

Implementasi Sistem Klasifikasi Udara Menggunakan Metode *Decision Tree*

1st Raden M Naufal H
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

radenmuhamad@student.telkomuniversity.ac.id

2nd Meta Kallista
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

Metakallista@telkomuniversity.ac.id

3rd Ashri Dinimaharawati
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

Ashridinimaharawati@telkomuniversity.ac.id

Abstrak— Polusi udara telah menyebar luas, dari perkotaan hingga pedesaan. Oleh karena itu, pemantauan kualitas udara menggunakan Indeks Standar Pencemaran Udara (ISPU) menjadi sangat penting untuk menilai tingkat kesehatan udara di berbagai lingkungan. Penggunaan metode *Decision Tree* dalam penelitian ini memungkinkan pengujian data untuk menghasilkan klasifikasi Indeks Kualitas Udara (Air Quality Index). Hasil penelitian ini disajikan melalui aplikasi mobile yang dirancang untuk digunakan pada perangkat smartphone. Pilihan aplikasi sebagai media penyampaian informasi bertujuan memberikan edukasi yang praktis. Hasil pengujian menunjukkan bahwa penggunaan *Criterion* entropi dengan Max Depth 7 dan Test Size 10/90 menghasilkan hasil terbaik untuk algoritma *Decision Tree*. Akurasi hasil pengujian ini mencapai 0.91%, dengan presisi sebesar 0.92, recall sebesar 0.91, dan F1-Score sebesar 0.95. Hasil akurasi ini akan digunakan sebagai indikator kesehatan udara melalui aplikasi mobile yang dikembangkan.

Kata kunci— *Decision tree Method, Air Quality Index, Classification, ISPU, Apps, Particulate matter.*

I. PENDAHULUAN

Polusi udara adalah salah satu masalah utama yang sering dihadapi masyarakat. Kota-kota besar di negara berkembang. Terlebih dari Kota Jakarta yang penurunan kualitas udara dari sejak 15 Juni 2022, konsentrasi PM2.5 (partikulat matter) meningkat, mencapai maksimum 148 g/m³ (mikrogram per meter kubik). Penurunan kualitas udara di dalam dan sekitar Jakarta disebabkan oleh kombinasi sumber emisi dari sumber polusi udara dan faktor meteorologi yang berkontribusi terhadap akumulasi konsentrasi PM2.5 [1].

Berbagai material yang terkandung dalam PM2.5 ini dapat menyebabkan berbagai gangguan saluran pernafasan seperti infeksi saluran pernafasan akut (ISPA), kanker paru-paru, kardiovaskular, kematian dini, dan penyakit paru-paru obstructive kronis [2]. Dibutuhkan cara untuk mencegah terjadinya penyakit dengan menyeluruh melalui edukasi informasi dan secara praktis, dengan penyederhanaan informasi yang akan diterima oleh masyarakat akan membuat kewaspadaan dan kesadaran masyarakat menjadi lebih meningkat terhadap kualitas udara.

Maka alasan itu menentukan mana penyebaran informasi yang lebih efektif dan praktis di gunakan semua orang yang sudah mempunyai alat itu untuk digunakan sehari-hari, yaitu

Smartphone. Dengan hal ini lebih dari 86,50% masyarakat kota Jakarta mempunyai Smartphone. Maka sebab itu lebih penyebaran informasi lebih efektif [3].

Dengan memperhatikan kesehatan fisik dan lingkungan, pengamatan kualitas udara dapat diukur dengan menggunakan Indeks Standar Pencemaran Udara (ISPU). ISPU mengukur konsentrasi pencemar udara, yang mencakup lima parameter utama: karbon monoksida (CO), ozon permukaan (O₃), partikulat matter (PM), nitrogen oksida (NO₂), dan sulfur dioksida (SO₂).

II. KAJIAN TEORI

A. Polusi Udara

Pencemaran udara adalah suatu kondisi yang menyebabkan perubahan pada komposisi udara dibandingkan keadaan normal sehingga membahayakan kehidupan dan kesehatan masyarakat. Menurut PP No. 41 Tahun 1999 [4], pencemaran udara adalah masuknya atau dimasukkannya zat, energi, dan/atau komponen lain ke dalam udara ambient oleh kegiatan manusia, sehingga mutu udara ambient turun sampai ke tingkat tertentu yang menyebabkan udara ambient tidak dapat memenuhi fungsinya. Sumber polusi utama berasal dari transportasi di mana hampir 60 % dari polutan yang dihasilkan terdiri dari karbon monoksida dan sekitar 15 % terdiri dari hidrokarbon.

Di daerah perkotaan dan industri, parameter bahan pencemar yang perlu diperhatikan dalam hubungannya dengan penyakit saluran pernapasan adalah parameter gas SO₂, gas CO, gas NO₂ dan partikel debu [5]. Sumber bahan pencemar udara menentukan jenis bahan pencemarnya.

B. Air Quality Index

Indeks Kualitas Udara (AQI) merupakan nilai yang digunakan oleh instansi pemerintah untuk memberikan gambaran kualitas udara dan prediksi terjadinya pencemaran kepada masyarakat.

AQI membutuhkan nilai konsentrasi polutan yang didata-ratakan dari waktu ke waktu dari hasil pemantauan udara. Ada banyak kemungkinan polutan udara, dan rumus untuk mengubah konsentrasi polutan menjadi nilai AQI berbeda untuk setiap polutan. Setiap rentang nilai AQI digambarkan

dengan warna tertentu sesuai konvensi [6] Badan Perlindungan Lingkungan AS (EPA) telah mengembangkan indeks kualitas air yang digunakan untuk melaporkan kualitas udara. Menurut AQI, polusi jatuh ke dalam enam kategori yang mewakili peningkatan risiko dan efek kesehatan. Pada Tabel 1.2, nilai AQI di atas 300 tergolong kualitas udara tidak aman dan di bawah 50 tergolong kualitas udara baik. AQI didasarkan pada lima standar polusi yang ditetapkan dalam Clean Air Act. Yaitu, kadar ozon permukaan (O₃), partikular (PM_{2.5} dan PM₁₀), karbon monoksida (CO), sulfur dioksida (SO), dan nitrogen dioksida. (NO₂).

TABEL 1.2
Level Pencemaran berdasarkan AQI

Nilai AQI	Level Kesehatan	Warna
0-50	Baik	Hijau
51-100	Sedang	Kuning
101-150	Tidak Sehat (Sensitif)	Jingga
151-200	Tidak Sehat	Merah
201-300	Sangat Tidak Sehat	Ungu
301-500	Berbahaya	Merah Tua

Berikut adalah rumus yang digunakan untuk mendapatkan nilai AQI:

$$I = \frac{I_{high} - I_{low}}{C_{high} - C_{low}} (C - C_{low}) + I_{low}$$

Keterangan : I = Indeks Kualitas U
C = Konsentrasi Polutan
C_{low} = Batas konsentrasi ≤ C
C_{high} = Batas Konsentrasi ≥ C
I_{low} = Batas Indeks berdasarkan C_{low}
I_{high} = Batas Indeks berdasarkan C_{high}

C. Machine Learning

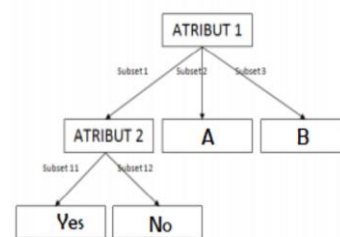
machine learning adalah bidang studi dari komputer sains yang memberikan sistem komputer kemampuan untuk belajar tanpa diprogram secara eksplisit. Kemampuan belajar suatu mesin dominan bergantung dengan algoritma yang digunakan. S. J. Russell (2014) membagi tipe pembelajaran menjadi tiga yaitu :

1. Pembelajaran terarah (*supervised learning*) Tipe ini belajar berdasarkan sekumpulan contoh pasangan *input* dan *output* yang diinginkan dalam jumlah yang cukup besar. Algoritme ini mengamati contoh-contoh tersebut dan kemudian menghasilkan sebuah model yang mampu memetakan masukan yang baru menjadi keluaran yang tepat.
2. Pembelajaran tak terarah (*unsupervised learning*) Tipe ini mempunyai tujuan untuk mempelajari dan mencari pola-pola menarik pada input yang diberikan. Meskipun tidak disediakan output yang tepat secara eksplisit. Salah satu algoritma *unsupervised learning* yang paling umum digunakan adalah clustering/pengelompokan.

3. Pembelajaran semi terarah (*semi-supervised learning*) *semi-supervised learning* merupakan tipe yang menggabungkan antara *supervised* dan *unsupervised* untuk menghasilkan suatu fungsi. Algoritme pembelajaran semi terarah menggabungkan kedua tipe algoritme di atas, di mana diberikan contoh masukan-keluaran yang tepat dalam jumlah sedikit dan sekumpulan masukan yang keluarannya belum diketahui.

D. Metode Decision tree

Metode *Decision tree* adalah jenis alur atau flowchart seperti pohon dimana setiap node menunjukkan suatu test pada suatu atribut, tiap branch merepresentasikan hasil dari test tersebut, dan leaf node menunjukkan kelas-kelas atau distribusi kelas. *Decision tree* berguna untuk mengeksplorasi data yang sudah melewati tahap *pre-processing* dan menemukan model yang tersembunyi dari data dengan sebuah target variabel sehingga dapat digunakan untuk membagi kumpulan data yang relatif besar terhadap variabel target menjadi kumpulan record yang lebih kecil.



Gambar 2. 1
Diagram Air Decision tree

(tegas) sebagai hasil akhirnya. Langkah-langkah yang dilakukan pada tahap ini, seperti:

1. Melakukan proses agregasi atau *composition* dari hasil yang diperoleh dari bagian inferensi.
2. Melakukan perhitungan nilai *crisp* menggunakan metode *Centroid Method* dengan persamaan.

Decision tree merupakan salah satu metode untuk mengklasifikasikan data (Romadloni et al., 2019). Menggunakan konsep diagram alir yang mirip dengan struktur II-7 pohon, dimana internal node menotasikan atribut yang diuji, setiap cabang nya merepresentasikan hasil dari atribut tes tersebut, dan leaf node merepresentasikan kelas-kelas tertentu atau distribusi kelas-kelas. Hasil dari proses klasifikasi yang berupa aturan-aturan dapat digunakan untuk memprediksi nilai atribut bertipe diskrit dari record yang baru [7].

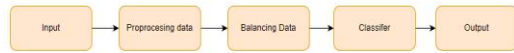
Pada umumnya ada beberapa ciri kasus yang cocok untuk diterapkan dengan algoritma ini, antara lain:

- a. Data dinyatakan dengan pasangan atribut dan nilainya misalnya atribut temperature dan nilainya adalah dingin.
- b. Label/ output data biasanya bernilai diskrit. Output nya bias bernilai ya atau tidak, sakit atau tidak sakit, diterima atau ditolak.
- c. Data mempunyai missing value. Misalkan untuk beberapa data, nilai dari suatu atribut nya tidak diketahui. Maka dalam keadaan ini *Decision tree* masih mampu member solusi yang baik.

III. METODE

A. Perancangan Diagram Sistem

Sistem klasifikasi prediksi Indeks Standar Pencemaran Udara (ISPU) dengan menggunakan metode *Decision tree* memiliki gambaran umum dimana terdapat Langkah-langkah untuk mendapatkan hasil akhir. Untuk lebih jelasnya dapat dilihat diagram proses di bawah ini.



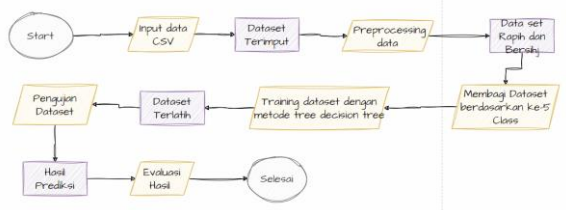
GAMBAR 4
(Langkah proses system)

Gambar 3, adalah Langkah – Langkah dari proses sistem yang dijelaskan seperti dibawah berikut. Melakukan proses klasifikasi. Penjelasan untuk setiap Langkah-langkahnya seperti berikut.

1. Pertama-tama, melakukan pengunduhan data set ISPU Jakarta yang ada pada *website Jakarta Open Data*.
2. Selanjutnya, melakukan *preprocessing* data, *preprocessing* data adalah proses mempersiapkan atau memperbaiki data agar proses klasifikasi berjalan dengan lancar.
3. Kemudian melakukan proses imbalanced data, yaitu membuat agar data nilainya seimbang sehingga akan melancarkan proses klasifikasi.
4. Setelah itu dilakukan klasifikasi data menggunakan *Decision tree*.
5. Yang terakhir, proses *output*, yaitu menampilkan hasil klasifikasi di situs web.

B. Perancangan Metode *Decision tree*

Perancangan program prediksi ISPU dimulai dari pengumpulan data, *preprocessing* data, pelatihan data dengan metode *Decision tree*, evaluasi hasil prediksi. Berikut gambar 3.7 yang menjelaskan diagram alir perancangan program ini.



GAMBAR 4
(Flow Metode *Decision Tree*)

C. *Confusion Matrix*

Confusion Matrix adalah sebuah tabel untuk mengevaluasi kinerja dari suatu model klasifikasi. Tabel ini menampilkan jumlah prediksi benar dan salah oleh model untuk setiap kelas target[10].

Confusion Matrix terdiri dari empat sel utama, yaitu:

- True Positive (TP): Jumlah sampel bernilai salah diklasifikasikan dengan benar sebagai positif.
- True Negative (TN): Jumlah sampel bernilai salah diklasifikasikan dengan benar sebagai negatif.
- False Positive (FP): Jumlah sampel bernilai salah diklasifikasikan sebagai positif.

- False Negative (FN): Jumlah sampel bernilai salah diklasifikasikan sebagai negatif. *Confusion Matrix* membantu dalam menghitung berbagai metrik evaluasi seperti akurasi, presisi, recall, dan F1-score.

D. *Classification Report*

Classification report adalah laporan memberikan ringkasan dari hasil klasifikasi pada model machine learning. Laporan ini menyediakan metrik evaluasi seperti akurasi, presisi, recall, support dan F1-score untuk setiap kelas target[10].

Laporan klasifikasi umumnya terdiri dari beberapa bagian, termasuk:

- a. Akurasi: Persentase dari prediksi benar secara keseluruhan.
- b. Presisi: Persentase dari prediksi positif benar.
- c. Recall: Persentase dari sampel positif berhasil diidentifikasi dengan benar.
- d. F1-score: Rata-rata harmonik antara presisi dan recall.
- e. Support: Jumlah sampel dalam setiap kelas target.

Rumus manual untuk menghasilkan laporan klasifikasi tidak ada dalam literatur. Namun, laporan klasifikasi umumnya mencakup metrik evaluasi seperti akurasi, presisi, recall, dan F1-score yang dapat dihitung menggunakan rumus berikut:

- Akurasi: $Akurasi = (TP + TN) / (TP + TN + FP + FN)$
- Presisi: $Presisi = TP / (TP + FP)$
- Recall: $Recall = TP / (TP + FN)$
- F1-score: $F1\text{-score} = 2 * (Presisi * Recall) / (Presisi + Recall)$

Di sini, TP adalah True Positive (jumlah sampel yang diklasifikasikan dengan benar sebagai positif), TN adalah True Negative (jumlah sampel yang diklasifikasikan dengan benar sebagai negatif), FP adalah False Positive (jumlah sampel yang salah diklasifikasikan sebagai positif), dan FN adalah False Negative (jumlah sampel yang salah diklasifikasikan sebagai negatif)[10].

E. *Cross-validation*

Cross-validation adalah teknik yang digunakan dalam machine learning dan statistik untuk mengevaluasi kinerja suatu model pada dataset yang independen[12].

Cross-validation membantu dalam menilai sejauh mana suatu model dapat menggeneralisasi data yang belum pernah dilihat sebelumnya dan dapat digunakan untuk menentukan *hyperparameter*, membandingkan model yang berbeda, atau memperkirakan kinerja model pada data baru.

cross-validation dapat dijelaskan sebagai berikut:

1. Bagi dataset menjadi k subset yang sama ukurannya.
2. Untuk setiap subset ke-i, gunakan subset tersebut sebagai data validasi dan gabungkan sisa subset sebagai data pelatihan.
3. Latih model pada data pelatihan dan evaluasi pada data validasi.
4. Hitung metrik evaluasi yang diinginkan (misalnya akurasi, presisi, atau *recall*).
5. Ulangi langkah 2-4 untuk setiap subset.

Hitung rata-rata dari metrik evaluasi yang diperoleh dari setiap iterasi.

Rumus *cross-validation* dapat dituliskan sebagai berikut:

$$[\text{Cross-validation score}] = \frac{1}{k} \sum_{i=1}^k [\text{Metric}_i]$$

Atau

$$\text{Cross validation score} = \frac{1}{k} \sum_{i=1}^k \text{Metric}_i$$

di mana k adalah jumlah subset yang digunakan dalam *cross-validation*, dan Metric_i adalah metrik evaluasi yang diperoleh dari subset ke- i [12].

IV. HASIL DAN PEMBAHASAN

Pada pengujian Implementasi klasifikasi prediksi kualitas udara melibatkan pengujian dan analisis kinerja algoritma yang digunakan. Untuk melakukan klasifikasi, digunakan beberapa alat, dengan bahasa pemrograman *Python* yang dijalankan melalui *Anaconda*. *Python Anaconda* adalah bahasa pemrograman yang digunakan untuk pengembangan machine learning dan memungkinkan penulisan kode melalui *Jupyter Notebook*.

Algoritma yang digunakan dalam proses klasifikasi yaitu Decision tree CART (*Classification and Regression Tree*) yang diperkenalkan Ross Quinlan.

A. Pengujian Dataset yang setelah *Preprocessing*

Dalam penelitian ini, terdapat 47,275 data pada dataset yang digunakan. Setiap data memiliki lima kelas yaitu Baik, Sedang, Tidak Sehat, Sangat Tidak Sehat, dan Berbahaya, dengan masing-masing kelas terdiri dari 9,455 data setelah melalui proses *preprocessing*. Setelah tahap *preprocessing* selesai, dilakukan proses klasifikasi menggunakan atribut yang telah disebutkan sebelumnya.

Nilai yang di pakai menggunakan *criterion* entropi dan gini dengan *Max Depth* 3 dengan *Test size* di bawah :

Criterion	Max Depth	Train/Test	Hasil Akurasi
Entropi	3	90/10	0.861886632
		80/20	0.858064516
		60/40	0.852829190
		40/60	0.852388506
		20/80	0.855023796
gini	3	90/10	0.86148665
		80/20	0.882191201
		60/40	0.888207297
		40/60	0.749127445
		20/80	0.599497620

Nilai yang di pakai menggunakan *criterion* entropi dan gini dengan *Max Depth* 4 dengan *Test size* di bawah :

TABEL 1
(Hasil kuesioer hiburan)

Criterion	Max Depth	Train/Test	Hasil Akurasi
Entropi	5	90/10	0.9585448392
		80/20	0.9598096245
		60/40	0.9595452141
		40/60	0.9592455490
		20/80	0.9590428344
gini	5	90/10	0.86148665
		80/20	0.9648900169
		60/40	0.9575885774
		40/60	0.9304071919
		20/80	0.9295963335

Nilai yang di pakai menggunakan *criterion* entropi dan gini dengan *Max Depth* 7 dengan *Test size* di bawah :

(Hasil kuesioer hiburan)

Criterion	Max Depth	Train/Test	Hasil Akurasi
Entropi	7	90/10	0.9932318104
		80/20	0.9929138022
		60/40	0.9918561607
		40/60	0.9930900757
		20/80	0.9925436277
gini	7	90/10	0.9862521150
		80/20	0.9862521150
		60/40	0.9839238498
		40/60	0.9839238498
		20/80	0.9692578882

Nilai yang di pakai menggunakan *criterion* entropi dan gini dengan *Max Depth* 9 dengan *Test size* di bawah :

Criterion	Max Depth	Train/Test	Hasil Akurasi
Entropi	9	90/10	0.9968274111
		80/20	0.9968274111
		60/40	0.9966684294
		40/60	0.9950290851

		20/80	0.9957165520
gini	9	90/10	0.9862521150
		80/20	0.9945002644
		60/40	0.9934955050
		40/60	0.9928080380
		20/80	0.9941300898

B. Pengujian Dataset Preprocessing setelah dipotong 5000

Pada pengujian kali ini, dataset yang sebelumnya telah dipotong menjadi 5000 data akan digunakan. Setiap kelas dataset memiliki jumlah data sebanyak 4455 setelah melalui proses *preprocessing*. Pengujian akan dilakukan dengan menggunakan *hyperparameter*, yaitu parameter yang tidak dipelajari dari data, melainkan ditetapkan oleh pengguna sebelum melatih model.

Di dapat atribut yang ideal untuk klasifikasi maka dari itu data akan di lakukan nya proses-mengulang untuk data yang di potong.

```

In [42]:
# Create a decision tree classifier
clf = DecisionTreeClassifier()

# Use cross-validation to tune the hyperparameters
params = {'max_depth': [1, 3, 5, 7], 'criterion': ['entropy', 'gini']}
dtc = GridSearchCV(clf, params, cv=5, scoring='accuracy')
dtc.fit(X_train, y_train)

# Print the best hyperparameters and score
print("Best hyperparameters: ", dtc.best_params_)
print("Best score: ", dtc.best_score_)
pred_dtest = dtc.predict(X_test)

Best hyperparameters: {'criterion': 'gini', 'max_depth': 7}
Best score: 0.9870304906230309

```

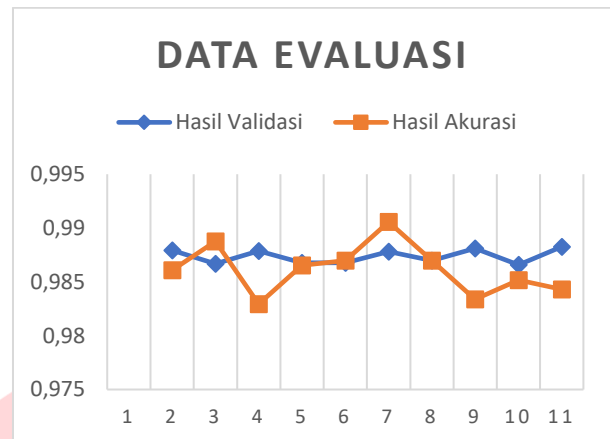
Hasil proses Cross-Validation

Untuk menyesuaikan *hyperparameter* pohon keputusan menggunakan validasi silang, kita dapat menggunakan fungsi “*GridSearchCV*” di Scikit-learn. Fungsi ini mengambil model, sekumpulan *hyperparameter* yang akan dicari, dan metrik penilaian, dan melakukan pencarian grid pada semua kemungkinan kombinasi *hyperparameter*.

Hasil terbaik ditemukan dengan *hyperparameter* terbaik yaitu *criterion*: 'gini' dan *Max Depth*: '7', dengan skor terbaik sebesar 0.9870304906230309. Selanjutnya, akan dilakukan pengujian sebanyak 10 kali untuk melihat apakah nilai akurasi akan berubah atau tidak. Hasil pengujian akan direkam dalam Tabel di bawah ini.

no	Hasil Validasi	Hasil Akurasi	Hasil Train	Hasil Test
1	0.987928383	0.986086176	0.985733526	0.985733526
2	0.986681301	0.988779174	0.984985285	0.988779174
3	0.987878358	0.982944345	0.986032823	0.982944345
4	0.986781201	0.986535009	0.984486457	0.986535009
5	0.986781226	0.986983842	0.985035167	0.986983842
6	0.987828645	0.990574506	0.986132588	0.990574506
7	0.986983842	0.986983842	0.986182471	0.986983842
8	0.988127835	0.983393178	0.986431885	0.983393178
9	0.986581663	0.98518851	0.985683643	0.98518851
10	0.988277498	0.984290844	0.986232354	0.984290844

Hasil dari membandingkan akurasi dan hasil validasi akan menjadi penentu apakah dataset menghasilkan akurasi yang tepat atau tidak.



GAMBAR 4. 1
Evaluasi hasil validasi

Dari Gambar 4.9, terlihat bahwa penggunaan teknik validasi silang (*cross-validation*) untuk mengukur kinerja model pada dataset yang di latih selama pelatihan menyebabkan model *Decision tree* mampu menghasilkan akurasi dan validasi hampir mencapai hasil yang sama. Hal ini menunjukkan tidak terjadinya *overfitting* pada akurasi model.

V. KESIMPULAN

Setelah melakukan dua pengujian model *Decision tree*, pada pengujian kedua dataset ISPU DKI Jakarta dengan metode SMOTE, telah dipotong menjadi 4,455 data untuk setiap dari lima kelas. Kemudian, dilakukan metode *cross-validation*, yang menghasilkan akurasi sebesar 0.9878784580081351 atau 0.98 dengan *hyperparameter* terbaik yaitu *criterion* gini dan *Max Depth* 7. Selanjutnya, pengujian diulangi hingga 10 kali dengan mencoba membandingkan hasil validasi dan akurasi. Hal ini menunjukkan tidak terjadinya *overfitting* pada akurasi model.

REFERENSI

- [1] [“Perkembangan Terakhir Kondisi Kualitas Udara di Wilayah Jakarta dan Sekitarnya | BMKG.” <https://www.bmkg.go.id/press-release/?p=perkembangan-terakhir-kondisi-kualitas-udara-di-wilayah-jakarta-dan-sekitarnya&tag=press-release&lang=ID> (accessed Oct. 21, 2022).
- [2] [2] Elsa Try Julita Sembiring, “HEALTH RISK OF EXPOSURE TO PM2,5 IN AMBIEN AIR TO STREET VENDORS UNDER FLYOVER PASAR PAGI ASEMKA JAKARTA,” *Jurnal Teknik*

- Lingkungan Volume 26 Nomor 1*, vol. 1, pp. 101–120, Apr. 2020.
- [3] [3] “66,3% masyarakat Indonesia Memiliki *Smartphone* | Indonesia Baik.” <https://indonesiabaik.id/infografis/663-masyarakat-indonesia-memiliki-Smartphone-8> (accessed Oct. 21, 2022).
- [4] [4] “PP No. 41 Tahun 1999 tentang Pengendalian Pencemaran Udara [JDIH BPK RI].” <https://peraturan.bpk.go.id/Home/Details/54332/pp-no-41-tahun-1999> (accessed Oct. 25, 2022).
- [5] [5] J. Rotton *et al.*, “Air Pollution, Weather, and Violent Crimes: Concomitant Time-Series Analysis of Archival Data,” 1985.
- [6] [6] M. Kattariya Kujaroentavon, B. Mahaisavariya, D. Thai Board, A. Supaporn Kiattisin, and C. Engineering, “Thesis entitled AIR QUALITY CLASSIFICATION IN THAILAND.”
- [7] [7] M. Muhamad, A. P. Windarto, and S. Suhada, “PENERAPAN ALGORITMA C4.5 PADA KLASIFIKASI POTENSI SISWA DROP OUT,” *KOMIK (Konferensi Nasional Teknologi Informasi dan Komputer)*, vol. 3, no. 1, Dec. 2019, doi: 10.30865/komik.v3i1.1688.
- [8] [8] N. V Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “SMOTE: Synthetic Minority Over-sampling Technique,” 2002.
- [9] [9] S. Aning and M. Przybyla-Kasperek, “Comparative Study of Twoing and Entropy Criterion for *Decision tree* Classification of Dispersed Data,” in *Procedia Computer Science*, Elsevier B.V., 2022, pp. 2434–2443. doi: 10.1016/j.procs.2022.09.301.
- [10] [10] Aurélien Geron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 2nd Edition*, 2nd ed., vol. 2. Sebastopol: O’Reilly Media, Inc, 2019. Accessed: Sep. 05, 2023. [Online]. Available: https://powerunit-ju.com/wp-content/uploads/2021/04/Aurelien-Geron-Hands-On-Machine-Learning-with-Scikit-Learn-Keras-and-Tensorflow_-Concepts-Tools-and-Techniques-to-Build-Intelligent-Systems-OReilly-Media-2019.pdf