

Pemilihan Neuron LSTM dan LSTM Bayesian Optimization Untuk Prediksi Curah Hujan Bulanan Berbasis Iklim

Muhammad Luthfan Hawali¹, Walid²

¹Matematika, Fakultas Matematika dan IPA, Universitas Negeri Semarang

¹luthfan76@students.unnes.ac.id*, ²walid.mat@mail.unnes.ac.id

Abstract

Accurate rainfall prediction is crucial for supporting climate resilience and mitigating hydrometeorological risks in tropical regions like Indonesia. This study develops a monthly rainfall prediction model using Long Short-Term Memory (LSTM) optimized with Bayesian Optimization (BO). The data used include monthly rainfall (1983–2024) from three meteorological stations in Java—Tunggul Wulung (Cilacap), Tegal, and Ahmad Yani (Semarang)—as well as global predictor variables such as SOI, Nino 3.4, IOD, WNPMI, AUSMI, and SST. The LSTM model consists of four nested layers with Dropout and Batch Normalization to prevent overfitting. BO is used to determine the optimal hyperparameter combination at each layer of the network. Evaluation using Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE) shows that LSTM-BO provides improved performance compared to standard LSTM at all three locations. In Cilacap, the RMSE decreased from 160.51 mm to 148.87 mm, and the MAE from 126.77 mm to 115.14 mm. In Tegal, the RMSE decreased from 89.00 mm to 86.19 mm, and the MAE from 65.73 mm to 58.84 mm. In Semarang, the RMSE decreased from 104.05 mm to 100.21 mm, and the MAE from 76.32 mm to 70.60 mm. The integration of time series, deep learning, and probabilistic optimization resulted in a more optimal model. LSTM tended to be stable and consistent, with results close to observations in most months, while LSTM-BO excelled in certain months or locations, although in some conditions its predictions were further from observations than LSTM.

Keywords: bayesian optimization, kalman filter, LSTM, rainfall prediction.

Abstrak

Prediksi curah hujan yang akurat penting untuk mendukung ketahanan iklim dan mitigasi risiko hidrometeorologis di wilayah tropis seperti Indonesia. Penelitian ini mengembangkan model prediksi curah hujan bulanan menggunakan Long Short-Term Memory (LSTM) yang dioptimasi dengan Bayesian Optimization (BO). Data yang digunakan meliputi curah hujan bulanan (1983–2024) dari tiga stasiun meteorologi di Pulau Jawa—Tunggul Wulung (Cilacap), Tegal, dan Ahmad Yani (Semarang)—serta variabel prediktor global seperti SOI, Nino 3.4, IOD, WNPMI, AUSMI, dan SST. Model LSTM terdiri dari empat lapisan bertingkat dengan Dropout dan Batch Normalization untuk mencegah overfitting. BO digunakan untuk menentukan kombinasi hiperparameter optimal pada setiap lapisan jaringan. Evaluasi menggunakan Root Mean Squared Error (RMSE) dan Mean Absolute Error (MAE) menunjukkan bahwa LSTM-BO memberikan peningkatan kinerja dibanding LSTM standar di ketiga lokasi. Di Cilacap, RMSE menurun dari 160,51 mm menjadi 148,87 mm, dan MAE dari 126,77 mm menjadi 115,14 mm. Di Tegal, RMSE turun dari 89,00 mm menjadi 86,19 mm, dan MAE dari 65,73 mm menjadi 58,84 mm. Di Semarang, RMSE berkurang dari 104,05 mm menjadi 100,21 mm, dan MAE dari 76,32 mm menjadi 70,60 mm. Integrasi time series, deep learning, dan optimasi probabilistik menghasilkan model yang lebih optimal. LSTM cenderung stabil dan konsisten dengan hasil mendekati observasi di sebagian besar bulan, sedangkan LSTM-BO unggul pada bulan atau lokasi tertentu meskipun pada beberapa kondisi prediksinya lebih jauh dari observasi dibandingkan LSTM.

Kata kunci: bayesian optimization, kalman filter, LSTM, prediksi curah hujan.

©This work is licensed under a Creative Commons Attribution -ShareAlike 4.0 International License

1. Pendahuluan

Curah hujan merupakan elemen penting dalam sistem iklim yang berdampak langsung terhadap berbagai sektor kehidupan manusia, khususnya pertanian, pengelolaan sumber daya air, dan mitigasi bencana hidrometeorologis. Ketepatan dalam memprediksi curah hujan menjadi aspek strategis yang diperlukan untuk merespons ketidakpastian iklim, terutama di kawasan tropis yang sangat sensitif terhadap perubahan

atmosferik. Analisis deret waktu memberikan pendekatan sistematis yang memungkinkan identifikasi pola dari data historis dan digunakan dalam proyeksi masa depan [1].

Data mining menyediakan kerangka kerja yang efektif untuk mengekstraksi informasi berharga dari volume data yang besar dan kompleks. Dalam konteks prediksi curah hujan, pendekatan ini memungkinkan identifikasi pola tersembunyi dalam data iklim, yang

mendukung pengambilan keputusan berbasis bukti [2]. *Machine learning* sebagai salah satu pilar dalam data mining memiliki kemampuan untuk menangani data multivariat dan nonlinier, serta mampu melakukan generalisasi terhadap pola data baru tanpa intervensi manusia secara langsung [3]. Model yang dibangun dari teknik ini mampu menghadirkan prediksi yang adaptif, relevan terhadap perubahan dinamis, dan bersifat skalabel terhadap data berukuran besar.

Model jaringan saraf tiruan *Long Short-Term Memory* (LSTM) merupakan bentuk arsitektur *deep learning* yang dirancang untuk mengatasi keterbatasan *Recurrent Neural Network* (RNN) dalam menangani ketergantungan jangka panjang. Keberadaan mekanisme gerbang dan sel memori internal dalam LSTM menjadikannya mampu menyimpan dan memproses informasi dari masa lalu secara lebih efektif [4]. Karakteristik ini menjadikan LSTM sangat relevan dalam konteks prediksi iklim yang memiliki dinamika temporal kompleks.

Kinerja optimal dari model LSTM sangat dipengaruhi oleh konfigurasi *hyperparameter*. Pemilihan kombinasi parameter yang tidak tepat dapat menurunkan performa model secara signifikan. Proses pencarian kombinasi parameter terbaik memerlukan pendekatan sistematis dan efisien seperti *Bayesian Optimization* (BO). Metode ini menggunakan pendekatan probabilistik—umumnya *Gaussian Process* (GP)—untuk memodelkan fungsi objektif dan memandu pencarian parameter dengan mempertimbangkan hasil evaluasi sebelumnya [5], [6]. *Bayesian Optimization* tidak hanya mempercepat proses pencarian, tetapi juga meminimalkan eksplorasi terhadap konfigurasi yang tidak menjanjikan, sehingga lebih efisien dibandingkan dengan *grid search* atau *random search* [7].

Bayesian Optimization terbukti mampu meningkatkan akurasi prediksi LSTM dalam berbagai studi, termasuk dalam konteks prediksi curah hujan bulanan. Penelitian oleh Cabrera et al. menunjukkan bahwa integrasi antara LSTM dan *Bayesian Optimization* berhasil meningkatkan akurasi prediksi hingga 84,4% pada data iklim tropis di wilayah pegunungan Andes [8]. Selain *Bayesian Optimization*, Lathika dan Sheeba menemukan bahwa model prediksi hujan berbasis *Bayesian Optimization* tidak hanya memberikan error yang lebih rendah tetapi juga menunjukkan kestabilan terhadap data cuaca yang sangat fluktuatif [9]. Keunggulan ini mengindikasikan bahwa *Bayesian Optimization* tidak hanya meningkatkan performa model, tetapi juga mengurangi risiko overfitting dengan secara adaptif mengevaluasi dan memprioritaskan kombinasi parameter yang paling relevan [5], [9].

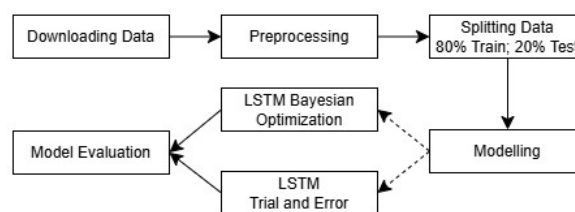
Pemanfaatan variabel prediktor tambahan dari indikator iklim global seperti *Southern Oscillation Index* (SOI), Anomali Nino 3.4, *Indian Ocean Dipole* (IOD), dan *Sea Surface Temperature* (SST) telah

terbukti meningkatkan ketepatan model prediksi curah hujan. Variabilitas iklim global yang dipengaruhi oleh ENSO dan IOD memberikan kontribusi signifikan terhadap fluktuasi curah hujan di wilayah tropis, termasuk Indonesia [10]. Nilai negatif SOI menunjukkan fase El Niño yang sering diasosiasikan dengan penurunan curah hujan di Indonesia, sedangkan nilai positifnya menunjukkan La Niña yang meningkatkan presipitasi [11]. Nino 3.4, yang mengukur anomali suhu permukaan laut di Pasifik tengah, memainkan peran penting dalam modulasi curah hujan musiman. Kontribusi signifikan dari indikator IOD terhadap curah hujan di Indonesia bagian barat tercermin melalui pergeseran suhu di Samudra Hindia yang mempengaruhi sirkulasi atmosfer regional [12]. Integrasi variabel-variabel ini ke dalam model prediksi memungkinkan representasi dinamika atmosfer yang lebih akurat.

Kombinasi antara analisis deret waktu, model *deep learning* berbasis LSTM, serta teknik optimasi seperti *Bayesian Optimization* menawarkan pendekatan yang kuat dalam membangun sistem prediksi curah hujan yang andal. Model prediktif yang dikembangkan melalui integrasi pendekatan ini dapat mendukung proses mitigasi perubahan iklim, pengelolaan sumber daya alam, serta perencanaan adaptif dalam menghadapi ketidakpastian cuaca yang semakin kompleks.

2. Metode Penelitian

Penelitian ini menggunakan pendekatan kuantitatif dengan metode LSTM dan LSTM yang dioptimasi menggunakan *Bayesian Optimization* untuk memodelkan dan memprediksi curah hujan bulanan berdasarkan data time series serta indikator iklim terkait. Desain penelitian sistematis mengenai tahapan-tahapan yang dilakukan selama proses penelitian sebagaimana disajikan dalam Gambar 1.



Gambar 1. Skema Penelitian

2.1. Data dan Sumber Data

Data yang diperoleh dari data online BMKG berbentuk file CSV, berisi curah hujan harian selama satu bulan dalam rentang 42 tahun (1983–2024) [13]. Dataset ini mencakup data curah hujan bulanan di Stasiun Meteorologi Tunggul Wulung (Cilacap), Stasiun Meteorologi Maritim Tegal (Tegal), dan Stasiun Meteorologi Maritim Tanjung Emas (Semarang).

Data iklim yang diperoleh dari berbagai lembaga penelitian internasional digunakan sebagai variabel prediktor dalam penelitian ini. Indeks Osilasi Selatan

(*Southern Oscillation Index*, SOI) diperoleh dari Bureau of Meteorology Australia [14], sementara data ENSO 3.4 diambil dari NOAA *Climate Prediction Center* [15]. Indeks Muson Pasifik Barat Laut (*Western North Pacific Monsoon Index*, WNPMI) dan Indeks Muson Australia (Australian Monsoon Index, AUSMI) berasal dari NOAA Physical Sciences Laboratory [16], yang menyediakan data reanalisis atmosfer global. *Data Dipole Mode Index* (Indian Ocean Dipole, IOD) diperoleh dari NOAA *Climate Prediction Center* [17]. Sementara itu, data Suhu Permukaan Laut (*Sea Surface Temperature*, SST) diambil dari COBE2 dataset, merupakan suhu permukaan laut secara global [18].

2.2. Preprocessing Data

Data yang diperoleh belum tentu bersih sehingga perlu melalui tahap prapemrosesan, yang mencakup pengisian data kosong menggunakan Kalman filter, agregasi data menjadi skala bulanan, dan penggabungan dengan data prediktor. Setelah itu, dilakukan normalisasi dan pembagian data menjadi data latih dan data uji dengan rasio 80:20 agar model dapat dijalankan.

Data yang diperoleh memiliki nilai yang hilang (missing value) yang dapat dilihat melalui fungsi `df.isnull().sum()`, yang menunjukkan jumlah data kosong pada setiap kolom, sebagai berikut.

Date	0
RR_Clep	1251
RR_Tgl	1064
RR_Smg	203

Kalman Filter (KF) digunakan dalam penelitian ini untuk mengisi data kosong berdasarkan pendekatan yang telah banyak diterapkan dalam berbagai bidang, termasuk pemrosesan data iklim. Metode ini bekerja secara rekursif melalui dua tahap utama, yaitu prediksi dan pembaruan, untuk memperkirakan nilai yang hilang dengan mempertimbangkan ketidakpastian data [19]. Dalam konteks estimasi presipitasi, Kalman Filter terbukti mampu meningkatkan akurasi meskipun dapat menimbulkan sedikit latensi pada beberapa produk [20].

Algoritma Kalman Filter Kalman Filter terdiri dari dua langkah utama. Tahap Prediksi, dilakukan estimasi terhadap keadaan saat ini berdasarkan keadaan sebelumnya dalam kondisi ketidakpastian.

Prediksi Keadaan

$$\hat{x}_{k|k-1} = F_{k-1} \hat{x}_{k-1|k-1} \quad (1)$$

Prediksi Kovarians Kesalahan

$$P_{k|k-1} = F_{k-1} P_{k-1|k-1} F_{k-1}^T + Q_k \quad (2)$$

Tahap Pembaruan, setelah pengukuran baru diperoleh, estimasi sebelumnya diperbarui menggunakan rata-rata tertimbang secara aritmetika.

Matriks Gain Kalman

$$K_k = P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T + R_k)^{-1} \quad (3)$$

Perbarui Estimasi Keadaan

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k (z_k - H_k \hat{x}_{k|k-1}) \quad (4)$$

Perbarui Kovarians Kesalahan

$$P_{k|k} = (I - K_k H_k) P_{k|k-1} \quad (5)$$

$\hat{x}_{k|k-1}$ merupakan estimasi keadaan sebelum pembaruan, sedangkan $P_{k|k-1}$ menunjukkan kovarians kesalahan dari prediksi keadaan. Nilai K_k adalah matriks gain Kalman yang digunakan untuk menyesuaikan prediksi berdasarkan pengukuran aktual. Estimasi keadaan setelah pembaruan dilambangkan dengan $\hat{x}_{k|k}$, dan $P_{k|k}$ merupakan matriks kovarians kesalahan setelah pembaruan. Parameter sistem yang digunakan dalam proses ini mencakup F_k , H_k , Q_k , dan R_k yang masing-masing merepresentasikan model transisi keadaan, model pengukuran, kovarians noise proses, dan kovarians noise pengukuran.

Normalisasi data merupakan langkah penting dalam analisis data dan *machine learning* yang bertujuan untuk memastikan bahwa setiap fitur memiliki skala yang konsisten dan berada dalam rentang yang sama. Dalam penelitian ini, teknik *Min-Max Scaling* dipilih sebagai metode normalisasi utama. Teknik ini menyesuaikan nilai-nilai dalam dataset sehingga berada dalam rentang antara 0 hingga 1, meskipun rentang ini dapat disesuaikan dengan kebutuhan analisis tertentu. Normalisasi dilakukan menggunakan formula yang sesuai untuk mengubah nilai setiap variabel ke dalam skala yang seragam. Proses ini menggunakan *MinMaxScaler* yang terdapat dalam library `sklearn.preprocessing`.

Teknik normalisasi, khususnya *Min-Max Normalization*, digunakan untuk mengubah data ke dalam rentang yang diinginkan, yaitu antara 0 dan 1. Proses ini dilakukan dengan persamaan sebagai berikut.

$$x' = \frac{x - X_{min}}{X_{max} - X_{min}} \quad (6)$$

Pada persamaan (6) x' adalah nilai data yang telah dinormalisasi, x adalah nilai data yang akan dinormalisasi, X_{max} adalah nilai maksimum dari variabel, dan X_{min} adalah nilai minimum dari variabel.

2.3. Modelling

2.3.1. LSTM

Model LSTM pada penelitian ini terdiri dari 4 lapis LSTM bertingkat, yang memungkinkan model menangkap pola jangka panjang dan memahami hubungan jangka pendek dalam data sekuensial. Struktur bertingkat ini diikuti oleh BatchNormalization untuk menstabilkan nilai aktivasi dan mempercepat proses pelatihan, serta dropout sebesar 0.2 untuk mengurangi risiko *overfitting*. Setelah melalui seluruh lapisan LSTM, terdapat dua lapisan *dense*, yaitu satu lapisan dengan fungsi aktivasi ReLU dan satu lapisan output yang digunakan untuk menghasilkan prediksi numerik. Proses pelatihan dilakukan selama maksimal 120 *epoch* dan *batch size* 32.

2.3.2 LSTM Bayesian Optimization

LSTM Bayesian Optimization adalah teknik optimasi yang digunakan untuk menemukan jumlah unit terbaik pada setiap lapisan dalam model *Long Short-Term Memory* (LSTM) secara otomatis. Pendekatan ini menggunakan *Bayesian Optimization*, yang merupakan metode berbasis probabilistik untuk mengoptimalkan fungsi objektif dengan jumlah evaluasi yang minimal. *Bayesian Optimization* membantu mencari kombinasi terbaik dari jumlah unit pada setiap lapisan dengan mempertimbangkan performa model, seperti nilai loss atau akurasi.

Proses optimasi ini dilakukan dengan menggunakan *Gaussian Process* atau metode berbasis *Tree-structured Parzen Estimator* (TPE) untuk membangun model probabilistik dari fungsi objektif. Penggunaan strategi ini, *Bayesian Optimization* mengeksplorasi ruang hyperparameter secara lebih efisien dibandingkan pendekatan berbasis grid search atau random search. Hasilnya, model LSTM yang dihasilkan memiliki arsitektur yang lebih optimal, dengan jumlah unit per lapisan yang disesuaikan untuk memberikan performa terbaik pada tugas prediksi atau klasifikasi yang diberikan. Model yang dibangun terdiri dari empat lapisan LSTM, di mana jumlah unit pada setiap lapisan dipilih dalam rentang tertentu berdasarkan hasil tuning. Setiap lapisan LSTM menggunakan fungsi aktivasi yang dapat berupa ReLU atau tanh, dan diikuti dengan Dropout (0.2) untuk mencegah *overfitting*. Setelah lapisan LSTM terakhir, model memiliki dua lapisan Dense, yaitu Dense (64, activation='relu') untuk menangkap pola kompleks, dan Dense (1) sebagai output layer yang menghasilkan prediksi nilai tunggal. Model dikompilasi menggunakan *Adam optimizer*, di mana *learning rate* dipilih dari beberapa nilai kandidat (0.001, 0.005, atau 0.01) berdasarkan hasil *Bayesian Optimization*. Fungsi loss yang digunakan adalah *mean squared error* (MSE) untuk mengukur kesalahan prediksi.

Konfigurasi *hyperparameter* terbaik ditentukan dengan menggunakan *Bayesian Optimization*, yang dilakukan dengan 40 percobaan, di mana setiap percobaan mengeksekusi model sekali (*executions_per_trial*=1).

Model dilatih dengan 120 *epoch*, menggunakan *batch size* 32, dan validasi sebesar 20% dari data pelatihan.

Tabel 1. Pemilihan *Hyperparameter LSTM Bayesian Optimization*

Hyperparameter	Nilai
units_1	100 – 300 (step 50)
activation_1	'relu' / 'tanh'
Dropout	0.2
units_2	50 – 200 (step 50)
activation_2	'relu' / 'tanh'
Dropout	0.2
units_3	10 – 100 (step 10)
activation_3	'relu' / 'tanh'
Dropout	0.2
units_4	10 – 50 (step 10)
activation_4	'relu' / 'tanh'
Dropout	0.2
Learning Rate	0.001 / 0.005 / 0.01
Batch Size	32 (fix)
Epochs	120 (fix)
max_trials	30
executions per trial	1

2.4. Model Evaluation

Efektivitas model dalam memberikan prediksi yang akurat pada prediksi curah hujan bulanan dievaluasi menggunakan metrik MAE (*Mean Absolute Error*) dan RMSE (*Root Mean Squared Error*). MAE mengukur rata-rata selisih absolut antara nilai aktual dan nilai prediksi, tanpa memperhatikan arah kesalahan [17]. Metrik ini bersifat intuitif serta tidak terlalu terpengaruh oleh keberadaan pencilan (*outlier*), sehingga sering digunakan saat kestabilan prediksi menjadi prioritas. Sebaliknya, RMSE menghitung akar dari rata-rata kuadrat selisih antara nilai aktual dan nilai prediksi, dengan memberikan penalti lebih besar pada kesalahan yang besar. Oleh karena itu, RMSE lebih sesuai ketika ketepatan estimasi menjadi hal yang krusial. Semakin rendah nilai MAE dan RMSE, semakin kecil kesalahan prediksi yang dihasilkan model, yang mencerminkan tingkat akurasi yang lebih tinggi.

$$MAE = \frac{1}{n} \sum_i^n |y_i - \tilde{y}_i| \quad (7)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_i^n (y_i - \tilde{y}_i)^2} \quad (8)$$

Nilai MAE dihitung menggunakan (7), sedangkan nilai RMSE dihitung menggunakan (8). Simbol \tilde{y}_i merupakan nilai prediksi, y_i sebagai nilai nyata, dan n Jumlah data. Perbandingan antara MAE dan RMSE menunjukkan karakter distribusi kesalahan. Jika RMSE jauh lebih besar dari MAE, kemungkinan terdapat outlier. Jika nilainya mendekati, berarti kesalahan prediksi model cenderung konsisten.

3. Hasil dan Pembahasan

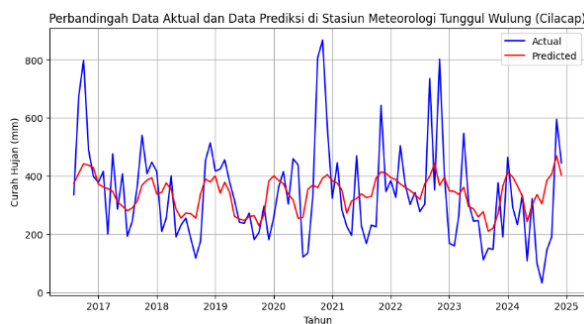
Konfigurasi jumlah neuron pada setiap lapisan LSTM ditentukan melalui dua pendekatan, yaitu *trial and error* dan optimasi menggunakan *Bayesian Optimization*. Tabel 2 menyajikan konfigurasi neuron yang didapatkan melalui pendekatan *trial and error*, sedangkan Tabel 3 menunjukkan hasil konfigurasi optimal yang diperoleh dari proses Bayesian Optimization. Masing-masing konfigurasi kemudian digunakan untuk membangun model yang dievaluasi berdasarkan hasil prediksi terhadap data aktual dan metrik evaluasi seperti RMSE dan MAE.

Tabel 2. Konfigurasi Neuron Model LSTM

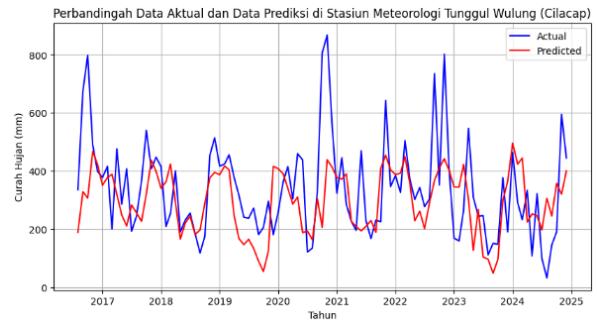
Layer	Neuron	Activation Function
1	128	ReLU
2	64	ReLU
3	32	ReLU
4	16	ReLU
Learning Rate	0.001	

Tabel 3. Konfigurasi Model LSTM-Bayesian Optimization

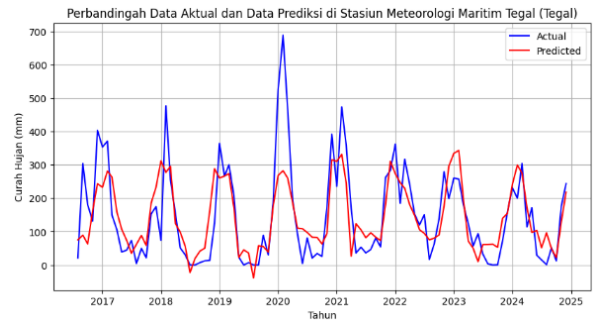
Layer	Cilacap		Tegal		Semarang	
	Neuron	Activation Function	Neuron	Activation Function	Neuron	Activation Function
1	300	ReLU	300	Tanh	300	ReLU
2	150	ReLU	50	ReLU	200	ReLU
3	40	Tanh	10	ReLU	30	Tanh
4	50	Tanh	40	ReLU	30	ReLU
Learning Rate	0.001		0.005		0.005	



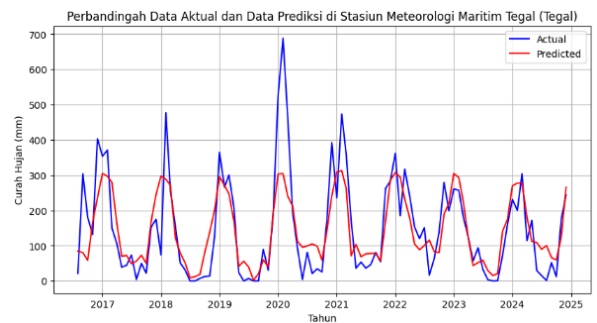
Gambar 2. Hasil Prediksi Data Test Model LSTM Lokasi Cilacap



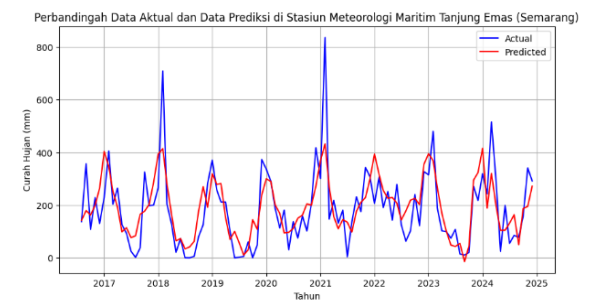
Gambar 3. Hasil Prediksi Data Test Model LSTM-BO Lokasi Cilacap



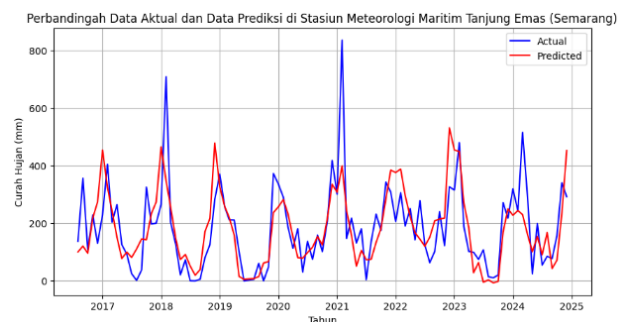
Gambar 4. Hasil Prediksi Data Test Model LSTM Lokasi Tegal



Gambar 5. Hasil Prediksi Data Test Model LSTM-BO Lokasi Tegal



Gambar 6. Hasil Prediksi Data Test Model LSTM Lokasi Semarang

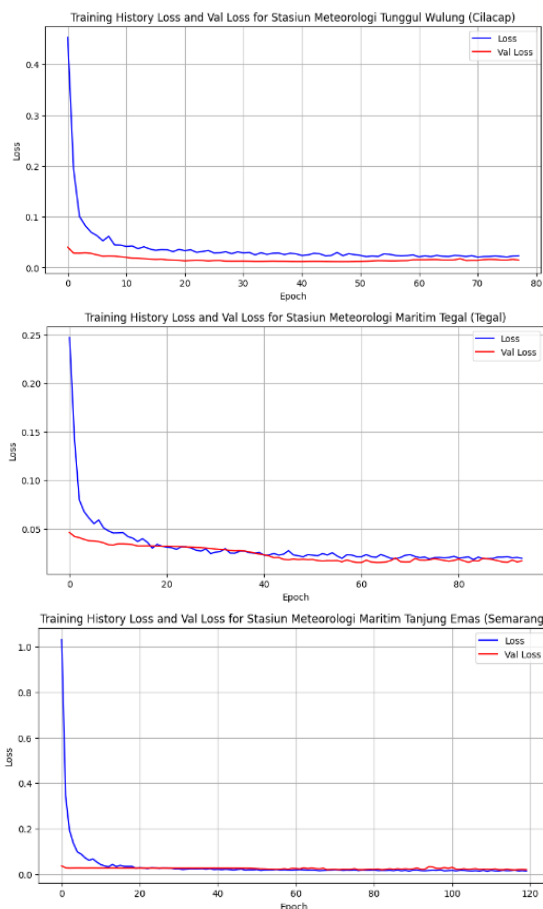


Gambar 7. Hasil Prediksi Data Test Model LSTM-BO Lokasi Semarang

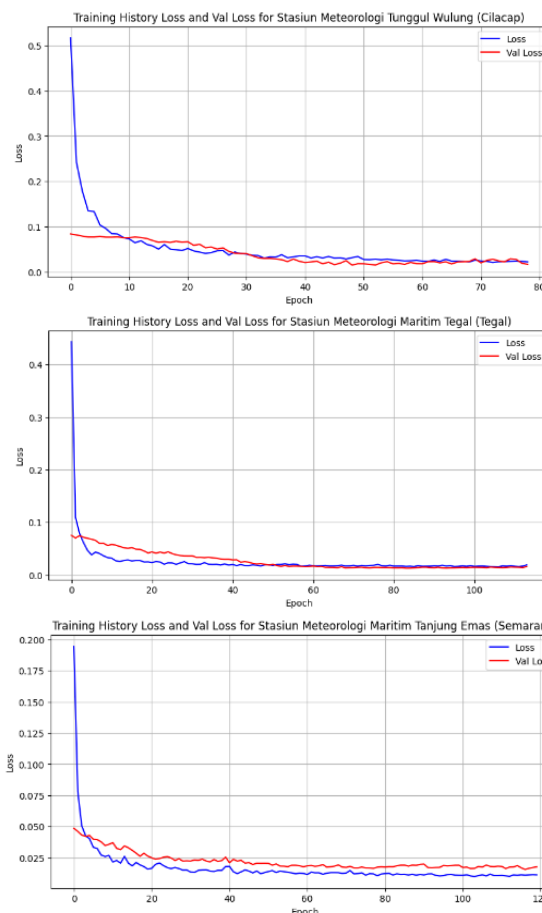
Gambar 2 merupakan hasil dari model LSTM dengan pendekatan *trial and error* yang tampak menghasilkan prediksi yang terlalu rata dan kurang mampu mengikuti lonjakan curah hujan ekstrem, terutama pada tahun 2017 dan 2021. Gambar 3 menunjukkan performa model LSTM yang telah dioptimasi menggunakan *Bayesian Optimization*, yang lebih adaptif terhadap fluktuasi musiman, meskipun masih terdapat underestimasi pada beberapa puncak curah hujan.

Gambar 4 dan gambar 5 menampilkan hasil prediksi di Stasiun Meteorologi Maritim Tegal. Model LSTM dan LSTM *Bayesian Optimization* keduanya menunjukkan kemampuan dalam merepresentasikan pola musiman. Namun, keduanya masih belum mampu menangkap lonjakan ekstrem secara akurat, khususnya pada awal tahun 2020, yang menyebabkan prediksi lebih rendah dibandingkan data aktual pada periode tersebut.

Gambar 6 model LSTM berbasis *trial and error* memberikan respons yang lebih tinggi terhadap variasi data aktual, namun dengan peningkatan *noise*. Gambar 7 model yang dioptimasi dengan *Bayesian Optimization* menunjukkan prediksi yang lebih halus dan stabil, namun mengorbankan akurasi dalam mendeteksi lonjakan ekstrem, menandakan adanya kompromi antara kompleksitas model dan generalisasi.



Gambar 8. Plot Training Val dan Val Loss Pelatihan Model LSTM



Gambar 9. Plot Training Val dan Val Loss Pelatihan Model LSTM Bayesian Optimization

Berdasarkan Gambar 8 dan Gambar 9 grafik training loss dan validation loss untuk kedua pendekatan LSTM, terlihat bahwa model hasil *trial and error* dan model yang dioptimasi menggunakan *Bayesian Optimization* menunjukkan pola konvergensi yang serupa hingga 120 *epoch*. Kedua model mengalami penurunan *loss* yang cukup cepat pada awal pelatihan dan stabil pada *epoch* selanjutnya tanpa indikasi *overfitting* yang signifikan. Meskipun pendekatan *Bayesian Optimization* secara teoritis menawarkan pencarian hiperparameter yang lebih sistematis dan efisien, hasil empiris menunjukkan bahwa performa keduanya relatif setara dalam hal kestabilan dan akurasi pelatihan pada kasus prediksi curah hujan di ketiga lokasi.

Tabel 4. Perbandingan Matriks Evaluasi

	LSTM		LSTM-BO	
	RMSE	MAE	RMSE	MAE
Cilacap	160,51	126,77	148,87	115,14
Tegal	89,00	65,73	86,19	58,84
Semarang	104,05	76,32	100,21	70,60

Berdasarkan tabel evaluasi, model LSTM-*Bayesian Optimization* (LSTM-BO) menunjukkan performa yang lebih baik dibandingkan model LSTM biasa di semua lokasi. Penurunan nilai RMSE dan MAE terjadi

secara konsisten, misalnya di Cilacap RMSE turun dari 160,51 menjadi 148,87 dan MAE dari 126,77 menjadi 115,14, menunjukkan prediksi LSTM-BO lebih dekat ke nilai aktual. Hal serupa juga terlihat di Tegal dan Semarang, di mana perbaikan akurasi tercermin dari berkurangnya selisih rata-rata dan kesalahan kuadrat antara hasil

4. Kesimpulan

Pendekatan *Bayesian Optimization* menghasilkan model yang lebih halus dan adaptif terhadap fluktuasi musiman, sementara pendekatan *trial and error* memberikan respons yang lebih tinggi terhadap variasi data aktual namun cenderung lebih banyak *noise*. Analisis *loss* menunjukkan bahwa kedua pendekatan memiliki kestabilan pelatihan yang serupa tanpa indikasi *overfitting* hingga 120 *epoch*. Pemilihan konfigurasi input layer melalui pendekatan *Bayesian Optimization* terbukti memberikan hasil prediksi yang lebih baik dibandingkan pendekatan *trial and error*, ditunjukkan oleh penurunan nilai RMSE dan MAE di seluruh lokasi. Di Cilacap, RMSE turun dari 160,51 menjadi 148,87 dan MAE dari 126,77 menjadi 115,14; di Tegal, RMSE dari 89,00 menjadi 86,19 dan MAE dari 65,73 menjadi 58,84; sedangkan di Semarang, RMSE dari 104,05 menjadi 100,21 dan MAE dari 76,32 menjadi 70,60. Peningkatan akurasi memang tidak terlalu signifikan, tetapi optimasi *input layer* membantu model LSTM lebih stabil dan konsisten dalam mempelajari pola musiman pada data curah hujan bulanan berbasis data iklim. Pendekatan *Bayesian Optimization* menunjukkan efektivitas lebih baik dalam mengatur keseimbangan antara akurasi dan generalisasi model. Penelitian lanjutan disarankan mengeksplorasi kombinasi enam variabel prediktor yang telah digunakan serta menambahkan variabel lain seperti tekanan udara, kelembaban, dan parameter oseanografi guna memperkaya konteks dinamika curah hujan dan meningkatkan ketepatan prediksi.

Daftar Rujukan

- [1] Zhang, G.P., 2003. Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing*, 50, pp.159–175.
- [2] Liyew, C.M. and Melese, H.A., 2021. Machine learning techniques to predict daily rainfall amount. *Journal of Big Data*, 8, pp.1–11.
- [3] Buslim, N. and Iswara, R.P., 2019. Pengembangan algoritma unsupervised learning technique pada big data analysis di media sosial sebagai media promosi online bagi masyarakat. *Jurnal Teknik Informatika*, 12(1), pp.79–96.
- [4] Lin, Y., Yan, Y., Xu, J., Liao, Y. and Ma, F., 2021. Forecasting stock index price using the CEEMDAN-LSTM model. *North American Journal of Economics and Finance*, 57, pp.1–14.
- [5] Makarova, A., Biedenkapp, G., Lindauer, M. and Hutter, F., 2021. Overfitting in Bayesian optimization: an empirical study and early-stopping solution. *NeurIPS Workshop on Neural Architecture Search*.
- [6] Ulum, D.S.N. and Girsang, A.S., 2022. Hyperparameter optimization of long short term memory using symbiotic organism search for stock prediction. *International Journal of Innovative Research and Scientific Studies*, 5(2), pp.121–133.

- [7] Snoek, J., Larochelle, H. and Adams, R.P., 2012. Practical Bayesian optimization of machine learning algorithms. *Advances in Neural Information Processing Systems*, 25.
- [8] Cabrera, D., Jiménez, F., Vélez, J.F. and Taborda, H., 2023. Rainfall forecasting using a Bayesian framework and long short-term memory multi-model estimation. *Earth Science Informatics*, 16(2), pp.1373–1388.
- [9] Lathika, P. and Sheeba, D.S., 2023. A novel model for rainfall prediction using hybrid stochastic-based Bayesian optimization algorithm. *Environmental Science and Pollution Research*, 30, pp.92555–92567.
- [10] Ardhitama, A. and Sholihah, R., 2013. Model simulasi prakiraan CH bulanan pada wilayah Riau dengan menggunakan input data SOI, SST, NINO 3.4, dan IOD. *Jurnal Sains & Teknologi Modifikasi Cuaca*, 14(2), pp.95–104.
- [11] Tjasyono, B.H.K., 2004. *Klimatologi*. Bandung: Institut Teknologi Bandung.
- [12] NOAA Climate.gov, 2025. Meet ENSO's neighbor, the Indian Ocean Dipole. [Online] Tersedia di: <https://www.climate.gov/> [Accessed 1 May 2025].
- [13] Badan Meteorologi, Klimatologi, dan Geofisika (BMKG), 2025. Data Online BMKG. [Online] Tersedia di: <https://dataonline.bmkg.go.id/home> [Accessed 1 May 2025].
- [14] Bureau of Meteorology, 2025. Southern Oscillation Index (SOI). [Online] Tersedia di: <https://www.bom.gov.au/climate/enso/soi/> [Accessed 1 May 2025].
- [15] NOAA Climate Prediction Center, 2025. ENSO: Cold & warm episodes by season. [Online] Tersedia di: https://origin.cpc.ncep.noaa.gov/products/analysis_monitoring/ensostuff/ONI_v5.php [Accessed 1 May 2025].
- [16] NOAA Physical Sciences Laboratory, 2025. NCEP/NCAR reanalysis surface data. [Online] Tersedia di: <https://downloads.psl.noaa.gov/Datasets/ncep.reanalysis/Monthlies/surface/> [Accessed 1 May 2025].
- [17] NOAA Climate.gov, 2025. Meet ENSO's neighbor, the Indian Ocean Dipole. [Online] Tersedia di: <https://www.climate.gov/> [Accessed 1 May 2025].
- [18] NOAA Physical Sciences Laboratory, 2025. COBE2 sea surface temperature dataset. [Online] Tersedia di: <https://psl.noaa.gov/data/gridded/data.cobe2.html> [Accessed 1 May 2025].
- [19] Khodarahmi, M. and Maihami, V., 2023. A review on Kalman filter models. *Archives of Computational Methods in Engineering*, 30, pp.727–747.
- [20] Mega, T., Ushio, T., Takahiro, M., Kubota, T., Kachi, M. and Oki, R., 2019. Gauge-adjusted global satellite mapping of precipitation. *IEEE Transactions on Geoscience and Remote Sensing*, 57(4), pp.1928–1935.
- [21] Hyndman, R.J. and Athanasopoulos, G., 2018. *Forecasting: Principles and practice*. 2nd ed. Melbourne: OTexts.