

Wave Height Forecasting in the Bay of Bengal Using Multivariate Hybrid Deep Learning Models

Phyu Sin Thet^{a,b}, Aifeng Tao^{a,b,*}, Tao Lv^{a,b}, Jinhai Zheng^{a,b}

^a Key Laboratory of Ministry of Education for Coastal Disaster and Protection, Hohai University, Nanjing 210024, China

^b College of Harbor, Coastal and Offshore Engineering, Hohai University, Nanjing, 210098, China

* Corresponding author.

E-mail address: aftao@hhu.edu.cn (A. Tao)

Abstract: The development in coastal engineering and maritime transport demands accurate wave height prediction. In this study, hybrid deep learning models, including CNN-LSTM, CNN-BiLSTM, CNN-GRU, and CNN-BiGRU, are employed to develop regional multivariate wave prediction models that incorporate multiple features, such as wave height, wind stress, water depth, pressure, and sea surface temperature (SST), for the entire Bay of Bengal area. Sensitivity analysis is performed to evaluate the accuracy using statistical metrics, such as the correlation coefficient, RMSE, and MAE. The findings demonstrate that regional multivariate models offer satisfactory results for the entire Bay of Bengal region. The multivariate model performs better compared to the univariate model as the forecast horizon increases. Performance assessment of each environmental factor, employing the integrated gradient method, reveals that sea surface temperature has the most significant influence, while wind stress is the least dominant factor in the wave prediction model. Among the tested models, the CNN-BiGRU has superior performance with a correlation of 0.9872, an RMSE of 0.1547, and an MAE of 0.1005 for the 3-hour prediction and is proposed as the optimal model. This study contributes to assessing the contribution of each environmental feature and improving the accuracy of regional wave prediction.

Keywords: CNN-LSTM, CNN-BiLSTM, CNN-GRU, CNN-BiGRU, univariate, multivariate, wave height, Bay of Bengal.

1. Introduction

Wave height prediction is a main issue in coastal engineering, maritime transport, and disaster management, especially in areas where severe and frequent wind conditions occur. Forecasting short-term wave height is highly essential for the development and utilization of energy [1] and marine engineering purposes [2]. However, analyzing and predicting wave height is challenging because of its complex nonlinear and non-stationary statistical properties [3].

The Bay of Bengal (BOB) is a semi-enclosed basin surrounded by Myanmar, Sri Lanka, Thailand, India and Bangladesh [4]. It plays a very important role in the Maritime Silk Road. It serves as a major sea lane for trade between East Asia and Europe. Wave height prediction is a challenging task in the BOB region because of its complex geometry with both shallow and deep-water areas, and the region is suffering from frequent interactions

of environmental factors, such as sea surface temperature (SST), atmospheric pressure, and wind field. These factors have an influence on wave characteristics, and it is necessary to consider their influences to improve wave prediction performance.

Traditional numerical models can make wave predictions using wind fields; however, they take hours to run and need supercomputers, while machine learning models can be run on a standard PC and provide predictions in seconds [5]. In addition, these traditional models struggle to offer satisfactory results because of the non-stationary nature of significant wave height (SWH) [6]. Although traditional models leverage both physical simulations and data-driven techniques to offer high-resolution predictions, they are computationally expensive and face significant limitations, such as time delays and uncertain accuracy [5]. In addition, this study focuses on the feasibility and potential of purely data-driven machine learning models for wave height prediction. The advancement of machine learning has made time series analysis a simple and efficient computational method because it requires only historical wave height data and learns from past data and wave patterns, reducing the computational costs to a minimum [7]. Recently, neural network techniques have gained wide applications in wave height prediction because they deliver lesser error and computational time [8].

Many researchers applied deep learning techniques in wave height prediction, but most of the studies focus on single-point wave forecasting (buoy wave height forecasting) [9]. However, regional wave predicting offers a better understanding of the spatiotemporal dynamics of wave behavior. To implement regional wave prediction, hybrid deep learning models (hereafter referred to as “hybrid ML”) should be applied because only using a convolutional neural network cannot provide the accurate prediction of SWH, and applying a hybrid model like CNN-BiLSTM can improve the prediction of SWH [9]. In this study, “hybrid ML” refers to the integration of different deep learning components, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), rather than the inclusion of physical models. They combine the strengths of different frameworks and approaches to improve the ability to handle noise in the data, capture temporal variability, and nonlinear relationships among input factors.

Hybrid deep learning models can be implemented by single variable or multivariable approaches based on the number of input features in the prediction. A single-variable approach relies on only one input feature, such as wave height, to forecast future wave height. This method is widely used because of its computational efficiency; however, wave height is influenced by a range of environmental factors. To improve the accuracy of wave prediction, the wave prediction models are performed using wave height, wind stress, and wind direction [10,11]. A multivariate analysis is based on the variation of several input features, and [11] stated that this analysis offers

a higher prediction percentage. Although hybrid deep learning models have been widely applied for wave height prediction, existing research focuses on wave prediction without fully incorporating multiple environmental factors, such as wind field, water depth, pressure, and SST, which influence wave dynamics. In this study, hybrid deep learning models like CNN-LSTM, CNN-GRU, CNN-BiLSTM, and CNN-BiGRU are applied for regional wave prediction with multivariable approaches.

Performance assessment of wave prediction models on different time horizons is crucial because they reflect different scales of prediction of interest to applications in coastal engineering, maritime operations, and disaster management. Although hybrid deep learning models have demonstrated strong performance in wave height prediction, comparative analyses of their performance over different horizons are still limited. Moreover, the contribution of additional meteorological variables has not been adequately examined. This study will contribute to addressing these limitations by developing hybrid deep learning models incorporating multiple features and evaluating models' accuracy across different time horizons.

In building wave prediction models, collecting reliable data is crucial, and many researchers have applied in-situ sensors such as buoys, which record time series of wave elevation. Buoys offer reliable sea state data representing the wave characteristics at a fixed location [12]. However, there are some limitations in collecting the wave height and environmental data from buoys in the BOB area [13]. In addition, this study focuses on regional wave prediction. Therefore, in this research, the marine meteorological data are sourced from ERA5 and GEBCO, which are reliable data sources.

This research aims to (1) evaluate the performance of univariate and multivariate models, (2) assess the influence of each environmental factor on regional wave height prediction, and (3) determine the optimal model for regional wave forecasting. The remaining sections of this paper include (1) Determining the optimal wind input variables, (2) Optimization of deep learning models using Bayesian Optimization(BO), (3) Performance comparison of univariate and multivariate models for different time horizons, (4) Performance evaluation of each environmental factor on wave height prediction, (5) Performance evaluation of hybrid models for different locations, and (6) Assessment of multivariate hybrid models' performance for different time horizons, and (7) Selecting the optimal model for wave prediction.

2. Materials and methods

2.1 Study area and data

To implement the regional wave prediction, the entire area of the BOB is selected as the study area. The employed boundary limitation ranges from 70°E to 100°E and 10°S to 23°N, including the whole region of the

BOB and extending into parts of the Indian Ocean to offer thorough coverage. Figure 1(a) illustrates the study area of this research. As this study employs a multivariable approach for regional wave prediction, multiple environmental factors are collected as the grided data. Bathymetry data is collected from the GEBCO (The General Bathymetric Chart of the Oceans), implemented by the United Nations (UN) for sustainable development. Figure 1(b) describes the bathymetry data of the study area obtained from GEBCO.

Other environmental factors such as wind field, pressure, SST, and SWH data recorded at 3-hour intervals from 2022 to 2023 are obtained from ERA5. ERA 5 is the fifth generation of ECMWF atmospheric reanalysis of the global climate with the combination of model data with observations available from 1959 to the present. Even though ERA5 can sometimes overestimate observed SWH, it generally shows good agreement with measurements at most coastal sites around India [14]. In this study, daily environmental data from ERA5 is downloaded from the Copernicus Climate Data Store: <https://cds.climate.copernicus.eu>. The employed data is in $0.5^\circ \times 0.5^\circ$ grid format, consisting of 4087 data points for each time.

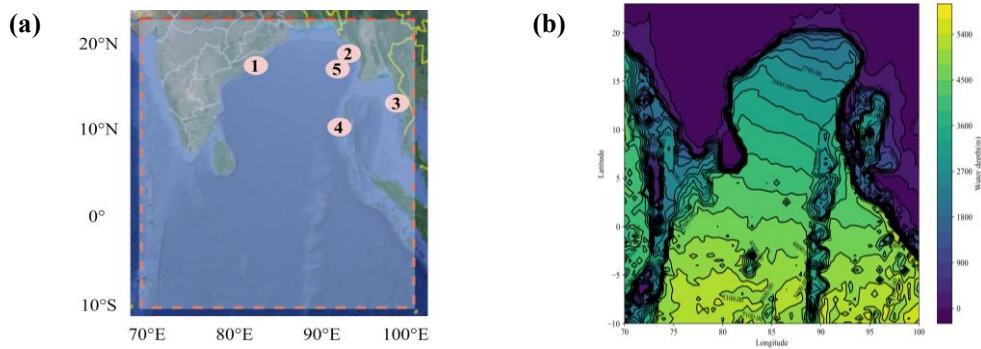


Figure 1. (a) Study area including different locations across the BOB for performance evaluation: Point 1- Visakhapatnam (Vizag), Point 2-Kyauk Phyu, Point 3-Dawei, Point 4-Port Blair, Point 5- Offshore point; (b) Bathymetry chart of the study area (BOB).

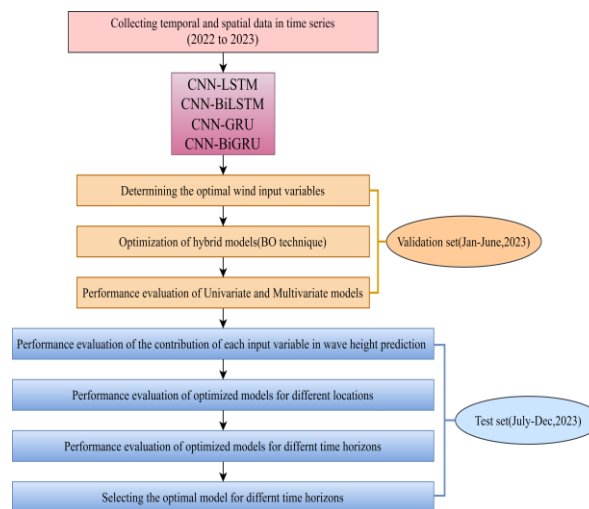


Figure 2. Flow chart of this study.

Figure 2 presents the flowchart of this study. To balance accuracy and computational efficiency for the regional dataset, the 3-hourly input data from GEMCO and ERA5 for the period 2022 to 2023 are collected. To avoid temporal information leakage and ensure that each dataset does not share future information, the dataset is organized into three subsets: the training set ranges from January 2022 to December 2022 (total 11,944,040 data points per feature), the validation set covers from January to June 2023 (total 5,928,196 data points per feature), and the test set spans from July to December 2023 (total 6,020,384 data points per feature).

2.2 Convolutional Neural Networks (CNNs)

CNNs are a class of deep learning models based on layered neurons and consist of convolutional and pooling layers for feature extraction [15,16], followed by fully connected layers for final prediction. The number and size of convolutional kernels, the stride, and the padding define convolutional layers. While the kernel size is a hyperparameter set, the kernel weights are learned through backpropagation. The Eq. (1) describes the convolution operation in a CNN, where $x_{i+i',j+j',k'}^{(l-1)}$ is the input value at position $(i+i',j+j')$ in the k' -th channel of the previous layer, $z_{i,j,k}^{(l)}$ is the output at layer (l) , $\omega_{i',j',k',k}^{(l)}$ is the kernel weight connecting it to the k -th output channel, $d^{(l-1)}$ is the number of channels in the previous layer, r refers to the kernel radius, and f is an activation function to introduce non-linearity. CNNs learn spatial features through backpropagation, where the weights ω^l are updated to minimize the loss function. The architecture of CNNs is illustrated in Figure 3a, and the corresponding equations are as follows:

$$z_{i,j,k}^{(l)} = \sum_{k'=1}^{d^{(l-1)}} \sum_{i'=-r}^r \sum_{j'=-r}^r x_{i+i',j+j',k'}^{(l-1)} \cdot \omega_{i',j',k',k}^{(l)} \quad (1)$$

$$x_{i,j,k}^{(l)} = f(z_{i,j,k}^{(l)}) \quad (2)$$

2.3 Long Short-Term Memory Networks (LSTMs)

LSTMs are a type of Recurrent Neural Network (RNN) proposed for sequential data, such as wave height forecasting. LSTM architecture includes the memory cell, which is controlled by input, forget, and output gates [17], and their structure is presented in Figure 3b. Unlike other traditional RNNs, LSTMs can learn long-term dependencies using gated control. Sigmoid and tanh functions control information flow within the memory cell. Eq. (3) represents the sigmoid function, where $\sigma(x)$ is sigmoid activation function, and x is the input.

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (3)$$

Forget Gate (f_t) determines how much past information should be forgotten using the sigmoid function ($f_t \approx 0$ means forget; $f_t \approx 1$ refers to retain).

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (4)$$

where W_f is the weight matrix for the forget gate, h_{t-1} is the previous hidden state, x_t is the current input at time step t , b_f is the bias term for the forget gate.

Input Gate(i_t) controls how much new information is necessary to add to the cell state. If i_t is close to 0, the information will be ignored. If i_t is close to 1, the information will be added.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (5)$$

Cell state($\tilde{C}_t$) applies the tanh function and compresses the values between -1 and 1.

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (6)$$

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \quad (7)$$

where C_t refers to the update to the cell state, $f_t \cdot C_{t-1}$ controls how much memory is necessary to be kept and $i_t \cdot \tilde{C}_t$ controls how much new information will be added.

Output gate (O_t) decides how much of the cell state will be sent to the hidden state. The equation is

$$O_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (8)$$

where W_o is the weight matrix for the output gate. If O_t is close to 0, the cell state will be hidden. The cell state will be passed on if O_t is close to 1.

The final output h_t decides how much of the information will be sent as output.

$$h_t = O_t \cdot \tanh(C_t) \quad (9)$$

2.4 Gated Recurrent Units (GRU)

GRU is another type of RNN, similar to LSTM, with more simplicity in its structure. While LSTM has memory cells that increase memory consumption, GRU has gating units to regularize the flow of information without separate cells [18]. With fewer parameters than LSTM, GRU has only reset gate (r_t), which controls how much past information to remember, and update gates (z_t), which controls the contribution of new input into the hidden state [19], shown in Figure 3c. Trainable weight parameters $W_r, W_z, U_r, U_z, W_h, U_h$, biases b_r, b_z, b_h , the candidate hidden state $\tilde{h}_t$, new hidden state h_t , and the sigmoid function σ regulate the information flow. If z_t is equal to 1, the unit ignores the previous hidden state and fully updates from $\tilde{h}_t$. If r_t is equal to 0, the influence of h_{t-1} is removed when computing $\tilde{h}_t$, effectively forgetting the past [18]. The relevant equations are as follows:

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \quad (10)$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \quad (11)$$

$$\tilde{h}_t = \tanh(U_h (r_t \cdot h_{t-1}) + W_h x_t + b_h) \quad (12)$$

$$h_t = (1 - z_t) \cdot h_{t-1} + z_t \cdot \tilde{h}_t \quad (13)$$

2.5 Bidirectional LSTM (BiLSTM) and Bidirectional GRU (BiGRU)

Bidirectional LSTM (BiLSTM) and Bidirectional GRU (BiGRU) are extensions of the traditional LSTM[20] and GRU architectures by processing of sequential data in both forward and backward directions. It consists of two networks: one that processes the forward direction of the input and another that processes the reverse direction. The output from both networks is combined to produce the final prediction. At each time step i , the forward LSTM/GRU receives input x_i and the previous hidden state h_{i-1} , and the backward LSTM/GRU receives input x_i and the previous hidden state h_{i+1} . The hidden states from both directions are concatenated to form the final output (y_i). The structure is shown in Figure 3d.

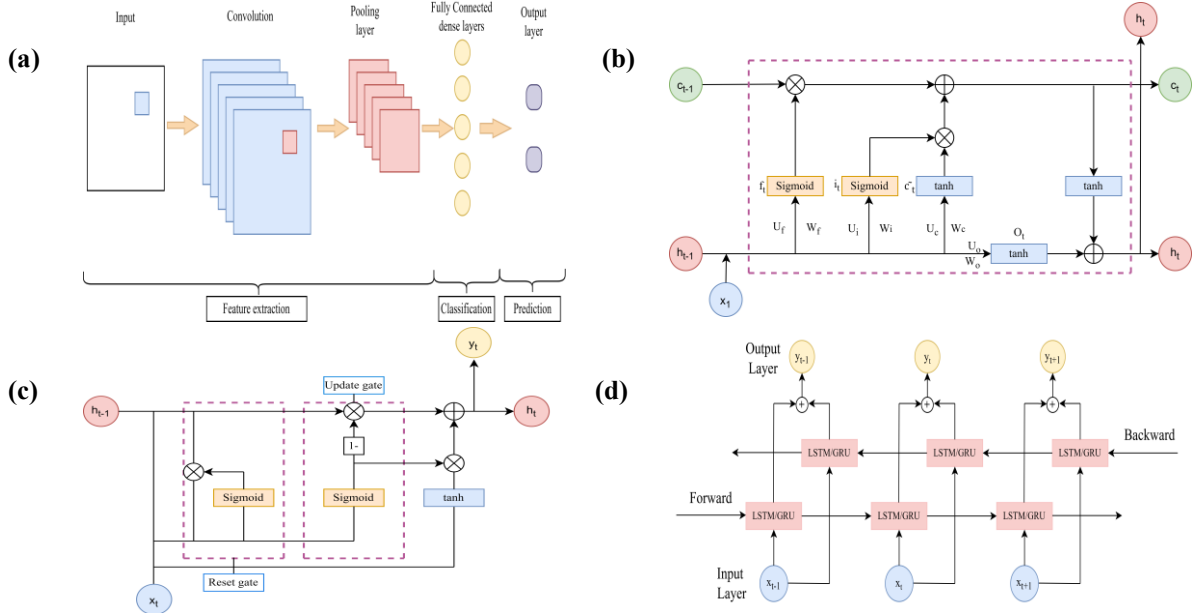


Figure 3. Architecture of (a) CNN, (b) LSTM, (c) GRU, and (d) BiLSTM vs. BiGRU.

2.6 Hybrid deep Learning Models with CNN and RNN Components

Hybrid deep Learning Models combine the efficiency of CNNs in obtaining spatial features with those of RNNs (LSTM, BiLSTM, GRU, and BiGRU) in capturing sequential dependencies in the data. A sequence of 2D spatial grids with multiple features is processed through TimeDistributed Conv2D and MaxPooling2D layers to extract spatial features at each time step. The resulting features are flattened and passed to RNN layers to learn temporal relationships. The output is then passed through the Dense and Reshape layers to generate predictions.

2.7 Bayesian Optimization (BO)

Applying optimal hyperparameters is crucial in machine learning, but as more parameters increase, the search space expands. Grid search and random search methods consume a long time and many resources [21]. To overcome this, the Bayesian Optimization (BO) algorithm has been explored [21,22]. BO can be applied to the

training data to identify the most effective network hyperparameters [23] and is a fast and effective method for hyperparameter tuning. It starts by making a prior guess about the objective function's probability and updates this assumption using Bayesian inference. This method utilizes Gaussian processes for the prior and applies Bayesian linear regression with kernel methods [24]. This method has two main components: a surrogate model (Gaussian process), which predicts performance for different hyperparameters, and a loss function, which determines the next promising point based on previous results and finds a near-optimal hyperparameter combination within fewer steps [22].

BO models $f(x)$ employ a Gaussian Process (GP) as the surrogate model and the Expected Improvement (EI) acquisition function to evaluate the globally optimal hyperparameters [25]. At each iteration, the next input point is selected by maximizing the EI. $Z = \frac{\mu - f(x^*)}{\sigma(x)}$ represents the standardization of the difference between the predicted mean ($\mu(x)$) and standard deviation ($\sigma(x)$) of the GP at the input x , Φ is the cumulative distribution function (CDF) of the standard normal distribution, and ϕ is the probability density function of the standard normal distribution. The conditional probability of a target variable y , denoted as $(y|x; D) = N(y|\mu(x), \sigma^2(x))$, represents a Gaussian distribution where D is the configuration space for hyper-parameters. The relevant equations are as follows:

$$EI(x) = \begin{cases} (\mu(x) - f_n^*)\Phi(Z) + \sigma(x)\phi(Z) & , \sigma(x) > 0 \\ 0, & \sigma(x) = 0 \end{cases} \quad (14)$$

2.8 Multicollinearity analysis

Multicollinearity refers to a linear relationship among multiple predictor variables, which affects the reliability of parameter estimation in the models [26]. The interconnections among various meteorological parameters can be applied to compensate for missing data [27]. Although many environmental features influence wave height prediction, incorporating all of them is not necessary. Incorporating all of them can lead to overfitting and reduce model performance. The Variance Inflation Factor (VIF) is a statistical metric used to find the presence of multicollinearity among independent variables in a regression model. The higher the value of VIF, the higher the collinearity between the input variables [28]. VIF_i defines the Variance Inflation Factor for the i -th predictor variable, R_i^2 refers to the coefficient of determination from a regression of the i -th independent variable on all the other independent variables [28]. VIF values can be calculated using Eq. (15):

$$VIF_i = \frac{1}{1 - R_i^2} \quad (15)$$

2.9 Integrated Gradient

Integrated gradient (IG) is a gradient-based feature attribution technique that measures the importance of input features to indicate their relevance to the output [29]. IG enhances interpretability by accumulating gradients along a linear path from a baseline input (such as zero) to the actual input, which provides a more reliable assessment of feature importance. Although deep learning models are capable of making accurate predictions, interpreting which input features are driving the predictions is still challenging. Applying IG is beneficial for assessing the influence of input features on the current prediction [30]. This method accumulates gradients along a straight-line path between a baseline input x' and the actual input x , capturing the contribution of each input feature along this path. $IG_i(x)$ can be computed by Eq. (16) :

$$IG_i(x) = (x_i - x'_i) \int_{\alpha=0}^1 \frac{\partial F(x' + \alpha(x - x'))}{\partial x_i} d\alpha \quad (16)$$

where α scales the input along a straight-line path from the baseline to the input. $\frac{\partial F}{\partial x_i}$ is the gradient of the model output with respect to the i -th input feature [30].

2.10 Evaluation of model performance

The different models' performance is evaluated by applying various statistical metrics such as correlation, RMSE, and MAE. In general, the correlation(R) value lies between -1 and 1. If the R is closer to 1, the model is said to have a high positive correlation, and the model results are closer to the actual observations. The Root Mean Squared Error (RMSE) is a measure of the square root of the average squared differences between the model and actual values. Mean Absolute Error (MAE) calculates the average magnitude of the absolute differences between the model and actual values.

$$\text{Correlation}(R) = \frac{\sum_{i=1}^n (P_i - \bar{P})(Q_i - \bar{Q})}{\sqrt{\sum_{i=1}^n (P_i - \bar{P})^2} \sqrt{\sum_{i=1}^n (Q_i - \bar{Q})^2}} \quad (17)$$

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (P_i - Q_i)^2} \quad (18)$$

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |P_i - Q_i| \quad (19)$$

where, P_i is the model value, Q_i is the actual data, n is the number of observations, $\bar{P}$ is the mean of P_i and $\bar{Q}$ is the mean of Q_i .

2.11 Models' network design

In this research, a two-dimensional CNN is applied to extract the hidden features from the input variables. By stacking convolutional and pooling layers, spatial feature information that influences SWH can be captured. The spatial features (SWH, wind stress, water depth, pressure, and SST) are concatenated along the

channel dimension, and each spatial grid at the provided time step is treated as a multi-channel (shape: $67 \times 61 \times 5$).

Figure 4 describes the process of hybrid deep learning models employed in this research. The input accepted for the model is of shape $(67, 61, 5)$, which indicates that each time step includes 67×61 spatial grids with five features. A TimeDistributed Conv2D layer with 64 filters and a kernel size of $(1, 1)$ is first applied to extract the local spatial features independently at each time step and to produce an output shape of $(67, 61, 64)$. Then, the TimeDistributed MaxPooling2D layer with a pool size of $(1, 5)$ is applied, which in sequence reduces the longitudinal dimension to $(67, 12, 64)$. To flatten these spatial features into a vector (with a shape of $51456 = 67 \times 12 \times 64$), a TimeDistributed Flatten layer is applied. RNNs are employed to capture temporal dependencies with 64 units in each direction. The Dense layer transforms this into a vector of length 4087 (67×61 spatial grid points). Then, they are converted back into a two-dimensional spatial grid through a reshape layer. Finally, a fully connected dense layer is applied to predict spatial wave distribution. All simulations are performed with Python 3.9.18 and TensorFlow 2.19.0.

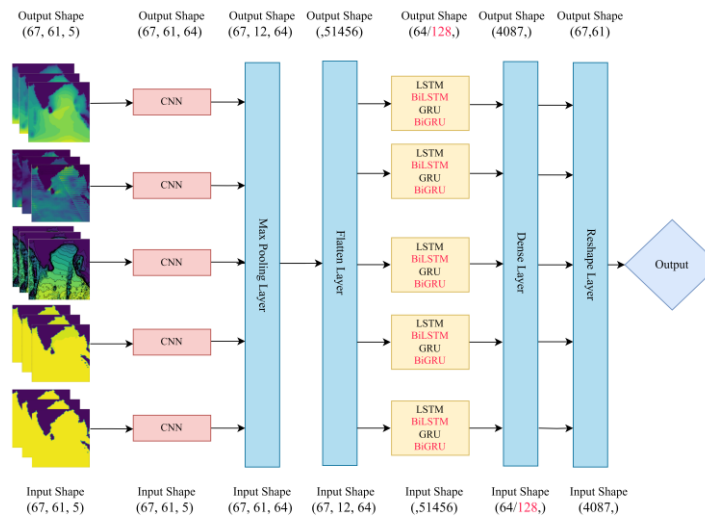


Figure 4. Process of hybrid deep learning models.

2.12 Determining the optimal wind input variables

In wave prediction studies, considering wind-related variables may improve the models' performance, as wind is a key dominant factor in wave formation and propagation. Wind speed is the most significant parameter in wave height prediction [31]. However, wind shear velocity is applied as a parameter instead of the wind speed to improve the prediction in cyclone conditions [10]. In addition, wind stress plays an important role in the study of the atmosphere and ocean air-sea interaction, ocean modeling, and ocean forecasting [32].

In this study, multiple sets of analyses are conducted based on different wind-related variables, such as wind stress, wind shear velocity, and wind speed, to determine the ideal wind input variable to couple with the proposed

models. As defined by [10], the wind shear velocity (U^*) and wind resistance coefficient formula (C_D) follow Eq. (20) and (21). The wind stress magnitude formula (τ) follows Eq. (22) where V_a is the wind speed at 10 m above the sea surface, ρ_a is the density of air and C_D is a dimensionless drag coefficient [32].

$$U^* = U_{10} \quad (20)$$

$$C_D = \begin{cases} 1.2875 \times 10^{-3}, & U_{10} < 7.5 \text{ m}\cdot\text{s}^{-1} \\ (0.8 + 0.065 \times U_{10}) \times 10^{-3}, & U_{10} \geq 7.5 \text{ m}\cdot\text{s}^{-1} \end{cases} \quad (21)$$

$$\tau = \rho_a C_D V_a^2 \quad (22)$$

To identify the optimal wind input variable, a hybrid CNN-LSTM model is constructed. The initial model configuration is structured in three main layers: an input layer, a hidden layer, and an output layer. CNN-LSTM has better prediction results for short-term prediction [16], and initial model configurations are based on their work. The model applies a two-dimensional CNN with 64 filters, a kernel with a size of 1×1 , the moving step is 1, and the size of the pooling layer is 1×5 . A long Short-Term Memory Network with 64 units and a batch size of 256 is implemented [16]. The model employs the ReLU activation, an Adam optimizer with a learning rate of 0.0001, a dropout rate of 0.2, and early stopping against overfitting with a patience of 10. The early Stopping method is employed to reduce overfitting and decrease unnecessary training time [8]. Min-max normalization is applied to scale the dataset values within the range of 0 to 1. The evaluation metrics are calculated to determine the performance of various wind input variables.

3. Results and Discussion

3.1 Determining the optimal time steps

It is essential to determine the optimal time step in order to identify how historical data impacts the wave height prediction. The optimal time step varies depending on the complexity of the relationship between input features. The Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) are used to explore temporal dependencies and identify possible trends and patterns within time series datasets [33]. Therefore, in this study, these two methods are employed to determine the optimal time step by learning the trends of each feature. The ACF method measures the linear relationship between observations at various lag intervals. The PACF method evaluates the direct correlation between a time series and its past values after removing the influence of intermediate lags.

As shown in Figure 5, there is a strong temporal dependency in the wind variables, with autocorrelation values above 0.8 up to lag 10. The PACF plot describes strong significant spikes at Lag 1 and Lag 2, which

indicates that the most recent wind input variables within a two-step window offer the most informative predictors for future conditions. According to these, the performances at every time steps 1, 2, and 3 are conducted based on the validation set to determine the optimal time step that can capture the temporal variation in wave height. The results show that using wind stress with a time step of 1 offers better performance, as described in Table 1.

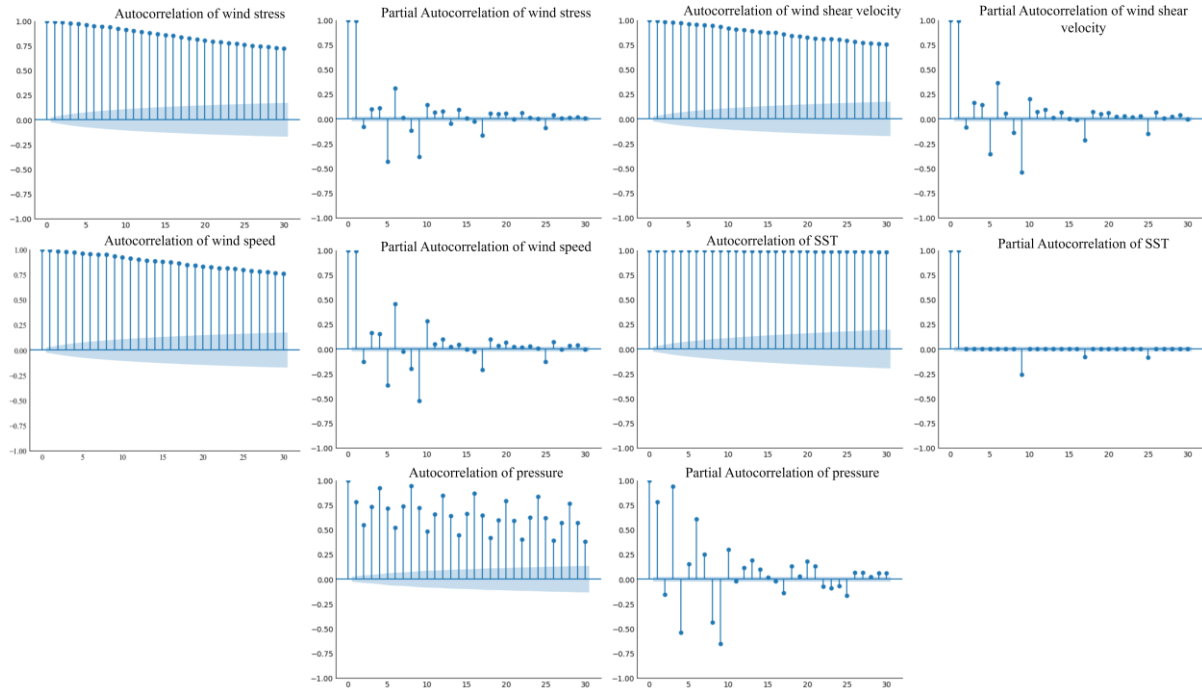


Figure 5. Autocorrelation and partial autocorrelation plots of input features.

Table 1. Sensitivity analysis of time steps.

	SWH + Wind Stress			SWH + Wind Shear Velocity			SWH + Wind Speed			Multivariate model		
Time steps	1	2	3	1	2	3	1	2	3	1	2	3
R	0.9781	0.9725	0.9773	0.9678	0.9737	0.9692	0.9647	0.9700	0.9574	0.9410	0.9773	0.9678
RMSE	0.1677	0.1871	0.1704	0.2031	0.1837	0.1987	0.2116	0.1953	0.2324	0.2722	0.1704	0.2035
MAE	0.1008	0.1089	0.0977	0.1242	0.1086	0.1202	0.1270	0.1153	0.1379	0.1669	0.1004	0.1236

Determining the optimal time step for wave prediction with the multivariable framework is also necessary because applying large time steps will lead to the loss of critical information in the input features while adopting small time steps might cause noise and increase computational costs without notable enhancements in model performance. Therefore, sensitivity analysis of the time steps for the multivariate model is performed, and the selection of time steps is conducted, as shown in Figure 5. Table 1 illustrates that a time step of 2 offers superior performance for a multivariate model. To evaluate the presence of multicollinearity among the input variables, VIF is calculated for each feature and demonstrated in Figure 6. VIF values of each feature are less than 5, which

means that there is no serious multicollinearity among the selected features, and they are sufficiently independent without concern for redundant information.

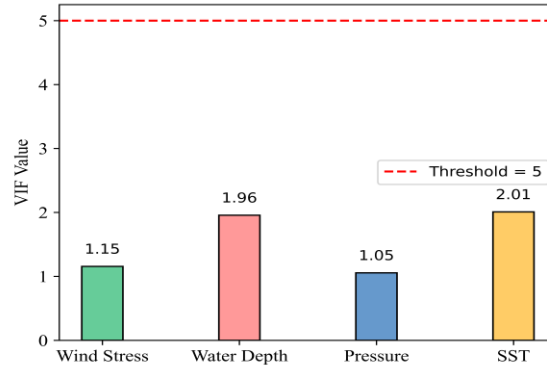


Figure 6. VIF of each input variable.

3.2 Optimization of hybrid models

To enhance the efficiency and performance in wave height prediction, optimization of models is one of the main factors. The models learn the pattern by training the data from January to December 2022, and fine-tuning is implemented regarding the validation set, which ranges from January to June 2023. This study employs the BO technique through the Keras Tuner framework. The model handles spatiotemporal input data organized by time steps, latitude, longitude, and relevant features. The hyperparameter search space includes the number of the convolutional filters and RNNs' units, which range from 32 to 128 with increments of 32. To prevent overfitting, the dropout rate is adjusted from 0.1 to 0.5.

The Adam optimizer is implemented with a learning rate between $1e-5$ and $1e-3$. A larger batch size reduces the models' generalization performance due to the adverse impacts associated with low-gradient noise [34]. In addition, a larger batch size will accelerate the computation, which will decrease the updates required for the network, and a batch size of 32 is a good one [35]. Therefore, a batch size of 32 is adopted as a balanced and practical choice. The tuner is configured to increase model performance by minimizing the validation loss up to 20 trials. Each trial involved training with a maximum of 50 epochs with early stopping. Moreover, multiple levels of LSTM or GRU are experimented with since the two-layer LSTM model provides better model performance [11]. Therefore, this study employs multiple layers of CNN and RNNs, integrating with BO. Figure 7 (a-c) shows heatmaps of statistical metrics for comparing the performance of multiple layers and reveals that implementing two layers of CNN and RNNs offers better performance. The set of hyperparameters with the lowest validation loss during the search is selected as the optimal configuration.

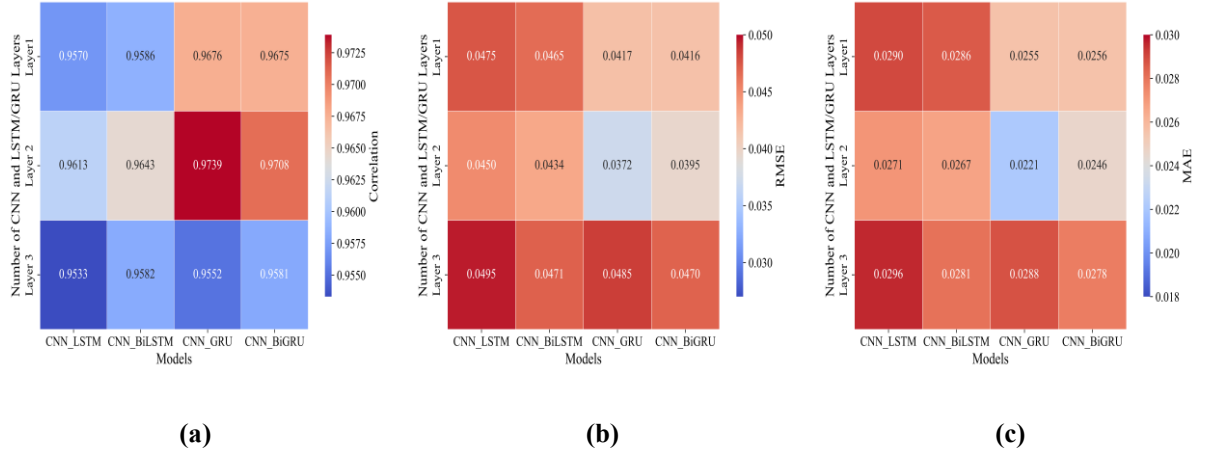


Figure 7. Statistical matrix heatmaps of Bayesian Optimization: (a) Correlation, (b) RMSE, and (c) MAE.

Error! Not a valid bookmark self-reference. describes hyperparameter tuning values and results conducted using the BO technique. The degree of improvement between initial and optimized configurations is illustrated in Table 3. The result indicates that the optimized hybrid CNN-LSTM model has an improvement of about 73.53% in RMSE and about 73.00% in MAE. The performance of the optimized CNN-BiLSTM has increased by about 71.37 % in RMSE and about 70.10% in MAE. Similarly, for the CNN-GRU model, a corresponding performance enhancement of 77.63% and 77.42% in RMSE and MAE, respectively. For the CNN-BiGRU model, it has a degree of improvement for RMSE by 72.89% and for MAE by 71.11 %.

Table 2. Hyperparameter tuning: values and results.

Parameter	Search interval	CNN-LSTM	CNN-BiLSTM	CNN-GRU	CNN-BiGRU
Number of CNN layers	[32,128]	1 st layer:96 2 nd layer:32	1 st layer:64 2 nd layer:64	1 st layer:128 2 nd layer:128	1 st layer:96 2 nd layer:32
Number of RNN layers	[32,128]	1 st layer:64 2 nd layer:128	1 st layer:32 2 nd layer:96	1 st layer:128 2 nd layer:128	1 st layer:64 2 nd layer:128
Learning rate	[1e-5 ,1e-3]	0.0003	0.00008	0.0009	0.0003
Dropout rate	[0.1, 0.5]	0.1	0.1	0.1	0.1

Table 3. Degree of improvement of optimized hybrid models' performance.

Models	Initial setting			Optimized setting			Degree of Improvement (%)	
	R	RMSE	MAE	R	RMSE	MAE	RMSE	MAE
CNN-LSTM	0.9773	0.170	0.1004	0.9613	0.0450	0.0271	73.53	73.00
CNN-BiLSTM	0.9821	0.1516	0.0893	0.9643	0.0434	0.0267	71.37	70.10
CNN-GRU	0.9786	0.1659	0.0979	0.9739	0.0371	0.0221	77.63	77.42
CNN-BiGRU	0.9834	0.1457	0.0848	0.9707	0.0395	0.0245	72.89	71.11

To determine whether the optimized models overfit or underfit the dataset, the training and validation loss curves are analyzed, as presented in Figure 8 (a-d). From observation, both losses remain very low and close, with

minimal fluctuations between 30-40 epochs. This close overlapping of the two curves demonstrates that the model has good generalization ability and performs without overfitting or underfitting. This confirms that these models with BO technique are reliable for further application or testing.

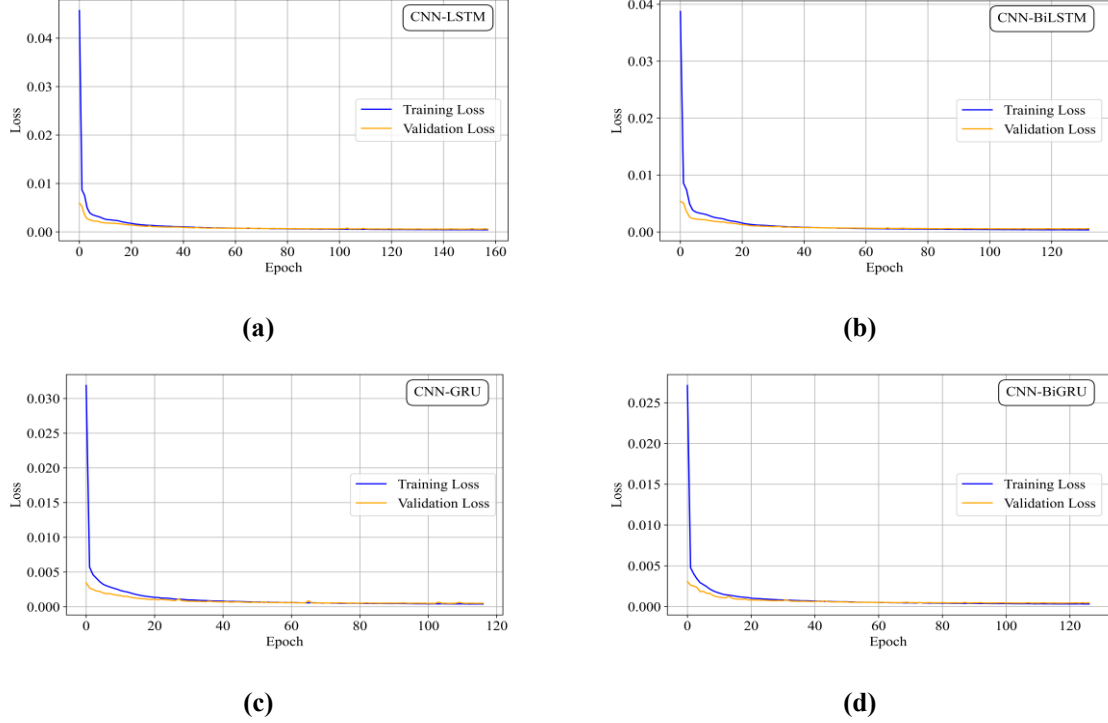


Figure 8. Training and validation loss curves of hybrid deep learning models: (a) CNN-LSTM, (b) CNN-BiLSTM, (c) CNN-GRU, and (d) CNN-BiGRU.

3.3 Performance evaluation of univariate and multivariate models

This study evaluates the performance of univariate and multivariate models across different time horizons (3, 6, 9, 12, 15, 18, 21, and 24 hour forecasts) using a hybrid CNN-LSTM model. Evaluation is conducted based on the test set covering from July to December 2023. The results presented in Figure 9(a-c) reveal that the performance of the multivariate model improves as the prediction horizon increases. The univariate model offers better performance in the case of short-term predictions (3, 6, and 9 hours). However, for long-term forecasting (12, 15, 18, 21, and 24 hours), prediction performance improves when incorporating multiple variables, such as wind stress, water depth, pressure, and SST. This enhancement in accuracy may be because the complex environmental influences on wave behavior can be better captured through the ability of multiple inputs over extended periods.

This result is consistent with the findings of [36], who observed that for one-step forecasting, using only the historical values of the target variable (in their case, photovoltaic output power) is sufficient to achieve accurate forecasting compared to multivariate models. This is because, in short-term prediction, the recent history of the

target variable contains most of the information needed to predict its immediate future, due to strong temporal autocorrelation. In contrast, for long-term forecasting, the influence of the target variable's recent history becomes less dominant as the forecast horizon extends. Therefore, additional features begin to play a larger role by capturing complex delayed interactions, which helps improve accuracy for longer lead times. It can be concluded that using a single variable input (wave height) is sufficient for short-term regional wave height forecasting, but for long-term prediction, applying multiple features is much better.

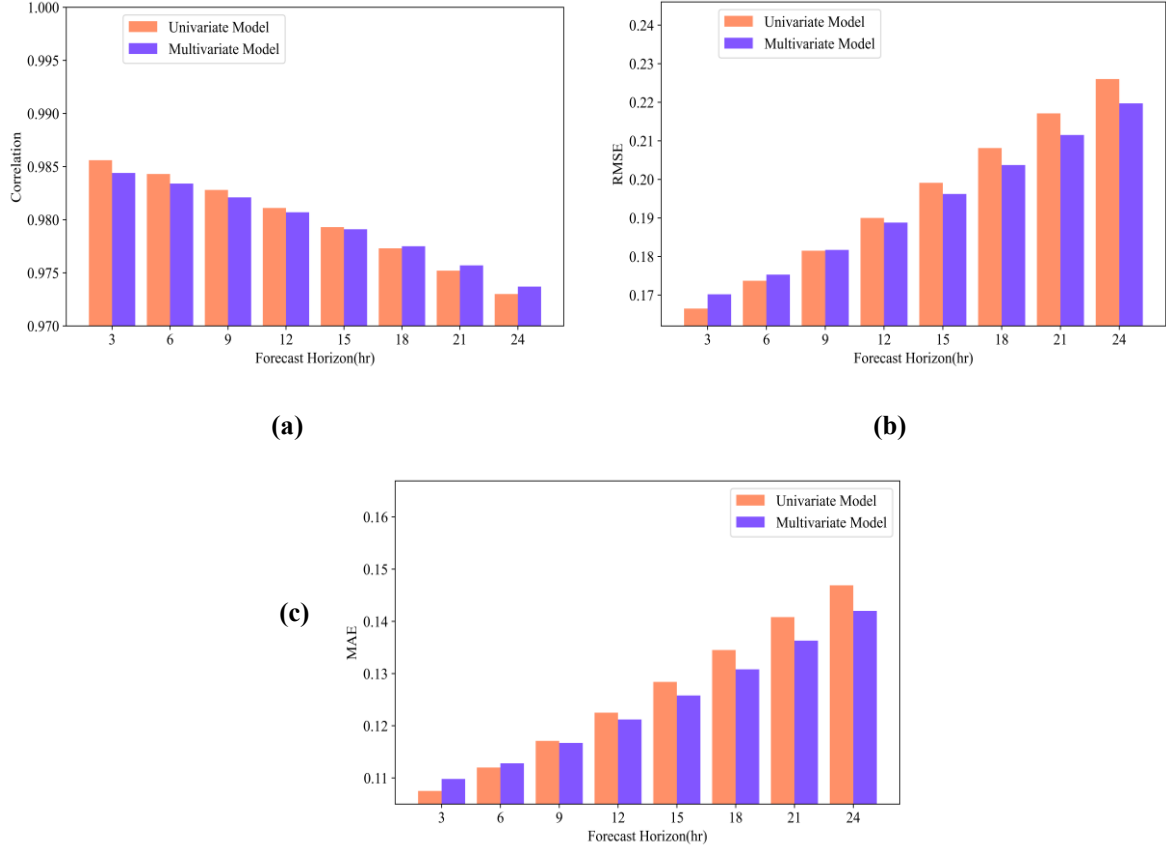


Figure 9. Statistical metrics of univariate vs. multivariate models: (a) Correlation, (b) RMSE, (c) MAE.

3.4 Contribution of each input variable in wave height prediction

To evaluate the contribution of each input variable in wave height prediction, the IG method is implemented based on the test set. Figure 10 describes the mean absolute integrated gradient values of each feature in wave height prediction. It can be observed that SST has the strongest influence, with mean absolute integrated gradient values increasing from 0.173 at the 3-hour prediction to 0.184 at the 24-hour forecast. Pressure plays the second most influential feature, with a contribution from 0.112 to 0.122 over the forecast steps. Water depth demonstrates a moderate impact, with values ranging from 0.105 to 0.111. Wind stress contributes the least among all features, from 0.044 to 0.049.

Although wind stress is the primary direct driver of wave growth, SST can indirectly influence wave height by modulating large-scale atmospheric conditions, cyclone intensity, and seasonal variability. This means that SST may appear to be the most influential predictor in the model, even if it is not the direct physical forcing of waves. This finding is consistent with [37], who found that SST had a distinct influence on wave prediction. Similarly, the analysis by [38] showed that adding location (latitude and longitude) contributed the most, followed by SST. In this study, the use of spatial grids already captures locational effects, and SST becomes the most influential predictor because it brings additional information related to seasonal and cyclone conditions. This aligns with the physical understanding of coastal processes, where SST modulates local wind-driven wave generation. Pressure plays a second role as it creates wind patterns, which directly influence wave generation. Water depth has moderate impacts because it influences waves when they approach the shore through processes like refraction and shoaling. Wind stress has the least impact because its influence on wave generation is already indirectly captured through SST and pressure.

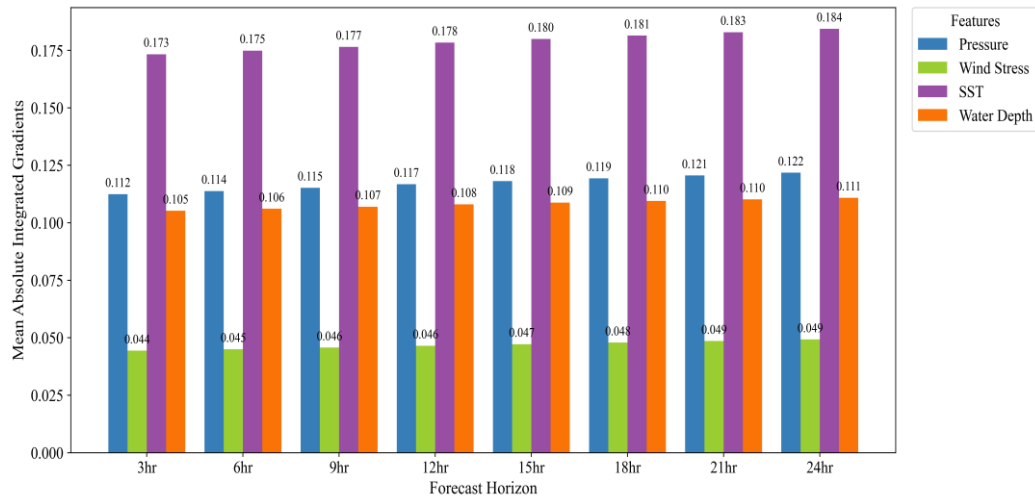


Figure 10. Mean absolute integrated gradient values of each feature in wave height prediction.

3.5 Evaluation of Model Generalizability

To address the generalizability of the proposed models, time series k-fold cross-validation with $k=5$ is implemented on four hybrid deep learning models by ensuring the validation set directly follows its corresponding training set. Figure 11 reveals that all models achieve high correlation coefficients, low RMSE, and low MAE which indicate that the models have stable and consistent predictive performance. Among the four models, the CNN-BiGRU model consistently outperforms the others in every fold and achieves the highest average correlation value of 0.9854, the lowest average RMSE of 0.1608, and the lowest average MAE of 0.1005.

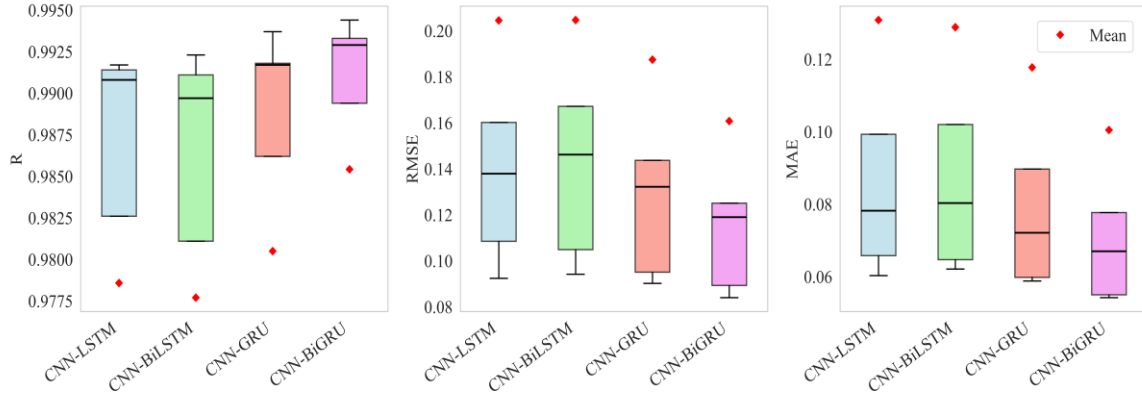
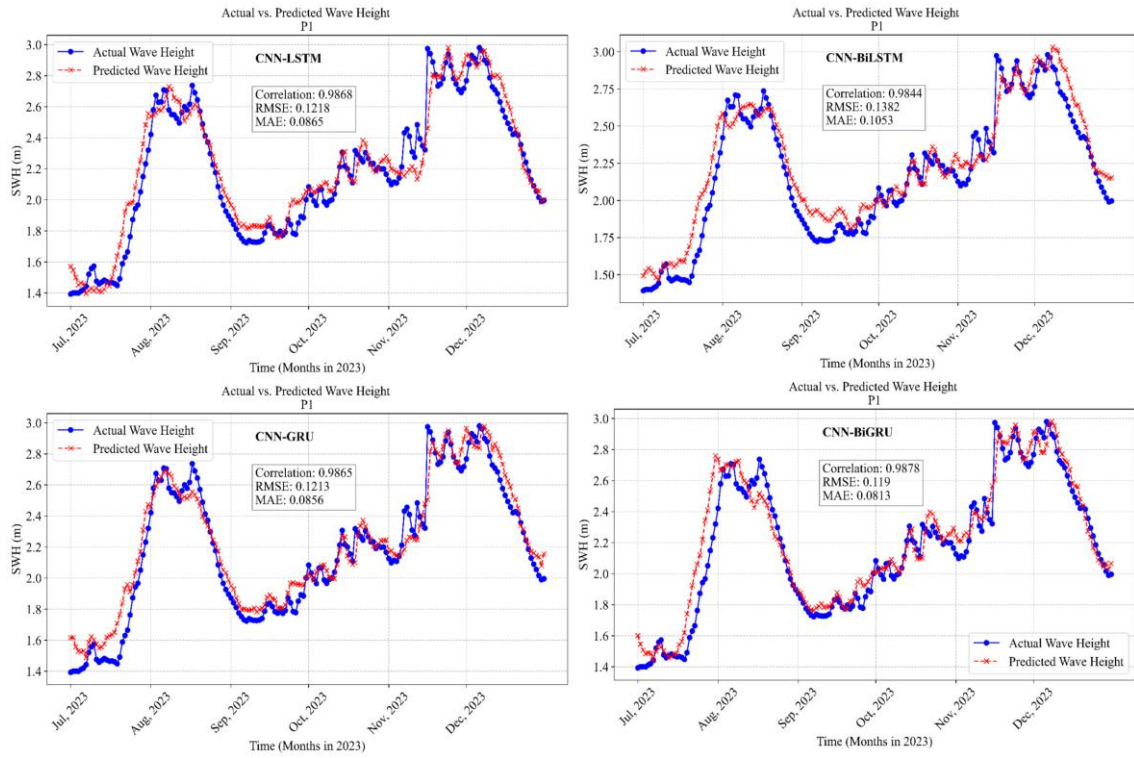


Figure 11. Evaluation of model performance metrics using time series k-fold cross-validation.

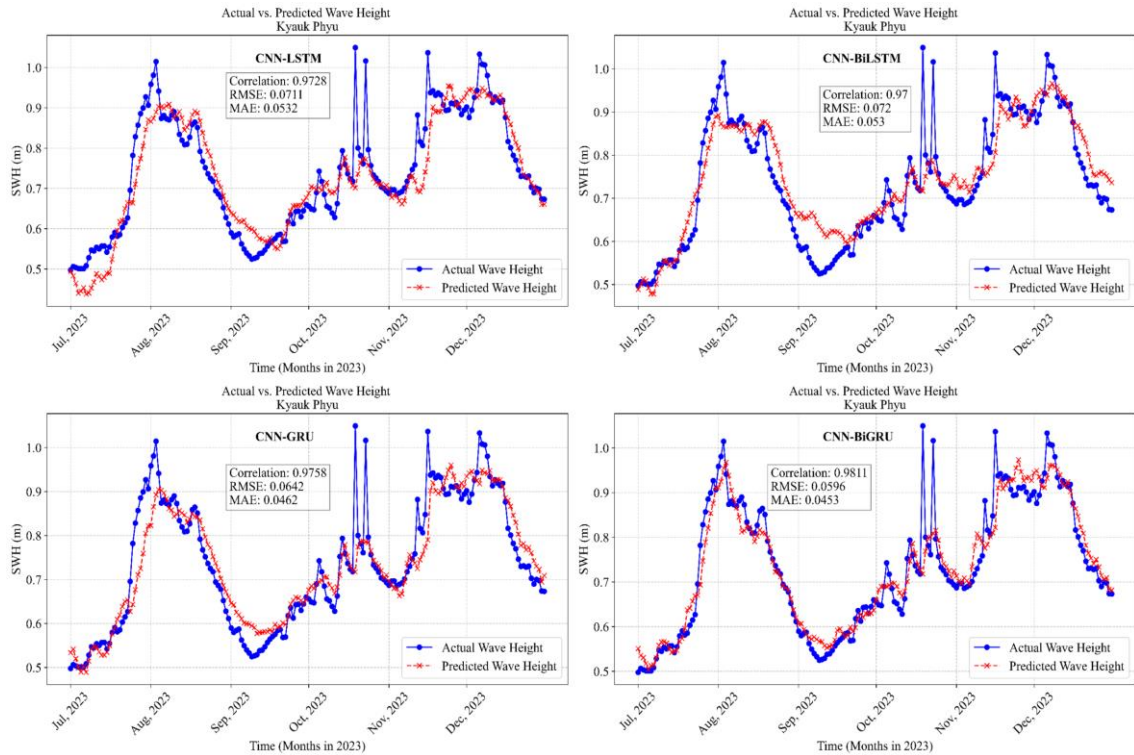
3.6 Performance evaluation of hybrid models for different locations

As this study is implemented for regional wave prediction, it is essential to assess the models' performance across different locations to ensure their accuracy. Therefore, in this study, five different locations within the BOB, such as P1(an offshore location at latitude 18.5°N and longitude 91.5°E), Kyauk Phyu at 19.5°N, 93.5°E, Dawei at 14°N, 98°E, Port Blair at 11.5°N, 92.5°E, and Visakhapatnam (referred to as “Vizag”) at 17.5°N, 83.5°E, are selected for performance evaluation, as illustrated in Figure 1. These locations represent a range of coastal environments in the BOB. The evaluation is conducted on the test set ranging from July to December 2023 to implement an assessment of model ability for unseen data. The evaluation metrics of various hybrid models are calculated for these locations to evaluate the accuracy of each model at different sites, shown in Figure 12(a-e).

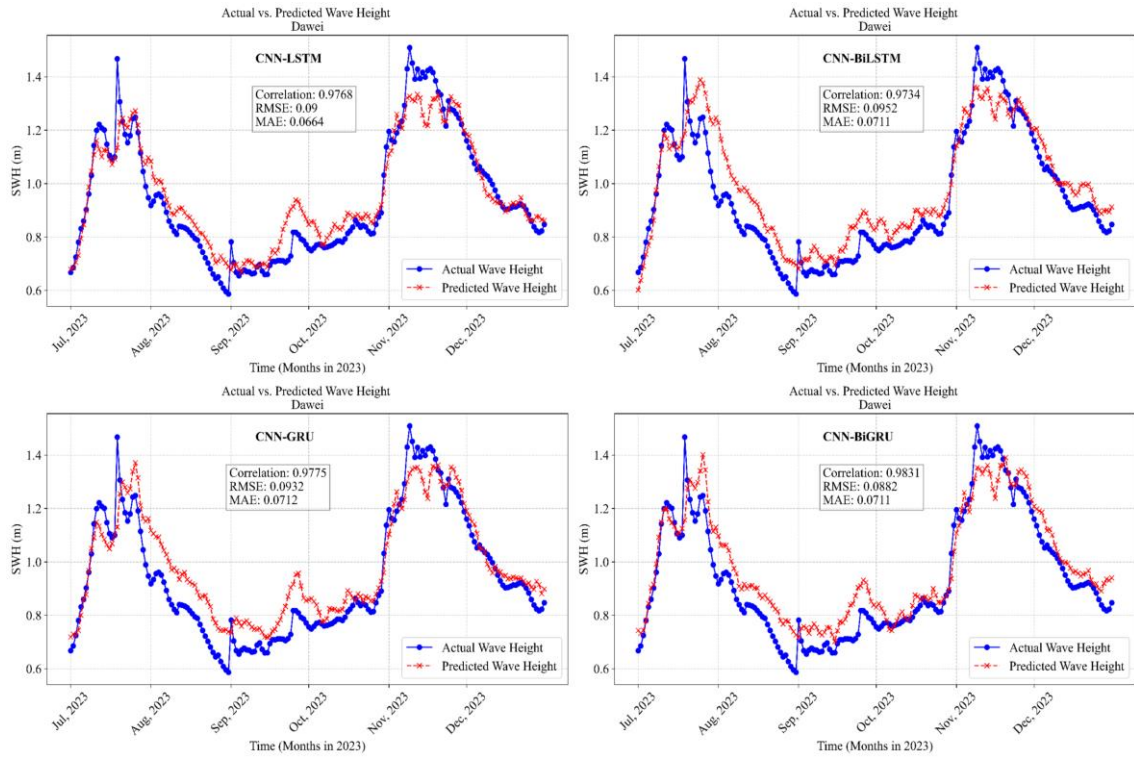
The model predicts significant wave height at a 3-hourly interval, and all performance metrics (correlation, RMSE, MAE) are computed using the full 3-hourly data to evaluate accuracy. For clarity of visualization in Figure 12, only daily points are plotted instead of all 3-hourly points. The results show that all models can predict well for different locations with good statistical metrics values. The correlation values exceed 0.9 for all locations, and the maximum RMSE and MAE values of these locations are around 0.16 and 0.11, respectively. The results indicate that the prediction performance of the models for these locations is also relatively high, and the performance of four hybrid models is satisfactory. It is important to note that in this study, the hybrid ML models are trained, validated, and tested using ERA5 SWH, and their performance is compared against ERA5 modeled wave data as a baseline due to the limited availability of in-situ buoy observations in the BOB.



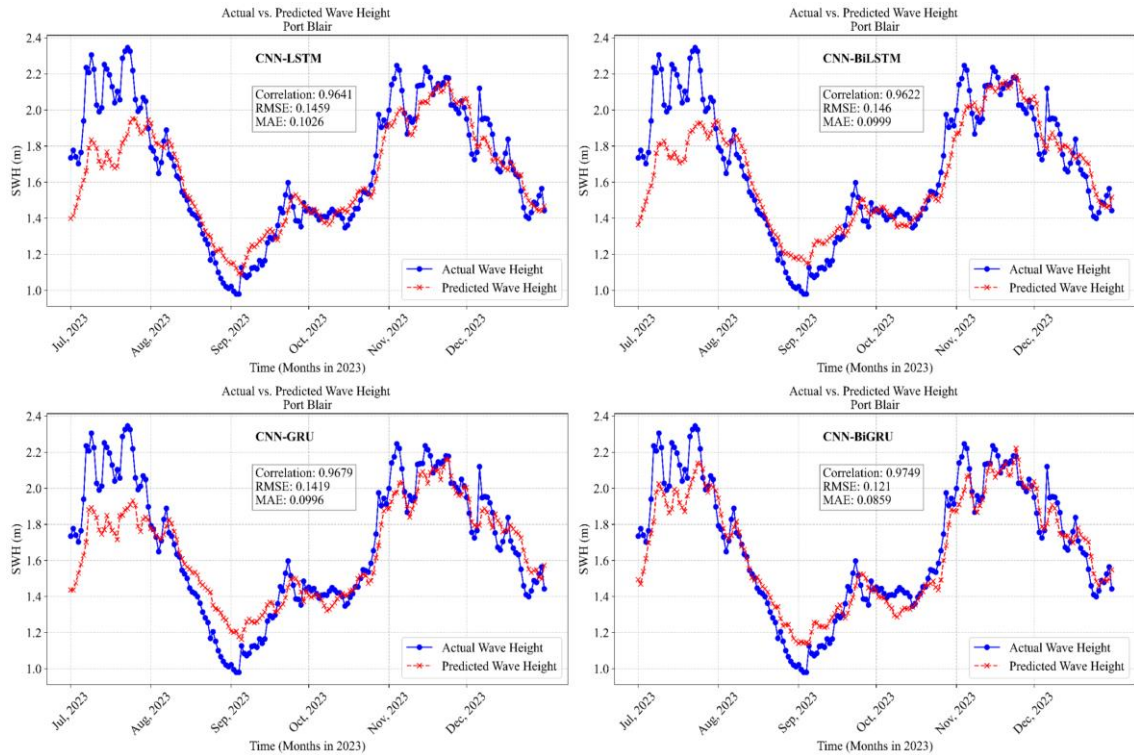
(a)



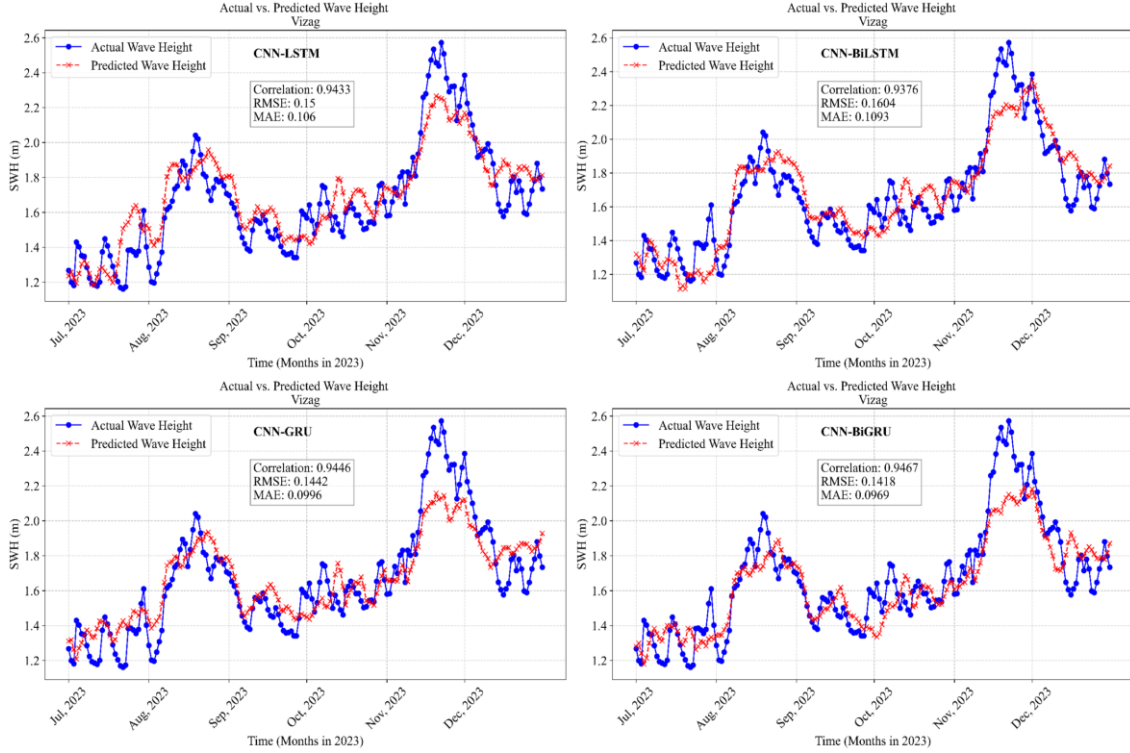
(b)



(c)



(d)



(e)

Figure 12. Analysis of actual and predicted wave heights at (a) P1, (b) Kyauk Phyu, (c) Dawei, (d) Port Blair, and (e) Vizag.

3.7 Performance evaluation of hybrid models for different time horizons

To examine the strengths and weaknesses of wave prediction models for different forecast steps, the performance of these models is evaluated by calculating statistical metrics for different time horizons: 3, 6, 9, 12, 15, 18, 21, and 24 hours, as shown in Figure 13(a-c). The result reveals that the CNN-BiLSTM and CNN-BiGRU models consistently perform better compared to the CNN-LSTM and CNN-GRU, respectively. This is due to the bidirectional nature of the LSTM and GRU layers, and this forward and backward processing structure can provide better prediction. Comparing the performance of all models reveals that the CNN-BiGRU has superior performance across all forecasting horizons. It achieves a correlation of 0.9872, an RMSE of 0.1547 and an MAE of 0.1005 at the 3-hour prediction and records a correlation of 0.9744, an RMSE of 0.2172 and an MAE of 0.1412 at the 24-hour horizon. While previous studies stated that CNN-LSTM is a better choice for short-term prediction [16], this study reveals the superior performance of CNN-BiGRU-BO and all models experience a decline in performance with longer horizons. The CNN-BiGRU model performs the best for wave prediction up to 24hour and all models' performance become nearly identical at the 24-hour forecast. It can be observed that the variation in models' performance with the prediction steps indicates that each model has particular strengths in capturing specific temporal patterns.

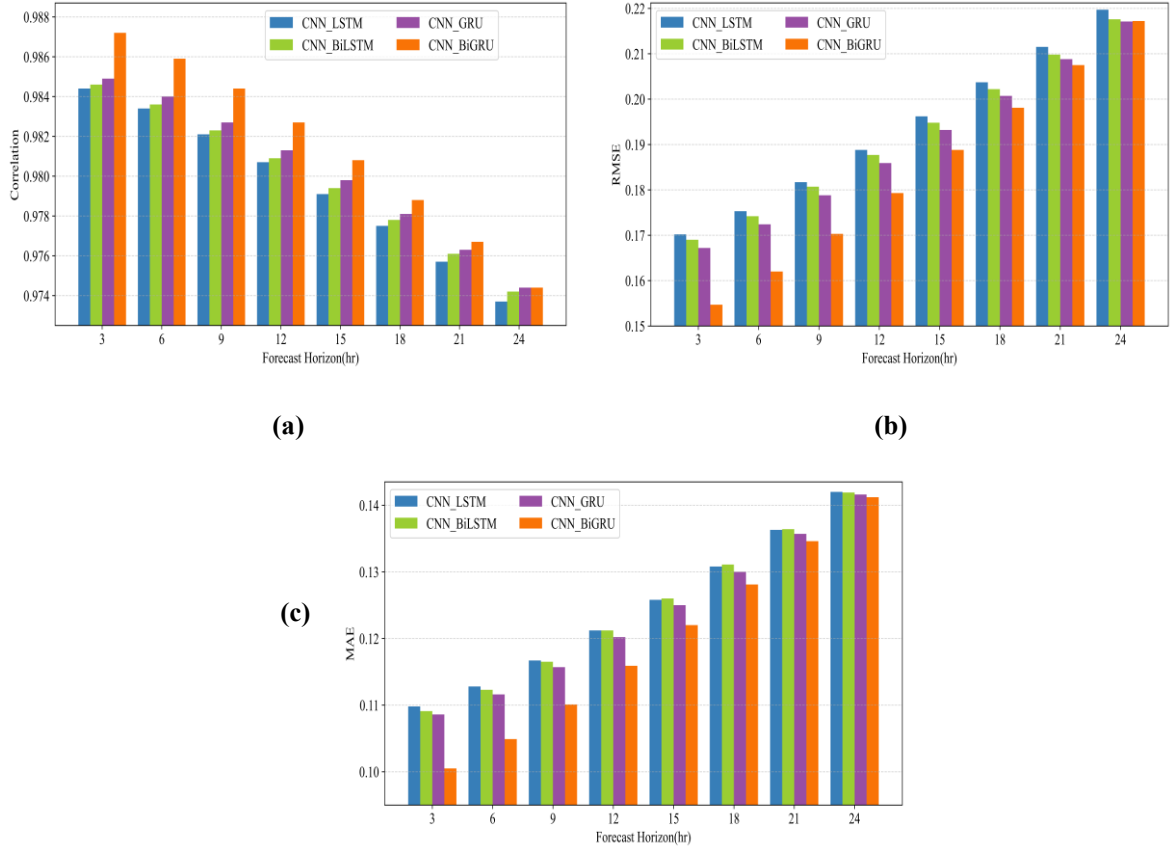


Figure 13. Statistical metrics of optimized models for different time horizons: (a) Correlation, (b) RMSE, and (c) MAE.

4. Conclusions

This study highlights the importance of incorporating multiple environmental factors to achieve high-performance regional wave prediction models in the BOB region. To address the challenges of regional wave prediction, this study employs hybrid deep learning models, including CNN-LSTM, CNN-BiLSTM, CNN-GRU, and CNN-BiGRU. The models are trained using 3-hourly data from January to December 2022, incorporating multiple features such as wind stress, water depth, pressure, SST, and SWH.

Optimization of models is carried out using the BO technique based on a validation set, ranging from January to June 2023, to improve the accuracy of the wave height predictions and develop wave prediction models. The results reveal that two layers of CNNs and RNNs with the BO technique offer better performance. Performance evaluation of the models is conducted using a test set covering the period from July to December 2023. It is observed that the multivariate model outperforms the univariate model as the prediction horizon increases. The analysis of environmental feature contributions using the IG method reveals that SST has the highest influence on the wave forecasting model, likely due to its role in modulating ocean surface conditions, while wind stress plays

the least role. This research enhances our understanding of how various environmental features impact regional wave predictions. Furthermore, the performance evaluation of all optimized models for different locations offers satisfactory results. The comparative analysis of the model's performance over different time horizons proves that CNN-BiGRU is the optimal model for wave predictions and emphasizes that models' effectiveness depends on forecasting horizons. This study offers important contributions to the development of effective and high-precision wave prediction systems, which can help enhance marine-related operations in the BOB region.

5. Limitations and future research

In this study, the validation of the ERA5 data against observed buoy data could not be performed due to the lack of local buoy data along the coast of Myanmar and the shortage of Indian buoy data with sufficient 3-hourly resolution. In addition, this study does not include direct benchmarking with physics-based models such as SWAN or WW3, because implementing this comparison would require substantial computational resources and is beyond the scope of this work. Only the IG method is applied in this study because, although SHAP(Shapley Additive exPlanations) offers strong interpretability for machine learning models, it becomes computationally expensive for larger datasets [39].

In the future, we aim to overcome the limitation of validation with measured data once they become available, and we plan to benchmark the proposed models against conventional physics-based models such as SWAN and WW3 to evaluate the performance improvements achieved. Moreover, a comparison of SHAP and the IG method will be conducted to enhance the interpretability of input features. We also intend to assess the models' performance under extreme conditions and analyze their predictive performance at extended forecast horizons of 48- and 72-hour. Additionally, expanding the research to other regions is suggested to evaluate the model's generalizability.

CRedit authorship contribution statement

Phyu Sin Thet: Conceptualization, Methodology, Model construction, Writing – original draft.

Aifeng Tao: Formal analysis, Methodology, Writing - original draft.

Tao Lv: Writing -Review & editing

Jinhai Zheng: Writing - review & editing.

Acknowledgement

This research was funded by the National Key R& D Program of China (No. 2023YFE0126300), the National Natural Science Foundation of China (No. 52271271), Postgraduate Research & Practise Innovation Program of Jiangsu Province (No. KYCX24_0877), and Major Science and Technology Projects of the Ministry of Water Resources (No. SKS-2022025)

References

- 473 1. Cornelius Stephanus Alfredo; Adytia, D.A. Time Series Forecasting of Significant Wave Height Using GRU, CNN-
474 GRU, and LSTM. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)* **2022**, 6, 776–781,
475 doi:10.29207/resti.v6i5.4160.
- 476 2. Jörges, C.; Berkenbrink, C.; Gottschalk, H.; Stumpe, B. Spatial Ocean Wave Height Prediction with CNN Mixed-Data
477 Deep Neural Networks Using Random Field Simulated Bathymetry. *Ocean Engineering* **2023**, 271,
478 doi:10.1016/j.oceaneng.2023.113699.
- 479 3. Shen, W.; Ying, Z.; Zhao, Y.; Wang, X. Significant Wave Height Prediction in Monsoon Regions Based on the VMD-
480 CNN-BiLSTM Model. *Front Mar Sci* **2024**, 11, doi:10.3389/fmars.2024.1503552.
- 481 4. Shuvo, S.D. Climatology of Frequency, Life Period, Energy and Speed for Tropical Disturbances and Cyclones over
482 the Bay of Bengal. *The Dhaka University Journal of Earth and Environmental Sciences* **2021**, 10, 23–31,
483 doi:10.3329/dujees.v10i1.56277.
- 484 5. Song, T.; Han, R.; Meng, F.; Wang, J.; Wei, W.; Peng, S. A Significant Wave Height Prediction Method Based on Deep
485 Learning Combining the Correlation between Wind and Wind Waves. *Front Mar Sci* **2022**, 9,
486 doi:10.3389/fmars.2022.983007.
- 487 6. Ding, T.; Wu, D.; Shen, L.; Liu, Q.; Zhang, X.; Li, Y. Prediction of Significant Wave Height Using a VMD-LSTM-
488 Rolling Model in the South Sea of China. *Front Mar Sci* **2024**, 11, doi:10.3389/fmars.2024.1382248.
- 489 7. Tang, G.; Du, H.; Hu, X.; Wang, Y.; Claramunt, C.; Men, S. An EMD-PSO-LSSVM Hybrid Model for Significant
490 Wave Height Prediction 2021.
- 491 8. Ji, Q.; Han, L.; Jiang, L.; Zhang, Y.; Xie, M.; Liu, Y. Short-Term Prediction of the Significant Wave Height and Average
492 Wave Period Based on VMD-TCN-LSTM Algorithm 2023.
- 493 9. Hao, P.; Li, S.; Yu, C.; Wu, G. A Prediction Model of Significant Wave Height in the South China Sea Based on
494 Attention Mechanism. *Front Mar Sci* **2022**, 9, doi:10.3389/fmars.2022.895212.
- 495 10. Han, L.; Ji, Q.; Jia, X.; Liu, Y.; Han, G.; Lin, X. Significant Wave Height Prediction in the South China Sea Based on
496 the ConvLSTM Algorithm. *J Mar Sci Eng* **2022**, 10, doi:10.3390/jmse10111683.
- 497 11. Domala, V.; Kim, T.W. A Univariate and Multivariate Machine Learning Approach for Prediction of Significant Wave
498 Height. In Proceedings of the Oceans Conference Record (IEEE); Institute of Electrical and Electronics Engineers Inc.,
499 2022; Vol. 2022-October.
- 500 12. Cornejo-Bueno, L.; Nieto Borge, J.C.; Alexandre, E.; Hessner, K.; Salcedo-Sanz, S. Accurate Estimation of Significant
501 Wave Height with Support Vector Regression Algorithms and Marine Radar Images. *Coastal Engineering* **2016**, 114,
502 233–243, doi:10.1016/j.coastaleng.2016.04.007.
- 503 13. Zhang, W.; Zhao, H.; Chen, G.; Yang, J. Assessing the Performance of SWAN Model for Wave Simulations in the Bay
504 of Bengal. *Ocean Engineering* **2023**, 285, doi:10.1016/j.oceaneng.2023.115295.
- 505 14. Anusree, A.; Kumar, V.S. Mean Wave Direction and Wave Height in the ERA5 Reanalysis Dataset: Comparison with
506 Measured Data in the Coastal Waters of India. *Dynamics of Atmospheres and Oceans* **2024**, 107,
507 doi:10.1016/j.dynatmoce.2024.101478.
- 508 15. Shewalkar, A.; nyavanandi, D.; Ludwig, S.A. Performance Evaluation of Deep Neural Networks Applied to Speech
509 Recognition: Rnn, LSTM and GRU. *Journal of Artificial Intelligence and Soft Computing Research* **2019**, 9, 235–245,
510 doi:10.2478/jaiscr-2019-0006.
- 511 16. Guan, X. Wave Height Prediction Based on CNN-LSTM. In Proceedings of the Proceedings - 2020 2nd International
512 Conference on Machine Learning, Big Data and Business Intelligence, MLBDBI 2020; Institute of Electrical and
513 Electronics Engineers Inc., October 1 2020; pp. 10–17.
- 514 17. Yao, Z.; Wang, Z.; Wang, D.; Wu, J.; Chen, L. An Ensemble CNN-LSTM and GRU Adaptive Weighting Model Based
515 Improved Sparrow Search Algorithm for Predicting Runoff Using Historical Meteorological and Runoff Data as Input.
516 *J Hydrol (Amst)* **2023**, 625, doi:10.1016/j.jhydrol.2023.129977.
- 517 18. Chung, J.; Gulcehre, C.; Cho, K.; Bengio, Y. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence

Modeling. **2014**.

19. Alizadeh, M.J.; Nourani, V. Multivariate GRU and LSTM Models for Wave Forecasting and Hindcasting in the Southern Caspian Sea. *Ocean Engineering* **2024**, *298*, doi:10.1016/j.oceaneng.2024.117193.
20. Martina Maria Pushpam, P.; Felix Enigo, V.S. Forecasting Significant Wave Height Using RNN-LSTM Models. In Proceedings of the Proceedings of the International Conference on Intelligent Computing and Control Systems, ICICCS 2020; Institute of Electrical and Electronics Engineers Inc., May 1 2020; pp. 1141–1146.
21. Yang, L.; Shami, A. On Hyperparameter Optimization of Machine Learning Algorithms: Theory and Practice. *Neurocomputing* **2020**, *415*, 295–316, doi:10.1016/j.neucom.2020.07.061.
22. Li, X.; Zhou, S.; Wang, F. A CNN-BiGRU Sea Level Height Prediction Model Combined with Bayesian Optimization Algorithm. *Ocean Engineering* **2025**, *315*, doi:10.1016/j.oceaneng.2024.119849.
23. Snoek, J.; Larochelle, H.; Adams, R.P. *Practical Bayesian Optimization of Machine Learning Algorithms*;
24. Shahriari, B.; Swersky, K.; Wang, Z.; Adams, R.P.; De Freitas, N. Taking the Human out of the Loop: A Review of Bayesian Optimization. *Proceedings of the IEEE* **2016**, *104*, 148–175.
25. Zhang, Q.; Hu, W.; Liu, Z.; Tan, J. TBM Performance Prediction with Bayesian Optimization and Automated Machine Learning. *Tunnelling and Underground Space Technology* **2020**, *103*, doi:10.1016/j.tust.2020.103493.
26. Alin, A. Multicollinearity. *Wiley Interdiscip Rev Comput Stat* **2010**, *2*, 370–374, doi:10.1002/wics.84.
27. Guan, L.; Yang, J.; Bell, J.M. Cross-Correlations between Weather Variables in Australia. *Build Environ* **2007**, *42*, 1054–1070, doi:10.1016/j.buildenv.2006.01.010.
28. Vu, D.H.; Muttaqi, K.M.; Agalgaonkar, A.P. A Variance Inflation Factor and Backward Elimination Based Robust Regression Model for Forecasting Monthly Electricity Demand Using Climatic Variables. *Appl Energy* **2015**, *140*, 385–394, doi:10.1016/j.apenergy.2014.12.011.
29. Sikdar, S.; Bhattacharya, P.; Heese, K. *Integrated Directional Gradients: Feature Interaction Attribution for Neural NLP Models*;
30. Kim, J.S. A Novel Approach for Brain Connectivity Using Recurrent Neural Networks and Integrated Gradients. *Comput Biol Med* **2025**, *184*, doi:10.1016/j.combiomed.2024.109404.
31. Salah, H.; Elbessa, M. Using Machine Learning Techniques to Predict Significant Wave Height Compared with Parametric Methods. *Engineering and Applied Sciences* **2024**, *9*, 106–128, doi:10.11648/j.eas.20240905.12.
32. Kara, A.B.; Wallcraft, A.J.; Metzger, E.J.; Hurlburt, H.E.; Fairall, C.W. Wind Stress Drag Coefficient over the Global Ocean. *J Clim* **2007**, *20*, 5856–5864, doi:10.1175/2007JCLI1825.1.
33. Bragone, F.; Morozovska, K.; Rosén, T.; Laneryd, T.; Söderberg, D.; Markidis, S. Automatic Learning Analysis of Flow-Induced Birefringence in Cellulose Nanofibrils. *J Comput Sci* **2025**, *85*, doi:10.1016/j.jocs.2025.102536.
34. Lee, S.; He, C.; Avestimehr, S. Achieving Small-Batch Accuracy with Large-Batch Scalability via Hessian-Aware Learning Rate Adjustment. *Neural Networks* **2023**, *158*, 1–14, doi:10.1016/j.neunet.2022.11.007.
35. Bengio, Y. *9 Practical Recommendations for Gradient-Based Training of Deep Architectures*;
36. Limouni, T.; Yaagoubi, R.; Bouziane, K.; Guissi, K.; Baali, E.H. Univariate and Multivariate LSTM Models for One Step and Multistep PV Power Forecasting. *International Journal of Renewable Energy Development* **2022**, *11*, 815–828, doi:10.14710/ijred.2022.43953.
37. Ahmed, A.A.M.; Jui, S.J.J.; AL-Musaylh, M.S.; Raj, N.; Saha, R.; Deo, R.C.; Saha, S.K. Hybrid Deep Learning Model for Wave Height Prediction in Australia's Wave Energy Region. *Appl Soft Comput* **2024**, *150*, doi:10.1016/j.asoc.2023.111003.
38. Li, Z.; Guo, F.; Zhang, X.; Guo, Y.; Zhang, Z. Analysis of Factors Influencing Significant Wave Height Retrieval and Performance Improvement in Spaceborne GNSS-R. *GPS Solutions* **2024**, *28*, doi:10.1007/s10291-023-01605-3.
39. Ranjbaran, G.; Recupero, D.R.; Roy, C.K.; Schneider, K.A. C-SHAP: A Hybrid Method for Fast and Efficient Interpretability. *Applied Sciences (Switzerland)* **2025**, *15*, doi:10.3390/app15020672.