

## Prediksi Kualitas Udara Daerah Tangerang Selatan Melalui Parameter ISPU dengan Metode LSTM

Doni Andriansyah<sup>1</sup>, Elly Mufida<sup>2</sup>, Hylenearti Hertiana<sup>3</sup>

<sup>1,2,3</sup>Universitas Bina SaranaInformatika

Jl, Kramat Raya 98, Indonesia

[doni.dad@bsi.ac.id](mailto:doni.dad@bsi.ac.id)<sup>1</sup>, [elly.elm@bsi.ac.id](mailto:elly.elm@bsi.ac.id)<sup>2</sup>, [hylenearti.hha@bsi.ac.id](mailto:hylenearti.hha@bsi.ac.id)<sup>3</sup>

Artikel Info : Diterima : 15-10-2025 | Direvisi : 23-10-2025 | Disetujui : 14-11-2025

**Abstrak** - Indeks Standar Pencemar Udara (ISPU) adalah angka yang tidak mempunyai satuan yang menggambarkan kondisi mutu udara ambien di suatu lokasi, berdasarkan dampaknya terhadap kesehatan manusia, estetika, serta makhluk hidup lainnya. Tangerang Selatan adalah salah satu kota di Provinsi Banten, Indonesia, yang merupakan bagian dari wilayah Jabodetabek. Tangerang Selatan berkembang pesat sebagai kota urban dan satelit Jakarta, dan menjadi area industri ringan dan komersial. Penelitian ini menggunakan model *Long Short-Term Memory* (LSTM) untuk memprediksi angka ISPU pada kota Tangerang Selatan. Berdasarkan hasil penelitian, model LSTM mampu memprediksi kualitas udara di Tangerang Selatan dengan akurasi yang cukup tinggi. Model berhasil mengikuti pola data historis selama 30 hari dengan baik dan menghasilkan prakiraan 7 hari ke depan yang cukup mendekati nilai aktual. Hal ini ditunjukkan oleh nilai evaluasi yang memuaskan, yaitu MSE sebesar 15,15, RMSE sebesar 3,89, dan MAE sebesar 2,86, mengindikasikan rata-rata kesalahan prediksi yang relatif kecil. Perbedaan antara nilai prediksi dan aktual kualitas udara harian berada dalam rentang yang dapat diterima.

Kata Kunci : Long Short-Term Memory, LSTM, Indeks Standar Polusi Udara

**Abstracts** - The Air Pollutant Standard Index (ISPU) is a number that does not have a unit that describes the condition of ambient air quality in a location, based on its impact on human health, aesthetics, and other living things. South Tangerang is one of the cities in Banten Province, Indonesia, which is part of the Jabodetabek area (Jakarta, Bogor, Depok, Tangerang, Bekasi). South Tangerang is developing rapidly as an urban city and satellite of Jakarta and becoming a light industrial and commercial area. This study uses the Long Short-Term Memory (LSTM) model to predict the ISPU figure in the city of South Tangerang. Based on the results of the study, the LSTM model can predict air quality in South Tangerang with quite high accuracy. The model successfully follows the historical data pattern for 30 days well and produces a 7-day forecast that is quite close to the actual value. This is indicated by the satisfactory evaluation value, namely MSE of 15.15, RMSE of 3.89, and MAE of 2.86, which indicates a relatively small average prediction error. The difference between the predicted and actual daily air quality values (maximum ISPU value) is within an acceptable range.

Keywords : Long Short-Term Memory, LSTM, Air Pollutant Standards Index

### PENDAHULUAN

Berdasarkan Peraturan Menteri Lingkungan Hidup Dan Kehutanan Republik Indonesia Nomor P.14/MENLHK/SETJEN/KUM.1/7/2020 Tentang Indeks Standar Pencemar Udara (ISPU), pada Pasal 1 dan Pasal 2 disebutkan bahwa “Indeks Standar Pencemar Udara yang selanjutnya disingkat ISPU adalah angka yang tidak mempunyai satuan yang menggambarkan kondisi mutu udara ambien di lokasi tertentu, yang didasarkan kepada dampak terhadap kesehatan manusia, nilai estetika dan makhluk hidup lainnya, meliputi parameter: partikulat (PM10), partikulat (PM2.5), karbon monoksida (CO), nitrogen dioksida (NO2), sulfur dioksida (SO2), ozon (O3) dan hidrokarbon (HC). Selanjutnya pada Pasal 7 disebutkan bahwa hasil perhitungan ISPU terdiri atas

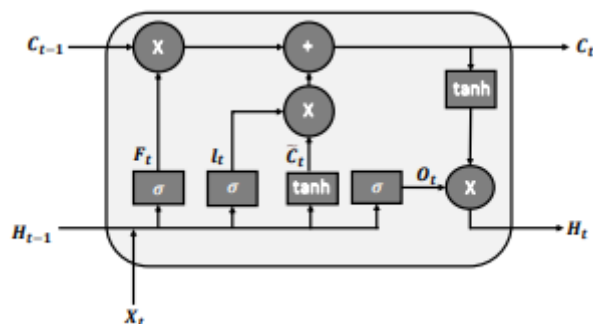


lima kategori, yaitu baik, sedang, tidak sehat, sangat tidak sehat, dan berbahaya. Penelitian terkait ISPU telah banyak dilakukan, diantaranya: Beberapa model yang pernah digunakan pada data ISPU diantaranya *Long Short-Term Memory* (LSTM) (Hidayatullah et al., 2022)(I Gusti Ayu Nandia Lestari & I Nyoman Dwi Arysna Mahendra, 2023), C4.5 (Prafanto et al., 2024), Random Forest (Hasna et al., 2024), Discrete Time Markov Chains (Fahim et al., 2024), Seasonal Autoregressive Integrated Moving Average atau SARIMA (Karyadi, 2022)(Lestariningati et al., 2013)(Putra et al., 2024), Markov Modeling (Zayed & Abbod, 2024).

Berdasarkan laporan kualitas udara dunia tahunan ke-6 yang dirilis oleh IQAir, perusahaan teknologi kualitas udara asal Swiss, Tangerang Selatan menjadi kota dengan kualitas udara terburuk di Indonesia sepanjang tahun 2023 lebih buruk dari Jakarta hingga Palembang (Tim/dmi, 2024). Tangerang Selatan sebagai kota termuda di Jabodetabek, berkembang pesat dari kawasan pinggiran menjadi pusat bisnis, pendidikan, dan hunian modern. Meskipun mengalami perkembangan pesat, Tangerang Selatan juga menghadapi beberapa tantangan, antara lain Kemacetan, Kepadatan Penduduk, serta Masalah Lingkungan terkait sampah, air bersih dan kualitas udara. (Riansyah, 2025). Pemerintah Kota Tangerang Selatan masih berusaha keras untuk memperbaiki kualitas udara yang masih memburuk dengan sejumlah langkah cepat Salah satunya adalah gencarnya penanaman pohon secara serentak di setiap kecamatan. Selaian itu dibutuhkan model yang dapat memprediksi kualitas udara dalam satu hari atau satu minggu kedepan secara valid, yang dapat digunakan oleh pemerintah setempat sebagai dasar dalam mengambil langkah mitigasi (Ginting, 2023).

*Recurrent neural network* (RNN) adalah arsitektur jaringan saraf tiruan yang prosesnya dipanggil secara berulang untuk data masukan yang sifatnya sekuensial. RNN merupakan bagian dari deep learning karena pemrosesan data dilakukan melalui banyak layer. LSTM adalah jenis arsitektur pengembangan dari RNN, sehingga terdapat kelebihan yang dimiliki yaitu dapat menyimpan informasi dalam jangka waktu yang panjang dengan menambahkan memory cell. Selain itu, LSTM dapat menjadi solusi saat memproses data sekuensial yang panjang sehingga mengatasi terjadinya vanishing gradient. Permasalahan vanishing gradient merupakan kekurangan dari RNN dalam menangkap long term dependencies yang mengurangi akurasi dari suatu prediksi (Oktaviani & Hustinawati, 2021). LSTM merupakan bagian dari metode RNN. Kelebihan metode LSTM dibandingkan dengan metode RNN yaitu, LSTM dapat mengingat data yang bersifat time series atau data dengan informasi long term dependency dan dapat menyimpan informasi terdahulu menggunakan sel yang terdapat pada LSTM (Lattifia et al., 2022). LSTM juga dapat digunakan untuk masalah klasifikasi, dan peramalan berdasarkan data rentang waktu (Hidayatullah et al., 2022).

Gambar 1 menyajikan isi dari lapisan tersembunyi pada arsitektur LSTM, yaitu sel memori yang tersembunyi. Sebuah sel memori di LSTM menyimpan sebuah nilai atau keadaan (cell state), baik untuk periode waktu yang panjang atau singkat (Wiranda & Sadikin, 2019). Secara umum LSTM terdiri dari tiga gate, yaitu input gate, output gate dan forget gate (I Gusti Ayu Nandia Lestari & I Nyoman Dwi Arysna Mahendra, 2023). Setiap gate pada LSTM berfungsi sebagai filter informasi.



Gambar 1. Lapisan tersembunyi pada arsitektur LSTM

$F_t$  merupakan Forget Gate,  $H_{t-1}$  adalah nilai dari Hidden State sebelumnya,  $X_t$  merupakan data yang diinputkan,  $W$  adalah bobotnya dan  $b$  adalah nilai biasnya.  $I_t$  ini merupakan Input Gate,  $\tilde{C}_t$  adalah nilai kandidat dan  $C_t$  merupakan Cell State. Terakhir,  $O_t$  merupakan nilai dari Output Gate (Hidayatullah et al., 2022). Berikut adalah penjelasan dari Input Gate  $I_t$ , Forget Gate  $F_t$ , dan Output Gate  $O_t$ .

1. Input Gate ( $I_t$ ). Input gate berperan mengambil output sebelumnya dan input baru serta melewati mereka melalui lapisan sigmoid. Gate ini mengembalikan nilai 0 atau 1. Rumus (1) merupakan rumus Input Gate  $I_t$ .

$$I_t = \sigma(W_i S_{t-1} + W_i X_t) \quad (1)$$

Dengan,  $W_i$  = Bobot dari Input Gate.  $S_{t-1}$  = State sebelumnya atau state pada waktu t-1.  $X_t$  = Input pada waktu t.  $\sigma$  = Fungsi aktivasi sigmoid

2. Forget Gate ( $F_t$ ). Forget gate adalah lapisan sigmoid yang mengambil output pada waktu t – 1 dan input pada waktu t dan menggabungkannya serta menerapkan fungsi aktivasi sigmoid. Karena sigmoid, output dari gate ini adalah 0 atau 1. Jika  $F_t = 0$  maka keadaan (state) sebelumnya akan dilupakan, sementara jika  $F_t = 1$  state sebelumnya tidak berubah. Rumus (2) adalah rumus Forgate Gate  $F_t$ .

$$F_t = \sigma(W_f S_{t-1} + W_f X_t) \quad (2)$$

Dengan,  $W_f$  = Bobot dari forget gate;  $S_{t-1}$  = State sebelumnya atau state pada waktu t – 1;  $X_t$  = Input pada waktu t;  $\sigma$  = Fungsi aktivasi sigmoid.

3. Output Gate ( $O_t$ ). Output Gate berfungsi untuk mengontrol keluaran yang dikirim ke langkah waktu berikutnya atau digunakan dalam tugas lainnya. Output gate mengontrol seberapa banyak state yang lewat ke output dan bekerja dengan cara yang sama dengan gate lainnya. Dan terakhir menghasilkan cell state yang baru ( $h_t$ ). Rumus (3) dan (4) adalah rumus  $o_t$  dan  $h_t$  secara berurutan.

$$O_t = \sigma(W_o S_{t-1} + W_o X_t) \quad (3)$$

$$h_t = O_t * \tanh(ct) \quad (4)$$

Dengan,  $W_o$  = Bobot dari output gate.  $S_{t-1}$  = State sebelumnya atau state pada waktu t - 1.  $X_t$  = Input pada waktu t.  $\sigma$  = Fungsi aktivasi sigmoid.

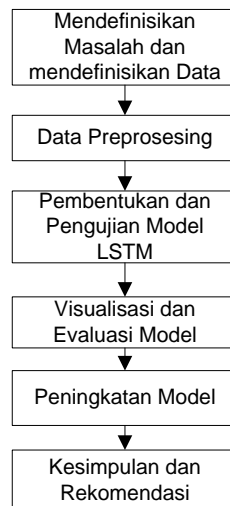
Pada Model *Long Short-Term Memory* (LSTM) terdapat hyper parameter yang cukup penting pada learning model, yaitu: parameter arsitektur (*units, stack LSTM, input shape, output activation function, dropout rate, bidirectional, dan activation function*); parameter connection (*dense layer dan regularization*) parameter *optimizer dan loss function*). Sedangkan parameter yang dilakukan pada training model adalah *learning rate, batch size, epochs, validation split, shuffle, dan callbacks*. Pengujian parameter dilakukan dengan mencoba kombinasi dari yang telah ditentukan untuk mendapat output yang baik. Pengujian dimulai dari mencari partisi data latih dan data uji, pencarian *epoch*, pencarian kombinasi *layer* dan *hidden neuron* dalam setiap *layer* dan terakhir pencarian optimasi (Hidayatullah et al., 2022). Dalam pemodelan dengan metode LSTM dan ANN salah satu parameter yang paling banyak diatur adalah jumlah neuronnnya (I Gusti Ayu Nandia Lestari & I Nyoman Dwi Arysna Mahendra, 2023).

Salah satu tahapan yang paling penting setelah tahapan pembentukan model pada machine learning adalah tahap evaluasi, yang bertujuan untuk mengetahui sejauh mana performa dari model yang dihasilkan (Arkadia et al., 2022). Pengukuran akurasi prediksi dilakukan untuk mengukur seberapa akurat prediksi tersebut dalam memprediksi nilai masa depan (Pangaribuan et al., 2023). Dalam model deret waktu (*time series*), evaluasi dapat dilakukan dengan menghitung *error* yang dihasilkan model (Ashari & Suhendar, 2024), diantaranya menggunakan *Mean Absolute Error* (MAE), *Root Mean Square Error* (RMSE), dan *Mean Absolute Percentage Error* (MAPE) (Agusmawati et al., 2023). Dari beberapa penelitian sebelumnya, Model LSTM dapat menggunakan *loss function* yang digunakan sebagai kriteria dalam mengevaluasi kinerja, yaitu *Root Mean Squared Error* (RMSE), *Mean Absolute Error* (MAE), *Mean Squared Error* (MSE) dan *Mean Absolute Percentage Error* (MAPE) (I Gusti Ayu Nandia Lestari & I Nyoman Dwi Arysna Mahendra, 2023)(Wiranda & Sadikin, 2019). Terdapat metode yang digunakan untuk mengoptimalkan kinerja model melalui evaluasi parameter, diantaranya *Adaptive Gradient* (ADAGRAD), *Adaptive Moment Gradient* (ADAM), *Root Mean Square Error* (RMSProp) (Cahyana et al., 2023)(Arkadia et al., 2022).

Banyak penelitian untuk memprediksi kualitas udara telah dilakukan menggunakan model statistik dan machine learning dengan akurasinya masih bervariasi. *Long Short-Term Memory* (LSTM) sebagai bagian dari deep learning memiliki karakteristik yang dapat digunakan dalam menangani *time series data* seperti kualitas udara, tetapi penerapannya di konteks lokal (khususnya Tangerang Selatan) masih terbatas. Tujuan penelitian ini adalah menganalisis tren dan pola kualitas udara di Tangerang Selatan berdasarkan parameter Indeks Standar Pencemaran Udara (ISPU) atau *Air Pollution Index* (API) dengan menggunakan metode LSTM, menilai akurasi model dan mengidentifikasi parameter paling berpengaruh terhadap prediksi. Penelitian ini fokus pada prediksi ISPU melalui data parameter PM10, SO2, CO, O3, NO2 dari tahun 2020 sampai 2022 yang ada di wilayah Tangerang Selatan.

## METODE PENELITIAN

Gambar 2 adalah bagan metode penelitian yang menunjukkan tahapan atau alur penelitian yang dilakukan oleh penulis. Berikut adalah penjelasa dari setiap tahapan tersebut:



Gambar 2. Metode penelitian yang digunakan oleh penulis

Tahap pertama adalah Mendefinisikan masalah dan mendefinisikan data. Pada tahapan ini, penulis merumuskan permasalahan dan tujuan yang akan diteliti, meliputi identifikasi masalah melalui telaah literatur serta batasan masalah agar tetap fokus pada aspek yang bisa menjawab penelitian. Data yang digunakan pada penelitian diambil dari laman <https://www.kaggle.com/datasets/ourwit/air-quality-in-south-tangerang-indonesia-20-23> pada tanggal 2 Februari 2025. *Dataset* tersebut memiliki pengukuran unsur-unsur dari polusi udara, yaitu Particulate Matter (PM10), Sulfur Dioxide (SO<sub>2</sub>), Carbon Monoxide (CO), Ozone (O<sub>3</sub>), and Natrium Dioxide (NO<sub>2</sub>). Pengukuran tersebut telah dikonversi kedalam *Pollutant Standards Index* (PSI) atau Indeks Standar Pencemaran Udara (ISPU). Dalam *dataset* yang digunakan pada penelitian ini terdapat atribut “Date” yang menandakan *dataset* merupakan time series data dan sebuah atribut target “Max” yang menandakan bahwa data masuk ke dalam supervised learning, serta 8 atribut yang mempengaruhi nilai “Max”.

Tahap kedua adalah *Data Preprocessing*. Tahapan *data preprocessing* dilakukan untuk mendapatkan kondisi terbaik dari *dataset* yang secara langsung akan meningkatkan performa model. Pada tahapan data preprocessing ini penulis melakukan: membersihkan Data (*Data Cleaning*), konversi data, transformasi data (*data transformation*), serta pembentukan data urutan (*sequence*) yang diperlukan bagi *time series data*.

- Membersihkan data (*Data Cleaning*) meliputi menangani *missing values*, duplikasi, dan *outlier*, mengoreksi kesalahan input atau format yang tidak konsisten.
- Mengubah format tanggal ke format yang bisa dibaca model.
- Mengonversi data kategori menjadi numerik dengan metode one-hot encoding.
- Normalisasi atau standarisasi fitur agar berada dalam skala yang sama.
- Menentukan fitur yang akan digunakan pada pembentukan model. Menghapus fitur yang tidak relevan atau sangat tidak berkorelasi agar model tidak *overfitting*.
- Mempersiapkan Data untuk Pelatihan dan Pengujian, yaitu dengan memisahkan data menjadi *training set*, *validation set*, dan *testing set* untuk evaluasi model secara adil.

Tahap ketiga adalah Pembentukan dan Pelatihan Model. Pada tahap ini penulis menggunakan model *Long Short-Term Memory* (LSTM) dimana model tersebut biasanya digunakan untuk memproses data berurutan seperti *time series data*. Penulis menggunakan *framework* TensorFlow/Keras untuk membangun model LSTM. Setelah Model terbentuk, selanjutnya penulis melakukan Evaluasi Model dengan tujuan memberikan informasi seberapa baik model bekerja pada data yang belum pernah dilihat sebelumnya, menghindari *overfitting*, serta keputusan dalam melakukan *tuning hiper parameters*. Pada evaluasi model, penulis menggunakan evaluasi matriks regresi *Means Squared Error* (MSE) dan *Mean Absolute Error* (MAE). MSE adalah kesalahan kuadrat rata-rata antara nilai aktual dan nilai prediksi. Metode MSE sering digunakan untuk menguji estimasi nilai *error* dalam melakukan prediksi, dimana nilai MSE yang rendah atau mendekati nol menunjukkan bahwa hasil prediksi pada model dengan data yang aktual dapat digunakan untuk melakukan prediksi di masa yang akan datang (Sianturi et al., 2023). MAE memberikan gambaran tentang seberapa besar kesalahan rata-rata dari nilai prediksi dibandingkan dengan nilai sebenarnya (Mubarak et al., 2026). Visualisasi model digunakan untuk membantu memahami, menganalisis, dan membuat keputusan yang lebih baik terhadap data dan model. Visualisasi dapat digunakan pada pengamatan distribusi, pola, tren, dan *outlier* dalam data. Evaluasi model akan menentukan dilakukannya tahapan Peningkatan Model. Peningkatan model untuk mendapatkan prediksi yang lebih optimal, dengan mengubah *batch size*. *Batch size* adalah parameter yang menunjukkan jumlah sampel (data) yang digunakan sekaligus dalam satu langkah pembaruan bobot (*weight update*) saat proses pelatihan

(training) model *neural network*. *Batch size* digunakan untuk mengontrol frekuensi *update* bobot, menyeimbangkan akurasi dan efisiensi, mempengaruhi konsumsi memori, serta mempengaruhi generalisasi model. Tahap akhir dari penelitian ini adalah membuat kesimpulan yang merangkum hasil penelitian secara ringkas dan jelas untuk menjawab apakah tujuan penelitian telah tercapai. Sedangkan rekomendasi adalah saran aplikatif atau pengembangan lanjutan pada stakeholder berdasarkan temuan yang diperoleh.

## HASIL DAN PEMBAHASAN

Peningkatan jumlah kendaraan bermotor, pembangunan, dan kegiatan industri yang semakin berkembang di daerah perkotaan khususnya Tangerang Selatan akan berdampak pada semakin memburuk kualitas udara di daerah tersebut. Masyarakat dan pemerintah membutuhkan sistem prediksi yang akurat untuk mengambil langkah mitigasi secara dini. Seiring dengan berkembangnya teknologi pemrograman khususnya pada machine learning, maka diperlukan penelitian berkelanjutan untuk menemukan model optimal dalam meramal kualitas udara di daerah tangerang selatan agar dapat digunakan oleh pemerintah setempat sebagai rekomendasi dalam membuat tindak lanjut untuk mengurangi atau meminimalkan dampak negatif dari menurunnya kualitas udara yang mungkin terjadi di masa depan. Rumusan masalah pada penelitian ini adalah bagaimana pola historis parameter ISPU di Tangerang Selatan, Seberapa baik model LSTM dapat memprediksi nilai ISPU berdasarkan data historis tersebut, dan Parameter pencemar udara apa yang paling berpengaruh terhadap hasil prediksi ISPU. *Dataset* yang digunakan pada penelitian memiliki 9 atribut pengukuran dan 1 atribut kategori. Atribut yang diukur pada *dataset* adalah atribut yang digunakan sebagai indikator ISPU, yaitu: Date, PM2.5, PM10, SO2, CO, O3, NO2, Max, Critical Component, dan Category. Dengan adanya atribut Date menandakan *dataset* merupakan *time series data*.

### 1. Data Preprocessing

*Dataset* yang digunakan pada penelitian ini berisi data kualitas udara di Tangerang Selatan dari tahun 2020-2022 dengan kolom-kolom disajikan pada tabel 1. Dengan adanya kolom Date, maka disimpulkan bahwa yang digunakan merupakan *timeseries data*.

Tabel 1. Atribut pada *dataset*

No	Atribut	Keterangan
1	Date	Tanggal pengukuran
2	PM2.5	Pengukuran partikel halus di udara
3	PM10	Pengukuran partikel materi
4	SO2	Pengukuran sulfur dioksida
5	CO	Pengukuran karbon monoksida
6	O3	Pengukuran ozon
7	NO2	Pengukuran natrium dioksida
8	Max	Nilai pengukuran tertinggi
9	Critical Component	Komponen yang memiliki nilai tertinggi
10	Category	Kategori kualitas udara, baik atau tidak

Untuk membangun model LSTM, penulis menggunakan bahasa pemrograman Python, dengan tahapan preprosesing sebagai berikut:

#### (a). Data Cleaning,

Penulis melakukan penanganan terhadap *missing value* dengan menggunakan metode *ffill* untuk mengisi nilai yang kosong (NaN) dengan nilai dari baris atau kolom sebelumnya. dalam *DataFrame*.

#### (b). Konversi Format Tanggal

Gambar 3 adalah code program untuk mengubah kolom Date menjadi format *datetime* dengan tujuan agar lebih mudah diproses, seperti sorting berdasarkan waktu. membuat indeks waktu (*time index*), atau melakukan operasi *time-series* seperti agregasi data, perhitungan tren, atau visualisasi data berbasis waktu.

```
df['Date'] = pd.to_datetime(df['Date'], format='%m/%d/%Y')
df = df.sort_values('Date')
```

Gambar 3. Code program untuk melakukan konversi format tanggal

#### (c). Seleksi Fitur

Memilih fitur atau kolom yang akan dimasukkan kedalam proses prediksi data. Adapun fitur yang disertakan adalah PM2.5, PM10, SO2, CO, O3, dan NO2.

#### (d). Normalisasi Data

Min-max normalization adalah teknik normalisasi data yang berfungsi untuk mengonversi data asli ke dalam rentang nilai yang baru, dengan tujuan menciptakan keseimbangan antara nilai sebelum dan sesudah proses normalisasi (Ramadhan & Fachrie, 2024). Pada penelitian ini normalisasi data dilakukan pada atribut Max dengan MinMaxscaled (rentang 0–1).

(e) Pembentukan Data Urutan (Sequence)

Gambar 4 adalah code program yang digunakan untuk membentuk Mmdel LSTM, yang memerlukan data dalam format sequence. Terdapat variable SEQ\_LEN yang merupakan rentang data selama 30 hari sebelumnya, dan variable FUTURE\_DAYS untuk menentukan rentang prediksi harian, dalam penelitian penulis memprediksi untuk 7 hari kedepan. Data target terletak pada hari ke 31 s.d hari ke 37.

```
SEQ_LEN = 30
FUTURE_DAYS = 7

def create_sequences(data, seq_length, future_steps):
    X, y = [], []
    for i in range(len(data) - seq_length - future_steps + 1):
        X.append(data[i:i + seq_length])
        y.append(data[i + seq_length:i + seq_length + future_steps])
    return np.array(X), np.array(y)

X, y = create_sequences(scaled_data, SEQ_LEN, FUTURE_DAYS)
```

Gambar 4. Code program untuk membentuk data urutan (sequence)

(f). Pemisahan Data

Pada tahap ini penulis membagi *dataset* menjadi data latih dan data uji, dengan komposisi 80:20. Berikut adalah code program yang digunakan untuk membagi *dataset* tersebut.

2. Pembentukan dan Pelatihan Model LSTM

(a). Pembentukan Model

Gambar 5 adalah code program yang digunakan untuk pembentukan model dengan framework TensorFlow/Keras untuk membangun model LSTM. Berikut adalah parameter yang didefinisikan dalam pembentukan model LSTM.

- **Input Layer:** Sesuaikan dimensi input (e.g., (30, number\_of\_features)).
- **LSTM Layers:** Tambahkan 1-2 lapisan LSTM dengan dropout untuk mencegah overfitting.
- **Dense Layer:** Gunakan lapisan Dense untuk menghasilkan output.

```
model = Sequential([
    LSTM(64, activation='relu', input_shape=(SEQ_LEN, 1)),
    Dense(32, activation='relu'),
    Dense(FUTURE_DAYS)
])

model.compile(optimizer='adam', loss='mse')
model.summary()
```

Gambar 5. Code program dalam menentukan parameter model

(b). Pelatihan Model

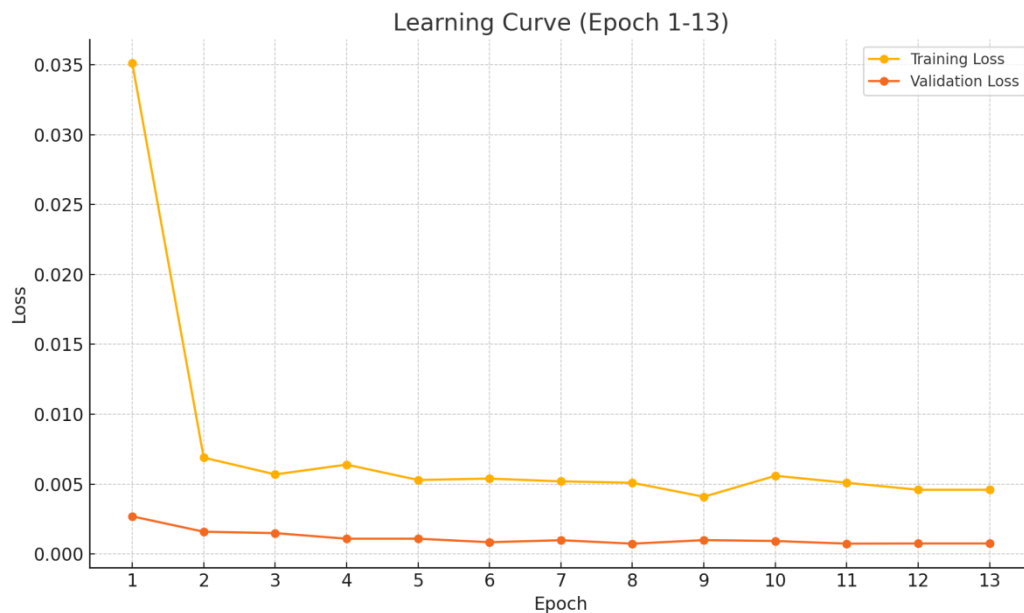
Gambar 6 adalah code program yang digunakan dalam melatih model, penulis menggunakan parameter *epoch*, *batch size*, dan *validation* untuk mendapatkan model *fit* pada data *training*.

```
# Train the model
history = model.fit(X_train, y_train, epochs=50, batch_size=16, validation_split=0.1)
```

Gambar 6. Code Program untuk mendapatkan *Learning Curve* (Epoch 1-13) pada pelatihan model

*Epoch x/50*: Menunjukkan bahwa proses pelatihan terdiri dari 50 *epoch*, dan yang ditampilkan adalah progres dari *epoch* ke-1 hingga ke-13. 48/48 berarti ada 48 *batch* dalam satu *epoch* (jumlah total data dibagi *batch*

size), dan semuanya telah selesai diproses. Time dan time/step, misalnya: 1s 15ms/step menunjukkan bahwa total waktu untuk menyelesaikan 1 epoch adalah 1 detik, dan tiap batch diproses dalam waktu 15 milidetik. Loss adalah error pada training data. Val\_loss adalah error pada validation data. Gambar 7 adalah



visualisasi hasil pelatihan model.

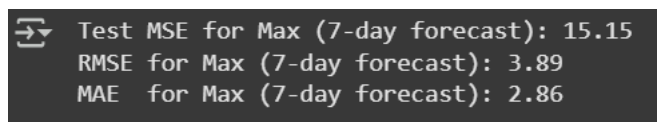
Gambar 3. Learning Curve (Epoch 1-13) pada pelatihan model

Dari hasil pelatihan model, dapat disimpulkan bahwa loss dan val\_loss Menurun dari epoch 1 ke 13, loss turun dari 0.0351 ke 0.0046, dan val\_loss turun dari 0.0027 ke 0.00075957, artinya: Model belajar dengan baik, tidak terjadi overfitting (karena val\_loss juga turun bersama loss), serta Model konsisten meningkat akurasinya terhadap data validasi. Angka val\_loss sangat kecil. menunjukkan bahwa model memiliki kemampuan prediksi yang sangat baik pada data yang tidak dilatih. Dari hasil pelatihan tersebut didapat bahwa model menunjukkan performa yang sangat baik hingga epoch ke-13. Jika tren ini berlanjut hingga epoch ke-50 (tanpa val\_loss naik tajam). Model menunjukkan performa yang sangat baik, dengan loss dan val\_loss yang menurun konsisten hingga nilai sangat kecil. Tidak terlihat tanda overfitting karena val\_loss tetap rendah dan mendekati loss. Pada tahap ini disimpulkan bahwa model memiliki kemampuan generalisasi yang bagus dan layak digunakan untuk prediksi lebih lanjut.

### 3. Evaluasi dan Visualisasi Model

#### (a). Evaluasi Model

Evaluasi model dilakukan dengan menghitung metrik seperti Mean Squared Error (MSE) dan Mean Absolute Error (MAE) pada data pengujian. Setelah model dilatih, hasil prediksi pada data pengujian ( $y_{pred}$ ) dibandingkan dengan data aktual ( $y_{test}$ ).



Gambar 4. Matriks RMSE dan MAE

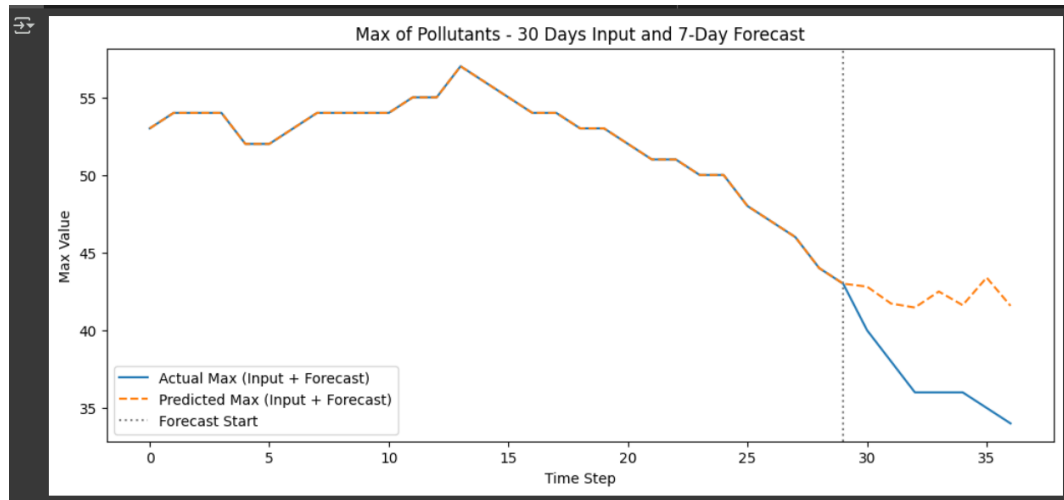
Gambar 8 adalah metrik RMSE dan MAE hasil evaluasi dari model peramalan (forecasting) selama 7 hari ke depan terhadap nilai atribut Max. Dalam 7 hari ke depan, model ini memiliki: Rata-rata kesalahan absolut sebesar 2.86 satuan (misal: derajat), Kesalahan rata-rata kuadrat sebesar 15.15, artinya ada variansi error yang lumayan, Kesalahan rata-rata akar kuadrat sebesar 3.89, artinya dalam praktik model bisa meleset sebesar 3.89 satuan dari nilai aktual.

MSE (Mean Squared Error) sebesar 15.15 adalah rata-rata dari kuadrat selisih antara nilai prediksi dan nilai aktual. MSE yang tinggi menunjukkan prediksi yang kurang akurat. RMSE (Root Mean Squared Error) sebesar 3.89 adalah akar kuadrat dari MSE, memiliki satuan yang sama dengan data yang prediksi sehingga lebih mudah diinterpretasikan dibanding MSE karena satuannya sama dengan target. MAE (Mean Absolute

*Error*) sebesar 2.86 adalah rata-rata dari nilai absolut selisih antara prediksi dan aktual, nilai ini tidak terlalu sensitif terhadap *outlier* seperti MSE sehingga dapat memberikan gambaran rata-rata kesalahan prediksi.

**(b). Visualisasi**

Gambar 9 adalah Grafik yang menampilkan hasil *forecasting* nilai Max of polutan 7 hari ke depan dari model untuk maksimum polutan harian berdasarkan data input 30 hari terakhir.



Gambar 5. Grafik Prediksi Max of Polutan untuk 7 hari dan 30 hari input pada pengujian Model

Sumbu X (*Time Step*) menunjukkan hari ke berapa dalam urutan waktu (dari 0 hingga sekitar 36). Hari 0–29 adalah data historis (input), hari 30–36 adalah periode ramalan (*forecast*). Sumbu Y (*Max Value*) menunjukkan nilai maksimum polutan (kemungkinan konsentrasi harian, misalnya PM2.5 atau CO). *Actual Max* (Garis Biru Solid) menunjukkan data aktual (nilai nyata yang terjadi). Tersedia untuk 30 hari input dan 7 hari *forecast* (meskipun pada hari ke-30 dan seterusnya mulai menurun tajam). *Predicted Max* (Garis Oranye Putus-putus) menunjukkan nilai hasil prediksi model. Ini mencakup juga 30 hari pertama (untuk menunjukkan akurasi selama pelatihan) dan 7 hari ke depan (*forecast* utama). *Forecast Start* (Garis Titik Vertikal) menandai titik transisi antara data historis dan periode *forecast* (yaitu pada time step ke-30), dari titik ini ke kanan adalah hasil ramalan murni dari model.

Pada 30 hari pertama, prediksi dan aktual hampir identik, dimana model mampu mengikuti pola historis dengan baik. Saat memasuki periode *forecast* model memprediksi tren penurunan secara perlahan, sedangkan data aktual justru menunjukkan penurunan yang lebih tajam. Ada sedikit underestimation oleh model terhadap kecepatan penurunan polutan.

**(c). Peningkatan Model**

Penulis melakukan peningkatan model dengan mengubah ukuran *batch size*. Pertama model dibentuk, penulis menggunakan ukuran *batch size* 16. Untuk meningkatkan model, penulis melakukan peningkatan jumlah *batch size* menjadi 32 dan 64. Tabel 2 menyajikan nilai MAE dari tiga ukuran *batch size* yang berbeda. Terlihat bahwa *batch size* 64 memiliki nilai MAE terkecil, disimpulkan *batch size* 64 memberikan kesalahan yang terkecil pada model, sehingga penulis melakukan peningkatan *batch size* menjadi 64.

Tabel 2 Hasil perhitungan MAE dari Peningkatan Model

Batch Size	MAE
16	+7,29
32	+6,29
64	+5,91

	Day	Predicted Max	Actual Max	Category
0	Day 1	42.809258	40.0	Good
1	Day 2	41.722733	38.0	Good
2	Day 3	41.462139	36.0	Good
3	Day 4	42.495506	36.0	Good
4	Day 5	41.614159	36.0	Good
5	Day 6	43.392059	35.0	Good
6	Day 7	41.570679	34.0	Good

Gambar 6. Hasil prediksi nilai Max of pollution selama 7 hari dari hasil peningkatan model

Dari hasil peningkatan model didapat hasil **cukup akurat** mengikuti tren historis (fit baik di 30 hari pertama), namun, untuk forecasting 7 hari ke depan, model **kurang responsif terhadap perubahan drastis** dalam data aktual (misal perubahan cuaca ekstrem, lockdown, dsb.). Hal tersebut dapat disebabkan oleh Kurangnya fitur tambahan (weather, kebijakan, dll.) dan model belum cukup kompleks atau perlu retraining. Tabel 3 adalah range nilai Max of Pollution yang direkomendasikan oleh Kementerian LHK Tahun 2020, dari data tersebut apabila di mapping kedalam range Peraturan Menteri LHK RI P.14/MENLHK/SETJEN/KUM.1/7/2020 maka hasil prediksi pada penelitian ini menghasilkan kategori Good untuk 7 hari kedepan.

Tabel 3 Range polusi udara menurut kementerian LHK Tahun 2020

Range	Category
0-50	Good
51-100	Moderate
101-200	Unhealthy
201-300	Very Unhealthy
300++	Dangerous

## KESIMPULAN

Model *Long Short-Term Memory* (LSTM) terbukti mampu memprediksi kualitas udara di Kota Tangerang Selatan dengan akurasi tinggi. Model berhasil mengikuti pola data historis 30 hari dan menghasilkan prakiraan 7 hari ke depan yang mendekati nilai aktual, dengan hasil evaluasi MSE sebesar 15,15, RMSE 3,89, dan MAE 2,86. Nilai tersebut menunjukkan tingkat kesalahan prediksi yang rendah. Model cenderung sedikit meng-*underestimasi* perubahan ekstrem, terutama saat terjadi penurunan kualitas udara secara tiba-tiba. Hal ini diperkirakan akibat terbatasnya variabel input dan kompleksitas model. Penyesuaian hyperparameter, khususnya *batch size* 64, berhasil menurunkan kesalahan prediksi (MAE ±5,91) dan meningkatkan sensitivitas model terhadap pola data. Model ini direkomendasikan untuk diterapkan sebagai sistem peringatan dini kualitas udara oleh pemerintah daerah, misalnya Dinas Lingkungan Hidup, untuk mendukung tindakan antisipatif dan mitigasi. Pengembangan selanjutnya disarankan mencakup penambahan variabel eksternal (data meteorologi dan kejadian khusus), peningkatan arsitektur model (seperti *deep LSTM* atau model hibrida), serta pelatihan ulang berkala agar model tetap akurat dan adaptif terhadap perubahan kondisi lingkungan.

## REFERENSI

- Agusmawati, N. K., Khoiriyah, F., & Tholib, A. (2023). Prediksi Harga Emas Menggunakan Metode Lstm Dan Gru. *Jurnal Informatika Dan Teknik Elektro Terapan*, 11(3), 620–627. <https://doi.org/10.23960/jitet.v11i3.3250>
- Arkadia, A., Hananto, B., & Prasvita, D. S. (2022). Optimasi Long Short Term Memory Dengan Adam Menggunakan Data Udara Kota DKI Jakarta. *Seminar Nasional Mahasiswa Ilmu Komputer Dan Aplikasinya (SENAMIKA)*, 92–101.
- Ashari, Y., & Suhendar, A. (2024). Implementasi Algoritma Long Short-Term Memory (Lstm) Untuk Memprediksi Harga Beras Di Jawa Tengah Berdasarkan Cuaca. *Djtechno: Jurnal Teknologi Informasi*, 5(3), 624–636. <https://doi.org/10.46576/djtechno.v5i3.5136>
- Cahyanita, J., Mujahidina, S., & Fiqar, T. P. (2023). Implementasi Metode Long Short Term Memory (LSTM) untuk Memprediksi Harga Bahan Pokok Nasional. *Jurnal Sistem Dan Teknologi Informasi (JUSTIN)*, 11. No. 2, 346–357. <https://doi.org/10.26418/justin.v11i2.57395>
- Fahim, K., Sholichah, S. A., Hanafi, L., & Surjanto, S. D. (2024). *Prediksi Indeks Standar Pencemar Udara (ISPU) di Kota Surabaya Menggunakan Rantai Markov Waktu Diskrit*. 12(2), 121–134.
- Ginting, F. P. (2023). Atasi Kualitas Udara di Tangsel Masih Buruk, Walikota Lakukan Langkah Penghijauan. *Pojoksatu.Id*. <https://www.pojoksatu.id/tangsel/1082707323/atasi-kualitas-udara-di-tangsel-masih-buruk->

walikota-lakukan-langkah-penghijauan

- Hasna, H., Nonong Amalita, Dony Permana, & Admi Salma. (2024). Random Forest Implementation for Air Pollution Standard Index Classification in DKI Jakarta 2022. *UNP Journal of Statistics and Data Science*, 2(2), 226–233. <https://doi.org/10.24036/ujsds/vol2-iss2/173>
- Hidayatullah, B. K., Kallista, M., Setianingsih, C., S1, P., & Komputer, T. (2022). Prediksi Indeks Standar Pencemar Udara Menggunakan Metode Support Vector Regression Berbasis Web (Studi Kasus Pada Kota Jakarta) Prediction of Air Pollutant Standard Index Using the Web-Based Long Short-Term Memory Method (Case Study in the Jakarta Capita. *E - Proceeding of Engineering*, 9(3), 1247–1255. <https://data.jakarta.go.id/>
- I Gusti Ayu Nandia Lestari, & I Nyoman Dwi Arysna Mahendra. (2023). Prediksi Kualitas Udara dengan Menggunakan Metode Long Short-Term Memory dan Artificial Neural Network. *Jurnal Sistem Dan Informatika (JSI)*, 17(2), 121–129. <https://doi.org/10.30864/jsi.v17i2.565>
- Karyadi, Y. (2022). Prediksi Kualitas Udara Dengan Metoda LSTM, Bidirectional LSTM, dan GRU. *JATISI (Jurnal Teknik Informatika Dan Sistem Informasi)*, 9(1), 671–684. <https://doi.org/10.35957/jatisi.v9i1.1588>
- Lattifia, T., Buana, P. W., & Rusjyanthi, N. K. D. (2022). Model Prediksi Cuaca Menggunakan Metode LSTM. *JITTER : Jurnal Ilmiah Teknologi Dan Komputer*, 3(1), 994. <https://doi.org/10.24843/jtrti.2022.v03.i01.p35>
- Lestarinigati, S. I., Putra, Y. H., & Teguh, R. (2013). Analisa Kualitas Layanan Aplikasi VOIP dan Implementasi Penggunaan Linux Zentyal sebagai Penyedia Layanan VOIP. *Komputika*, 2(1), 29–35.
- Mubarak, R., Maesaroh, S., Yusuf, M., Budiana, K., & Fahreza, M. R. (2026). *JOURNAL CERITA : Visualisasi Prediksi Prevalensi Balita Menggunakan*. 8(225), 164–171.
- Oktaviani, A., & Hustinawati. (2021). Prediksi Rata-Rata Zat Berbahaya Di Dki Jakarta Berdasarkan Indeks Standar Pencemar Udara Menggunakan Metode Long Short-Term Memory. *Jurnal Ilmiah Informatika Komputer*, 26(1), 41–55. <https://doi.org/10.35760/ik.2021.v26i1.3702>
- Pangaribuan, J. J., Fanny, F., Barus, O. P., & Romindo, R. (2023). Prediksi Penjualan Bisnis Rumah Properti Dengan Menggunakan Metode Autoregressive Integrated Moving Average (ARIMA). *Jurnal Sistem Informasi Bisnis*, 13(2), 154–161. <https://doi.org/10.21456/vol13iss2pp154-161>
- Prafanto, A., Astuti, I. F., Salamah, U., Agus, F., Kridalaksana, A. H., & Kamila, V. Z. (2024). Air Pollution Assessment of Samarinda Using the C4.5 Algorithm. *Buletin Poltanesa*, 24(2), 235–241. <https://doi.org/10.51967/tanesa.v24i2.2946>
- Putra, I. K. P. K. A., Sediono, Mardianto, M. F. F., & Pusporani, E. (2024). Analisis Prediktif Menggunakan Metode Hybrid Seasonal Autoregressive Integrated Moving Average – Artificial Neural Network pada Data Konsentrasi PM2.5 Harian di DKI Jakarta. *G-Tech: Jurnal Teknologi Terapan*, 8(1), 565–575. <https://doi.org/10.33379/gtech.v8i1.3896>
- Ramadhan, F., & Fachrie, M. (2024). Implementasi Algoritma Long Short-Term Memory Pada Sistem Prediksi Hasil Panen Sawit. *Jurnal Informatika Teknologi Dan Sains (Jinteks)*, 6(4), 937–944. <https://doi.org/10.51401/jinteks.v6i4.4876>
- Riansyah, H. (2025). Kota Tangerang Selatan: Dari Kawasan Pinggiran Menuju Kota Metropolitan Modern. *Indonesiana.Id*. <https://www.indonesiana.id/read/181194/kota-tangerang-selatan-dari-kawasan-pinggiran-menuju-kota-metropolitan-modern>
- Sianturi, T. B., Cholissodin, I., & Yudistira, N. (2023). Penerapan Algoritma Long Short-Term Memory (LSTM) berbasis Multi Fungsi Aktivasi Terbobot dalam Prediksi Harga Ethereum. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 7(3), 1101–1107. <http://j-ptiik.ub.ac.id>
- Tim/dmi. (2024). Tangsel Jawara Polusi Udara RI 2023 Kalahkan Jakarta. *Cnnindonesia.Com*. <https://www.cnnindonesia.com/teknologi/20240319140214-641-1076148/tangsel-jawara-polusi-udara-ri-2023-kalahkan-jakarta>
- Wiranda, L., & Sadikin, M. (2019). Penerapan Long Short Term Memory pada Data Time Series untuk Memprediksi Penjualan Produk PT. Metiska Farma. *Jurnal Nasional Pendidikan Teknik Informatika (JANAPATI)*, 8(3), 184–196.
- Zayed, R., & Abbod, M. (2024). Air Quality Index Prediction Using DNN-Markov Modeling. *Applied Artificial Intelligence*, 38(1). <https://doi.org/10.1080/08839514.2024.2371540>