ALGORITMA XGBOOST UNTUK KLASIFIKASI KUALITAS AIR MINUM

Muhammad Dava Maulana, Asep Id Hadiana, Fajri Rakhmat Umbara
Informatika, Universitas Jenderal Achmad Yani Cimahi
Jl. Terusan Jend. Sudirman, Cibeber, Kec. Cimahi Sel., Kota Cimahi, Jawa Barat 40531
muhammad.dava@student.unjani.ac.id

ABSTRAK

Air adalah sumber daya alam yang penting bagi keberlangsungan hidup organisme di bumi, termasuk manusia, hewan, dan tumbuhan. Namun, potensi tercemarnya air oleh bakteri dan zat mineral berbahaya mengancam kesehatan manusia. Penyebab utama tercemarnya air meliputi aktivitas manusia yang intens, proses industri kompleks, dan kurangnya kesadaran akan dampak limbah yang dibuang sembarangan ke perairan. Kualitas air bisa diukur dengan berbagai parameter, dan pendekatan machine learning dapat digunakan untuk klasifikasi kualitas air. Salah satu metode yang efektif adalah XGBoost, sebuah varian gradient tree boosting yang mampu meningkatkan akurasi prediksi dengan mengurangi overfitting. Penelitian sebelumnya telah menunjukkan bahwa XGBoost memiliki performa yang baik dalam klasifikasi, bahkan mengungguli beberapa algoritma lain seperti Naïve Bayes dan Random Forest dalam konteks tertentu. Penelitian ini juga bertujuan untuk mengukur akurasi algoritma XGBoost dalam melakukan klasifikasi kualitas air. Hasil percobaan menunjukan dengan dataset sebanyak 2400 dengan pembagian data latih 80% dan data uji 20%, yang terdiri dari 1200 data kelas dapat diminum dan 1200 data kelas tidak dapat diminum. Akurasi yang didapat adalah 82.29%, precision 78.62%, recall 85.90% dan f1-score 82.09%.

Kata kunci: *Air, Data Mining, Klasifikasi, XGBoost, Confusion Matrix*

1. PENDAHULUAN

Air adalah sumber daya alam yang sangat penting serta diperlukan untuk aktivitas dan untuk menopang keberlangsungan hidup organisme, termasuk manusia, hewan, dan tumbuhan di bumi[1]. Tetapi tidak semua air aman untuk dikonsumsi karena air bisa tercemar oleh bakteri serta zat mineral yang merugikan bagi kesehatan manusia[2]. Hal ini bisa terjadi sebab sumber air atau lingkungan disekitarnya tercemar, sehingga menyebabkan kualitas air menurun[2], [3]. Faktor yang mengakibatkan air tercemar ialah aktivitas manusia yang semakin intens, proses industri yang semakin rumit, pembuangan limbah secara tidak benar ke sungai, dan kurangnya kesadaran dari masyarakat akan bahaya dari pembuangan limbah sembarangan ke air sungai, sehingga air tidak dapat dikonsumsi lagi[4], [5].

Untuk mengetahui kualitas air biasanya digambarkan dalam bentuk parameter dan variabel, berbagai jenis parameter digunakan sebagai dasar untuk membangun sebuah model dalam machine learning[3]. Dalam machine learning terdapat teknik klasifikasi yang digunakan untuk membuat prediksi dan mengelompokkan data ke dalam kelas-kelas tertentu berdasarkan karakteristik atau ciri-ciri yang sama[6]. Terdapat banyak metode klasifikasi dalam machine learning salah satunya yaitu *XGBoost*.

XGBoost merupakan singkatan dari "*Extreme Gradient Boosting*" yang merupakan varian dari *gradient tree boosting*[7]. *Gradient tree boosting* adalah metode peningkatan pohon yang menggabungkan sekumpulan pengklasifikasian lemah untuk membuat pengklasifikasian yang kuat[8]. *XGBoost* meningkatkan performa dengan mengatur kompleksitas pohon dengan menggunakan metode regularisasi yang berbeda dan memiliki kinerja yang sangat baik dalam memprediksi data serta memecahkan masalah klasifikasi karena *XGBoost* ini dapat membantu meminimalkan *overfitting* dan membuat model lebih akurat[8].

Berdasarkan penelitian terdahulu, klasifikasi dilakukan dengan membandingkan akurasi antara algoritma *XGBoost* dan naïve bayes pada kasus penyakit diabetes, algoritma *XGBoost* mendapatkan hasil akurasi tinggi dengan akurasi 90,10% dan metode *Naïve Bayes* 79,68%[9]. Penelitian selanjutnya tentang klasifikasi menggunakan *Random Forest* dan *XGBoost* pada kasus prediksi hujan mendapatkan akurasi *Random Forest* 89.54% dan Untuk *XGBoost* 88,96%[10]. Hal tersebut menunjukkan algoritma *XGBoost* menunjukkan performa yang cukup baik dalam melakukan klasifikasi.

Penelitian selanjutnya yaitu dengan menerapkan proses *resampling* data dengan metode *oversampling* untuk menyeimbangkan kelas mendapatkan hasil kinerja algoritma yang meningkat, salah satunya Algoritma *Decision Tree* yang meningkat 22% pada metrik akurasi[11].

Oleh karena itu berdasarkan pemaparan diatas, maka penelitian ini akan menetapkan beberapa pendekatan, dengan melakukan metode *oversampling* dan menetapkan algoritma *XGBoost* untuk melakukan klasifikasi kualitas air serta mengetahui akurasi dari algoritma *XGBoost*.

2. TINJAUAN PUSTAKA

2.1. *Data Mining*

Data mining merupakan proses mengumpulkan, mengelola, dan menganalisis data besar baik yang terstruktur juga yang tidak terstruktur[12]. Konsep dari

data mining ialah proses mengekstraksi pengetahuan dari himpunan data yang besar menggunakan algoritma atau rumus yg tersedia. Pengetahuan yg dihasilkan dari proses data mining dapat digunakan untuk tujuan tertentu. Data mining sangat bermanfaat dalam berbagai bidang seperti pemasaran, penjualan, penemuan ilmiah, pengembangan produk, dan prediksi data untuk mengidentifikasi hubungan yang tersembunyi dan tidak terduga di antara data[13], [14].

2.2. Klasifikasi

Klasifikasi adalah proses pengelompokan suatu obyek kedalam kategori atau kelas yang telah ditentukan sebelumnya[15]. Klasifikasi artinya proses pembuatan model yg mengelompokkan objek berdasarkan atribut-atribut yg dimilikinya[16]. Proses klasifikasi data atau dokumen bisa dimulai dengan membuat aturan klasifikasi yang menggunakan data latih (*training data*), dan pengujian menggunakan data uji (*testing data*)[6].

2.3. *XGBoost*

XGBoost adalah sebuah algoritma yang ditingkatkan dari gradient boosting decision tree yang efisien dalam membangun boosted trees[17]. XGBoost adalah metode pembelajaran mesin yang digunakan untuk menyelesaikan permasalahan regresi dan klasifikasi dengan menggunakan *Gradient Boosting Decision Tree* (GBDT)[18]. *XGBoost* termasuk salah satu teknik boosting yang terdiri dari beberapa *decision tree*, dimana setiap pohon diperkuat oleh pohon sebelumnya dan pohon berikutnya yang saling tergantung satu sama lain[19]. Ketika melakukan klasifikasi, *XGBoost* akan memperbarui bobot pada setiap pohon yg dibangun sehingga diperoleh pohon klasifikasi yg kuat[8].

2.4. Confusion Matrix

Confusion matrix adalah metode yang dipakai untuk mengukur performa model klasifikasi dengan membandingkan prediksi model terhadap label yang sebenarnya. Matriks akan mendeskripsikan jumlah prediksi yang benar dan salah yang dibuat oleh model untuk setiap kelas. Confusion matrix terdiri dari empat komponen utama, yaitu True Positive (TP), True Negative (TN), False Positive (FP), dan False Negative (FN)[20].

- *True Positive* (TP): Menunjukkan jumlah nilai positif yang diklasifikasian sebagai air layak minum oleh model.
- *True Negative* (TN): Menunjukkan jumlah nilai negatif yang diklasifikasikan air tidak layak minum oleh model.
- *False Positive* (FP): Menunjukkan jumlah nilai negatif air tidak layak minum yang salah diklasifikasikan sebagai positif oleh model.
- *False Negative* (FN): Menunjukkan jumlah nilai positif air layak minum yang salah diklasifikasikan sebagai negatif oleh model.

Pengukuran performa model dalam penelitian ini terdiri dari Akurasi, Precission, Recall, dan F1-score.

- Akurasi untuk menggambarkan sejauh mana model yang dibuat dapat mengklasifikasikan dengan akurat. Untuk menghitung akurasi menggunakan persamaan berikut:

$$Akurasi = \frac{TP + TN}{TP + FP + TN + FN} x100 \quad (1)$$

- Precision untuk mengukur sejauh mana prediksi positif atau air dapat diminum yang diberikan oleh model adalah benar. Untuk menghitung precision menggunakan persamaan berikut:

$$Precision = \frac{TP}{TP + FP} x100 \quad (2)$$

- Recall untuk mengukur sejauh mana model mampu mengidentifikasi dan mengklasifikasikan dengan benar semua kasus positif atau air dapat diminum yang sebenarnya, dengan menggunakan persamaan berikut:

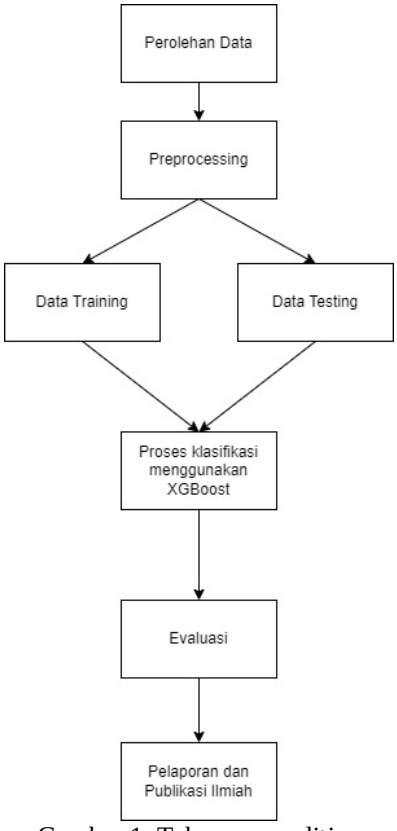
$$Recall = \frac{TP}{TP + FN} x100 \quad (3)$$

- F1-score untuk menggambarkan perbandingan rata-rata precision dan recall yang dibobotkan. Untuk menghitung f1-score menggunakan persamaan berikut:

$$F1\text{-}score = \frac{2 * Precision * Recall}{Precision + Recall} x100 \quad (4)$$

3. METODE PENELITIAN

Berikut gambaran tahapan penelitian yang dilakukan



Gambar 1. Tahapan penelitian

3.1. Perolehan Data

Pada tahap ini, peneliti melakukan perolehan data. Data diperoleh dari website kaggle.com "https://www.kaggle.com/datasets/adityakadiwal/water-potability". Data yang diperoleh dari website kaggle tersebut berformat CSV.

3.2. *Preprocessing*

Tahap selanjutnya yaitu preprocessing. Pada tahap ini peneliti melakukan proses pengolahan data untuk membuang data yang bernilai kosong/tidak lengkap (missing value) dan melakukan class balancing menggunakan teknik oversampling.

3.3. Pembagian Data

Tahap selanjutnya yaitu membagi data menjadi data training dan data testing. Pada tahap ini dilakukan pembagian menjadi data training untuk membentuk model/pola/knowledge dan data testing dilakukan untuk pengujian model.

3.4. Penerapan XGBoost

Tahap selanjutnya yaitu proses klasifikasi menggunakan *XGBoost*. Pada tahap ini melatih model menggunakan data latih.

3.5. Evaluasi

Tahap terakhir yaitu evaluasi. Pada tahap ini peneliti melakukan evaluasi hasil klasifikasi untuk mengukur tingkat akurasi dari algoritma XGBoost dengan menggunakan confusion matrix.

4. HASIL DAN PEMBAHASAN

Hasil dan pembahasan akan membahas isi dari tahapan yang dilakukan pada metode penelitian

4.1. Perolehan Data

Pada penelitian ini dataset diperoleh dari website kaggle.com"https://www.kaggle.com/datasets/adityakadiwal/water-potability". Data yang diperoleh dari website kaggle tersebut berformat CSV dan terdapat record data sebanyak 3276 baris dan terdapat 9 atribut dan 1 atribut labelling. Berikut adalah sampel data water potability yang dapat dilihat pada tabel 1.

Tabel 1. Dataset water potabilty

ph	Hardness	…	Turbidity	Potability
8.316766	214.373394	…	4.628771	0
9.092223	181.101509	…	4.075075	0
…	…	…	…	…
6.957434	214.380139	…	4.892050	1
6.509622	197.759503	…	3.802820	1

Dari Tabel 1 merupakan sampel dari dataset water potability

Tabel 2. Atribut dataset water potability

No	Atribut Data
1	ph
2	Hardness
3	Solids
4	Chloramines
5	Sulfate
6	Conductivity
7	Organic_carbon
8	Trihalomethanes
9	Turbidity
10	Potability

Dari Tabel 2 dapat dilihat bahwa atribut yang terdapat pada dataset water potability terdapat 10 atribut.

4.2. *Data Cleaning*

Pada tahap ini melakukan data cleaning untuk mengidentifikasi dan penanganan terhadap kesalahan, inkonsistensi, dan ketidakakuratan yang ada dalam dataset yang digunakan. Dalam penelitian ini proses data cleaning yang digunakan yaitu dengan cara penghapusan nilai yang hilang (*missing value*) pada dataset.

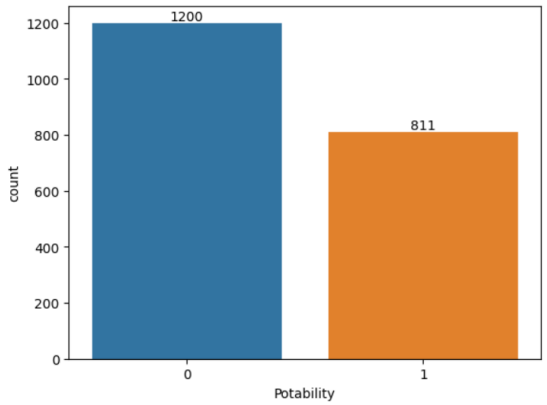
Tabel 3. Missing value pada dataset water potability

Atribut	Missing Value
ph	491
Hardness	0
Solids	0
Chloramines	0
Sulfate	781
Conductivity	0
Organic_carbon	0
Trihalomethanes	162
Turbidity	0
Potability	0

Dapat dilihat pada tabel 3 terdapat missing value pada atribut ph sebanyak 491, Sulfate 781, dan Trihalomethanes 162. Untuk mengatasi nilai yang hilang (missing value) yaitu dengan cara menghapus baris atau kolom yang mengandung nilai yang hilang tersebut. Setelah melakukan data cleaning jumlah record data menjadi 2011 baris.

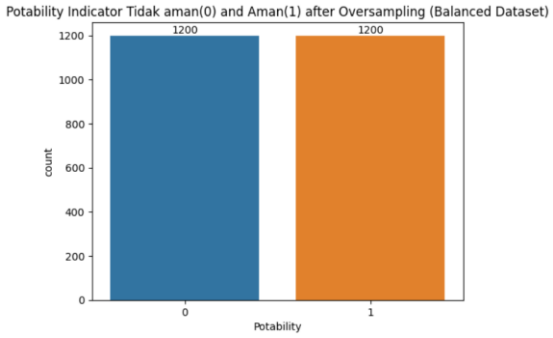
4.3. *Class Balancing*

Pada tahap ini melakukan class balancing dari dataset water potability yang telah melalui proses data cleaning. Tahapan ini untuk mengatasi ketidak seimbangan class yang dapat mempengaruhi kinerja model pembelajaran mesin, karena model cenderung akan memihak pada kelas mayoritas dibanding kelas minoritas karena yang memiliki jumlah sampel yang lebih banyak. Tahap class balancing ini menggunakan teknik oversampling untuk meng-duplikasi atau replikasi sampel dari kelas minoritas sehingga jumlah sampel pada kelas minoritas menjadi seimbang dengan kelas mayoritas.



Gambar 2. Kelas sebelum oversampling

Dari gambar 2 dapat dilihat class 0 (tidak dapat diminum) terdapat 1200 record data dan 1 (dapat diminum) terdapat 811 record data, sehingga terdapat unbalanced class pada kelas 0 = 59.67% dan kelas 1 = 40.33%.



Gambar 3. Kelas sesudah oversampling

Dari gambar 3 adalah hasil dari setelah penerapan class balancing dengan teknik oversampling. Class potability menjadi seimbang dengan record data menjadi 2400 data, class 0 memiliki 1200 record data dan class 1 memiliki 1200 data.

4.4. *Penerapan Model XGBoost*

Sebelum melakukan penerapan algoritma XGBoost, perlu melakukan pembagian data atau split data menjadi dua bagian, yaitu data latih dan data uji. Pada penelitian ini akan menggunakan split data dengan proporsi 80:20 digunakan untuk membagi data menjadi data latih sebesar 80% dan data uji sebesar 20%.

	ph	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity
3164	9.221956	151.454831	26502.595774	8.097194	294.192754	541.601438	13.951753	39.572407	4.294252
2030	7.830608	213.249258	18821.389612	6.475113	316.572415	405.278738	16.872395	55.905770	3.551568
351	8.848586	188.919983	32033.332019	13.127000	182.397370	479.791975	12.070444	77.671337	4.014682
2408	5.729689	173.975027	14304.763083	5.340088	369.153970	471.679901	16.424227	58.157688	5.100552
1531	6.692532	200.404351	37033.152091	4.727072	280.844916	376.672794	12.485257	57.818484	2.942115
...	...	...	...	...	...	...	...	...	...
2680	6.153496	163.205548	48175.852093	7.153803	299.566751	344.716976	5.159380	55.528949	4.819970
279	6.286807	258.300052	13777.376191	7.483258	328.680650	563.434775	16.460837	73.516654	4.104863
2053	3.846454	211.757205	14686.283390	7.557873	326.912134	371.269374	21.765170	35.761529	3.143431
2178	7.879543	170.190912	37000.955674	6.217223	346.063677	517.576762	15.871770	68.911185	4.066229
1110	4.736405	203.276420	28698.729965	5.271367	323.683683	508.820341	14.063140	65.155182	4.995602

1920 rows × 9 columns

Gambar 4. Data latih

	ph	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity
1105	8.275796	163.355961	22013.964905	5.742829	307.667123	351.631315	20.057619	78.501492	4.721340
561	6.624573	172.055471	14877.289737	7.079934	338.441277	405.818097	15.656149	58.560531	4.333721
1121	9.434006	158.387840	20474.820824	5.857333	285.395112	476.842011	14.388350	73.164778	4.201054
3148	8.076126	132.670305	10816.273367	9.513269	314.970495	479.758924	14.685851	72.638303	4.291355
973	4.443851	240.167901	24070.263651	9.768406	338.052437	533.968647	11.877582	69.036523	5.038119
...	...	...	...	...	...	...	...	...	...
1790	7.130099	275.679780	9480.617796	8.415948	295.618838	383.455068	18.322879	94.416301	1.986192
1058	7.775386	193.077168	15704.482093	7.881197	324.336203	301.753477	13.378165	89.051957	3.309472
2036	7.276863	165.797952	30396.904619	7.835639	281.896882	475.646051	12.326184	83.399889	6.083772
782	7.998090	241.000277	9609.740605	9.842346	229.575561	428.882366	14.990567	39.842379	2.626547
1544	6.777506	207.050748	15228.912909	6.182409	321.399404	523.241732	12.683724	86.165971	4.263180

480 rows × 9 columns

Gambar 5. Data uji

Pada penelitian ini menggunakan proporsi 80:20, karena penelitian sebelumnya telah melakukan perbandingan *split* data dan menemukan bahwa sebagian besar algoritma memberikan akurasi yang lebih baik ketika data dibagi dengan skema 80% untuk data latih dan 20% untuk data uji[11].

Lalu masuk ke tahapan tuning parameter, pada penelitian ini menggunakan metode pencarian grid untuk melakukan tuning parameter pada algoritma XGBoost.

Tabel 4. Grid Search Parameter Tuning

Parameter	***Grid Search Values***	***Nilai Parameter Terbaik***
n_estimators	100, 300, 500	500
max_depth	3, 4, 5, 8	8
learning_rate	0.05, 0.1, 0.01, 1	0.05

Dari tabel 4 merupakan parameter dan nilai parameter optimal yang dapat meningkatkan performa algoritma pada eksperimen yang dilakukan. Selanjutnya, konfigurasi parameter tersebut diterapkan pada proses pelatihan model XGBoost dengan menggunakan seluruh data latih.

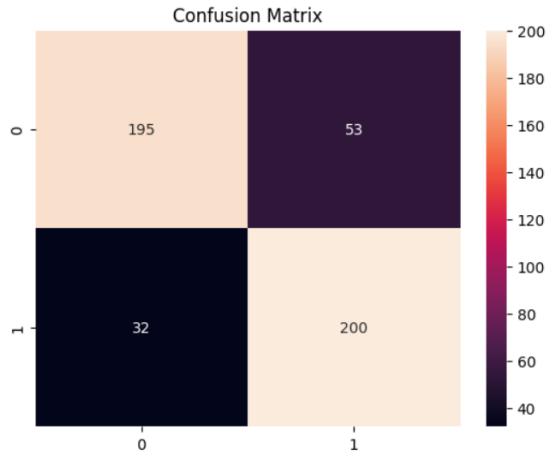
4.5. Evaluasi

Evaluasi penelitian adalah proses untuk menguji sejauh mana model yang digunakan memberikan hasil yang akurat. Untuk pengujian akurasi disini menggunakan confusion matrix. Penelitian ini menggunakan skema pengujian dengan proporsi data latih dan data uji sebesar 80%:20% dengan mengimplementasikan beberapa tahap preprocessing yaitu *data cleaning* dan *class balancing* menggunakan teknik oversampling, dan menggunakan parameter tuning. Berikut adalah nilai *confusion matrix* dari eksperimen yang dilakukan.

Tabel 5. Nilai confusion matrix

True Positive (TP)	**False Positive (FP)**	**False Negative (FN)**	**True Negative (TN)**
195	53	32	200

Dari tabel 5 adalah hasil dari confusion matrix dengan mendapatkan nilai True Positive sebesar 195, False Positive sebesar 53, False Negative sebesar 32, dan True Negative sebesar 200. Nilai-nilai tersebut dapat dihitung menggunakan persamaan 1, 2, 3, dan 4 untuk mendapatkan akurasi, precision, recall, dan f1-score.



Gambar 5. Confusion Matrix

Dari gambar 5 adalah hasil dari evaluasi yang dilakukan menggunakan confusion matrix.

Tabel 6. Matrix Evaluasi

Matrix Evaluasi	Skor
Akurasi	82.29%
Precision	78.62%
Recall	85.90%
F1-score	82.09%

Dari Tabel 6 dapat dilihat bahwa skor yang didapatkan dari perhitungan menggunakan persamaan 1, 2, 3, dan 4 mendapatkan hasil akurasi 82.29%, precision 78.62%, recall 85.90% dan f1-score 82.09%.

5. KESIMPULAN DAN SARAN

Berdasarkan penilitian yang telah dilakukan menggunakan algoritma XGBoost dengan menggunakan data kualitas air sebanyak 2400 record data. Dataset ini dibagi menjadi data latih dan data uji dengan perbandingan 80:20, di mana 80% digunakan sebagai data latih dan 20% sebagai data uji. Kemudian menerapkan beberapa tahap preprocessing yaitu data cleaning dan class balancing. Dan menerapkan teknik optimasi yaitu proses hyperparameter tuning menggunakan 3 hyperparameter dengan menggunakan metode pencarian grid untuk mencari nilai terbaik.

Dari hasil evaluasi menggunakan confusion matrix mendapatkan hasil dengan tingkat akurasi 82.29%, precision 78.62%, recall 85.90%, dan f1-score 82.09%. Jadi algoritma XGBoost dengan mengimplementasikan teknik oversampling dan parameter tuning merupakan metode klasifikasi yang cukup baik untuk diterapkan pada kasus klasifikasi kualitas air yang layak minum dan tidak layak minum.

DAFTAR PUSTAKA

[1] M. A. Rahman, N. Hidayat, and A. A. Supianto, "Komparasi Metode Data Mining K-Nearest Neighbor Dengan Naïve Bayes Untuk Klasifikasi Kualitas Air Bersih (Studi Kasus PDAM Tirta Kencana Kabupaten Jombang)," 2018. [Online]. Available: http://j-ptiik.ub.ac.id

[2] B. N. Azmi, A. Hermawan, and D. Avianto, "Jurnal Sistem dan Teknologi Informasi Analisis Pengaruh PCA Pada Klasifikasi Kualitas Air Menggunakan Algoritma K-Nearest Neighbor dan Logistic Regression," vol. 7, no. 2, 2022, [Online]. Available: http://jurnal.unmuhjember.ac.id/index.php/JUSTINDO

[3] P. A. Riyantoko, T. M. Fahrudin, K. M. Hindrayani, S. Data, and J. Timur, "Analisis Sederhana Pada Kualitas Air Minum Berdasarkan Akurasi Model Klasifikasi Dengan Menggunakan Lucifer Machine Learning," *Seminar Nasional Sains Data*, vol. 2021.

[4] A. Kustanto, "Water quality in Indonesia: The role of socioeconomic indicators," *Jurnal Ekonomi Pembangunan*, vol. 18, no. 1, pp. 47–62, Jul. 2020, doi: 10.29259/jep.v18i1.11509.

[5] R. Marten Sahalatua Tumangger and N. Hidayat, "Komparasi Metode Data Mining Support Vector Machine dengan Naive Bayes untuk Klasifikasi Status Kualitas Air," 2019. [Online]. Available: http://j-ptiik.ub.ac.id

[6] S. Raharjo and E. Winarko, "KLASTERISASI, KLASIFIKASI DAN PERINGKASAN TEKS BERBAHASA INDONESIA," *Universitas Gunadarma-Depok*, vol. 8, 2014.

[7] J. H. Friedman, "999 REITZ LECTURE GREEDY FUNCTION APPROXIMATION: A GRADIENT BOOSTING MACHINE 1," 2001.

[8] Z. Salam Patrous, "Evaluating XGBoost for User Classification by using Behavioral Features Extracted from Smartphone Sensors," 2018.

[9] M. K. Nasution, R. Rohmat Saedudin, and V. P. Widartha, "PERBANDINGAN AKURASI ALGORITMA NAÏVE BAYES DAN ALGORITMA XGBOOST PADA KLASIFIKASI PENYAKIT DIABETES," 2021.

[10] Ghaitsa Amany Mursianto, Isma'il Muhammad Falih, Muhammad Irfan3, Tiara Sakinah, and Desta Sandya Prasvita, "Perbandingan Metode Klasifikasi Random Forest dan XGBoost Serta Implementasi Teknik SMOTE pada Kasus Prediksi Hujan," 2021.

[11] G. L. Pritalia, "Analisis Komparatif Algoritme Machine Learning pada Klasifikasi Kualitas Air Layak Minum," 2022.

[12] J. Liu *et al.*, "Data Mining and Information Retrieval in the 21st century: A bibliographic review," *Computer Science Review*, vol. 34. Elsevier Ireland Ltd, Nov. 01, 2019. doi: 10.1016/j.cosrev.2019.100193.

[13] J. S. Komputer, K. Buatan, and A. Ridwan, "Penerapan Algoritma Naïve Bayes Untuk Klasifikasi Penyakit Diabetes Mellitus," 2020.

[14] D. Papakyriakou and I. S. Barbounakis, "Data Mining Methods: A Review," *Int J Comput Appl*, vol. 183, no. 48, pp. 5–19, Jan. 2022, doi: 10.5120/ijca2022921884.

[15] A. M. Puspitasari, D. E. Ratnawati, and A. W. Widodo, "Klasifikasi Penyakit Gigi Dan Mulut Menggunakan Metode Support Vector Machine," 2018. [Online]. Available: http://j-ptiik.ub.ac.id

[16] E. Susilowati, M. Kania Sabariah, and A. Akbar Gozali, "IMPLEMENTASI METODE SUPPORT VECTOR MACHINE UNTUK MELAKUKAN KLASIFIKASI KEMACETAN LALU LINTAS PADA TWITTER IMPLEMENTATION SUPPORT VECTOR MACHINE METHOD FOR TRAFFIC JAM CLASSIFICATION ON TWITTER."

[17] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Association for Computing Machinery, Aug. 2016, pp. 785–794. doi: 10.1145/2939672.2939785.

[18] IEEE Communications Society., International Federation for Information Processing, and Institute of Electrical and Electronics Engineers, *2019 Wireless Days (WD)*.

[19] M. Syukron, R. Santoso, and T. Widiharih, "PERBANDINGAN METODE SMOTE RANDOM FOREST DAN SMOTE XGBOOST UNTUK KLASIFIKASI TINGKAT PENYAKIT HEPATITIS C PADA IMBALANCE CLASS DATA", [Online]. Available: https://ejournal3.undip.ac.id/index.php/gaussian/

[20] M. Raihan-Al-Masud and M. Rubaiyat Hossain Mondal, "Data-driven diagnosis of spinal abnormalities using feature selection and machine learning algorithms," *PLoS One*, vol. 15, no. 2, Feb. 2020, doi: 10.1371/journal.pone.0228422.