

Implementasi Convolutional Neural Network dengan SMOTE+ENN untuk Klasifikasi Kualitas Udara Berdasarkan Data Deret Waktu Polutan

Cahyono Budy Santoso*, Maria Rachel Kesya Makarena

Fakultas Teknik dan Desain, Sistem Informasi, Universitas Pembangunan Jaya, Tangerang Selatan, Indonesia

Email: ^{1,*}cahyono.budy@upj.ac.id, ²maria.rachelkesya@student.upj.ac.id

Email Penulis Korespondensi: cahyono.budy@upj.ac.id

Submitted: 18/07/2025; Accepted: 01/09/2025; Published: 04/09/2025

Abstrak—Penurunan kualitas udara di kota metropolitan seperti Jakarta telah menjadi isu lingkungan dan kesehatan masyarakat yang mendesak, dengan dampak langsung terhadap peningkatan risiko berbagai penyakit. Tujuan penelitian ini adalah untuk mengembangkan dan mengevaluasi efektivitas model klasifikasi kualitas udara berbasis *Convolutional Neural Network* (CNN), khususnya dalam mengatasi masalah ketidakseimbangan data menggunakan teknik resampling hibrida SMOTE+ENN. Model ini dilatih menggunakan data historis dari Stasiun HI Jakarta periode 2010-2021, dengan parameter polutan utama (PM10, SO₂, CO, O₃, dan NO₂) sebagai *input* untuk memprediksi kategori kualitas udara sesuai standar ISPU. Salah satu tantangan utama dalam data ini adalah ketidakseimbangan kelas. Oleh karena itu, penelitian ini melakukan perbandingan antara model CNN standar yang dilatih pada data asli dengan model yang dioptimalkan menggunakan teknik *resampling* hibrida, yaitu *Synthetic Minority Over-sampling Technique* dan *Edited Nearest Neighbor* (SMOTE + ENN). Data dibagi dengan proporsi 80% untuk pelatihan dan 20% untuk pengujian. Hasil evaluasi menunjukkan bahwa penerapan SMOTE + ENN secara signifikan meningkatkan kinerja. Model *final* yang dioptimalkan berhasil mencapai akurasi unggul sebesar 98,98%, jauh melampaui model dasar. Temuan ini membuktikan bahwa penggabungan CNN dengan strategi SMOTE + ENN menghasilkan sistem klasifikasi yang terbukti sangat efektif dan andal untuk kualitas udara di Jakarta. Meskipun demikian, pengujian lebih lanjut pada *dataset* yang lebih beragam sangat direkomendasikan untuk memastikan kemampuan model untuk beradaptasi dan tetap andal di masa depan.

Kata Kunci: CNN; ISPU; Klasifikasi; Pembelajaran Mesin; SMOTE

Abstract—The degradation of air quality in metropolitan areas, such as Jakarta, constitutes a significant environmental and public health challenge, contributing directly to an elevated risk of various diseases. The primary objective of this study is to develop and evaluate the effectiveness of an air quality classification model based on a Convolutional Neural Network (CNN), with a specific focus on addressing class imbalance using the hybrid resampling technique SMOTE+ENN. Utilizing a historical dataset from the HI Jakarta Station spanning 2010-2021, the model leverages key pollutant parameters (PM10, SO₂, CO, O₃, and NO₂) to classify air quality according to the Indonesian Air Quality Index (ISPU) standard. To mitigate the inherent challenge of class imbalance within the dataset, this study conducts a comparative analysis between a baseline CNN model and an optimized model enhanced with the hybrid resampling technique, Synthetic Minority Over-sampling Technique and Edited Nearest Neighbours (SMOTE + ENN). The dataset was partitioned into an 80% training set and a 20% testing set. Empirical results demonstrate that the application of SMOTE + ENN yields a substantial improvement in performance. The final optimized model achieves a superior accuracy of 98.98%, significantly outperforming the baseline model. This outcome confirms that integrating CNN with the SMOTE + ENN strategy produces a highly effective and robust framework for air quality classification in Jakarta. Nonetheless, subsequent validation on more diverse datasets is recommended to ascertain the model's generalization capabilities and long-term reliability.

Keywords: CNN; ISPU; Classification; Machine Learning; SMOTE

1. PENDAHULUAN

Kualitas udara yang baik merupakan prasyarat fundamental untuk menjaga kesehatan manusia serta keberlangsungan ekosistem [1]. Namun, seiring dengan meningkatnya laju industrialisasi dan urbanisasi, pencemaran udara telah menjadi krisis global yang berdampak negatif secara langsung pada kesehatan publik dan keseimbangan lingkungan [1], [2]. Paparan jangka panjang terhadap polutan udara berbahaya seperti Partikulat (PM10), Karbon Monoksida (CO), Sulfur Dioksida (SO₂), Nitrogen Dioksida (NO₂), dan Ozon (O₃) telah terbukti secara ilmiah dapat memicu berbagai penyakit kronis, terutama pada sistem pernapasan dan kardiovaskular [3], serta menimbulkan beban ekonomi dan sosial yang signifikan akibat biaya perawatan kesehatan dan hilangnya produktivitas. Jakarta, sebagai pusat ekonomi dan pemerintahan Indonesia, secara konsisten tercatat sebagai salah satu kota dengan tingkat kualitas udara yang buruk dan tidak sehat, sebagaimana dilaporkan oleh berbagai lembaga pemantau global. Kondisi yang mengkhawatirkan ini menggarisbawahi urgensi adanya sistem pemantauan dan klasifikasi kualitas udara yang akurat, andal, dan dapat diakses sebagai fondasi mitigasi risiko kesehatan masyarakat.

Sebagai respons terhadap tantangan tersebut, Kementerian Lingkungan Hidup dan Kehutanan (KLHK) Republik Indonesia telah menetapkan Indeks Standar Pencemar Udara (ISPU) sebagai acuan resmi untuk menginformasikan kondisi kualitas udara ambien kepada publik di seluruh wilayah Indonesia. ISPU berfungsi sebagai alat komunikasi risiko dengan menyederhanakan data konsentrasi polutan yang kompleks ke dalam beberapa kategori yang mudah dipahami, seperti 'Baik', 'Sedang', dan 'Tidak Sehat', berdasarkan lima parameter utama: CO, SO₂, NO₂, O₃, dan PM10. Akurasi dalam proses klasifikasi tingkat ISPU menjadi faktor krusial, karena berfungsi sebagai sistem peringatan dini bagi masyarakat dan menjadi landasan data bagi pemerintah dalam merumuskan kebijakan strategis pengendalian pencemaran udara yang efektif [4].

Sejumlah penelitian telah mengeksplorasi penerapan metode *machine learning* untuk memprediksi dan mengklasifikasikan tingkat kualitas udara, khususnya di DKI Jakarta. Penelitian yang dilakukan oleh Julia Putri [5] menerapkan metode *Support Vector Machine* (SVM) untuk klasifikasi ISPU dan berhasil mencapai akurasi sebesar 93,47% dengan hanya mempertimbangkan parameter pm_{10} dan pm_{25} . Penelitian lain oleh Putri Salsabila [6] menggunakan algoritma *Naïve Bayes* yang menghasilkan akurasi rata-rata 92%. Pendekatan berbeda dilakukan oleh Sukiman [7] yang menerapkan metode *Extreme Learning Machine* (ELM) dalam studi kasus klasifikasi kualitas udara Jakarta. Dengan menggunakan teknik *oversampling* untuk menyeimbangkan data, penelitian tersebut berhasil memperoleh akurasi sebesar 96%. Studi komparatif oleh Ma'û Luthfi dan Fauzi [8] membandingkan performa *Random Forest*, SVM, dan LGBM, yang menunjukkan variasi hasil tergantung pada algoritma yang digunakan. Lebih lanjut, penelitian sebelumnya oleh Shafira [9] juga telah mengaplikasikan *Decision Tree* dan *Fuzzy Mamdani* untuk penelitian serupa. Secara kolektif, penelitian-penelitian ini mengonfirmasi kelayakan penggunaan *machine learning* untuk analisis ISPU, namun juga secara implisit menyoroti adanya ruang untuk peningkatan akurasi melalui eksplorasi arsitektur model yang lebih canggih. Dalam hal ini, *Convolutional Neural Network* (CNN) menunjukkan potensi yang signifikan. Keberhasilannya dalam mengenali pola kompleks pada data deret waktu telah dibuktikan dalam penelitian oleh Taliningsih [10], yang berhasil menerapkan CNN 1 Dimensi untuk otentikasi biometrik dari sinyal EKG. Keberhasilan ini menunjukkan potensi besar CNN dalam mengenali pola kompleks pada data deret waktu, seperti data parameter ISPU yang digunakan dalam penelitian ini. Mengingat studi lain juga menunjukkan keunggulan akurasi CNN dibandingkan SVM dalam domain yang berbeda [11], penerapannya untuk klasifikasi ISPU berbasis data multi-parameter di Indonesia menjadi sangat relevan untuk dieksplorasi lebih dalam.

Meskipun studi-studi tersebut memberikan landasan yang penting, terdapat celah penelitian yang signifikan. Pertama, sebagian besar penelitian sebelumnya belum mengeksplorasi arsitektur *deep learning* yang lebih canggih, yang berpotensi menawarkan keunggulan dalam pengenalan pola pada data polutan yang kompleks. Selain itu, tantangan spesifik mengenai ketidakseimbangan data antar kategori ISPU (misalnya, jumlah hari dengan status 'Baik' jauh lebih sedikit dibandingkan 'Sedang'), yang dapat menurunkan performa dan keandalan model, belum menjadi fokus utama dalam penelitian-penelitian tersebut. Celah inilah yang menjadi justifikasi untuk mengusulkan pendekatan baru yang lebih andal.

Keberhasilan penerapan metode-metode tersebut menjadi landasan yang kuat untuk mengeksplorasi pendekatan yang lebih baru, sejalan dengan perkembangan pesat di bidang kecerdasan buatan. Oleh karena itu, arsitektur *deep learning* kini menjadi alternatif yang relevan untuk dikaji lebih lanjut. Secara khusus, *Convolutional Neural Network* (CNN) menawarkan keunggulan yang signifikan. Arsitektur ini pada dasarnya terdiri dari lapisan-lapisan kunci seperti konvolusi (*convolution*), *pooling*, dan *Fully-Connected Layer* yang telah terbukti sangat efektif dalam tugas identifikasi pada data citra atau dua dimensi [12]. Namun, keunggulan utama CNN tidak terbatas pada analisis citra. Kemampuannya dalam mengekstraksi fitur secara hierarkis juga sangat efektif ketika diadaptasi untuk data sekuensial atau deret waktu satu dimensi.

Oleh karena itu, penelitian ini bertujuan untuk merancang dan mengimplementasikan model *Convolutional Neural Network* (CNN) untuk klasifikasi kualitas udara berdasarkan data deret waktu polutan di Jakarta, mengintegrasikan metode *resampling* hibrida *Synthetic Minority Over-sampling Technique* dan *Edited Nearest Neighbor* (SMOTE+ENN) untuk menangani masalah ketidakseimbangan data, serta mengevaluasi apakah model CNN yang dioptimalkan mampu mencapai tingkat akurasi dan keandalan yang lebih tinggi dibandingkan metode pada penelitian sebelumnya. Dengan demikian, kontribusi utama penelitian ini adalah menyediakan model klasifikasi ISPU yang menggabungkan arsitektur CNN dengan penanganan data tidak seimbang, sehingga menghasilkan kerangka kerja yang lebih akurat dan andal sebagai alat bantu pengambilan kebijakan lingkungan.

2. METODOLOGI PENELITIAN

2.1 Objek Penelitian

Bagian ini mendefinisikan tiga komponen utama dalam penelitian ini: domain masalah yang dikaji (ISPU), model klasifikasi yang diterapkan (CNN), dan teknik penanganan data yang digunakan (SMOTE+ENN).

2.1.1 Indeks Standar Pencemar Udara (ISPU)

Indeks Standar Pencemar Udara (ISPU) adalah angka acuan resmi yang digunakan oleh Pemerintah Indonesia untuk menginformasikan kondisi kualitas udara ambien kepada masyarakat secara sederhana dan seragam di seluruh wilayah Indonesia. ISPU ditetapkan oleh Kementerian Lingkungan Hidup dan Kehutanan (KLHK) dan berfungsi sebagai sistem peringatan dini kesehatan masyarakat dengan menyederhanakan data konsentrasi polutan yang kompleks ke dalam format yang mudah dipahami [13]. Secara esensial, ISPU berfungsi untuk menerjemahkan perbandingan antara data konsentrasi polutan dengan ambang batas amannya menjadi sebuah indikator sederhana, sehingga dapat menjadi alat komunikasi risiko yang efektif bagi masyarakat untuk mengambil tindakan perlindungan [14].

Berdasarkan Peraturan Menteri Lingkungan Hidup dan Kehutanan Nomor 14 Tahun 2020, perhitungan ISPU didasarkan pada konversi data konsentrasi dari enam parameter polutan utama, yaitu Partikulat (PM₁₀ dan PM₂₅), Karbon Monoksida (CO), Sulfur Dioksida (SO₂), Nitrogen Dioksida (NO₂), Ozon (O₃), dan Hidrokarbon (HC)[13]. Nilai konsentrasi dari setiap parameter diubah menjadi skala indeks, dan nilai indeks tertinggi di antara semua

parameter tersebut akan menjadi nilai ISPU yang dilaporkan pada hari itu. Selanjutnya, nilai ISPU tersebut diklasifikasikan ke dalam lima kategori kesehatan sebagai berikut:

- a. 0 – 50: Baik
- b. 51 – 100: Sedang
- c. 101 – 200: Tidak Sehat
- d. 201 – 300: Sangat Tidak Sehat
- e. > 301: Berbahaya

2.1.2 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) adalah salah satu jenis arsitektur *deep learning* yang terinspirasi dari cara kerja sistem visual biologis. Keunggulan utama CNN terletak pada kemampuannya untuk melakukan ekstraksi fitur secara otomatis dan hierarkis langsung dari data mentah, sehingga mengurangi kebutuhan untuk rekayasa fitur manual (*manual feature engineering*) yang kompleks [15]. Meskipun awalnya meraih popularitas masif dalam tugas-tugas berbasis data dua dimensi (2D) seperti klasifikasi citra, kapabilitas CNN tidak terbatas pada domain tersebut [16]. Secara umum, arsitektur 1D-CNN yang digunakan dalam penelitian ini terdiri dari tiga jenis lapisan utama [15]:

- a. Lapisan Konvolusi (*Convolutional Layer*): Lapisan ini menerapkan serangkaian filter untuk melakukan operasi konvolusi pada data *input*. Setiap filter bertugas untuk mendeteksi fitur atau pola spesifik, dan hasilnya adalah sebuah *feature map* yang merepresentasikan deteksi pola tersebut di sepanjang data.
- b. Lapisan *Pooling* (*Pooling Layer*): Setelah lapisan konvolusi, lapisan *pooling* (umumnya *Max Pooling*) diterapkan untuk mengurangi dimensi spasial dari *feature map*. Proses ini membantu membuat representasi fitur lebih andal terhadap variasi posisi dan mengurangi jumlah parameter komputasi, sehingga mencegah *overfitting*.
- c. Lapisan *Fully-Connected* (*Dense Layer*): Lapisan ini berfungsi sebagai classifier akhir. Setelah fitur-fitur tingkat tinggi diekstraksi oleh lapisan-lapisan sebelumnya, lapisan *Fully-Connected* akan memetakan fitur-fitur tersebut ke kelas-kelas target. Pada lapisan *output*, fungsi aktivasi *SoftMax* digunakan untuk menghasilkan probabilitas untuk setiap kategori.

2.1.3 Penanganan Data Tidak Seimbang: SMOTE+ENN

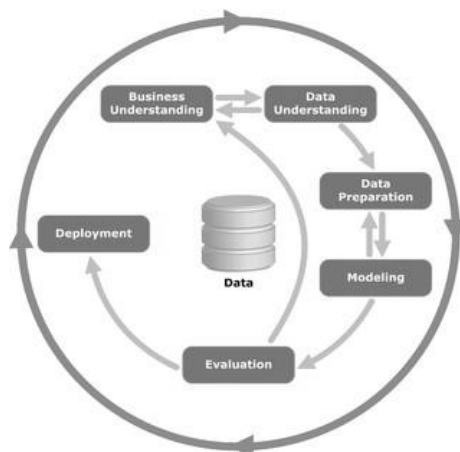
Salah satu tantangan yang umum ditemukan adalah masalah ketidakseimbangan kelas (*class imbalance*), yaitu kondisi di mana distribusi data antar kelas sangat timpang, dengan satu kelas (mayoritas) jauh lebih banyak daripada kelas lainnya (minoritas). Penggunaan data yang tidak seimbang tanpa pra-pemrosesan dapat menghasilkan model yang bias, yang cenderung berkinerja baik pada kelas mayoritas tetapi sangat buruk dalam mengenali kelas minoritas, yang seringkali merupakan kelas yang paling penting untuk dideteksi [17]. Untuk mengatasi masalah ini, penelitian ini menerapkan. Untuk mengatasi masalah ini, penelitian ini menerapkan SMOTE+ENN, sebuah teknik resampling hibrida yang terbukti efektif untuk menyeimbangkan distribusi kelas sekaligus meningkatkan kualitas data [18]. Pendekatan ini terdiri dari dua tahap utama:

- a. *Oversampling* dengan SMOTE (*Synthetic Minority Over-sampling Technique*)
SMOTE adalah teknik *oversampling* yang membuat sampel sintetis baru untuk kelas minoritas, bukan hanya menduplikasi data yang ada. Prosesnya bekerja dengan memilih sebuah sampel dari kelas minoritas, kemudian membuat sampel sintetis baru di sepanjang segmen garis yang menghubungkan sampel tersebut dengan tetangga terdekatnya (*k-nearest neighbors*) yang juga berasal dari kelas minoritas [19]. Dengan cara ini, SMOTE memperkaya kelas minoritas dengan data baru yang beragam, membantu model untuk melakukan generalisasi dengan lebih baik dan mengurangi risiko *overfitting*.
- b. Pembersihan Data dengan ENN (Edited Nearest Neighbors)
Setelah proses *oversampling* dengan SMOTE, teknik ENN diterapkan sebagai mekanisme pembersihan data. Tujuan ENN adalah untuk menghapus instansi data yang dianggap *noise* atau ambigu dari kedua kelas. ENN bekerja dengan cara memeriksa setiap sampel dan menghapus sampel yang label kelasnya berbeda dari mayoritas label kelas tetangga terdekatnya. Proses pembersihan ini penting untuk menghilangkan sampel sintetis yang mungkin tumpang tindih dengan kelas mayoritas serta membersihkan data asli dari outliers, sehingga menghasilkan batas keputusan antar kelas yang lebih jelas dan akurat [19].

SMOTE+ENN terbukti lebih unggul karena tidak hanya menyeimbangkan jumlah data, tetapi juga meningkatkan kualitas data secara keseluruhan, yang mengarah pada performa klasifikasi yang lebih baik dibandingkan menggunakan SMOTE atau ENN secara terpisah. Studi menunjukkan bahwa SMOTE+ENN secara konsisten menghasilkan akurasi yang lebih tinggi dan tingkat kesalahan yang lebih rendah, serta mampu menciptakan batas keputusan yang lebih bersih pada *dataset*.

2.2 Tahapan Penelitian

Untuk memastikan proses penelitian berjalan secara terstruktur dan sistematis, metodologi yang digunakan mengacu pada kerangka kerja *Cross-Industry Standard Process for Data Mining (CRISP-DM)*. CRISP-DM merupakan metodologi standar industri yang teruji untuk memandu proyek analisis data secara bertahap dan iteratif [20]. Alur kerja dari metodologi ini, yang terdiri dari enam fase utama, diilustrasikan pada Gambar 1.



Gambar 1. Tahapan CRISP-DM

Berdasarkan Gambar 1, keenam tahapan utama tersebut, meliputi *Business Understanding*, *Data Understanding*, *Data Preparation*, *Modeling*, *Evaluation*, dan *Deployment*. Berikut adalah rincian dari setiap fase yang dilakukan dalam penelitian ini:

a. *Business Understanding*

Masalah pencemaran udara menjadi perhatian penting di tengah masyarakat karena berdampak buruk bagi kesehatan manusia maupun kelestarian lingkungan. Di Indonesia, kualitas udara dinilai menggunakan Indeks Standar Pencemar Udara (ISPU), yang dihitung berdasarkan konsentrasi beberapa zat seperti partikel halus berukuran 10 mikron (PM10), karbon monoksida (CO), sulfur dioksida (SO₂), ozon (O₃), dan nitrogen dioksida (NO₂). Untuk mengelompokkan kualitas udara sesuai dengan standar ISPU, digunakan algoritma *Convolutional Neural Network* (CNN) sebagai metode klasifikasi.

b. *Data Understanding*

Dataset yang dianalisis bersumber dari Satu Data Jakarta, mencakup rentang waktu antara tahun 2010 hingga 2021. Sebanyak 4383 data diambil dari Stasiun Indeks Standar Pencemar Udara (ISPU) yang berlokasi di Bundaran HI. Data ini terdiri dari beberapa variabel, yaitu tanggal, stasiun, pm10, so2, co, o3, no2, max, critical, kategori, dan pm25 sesuai yang dideskripsikan pada Tabel 1.

Tabel 1. Variabel *Dataset*

Variabel	Deskripsi
tanggal	Tanggal ketika pengukuran Indeks Kualitas Udara (IKU) dicatat.
stasiun	Nama atau pengenal stasiun pemantauan tempat pengukuran dilakukan.
pm10	Konsentrasi partikulat dengan diameter 10 mikrometer atau kurang (PM10), diukur dalam mikrogram per meter kubik (µg/m ³).
so2	Konsentrasi sulfur dioksida (SO ₂), diukur dalam parts per million (ppm).
co	Konsentrasi karbon monoksida (CO), diukur dalam parts per million (ppm).
o3	Konsentrasi ozon (O ₃), diukur dalam parts per million (ppm).
no2	Konsentrasi nitrogen dioksida (NO ₂), diukur dalam parts per million (ppm).
max	Nilai maksimum yang tercatat di antara semua polutan pada tanggal dan stasiun tersebut.
critical	Polutan yang memiliki konsentrasi tertinggi pada tanggal dan stasiun tersebut.
kategori	Kategori kualitas udara berdasarkan nilai 'Max' yang menggambarkan tingkat kualitas udara.
pm25	Konsentrasi partikulat dengan diameter 2,5 mikrometer atau kurang (PM2.5), diukur dalam mikrogram per meter kubik (µg/m ³).

Berdasarkan deskripsi pada Tabel 1, terlihat bahwa *dataset* ini mengandung informasi konsentrasi untuk berbagai polutan serta kategori kualitas udaranya. Untuk membangun model klasifikasi, langkah selanjutnya adalah menentukan variabel mana yang akan berfungsi sebagai fitur *input* (prediktor) dan variabel mana yang menjadi target (label).

Dari keseluruhan variabel tersebut, ditentukan variabel yang berperan sebagai fitur (X) dan variabel yang digunakan sebagai label atau target (Y) untuk keperluan analisis, sebagaimana ada dalam Tabel 2.

Tabel 2. *Data Understanding*

Features (X)	Target (Y)
pm10, so2, co, o3, no2	kategori

Sebagaimana ditunjukkan pada Tabel 2, lima parameter polutan utama yaitu pm10, so2, co, o3, dan no2 ditetapkan sebagai fitur (X). Kelima variabel ini merupakan *input* standar yang digunakan untuk perhitungan ISPU di

Indonesia. Sementara itu, variabel kategori dipilih sebagai target (Y) karena merepresentasikan kelas kualitas udara ('Baik', 'Sedang', 'Tidak Sehat') yang menjadi tujuan dari penelitian klasifikasi ini.

c. *Data Preparation*

Dalam *dataset* tersebut, terdapat 110 entri dengan label kategori “TIDAK ADA DATA”. Setelah itu, dilakukan pemeriksaan terhadap adanya nilai yang hilang (*missing values*) pada data yang dapat dilihat pada Tabel 3. Tabel 3 di bawah ini merangkum jumlah nilai yang hilang untuk setiap variabel fitur yang akan digunakan.

Tabel 3. Variabe *Dataset*

Variabel	Jumlah Data
pm10	102
so2	65
co	40
o3	93
no2	83
kategori	0

Dari Tabel 3, terlihat bahwa setiap variabel fitur (pm10, so2, co, o3, no2) memiliki sejumlah nilai yang hilang, sementara variabel target (kategori) tidak memiliki data kosong. Keberadaan *missing values* ini, bersama dengan data berlabel “TIDAK ADA DATA”, perlu ditangani pada tahap pembersihan agar tidak mengganggu kinerja model.

Selanjutnya, dilakukan proses pembersihan data dengan menghapus seluruh baris yang mengandung *missing values* dan baris dengan label TIDAK ADA DATA. Perbandingan jumlah data untuk setiap kategori sebelum dan sesudah proses pembersihan disajikan pada Tabel 4 dan Tabel 5.

Tabel 4. Data Sebelum Dilakukan Pembersihan

Categori	Jumlah Data
SEDANG	3065
BAIK	1054
TIDAK SEHAT	154
TIDAK ADA DATA	110

Sebagaimana dirangkum pada Tabel 4, distribusi awal *dataset* menunjukkan adanya 110 entri dengan kategori "TIDAK ADA DATA". Kategori tersebut tidak memberikan informasi yang relevan untuk tujuan klasifikasi dan perlu ditangani. Selain itu, tabel ini menyajikan jumlah data mentah untuk setiap kategori sebelum proses pembersihan dilakukan. Langkah selanjutnya adalah menghapus data yang tidak relevan ini serta baris-baris lain yang mengandung *missing values* untuk memastikan integritas *dataset*. Setelah data yang tidak relevan dan tidak lengkap dihapus, distribusi data yang bersih disajikan pada Tabel 5.

Tabel 5. Data Setelah Dilakukan Pembersihan

Categori	Jumlah Data
SEDANG	2877
BAIK	928
TIDAK SEHAT	145

Langkah ini diambil untuk memastikan keakuratan data tetap terjaga. Penghapusan baris dianggap efektif karena jumlah data yang mengandung *missing values* relatif kecil dibandingkan total *dataset*. Tabel 5 menunjukkan bahwa distribusi data di setiap kategori tidak seimbang. Akibatnya, model berpotensi menjadi condong dan lebih sering memprediksi kategori 'sedang' yang memiliki jumlah data paling banyak. Oleh karena itu, untuk menanggulangi isu ketidakseimbangan distribusi data, umumnya dapat ditangani menggunakan *resampling* data. Penelitian ini mengimplementasikan *Synthetic Minority Oversampling Technique* (SMOTE) dan *Edited Nearest Neighbor* (ENN) dalam mengatasi ketidakseimbangan data [21].

Metode SMOTE dilakukan dengan mencari tetangga terdekat dari sebuah fitur menggunakan pengukuran *Euclidean Distance*. Berdasarkan titik terdekat yang ditemukan, metode ini kemudian menciptakan sebuah sampel data sintesis baru [22]. Metode ENN adalah sebuah pendekatan *undersampling* yang dirancang untuk membersihkan kelas mayoritas. Caranya adalah dengan mengeliminasi sampel-sampel yang dianggap sebagai *noise* atau pengganggu. Penentuan sebuah sampel sebagai *noise* dilakukan menggunakan aturan tetangga terdekat: jika tetangga terdekat dari sebuah sampel mayoritas (yang diukur berdasarkan jarak) ternyata memiliki label kelas yang berbeda, maka sampel mayoritas tersebut akan dihapus [23]. Kombinasi SMOTE-ENN membuat distribusi data yang timpang dapat diatasi, sehingga model dapat menjadi lebih andal dan efektif dalam mendeteksi pola-pola yang terdapat pada kelas minoritas [24].

Resampling data dengan SMOTE dan ENN diawali dengan proses *Oversampling* yang dilakukan pada data dengan kategori “BAIK” dan “TIDAK SEHAT”. Tabel 6 menunjukkan distribusi data sebelum proses, sementara Tabel 7 menampilkan hasilnya setelah *oversampling*.

Tabel 6. Data Sebelum Dilakukan *Oversampling*

Categori	Jumlah Data
SEDANG	2877
BAIK	928
TIDAK SEHAT	145

Berdasarkan Tabel 6, terlihat jelas kondisi ketidakseimbangan data di mana kelas “BAIK” dan “TIDAK SEHAT” merupakan kelas minoritas. Langkah selanjutnya adalah menerapkan SMOTE untuk menyeimbangkan distribusi ini. Distribusi data yang baru dan seimbang ditunjukkan pada Tabel 7.

Tabel 7. Hasil *Oversampling* Data Dengan SMOTE

Categori	Jumlah Data
SEDANG	2877
BAIK	2877
TIDAK SEHAT	2877

Seperti yang ditunjukkan pada Tabel 7, setelah penerapan SMOTE, jumlah data pada setiap kategori kini menjadi seimbang. Metode ini berhasil membuat sampel sintetis untuk kelas “BAIK” dan “TIDAK SEHAT” hingga jumlahnya setara dengan kelas mayoritas SEDANG, yaitu 2877.

Setelah dilakukan *oversampling* menggunakan metode SMOTE, kemudian dilanjutkan dengan pembersihan “noise” pada seluruh data menggunakan ENN. Hasil dari penerapan metode ENN dapat dilihat pada Tabel 8.

Tabel 8. Hasil *Undersampling* Data Dengan ENN

Categori	Jumlah Data
SEDANG	2860
BAIK	2797
TIDAK SEHAT	2638

Tabel 8 menunjukkan distribusi data *final* setelah melalui tahap pembersihan ENN. Terjadi sedikit pengurangan jumlah data pada setiap kelas karena ENN berhasil mengeliminasi sampel-sampel yang dianggap *noise* atau ambigu. Hasilnya adalah *dataset* yang tidak hanya seimbang tetapi juga lebih bersih, sehingga siap untuk dibagi menjadi data latih dan data uji.

Setelah tahap *preprocessing* dan *resampling* data selesai, data dibagi menjadi dua bagian, yaitu data pelatihan dan data pengujian, dengan proporsi label yang tetap seimbang. Pembagian ini menggunakan metode *Stratified Shuffle Split* (atau *Stratified Random Sampling*) yang diimplementasikan melalui pustaka *Scikit-Learn*. *Dataset* dibagi dengan komposisi 80% untuk pelatihan dan 20% untuk pengujian, serta menggunakan *random state* sebesar 42.

d. Modeling

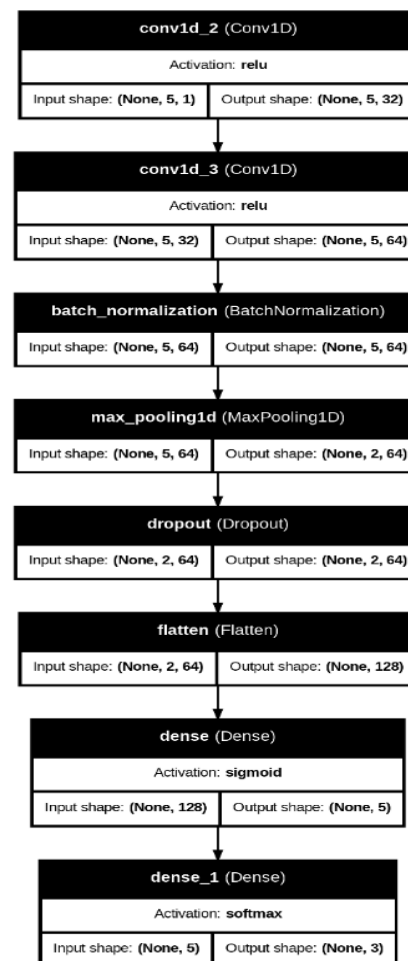
Convolutional Neural Networks (CNN) merupakan salah satu jenis arsitektur jaringan saraf tiruan yang awalnya populer untuk tugas pengenalan citra. Meskipun demikian, arsitektur ini, khususnya varian 1-Dimensi (1D-CNN), sangat efektif untuk diadaptasikan pada analisis data time-series. Kemampuan 1D-CNN untuk menerapkan operasi konvolusi secara sekuensial memungkinkannya mengekstraksi fitur-fitur lokal dari pola temporal dalam data, seperti mengenali perubahan atau pola berulang dalam interval waktu pendek. Sebuah penelitian yang menggunakan 1D-CNN untuk prediksi daya listrik berbasis *time-series* [25]. Struktur 1D-CNN umumnya terdiri dari *input layer*, diikuti oleh beberapa lapisan konvolusi (*Convolutional Layer*) dan *pooling layer* yang bertugas menangkap fitur penting sambil mengurangi dimensi data. Hasil dari lapisan-lapisan ini kemudian diteruskan ke *dense layer* untuk menghasilkan prediksi akhir. Efisiensi dan kemampuannya dalam menangkap pola jangka pendek menjadikan 1D-CNN sebagai model yang sangat relevan untuk berbagai aplikasi prediksi berbasis waktu, termasuk dalam konteks kualitas udara.

e. Evaluation

Setelah model selesai dibangun, tahap selanjutnya adalah evaluasi untuk menilai kualitas performa model. Salah satu metode yang efektif dalam menganalisis performa *multi-class classifier*, dalam hal ini untuk kategori BAIK, SEDANG, dan TIDAK SEHAT adalah dengan menggunakan *Confusion Matrix* dua dimensi yang menggambarkan distribusi kesalahan prediksi atau *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN) [26]. Dari *Confusion Matrix* ini dapat dihitung berbagai metrik evaluasi, seperti *accuracy* (mengukur sejauh mana model klasifikasi mencocokkan hasilnya dengan nilai sebenarnya), *precision* (ketepatan prediksi untuk setiap kelas positif), *recall* (kemampuan model dalam mengidentifikasi seluruh *instance* dari kelas positif), dan *F1-score* (rata-rata harmonik antara *precision* dan *recall*) untuk masing-masing kelas [27]. Selain itu, nilai *micro*, *macro*, dan *weighted average* dari ketiga kelas digunakan untuk memberikan gambaran menyeluruh tentang kinerja model. Perhitungan seluruh metrik tersebut dilakukan dengan menggunakan pustaka *Scikit-Learn*, khususnya fungsi *confusion_matrix* dan *classification_report*.

3. HASIL DAN PEMBAHASAN

Model CNN yang dirancang ini terdiri dari blok utama berisi lapisan konvolusi (*Convolutional Layer*) dan lapisan *pooling* yang juga dilengkapi lapisan tambahan *BatchNormalization* dan *Dropout* untuk meningkatkan performa, serta diakhiri dengan blok lapisan *fully connected* untuk melakukan klasifikasi. Arsitektur model disusun berdasarkan urutan lapisan yang ditunjukkan pada Gambar 2. Struktur ini dirancang untuk menangkap pola penting dalam data dan melakukan klasifikasi dengan akurasi yang optimal.



Gambar 2. Susunan *Layer* Model CNN

3.1 Implementasi

Proses pembangunan model CNN diimplementasikan menggunakan bahasa pemrograman Python dengan *dataset* yang telah melalui tahap *preprocessing*. Atribut yang digunakan sebagai *input* model meliputi ‘pm10’, ‘so2’, ‘co’, ‘o3’, dan ‘no2’, sementara label targetnya adalah ‘kategori’. Adapun tahapan inti dari pembentukan model CNN dijelaskan sebagai berikut:

- Proses dimulai dengan menginisialisasi model secara sekuensial, yang berarti setiap lapisan dalam arsitektur akan disusun secara berurutan dari *input* hingga *output*.
- Lapisan pertama adalah lapisan konvolusi satu dimensi (*Conv1D*) yang menerima data *input* dengan bentuk $(None, 5, 1)$. Lapisan ini memiliki 32 filter dan menggunakan fungsi aktivasi *ReLU*. Lapisan ini diikuti oleh lapisan *Conv1D* kedua dengan 64 filter dan fungsi aktivasi *ReLU*. Tujuan dari kedua lapisan konvolusi ini adalah untuk mengekstraksi fitur secara hierarkis dari data deret waktu atau sekuensial.
- Setelah ekstraksi fitur oleh lapisan konvolusi, diterapkan *Batch Normalization* untuk menstabilkan dan mempercepat proses pelatihan. Selanjutnya, dilakukan *Max Pooling 1D* yang mereduksi dimensi fitur dari 5 menjadi 2, bertujuan untuk mengambil fitur yang paling dominan dan mengurangi komputasi. Setelah itu, lapisan *Dropout* ditambahkan untuk mencegah *overfitting* dengan menonaktifkan beberapa *neuron* secara acak selama pelatihan.
- Hasil dari lapisan sebelumnya kemudian diratakan (*flatten*) menggunakan lapisan *Flatten*, yang mengubah data dari matriks 2D $(2, 64)$ menjadi vektor satu dimensi dengan ukuran 128. Vektor ini kemudian menjadi *input* untuk lapisan *fully connected*.

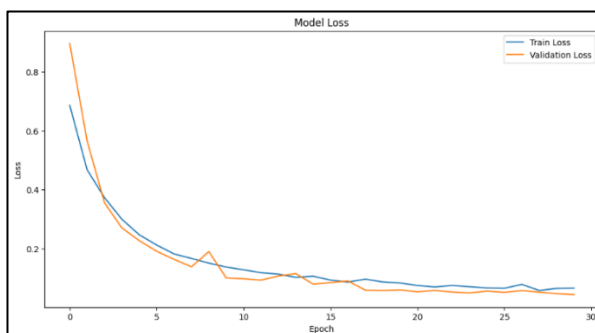
- e. Arsitektur ini memiliki dua lapisan *fully connected (Dense)*. Lapisan *Dense* pertama memiliki 5 *neuron* dengan fungsi aktivasi sigmoid. Lapisan terakhir atau lapisan *output* adalah lapisan *Dense* dengan 3 *neuron* dan fungsi aktivasi *SoftMax*, yang berfungsi untuk melakukan klasifikasi multi-kelas dengan menghasilkan distribusi probabilitas untuk 3 kelas yang berbeda.
- f. Untuk optimasi, digunakan algoritma Adam dengan *learning rate* sebesar 0.001. Fungsi *loss* yang digunakan adalah *sparse_categorical_crossentropy*, yang cocok untuk kasus klasifikasi multi-kelas di mana label target berupa bilangan bulat (*integer*).
- g. Model dilatih selama 30 *epoch* dengan ukuran *batch* 32. Evaluasi model dilakukan pada data validasi, yang diambil sebanyak 20% dari data pelatihan.

3.2 Evaluasi

Setelah model dirancang dan dilatih, dilakukan serangkaian evaluasi untuk mengukur kinerja dan kelayakannya. Evaluasi ini mencakup analisis proses pembelajaran model, pengukuran metrik performa klasifikasi, perbandingan dengan model dasar, serta uji coba fungsional untuk demonstrasi praktis.

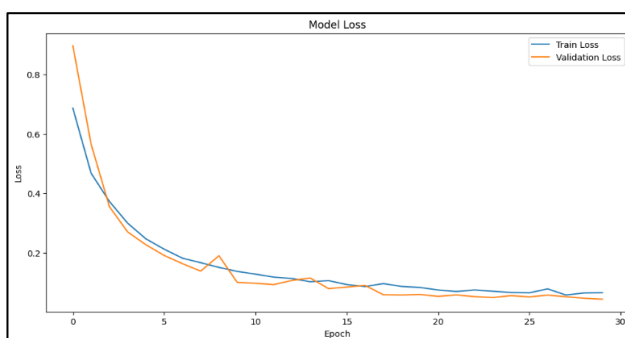
3.2.1 Kinerja Klasifikasi Model Dengan Resampling

Dalam evaluasi kelayakan model, dilakukan pemantauan terhadap nilai *training loss* dan *validation loss* yang diperoleh dengan pembagian data 80% untuk pelatihan dan 20% untuk validasi. Selain itu, diperhatikan juga sebaran nilai *training accuracy* dan *validation accuracy*. Hasil evaluasi menunjukkan bahwa model mampu belajar dengan baik tanpa mengalami *overfitting* maupun *underfitting*. Temuan ini didukung oleh grafik kurva pembelajaran. Visualisasi dari pergerakan nilai *loss* selama proses pelatihan dan validasi disajikan pada Gambar 3.



Gambar 3. Plot Curve Model Loss

Grafik pada Gambar 3 memvisualisasikan penurunan nilai *loss* pada data pelatihan dan data validasi di setiap epoch. Terlihat bahwa kedua kurva, baik *training loss* maupun *validation loss*, menunjukkan tren penurunan yang konsisten dan kemudian stabil di nilai yang rendah. Hal yang terpenting, kedua kurva bergerak beriringan dan tidak menunjukkan adanya *gap* yang signifikan, yang mengonfirmasi bahwa model tidak mengalami *overfitting* dan proses pelatihan berjalan dengan stabil. Sementara itu, grafik peningkatan akurasi model pada data pelatihan dan validasi dapat dilihat pada Gambar 4.



Gambar 4. Plot Curve Model Accuracy

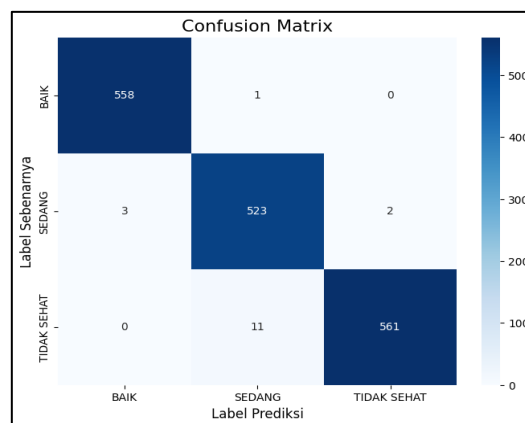
Grafik pada Gambar 4 melengkapi analisis dengan menunjukkan peningkatan akurasi pada data pelatihan dan validasi. Sejalan dengan penurunan *loss*, kedua kurva akurasi ini naik secara konsisten dan mencapai tingkat yang sangat tinggi. Konvergensi antara akurasi pelatihan dan validasi ini semakin memperkuat kesimpulan bahwa model yang dibangun memiliki kemampuan generalisasi yang sangat baik dan tidak *overfit*.

Selain itu, dilakukan evaluasi performa model menggunakan metrik klasifikasi yang lebih mendetail untuk menilai bagaimana model melakukan prediksi terhadap setiap kelas. Laporan klasifikasi yang merinci metrik performa seperti *precision*, *recall*, dan *F1-score* untuk setiap kelas disajikan pada Gambar 5.

Classification Report:				
	precision	recall	f1-score	support
BAIK	0.9947	0.9982	0.9964	559
SEDANG	0.9776	0.9905	0.9840	528
TIDAK SEHAT	0.9964	0.9808	0.9885	572
accuracy			0.9898	1659
macro avg	0.9896	0.9898	0.9897	1659
weighted avg	0.9898	0.9898	0.9898	1659

Gambar 5. Classification Report

Untuk melihat distribusi hasil prediksi benar dan salah secara visual, confusion matrix dari model disajikan pada Gambar 6.



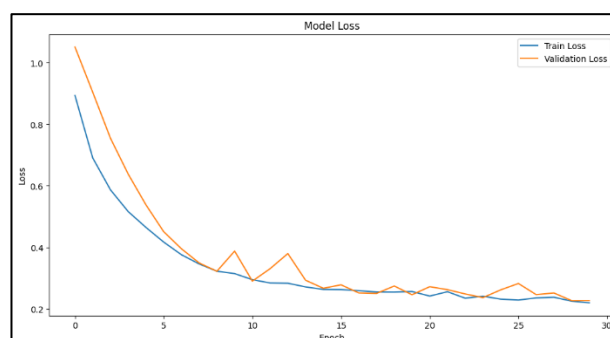
Gambar 6. Confusion Matrix

Model yang dibangun berhasil mencapai tingkat akurasi sebesar 98.98%, seperti yang dirangkum pada Gambar 5. Hal ini juga tercermin pada *Confusion Matrix* di Gambar 6, di mana model dengan benar mengklasifikasikan 558 dari 559 sampel untuk kelas 'BAIK'. Selanjutnya, model mengklasifikasikan dengan tepat 523 dari total 528 sampel untuk kelas 'SEDANG'. Untuk kelas 'TIDAK SEHAT', model berhasil mengklasifikasikan 561 dari total 572 sampel dengan benar.

Berdasarkan *Classification Report* pada Gambar 5, terlihat bahwa model memiliki performa yang sangat tinggi di setiap kelas. Kelas 'BAIK' dan 'TIDAK SEHAT' menunjukkan presisi dan *F1-score* yang nyaris sempurna, masing-masing dengan *F1-score* 0.9964 dan 0.9885. Performa sedikit lebih rendah tercatat pada kelas 'SEDANG' yang memiliki presisi 0.9776 dan *F1-score* 0.9840. Hal ini dapat dianalisis lebih lanjut melalui *Confusion Matrix* pada Gambar 6, yang menunjukkan bahwa sebagian besar kesalahan klasifikasi terjadi pada kelas ini. Kesalahan utama model adalah salah mengklasifikasikan 11 sampel 'TIDAK SEHAT' sebagai 'SEDANG'. Selain itu, beberapa sampel 'SEDANG' juga keliru diprediksi sebagai 'BAIK' (3 sampel) dan 'TIDAK SEHAT' (2 sampel). Meskipun demikian, kesalahan ini tergolong sangat kecil dalam skala keseluruhan, yang dibuktikan dengan nilai *weighted average F1-score* yang mencapai 0.9898, menandakan bahwa model sangat andal dan seimbang dalam melakukan klasifikasi.

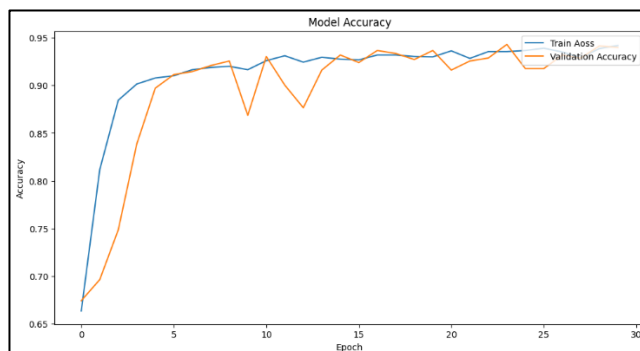
3.2.2 Perbandingan dengan Model Dasar (Tanpa Resampling)

Sebagai pembandingan, proses pemodelan juga dilakukan pada *dataset* awal yang tidak seimbang. Untuk mengevaluasi model dasar ini, langkah pertama adalah menganalisis kurva pembelajarannya. Grafik pergerakan nilai *loss* selama proses pelatihan dan validasi disajikan pada Gambar 7.



Gambar 7. Plot Curve Model Loss Dengan Dataset Tidak Di-Resampling

Grafik pada Gambar 7 menunjukkan proses pelatihan yang stabil, di mana kurva *training* dan *validation loss* sama-sama menurun dan konvergen. Hal ini menandakan tidak adanya *overfitting*, meskipun kurva ini belum tentu menjamin performa yang merata pada setiap kelas mengingat *dataset* yang digunakan tidak seimbang. Sejalan dengan analisis *loss*, peningkatan akurasi model pada data latih dan validasi dapat diamati pada Gambar 8.



Gambar 8. Plot Curve Model Accuracy Dengan Dataset Tidak Di-Resampling

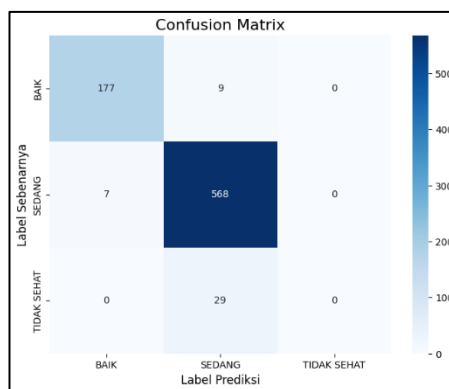
Gambar 8 melengkapi analisis dengan menunjukkan kurva akurasi. Grafik ini mengonfirmasi bahwa model mencapai tingkat akurasi yang tinggi secara stabil, di mana kurva akurasi validasi bergerak beriringan dengan kurva akurasi pelatihan. Hal ini memperkuat temuan bahwa model tidak mengalami *overfitting*.

Penilaian performa model juga dilakukan pada *dataset* yang tidak di-resampling. Hasil evaluasi model tersebut dapat dilihat pada Gambar 9 yang menunjukkan *Classification Report*.

Classification Report:				
	precision	recall	f1-score	support
BAIK	0.9620	0.9516	0.9568	186
SEDANG	0.9373	0.9878	0.9619	575
TIDAK SEHAT	0.0000	0.0000	0.0000	29
accuracy			0.9430	790
macro avg	0.6331	0.6465	0.6396	790
weighted avg	0.9087	0.9430	0.9254	790

Gambar 9. Classification Report Dengan Dataset Tidak Di-Resampling

Laporan klasifikasi pada Gambar 9 menunjukkan akurasi keseluruhan sebesar 94,30%. Meskipun angka ini terlihat tinggi, metrik *recall* untuk kelas 'Tidak Sehat' bernilai 0.00, yang mengindikasikan bahwa model gagal total dalam mengidentifikasi kelas minoritas tersebut. Selain itu, distribusi visual dari hasil prediksi dan kesalahan klasifikasi untuk model ini dapat dilihat secara rinci pada confusion matrix di Gambar 10.



Gambar 10. Confusion Matrix Dengan Dataset Tidak Di-Resampling

Proses pelatihan model berjalan stabil tanpa adanya *overfitting* atau *underfitting*, tetapi evaluasi model menghasilkan akurasi sebesar 94,30%, yang mana lebih rendah dari model pembandingan dengan data *resampling*. Rincian dari *Confusion Matrix* pada Gambar 10 memperjelas bahwa performa model tidak merata dengan kelas ‘Baik’ dan ‘Sedang’ memiliki tingkat keberhasilan prediksi yang tinggi (masing-masing memprediksi benar 177 dari 186 untuk kelas ‘Baik’ dan memprediksi benar 568 dari 575 untuk kelas ‘Sedang’) Namun, kelemahan signifikan teridentifikasi pada ketidakmampuan model untuk mengenali seluruh 29 sampel ‘Tidak Sehat’.

Pelatihan model dengan dua skenario data di-*resampling* dan tidak di-*resampling* menghasilkan perbedaan akurasi dan *loss*. Perbandingan antara hasil pelatihan model dapat dilihat lebih jelas pada Tabel 9.

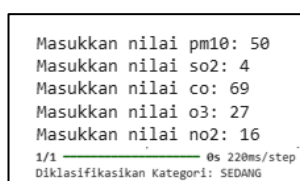
Tabel 9. Perbandingan Hasil Pelatihan Model

Teknik	Accuracy	Loss
Dataset Tidak Resampling	0.9430	0.2153
Resampling SMOTE + ENN	0.9898	0.0437

Tabel 9 menunjukkan perbandingan performa akhir antara kedua model. Hasil tersebut membuktikan keunggulan signifikan dari model yang dilatih menggunakan data hasil resampling SMOTE + ENN. Model ini tidak hanya mencapai akurasi yang lebih tinggi (0.9898 vs 0.9430), tetapi juga menghasilkan nilai *loss* yang jauh lebih rendah (0.0437 vs 0.2153). Perbedaan ini mengonfirmasi bahwa penanganan data tidak seimbang merupakan langkah krusial yang berhasil meningkatkan kemampuan model untuk belajar dan melakukan klasifikasi secara lebih efektif dan andal.

3.2.3 Uji Coba Fungsional

Setelah divalidasi memiliki performa unggul dalam mengidentifikasi berbagai kategori, model yang dilatih dengan data hasil *resampling* kemudian disimpan dalam format h5. Penyimpanan ini bertujuan untuk mempermudah proses uji coba klasifikasi, sebagaimana diilustrasikan pada Gambar 11.

**Gambar 11.** Hasil Percobaan *Input* Parameter Untuk Klasifikasi Kategori

Untuk mendemonstrasikan fungsionalitas model secara praktis, dilakukan sebuah uji coba klasifikasi tunggal. Seperti terlihat pada Gambar 11, sistem meminta *input* untuk lima parameter, dimana nilai yang dimasukkan adalah PM10 sebesar 50, SO₂ sebesar 4, CO sebesar 69, O₃ sebesar 27, dan NO₂ sebesar 16. Setelah data diproses, yang memakan waktu sekitar 220 milidetik, model memberikan hasil klasifikasi yaitu kategori 'SEDANG'. Uji coba ini mengonfirmasi bahwa model mampu menerima data baru dan menghasilkan hasil klasifikasi kategori kualitas udara secara cepat dan akurat.

3.3 Evaluasi Praktis dan Potensial Implementasi

Model CNN yang dioptimalkan dengan SMOTE+ENN menunjukkan performa yang tinggi dengan akurasi 98,98%, melampaui model pada data tidak seimbang maupun metode konvensional. Meskipun arsitektur model ini lebih kompleks, analisis *trade-off* antara akurasi dan kecepatan komputasi terbukti sangat menguntungkan. Dengan waktu prediksi tunggal hanya sekitar 220 milidetik, model ini berhasil memberikan peningkatan akurasi yang signifikan tanpa mengorbankan kecepatan yang esensial untuk aplikasi praktis, sehingga menegaskan kelayakannya untuk implementasi di dunia nyata.

Dengan akurasi tinggi dan waktu prediksi yang cepat, model yang dikembangkan ini memiliki potensi implementasi yang luas dalam sistem nyata. Model ini dapat diintegrasikan ke dalam aplikasi publik sebagai sistem peringatan dini yang memberikan klasifikasi kualitas udara secara real-time kepada masyarakat. Selain itu, bagi instansi pemerintah, model ini dapat berfungsi sebagai alat bantu pengambilan kebijakan dengan mensimulasikan dampak dari berbagai skenario kebijakan lingkungan.

Meskipun model menunjukkan performa akurasi yang sangat tinggi pada data validasi, penting untuk mengakui adanya keterbatasan generalisasi saat dihadapkan pada data *real-world* yang sepenuhnya baru. Model ini dilatih secara spesifik menggunakan data historis dari satu lokasi, yaitu stasiun pemantauan Bundaran HI di Jakarta. Karakteristik polusi udara dapat sangat bervariasi antar wilayah karena perbedaan sumber emisi, kondisi geografis, dan pola cuaca. Oleh karena itu, performa model ini perlu diuji lebih lanjut pada *dataset* dari stasiun pemantauan lain untuk memvalidasi kemampuannya dalam melakukan generalisasi. Tanpa validasi eksternal ini, ada kemungkinan model akan mengalami penurunan performa jika diterapkan langsung pada data dari lingkungan yang belum pernah ditemuinya.

4. KESIMPULAN

Penelitian ini memberikan kontribusi signifikan dengan menunjukkan efektivitas arsitektur *deep learning* untuk klasifikasi kualitas udara di Indonesia. Kebaruan utama dari penelitian ini terletak pada penerapan kombinasi model *Convolutional Neural Network* satu dimensi (1D-CNN) dengan teknik resampling hibrida SMOTE+ENN untuk mengatasi masalah klasifikasi ISPU yang tidak seimbang. Pendekatan ini terbukti mampu melampaui keterbatasan model pada penelitian sebelumnya yang umumnya masih mengandalkan algoritma *machine learning* konvensional dan belum secara spesifik menangani tantangan ketidakseimbangan data secara efektif. Berdasarkan penelitian yang telah dilakukan, pemodelan kualitas udara menggunakan *Convolutional Neural Network* (CNN) berhasil diterapkan



pada *dataset* dari Stasiun HI Jakarta periode 2010-2021, yang setelah melalui tahap *preprocessing* dan *resampling* berjumlah total 8295 data. *Dataset* tersebut kemudian dibagi menjadi data pelatihan dan pengujian dengan proporsi 80% dan 20%. Dalam prosesnya, dilakukan perbandingan antara model yang dilatih pada *dataset* asli dengan model yang datanya diseimbangkan terlebih dahulu menggunakan teknik *resampling* SMOTE + ENN. Hasil perbandingan menunjukkan bahwa penerapan SMOTE + ENN secara signifikan meningkatkan performa, sehingga model ini dipilih sebagai model *final*. Model terpilih ini menunjukkan kinerja yang sangat unggul dengan berhasil mencapai tingkat akurasi keseluruhan 98,98% dan nilai *loss* yang sangat rendah sebesar 0.0437. Tingkat akurasi yang tinggi ini membuktikan bahwa model mampu mengklasifikasikan indeks kualitas udara sesuai standar ISPU dengan tingkat kesalahan yang sangat minimal di setiap kategori. Meskipun berkinerja sangat baik, beberapa rekomendasi diajukan untuk pengembangan selanjutnya. Disarankan untuk melakukan validasi eksternal menggunakan data dari lokasi lain untuk menguji generalisasi model. Selain itu, akurasi model berpotensi ditingkatkan dengan mengintegrasikan fitur meteorologi seperti suhu dan kecepatan angin. Terakhir, penerapan strategi *continual learning* dapat memastikan model tetap adaptif terhadap tren polusi jangka panjang, sehingga relevansinya terjaga di masa depan.

REFERENCES

- [1] P. Imas Agista, N. Gusdini, and M. Dewi Dyah Maharani, “Analisis Kualitas Udara Dengan Indeks Standar Pencemar Udara (Ispu) Dan Sebaran Kadar Polutannya Di Provinsi Dki Jakarta,” *Sustainable Environmental and Optimizing Industry Journal*, vol. 2, no. 2, pp. 39–57, 2020, doi: <https://doi.org/10.36441/seoi.v2i2.491>.
- [2] A. D. Wiranata, S. Soleman, I. Irwansyah, I. K. Sudaryana, and R. Rizal, “Klasifikasi Data Mining Untuk Menentukan Kualitas Udara Di Provinsi Dki Jakarta Menggunakan Algoritma *K-Nearest Neighbors* (K-NN),” *Infotech: Journal of Technology Information*, vol. 9, no. 1, pp. 95–100, Jun. 2023, doi: [10.37365/jti.v9i1.164](https://doi.org/10.37365/jti.v9i1.164).
- [3] World Health Organization (WHO), “Ambient (outdoor) air pollution.” Accessed: Aug. 02, 2025. [Online]. Available: [https://www.who.int/news-room/fact-sheets/detail/ambient-\(outdoor\)-air-quality-and-health](https://www.who.int/news-room/fact-sheets/detail/ambient-(outdoor)-air-quality-and-health)
- [4] Kementerian Lingkungan Hidup dan Kehutanan, “Indeks Standar Pencemar Udara (ISPU) Sebagai Informasi Mutu Udara Ambien di Indonesia,” DITPPU Kementerian Lingkungan Hidup dan Kehutanan. Accessed: Jul. 24, 2025. [Online]. Available: <https://ditppu.menlhk.go.id/portal/read/indeks-standar-pencemar-udara-ispu-sebagai-informasi-mutu-udara-ambien-di-indonesia>
- [5] N. Julia Putri, M. Awaludin, F. Risyda, A. G. Gani, and U. Dirgantara Marsekal Suryadarma, “Penerapan Algoritma Support Vector Machine (SVM) untuk Klasifikasi Kualitas Udara di Wilayah Halim Perdanakusuma,” *Jurnal Mahasiswa Informatika dan Desain (JURMASIN)*, vol. 1, no. 1, pp. 317–323, 2024, doi: <https://doi.org/10.35968/pgjw6t89>.
- [6] B. Putri Salsabila, P. Belva Cynara Trana Putri, N. Ramadhani, and A. Puspita Sari, “Penerapan Algoritma Naive Bayes Terhadap Kualitas Udara Di Jakarta Dan Rekomendasi Aktivitas Masyarakat,” *Jurnal Mahasiswa Teknik Informatika*, vol. 8, no. 6, 2024, doi: <https://doi.org/10.36040/jati.v8i6.11592>.
- [7] W. M. Sukiman, M. Kallista, Ig. P. Dwi Wibawa, and P. D. Kusuma, “Case Study: Jakarta AQI Classification Using Extreme Learning Machine Method with *Oversampling* Technique,” in *2023 IEEE Asia Pacific Conference on Wireless and Mobile (APWiMob)*, IEEE, Oct. 2023, pp. 225–231. doi: [10.1109/APWiMob59963.2023.10365645](https://doi.org/10.1109/APWiMob59963.2023.10365645).
- [8] A. Ma’u Luthfi and F. Fauzi, “Perbandingan Klasifikasi Random Forest, Support Vector Machines, dan LGBM Pada Klasifikasi Kualitas Udara di Jakarta,” *JUSTINDO (Jurnal Sistem dan Teknologi Informasi Indonesia)*, vol. 9, no. 2, pp. 99–108, Aug. 2024, doi: [10.32528/justindo.v9i2.1912](https://doi.org/10.32528/justindo.v9i2.1912).
- [9] N. Z. Shafira, M. Sayyid Imaduddin, R. Syaifullah, G. Puji, and R. Ni’mah, “Inferensi Fuzzy Mamdani dan Decision Tree Untuk Deteksi Kualitas Udara Kota Jakarta,” *Seminar Nasional Sains Data*, vol. 2024, no. 1, 2024, doi: <https://doi.org/10.33005/senada.v4i1.419>.
- [10] F. F. Taliningsih, Y. N. Fu’adah, S. Rizal, A. Rizal, And M. A. Pramudito, “Sistem Otentikasi Biometrik Berbasis Sinyal EKG Menggunakan *Convolutional Neural Network* 1 Dimensi,” *MIND Journal*, vol. 7, no. 1, pp. 1–10, Jun. 2022, doi: [10.26760/mindjournal.v7i1.1-10](https://doi.org/10.26760/mindjournal.v7i1.1-10).
- [11] A. Rusdy Prasetyo, Sussi, and B. S. Aditya, “ANALISIS PERBANDINGAN ALGORITMA SUPPORT VECTOR MACHINE (SVM) DAN *CONVOLUTIONAL NEURAL NETWORK* (CNN) UNTUK SISTEM DETEKSI KATARAK,” *eProceedings of Engineering*, vol. 3, no. 1, 2023, doi: <https://doi.org/10.51903/juritek.v3i1.604>.
- [12] A. N. Ridho, G. Mellyka, F. Saputra, and A. Padmo Azam Masa, “Implementasi Metode *Convolutional Neural Network* (Cnn) Untuk Klasifikasi Gambar Mobil Dan Motor Menggunakan Keras,” *Jurnal Jambo Digitech*, vol. 1, no. 1, 2024, doi: <https://doi.org/10.31603/komtika.v6i2.8054>.
- [13] Kementerian Lingkungan Hidup dan Kehutanan Republik Indonesia, *Peraturan Menteri Lingkungan Hidup dan Kehutanan tentang Indeks Standar Pencemar Udara*. 2020. Accessed: Aug. 02, 2025. [Online]. Available: https://jdih.menlhk.go.id/new2/uploads/files/P_14_2020_ISPU_menlhk_07302020074834.pdf
- [14] B. Siregar, A. Nur Nasution, and D. Arisandi, “Air Pollution Monitoring System Using Waspote Gases Sensor Board in Wireless Sensor Network,” in *International Conference on Data Science, Artificial Intelligence, and Business Analytics (DATABIA)*, IEEE, 2020. doi: [10.1109/DATABIA50434.2020.9190503](https://doi.org/10.1109/DATABIA50434.2020.9190503).
- [15] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D. J. Inman, “1D *Convolutional Neural Networks* and applications: A survey,” *Mech Syst Signal Process*, vol. 151, Apr. 2021, doi: [10.1016/j.ymsp.2020.107398](https://doi.org/10.1016/j.ymsp.2020.107398).
- [16] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, “A survey of the recent architectures of deep *Convolutional Neural Networks*,” *Artif Intell Rev*, vol. 53, no. 8, pp. 5455–5516, Dec. 2020, doi: [10.1007/s10462-020-09825-6](https://doi.org/10.1007/s10462-020-09825-6).
- [17] C. H. Yang, C. H. Wu, K. H. Luo, H. C. Chang, S. C. Wu, and H. Y. Chuang, “Use of *machine learning* algorithms to determine the relationship between air pollution and cognitive impairment in Taiwan,” *Ecotoxicol Environ Saf*, vol. 284, Oct. 2024, doi: [10.1016/j.ecoenv.2024.116885](https://doi.org/10.1016/j.ecoenv.2024.116885).
- [18] G. Husain *et al.*, “SMOTE vs. SMOTEENN: A Study on the Performance of Resampling Algorithms for Addressing *Class imbalance* in Regression Models,” *Algorithms*, vol. 18, no. 1, Jan. 2025, doi: [10.3390/a18010037](https://doi.org/10.3390/a18010037).



- [19] A. Chahal *et al.*, “Predictive analytics technique based on hybrid sampling to manage unbalanced data in smart cities,” *Heliyon*, vol. 10, no. 24, Dec. 2024, doi: 10.1016/j.heliyon.2024.e39275.
- [20] A. F. Fadhilah, A. Ratna Juwita, Y. E. Wicaksana, and T. Al Mudzakir, “Air Quality Classification Using Naive Bayes Algorithm With SMOTE Technique Based on ISPU Data,” *Jurnal Informatika dan Sains*, vol. 8, no. 1, 2025, doi: <https://doi.org/10.31326/jisa.v8i1.2181>.
- [21] I. Riantika, B. Sartono, and K. Anwar Notodiputro, “Effectiveness of SMOTE-ENN to Reduce Complexity in Classification Model,” *Indonesian Journal of Statistics and Its Applications*, vol. 8, no. 1, pp. 70–82, Jun. 2024, doi: 10.29244/ijisa.v8i1p70-82.
- [22] G. Ayu, N. Lestari, K. Agus, and A. Aryanto, “Peningkatan Akurasi Klasifikasi Kualitas Udara melalui *Oversampling* dengan Metode Support Vector Machine dan Random Forest,” *JURNAL SISTEM DAN INFORMATIKA*, vol. 18, no. 1, Nov. 2023, doi: <https://doi.org/10.30864/jsi.v18i1.596>.
- [23] Z. Darojah, R. Susetyoko, and N. Ramadijanti, “Strategi Penanganan Imbalance Class Pada Model Klasifikasi Penerima Kartu Indonesia Pintar Kuliah Berbasis Neural Network Menggunakan Kombinasi Smote Dan Enn,” vol. 10, no. 2, pp. 457–466, 2023, doi: 10.25126/jtiik.2023106480.
- [24] N. Yogi Aptana, A. Nur Ikhsan, W. Maulana Baihaqi, and C. Raras Ajeng Widiawati, “Perbandingan Random Forest dan *K-nearest neighbors* untuk Klasifikasi Body Mass Index Menggunakan SMOTE-ENN untuk Mengatasi Ketidakseimbangan Data pada Analisis Kesehatan,” vol. 16, no. 01, 2025, doi: 10.35970/infotekmesin.v16i1.2553.
- [25] S. SUKATMO, H. A. NUGROHO, B. H. RUSANTO, and S. SOEKIRNO, “Performance Comparison of 1D-CNN and LSTM *Deep learning* Models for Time Series-Based Electric Power Prediction,” *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, vol. 13, no. 1, p. 44, Feb. 2025, doi: 10.26760/elkomika.v13i1.44.
- [26] N. Syahira and D. B. Arianto, “Prediksi Tingkat Kualitas Udara Dengan Pendekatan Algoritma K-Nearest Neighbor,” *Jurnal Ilmiah Informatika Komputer*, vol. 29, no. 1, pp. 45–59, 2024, doi: 10.35760/ik.2024.v29i1.10069.
- [27] A. A. Ahmed, W. Ali, T. A. A. Abdullah, and S. J. Malebary, “Classifying Cardiac Arrhythmia from ECG Signal Using 1D CNN *Deep learning* Model,” *Mathematics*, vol. 11, no. 3, Feb. 2023, doi: 10.3390/math11030562.