

PERBANDINGAN KINERJA MODEL DERET WAKTU ARIMA, LSTM, DAN GRU UNTUK PERAMALAN PM10 HARIAN DI JAKARTA

Khusnia Nurul Khikmah*

Program Studi Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Palangka Raya,
Kota Palangka Raya, Kalimantan Tengah, 74874 Indonesia

*Corresponding Email: khusnia.nurulkhikmah@mipa.upr.ac.id

A'yunin Sofro

Program Studi Sains Aktuaria, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Negeri Surabaya,
Kota Surabaya, Jawa Timur, 60231 Indonesia

Email: ayuninsofro@unesa.ac.id

Abstrak

Polusi udara khususnya *partikulat matter* kategori PM10 merupakan masalah lingkungan utama di kota besar seperti Jakarta. Dampaknya terhadap kesehatan masyarakat dan kebutuhan akan kebijakan mitigasi yang tepat menjadikan analisis kualitas udara sangat mendesak untuk dilakukan. Salah satu pendekatan yang dapat digunakan untuk menganalisis masalah ini adalah melalui peramalan konsentrasi PM10. Penelitian ini bertujuan membandingkan kinerja model statistik tradisional yaitu *autoregressive integrated moving average* (ARIMA) dengan model *deep learning* yaitu *long short-term memory* (LSTM) dan *gated recurrent unit* (GRU) dalam meramalkan konsentrasi PM10 harian di Jakarta. Data yang digunakan adalah data PM10 periode Januari 2010 hingga Februari 2025 dengan total 5538 observasi harian. Hasil penelitian menunjukkan bahwa model *deep learning* GRU kemudian LSTM memberikan hasil akurasi yang lebih baik dibanding model statistik tradisional ARIMA, yang dibuktikan secara berurutan melalui nilai RMSE (13,366, 13,441, dan 17,526), MAE (8,918, 8,958, dan 12,416), MSE (178,647, 189,674, dan 307,163), dan MAPE (21,211%, 21,361%, dan 31,228%). Hasil ini menunjukkan bahwa GRU dan LSTM lebih baik dalam menangkap pola *non-linear* dan ketergantungan jangka panjang pada PM10 dibandingkan metode ARIMA tradisional.

Kata Kunci: ARIMA, GRU, kualitas udara, LSTM, Peramalan PM10.

Abstract

Air pollution, particularly PM10 particulate matter, is a major environmental problem in large cities such as Jakarta. Its impact on public health and the need for appropriate mitigation policies make air quality analysis an urgent priority. One approach to analyzing this issue is to forecast PM10 concentrations. This study aims to compare the performance of a traditional statistical model, namely the *autoregressive integrated moving average* (ARIMA), with deep learning models, namely *long short-term memory* (LSTM) and *gated recurrent unit* (GRU), in forecasting daily PM10 concentrations in Jakarta. The data used consists of PM10 data from January 2010 to February 2025, totaling 5538 daily observations. The results show that the deep learning models GRU and LSTM provide better accuracy than the traditional statistical model ARIMA, as evidenced by their respective RMSE values (13,366, 13,441, and 17,526), MAE (8,918, 8,958, and 12,416), MSE (178,647, 189,674, and 307,163), and MAPE (21,211%, 21,361%, and 31,228%). These results indicate that GRU and LSTM models are better at capturing nonlinear patterns and long-term dependencies in PM10 than traditional ARIMA methods.

Keywords: ARIMA, GRU, air quality, LSTM, PM10 forecasting.

PENDAHULUAN

Kualitas udara di kota besar seperti Jakarta menjadi permasalahan penting dalam beberapa tahun terakhir. *Partikulat matter* dengan diameter kurang dari 10 mikrometer (PM10) yang merupakan polutan hasil emisi kendaraan bermotor, aktivitas

industri, dan pembakaran terbuka (Perdana et al., 2023). Konsentrasi PM10 yang tinggi telah terbukti berkorelasi positif dengan peningkatan kasus penyakit pernapasan, kardiovaskular hingga kematian (Wellid et al., 2024). Data menunjukkan bahwa fluktuasi PM10 yang signifikan dari waktu ke

waktu dengan data maksimum sebesar $403 \mu\text{g}/\text{m}^3$. Oleh karena itu, kemampuan untuk meramalkan konsentrasi PM10 secara akurat penting dalam memberikan peringatan dini kepada masyarakat dan mitigasi pengendalian polusi yang tepat sasaran.

Peramalan data deret waktu telah lama didominasi oleh metode statistik tradisional seperti *autoregressive integrated moving average* (ARIMA) (Khikmah & Sofro, 2025; Siami-Namini et al., 2018). Metode ini bekerja dengan mengidentifikasi struktur autokorelasi dalam data dan telah berhasil diterapkan dalam berbagai bidang. Namun demikian, ARIMA memiliki keterbatasan fundamental karena mengasumsikan hubungan linear antar observasi dan membutuhkan stasioneritas data. Selain itu, peramalan deret waktu memiliki perkembangan dengan pembelajaran mesin, seperti *long short-term memory* (LSTM) yang dirancang khusus untuk mengatasi masalah *vanishing gradient* pada RNN standar, sehingga mampu mempelajari ketergantungan jangka panjang dalam data (Ławryńczuk & Zarzycki, 2025; R. Zhang et al., 2021). Metode pengembangan lainnya yaitu, *gated recurrent unit* (GRU) yang merupakan varian LSTM dengan arsitektur lebih sederhana yang menggabungkan *input gate* dan *forget gate* menjadi satu *update gate*, sehingga lebih efisien secara komputasi (Pierre et al., 2023). Sehingga, ketiga metode tersebut cocok untuk digunakan dalam meramalkan data PM10 yang menunjukkan pola *non-linear* yang kompleks akibat interaksi berbagai faktor meteorologi dan antropogenik.

Oleh karena itu, penelitian ini bertujuan untuk membandingkan kinerja tiga model peramalan yaitu ARIMA, LSTM, dan GRU dalam meramalkan konsentrasi PM10 harian di Jakarta berdasarkan data aktual dari stasiun DKI1. Perbandingan dilakukan menggunakan metrik evaluasi standar yaitu *root mean squared error* (RMSE), *mean absolute error* (MAE), *mean squared error* (MSE), dan *mean absolute percentage error* (MAPE). Hasil penelitian ini diharapkan dapat memberikan rekomendasi model terbaik untuk sistem peramalan kualitas udara operasional di Jakarta.

KAJIAN TEORI

AUTOREGRESSIVE INTEGRATED MOVING AVERAGE

Autoregressive integrated moving average (ARIMA) adalah satu model peramalan data deret waktu yang menggabungkan tiga komponen utama, yaitu *autoregressive* (AR), *integrated* (I), dan *moving average* (MA), yang mana modelnya dinotasikan sebagai $ARIMA(p, d, q)$ di mana p adalah orde AR, d adalah orde *differencing*, dan q adalah orde MA. Ketiga komponen tersebut bekerja secara sinergis untuk menangkap struktur temporal dalam data. Secara matematis, model $ARIMA(p, d, q)$ didefinisikan sebagai berikut (Mahia et al., 2019; Tandon et al., 2022).

$$\phi_p(B)(1 - B)^d Y_t = \mu + \theta_q(B)e_t \quad (1)$$

Dalam persamaan (1), Y_t adalah nilai observasi pada waktu ke- t , B adalah operator *backshift* yang berguna untuk memundurkan indeks waktu, $\phi_p(B)$ adalah polinomial *autoregressive* dengan orde p yang mencerminkan bobot dari nilai-nilai masa lalu, $\theta_q(B)$ adalah polinomial *moving average* dengan orde q yang mencerminkan bobot dari residual masa lalu, dan e_t adalah *white noise* atau kesalahan acak yang mana $e_t \sim N(0, \sigma^2)$ (Khikmah et al., 2023). Operator $(1 - B)^d$ merupakan proses *differencing* yang dilakukan sebanyak d kali untuk menghilangkan tren atau ketidakstasioneran dalam data. Keunggulan utama ARIMA adalah kesederhanaan dan interpretabilitas, tetapi memiliki kelemahan pada asumsi linearitas (Kobiela et al., 2022; M. Zhang, 2018).

LONG SHORT-TERM MEMORY

Long short-term memory (LSTM) adalah salah satu model pembelajaran mesin yang arsitekturnya merupakan perbaikan dari model *recurrent neural network* (RNN) dengan mengatasi masalah *vanishing gradient* ketergantungan jangka panjang. LSTM mengatasi masalah RNN melalui struktur sel memori dengan tiga gerbang yang mengatur aliran informasi secara adaptif, yaitu *input gate*, *forget gate*, dan *output gate* (Huang et al., 2020; Olah, 2015). *Input gate* bertanggung jawab untuk menentukan informasi baru mana yang akan disimpan ke dalam sel memori, *forget gate* menentukan informasi lama mana yang akan dihapus, dan *output gate* menentukan informasi mana yang akan dikeluarkan dari sel memori untuk menghasilkan

keluaran pada waktu tertentu. Secara matematis, mekanisme LSTM dinyatakan melalui serangkaian persamaan. *Input gate* dihitung dengan persamaan (2) (Alizadegan et al., 2025; Khikmah et al., 2025).

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \quad (2)$$

Sedangkan untuk *forget gate* dapat dihitung dengan persamaan (3) berikut.

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \quad (3)$$

dan *output gate* yang dapat dihitung dengan persamaan (4) berikut.

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \quad (4)$$

Sel memori yang diperbarui dapat dihitung melalui persamaan (5) berikut.

$$C_t = f_t \odot C_{t-1} + i_t \odot \tanh(W_c x_t + U_c h_{t-1} + b_c) \quad (5)$$

Sementara keluaran tersembunyi (*hidden state*) dihitung dengan persamaan (6) berikut.

$$h_t = o_t \odot \tanh(C_t) \quad (6)$$

Kemampuan LSTM memiliki keunggulan utama LSTM adalah kemampuannya untuk mempelajari struktur temporal secara otomatis tanpa perlu menentukan *lag* secara eksplisit, namun kelemahannya adalah jumlah parameter yang besar sehingga membutuhkan data yang cukup banyak dan waktu komputasi yang relatif lama (Bouktif et al., 2018; R. Zhang et al., 2021).

GATED RECURRENT UNIT

Gated recurrent unit (GRU) merupakan varian dari LSTM dengan arsitektur yang lebih sederhana namun tetap mempertahankan kemampuan untuk menangkap ketergantungan jangka panjang. Perbedaan utama antara GRU dan LSTM terletak pada penggabungan *input gate* dan *forget gate* menjadi satu gerbang yang disebut *update gate*, serta penggunaan *reset gate* untuk mengontrol seberapa banyak informasi masa lalu yang akan (Farah et al., 2022; Sheng et al., 2023). Arsitektur GRU yang lebih sederhana sehingga dalam proses pelatihan dan memiliki risiko *overfitting* yang lebih rendah. Secara matematis, mekanisme GRU untuk *update gate* yang menentukan seberapa banyak informasi masa lalu yang dipertahankan dapat dihitung melalui persamaan (7) berikut (Pirani et al., 2022).

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \quad (7)$$

Kedua, *reset gate* pada gerbang ini menentukan seberapa banyak informasi masa lalu yang diabaikan dihitung melalui persamaan (8) berikut.

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \quad (8)$$

Ketiga, *candidate hidden state* dihitung melalui persamaan (9) berikut.

$$\bar{h}_t = \tanh(W_h x_t + U_h (r_t \cdot h_{t-1}) + b_h) \quad (9)$$

Terakhir, *hidden state* diperbarui dengan persamaan (10) berikut.

$$h_t = (1 - z_t) \cdot h_{t-1} + z_t \cdot \bar{h}_t \quad (10)$$

Keunggulan GRU dalam hal efisiensi komputasi dan kemampuan generalisasi untuk dataset berukuran besar dengan biaya komputasi yang lebih rendah (Lin et al., 2022).

METRIK EVALUASI

Evaluasi kinerja model peramalan adalah pengukuran seberapa baik model dalam memprediksi nilai masa depan dibandingkan dengan nilai aktual. Empat metrik evaluasi yang digunakan, yaitu *root mean squared error* (RMSE), *mean absolute error* (MAE), *mean squared error* (MSE), dan *mean absolute percentage error* (MAPE). Metrik evaluasi RMSE digunakan untuk mengukur akar dari rata-rata kuadrat selisih antara nilai aktual dan nilai prediksi, dimana dapat dihitung dengan persamaan (11) sebagai berikut (Hodson, 2022).

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (11)$$

MAE mengukur rata-rata nilai absolut selisih antara nilai aktual dan prediksi. MAE lebih mudah diinterpretasikan karena menggunakan skala yang sama dengan data asli dan kurang sensitif terhadap *outlier* dibandingkan RMSE, persamaan MAE dihitung dengan persamaan (12) berikut (Maciejowska et al., 2026).

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (12)$$

MSE merupakan kuadrat dari RMSE. MSE sering digunakan sebagai fungsi *loss* dalam pelatihan model *deep learning* karena sifatnya yang terdiferensiasi dengan baik yang dihitung dengan persamaan (13) berikut (Serefoglu Cabuk et al., 2024).

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (13)$$

MAPE merupakan menyatakan kesalahan peramalan dalam bentuk persentase sehingga mudah dibandingkan antar model yang memiliki skala data berbeda yang dihitung dengan persamaan (14) berikut.

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100\% \quad (14)$$

Menurut kriteria kebaikan nilai MAPE kurang dari 10% mengindikasikan akurasi peramalan yang sangat baik, MAPE antara 10%-20% menunjukkan akurasi yang baik, MAPE antara 20%-50% menunjukkan akurasi yang cukup, dan MAPE lebih dari 50% menunjukkan akurasi yang buruk (Blasco et al., 2013).

METODE

DATA

Data yang digunakan dalam penelitian ini adalah data sekunder terkait konsentrasi PM10 harian di Jakarta, dari Januari 2010 hingga Februari 2025 (5.538 observasi). Statistik deskriptif disajikan pada Tabel 1.

Tabel 1. Statistik Deskriptif Data PM10

Statistik	Nilai
Rata-rata	56,56
Media	53,00
Standar Deviasi	45,23
Minimum	18,00
Maksimum	403,00

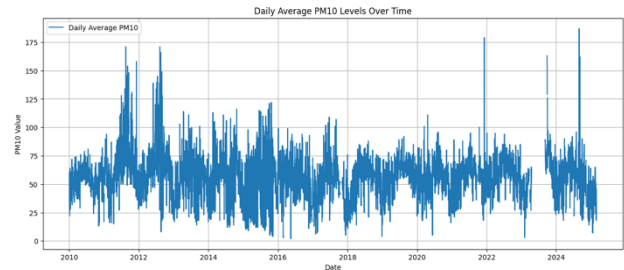
TAHAPAN ANALISIS

Penelitian ini dilaksanakan melalui tujuh tahapan sistematis yang dijelaskan secara rinci sebagai berikut.

1. Tahap 1. Pengumpulan data
2. Tahap 2. Pra-pemrosesan dan eksplorasi awal
3. Tahap 3. Identifikasi pola dan karakteristik data. Analisis stasioneritas data dengan uji *Augmented Dickey-Fuller* (ADF) dan Identifikasi pola autokorelasi melalui plot *autocorrelation function* (ACF), *partial autocorrelation function* (PACF), dan *extended autocorrelation function* (EACF)
4. Tahap 4. Pemodelan *baseline* ARIMA sebagai model *baseline* statistik tradisional.
5. Tahap 5. Pemodelan *deep learning* LSTM dan GRU. Data dinormalisasi ke rentang [0,1] menggunakan *MinMaxScaler*.
6. Tahap 6. Evaluasi, validasi, dan perbandingan model. Perbandingan kinerja ketiga model (ARIMA, LSTM, GRU) menggunakan metrik evaluasi RMSE, MAE, MSE, dan MAPE.

HASIL DAN PEMBAHASAN

Data PM10 yang digunakan dalam penelitian ini adalah data harian dari Januari 2010 hingga Februari 2025 yang telah dilakukan praproses dimana untuk penanganan data hilang dengan *forward filling*, dimana memiliki visualisasi yang ditunjukkan oleh Gambar 1.



Gambar 1. Visualisasi Data Harian PM10

Selanjutnya data hasil pra-pemrosesan dilakukan analisis, yaitu dengan pemodelan ARIMA. Tahap pertama pemodelan dengan ARIMA, yaitu dengan pengujian hipotesis kestasioneran data. Hasil uji *augmented dickey-fuller* (ADF) data konsentrasi PM10 harian di Jakarta menunjukkan statistik uji ADF sebesar -6,147 dimana $p\text{-value } 7,71 \times 10^{-8} < 0.05$, sehingga H_0 ditolak, artinya data PM10 telah stasioner pada level awal ($d = 0$) tanpa memerlukan transformasi *differencing*.

Selanjutnya data ini dilakukan identifikasi kandidat model, dimana identifikasi didasarkan pada pola autokorelasi melalui plot ACF, PACF dan EACF. Model terbaik yang didapatkan adalah *ARIMA*(1,0,1) nilai AIC terkecil. Model yang diperoleh dilakukan estimasi parameter dimana hasil semua parameter signifikan secara statistik pada tingkat signifikansi $\alpha = 5\%$. Hasil estimasi parameter ditunjukkan pada Tabel 2.

Tabel 2. Estimasi Parameter model *ARIMA*(1,0,1)

Parameter	Koefisien	Std. Error	p-value
μ	56,5565	1,1168	0,000
ϕ_1	0,9331	0,0006	0,000
θ_1	-0,7065	0,011	0,000
σ^2	297,3958	4,126	0,000

Berdasarkan hasil estimasi parameter model $ARIMA(1,0,1)$ persamaan model yang diperoleh adalah sebagai berikut.

$$\phi_p(B)(1-B)^d Y_t = \mu + \theta_q(B)e_t$$

$$Y_t = \mu + \phi_1 Y_{t-1} - \theta_1 e_{t-1} + e_t$$

$$Y_t = 56,5565 + 0,9331Y_{t-1} - 0,7065e_{t-1} + e_t$$

Hasil pemodelan di atas menunjukkan bahwa nilai PM10 pada hari ke- t sangat dipengaruhi oleh nilai PM10 pada hari sebelumnya dengan koefisien AR mendekati 1, yang menunjukkan persistensi data yang tinggi, serta kesalahan peramalan pada hari sebelumnya dengan koefisien MA negatif yang berfungsi sebagai koreksi.

Selanjutnya dilakukan diagnostik terhadap sisaan model, dimana ringkasan hasil ujinya tersaji pada Tabel 3 berikut.

Tabel 3. Diagnostik model $ARIMA(1,0,1)$

Uji	P-value	Keputusan	Penjelasan
Ljung-Box (Uji Autokorelasi)	0,000	H_0 Ditolak	Residual masih berautokorelasi, dimana model belum optimal
Jarque-Bera (Normalitas)	0,000	H_0 Ditolak	Residual tidak normal
Heteroskedastisitas	0,000	H_0 Ditolak	Varians residual tidak konstan
Skewness	0,29	-	Distribusi condong ke kanan
Kurtosis	5,97	-	Distribusi lebih runcing dengan ekor tebal

*Tarf signifikansi: $\alpha = 5\%$

Hasil diagnostik sisaan model pada Tabel 3 di atas menunjukkan bahwa model ARIMA belum mampu menangkap karakteristik data PM10 yang kompleks, *non-linear*, dan volatil, sehingga selanjutnya dilakukan pendekatan *deep learning*.

Peramalan kedua yang dilakukan penelitian ini adalah dengan metode *long short-term memory* (LSTM) dan *gated recurrent unit* (GRU). Metode LSTM dan GRU dikembangkan adalah dengan arsitektur *input sequence* 30 hari (*look-back=30*). Dimana

hyperparameter yang digunakan adalah *hyperparameter* inisialisasi dan tersaji pada Tabel 4 berikut.

Tabel 4. Hyperparameter Arsitektur LSTM dan GRU

Hyperparameter	Nilai
Hidden Layer	1
Unit/neuron	50
Dense output layer	1
Optimizer	Adam
Epoch	100
Batch size	64

Data yang digunakan untuk peramalan dengan LSTM dan GRU sebelum pelatihan dilakukan normalisasi ke rentang [0,1] menggunakan metode *MinMaxScaler*. Hal ini dilakukan untuk menyeragamkan skala *input*.

Data yang telah dilakukan normalisasi selanjutnya dilakukan proses pelatihan. Dimana hasil evaluasi pelatihan metode GRU pada data uji menunjukkan kinerja yang sedikit lebih baik dibandingkan LSTM pada semua metrik evaluasi. Secara umum perbandingan evaluasi ketiga metode yang digunakan tersaji pada Tabel 5 berikut.

Tabel 5. Ringkasan Kinerja ARIMA, LSTM, dan GRU

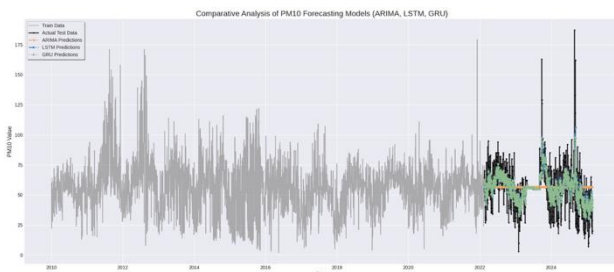
Metrik Evaluasi	Metode		
	ARIMA	LSTM	GRU
RMSE	17,526	13,441	13,366
MAE	12,416	8,958	8,918
MSE	307,163	180,674	178,647
MAPE (%)	31,228	21,361	21,211

Perbandingan kinerja ketiga metode peramalan menunjukkan bahwa metode *deep learning* (LSTM dan GRU) memiliki hasil evaluasi yang lebih baik dibandingkan ARIMA. Model ARIMA sebagai *baseline* hanya memiliki nilai MAPE sebesar 31,228% yang termasuk dalam kategori akurasi cukup berdasarkan kriteria. Keterbatasan ARIMA ini terletak pada asumsi linearitasnya, sehingga model gagal memprediksi fluktuasi ekstrem pada data PM10.

Sementara itu, metode LSTM berhasil meningkatkan kinerja dengan nilai MAPE sebesar 21,361% atau turun sebesar 9,87% dari ARIMA. Peningkatan terbesarnya ditunjukkan pada metrik MSE bahwa LSTM lebih baik dalam meminimalkan

error kuadrat besar, yang penting untuk menghindari kesalahan prediksi pada kondisi polusi tinggi, kemampuan tersebut mampu menangkap pola *non-linear* dan ketergantungan jangka panjang melalui mekanisme tiga gerbang adaptif.

Selanjutnya, model GRU menunjukkan kinerja terbaik di antara ketiga model. Dibandingkan LSTM, GRU memiliki RMSE lebih rendah 0,075 poin, MAE lebih rendah 0,040 poin, MSE lebih rendah 2,027 poin, dan MAPE lebih rendah 0,15%. Meskipun perbedaannya relatif kecil, konsistensi keunggulan GRU pada semua metrik mengindikasikan bahwa arsitektur GRU yang lebih sederhana dimana gabungan gerbang *input* dan *forget* menjadi satu *update gate* sehingga, lebih sesuai untuk data PM10 karena memiliki jumlah parameter lebih sedikit, risiko *overfitting* lebih rendah, serta generalisasi yang lebih baik. Oleh karena itu, GRU direkomendasikan sebagai model terbaik pada penelitian ini untuk peramalan PM10 harian di Jakarta. Hasil validasi terhadap peramalan data PM10 di Jakarta ini juga dapat dilihat melalui Gambar 2.



Gambar 2. Visualisasi Hasil Peramalan Data Harian PM10

PENUTUP

SIMPULAN

Berdasarkan hasil analisis di atas perbandingan kinerja model ARIMA, LSTM, dan GRU dalam meramalkan konsentrasi PM10 harian di Jakarta, dapat disimpulkan bahwa model deep learning (LSTM dan GRU) secara signifikan memberikan hasil peramalan yang lebih baik dibandingkan model statistik tradisional ARIMA, dengan GRU sebagai model terbaik yang mencapai RMSE = 13,366, MAE = 8,918, MSE = 178,647, dan MAPE = 21,211%, diikuti LSTM (MAPE = 21,361%), sementara ARIMA hanya mencapai MAPE = 31,228%.

SARAN

Penelitian selanjutnya disarankan untuk menambahkan variabel eksogen seperti suhu, kelembaban, kecepatan angin, dan data lalu lintas untuk menguji coba arsitektur yang lebih kompleks seperti stacked LSTM/GRU, serta menerapkan metodologi ini pada stasiun pemantauan lain maupun kota-kota lain di Indonesia guna memperluas generalisasi hasil.

DAFTAR PUSTAKA

- Alizadegan, H., Rashidi Malki, B., Radmehr, A., Karimi, H., & Ilani, M. A. (2025). Comparative study of long short-term memory (LSTM), bidirectional LSTM, and traditional machine learning approaches for energy consumption prediction. *Energy Exploration & Exploitation*, 43(1), 281–301.
- Blasco, B. C., Moreno, J. J. M., Pol, A. P., & Abad, A. S. (2013). Using the R-MAPE index as a resistant measure of forecast accuracy. *Psicothema*, 25(4), 500–506.
- Bouktif, S., Fiaz, A., Ouni, A., & Serhani, M. A. (2018). Optimal deep learning lstm model for electric load forecasting using feature selection and genetic algorithm: Comparison with machine learning approaches. *Energies*, 11(7), 1636.
- Farah, S., Humaira, N., Aneela, Z., & Steffen, E. (2022). Short-term multi-hour ahead country-wide wind power prediction for Germany using gated recurrent unit deep learning. *Renewable and Sustainable Energy Reviews*, 167, 112700.
- Hodson, T. O. (2022). Root mean square error (RMSE) or mean absolute error (MAE): When to use them or not. *Geoscientific Model Development Discussions*, 2022, 1–10.
- Huang, L., Cai, T., Zhu, Y., Zhu, Y., Wang, W., & Sun, K. (2020). LSTM-based forecasting for urban construction waste generation. *Sustainability*, 12(20), 8555.
- Khikmah, K. N., Sadik, K., & Indahwati, I. (2023). TRANSFER FUNCTION AND ARIMA MODEL FOR FORECASTING BI RATE IN INDONESIA. *BAREKENG: Jurnal Ilmu Matematika Dan Terapan*, 17(3), 1359–1366.
- Khikmah, K. N., Sadik, K., & Notodiputro, K. A. (2025). Simulation and Empirical Studies of Long Short-Term Memory Performance to Deal with Limited Data. *Jurnal Online Informatika*, 10(1), 216–226.
- Khikmah, K. N., & Sofro, A. (2025). PERBANDINGAN KINERJA MODEL ARIMA

- DAN LSTM UNTUK PERAMALAN CURAH HUJAN DI KALIMANTAN TENGAH. *Jurnal Riset Dan Aplikasi Matematika (JRAM)*, 9(2), 199–207.
- Kobiela, D., Krefta, D., Król, W., & Weichbroth, P. (2022). ARIMA vs LSTM on NASDAQ stock exchange data. *Procedia Computer Science*, 207, 3836–3845.
- Ławryńczuk, M., & Zarzycki, K. (2025). LSTM and GRU type recurrent neural networks in model predictive control: A Review. *Neurocomputing*, 632, 129712.
- Lin, H., Gharehbaghi, A., Zhang, Q., Band, S. S., Pai, H. T., Chau, K.-W., & Mosavi, A. (2022). Time series-based groundwater level forecasting using gated recurrent unit deep neural networks. *Engineering Applications of Computational Fluid Mechanics*, 16(1), 1655–1672.
- Maciejowska, K., Lipiecki, A., & Uniejewski, B. (2026). Statistical and economic evaluation of forecasts in electricity markets: Beyond RMSE and MAE. *Energy Conversion and Management*, 356, 121408.
- Mahia, F., Dey, A. R., Masud, M. A., & Mahmud, M. S. (2019). Forecasting electricity consumption using ARIMA model. *2019 International Conference on Sustainable Technologies for Industry 4.0 (STI)*, 1–6.
- Olah, C. (2015). *Understanding LSTM Networks*. <https://Colah.Github.io/Posts/2015-08-Understanding-LSTMs/>.
- Perdana, A. R., Pangastuti, A. I., & Haryanto, Y. D. (2023). Analisis Konsentrasi PM10 dan PM2. 5 Pada Titik Pemantauan Bundaran Hi Jakarta Pusat Peri-Ode Data Februari-Oktober 2021. *Jurnal Samudra Geografi*, 6(1), 1–8.
- Pierre, A. A., Akim, S. A., Semenyio, A. K., & Babiga, B. (2023). Peak electrical energy consumption prediction by ARIMA, LSTM, GRU, ARIMA-LSTM and ARIMA-GRU approaches. *Energies*, 16(12), 4739.
- Pirani, M., Thakkar, P., Jivrani, P., Bohara, M. H., & Garg, D. (2022). A comparative analysis of ARIMA, GRU, LSTM and BiLSTM on financial time series forecasting. *2022 IEEE International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE)*, 1–6.
- Serefoglu Cabuk, K., Cengiz, S. K., Guler, M. G., Topcu, H., Cetin Efe, A., Ulas, M. G., & Poslu Karademir, F. (2024). Chasing the objective upper eyelid symmetry formula; R2, RMSE, POC, MAE, and MSE. *International Ophthalmology*, 44(1), 303.
- Sheng, Z., Wen, S., Feng, Z.-K., Shi, K., & Huang, T. (2023). A novel residual gated recurrent unit framework for runoff forecasting. *IEEE Internet of Things Journal*, 10(14), 12736–12748.
- Siame-Namini, S., Tavakoli, N., & Namin, A. S. (2018). A comparison of ARIMA and LSTM in forecasting time series. *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 1394–1401.
- Tandon, H., Ranjan, P., Chakraborty, T., & Suhag, V. (2022). Coronavirus (COVID-19): ARIMA-based Time-series Analysis to Forecast near Future and the Effect of School Reopening in India. *Journal of Health Management*, 09720634221109087.
- Wellid, I., Simbolon, L. M., Falahuddin, M. A., Nurfitriani, N., Sumeru, K., bin Sukri, M. F., & Yuningsih, N. (2024). Evaluasi polusi udara PM2. 5 dan PM10 di kota Bandung serta kaitannya dengan infeksi saluran pernafasan akut. *Jurnal Kesehatan Lingkungan Indonesia*, 23(2), 128–136.
- Zhang, M. (2018). Time series: Autoregressive models ar, ma, arma, arima. *University of Pittsburgh*.
- Zhang, R., Guo, Z., Meng, Y., Wang, S., Li, S., Niu, R., Wang, Y., Guo, Q., & Li, Y. (2021). Comparison of ARIMA and LSTM in forecasting the incidence of HFMD combined and uncombined with exogenous meteorological variables in Ningbo, China. *International Journal of Environmental Research and Public Health*, 18(11), 6174.