

PREDIKSI INDEKS STANDAR PENCEMAR UDARA MENGGUNAKAN METODE *LONG SHORT-TERM MEMORY* 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 CAPITAL SPECIAL REGION)

Benny Khalid Hidayatullah¹, Meta Kallista², Casi Setianingsih³

^{1,2,3}Prodi S1 Teknik Komputer, Fakultas Teknik Elektro, Universitas Telkom

¹bennykhalid@student.telkomuniversity.ac.id, ²metakallista@telkomuniversity.ac.id,

³setiacasie@telkomuniversity.ac.id

Abstrak

Penggunaan *long short-term memory* karena metode tersebut menghasilkan prediksi angka sangat baik untuk data *time series*. prediksi yang telah dilakukan lalu diimplementasikan ke dalam *website* menggunakan kerangka flask. Penelitian menggunakan 5 parameter zat yang terdapat dalam ISPU sehingga mendapatkan hasil pengujian berbeda di setiap parameter zat. Hasil pengujian partisi data tiap zat adalah PM₁₀ 80% data latih 20% data uji, SO₂ 50% data latih 50% data uji, CO 80% data latih 20% data uji, O₃ 80% data latih 20% data uji, dan NO₂ 50% data latih 50% data uji. Dari hasil pengujian partisi data ditemukan pemodelan terbaik dari PM₁₀ dengan *epoch* = 100, *hidden layer* = 1, *neuron* = 128, *optimizer* = Rmsprop, RMSE sebesar 0.007237, SO₂ dengan *epoch* = 50, *hidden layer* = 1, *neuron* = 64, *optimizer* = Adamax, RMSE sebesar 0.05841, CO dengan *epoch* = 100, *hidden layer* = 1, *neuron* = 64, *optimizer* = Adamax, RMSE sebesar 0.05474, O₃ dengan *epoch* = 50, *hidden layer* = 1, *neuron* = 64, *optimizer* = Adam, RMSE sebesar 0.04465, dan NO₂ dengan *epoch* = 50, *hidden layer* = 1, *neuron* = 64, *optimizer* = Adam, RMSE sebesar 0.0431. berdasarkan *usability testing* yang telah dilakukan, *website* yang dirancang dapat membantu dalam memprediksikan ISPU dengan presentase nilai 80.7%.

Kata Kunci: udara, *long short-term memory*, ISPU, DKI Jakarta, *website*

Abstract.

The use of the *long short-term memory* method because the method produces excellent number predictions for time series data. Predictions that have been made and then implemented into the web using the flask framework. The study used 5 substance parameters contained in ISPU so as to get different test results in each substance parameter. The results of data partition testing of each substance are PM10 80% training data 20% test data, SO2 50% training data 50% test data, CO 80% training data 20% test data, O3 80% training data 20% test data, and NO2 50% training data 50% test data. From the results of data partition testing found the best modeling of PM10 with *epoch* = 100, *hidden layer* = 1, *neuron* = 128, *optimizer* = Rmsprop, RMSE of 0.007237, SO2 with *epoch* = 50, *hidden layer* = 1, *neuron* = 64, *optimizer* = Adamax, RMSE of 0.05841, CO with *epoch* = 100, *hidden layer* = 1, *neuron* = 64, *optimizer* = Adamax, RMSE of 0.05474, O3 with *epoch* = 50, *hidden layer* = 1, *neuron* = 64, *optimizer* = Adam, RMSE of 0.04465, and NO2 with *epoch* = 50, *hidden layer* = 1, *neuron* = 64, *optimizer* = Adam, RMSE of 0.0431. Based on *usability testing* that has been done, the *website* designed can help in predicting ISPU with a percentage value of 80.7%.

Keywords: air, *long short-term memory*, ISPU, DKI Jakarta, *website*

1. Pendahuluan

Pencemaran udara merupakan salah satu ancaman terbesar bagi kesehatan serta lingkungan. Dari data WHO, sekitar lebih dari 80% populasi yang tinggal di daerah urban yang termonitor pencemaran udara telah terpapar polusi udara yang melebihi ambang batas normal dari indeks WHO. Dengan menurunnya kualitas udara, risiko penyakit pernapasan seperti kanker paru-paru, asma, serta penyakit jantung akan meningkat pesat pada orang yang hidup di daerah yang terpapar.

Pencemaran udara tentu menjadi masalah utama bagi setiap daerah baik perkotaan di negara maju maupun berkembang terlebih meningkatnya jumlah kendaraan, daerah industri serta sanitasi udara yang tidak layak. Dengan meningkatnya pencemaran udara, maka suatu negara menerapkan indeks standar akan pencemaran udara, salah satunya ISPU. ISPU adalah indeks standar pencemaran udara untuk menggambarkan kategori serta status kualitas udara. Terdapat lima parameter dalam ISPU sebagai pengamatan kualitas udara seperti : Tingkat Partikulat (PM10), Oksida Nitrogen (NO2), Sulfur Dioksida (SO2), Karbon Monoksida (CO), dan ozon permukaan (O3).

Maka dari itu, hal yang dapat kita lakukan adalah dengan melakukan prediksi atau peramalan. Prediksi dilakukan sebagai upaya kita mengetahui danantisipasi terhadap pencemaran udara di masa mendatang. Data yang digunakan

adalah time series, selanjutnya data tersebut digunakan untuk melatih mesin pembelajaran. Salah satu algoritma yang umum digunakan dalam melakukan prediksi pada data time series adalah algoritma long short-term memory.

Long short-term memory (LSTM) adalah sebuah metode dari cabang recurrent neural network (RNN) yang bisa memproses data bentuk sequential. Seiring berkembang waktu, ditemukan bahwa RNN yang memiliki kelemahan dalam mempelajari data jarak jauh (long range dependencies), maka dibuatlah algoritma LSTM.

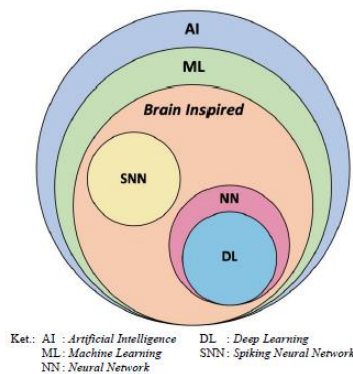
Beberapa penelitian telah menggunakan algoritma atau metode LSTM seperti dari Air Quality Prediction in Visakhapatnam with LSTM based Recurrent Neural Networks [1], Prediksi Polusi Udara Perkotaan Di Surabaya Menggunakan Recurrent Neural Network – Long Short Term Memory [2], Predicting air quality with deep learning LSTM: Towards comprehensive models [3] dan beberapa penelitian lainnya.

Dalam penelitian Tugas Akhir ini, penelitian memprediksikan Indeks kelima parameter zat yang telah disebutkan menjadi acuan penelitian. Kelima zat tersebut dilatih secara satu per satu menggunakan metode LSTM dengan data rentang waktu dari tahun 1 Januari 2015 sampai 31 Oktober 2021 agar model dapat mempelajari data lebih baik. Data yang didapat berasal dari situs web portal pemerintah provinsi DKI Jakarta yakni <https://data.jakarta.go.id/> dalam bentuk comma separated values (CSV). Selain sistem prediksi, dibuat juga sebuah website untuk memvisualisasikan prediksi ISPU yang telah dibuat sehingga pengguna dapat memprediksikan kualitas udara beberapa bulan ke depan.

2. Landasan Teori

2.1 Deep learning

Sama halnya dengan Pembelajaran Mesin (machine learning), Deep learning adalah sebuah produk dari Kecerdasan Buatan (artificial intelligence). Namun, jika pembelajaran mesin menghasilkan neural network, sedangkan Deep learning adalah implementasi dari neural network itu sendiri yang bisa dilihat pada Gambar 2.1 [8].



Gambar 2.11 Kategorisasi di dalam artificial intelligence.

Deep learning menggunakan pendekatan pembelajaran hierarkis. Dalam hal ini, *learning* adalah suatu urutan yang berisi proses memperkirakan parameter suatu model sehingga model yang dikembangkan dapat memecahkan masalah tertentu [8]. *Deep learning* juga menggunakan lapisan (*layer*) yang berbeda antara lapisan *input* dan lapisan *output*. Arsitektur ini dapat digunakan untuk melakukan pemrosesan multi-langkah non linier, yang hasilnya dapat digunakan untuk pembelajaran fitur dan klasifikasi pola [9], [10]. Lapisan *deep learning* dapat digunakan untuk melakukan abstraksi pada level yang berbeda [11].

2.2 Long Short-Term Memory

Long short-term memory (LSTM) adalah algoritma turunan dari RNN. LSTM sendiri terdiri dari cell, input gate, output gate dan forget gate [16][17][18]. Jaringan saraf LSTM biasa digunakan untuk klasifikasi, processing dan bahkan peramalan berdasarkan data rentang waktu.

Fungsi aktivasi merupakan sebuah fungsi matematis yang berguna menentukan output dari jaringan saraf. Aktivasi ini dimasukkan kedalam setiap saraf disetiap jaringan kecuali saraf yang berada didalam input layer untuk menentukan apakah harus aktif atau tidak berdasarkan input setiap saraf relevan untuk peramalan model. Didalam LSTM, aktivasi Sigmoid digunakan untuk mengaktifkan gate input, output serta forget. Dengan rumus seperti berikut [19]:

$$\sigma(x) = \frac{1}{e^{-x}} \quad (2.2)$$

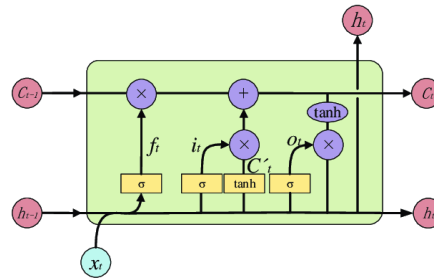
Keterangan:

σ = fungsi sigma

e = eksponensial

Adapun fungsi tanh dengan rumus berikut:

$$\tanh(x) = 2 * \sigma (2 * x) - 1 \tag{2.3}$$



Gambar 2.2 Arsitektur dari long short-term memory.

Pada Gambar 2.3, terdapat 3 jenis gerbang (gate) diantaranya gerbang keluaran O_t , gerbang lupa atau forget gate f_t dan juga gerbang masukan i_t . Setiap gerbang menggunakan aktivasi Sigmoid untuk menentukan apakah gerbang tersebut bernilai aktif atau tidak (0/1). Forget gate digunakan untuk menentukan apakah data yang datang perlu disimpan atau sebaliknya. Dengan rumus berikut [20]:

$$f_t = \sigma(U_f x_t + w_f h_{t-1} + b_f) \tag{2.4}$$

Keterangan:

- f_t = forget cell
- w_f, U_f = bobot
- h_{t-1} = output terdahulu
- x_t = data masukan
- σ = fungsi sigmoid

Gerbang masukan (input) berguna untuk memasukkan data perlu diperbarui atau tidak

$$i_t = \sigma(w_i x_t + U_i h_{t-1} + b_i) \tag{2.5}$$

Keterangan:

- i_t = input cell
- w_i, U_i = bobot

Cell state berguna untuk tempat masukan data lampau sehingga bisa dipakai untuk menentukan keluaran dari LSTM, dengan \tilde{c}_t adalah value dari cell state yang akan diperbarui dan c_t adalah variabel cell state yang memiliki memori sebelumnya.

$$\tilde{c}_t = \tanh(U_c x_t + w_c h_{t-1} + b_c) \tag{2.6}$$

Keterangan:

- \tilde{c}_t = cell state yang akan diperbarui
- w_c, U_c = bobot
- Tanh = aktivasi tanh

$$C_t = f_t * C_{t-1} + i_t * \tilde{c}_t \tag{2.7}$$

Keterangan:

- C_{t-1} = value cell state sebelumnya

Dan terakhir gerbang keluaran berguna untuk mengeluarkan hasil peramalan berdasarkan hasil perkalian antara cell state dan input data.

$$O_t = \sigma(U_o x_t + w_o h_{t-1} + b_o) \tag{2.8}$$

$$h_t = O_t * \tanh (c_t) \tag{2.9}$$

Keterangan:

- O_t = output cell
- h_t = output peramalan

2.3 Indeks Standar Pencemaran Udara

Di dalam Peraturan Menteri Lingkungan Hidup dan kehutanan nomor 14 tahun 2020 pasal 1, Indeks Standar Pencemaran 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.

Kategori ISPU sendiri terbagi atas lima kategori dengan awal berstatus baik hingga terberat yaitu berbahaya dengan rentang masing-masing. Kategori sebagaimana dari tabel berikut.

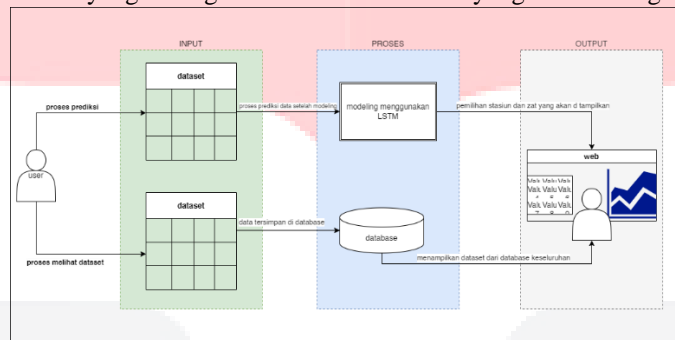
Tabel 2.1 Kategori ISPU, status warna, dan nilai rentang.

kategori	Status warna	rentang
Baik		1 – 50
sedang		51 – 100
Tidak Sehat		101 – 200
Sangat Tidak Sehat		201 – 300
Berbahaya		≥301

3. Perancangan Sistem

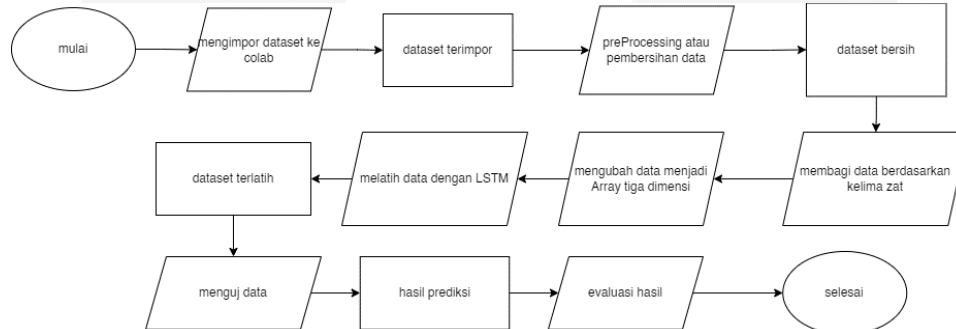
3.1 Gambaran Umum Sistem

Untuk perancangan model yang dibangun secara umum sistem yang akan dibangun seperti gambar berikut:



3.2 Perancangan Metode Long Short-Term Memory

Pembuatan program Prediksi Indeks Standar Pencemar Udara dimulai dari pengumpulan data, menyiapkan data, pembersihan data, melatih data dengan model sampai *output* berupa prediksi itu sendiri. Dengan diagram alir sebagai berikut:



Gambar 3.3 Perancangan metode long short-term memory.

3.2.1 Sumber Data

Data yang didapat berasal dari pemerintah Provinsi DKI Jakarta di laman website <https://data.jakarta.go.id/> yang didistribusikan langsung oleh pihak Dinas Lingkungan Hidup provinsi DKI Jakarta dengan rentang 1 Januari 2015 sampai 31 Oktober 2021 berjumlah 12480 data. Data ini merupakan data harian.

A	B	C	D	E	F	G
tanggal	stasiun	pm10	so2	co	o3	no2
2015-01-01	DKI1 (Bunderan HI)	49	7	18	26	10
2015-01-02	DKI1 (Bunderan HI)	25	7	10	25	10
2015-01-03	DKI1 (Bunderan HI)	27	6	12	36	10
2015-01-04	DKI1 (Bunderan HI)	22	5	10	38	12
2015-01-05	DKI1 (Bunderan HI)	21	5	8	27	7
2015-01-06	DKI1 (Bunderan HI)	29	7	11	27	11
2015-01-07	DKI1 (Bunderan HI)	32	7	11	35	12
2015-01-08	DKI1 (Bunderan HI)	29	6	10	36	13
2015-01-09	DKI1 (Bunderan HI)	23	6	8	32	7
2015-01-10	DKI1 (Bunderan HI)	25	6	9	38	8
2015-01-11	DKI1 (Bunderan HI)	26	6	7	41	11
2015-01-12	DKI1 (Bunderan HI)	26	7	23	37	13
2015-01-13	DKI1 (Bunderan HI)	45	5	26	28	18
2015-01-14	DKI1 (Bunderan HI)	41	5	24	34	16
2015-01-15	DKI1 (Bunderan HI)	39	4	27	36	20
2015-01-16	DKI1 (Bunderan HI)	50	5	20	33	15

Gambar 3.4 Data berbentuk CSV.

3.2.2 Preprocessing Data

Tentunya data yang didapat masih dalam bentuk kotor atau belum layak dimasukkan ke dalam modeling,

sehingga perlu pembersihan data, dengan proses sebagai berikut:

- 1 Menyamakan bentuk dari kolom tanggal. bentuk tanggal yang digunakan diseragamkan menjadi format '%Y-%m-%d'
- 2 Mengubah data kosong '---' menjadi *NaN* untuk proses interpolasi.
- 3 Mengganti format nama stasiun yang berbeda. Pada DKI 5 terdapat 2 format nama, yaitu 'DKI5 (Kebon Jeruk)' dan 'DKI5 (Kebon Jeruk) Jakarta Barat'.
- 4 Mengubah menjadi bilangan *float*, data yang akan diinterpolasi tentunya harus berbentuk *float* atau desimal.
- 5 Menginterpolasi data, data selanjut dilakukan proses interpolasi atau pengisian data kosong. Teknik ini umum digunakan untuk *PreProcessing* data sehingga data terisi karena syarat modeling adalah data haruslah lengkap tanpa ada nilai kosong. Interpolasi yang digunakan adalah interpolasi linier.

$$y = y_1 + \frac{(x-x_1)}{(x_2-x_1)} \times (y_2 - y_1) \quad (3.1)$$

- 6 Transformasi data, data yang telah di interpolasi selanjutnya dilakukan proses transformasi dengan cara normalisasi data. Normalisasi bertujuan menyederhanakan angka dalam parameter zat menjadi skala 0 – 1 sehingga proses pembelajaran mesin menjadi lebih cepat dan akurasi lebih baik karena data pada rentang yang sama. Normalisasi yang digunakan adalah *MinMax scaler*.

$$X' = \frac{X - \min(X)}{\max(X) - \min(X)} \quad (3.4)$$

X' = hasil normalisasi

X = data awal

$\min(X)$ = data terkecil

$\max(X)$ = data terbesar

3.2.3 Proses Pelatihan Model

Berikut ini contoh perhitungan manual dari metode *long short-term memory*:

Input merupakan data sampel yang dimana:

$$\text{Input } 1 = x_t = [4 \ 0]$$

$$C_{t-1} = 0.814$$

$$h_{t-1} = 0.672$$

Diketahui bahwa bobot awal sebagai contoh:

$$w_f = 0.541, w_i = 0.654, w_c = 0.691, w_o = 0.842$$

$$U_f = \begin{bmatrix} 0.422 \\ 0.231 \end{bmatrix} \quad U_i = \begin{bmatrix} 0.291 \\ 0.378 \end{bmatrix} \quad U_c = \begin{bmatrix} 0.696 \\ 0.384 \end{bmatrix} \quad U_o = \begin{bmatrix} 0.844 \\ 0.232 \end{bmatrix}$$

Berdasarkan rumus (2.4) didapat hasil *forget gate* adalah:

$$f_t = \sigma \left(\begin{bmatrix} 0.422 \\ 0.231 \end{bmatrix} * [4 \ 0] + 0.541 * 0.672 \right) = 0.88$$

Berdasarkan rumus (2.5) didapat hasil *input gate* adalah:

$$i_t = \sigma \left(\begin{bmatrix} 0.291 \\ 0.378 \end{bmatrix} * [4 \ 0] + 0.654 * 0.672 \right) = 0.832$$

Berdasarkan rumus (2.6) didapat hasil *candidate value cell* adalah:

$$\tilde{c}_t = \sigma \left(\begin{bmatrix} 0.696 \\ 0.384 \end{bmatrix} * [4 \ 0] + 0.691 * 0.672 \right) = 0.962$$

Berdasarkan rumus (2.7) didapat hasil *candidate cell state* adalah:

$$C_t = 0.88 * 0.814 + 0.832 * 0.962 = 1.516$$

Berdasarkan rumus (2.8) didapat hasil *output gate* adalah:

$$o_t = \sigma \left(\begin{bmatrix} 0.844 \\ 0.232 \end{bmatrix} * [4 \ 0] + 0.842 * 0.672 \right) = 0.980$$

Berdasarkan rumus (2.9) didapat hasil *output gate* adalah:

$$h_t = 0.980 * \tanh(1.516) = 0.889$$

3.2.4 Perhitungan Evaluasi Hasil

Root Mean Square Error:

$$RMSE = \sqrt{\frac{\sum_{t=1}^n (A_t - F_t)^2}{N}} \quad (3.8)$$

Keterangan:

A_t = nilai asli

F_t = nilai prediksi

N = jumlah dari keseluruhan data

Skor R^2 :

$$R^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2} \quad (3.9)$$

Keterangan:

Y_i = nilai asli

\hat{y}_i = nilai prediksi

\bar{y} = nilai jumlah keseluruhan data dari nilai asli

4. Hasil dan Analisis

4.1 Pengujian Parameter Terbaik

Pengujian parameter adalah pengujian yang 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.

4.1.1 Pencarian Partisi Terbaik

Parameter pertama yang dipakai adalah parameter umum digunakan pada metode LSTM, dengan parameter dan hasil pengujian seperti berikut:

Tabel 4.1 Parameter inialisasi.

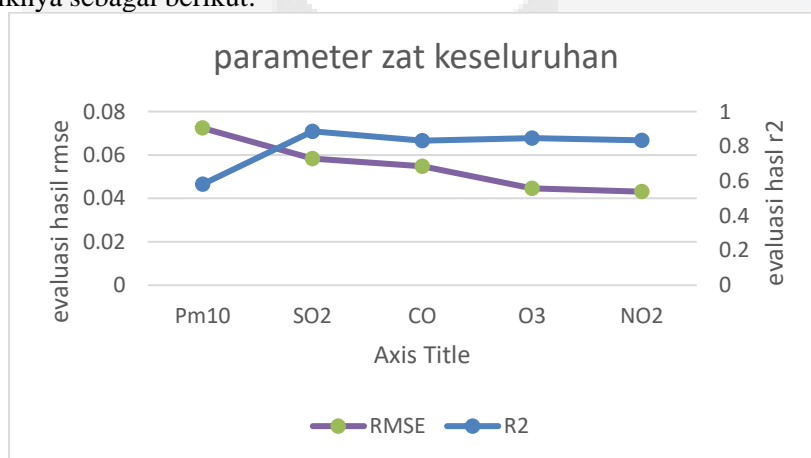
Modeling	jumlah parameter
hidden neuron (per 1 layer)	64
learning rate	0.001
Activation	sigmoid
Epoch	50
optimizer	Adam
output	1

Maka hasil dari pencarian parameter kelima zat sebagai berikut:

Tabel 4.6 Parameter yang telah didapatkan.

Zat	KD	E	HL & N	O	Lr	RMSE	R2
PM ₁₀	80% 20%	100	128	Rmsprop	0.001	0.07237	0.58157
SO ₂	50% 50%	50	64	Adamax	0.001	0.05841	0.88621
CO	80% 20%	100	64	Adamax	0.001	0.05474	0.83185
O ₃	80% 20%	50	64	Adam	0.001	0.04465	0.84706
NO ₂	50% 50%	50	64	Adam	0.001	0.0431	0.83509

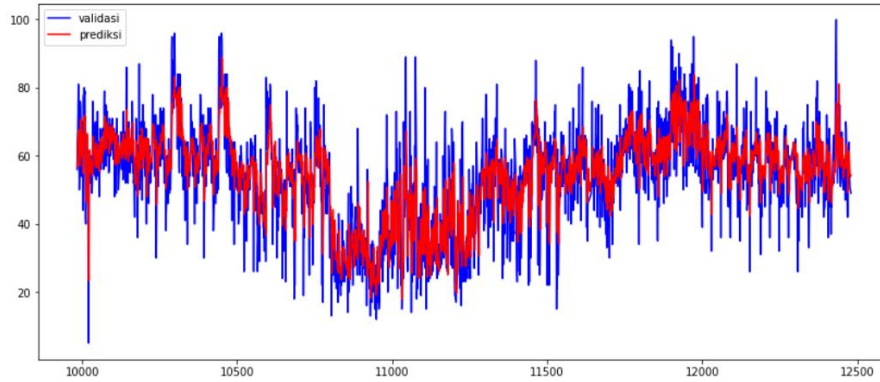
Dan grafiknya sebagai berikut:



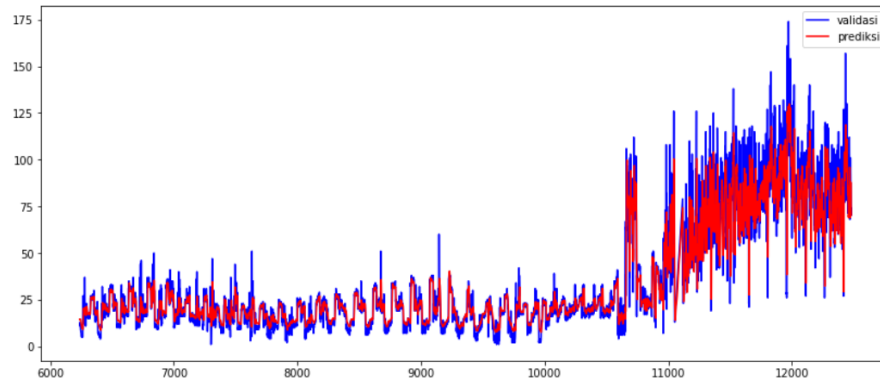
Gambar 4.1 Parameter zat yang didapatkan dari pengujian.

4.2 Penggunaan Parameter untuk Prediksi

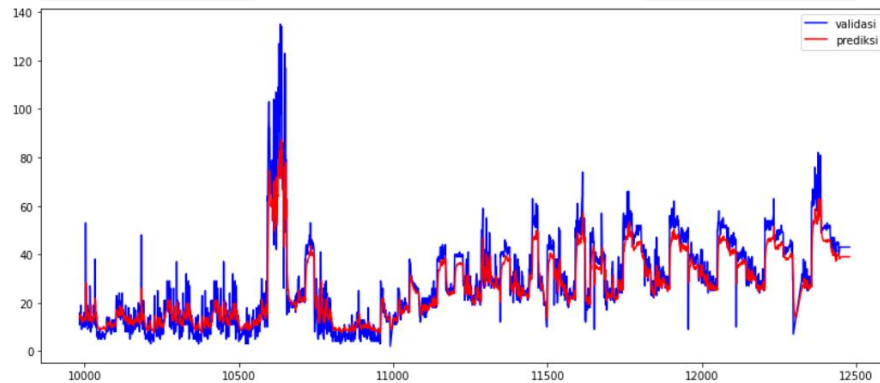
Dengan grafik komparasi antara nilai asli dan nilai prediksi seperti berikut:



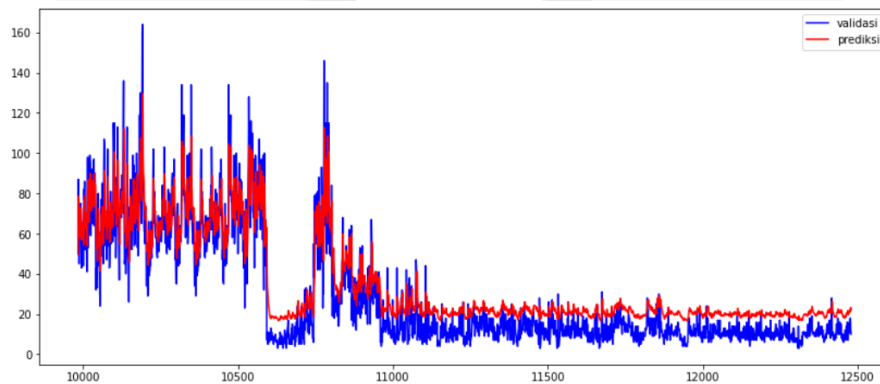
Gambar 4.2 Prediksi PM₁₀.



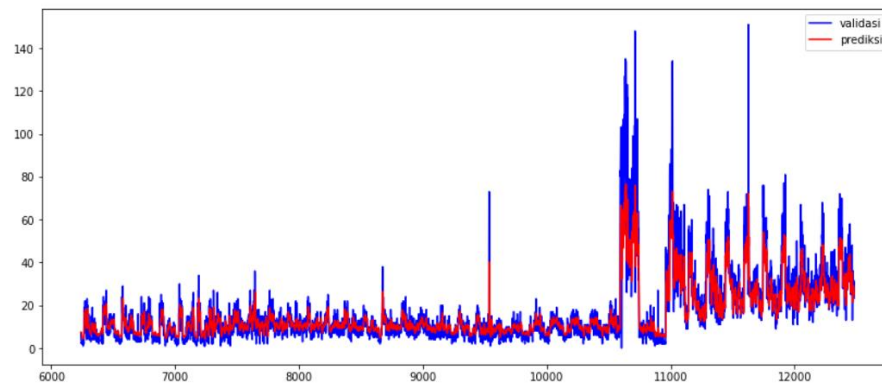
Gambar 4.3 Prediksi SO₂.



Gambar 4.4 Prediksi CO.



Gambar 4.5 Prediksi O₃.

Gambar 4.6 Prediksi NO₂.

5. Kesimpulan dan Saran

5.1 Kesimpulan

- 1 Sistem *website* untuk memvisualkan hasil prediksi ISPU ke dalam *website* menggunakan metode LSTM berjalan baik dengan hasil pengujian alpha atau uji fungsionalitas sebesar 100%.
- 2 Hasil dari proses pengujian partisi data setiap zat adalah PM₁₀ 80% data latih 20% data uji, zat SO₂ 50% data latih 50% data uji, zat CO 80% data latih 20% data uji, zat O₃ 80% data latih 20% data uji, dan zat NO₂ 50% data latih 50% data uji. Selain itu, hasil pengujian parameter terbaik dari kelima zat tersebut menghasilkan rentang nilai RMSE dari 0.0431 sampai 0.07237 dan skor R² dari rentang nilai 0.58157 hingga 0.88621.

5.2 Saran

- 1 Menggunakan parameter zat lain seperti parameter partikulat (PM 2.5) atau hidrokarbon (HC) sebagai klasifikasi maupun regresi, ini merujuk kepada Peraturan Menteri LHK nomor P.14/MENLHK /SETJEN/KUM.1/7/2020 yang menambahkan kedua parameter tersebut sebagai parameter pemantauan.
- 2 Menggunakan jenis LSTM lain seperti Bi-LSTM atau varian RNN lain seperti GRU.

DAFTAR PUSTAKA

- [1] K. S. Rao, G. L. Devi, and N. Ramesh, "Air Quality Prediction in Visakhapatnam with LSTM based Recurrent Neural Networks," *Int. J. Intell. Syst. Appl.*, vol. 11, no. 2, pp. 18–24, 2019, doi: 10.5815/ijisa.2019.02.03.
- [2] A. N. Irfansyah, "Prediksi polusi udara perkotaan di surabaya menggunakan," vol. 18, pp. 102–114, 2020.
- [3] R. Navares and J. L. Aznarte, "Predicting air quality with deep learning LSTM: Towards comprehensive models," *Ecol. Inform.*, vol. 55, p. 101019, 2020, doi: 10.1016/j.ecoinf.2019.101019.
- [4] A. L. Samuel, "Some Studies in Machine Learning."
- [5] A. S. and S. V. N. Vishwanathan, "Introduction to machine learning," *Methods Mol. Biol.*, vol. 1107, pp. 105–128, 2014, doi: 10.1007/978-1-62703-748-8_7.
- [6] Herdianto, "Prediksi Kerusakan Motor Induksi Menggunakan Tesis Oleh Herdianto Fakultas Teknik," 2013.
- [7] V. Dhar, "Data Science and Prediction Vasant Dhar Professor, Stern School of Business Director, Center for Digital Economy Research," *Commun. ACM*, no. May, pp. 64–73, 2012, [Online]. Available: <http://archive.nyu.edu/bitstream/2451/31553/2/Dhar-DataScience.pdf>.
- [8] M. Z. Alom et al., "A state-of-the-art survey on deep learning theory and architectures," *Electron.*, vol. 8, no. 3, 2019, doi: 10.3390/electronics8030292.
- [9] J. Schmidhuber, "Deep Learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, 2015, doi: 10.1016/j.neunet.2014.09.003.
- [10] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015, doi: 10.1038/nature14539.
- [11] M. Monteleone, "NooJ local grammars and formal semantics: Past participles vs. adjectives in Italian," *Commun. Comput. Inf. Sci.*, vol. 607, no. 8, pp. 83–95, 2016, doi: 10.1007/978-3-319-42471-2_8.
- [12] X. Zhang, M. H. Chen, and Y. Qin, "NLP-QA framework based on LSTM-RNN," 2018 2nd Int. Conf. Data Sci. Bus. Anal., pp. 307–311, 2018, doi: 10.1109/ICDSBA.2018.00065.
- [13] S. Abujar, A. K. M. Masum, S. M. M. H. Chowdhury, M. Hasan, and S. A. Hossain, "Bengali Text generation Using Bi-directional RNN," 2019 10th Int. Conf. Comput. Commun. Netw. Technol. ICCCNT 2019, pp. 1–5, 2019, doi: 10.1109/ICCCNT45670.2019.8944784.
- [14] L. Guo, T. Ge, and J. S. Chang, "A 101 dB PSRR, 0.0027% THD + N and 94% power-efficiency filterless class D amplifier," *IEEE J. Solid-State Circuits*, vol. 49, no. 11, pp. 2608–2617, 2014, doi: 10.1109/JSSC.2014.2359913.
- [15] A. Xiao, J. Liu, Y. Li, Q. Song, and N. Ge, "Two-phase rate adaptation strategy for improving real-time video QoE in mobile networks," *China Commun.*, vol. 15, no. 10, pp. 12–24, 2018, doi: 10.1109/CC.2018.8485465.
- [16] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to Forget: Continual Prediction with LSTM," *Neural Comput.*, vol. 12, no. 10, pp. 2451–2471, Oct. 2000, doi: 10.1162/089976600300015015.
- [17] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

- [18] V. Turchenko et al., "Stanford CS224d: Deep Learning for NLP Lecture 4," *Sigir*, pp. 655–665, 2015, [Online]. Available: <http://dx.doi.org/10.3115/v1/w15-1509><http://disi.unitn.it/~severyn/papers/sigir-2015-long.pdf><http://emnlp2014.org/papers/pdf/EMNLP2014181.pdf><http://goo.gl/EsQCuC><http://arxiv.org/abs/1512.01596>.
- [19] S. Sharma, S. Sharma, and A. Athaiya, "Activation Functions in Neural Networks," *Int. J. Eng. Appl. Sci. Technol.*, vol. 04, no. 12, pp. 310–316, 2020, doi: 10.33564/ijeast.2020.v04i12.054.
- [20] U. Ugurlu, I. Oksuz, and O. Tas, "Electricity price forecasting using recurrent neural networks," *Energies*, vol. 11, no. 5, pp. 1–23, 2018, doi: 10.3390/en11051255.
- [21] S. Chaudhury and T. Yamasaki, "Robustness of Adaptive Neural Network Optimization Under Training Noise," *IEEE Access*, vol. 9, pp. 37039–37053, 2021.
- [22] R. A. Holzworth, G.C. & Cormick, "Air Pollution third edition. Academy Press," Third., new york: In A.C Stren (Eds), 1976.
- [23] J. Feng, X. He, Q. Teng, C. Ren, H. Chen, and Y. Li, "Reconstruction of porous media from extremely limited information using conditional generative adversarial networks," *Phys. Rev. E*, vol. 100, no. 3, p. 33308, 2019, doi: 10.1103/PhysRevE.100.033308.