*k*) lainnya. Variasi akurasi yang cukup besar di beberapa titik menunjukkan bahwa model sensitif terhadap pemilihan (*k*), dimana nilai (*k*) yang terlalu kecil dapat menyebabkan *overfitting*, sementara (*k*) yang terlalu besar dapat menyebabkan *underfitting*. Dengan demikian, pemilihan (*k*) yang optimal sangat penting untuk menyeimbangkan bias dan varians model, dan dari hasil ini, (*k*) = 2 tampaknya memberikan performa terbaik meskipun masih terdapat fluktuasi. Untuk meningkatkan stabilitas model, teknik *cross-validation* dapat digunakan guna mengurangi variabilitas dalam performa prediksi.

DAFTAR PUSTAKA

[1] F. C. Limuris, "Hak Rakyat Atas Air Bersih Sebagai Derivasi Hak Asasi Manusia Dalam Deklarasi Universal Hak Asasi Manusia", JURNAL JENTERA, Volume 4, No. 2 Desember 2021.

[2] M. Y. Shams, "Water quality prediction using machine learning models based on grid search method", Multimedia Tools and Applications, 2024.

[3] Nurmahaludin, "Klasifikasi Kualitas Air PDAM Menggunakan Algoritma KNN dan K-Means", Seminar Nasional Riset Terapan, 2019.

[4] M. P. Firdaus, "Perbandingan Algoritma KNN dan NBC dengan Ekstraksi Fitur TF-IDF dan N-Gram", UIN Syarif Hidayatullah, 2023.

[5] P. A. Riyantoko, "Analisis Sederhana Pada Kualitas Air Minum Berdasarkan Akurasi Model Klasifikasi Dengan Menggunakan Lucifer Machine Learning", Seminar Nasional Sains Data, 2021.

[6] M. A. Rahman, "Komparasi Metode Data Mining K-Nearest Neighbor Dengan Naïve Bayes Untuk Klasifikasi Kualitas Air Bersih (Studi Kasus PDAM Tirta Kencana Kabupaten Jombang)", Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, Vol. 2, No. 12, 2018.

[7] R. Hutami, "Implementasi Metode K-Nearest Neighbor Untuk Prediksi Penjualan Furniture Pada CV.Octo Agung Jepara", Universitas Dian Nuswantoro Semarang, 2016.

[8] C. A. Rahardja, "Implementasi Algoritma K-Nearest Neighbor Pada Website Rekomendasi Laptop", J. Buana Inform, 2019.

[9] Y. Yahya, "Penerapan Algoritma K-Nearest Neighbor Untuk Klasifikasi Efektivitas Penjualan Vape (Rokok Elektrik) pada 'Lombok Vape On", Infotek J. Inform. dan Teknol., vol. 3, no. 2, pp. 104–114, 2020.

[10] https://www.kaggle.com/datasets/adityakadiwal/water-potability diakses pada tanggal 10 Januari 2025.