Hybrid Neural Network–ARIMA for Time-Series Bias Correction of GFS Wind Speed Data to Support Renewable Energy Assessment in Java, Indonesia

Silvy R.Fithri[1,2], *Akhmad* Faqih[1*], *Agus* Nurrohim[2]

[1]Department of Geophysics and Meteorology, IPB University, Bogor, Indonesia
[2]Research Center for Conversion and Conservation of Energy – National Research and Innovation Agency Republic of Indonesia (BRIN) – South Tangerang, Indonesia

Abstract. Bias correction of Global Forecast System (GFS) wind speed data is essential for accurate wind resource assessment in Indonesia, particularly in regions where observations from Automatic Weather Stations (AWS) are sparse and wind variability is high. This study develops a Model Output Statistics (MOS)-based post-processing framework that combines nonlinear deep learning models, including a Convolutional Neural Network–Long Short-Term Memory (CNN–LSTM) model, linear statistical models such as Autoregressive Integrated Moving Average (ARIMA) and Seasonal Autoregressive Integrated Moving Average with eXogenous variables (SARIMAX), and hybrid configurations to correct time-series bias in GFS wind speed at five locations in Java. One year of GFS hindcast data and AWS observations was used to train and validate six predictive model schemes under single- and multi-predictor settings using an expanding-window cross-validation strategy. Model performance was evaluated using multiple error metrics and a composite index to identify the best-performing configuration at each site. The results show consistent improvements relative to the GFS baseline, with performance differences associated with local wind variability and the interaction between linear and nonlinear components in the time series. Overall, the proposed framework provides a robust and adaptable approach for improving GFS-based wind information, with practical relevance for wind energy assessment and operational forecasting in Indonesia.

1 Introduction

Indonesia's inland wind energy potential is estimated at 60.65 GW and is mainly concentrated in coastal regions such as South Java, South Sulawesi, Maluku, and East Nusa Tenggara, with average wind speeds of 6–8 m/s and power potential of 400–500 W/m [1]. [*]Despite this potential, large-scale utilization remains limited due to generally low wind

[*]Corresponding author : akhmadfa@apps.ipb.ac.id

regimes, complex terrain, and sparse surface observations. As a result, the freely available Global Forecast System (GFS), which offers broad spatial and temporal coverage, is widely used. However, its coarse horizontal resolution of approximately 27.8 km introduces systematic bias, making Model Output Statistics (MOS)–based post-processing necessary to align GFS outputs with local Automatic Weather Station (AWS) observations for wind energy assessment. Accurate wind forecasts are therefore critical for wind power plant operation, as they directly affect planning efficiency and cost savings in renewable energy systems.

Recent studies show that post-processing bias correction improves GFS wind forecasts. Deep learning models, including Long Short-Term Memory (LSTM), Convolutional LSTM (ConvLSTM), and convolutional neural networks (CNN), effectively reduce GFS biases over complex terrain and oceanic regions [2], [3]. In parallel, classical statistical approaches within the MOS framework, such as systematic bias correction (SBC) and MOS/MOS2, remain effective in coastal areas, including the Pearl River Estuary [4]. Together, these approaches demonstrate the complementary strengths of deep learning and statistical post-processing for enhancing GFS-based wind forecasts.

Previous studies highlight a clear research gap in post-processing bias correction for global numerical wind speed products, particularly in data-sparse regions such as Indonesia where in situ observations are limited. This study addresses this gap by adopting a MOS–based post-processing framework to quantify and correct bias between GFS outputs and AWS observations. Owing to the complex, intermittent, and non-stationary nature of wind speed, single-model approaches are often insufficient [5]. While nonlinear models such as Convolutional Neural Network–Long Short-Term Memory (CNN–LSTM) effectively capture nonlinear structures, they may leave residual linear dependencies that can be further modeled using Autoregressive Integrated Moving Average (ARIMA) or Seasonal Autoregressive Integrated Moving Average with eXogenous variables (SARIMAX) techniques [6]. These considerations motivate the combined CNN-LSTM-ARIMA/SARIMAX design as the main framework for GFS–AWS bias correction in this work :

1. Develop and evaluate six (6) sets of prediction models for bias correction, including a single deep learning (CNN-LSTM)-based model, linear statistical models (residual ARIMA, ARIMA on the y variable, and SARIMAX), and several hybrid model variants that combine CNN-LSTM with these linear models. The best-performing model for each location will be presented in graphs and tables;
2. Analyze the effect of using a single predictor (GFS wind speed) compared to multiple predictors (wind speed, humidity, and temperature from the GFS), and evaluate the effectiveness of the hybrid model compared to a single model in correcting GFS wind speed bias at several locations on the island of Java.

2 Materials and Methods

2.1 Research Data and Locations

This data-driven numerical study uses one year of Global Forecast System (GFS) hindcast data from November 2024 to November 2025, including 10 m wind speed, relative humidity, and air temperature at a 6-hour temporal resolution. AWS observations from the same period were originally recorded at an hourly interval in Western Indonesian Time (WIB, UTC+7), converted to Coordinated Universal Time (UTC), and subsequently sampled at 6-hour intervals to match the GFS data. Data were obtained from five stations in Java: Sukabumi (−7.16, 106.51), Pekalongan (−7.18, 109.70), Kediri (−7.89, 112.09), Banyuwangi (−8.27,

114.18), and Subang (−6.28, 107.87). These stations were selected based on data availability, coastal–inland representativeness, and proximity to the nearest GFS grid points. GFS outputs were used as predictors and AWS observations as targets under two model configurations: a single-predictor setup using wind speed only, and a multi-predictor setup incorporating wind speed, relative humidity, and air temperature. All modeling experiments were conducted in Python using Google Colab. The study locations are shown in Fig. 1, overlaid on the Global Wind Atlas 100 m wind speed map, to classify each site into low-, medium-, or high-wind-speed zones.

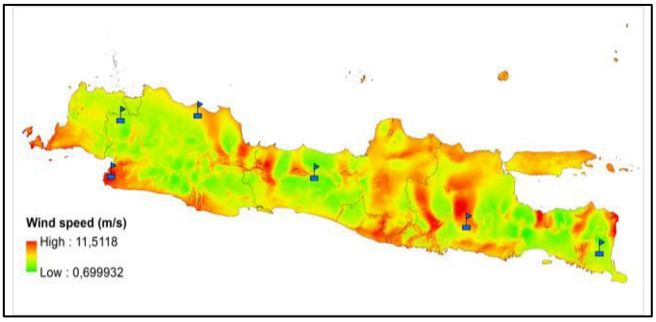

Fig. 1 Distribution of Research Locations Overlaid with the Global Wind Atlas Wind Speed Map at an Altitude of 100 m.

2.2 Preprocessing

The data were uniformly preprocessed through GFS–AWS temporal resolution adjustment, MinMaxScaler normalization applied only to the training set, and sliding-window formation for input–target pairs. The data were then chronologically and time-ordered split into training, validation, and test subsets (70%–15%–15%) without shuffling or mixing**,** ensuring consistency across all locations and preventing information leakage. All preprocessing steps were applied identically to all models and predictor configurations.

2.3 Model Training, Cross-Validation, and Hyperparameter Tuning

All models were trained using expanding-window cross-validation to preserve temporal dependence and prevent data leakage [7]. CNN–LSTM models used sliding-window inputs with a fixed length of $T = 16$ lags, using either a single predictor (GFS wind speed) or a multi-predictor set (GFS wind speed, temperature, and relative humidity). All inputs were normalized using Min–Max scaling fitted on the training subset only. Hyperparameters were optimized via Random Search (40 trials per fold) with learning rate (1×10^{-3}, 5×10^{-4}), batch size (16, 32), drop out (0.10, 0,20) and LSTM unit (64, 96) for first layer and unit (32, 48) for second layer). Training used the Adam optimizer with MSE loss for up to 300 epochs, Early Stopping (patience = 10), and ReduceLROnPlateau (factor = 0.5, patience = 8–10, minimum learning rate = 1×10−5; shuffling was disabled. A fixed Conv1D–LSTM architecture was adopted—Conv1D (64, kernel size 3, causal padding, ReLU) → Batch Normalization → MaxPooling1D → Dropout → two LSTM layers → Dropout → Dense (64, ReLU) → Dense (1)—to ensure performance differences arise from post-processing or hybrid strategies rather than architectural variations [8]. The validation-tail refers to a subset of validation data used solely for fusion-weight estimation.

2.4 Predictive Models for Time-Series Bias Correction

Building on this framework, six models were defined by combining a deep learning model (CNN-LSTM) with linear statistical models (residual ARIMA, direct ARIMA on y, and SARIMAX), arranged as three single-predictor and three multi-predictor variants. All follow the same pipeline (preprocessing → expanding-window cross-validation → hyperparameter tuning); the main differences lie in the choice of linear component and the CNN-LSTM, linear integration scheme within the hybrid architecture. The linear parts and their roles are highlighted with red dashed boxes in Figure 2.

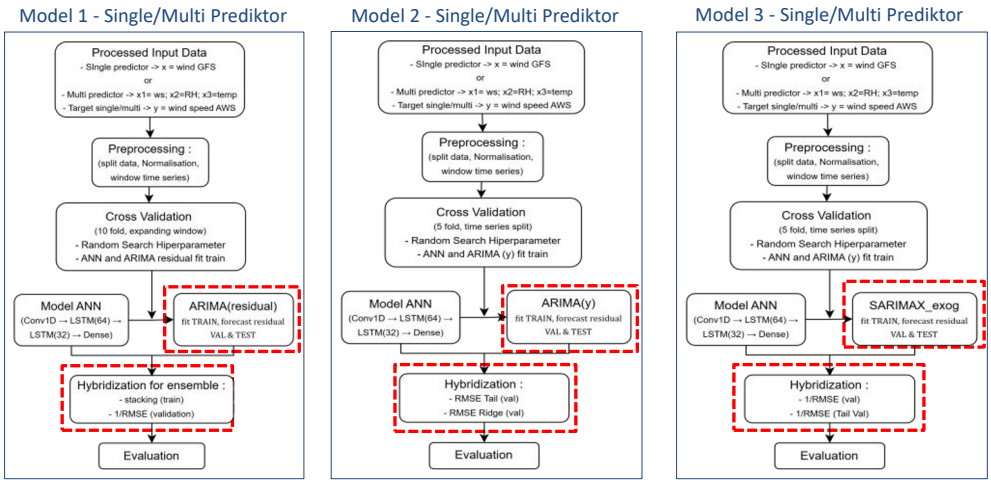

Fig. 2 Flowchart of the methodology of Model 1-Single/Multi Predictor (CNN-LSTM–ARIMA Residual), Model 2-Single/Multi Predictor (CNN-LSTM–ARIMA Direct), and Model 3-Single/Multi Predictor (CNN-LSTM–SARIMAX Exog).

2.5 Statistical Component (ARIMA/SARIMAX)

ARIMA models are fitted using the training segment only, either to the CNN–LSTM residual series (residual hybrid) or directly to the AWS wind speed series (direct hybrid). The (p,d,q) order is selected from a predefined candidate set and finalized based on validation performance under the expanding-window scheme, and the final orders are reported for each site/model. Stationarity and invertibility constraints are relaxed (enforce_stationarity=False, enforce_invertibility=False) to improve numerical stability, and forecasts are generated forward for validation and test periods without refitting on future data. For SARIMAX, the model is trained on the training segment with exogenous predictors (single: GFS wind speed; multi: wind speed, temperature, relative humidity) and forecast forward using the same leakage-safe protocol.

All preprocessing parameters (Min–Max scaling) are fitted using the TRAIN subset only and then applied unchanged to VALIDATION and TEST. In the residual-hybrid model, the ARIMA component is trained only on residuals from TRAIN, where residuals are computed using CNN–LSTM predictions generated within the training segment. VALIDATION data (including the validation-tail window defined by VAL_TAIL_FRAC) are used only to estimate fusion weights and select the final configuration, and are not used to fit ARIMA/SARIMAX parameters. The TEST subset is used exclusively for the final out-of-sample evaluation.

2.6 Hybrid Integration Scheme

Let y_t denote the observed AWS wind speed and $\hat{y}_t^{(N)}$ and $\hat{y}_t^{(L)}$ denote the nonlinear (CNN–LSTM) and linear (ARIMA/SARIMAX) predictions at time t, respectively. All base models are fitted on the training segment only, and forecasts are generated forward for validation and test without refitting on future data.
(i) Inverse-error (1/RMSE) fusion.
Fusion weights are computed on the validation segment. Define the validation index set V, and its "validation-tail" subset V_{tail} as the last α fraction of validation samples, where α = VAL_TAIL_FRAC (implementation uses the last portion of validation only for estimating weights).
Compute

$$RMSE_N = \sqrt{\frac{1}{|V_{tail}|}} \sum_{t\in Vail}\left(y_t - \hat{y}_t^{(N)}\right)^2, RMSE_L = \sqrt{\frac{1}{|V_{tail}|}} \sum_{t\in Vail}\left(y_t - \hat{y}_t^{(L)}\right)^2 \quad (1)$$

And set

$$w_N = \frac{1/RMSE_N}{(1/RMSE_N)+ (1/RMSE_L)}, \; w_L = 1 - w_N \quad (2)$$

The fused prediction is

$$\hat{y}_{it}^{(hyb)} = w_N \hat{y}_t^{(N)} + w_L \hat{y}_t^{(L)} \quad (3)$$

(ii) Ridge fusion (stacking with regularization).
Ridge coefficients are also on V_tail using predictions as regressors :

$$w^* = \arg min_{w\geq 0} \; \|y - Xw\|_2^2 + \lambda\|w\|_2^2 \quad (4)$$

Where $X = \left[\hat{y}^{(N)}\, \hat{y}^{(L)}\right]$ and y are taken from the validation-tail window. The resulting weights are normalized to sum to one before applying them to produce $\hat{y}_{it}^{(hyb)}$.
(iii) Auto-select fusion rule (validation RMSE)
In this scheme, the model automatically selects the fusion rule (inverse-RMSE vs ridge regression) based on the RMSE computed over the full validation window V. The fusion rule that yields the lowest RMSE on the validation data is chosen. Once the best fusion rule is selected, the corresponding fusion weights are applied to generate the final hybrid prediction for the test set.

Model 1 (CNN-LSTM–ARIMA Residual–Hybrid) – Single/Multi Predictor

This model uses a CNN–LSTM for nonlinear prediction, followed by an ARIMA model trained on the CNN–LSTM residuals to capture remaining linear autocorrelation, following a residual-based hybrid forecasting framework [9]. The CNN–LSTM prediction and the ARIMA residual correction are combined using validation-based inverse-error weighting (1/RMSE), ensuring that each component contributes according to its relative predictive skill.

Model 2 (CNN–LSTM-ARIMA(y) Direct–Hybrid) – Single/Multi Predictor

In this configuration, the CNN–LSTM and ARIMA are trained in parallel on the training segment, with ARIMA applied directly to the AWS wind speed series. The CNN–LSTM captures nonlinear temporal dependencies, while ARIMA models linear and autoregressive structure. The two outputs are fused using Ridge regression, with coefficients estimated on the validation segment to reduce overfitting and ensure stable model integration [10].

Model 3 (CNN-LSTM–SARIMAX Exogenous–Hybrid) – Single/Multi Predictor

This model integrates CNN–LSTM and SARIMAX to exploit complementary nonlinear and linear time-series characteristics [11]. The CNN–LSTM models nonlinear relationships between predictors and AWS wind speed, while SARIMAX is trained on the training segment only with exogenous predictors (single: GFS wind speed; multi: wind speed, temperature, and relative humidity). The final prediction is obtained using validation-based inverse RMSE weighting, ensuring leakage-safe hybrid forecasting.

2.7 Performance Metrics

The performance of the six model configurations (single and hybrid) in correcting GFS wind speed time-series bias is evaluated using six metrics: Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), symmetric MAPE (sMAPE), bias, and Pearson correlation. Let y_t denote the observed AWS wind speed and $\hat{y}_t$ the bias-corrected estimate at time step t, for $t = 1, \ldots, n$. Because percentage-based errors can become unstable when observed wind speeds approach zero, MAPE is interpreted with caution and reported as a complementary indicator only. Model comparison therefore primarily emphasizes RMSE, MAE, and sMAPE, which are more robust under low-wind conditions. For numerical stability, a small constant $\varepsilon = 10^{-6}$ is included in the sMAPE denominator to avoid division by zero. Correlation is computed using the Pearson correlation coefficient. To identify the best-performing model at each site, the normalized metric values are aggregated using the Simple Additive Weighting (SAW) method to form a composite performance index. The evaluation metrics are defined as follows.

$$\text{RMSE} = \sqrt{\sum_{t=1}^{n} \frac{(\hat{y}_t - y_t)^2}{n}} \quad (5)$$

$$\text{MAE} = \frac{\sum_{t=1}^{n} |\hat{y}_t - y_t|}{n} \quad (6)$$

$$\text{MAPE} = \frac{1}{n} \sum_{t=1}^{n} \left| \frac{\hat{y}_t - y_t}{y_t} \right| \quad (7)$$

$$\text{sMAPE} = \frac{1}{n} \sum_{t=1}^{n} \frac{|\hat{y}_t - y_t|}{(|\hat{y}_t| + |y_t| + \varepsilon)/2}, \varepsilon = 10^{-6} \quad (8)$$

$$\text{Bias} = \frac{1}{n} \sum_{t=1}^{n} (\hat{y}_t - y_t) \quad (9)$$

$$\text{Correlation} = \frac{\sum_{t=1}^{n} (\hat{y}_t - \bar{\hat{y}})(y_t - \bar{y})}{\sqrt{\sum_{t=1}^{n} (\hat{y}_t - \bar{\hat{y}})^2 \sum_{i=1}^{n} (y_t - \bar{y})^2}} \quad (10)$$

3 Results

The following results show the best performance of the six prediction model schemes (single and hybrid) at each study site. The best model performance is presented in graphs before and after time-series bias correction, as well as in a performance metrics table 1.

3.1 Sukabumi

Figure 3 (left) shows that the raw GFS (x) wind speed is smoother and tends to underestimate AWS (y), especially during higher-wind episodes where AWS exhibits stronger variability and sharper peaks. This mismatch is reflected by the large baseline errors (RMSE = 2.67; MAE = 2.25) and near-zero correlation (Corr = −0.01) in Table 1.

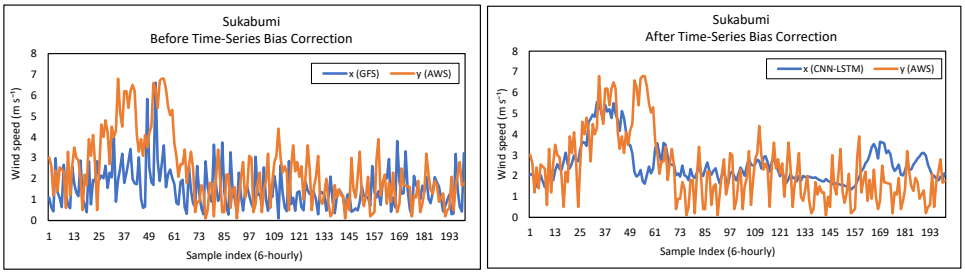

Fig. 3 Wind speed time series at Sukabumi before time-series bias correction (left) and after time-series bias correction using Model 2 Single–CNN–LSTM (right). The x-axis shows sample index (6-hourly)

After time-series bias correction (Figure 3, right), the corrected series follows AWS more closely in both magnitude and temporal pattern, with errors reduced across models (RMSE = 1.54–1.84; MAE = 1.30–1.52) and correlation improving to 0.21–0.33 (Table 1). Based on the composite index, the best model at Sukabumi is Model 2 Single–CNN–LSTM (score = 4.653), which provides balanced improvements in RMSE, MAE, sMAPE, bias, and correlation, although some extreme peaks remain partially underestimated; the best models for all sites are summarized in Table 1.

3.2 Pekalongan

Figure 4 shows the wind speed time series in Pekalongan before and after time-series bias correction. Before correction, the GFS wind speed exhibits smoother and lower-amplitude fluctuations than AWS, failing to capture several sharp peaks present in the observed data. This mismatch indicates partial underestimation of high-wind events and is reflected in the baseline performance, with RMSE of 0.96 and a moderate correlation of 0.24 (Table 1).

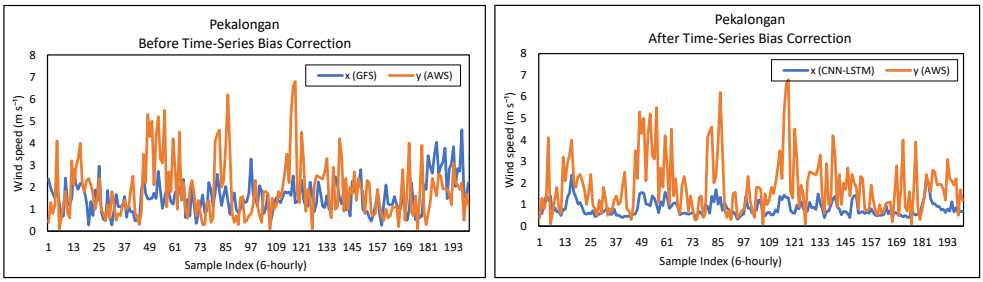

Fig. 4. Wind speed in Pekalongan before time-series bias correction using GFS (left) and after correction using Model 2 Single–CNN–LSTM (right).

After time-series bias correction, the CNN–LSTM output follows the AWS temporal pattern more closely, with improved alignment in both magnitude and timing. Among all evaluated models, Model 2 Single–CNN–LSTM achieves the highest composite score (6.123), driven by the lowest RMSE (0.70), reduced MAE (0.48), and the largest increase in correlation (0.41) at this site. Although some extreme peaks remain underestimated, the corrected series represents a substantial improvement over the baseline. A summary of the best-performing models across all locations is provided in Table 1..

3.3 Kesdiri

Figure 5 shows the wind speed time series in Kediri before and after time-series bias correction. Before correction, the GFS wind speed exhibits much larger variability than AWS and generally overestimates the observed values, particularly during high-wind episodes. Although the temporal pattern of GFS broadly follows AWS, the magnitude mismatch is substantial, resulting in large baseline errors (RMSE = 3.65; MAE = 3.25) and a pronounced positive bias (3.20), as summarized in Table 1.

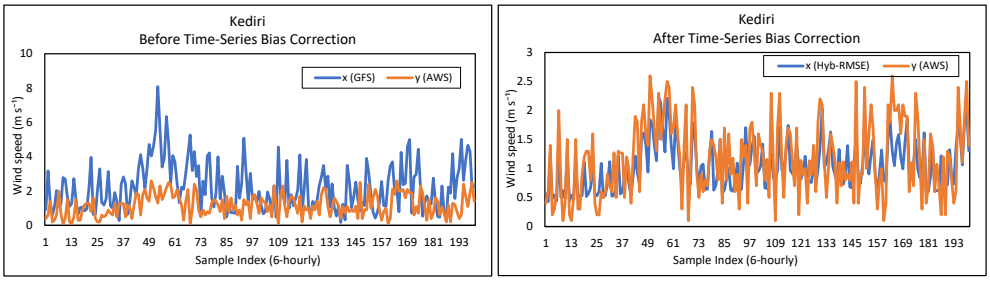

Fig 5. Wind Speed in Kediri Before Time-Series Bias Correction Using GFS (left) and After Correction Using Model 1 Multi-Hybrid-RMSE (right).

After time-series bias correction, the corrected series aligns closely with AWS in both amplitude and temporal evolution, with a strong reduction in overestimation. Error metrics decrease sharply across all corrected models (RMSE = 0.52–0.60; MAE = 0.40–0.48), bias approaches zero, and correlation increases markedly to approximately 0.70–0.77. Based on the composite performance index, Model 1 Multi–Hybrid-RMSE achieves the best overall performance at Kediri (score = 6.028), reflecting its balanced improvements across RMSE, MAE, bias, and correlation. Other hybrid configurations yield comparable gains but rank slightly lower due to less consistent performance across all metrics. The best-performing models for all locations are summarized in Table 1.

3.4 Banyuwangi

Figure 6. shows the wind speed time series in Banyuwangi before and after time-series bias correction. Before correction, the GFS wind speed exhibits much stronger and more erratic fluctuations than AWS, with frequent high-amplitude spikes that are not present in the observed series. This behavior leads to systematic overestimation and poor agreement in magnitude, despite broadly similar temporal patterns, as reflected by the baseline performance with RMSE of 1.48, MAE of 1.05, and a negative correlation (−0.26) in Table 1.

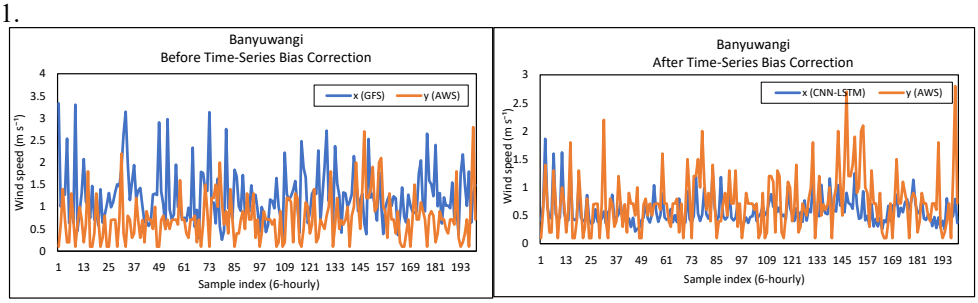

Fig 6. Wind Speed in Banyuwangi Before Time-Series Bias Correction Using GFS (left) and After Correction Using Model 2 Single-CNN-LSTM (right).

After time-series bias correction, the corrected series aligns substantially better with AWS, showing reduced variability and improved tracking of observed fluctuations. Error metrics decrease sharply across all corrected models (RMSE = 0.36–0.40; MAE = 0.28–0.32), bias is reduced to values close to zero, and correlation increases markedly to approximately 0.73–0.78 (Table 1). Based on the composite performance index, the best-performing model at this site is Model 2 Single–CNN–LSTM (score = 6.207), closely followed by Model 1 Single–Hybrid Stack (score = 6.206). These models provide the most balanced improvements across error reduction, bias correction, and temporal correlation, while other configurations show Hybrid modeling approaches combining statistical and machine learning techniques have shown improved robustness under complex wind regimesslightly less consistent performance across all metrics. The best models for all locations are summarized in Table 1.

3.5 Subang

Figure 7. shows the wind speed time series in Subang before and after time-series bias correction. Before correction, the GFS wind speed exhibits much larger variability than AWS and systematically overestimates the observed values, particularly during high-wind periods. Although the general temporal patterns are similar, the magnitude mismatch is substantial, resulting in large baseline errors (RMSE = 3.27; MAE = 2.72) and a pronounced positive bias (2.35), as summarized in Table 1.

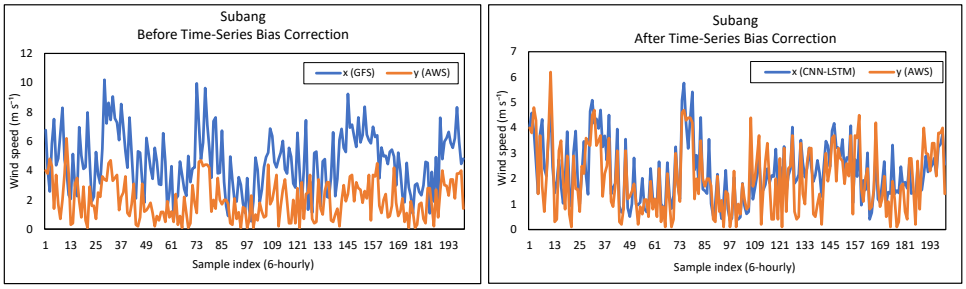

Fig 7. Wind speed in Subang Before Time-Series Bias Correction Using GFS (left) and After Correction Using Model 3 Multi-CNN-LSTM (right).

After time-series bias correction, the corrected series follows AWS more closely in both amplitude and temporal evolution, with a marked reduction in overestimation. Across the corrected models, RMSE decreases to 0.68–0.76, MAE to 0.53–0.61, bias approaches zero, and correlation increases substantially to approximately 0.80–0.84 (Table 1). Based on the composite performance index, the best-performing models at this site are Model 2 Single–CNN–LSTM (score = 6.032) and Model 3 Multi–CNN–LSTM (score = 6.044), both of which achieve high correlation and low bias while maintaining consistently low error metrics. Other configurations show less balanced performance across metrics, resulting in lower overall scores. The best models for all locations are summarized in Table 1

Table 1 summarizes the performance of the GFS baseline model and the best bias correction models developed from six (6) prediction model schemes (single and hybrid) at five (5) study sites. Overall, the developed models substantially reduced error values relative to the GFS baseline across locations, as reflected by lower RMSE/MAE/sMAPE and higher correlation. These results confirm the effectiveness of the proposed modeling approach and form the basis for further analysis in the discussion chapter.

Table 1. Model Comparison Metrics Across All Sites

Location	Model (M1/M2/M3) Single or Multi Predictor	RMSE	MAE	sMAPE	Bias	Corr	Composite Score
Sukabumi	GFS (baseline)	2.67	2.25	0.98	-2.11	-0.01	1.376
	M1 Single-Hyb-Stck	1.82	1.49	0.55	-1.09	0.28	4.441
	M2 Single-CNN-LSTM	**1.84**	**1.52**	**0.53**	**-1.19**	**0.31**	**4.653**
	M3 Singl-Hyb-1/RMSE	1.54	1.30	0.44	-0.62	0.30	4.516
	M1 Multi-Hyb-Stck	1.65	1.39	0.49	-0.70	0.21	3.909
	M2 Multi-Hyb-RMSE	1.66	1.38	0.49	-0.90	0.33	4.651
	M3 Multi-Hyb-1/RMSE	1.71	1.41	0.50	-0.92	0.24	4.257
Pekalongan	GFS (baseline)	0.96	0.78	0.84	0.47	0.24	0.587
	M1 Single-CNN-LSTM	0.74	0.49	0.66	-0.06	0.29	5.656
	M2 Single-CNN-LSTM	**0.70**	**0.48**	**0.65**	**-0.08**	**0.41**	**6.123**
	M3 Singl-Hyb-1/RMSE	0.71	0.48	0.66	-0.05	0.40	6.073
	M1 Multi-CNN-LSTM	0.79	0.56	0.71	0.11	0.21	4.386
	M2 Multi-Hyb-RMSE	0.74	0.48	0.66	-0.07	0.28	5.616
	M3 Multi-Hyb-1/RMSE	0.73	0.47	0.65	-0.10	0.32	5.806
Kediri	GFS (baseline)	3.65	3.25	1.02	3.20	0.39	0.624
	M1 Single-Hyb-Stack	0.54	0.43	0.33	0.12	0.77	5.955
	M2 Single-CNN-LSTM	0.58	0.47	0.36	-0.05	0.71	5.822
	M3 Single-CNN-LSTM	0.60	0.48	0.37	-0.13	0.70	5.760
	M1 Multi-Hyb-RMSE	**0.52**	**0.40**	**0.32**	**-0.01**	**0.77**	**6.028**
	M2 Multi-Hyb-Ridge	0.57	0.45	0.35	-0.04	0.73	5.858
	M3 Multi-Hyb-1/RMSE	0.55	0.45	0.35	0.09	0.77	5.886
Banyuwangi	GFS (baseline)	1.48	1.05	0.94	0.63	-0.26	0.065
	M1 Single-Hyb-Stack	0.36	0.28	0.52	0.00	0.77	6.206
	M2 Single-CNN-LSTM	**0.36**	**0.28**	**0.53**	**-0.09**	**0.78**	**6.207**
	M3 Single-CNN-LSTM	0.39	0.29	0.56	-0.08	0.74	6.045
	M1 Multi-Hyb-RMSE	0.38	0.30	0.53	0.013	0.73	6.084
	M2 Multi-Hyb-Ridge	0.38	0.29	0.54	-0.02	0.75	6.110
	M3 Multi-Hyb-1/RMSE	0.40	0.32	0.57	0.03	0.77	5.905
Subang	GFS (baseline)	3.27	2.72	1.06	2.35	-0.21	0.056
	M1-Single-Hyb-RMSE	0.71	0.56	0.47	0.02	0.82	6.005
	M2 Single-CNN-LSTM	0.68	0.54	0.48	-0.01	0.84	6.032
	M3 Singl-Hyb-1/RMSE	0.76	0.61	0.51	0.02	0.80	5.869
	M1 Multi-CNN-LSTM	0.98	0.73	0.55	0.28	0.68	5.344
	M2 Multi-Hyb-RMSE	0.83	0.68	0.55	0.07	0.77	5.644
	M3 Multi-CNN-LSTM	**0.677**	**0.53**	**0.47**	**0.03**	**0.83**	**6.044**

The "Best Model" for each site is determined based on the composite performance index, which is calculated from the weighted averages of RMSE, MAE, sMAPE, MAPE, bias and correlation values. The model with the highest composite score is selected as the best model for each location.

4 Discussions

Empirical findings indicate that variations in bias correction performance at each location are closely related to the characteristics of AWS variability and the proportion of linear-nonlinear signals in the time series. These factors appear to contribute to determining whether a single model (CNN-LSTM) or a hybrid model (CNN-LSTM-ARIMA/SARIMAX) provides more effective results at each location.

In Sukabumi and Pekalongan, the AWS exhibits very high fluctuations with sporadic extreme peaks. Under these conditions, the single-predictor CNN-LSTM configuration performed best, while all the hybrid model variants did not yield consistent improvements over the single-predictor CNN–LSTM under these highly variable conditions. This indicates that the linear component that ARIMA can model is relatively weak compared to the nonlinear component. This pattern aligns with Porto et al., who

reported that infrequent and sharply fluctuating extreme events are difficult for machine learning models to accurately represent, resulting in high errors at extreme points [12]. The addition of exogenous predictors (GFS temperature and humidity) also did not improve accuracy, indicating that these two variables are weak or redundant relative to GFS wind speed, potentially adding noise and the risk of multicollinearity [13]. To clarify the observed performance differences between single and multi predictor configurations, the inter-predictor correlation structure was examined (table 2). The multicollinearity level summarizes the overall degree of inter-predictor dependence per site and is determined by the strongest absolute inter-predictor correlation.

Table 2. Summary of inter-predictor correlations and multicollinearity levels

Location	r(WS–RH)	r(WS–T)	r(RH–T)	Overall multicollinearity
Sukabumi	−0.56	0.55	−0.95	Very high
Pekalongan	−0.41	0.32	−0.81	High
Kediri	−0.36	0.11	−0.90	Very high
Banyuwangi	−0.75	0.69	−0.87	Very high
Subang	−0.28	0.34	−0.79	High

The r denotes the Pearson correlation coefficient; multicollinearity level is assessed based on the strongest absolute inter-predictor correlation per site. The consistently high inter-predictor correlations explain the limited advantage of multi-predictor models across sites.

In Kediri, AWS variability is more moderate and stable, making the GFS–AWS relationship easier to study. Under these conditions, hybrid models performed more effectively than single models, with multi-predictor configurations showing a slight, though not always significant, advantage, consistent with literature showing that combinations of linear and nonlinear components (e.g., ARIMA–LSTM) often provide higher accuracy than single models [14] and supporting the view that single-model approaches are adequate for many regions.

Similarly, the AWS in Banyuwangi also showed a relatively stable pattern, but the best models at this location came from single-predictor configurations, using either a single model (CNN-LSTM) or a residual-based hybrid (CNN-LSTM–ARIMA residuals), indicating that the primary signal for bias correction is already largely reflected in GFS wind speeds such that adding exogenous predictors does not provide substantial benefit, in line with Valsaraj [14], who showed that historical wind series alone can provide an adequate basis for machine learning modeling without requiring complex input structures.

Subang is the most stable location across all study sites. The low variability of the AWS results in nearly identical performance for single- and multi-predictor CNN-LSTM models. Under these conditions, adding exogenous predictors or integrating linear models does not provide substantial added value. This aligns with literature emphasizing the need to limit the number of correlated predictors in multivariate time series modeling to maintain model stability and generalizability [15].

Beyond statistical improvements, the bias-corrected wind speed series better supports wind resource assessment by reducing systematic over/underestimation of mean wind speed. Because wind power density scales with wind speed cubed, the observed reductions in RMSE/MAE help limit bias in energy-yield and capacity factor estimates. Improved correlation also enhances the representation of wind variability and the frequency of wind speeds exceeding turbine cut-in thresholds, supporting more reliable screening and operational planning.

Across locations, the single-predictor CNN-LSTM is the best model in four sites (Sukabumi, Pekalongan, Banyuwangi, and Subang), showing its ability to capture nonlinear wind speed patterns under low, moderate, and high variability, consistent with findings that even hybrid models still share similar limitations with single models at extreme peaks. In contrast, multi-predictor models are constrained by multicollinearity, which can weaken performance [15]. Another key result is the consistently strong residual CNN-LSTM–ARIMA model in Kediri and Banyuwangi, where residual ARIMA directly models the remaining linear structure of CNN-LSTM predictions, outperforming parallel ARIMA(y) and reflecting one of the most widely used strategies in hybrid forecasting.

This study uses ~1 year of data at 6-hour resolution, which may not represent interannual variability or short-term gust extremes. Therefore, the reported performance reflects the analyzed period and temporal scale; future work should evaluate longer multi-year records and higher-resolution observations

5 Conclusions

This study confirms that MOS-based wind speed bias correction can be improved through nonlinear, linear, or hybrid modeling, with effectiveness depending on local wind variability and the linear–nonlinear signal balance in the time series. The CNN-LSTM approach generally outperforms locations with high variability, while the hybrid model is more effective when the linear structure remains strong. These findings suggest that the selection of bias correction models must be tailored to local characteristics and CNN-LSTM not be generalized across regions. For further development, signal decomposition-based approaches such as EMD, VMD, or wavelet have the potential to improve modeling robustness, especially for data with extreme wind variability. Furthermore, bias correction needs to be continued at the distribution correction stage to ensure statistical consistency and improve the reliability of corrected wind data for wind energy applications and operational forecasting.

References

[1] D. G. Cendrawati, N. W. Hesty, B. Pranoto, A. Aminuddin, A. H. Kuncoro, and A. Fudholi, “Short-Term Wind Energy Resource Prediction Using Weather Research Forecasting Model for a Location in Indonesia,” *Int. J. Technol.*, vol. 14, p. **584**, (2023), doi: https://doi.org/10.14716/ijtech.v14i3.5803.

[2] V. Gomes, D. Carvalho, and S. Gouveia, “On the Correction of GFS Wind Speed Forecasts in Portugal Using LSTM Networks,” (2026), pp. **321–332**. doi: https://doi.org/10.1007/978-3-031-99568-2_26.

[3] C. Pang, T. Song, H. Sun, X. Li, and D. Xu, “A deep learning method for bias correction of wind field in the South China Sea,” *Front. Mar. Sci.*, vol. 11, (2025), doi: https://doi.org/10.3389/fmars.2024.1429057.

[4] X. Sun *et al.*, “Short-Term Wind Speed Forecasts over the Pearl River Estuary: Numerical Model Evaluation and Deterministic Post-Processing,” *J. Trop. Meteorol.*, vol. 30, no. 4, pp. **390–404**, Dec. (2024), doi: 10.3724/j.1006-8775.2024.035.

[5] S. R. Fithri, N. W. Hesty, R. P. Wijayanto, and B. Pranoto, “Enhancing wind energy prediction accuracy with a hybrid Weibull distribution and ANN model : a case study across ten locations in Java Island , Indonesia,” vol. 41, no. 1, pp. **180–190**, (2026), doi: 10.11591/ijeecs.v41.i1.pp180-190.

[6] M. K. Sallam Ma’aitah, J. B. Idoko, A. Alwhelat, K. Smart, and Z. Alwaeli,

"Evaluation of Hyperparameter Optimization Techniques in Deep Learning Considering Accuracy, Runtime, and Computational Efficiency Metrics," *J. Soft Comput. Data Min.*, vol. 6, (2025), doi: https://doi.org/10.30880/jscdm.2025.06.01.013.

[7] N. T. H. Thu, P. N. Van, N. V. N. Nam, P. H. Minh, and P. Q. Bao, "Forecasting Wind Speed Using A Hybrid Model Of Convolutional Neural Network And Long-Short Term Memory With Boruta Algorithm-Based Feature Selection," *J. Appl. Sci. Eng.*, vol. 26, pp. **1055–1062**, (2023), doi: https://doi.org/10.6180/jase.202308_26(8).0001.

[8] A. Neagoe, E.-I. Tică, L.-I. Vuță, O. Nedelcu, G.-E. Dumitran, and B. Popa, "Hybrid LSTM-ARIMA Model for Improving Multi-Step Inflow Forecasting in a Reservoir," *Water*, vol. 17, p. **3051**, (2025), doi: https://doi.org/10.3390/w17213051.

[9] G. Çınarer, "Hybrid Deep Learning and Stacking Ensemble Model for Time Series-Based Global Temperature Forecasting," *Electronics*, vol. 14, p. **3213, (**2025), doi: https://doi.org/10.3390/electronics14163213.

[10] I. A. Kachalla, C. Ghiaus, A. Ademuwagun, O. B. Odeyinde, and M. Baseer, "Data-driven hybrid SARIMAX-MLP framework for energy consumption prediction in residential micro-grid," *Results Eng.*, vol. 26, p. **105336**, (2025), doi: https://doi.org/10.1016/j.rineng.2025.105336.

[11] A. F. Gonzalez-Mora, E. Foulon, and A. N. Rousseau, "A climate-informed statistical framework to indirectly estimate trends in future seasonal high flows in snow-dominated watersheds using short-term climate variability indices," *J. Hydrol.*, vol. 664, p. **134441**, (2026), doi: https://doi.org/10.1016/j.jhydrol.2025.134441.

[12] D. Xu, Q. Zhang, Y. Ding, and D. Zhang, "Application of a hybrid ARIMA-LSTM model based on the SPEI for drought forecasting," *Environ. Sci. Pollut. Res.*, vol. 29, no. 3, pp. **4128–4144**, Jan. (2022), doi: https://doi.org/10.1007/s11356-021-15325-z.

[13] V. Bali, A. Kumar, and S. Gangwar, "A Novel Approach for Wind Speed Forecasting Using LSTM-ARIMA Deep Learning Models," *Int. J. Agric. Environ. Inf. Syst.*, vol. 11, no. 3, pp. **13–30**, Jul. (2020), doi: https://doi.org/10.4018/IJAEIS.2020070102.

[14] P. Valsaraj, D. Alex Thumba, and K. Satheesh Kumar, "Spatio-temporal independent applicability of one time trained machine learning wind forecast models: a promising case study from the wind energy perspective," *Int. J. Sustain. Energy*, vol. 41, pp. **1164–1182**, (2022), doi: https://doi.org/10.1080/14786451.2022.2032060.

[15] J. Kim, H. Kim, H. Kim, D. Lee, and S. Yoon, "A comprehensive survey of deep learning for time series forecasting: architectural diversity and open challenges," *Artif. Intell. Rev.*, vol. 58, p. **216**, (2025), doi: https://doi.org/10.1007/s10462-025-11223-9.