

A Real-Time Spatiotemporal Machine Learning Framework for the Prediction of Nearshore Wave Conditions

JIAXIN CHEN¹,^a IAN G. C. ASHTON,^a EDWARD C. C. STEELE,^b AND AJIT C. PILLAI^a

^a *Renewable Energy Group, Department of Engineering, Faculty of Environment, Science, and Economy, University of Exeter, Penryn, United Kingdom*

^b *Met Office, Exeter, Devon, United Kingdom*

(Manuscript received 3 May 2022, in final form 28 October 2022)

ABSTRACT: The safe and successful operation of offshore infrastructure relies on a detailed awareness of ocean wave conditions. Ongoing growth in offshore wind energy is focused on very large-scale projects, deployed in ever more challenging environments. This inherently increases both cost and complexity and therefore the requirement for efficient operational planning. To support this, we propose a new machine learning framework for the short-term forecasting of ocean wave conditions to support critical decision-making associated with marine operations. Here, an attention-based long short-term memory (LSTM) neural network approach is used to learn the short-term temporal patterns from in situ observations. This is then integrated with an existing, low computational cost spatial nowcasting model to develop a complete framework for spatiotemporal forecasting. The framework addresses the challenge of filling gaps in the in situ observations and undertakes feature selection, with seasonal training datasets embedded. The full spatiotemporal forecasting system is demonstrated using a case study based on independent observation locations near the southwest coast of the United Kingdom. Results are validated against in situ data from two wave buoy locations within the domain and compared to operational physics-based wave forecasts from the Met Office (the United Kingdom's national weather service). For these two example locations, the spatiotemporal forecast is found to have an accuracy of $R^2 = 0.9083$ and 0.7409 in forecasting 1-h-ahead significant wave height and $R^2 = 0.8581$ and 0.6978 in 12-h-ahead forecasts, respectively. Importantly, this represents respectable levels of accuracy, comparable to traditional physics-based forecast products, but requires only a fraction of the computational resources.

SIGNIFICANCE STATEMENT: Spectral wave models, based on modeling the underlying physics and physical processes, are traditionally used to generate wave forecasts but require significant computational cost. In this study, we propose a machine learning forecasting framework developed using both in situ buoy observations and a surrogate regional numerical wave model. The proposed framework is validated against in situ measurements at two renewable energy sites and found to have very similar 12-h forecasting errors when benchmarked against the Met Office's physics-based forecasting model but requires far less computational power. The proposed framework is highly flexible and has the potential for offering a low-cost, low computational resource approach for the provision of short-term forecasts and can operate with other types of observations and other machine learning algorithms to improve the availability and accuracy of the prediction.

KEYWORDS: Oceanic waves; Buoy observations; Time series; Forecasting; Deep learning

1. Introduction

A growing international focus on the “blue economy” represents a changing industrial landscape for our seas, which is exemplified by the rapid growth of offshore wind throughout the world (Global Wind Energy Council 2020; Musial et al. 2021). Forecasting of ocean wave conditions is a fundamental requirement for the operation and maintenance of offshore and coastal infrastructure. As the industry increases, both the scale of projects and the complexity of the environments in which they operate, the accurate prediction of wave conditions can significantly reduce the cost, time, and technological requirements in relation to the operation and maintenance of offshore assets (James 1957; Johnston and Poole 2017; Ren et al. 2021). In the context of more extreme weather related

to climate change (Stott 2016) and sea level rise, offshore and coastal locations are likely to be increasingly affected by ocean waves (Moon 2005). As such, improving the accuracy with which ocean wave conditions can be estimated is acknowledged as an important component of risk mitigation and highlighted in offshore warranty standards (Ardente et al. 2008; Balog et al. 2016; DNVGL 2018; Gentry et al. 2017; Guşatu et al. 2021; Reikard et al. 2017).

Presently, most wave forecasts rely on computing time series estimates of spatial wave conditions using phase-averaging physics-based numerical models. A series of third-generation wave models such as the Wave Model (WAM; Günther et al. 1992; Komen et al. 1996; The Swamp Group 2013; WAMDI

Corresponding author: Jiaxin Chen, jc1083@exeter.ac.uk



This article is licensed under a [Creative Commons Attribution 4.0 license](http://creativecommons.org/licenses/by/4.0/) (<http://creativecommons.org/licenses/by/4.0/>).

DOI: 10.1175/AIES-D-22-0033.1 e220033

© 2023 American Meteorological Society. For information regarding reuse of this content and general copyright information, consult the [AMS Copyright Policy](http://www.ametsoc.org/PUBSReuseLicenses) (www.ametsoc.org/PUBSReuseLicenses).

Group 1988), WAVEWATCH III (WWIII; Tolman 2009; Tolman et al. 2002), and Simulating Waves Nearshore (SWAN; Booij et al. 1999; Ris et al. 1999) have become universal numerical approaches. These models mathematically represent the physical processes governing the generation, propagation, and dissipation of wind-generated waves: accurately providing wave data at a range of scales, from global models to higher-resolution (regional) studies and applied to characterizing the long-term wave climate and the production of real-time operational forecasts of wave conditions alike. However, due to the complexity of the calculations, these demand significant computational resources, inherently linked with both the level of detail at which physical processes are represented within the model as well as the associated temporal and spatial resolution considered. In an operational context, this constrains the viability of these approaches to only those organizations with sufficient resources to run them, with the incorporation of direct observational data assimilation schemes for the improvement of nowcast/very short-range forecast applications (e.g., Saulter et al. 2020) also potentially adding to this overhead.

In addition to the developments in numerical modeling systems, the availability of wave data from monitoring systems has also improved over time. In situ wave buoys provide high-accuracy, real-time data. Satellite-derived remote sensing offers reliable global coverage with increasing resolution (Boy et al. 2017; Perotto et al. 2020) and the potential for spatial imagery (Bondur et al. 2016). Conversely, observations of wave conditions obtained from buoys (Centurioni 2018) and in situ or remote-sensing devices (Bailey et al. 2019) only provide data at limited locations within a region. Due to the high cost associated with their deployment and maintenance, it is prohibitively expensive to deploy wave observation networks throughout a region. Similarly, satellite remote sensing suffers from poor temporal resolution, which particularly limits suitability for short-term forecasting.

In recent years, machine learning (ML) methods and neural networks have been widely employed to forecast environmental variables given limited data inputs (Schultz et al. 2021). In contrast to traditional approaches, these rely on modeling the empirical relationship between inputs and outputs, rather than any physical relationship. For the case of ocean waves, this is an area of emerging interest with research being conducted into the capabilities of data-driven wave forecasting that combines observations with numerical model data to derive reliable, accurate, and low-cost wave data with both high temporal and spatial resolution.

Generally, these methods can be classified into three categories:

- 1) data-driven model based on observations only (Desouky and Abdelkhalik 2019; Pirhooshayan et al. 2020; Sadeghifar et al. 2017);
- 2) surrogate numerical models (James et al. 2018; O'Donncha et al. 2018; Oh and Suh 2018); and
- 3) hybrid models considering both numerical modeling data and observations (Serras et al. 2019).

However, despite the significant opportunity for empirical forecasts and spatial correlation methods to provide a step change in marine data provision, the specific methods to

achieve this potential have not been recognized. The present paper proposes and benchmarks a new ML framework to combine a spatial model with a temporal forecasting model to provide real-time, short-term forecasts covering the domain of a previously computed hindcast model. The work builds directly on a recent ML surrogate (spatial nowcasting) model trained on an available spectral wave model that has been previously described and validated in Chen et al. (2021). In that work, wave observations were combined with numerical modeling data to provide a low computational cost, high-accuracy nowcast of spatially distributed wave parameters. In this paper, we focus on the temporal model development and the application of the combined spatiotemporal model to demonstrate the complete framework for the prediction of the wave field in the southwest of the United Kingdom. This is validated against independent buoy observations at two potential renewable energy sites to test the accuracy and stability of the spatial forecasting model. The outputs of the proposed ML framework are also compared against an independent traditional physics-based forecast output from the Met Office (UKMO).

This paper is organized as follows: section 2 introduces the components of the proposed model framework, including the basic principles of LSTM networks, the approach used for temporal forecasting at a fixed location, and the integration with the (existing) spatial extrapolation method. Section 3 introduces a case study experiment run with the complete framework. This compares different model configurations for temporal forecasting, including the presentation of the data preprocessing and model configuration implemented. It also presents the results of the experiments, including validation and benchmarking of the proposed model framework. The results are discussed in detail in section 4, and the final conclusions of the work are presented in section 5. A list of terms and definitions used in this paper can be found in appendix A.

2. Design and methodology of the proposed modeling framework

a. Problem formulation and framework overview

The complete framework outlined in this paper is made up of four principal steps that are outlined in Fig. 1:

- 1) data preprocessing of the in situ buoy data;
- 2) a temporal forecasting model used to forecast the conditions at these buoy locations;
- 3) a spatial nowcasting model driven by the conditions at these discrete locations (Chen et al. 2021); and
- 4) data postprocessing and visualization.

By using the output of the temporal forecasting model to drive the spatial nowcasting model, the proposed model framework describes a spatiotemporal modeling approach providing a short-term forecast across the entire model domain.

b. Data preprocessing

1) GAP FILLING

Initial preprocessing is used to prepare the in situ datasets for the temporal model. The proposed methodology uses

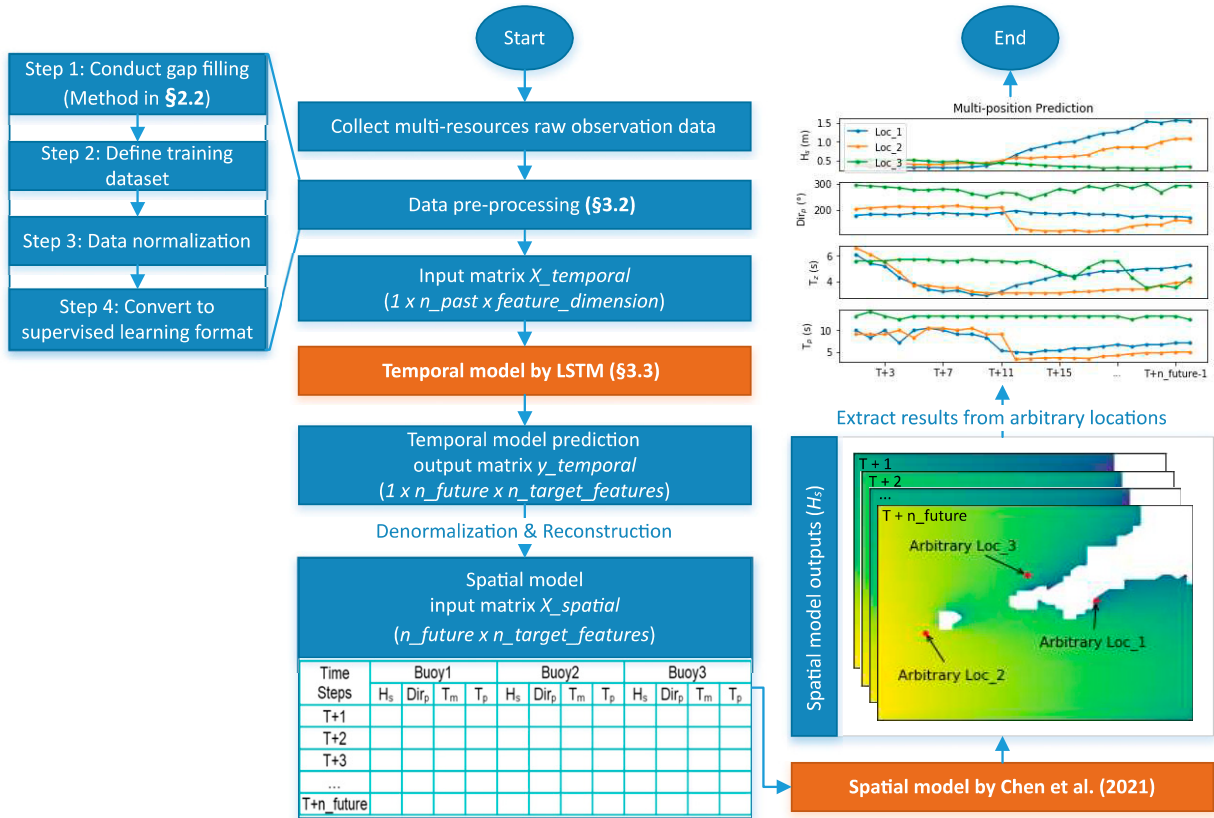


FIG. 1. Spatiotemporal wave prediction model flowchart. Orange boxes highlight the temporal and spatial model that are the subject of this paper.

three wave buoys as input, as described in section 3, and the training procedure for the neural networks therefore requires concurrent data for all input buoys. Ignoring discontinuities in any datasets will limit the quantity of data and potentially, if malfunctions occur during similar environmental conditions, skew the statistics of the observations. At the same time, improper methods for filling these gaps would not maintain the time dependency of the series.

In this study, data were preprocessed to fill gaps in the time series buoy data considering all the variables measured. This is important for the model framework, both when training the temporal model and when running the model for real-time predictions.

In this model framework, a low-rank tensor completion model with a truncated nuclear norm (LRTC-TNN) proposed by Chen et al. (2020) is used for gap filling the missing records in the wave observation data. This model includes the following five steps:

- 1) combining the observations from different buoys together as a data matrix, utilizing the same time index as rows and features as columns, and filling the blanks and replacing invalid values with zero;
- 2) adapting the observation matrix to a tensor structure with three dimensions—parameters \times number of days in total \times time steps over one day;

- 3) manually “masking” missing entries with certain missing data type to replace the original observations in the training and validation dataset;
- 4) training incomplete tensor data using the LRTC-TNN model; and
- 5) filling in actual missing entries with imputed values.

After training, a new augmented dataset containing the same number of samples is generated with no missing values.

2) DEFINING TRAINING DATASETS

Experimental datasets are commonly split by retaining the first 80% for training validation and the last 20% for testing (Du et al. 2020). However, as wave parameters are known to exhibit seasonal behavior and interannual trends, it is not always appropriate to simply rely on the observed temporal pattern to predict the future behavior. Therefore, to generalize the temporal forecasting model, the seasonal trends must be considered when selecting the training data. For wave data, wave conditions in each month will show a similar pattern across different years and will be highly correlated both with the preceding and succeeding months. To account for this and capture the seasonality, test datasets can be constructed for each month, using a training dataset made up of the month itself, the preceding month, and the subsequent month (as shown in Table 1).

TABLE 1. Temporal model input/output data.

Model	Training and validation (2010–19)		Test (2020)	
	Input: selected input features	Output: target feature	Input: selected input features	Output: target feature
Jan	Dec, Jan, Feb	Dec, Jan, Feb	Jan	Jan
Feb	Jan, Feb, Mar	Jan, Feb, Mar	Feb	Feb
Mar	Feb, Mar, Apr	Feb, Mar, Apr	Mar	Mar
Apr	Mar, Apr, May	Mar, Apr, May	Apr	Apr
May	Apr, May, Jun	Apr, May, Jun	May	May
Jun	May, Jun, Jul	May, Jun, Jul	Jun	Jun
Jul	Jun, Jul, Aug	Jun, Jul, Aug	Jul	Jul
Aug	Jul, Aug, Sep	Jul, Aug, Sep	Aug	Aug
Sep	Aug, Sep, Oct	Aug, Sep, Oct	Sep	Sep
Oct	Sep, Oct, Nov	Sep, Oct, Nov	Oct	Oct
Nov	Oct, Nov, Dec	Oct, Nov, Dec	Nov	Nov
Dec	Nov, Dec, Jan	Nov, Dec, Jan	Dec	Dec

3) DATA NORMALIZATION

In multivariate problems, the scale and distribution of the data may be different for each of the variables. For deep neural network (DNN) models, large input values can result in a model that learns large weight values, which may then make the model unstable and suffer from poor performance and sensitivity to input values. LSTM models, specifically, may experience gradient explosion during training, causing the learning process to fail by repeatedly multiplying gradients through the network layers if the values are larger than 1.0. Linear rescaling of the input (feature) values before they are presented to a network can help avoid this problem (Bishop 1995). Therefore, in the proposed model framework, all features in the dataset are normalized to the interval $[0, 1]$, using a minimum–maximum normalization process:

$$x'_t = \frac{x_t - \min(x)}{\max(x) - \min(x)}, \quad (1)$$

where x'_t is the normalized feature value at time step t , x_t is the measured feature value at time step t , and $\min(x)$ and $\max(x)$ are the minimum and maximum feature value, respectively, within the target periods of the dataset. This normalization is performed to prevent high-magnitude variables (such as wave direction) from overwhelming variables of lower magnitudes (such as significant wave height) during cell activation.

When used in the forecasting model, output predictions are also normalized between 0 and 1. The outputs of the networks must therefore be postprocessed with the linear rescaling to transform back to absolute values by

$$o_t = \{o'_t \times [\max(x) - \min(x)]\} + \min(x), \quad (2)$$

where o'_t represents the normalized model output at time step t , and o_t refers to the prediction output in the absolute scale.

4) TRAINING DATASET TRUNCATION

The temporal model uses a multistep lookback, multistep forecast, where the previous n_b steps (from $t - n_b + 1$ to t) are considered as the input, and the subsequent n_f steps (from

$t + 1$ to $t + n_f$) are considered as the output. Pirhooshyaran et al. (2020) had previously found that a lookback period twice the length of the forecast horizon is optimal with respect to the forecast accuracy. Therefore, in the present implementation, forecasts are generated up to 12 h ahead at a half-hourly interval (i.e., 24 time steps; $n_f = 24$), considering the previous 24 h (i.e., 48 time steps; $n_b = 48$) of observations as the input to the model. Therefore, we truncate the sequences into smaller slices using a sliding window with $n_b = 48$ steps and an output window with $n_f = 24$ steps, as illustrated in Fig. 2. To generalize the model for different output features, both the training input and output matrices are prepared to include the set of all feature dimensions D_{Total} , i.e., the input array has the dimension of $(n_s \times n_b \times D_{\text{Total}})$ and the output array has the dimension of $(n_s \times n_f \times D_{\text{Total}})$ in which n_s means the number of samples considered. Before training, feature selection is undertaken on the input array to reduce the input dimensions from D_{Total} to D_i , and the target output feature will be selected from D_{Total} to have output dimension D_o .

5) FEATURE SELECTION

Although some domain knowledge is required to select the most important input variables, these data are still often characterized by either insufficient information or providing an excess that can result in overfitting. Using the best candidate features for prediction and classification of environmental parameters can save training time and yield better prediction results (Effrosynidis and Arampatzis 2021). Within renewable energy forecasting applications, it has been shown that a reduced number of features can improve the prediction accuracy by 5%–40% (Salcedo-Sanz et al. 2018).

Effrosynidis and Arampatzis (2021) compared different feature selection methods in environmental models and found that so-called “embedded methods” and “ensemble methods” are better options with respect to performance, stability, and computational time. Pirhooshyaran and Snyder (2020) used an embedded elastic net algorithm for the feature selection when forecasting ocean waves along the East Coast of the United States. Their work highlighted the benefit of using the elastic net. Furthermore, Meyer et al. (2018) emphasized

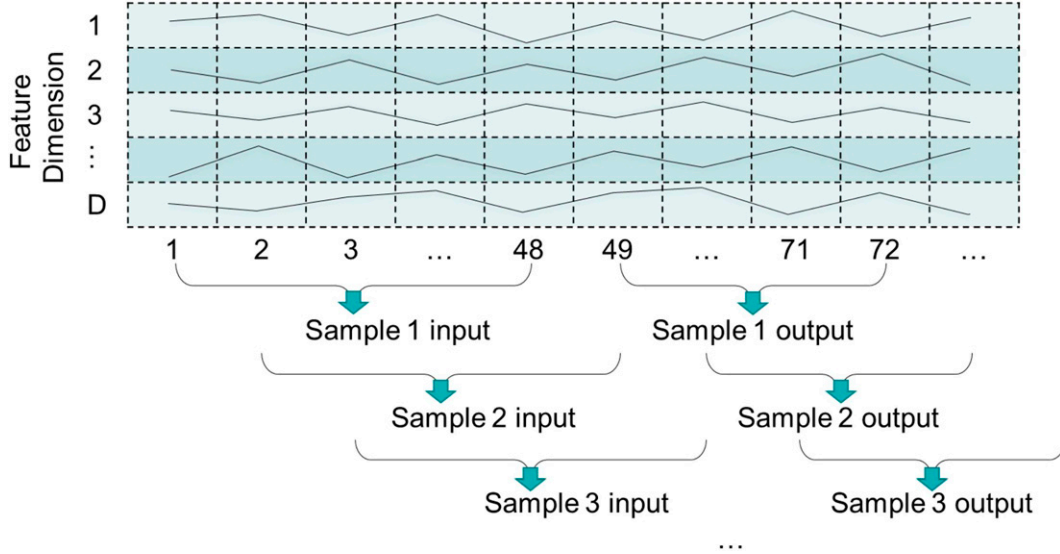


FIG. 2. Data truncation. The lookback (input) time steps n_b (48) and forecast horizon (output) time steps n_f (24).

the importance of cross validation (CV) and that feature selection in conjunction with target-oriented CV can reduce overfitting. In the present work, we have therefore implemented the elastic net algorithm in conjunction with CV to select the best features for training the temporal model. The elastic net (Zou and Hastie 2005) regression loss function includes two regularizers, ℓ_1 norm and ℓ_2 norm, which correspond to the penalties of the least absolute shrinkage and selection operator (LASSO) method (Tibshirani 1996) and the Ridge method (Hoerl and Kennard 1970). The elastic net algorithm loss function is shown in Eq. (3):

$$f_{\omega}(x) = \frac{1}{n} \sum_{i=1}^n [y^i - N(x^i; \omega)]^2 + \lambda_2 \omega_2^2 + \lambda_1 \omega_1, \quad (3)$$

where the ℓ_1 norm is $\omega_1 = \sum_{j=1}^p |\omega_j|$, the ℓ_2 norm is $\omega_2^2 = \sum_{j=1}^p |\omega_j|^2$, and λ_1 and λ_2 are their ratios, respectively. The target of this feature selection is to find the most relevant features for the target feature prediction. For normal cases, the longer the target forecast, the less accurate we expect this forecast to be. Therefore, as the target forecast time step in this paper is n_f , this feature selection is designed to achieve the best regression performance from the input features at step t to the target output features at step $t + n_f$.

c. Temporal model using LSTM neural networks

In recent years, multivariate time series forecasting models have attracted significant research interest: the main objective being the analysis of multidimensional time series data from past observations to develop a statistical model that can capture the basic pattern of the relationships to enable the prediction of future values. DNNs (LeCun et al. 2015) have been frequently studied and practically deployed to solve highly complex problems. A special type of DNN, known as a recurrent neural network (RNN), is designed for sequence-dependent modeling that can learn and retain information over long time periods and tackle arbitrarily sized inputs.

In this study, a variation of the canonical RNN, referred to as long short-term memory (Hochreiter and Schmidhuber 1997) is used to deal with the long sequence data forecast problem (Greff et al. 2017). The LSTM introduces a “gate” concept that can record and transmit information across multiple time steps. This enables processing of the sequence relationship of the time series data and makes both the network predictions and the characteristics of input data highly correlated temporally. The standard LSTM equations and description are provided in appendix B. Different architectures of LSTM networks are implemented including the following:

- 1) a simple “vanilla” single hidden layer LSTM;
- 2) an encoder–decoder LSTM; and
- 3) an attention-based encoder–decoder LSTM.

The above LSTM architectures are implemented and compared to identify the best performing architecture for implementation in the temporal component of the model framework.

1) VANILLA LSTM MODEL

In the proposed model framework, we apply a simple, vanilla LSTM model composed of a single LSTM layer and a single full connection (FC) layer as the benchmark LSTM structure for the forecasts (Fig. 3). The output of this vanilla LSTM model is not a sequence but rather a one-dimensional vector wherein each time step corresponds to an individual value in the vector. The vanilla LSTM can be extended to a model with multiple LSTM layers, which are connected by returning sequences from the previous LSTM layers to the forward LSTM layers via a so-called stacked LSTM. Increasing the complexity in this manner can also make use of more fully connected layers or higher dimensionality of the LSTM output space to increase the depth and complexity of the neural networks (Hochreiter and Schmidhuber 1997).

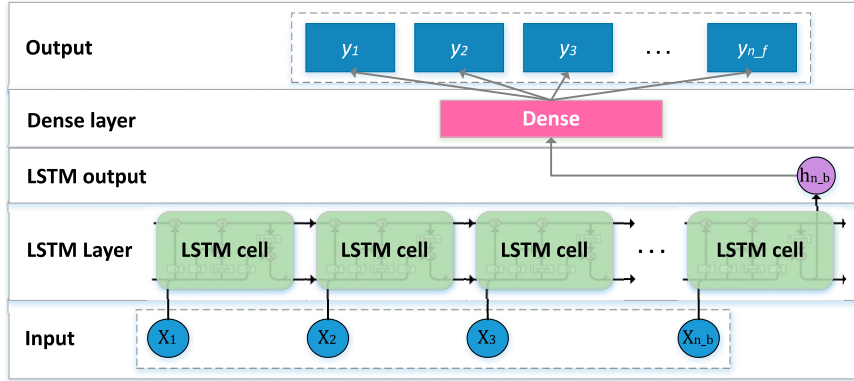


FIG. 3. Vanilla LSTM architecture. In this work the vanilla LSTM is a two-layer network composed of an LSTM layer as the first layer and a full connection as the second. For training and testing the model, input \mathbf{X}_{Te} is a three-dimensional array ($n_s \times n_b \times D_i$). The input \mathbf{X}_{Te} is supplied to a single LSTM layer; the output vector of this LSTM layer $\mathbf{h}_{n,b}$ then goes through a single fully connected layer to output \mathbf{y}_{Te} , a two-dimensional matrix ($n_s \times n_f$).

2) ENCODER–DECODER LSTM MODEL

The main application considered in this paper is a sequence-to-sequence time series model to forecast results over multiple future time steps, relying on a knowledge of the past conditions over multiple past time steps. An encoder–decoder model, a common type of sequence-to-sequence model, is an extension of an LSTM model that aims to transform an input sequence to an output sequence, both with arbitrary lengths. It was originally conceived in the field of language modeling but has now found wider applications (Sutskever et al. 2014). For a given data series, an encoder–decoder LSTM is configured to read the input sequence, encode it, decode it, and transfer it to further layers. The performance of the whole model is evaluated based on the model’s ability to regress the output sequence.

The first element of an encoder–decoder LSTM, the encoder, is usually a single LSTM or a chain of LSTM cells that read (i.e., encode) the input of an arbitrary length and map this to a fixed-sized output, while the second element, the decoder, receives the output vector from encoder and maps (i.e.,

decodes) it to a target sequence with the same length as the forecast prediction. Compared to the vanilla LSTM model, which generates two-dimensional outputs in which each time step corresponds to an individual vector, the encoder–decoder LSTM model output is a three-dimensional matrix that also takes the output feature dimension into consideration (see Fig. 4). In our implementation, a time-distributed dense layer between the encoder–decoder layer and this output is added, further increasing the complexity and potentially increasing the accuracy of the network.

3) ATTENTION ENCODER–DECODER LSTM

A limitation of this simple encoder–decoder model is that only the last state of the encoder is compressed into a fixed-length vector and used as an input to the decoder. Therefore, when the sequence is very long, the encoder would have much weaker memory of earlier time steps, and the prediction ability would gradually decrease as the forecast horizon increases (Du et al. 2020). The attention mechanism solves this problem

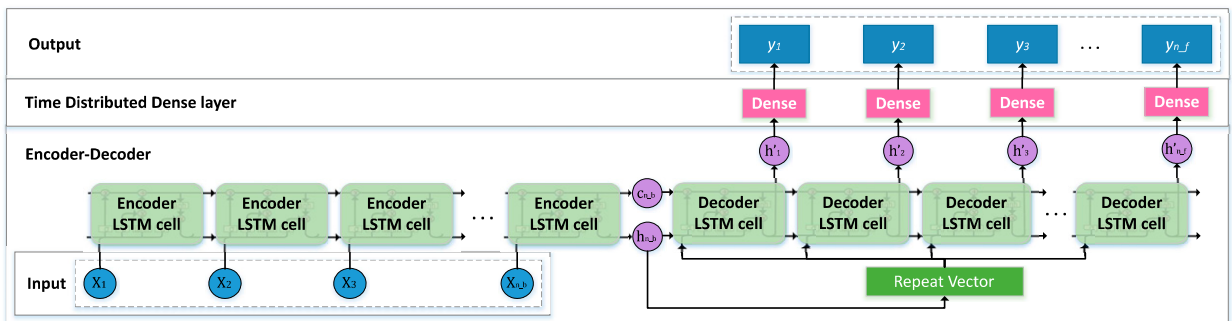


FIG. 4. Encoder–decoder sequence-to-sequence network. This is implemented as a two-layer network with an encoder–decoder layer followed by a dense layer. For training and testing the model, the input \mathbf{X}_{Te} ($n_s \times n_b \times D_i$) is input to the encoder LSTM layer similar to the vanilla LSTM model. The hidden state and cell state at the last time step are treated as the initial state for the decoder layer, and the last hidden state of encoder layer is repeated forecast time steps n_f times, as inputs to the decoder LSTM layer. The hidden states of decoder LSTM at all n_f steps are then transferred to individual fully connected dense layers, called the time-distributed dense layers, to provide the outputs. The output \mathbf{y}_{Te} is a three-dimensional array ($n_s \times n_f \times D_o$).

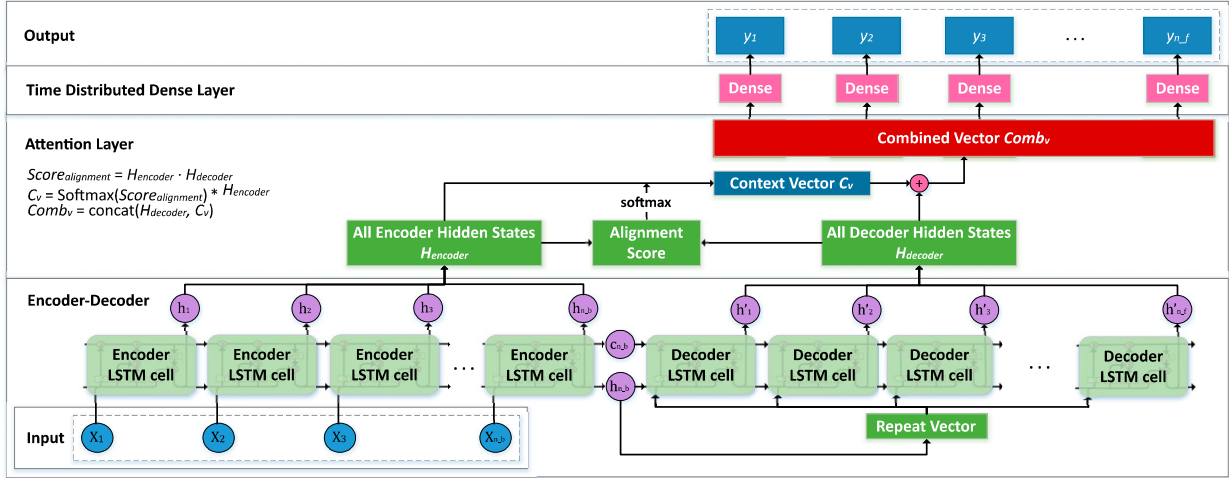


FIG. 5. Luong attention-based LSTM structure. The encoder–decoder and time-distributed dense layers in this architecture are similar to those previously described (Fig. 4). An attention layer is added following the encoder–decoder layer, which takes the hidden states at all time steps in both the encoder and decoder LSTM cells into consideration and generates a combined vector **Comb_v**, through calculating the alignment score and then vector concatenation. The **Comb_v** is then transferred to the time-distributed dense layer to get the outputs.

by helping a neural network identify which parts of the input are more correlated with the target elements in a prediction task (Fig. 5). The mechanism (commonly referred to as “additive attention”) was initially proposed by Bahadori et al. (2014) for sequence-to-sequence machine translation models and then extended by Luong et al. (2015; commonly referred to as “multiplicative attention”). The multiplicative attention includes three approaches to calculate an alignment score in which the dot product alignment score function was observed to perform well for the global attention. Since the inputs of the Luong et al. (2015) algorithm are single-variate sequences of sentences, this attention mechanism is basically a temporal attention.

Compared to the encoder–decoder model, which only outputs the hidden state of the last time step from the encoder, the attention model requires access to the outputs from both the encoder and decoder at each input time step. All encoder hidden states $H_{encoder}$ and all decoder hidden states $H_{decoder}$ can be regarded as the probability of the target sequence, computed as a conditional distribution [given in Eqs. (4) and (5)]. Then, the alignment score S_a is calculated by the dot product [Eq. (6)] of $H_{encoder}$ and $H_{decoder}$ and is normalized by a softmax function before generating the context vector C_v [Eq. (7)]. After that a combined vector **Comb_v** is generated by concatenating $H_{decoder}$ and C_v . This combined vector is then input to the time-distributed dense layer to get the target outputs [Eq. (8)]:

$$H_{encoder} = p(h_1, h_2, \dots, h_{n_b} | X_1, X_2, \dots, X_{n_b}), \quad (4)$$

$$H_{decoder} = p(h'_1, h'_2, \dots, h'_{n_f} | X_1, X_2, \dots, X_{n_b}) \\ = p(c_{n_b}, h_{n_b} | X_1, X_2, \dots, X_{n_b}) \cdot p(h'_1, h'_2, \dots, h'_{n_f} | c_{n_b}, h_{n_b}), \quad (5)$$

$$S_a = H_{encoder} \cdot H_{decoder}, \quad (6)$$

$$C_v = \text{softmax}(\text{score}_{\text{alignment}}) \times H_{encoder}, \quad (7)$$

$$\mathbf{Comb}_v = \text{concatenate}(H_{decoder}, C_v), \quad (8)$$

where $(X_1, X_2, \dots, X_{n_b})$ denotes the input observation array for lookback time steps n_b , and $(h_1, h_2, \dots, h_{n_b})$ and $(h'_1, h'_2, \dots, h'_{n_f})$ are the output hidden state of encoder LSTM cells and decoder LSTM cells, respectively, in which n_b can differ from the forecast time steps n_f .

d. Integration with the spatial model

The temporal forecasting problem at discrete point locations can be regarded as a sequence-to-sequence wave prediction problem. In this study, we apply a multi-input and multioutput strategy, which generates multistep outputs in a single integrated model without requiring a recursive process. This output y_{Te} is therefore a multidimensional vector (rather than single values), which contains the wave parameter(s) of interest up to 12 h ahead. For each time sample t , the temporal model results are the forecast values of four wave parameters H_s , Dir , T_m , and T_p at the three buoy locations. The term y_{Te} is therefore a time series with dimensions of 24 forecast steps \times 12 features.

Executing the spatial nowcasting model using the temporal forecasts at the input locations provides outputs as a series of parameter distributions maps at the different time steps. As a result, the forecasts of the wave climate at arbitrary positions within the domain can be extracted.

3. Case study: Model development

To demonstrate the proposed ML framework, a case study in the water off the southwest coast of the United Kingdom,

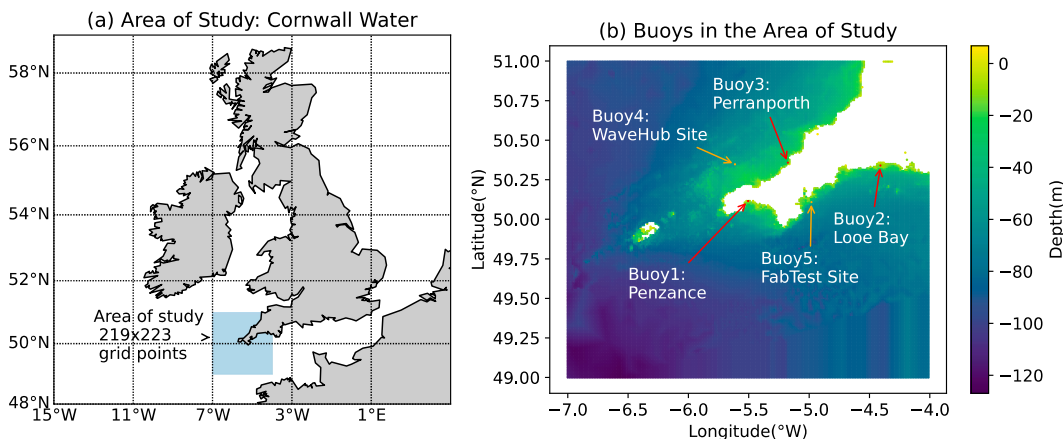


FIG. 6. (a) The case study considers the waters around Cornwall in the United Kingdom ranging from 4° to 7° W in longitude and from 49° to 51° N in latitude. (b) Five wave buoys are used in this study in which the three red points (Penzance, Looe Bay, and Perranporth) represent buoys used as inputs for the spatiotemporal model, and the two orange points (WaveHub and FabTest) represent the buoys used for validating the model outputs and benchmarking the proposed model framework.

surrounding Cornwall, is considered. The western boundary of the model domain faces the North Atlantic, which allows significant fetch into this active, highly seasonal shelf sea. As such, the wave climate is a mix of locally generated wind wave and incoming swell, which is dominated by storms that form in the North Atlantic and pass through the region.

This section describes an implementation of the proposed model framework and details how the specific spatiotemporal model for this region was developed using the techniques described in section 2 and combined with the spatial wave nowcasting model described in section 3a(2) and by Chen et al. (2021). This makes use of data from in situ wave measurement buoys [section 3a(1)] and a regional numerical hindcast wave model from 1989 to 2009 [section 3a(2)] that were also used in the present case study.

a. Data description

1) IN SITU BUOY OBSERVATIONS

Along the southwest coast of the United Kingdom, in situ wave measurements are routinely collected from Datawell Directional Wave Rider Mk III buoys, operated as part of regional coastal monitoring programs collated by the Channel Coastal Observatory (Channel Coastal Observatory 2021). Here, data from three buoys within the region were used, as deployed at Looe Bay, Penzance, and Perranporth (Fig. 6).

The Perranporth buoy was deployed in December 2006, while the Penzance and Looe Bay buoys only started collecting data in April 2007 and June 2009, respectively. From these, an 11-yr dataset from 2010 to 2020 was extracted during which data from all three buoys is available. Each dataset provides observations of seven parameters: significant wave height H_s , maximum observed wave height H_{max} , spectrally derived zero-crossing wave period T_z , peak wave period T_p , peak wave direction Dir_p , wave directional spread Dir_{sp} , and sea surface temperature (SST) at a frequency of 30 min. To be more specific, the T_p in the buoy observation is the inverse

of the frequency at which the wave energy spectrum reaches its maximum, and the Dir_p is the direction corresponding to the T_p .

None of the wave buoys used in this study were assimilated by either of the two physics-based numerical models described in the following sections.

2) SPATIAL NOWCASTING MODEL

The specific spatial nowcasting model used is that developed by the authors and previously described by Chen et al. (2021). The model has been trained using the previously validated outputs from a regional nearshore wave hindcast model (SWAN) that covers the period 1989–2009, as described in van Nieuwkoop et al. (2013), to define the correlation between the discrete buoy locations used in this paper and the rest of the model domain. The SWAN model spatially ranges from 4° to 7° W and from 49° to 51° N (see Fig. 6a) with a grid resolution of $1\text{ km} \times 1\text{ km}$, which results in 219×223 grid points in the model domain. The wave and wind input for the regional SWAN model was provided by the European Centre for Medium-Range Weather Forecasts (ECMWF) interim reanalysis (ERA-Interim) dataset with a grid resolution of $1.5^{\circ} \times 1.5^{\circ}$, the global wave component of which utilizes the WAM model. The SWAN model results have been validated by measurement data including the four buoys used in our proposed model with the relative bias of H_s and wave energy period $T_{m-1.0}$ ranging 1%–15% and 7%–20%, respectively, and the absolute difference of mean direction ranging 5° – 23° (van Nieuwkoop et al. 2013). To be consistent with the observations, H_s , T_z , T_p , and wave direction can be extracted from the SWAN model outputs. This wave direction available from the SWAN model is a mean wave direction Dir_m calculated in degrees over all frequencies and directions of the two-dimensional wave spectrum. In this paper, we regarded the Dir_p (as provided by the buoys) and Dir_m (from the SWAN

TABLE 2. Overview of data used in the present study.

Name	Lon (°W)	Lat (°N)	Distance to nearest SWAN grid point (km)	Water depth (m)	Observation availability	Present framework outputs	Met Office forecast outputs	Comparison period
Penzance	5.5032	50.114	0.2835	8.84	2010–20, 0.5-h interval	—	—	—
Looe Bay	4.4110	50.339	0.3133	10.32	2010–20, 0.5-h interval	—	—	—
Perranporth	5.1756	50.353	0.5330	19.97	2010–20, 0.5-h interval	—	—	—
WaveHub	5.6143	50.347	0.2334	35.85	2016, 0.5-h interval	2016; issued every 0.5 h, 12-h horizon	2016; issued every 6 h, 2.5-day horizon	2016; 0-, 1-, 6-, and 12-h horizon
FabTest	4.9807	50.109	0.3906	29.94	2020, 0.5-h interval	2020; issued every 0.5 h, 12-h horizon	2020; issued every 6 h, 2.5-day horizon	2020; 0-, 1-, 6-, and 12-h horizon

model) as equivalent in representing the wave direction, although this will introduce some discrepancy.

3) PHYSICS-BASED FORECAST

Using the proposed model framework, forecasts can be obtained at any arbitrary location within the model domain. However, for the present work, we focus on the results for WaveHub and FabTest sites, two locations where buoys were also deployed (see Fig. 6). Being independent of those used for the training, these allow the framework to be validated with in situ observations and compared with traditional physics-based wave forecasts from the UKMO for the same site, to contextualize the performance of the proposed model framework with existing forecast data.

In contrast to provenance of the wave hindcast (SWAN; van Nieuwkoop et al. 2013) used for the training of the spatial nowcasting model, the UKMO regional wave forecast is an instance of the WAVEWATCH III model, whose domain covers the seas on the northwest European continental shelf, forced by 10-m winds from the UKMO global atmospheric Unified Model (Walters et al. 2011), with lateral wave boundary conditions and surface current inputs from the UKMO global wave forecast (Saulter et al. 2016) and UKMO Atlantic Margin Model ocean physics forecast (Tonani et al. 2019), respectively. The UKMO operational model is under constant development and improvement, and therefore the archived forecasts used are the model outputs at the time that they were originally run and issued. The results for 2016, therefore, represent the UKMO forecast model configuration in 2016, while the results from 2020 represent the UKMO forecast model configuration in 2020. Between 2016 and 2020, the system was substantially improved, most notably moving from a regular 4-km grid to a two-tiered spherical multicell grid (where the open water cells are resolved at ~ 3 km and the coastal cells with a water depth less than 40 m are resolved at ~ 1.5 km), and

accompanied by an improvement to the source term physics scheme (that represents wave growth and dissipation processes in the model). This model is run on eight nodes of the UKMO Cray XC40 supercomputer, where each node comprises 36 Intel Haswell CPUs (at 2.1 GHz) and 128-GB RAM, corresponding to an execution time of ~ 5 min for a 12-h forecast.

In the present work, the proposed model framework produces forecasts (of 30-min output frequency) up to 12 h ahead, issued every 30 min, that can assimilate wave observations, while the UKMO operational wave model produces forecasts (of hourly output frequency) up to 2.5 days ahead, issued every 6 h. However, for the purpose of this comparison, only data that were issued at the same point in time with the same forecast horizon are used. The proposed model framework therefore has access to more recent in situ data than the UKMO forecast. The comparison was therefore made every 6 h to align the physics-based and ML-based models (Table 2). In addition, the $T + 0$ -h-ahead result represents the nowcast result obtained from using the spatial ML model only and compared against the UKMO $T + 0$ -h forecast (where T is the analysis time).

4) PERFORMANCE EVALUATION

To assess the performance of the proposed model framework, five error metrics are considered for the subsequent analysis: root-mean-square error (RMSE), mean absolute error (MAE), mean absolute error percentage (MAPE), mean arctangent absolute percentage error (MAAPE), and the coefficient of determination R^2 . Several metrics must be used as each capture different elements of the model accuracy. RMSE and MAE are scale dependent, while MAPE, MAAPE, and R^2 are dimensionless and scale independent. MAAPE has more intuitive meaning compared with MAPE and can overcome problematic cases when the true values y_i are zero. These five error metrics (Wilks 2019) are calculated by Eqs. (9)–(13), respectively:

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (\hat{y}_i - y_i)^2}, \quad (9)$$

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^N |\hat{y}_i - y_i|, \quad (10)$$

$$\text{MAPE} = \frac{1}{N} \sum_{i=1}^N \left| \frac{\hat{y}_i - y_i}{y_i} \right| \times 100\%, \quad (11)$$

$$\text{MAAPE} = \frac{1}{N} \sum_{i=1}^N \arctan\left(\left| \frac{\hat{y}_i - y_i}{y_i} \right|\right) \times 100\%, \quad (12)$$

$$R^2 = 1 - \frac{\sum_{i=1}^N (\hat{y}_i - y_i)^2}{\sum_{i=1}^N (\bar{y} - y_i)^2}, \quad (13)$$

where N denotes the number of predicted samples, and \hat{y} , y , and \bar{y} represent the predicted value from the neural network, the true value, and the mean of true values, respectively.

As wave direction is a circular parameter, the results near 0° and 360° are equivalent but may have large absolute differences. To account for this, the computation of the wave direction bias is therefore limited to be within 0° and 180° using a circular bias transformation. This circular bias term is then used in place of the traditional bias $\hat{y}_i - y_i$ in the error metrics (RMSE, MAE, MAPE, and MAAPE). The R^2 is, however, not considered in the assessment of wave direction. The observed statistical wave direction parameters Dir_m and Dir_p fluctuate frequently over a wide range, while our model target is to have prediction with constant and stable accuracy; a directional bias within 20° may therefore be considered satisfactory, but its corresponding R^2 may be a negative value.

b. Data preprocessing

Within the data used, the variables correspond to different observed wave parameters at the three buoy locations, resulting in a total of 21 features; the number of days correspond to the period between 0000:00 UTC 1 January 2010 and 2330:00 UTC 31 December 2020, i.e., 4018 days with 48 time steps (half-hourly interval) in each day, resulting in 192 864 time samples. The observations contain both missing and invalid values (the number of records and missing ratios from the three buoys are shown in Table 3). The situations where missing values occur in any single parameter or in any buoy account for 5.17% of the total dataset. The LRTC-TNN algorithm described in section 2b(1) was therefore used for gap filling the data (for all three buoys) over the 11-yr period.

To evaluate the performance of the gap-filling method, additional gaps were artificially introduced. The training sought to minimize the normalized RMSE between the last two matrices, terminating either when the tolerance reaches 10^{-4} or the number of iterations reaches 150. The other details of the algorithm, as well as the results and discussion of the algorithm performance in relation to the present wave dataset, are

TABLE 3. The number of records and their missing ratios against total samples (192 864) from each of the three input buoys, and the combined buoy means and number of records with all three buoys available concurrently.

Buoy	Recorded	Missing proportion
Penzance	189 459	1.77%
Looe Bay	189 542	1.72%
Perranporth	188 983	2.01%
Combined	182 887	5.17%

discussed in Chen et al. (2022). Here, the gap-filling results were obtained by training on a portion of the raw data by randomly replacing 20% of valid values with gaps. The RMSE, MAPE, and R^2 between the filled gaps and raw observations at artificial missing entries were compared (Table 4). Here, it is seen that R^2 exceeded 0.93 for equivalent mean parameters, such as H_s and T_z , and exceeded 0.998 for SST from three buoys, with MAPE below 10%. Direction-related parameters and extreme or peak parameters such as H_{\max} and T_p generally had larger errors but were still generally within 20% MAPE.

After the gap-filling process, the entire 11-yr reconstructed multivariate observations matrix (192 816 time samples \times 21 features) was then used to train and test the temporal forecasting neural networks. Separate test datasets were constructed for each month of 2020, each relying on a training dataset made up of the month itself, the preceding month, and the subsequent month for each of the years from 2010 to 2019. For example, for the case where October 2020 is the test dataset, the corresponding model would be trained using data from the months of September, October, and November from the previous 10 years of available data.

The ElasticNetCV algorithm (Zou and Hastie 2005) was applied to the complete reconstructed observation matrix to find the optimal input features D_i against each individual target output feature. The input features selected for single-variate output can be seen in Table 5 for H_s and T_z . The impacts of this feature selection are explored in appendix C.

Following the feature selection process, it was found that the 12-h predictions of H_s for the Penzance buoy only requires consideration of H_s and H_{\max} at Penzance, while the predictions for other parameters not only make use of the wave parameters at the buoy location but also consider the parameters from the other buoy locations. This highlights the ability for the proposed model framework to make use of sensor networks that can capture spatial correlations within the study area, which is potentially at play in the region studied here. Other parameters such as SST, though available at the buoy locations, were identified through the feature selection process as unnecessary at this stage.

c. Temporal model development

1) MODEL DESCRIPTION

The models were constructed and executed on a laptop with an Intel(R) Core (TM) i7-8550U CPU at 1.80 GHz and 16.0-GB RAM. The open-source ML library scikit-learn

TABLE 4. Statistics comparing the data for complete in situ wave parameter datasets to those that have artificially had gaps inserted and then filled using the LRTC-TNN model.

	H_s	H_{max}	T_p	T_z	Dir_p	Spd	SST
Penzance							
RMSE	0.0719	0.2949	1.9484	0.2630	16.6057	4.6092	0.1003
MAPE	0.0983	0.1680	0.2088	0.0437	0.0806	0.1591	0.0059
R^2	0.9815	0.8867	0.7323	0.9398	0.5824	0.6613	0.9987
Looe bay							
RMSE	0.0762	0.4840	2.0458	0.2490	16.8883	4.6418	0.1202
MAPE	0.0844	0.1407	0.2142	0.0396	0.0716	0.1609	0.0061
R^2	0.9860	0.8086	0.6978	0.9447	0.7434	0.5793	0.9982
Perranporth							
RMSE	0.1131	0.7889	1.3778	0.2450	13.5401	6.4509	0.1023
MAPE	0.0585	0.1474	0.1088	0.0315	0.0642	0.1701	0.0052
R^2	0.9861	0.7805	0.7982	0.9676	0.5507	0.6434	0.9987

(Pedregosa et al. 2011) and deep learning framework Tensor-Flow (Abadi et al. 2016) in Python were used to implement the models.

Hyperparameter tuning is not the principal focus of this paper, and a specific learning rate and network size are picked for all LSTM architectures rather than iteratively optimized through the training process. It is therefore noted that each of these architectures could be further improved given further tuning.

For initial comparison, the vanilla encoder–decoder and attention-based encoder–decoder LSTM were compared against one another to identify the best-performing LSTM architecture. For each LSTM architecture, the weight matrices were initialized with a truncated normal distribution called a “Glorot normal” (Glorot and Bengio 2010). This distribution is centered on 0 with a standard deviation $\sigma = \sqrt{2/(\text{fan}_{in} + \text{fan}_{out})}$, where fan_{in} is the number of input units in the weight tensor, and fan_{out} is the number of output units in the weight tensor. Further, initial states h_0 and c_0 are initialized by zero vectors. For each model, the batch size is set to 16, and the number of epochs is set to 25 as default, consistent with recommendations made by Pirhooshyaran and Snyder (2020). Other model parameters settings can be found in Table 6.

We used the mean-square error (MSE) as the loss function for training the neural network structures and the Adam

optimization algorithm to minimize the loss function due to its strengths with respect to stability and speed of convergence (Pirhooshyaran et al. 2020).

2) MODEL COMPARISON

The four proposed LSTM architectures are the vanilla LSTM model, the stacked LSTM model, the encoder–decoder LSTM model, and the attention-based encoder–decoder LSTM model (as described in Table 6). These were compared against a baseline naïve persistence model, which takes the initial value as the prediction for all steps (i.e., the forecast assumes the conditions do not change).

As a result of our training strategy [see section 2b(2)], the predictions for each month required separate training models. The model performance in different months exhibits a similar trend, but in this section, only the results for May 2020 are presented as an illustrative comparison.

Table 7 shows the overall (24 time step) error comparison for each of the three buoys, in terms of H_s and T_z prediction using the four LSTM network models. It is seen that all LSTM structures outperform the baseline persistence model with respect to RMSE, MAE, and R^2 . Here, the stacked LSTM (model 2) performs better than the vanilla LSTM model (model 1) due to the addition of an LSTM layer. The encoder–decoder LSTM model with (model 4) or without (model 3) attention generally has the higher accuracy in most cases for all three error metrics. For example, the proposed attention encoder–decoder LSTM model (model 4) shows an improvement in R^2 from 0.7795 to 0.8512 over the baseline approach with respect to H_s over 24 forecast time steps at Perranporth.

Figure 7 presents the R^2 variations as a function of forecast time steps. Not surprisingly, for all models the R^2 values from the sequence prediction models decreases as the forecast horizon increases, although this happens more slowly in the LSTM-based models than in the baseline model, and it is particularly shown that model 4, the attention-based LSTM, generally obtains the highest R^2 values across the full forecast

TABLE 5. Feature selection results (H_s , T_z). The correlation time horizon is 24 steps. Subscript indices indicate different buoy locations where 1 corresponds to Penzance, 2 corresponds to Looe Bay, and 3 corresponds to Perranporth.

Buoy	Input features	Output feature
Penzance	H_{s_1}, H_{max_1}	H_{s_1}
	$H_{s_1}, T_{z_1}, H_{s_2}, T_{z_2}$	T_{z_1}
Looe Bay	$H_{s_1}, H_{max_1}, T_{z_1}, H_{s_2}, H_{max_2}$	H_{s_2}
	$H_{s_2}, H_{max_2}, T_{z_2}$	T_{z_2}
Perranporth	$H_{s_1}, H_{max_2}, H_{s_3}, H_{max_3}, T_{z_3}$	H_{s_3}
	$H_{s_1}, T_{p_3}, T_{z_3}$	T_{z_3}

TABLE 6. Experiment LSTM model structures parameter setup. For encoder–decoder models, a slash represents encoder/decoder layers.

Model No.	Model architecture	LSTM layers	Units in LSTM layers	FC layers	Units in FC layer 1	Units in FC output layer	FC layer output dimension
1	Vanilla LSTM	1	200	1	—	n_f	$n_s \times n_f$
2	Stacked LSTM	2	200	1	—	n_f	$n_s \times n_f$
3	Encoder–decoder LSTM	1/1	200/200	2	100	D_o	$n_s \times n_f \times D_o$
4	Attention encoder–decoder LSTM	1/1	200/200	2	100	D_o	$n_s \times n_f \times D_o$

horizon. The relative time complexity of the four models is 397, 826, 567, and 592 s, respectively, when training 6000 samples only. Though model 1 has the shortest training time, it is not resilient, and the performance is significantly worse when compared to the others (Table 7). Models 3 and 4 consistently performed the best and had similar computational times. On this basis, and since it benefits from more flexibility and potential for further extension, model 4 was selected for implementation.

d. Combined spatiotemporal model performance

To validate the proposed spatiotemporal model framework, its performance was assessed by comparing the results with both wave buoy observations and model outputs from the UKMO operational forecast model at two potential renewable energy sites (WaveHub and FabTest) located off the southwest coast of the United Kingdom. As discussed in section 3c, the proposed model framework implements LSTM model 4 (see Table 4) for the temporal forecast model at the buoy locations, and the outputs from this are used as inputs to the surrogate-SWAN spatial nowcasting model to produce the resulting spatiotemporal forecast. The proposed model framework can output continuous 12-h spatial wave parameter forecasts at 30-min intervals in a single model. The predictions for the two validation locations are

then extracted from outputs of this coupled spatiotemporal model.

1) MODEL PERFORMANCE AT WAVEHUB SITE

A qualitative comparison of time series data from the test dataset (January–December 2016) is presented in Fig. 8. Both the proposed model framework and UKMO model show consistency with the buoy observations, albeit with the largest errors occurring when the actual values are very low (i.e., <0.5 m). Quantitatively, the proposed model framework exhibits a comparable level of accuracy to the UKMO physics-based model across all the wave parameters (Table 8). With respect to H_s , the UKMO model performed better against measured data, with R^2 values exceeding 0.92 at all forecast horizons. In contrast, with respect to T_z , the proposed model framework achieves a higher level of accuracy than the UKMO model in nowcast and short-term (i.e., less than 6 h) forecast horizons, but this accuracy decreases as the forecast lead time increases. For Dir_m , with the circular scale metrics transformation, the UKMO model performs better with MAAPE smaller than 9%, while the MAAPE of the proposed model framework is 13%.

In terms of stability, the UKMO model results indicate a consistent performance in forecasting up to 12 h ahead,

TABLE 7. Model structure evaluation metrics comparison. Training data: April, May, and June of 2010–19. Test data: May 2020. Historical time steps = 48 and forecast time steps = 24. Feature selection applied before training.

Buoy	Model No.	H_s			T_z		
		Overall RMSE (m)	Overall MAE (m)	Overall R^2	Overall RMSE (s)	Overall MAE (s)	Overall R^2
Penzance	Baseline	0.1606	0.0983	0.7008	0.9148	0.6449	0.0521
	1	0.1492	0.0938	0.7418	0.7535	0.5844	0.3569
	2	0.1489	0.0937	0.7428	0.7514	0.5820	0.3605
	3	0.1487	0.0931	0.7434	0.7519	0.5787	0.3597
	4	0.1488	0.0935	0.7429	0.7466	0.5735	0.3687
Looe Bay	Baseline	0.1894	0.1256	0.7176	0.8378	0.5627	0.1822
	1	0.1553	0.1085	0.8101	0.6983	0.5164	0.4318
	2	0.1541	0.1071	0.8130	0.6834	0.5032	0.4559
	3	0.1529	0.1059	0.8158	0.6754	0.4975	0.4685
	4	0.1530	0.1073	0.8156	0.6794	0.5015	0.4622
Perranporth	Baseline	0.3346	0.1898	0.7795	1.1046	0.7317	0.4654
	1	0.2892	0.1818	0.8352	0.9839	0.7060	0.5759
	2	0.2814	0.1776	0.8440	0.9790	0.7017	0.5800
	3	0.2768	0.1725	0.8491	0.9841	0.7029	0.5757
	4	0.2749	0.1731	0.8512	0.9771	0.6942	0.5817

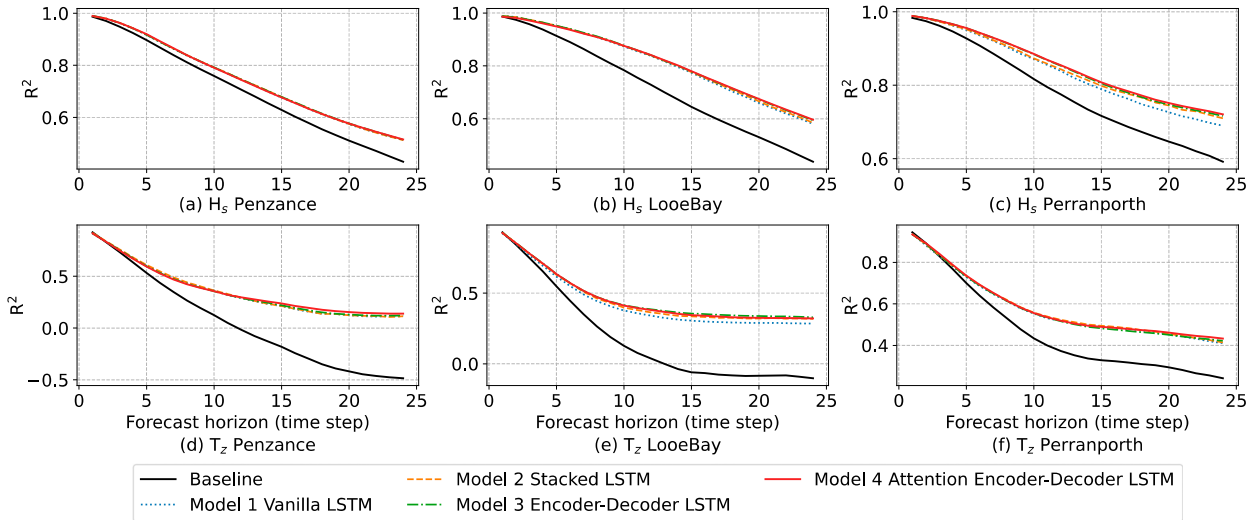


FIG. 7. LSTM model comparison of 24-step forecasting results in terms of R^2 over test data. Training data: April, May, and June of 2010–19, and test data: May 2020. Recurrent time steps = 48, forecast time steps = 24, and feature selection applied before training.

whereas the accuracy of the proposed model framework decreased (i.e., errors increase) with increased forecast lead time (see Table 8; Fig. 9) as shown in scatterplots of the proposed model (Figs. 10, 11, top row) that exhibit increased spread with increased forecast lead time, though this is not apparent for the UKMO model (Figs. 10, 11, bottom row). For both H_s and T_z , the UKMO model appears to slightly overpredict large values while the proposed model framework underpredicts. However, the shape of scatterplots does not change significantly with respect to the forecast horizon for either model. Focusing on storm events shows the proposed method performs inconsistently with some large H_s events well estimated, while the largest peak is significantly underestimated (Fig. 12).

2) MODEL PERFORMANCE AT FABTEST SITE

A qualitative comparison of the FabTest site time series data (January–August 2020) is presented in Fig. 13. When compared to wave buoy observations, both the proposed model framework and the UKMO model are less accurate at FabTest than at WaveHub. However, the proposed model framework is consistently less accurate (i.e., higher errors) and the UKMO model is more accurate in all statistics for H_s , T_z , and Dir_m over the forecast time horizon (Table 9; Fig. 14).

As with WaveHub, the UKMO model results indicated more stable accuracy, while the accuracy of the proposed model framework again reduces with forecast lead time (Fig. 15). However, in this case, the proposed model framework shows a group of results with consistent underprediction of H_s .

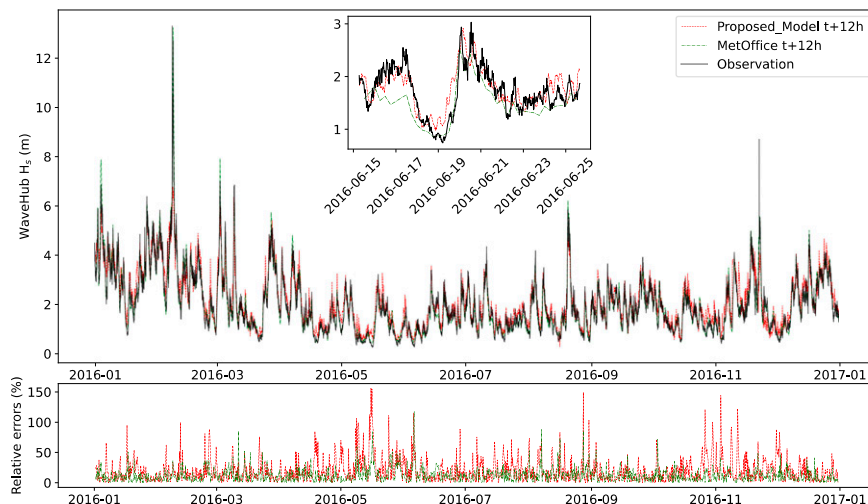


FIG. 8. Overall H_s , 12-h forecasting and relative errors for WaveHub in 2016.

TABLE 8. Validation and benchmarking of proposed model framework with Met Office forecast and in situ measurements at WaveHub.

WaveHub 2016		Proposed model framework				Met Office forecast			
Parameter	Horizon	RMSE	MAE	MAAPE	R^2	RMSE	MAE	MAAPE	R^2
H_s	0 h	0.3331	0.2537	0.1456	0.9156	0.3024	0.2138	0.1176	0.9324
	1 h	0.3469	0.2648	0.1516	0.9083	0.3335	0.2170	0.1169	0.9210
	6 h	0.3882	0.2936	0.1741	0.8852	0.3045	0.2139	0.1175	0.9315
	12 h	0.4317	0.3203	0.1888	0.8581	0.3169	0.2170	0.1185	0.9258
T_z	0 h	0.7553	0.5831	0.0995	0.7017	0.8380	0.6356	0.1032	0.6401
	1 h	0.7297	0.5652	0.0949	0.7210	0.8415	0.6415	0.1046	0.6331
	6 h	0.7957	0.6171	0.1028	0.6682	0.8423	0.6416	0.1041	0.6365
	12 h	0.8731	0.6740	0.1119	0.6006	0.8486	0.6490	0.1055	0.6307
Dir	0 h	37.448	26.068	0.1228	-0.142	27.752	17.94	0.0886	0.502
	1 h	35.772	25.387	0.1254	-0.393	27.982	17.837	0.090	0.504
	6 h	36.168	26.212	0.1303	-0.700	28.137	17.867	0.0884	0.493
	12 h	36.007	26.468	0.1320	-1.352	29.115	18.175	0.0892	0.467

Filtering the results to highlight values with a proportional difference between observations and model prediction greater than 30% shows that these underpredictions most commonly occur when the mean wave direction was approximately 100° (Fig. 16). The errors associated with wave direction were only evident in the H_s values, and errors in T_z were not limited to this subset of conditions. Table 9 and Fig. 17 show consistent differences between observations and the proposed model framework. Focusing on the storms again shows inconsistent results, with some peaks well forecast while others are underpredicted (Fig. 18).

4. Discussion

The results presented in this paper demonstrate the capacity for an ML-based wave forecasting framework. The proposed model framework is based on coupling forecast parameters at discrete locations using an LSTM neural network to a spatial nowcasting model. It is shown to provide reliable short-term forecasts for wave parameters (e.g., H_s , T_z) across a spatial domain. The work presented has highlighted that this approach can achieve results similar to a physics-based forecast for short-range prediction, with low computational cost.

This framework is able to produce spatiotemporal forecasts through a data-driven approach relying on in situ sensor networks without any requirement for measurements at the

target output locations and instead relying on a previously executed hindcast model. Compared to traditional wave forecasting methods, this is therefore able to run in a fraction of the time with reduced computational requirements. According to Hu et al. (2021), for a 2-yr run on their computation domain (with an area similar to our model), WWIII needed 24 h with 60 CPUs. In contrast, the trained LSTM for temporal model together with the random forest for the spatial model needed less than 30 s on one CPU to get 12-h spatial wave prediction with a half-hour interval over 2 years, and most of this time is spent for data loading and transfer rather than for computing the forecasts. This framework takes advantage of both in situ observations, which enable the model to issue more frequent forecasts, and of regional hindcast models, which enable the model to achieve higher spatial resolution. Therefore, this work has the potential to support decision-making and reduce costs for maritime-related applications such as offshore renewable energy and also represent a step change in how environmental forecasts can be done using limited but real-time onboard observation data such as operating unmanned vessels.

It is noted that the observations have been gap filled from the raw data prior to being used as inputs for the spatiotemporal model. Using the raw data with gaps reduces the training dataset by 5.17%. This is larger than the proportion of

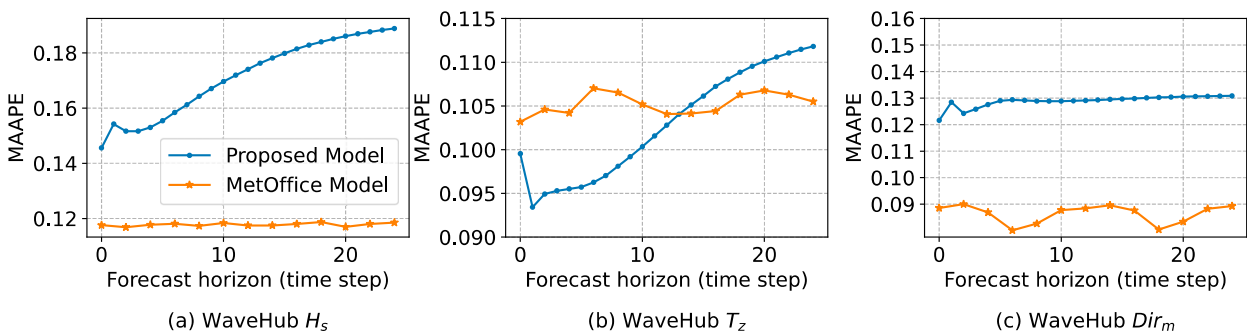


FIG. 9. Accuracy against forecast horizon of present framework and Met Office operational forecasting model compared at WaveHub buoy location for 2016.

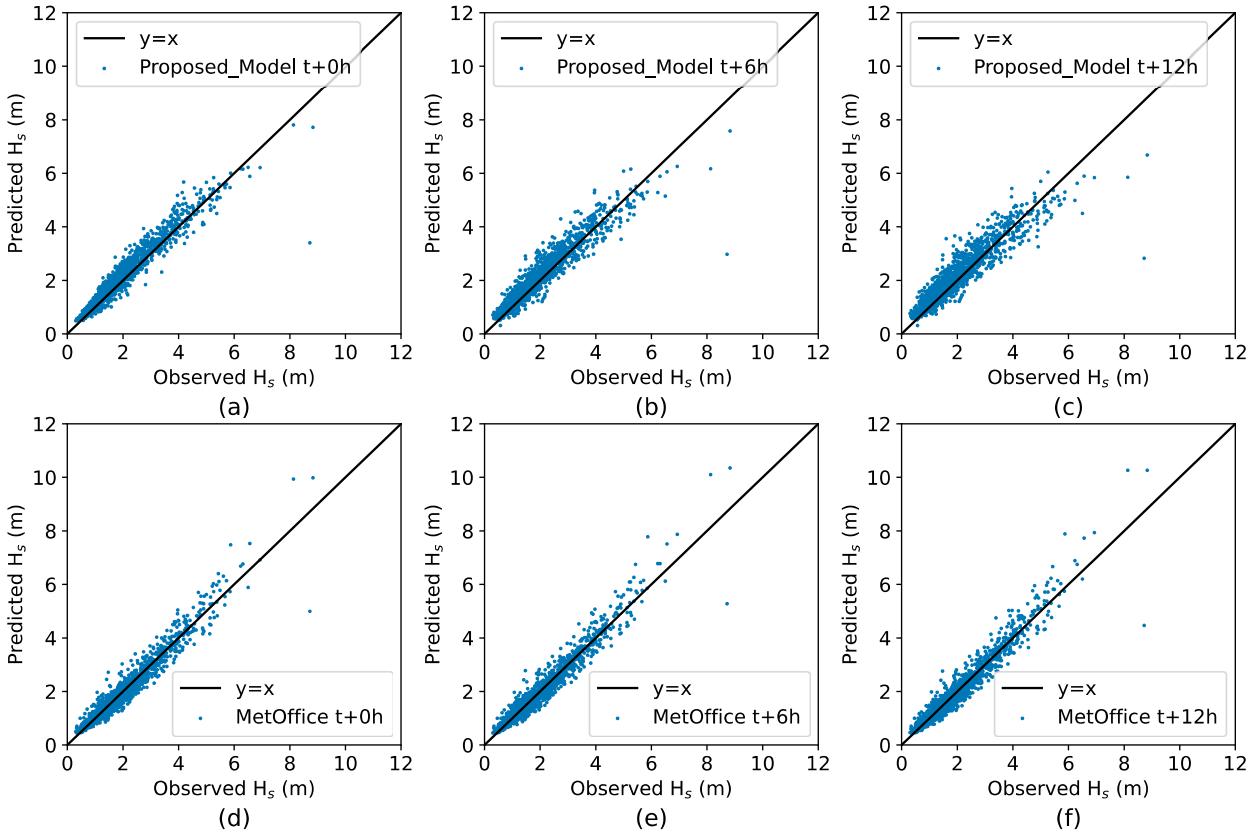


FIG. 10. Scatterplots of model (y) vs observed (x) H_s using buoy observations at WaveHub site: (a) proposed model framework at 0-h horizon, (b) proposed model framework at 6-h horizon, (c) proposed model framework at 12-h horizon, (d) Met Office model at 0-h horizon, (e) Met Office model at 6-h horizon, and (f) Met Office model at 12-h horizon.

missing data for any individual parameter as if any one parameter is missing, then there are no inputs for that time step. While gap filling does not provide a perfect reconstruction of the time series (Chen et al. 2022), the reduction in training data would be more detrimental to the accuracy of the trained model than the uncertainties resulting from the gap-filled data. This also alleviates any bias in the data with gaps, where certain conditions are more likely to result in missing data and therefore are consistently omitted from the training data. Finally, gaps are frequently only for a single parameter in a time step with the others being the original raw data, which further reduces the impact of uncertainties resulting from the gap-filled data.

The results have highlighted that the proposed model framework is able to achieve estimates of H_s and T_z at a similar level of accuracy to the physics-based numerical model. The assessment of the direction is more limited than that of wave height or period in the present work. This is in part due to the differences in wave direction data between the models and the in situ measurements. In the SWAN model, the wave direction refers to the mean wave direction across all frequencies and directions of a two-dimensional wave spectrum. The in situ buoy data from the Channel Coastal Observatory, however, report the peak wave direction, which represents the wave direction at the frequency with the greatest energy

in the wave spectrum. The proposed model framework is sensitive to various factors, including how the training set is selected, what wave parameters are used to train the model, and the overarching architecture of the neural networks. The factor study found that considering seasonality in the training dataset using the best candidate features as inputs and using an attention mechanism-based temporal model that considers the correlation among the input multivariate time series can all help capture the temporal patterns in the wave conditions and thus improve the temporal forecast accuracy.

The proposed model framework performs better at the WaveHub location compared with the FabTest location. Previous examination of numerical model performance found that the regional numerical wave model (SWAN) failed to represent wave systems arriving from the east (i.e., mean direction around 100°) when forced by ECMWF boundary conditions (Ashton and Johannung 2014). This will severely limit the surrogate spatial nowcasting model’s ability to accurately learn the relative conditions across the domain. In contrast, boundary conditions in the northwest of the domain are better defined by the global models used to drive the SWAN training dataset.

Results from FabTest showed consistent errors in the prediction of T_z , which were greater than results from the other sites. These were not limited to wave systems approaching

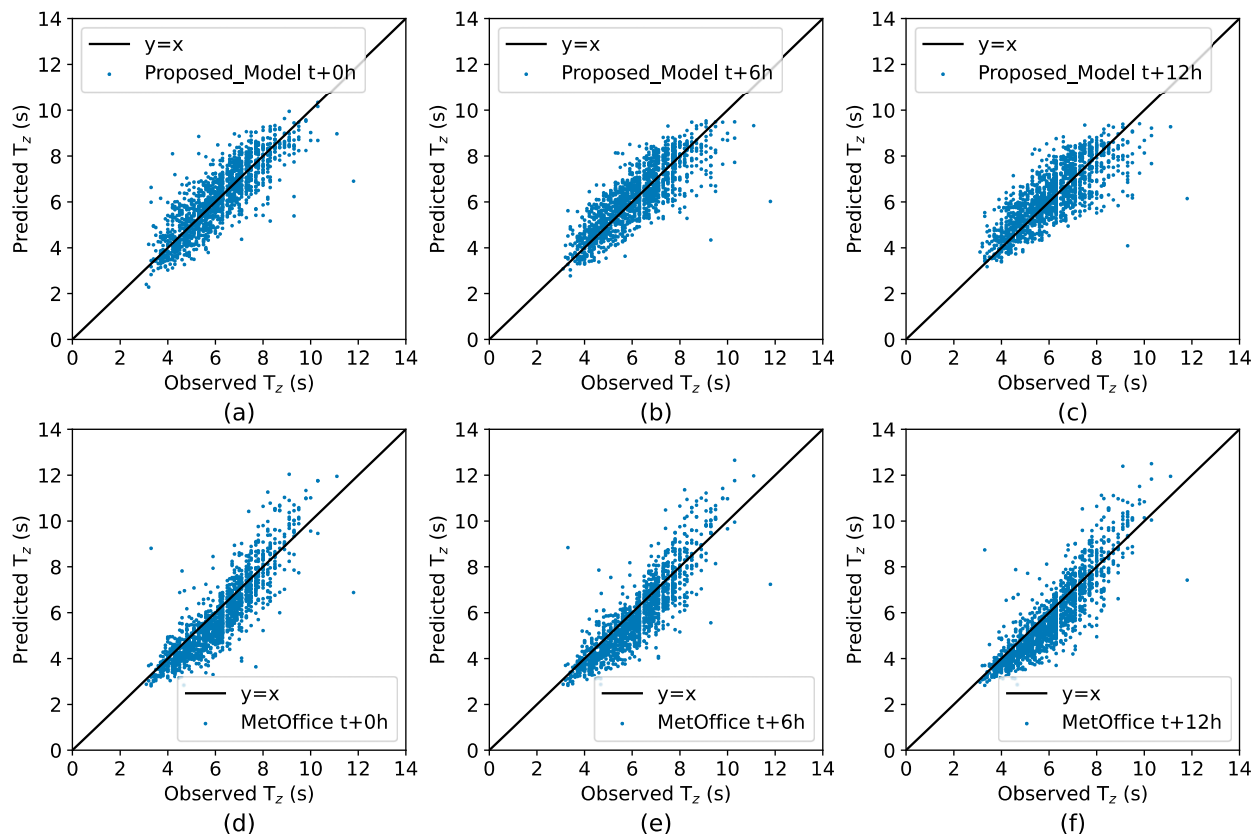


FIG. 11. Scatterplots of model (y) vs observed (x) T_z buoy observations at WaveHub site: (a) proposed model framework at 0-h horizon, (b) proposed model framework at 6-h horizon, (c) proposed model framework at 12-h horizon, (d) Met Office model at 0-h horizon, (e) Met Office model at 6-h horizon, and (f) Met Office model at 12-h horizon.

from the east. [Van-Nieuwkoop et al. \(2013\)](#) attributed differences between the SWAN model and observations to low-frequency energy in the wave spectra, which is underestimated in the ECMWF boundary input. This has the potential to cause in-model errors of processes related to the wave period such as refraction. Where these errors are included in the training dataset, they will be propagated to the surrogate model. The geographic location of FabTest means that incident waves from the predominant west-southwest directions will be more affected by refraction than the other sites. As

such, this could explain the increased errors in T_z at FabTest. However, continued development of the proposed model with different hindcast/reanalysis systems will help to establish the relative contribution of in-model and boundary data errors for future applications.

[Ashton and Johannng \(2014\)](#) and [Van-Nieuwkoop et al. \(2013\)](#) identified that using a UKMO wave model product at the boundaries reduced observed errors, although this was not available at the time to produce the full dataset. This outcome suggests that the surrogate model procedure, while

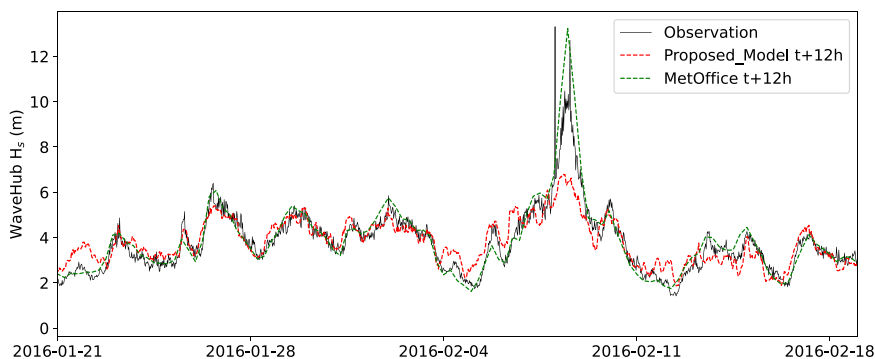


FIG. 12. Example 12-h forecasts of H_s during storm event for WaveHub.

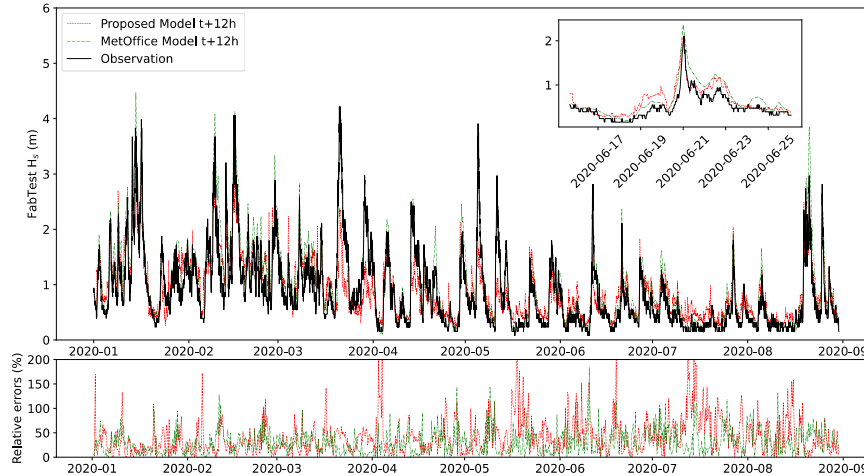


FIG. 13. Overall H_s 12-h forecasting and relative errors for FabTest site in 2020.

capable of reducing errors caused by uncertainty in the boundary datasets (Chen et al. 2021), cannot overcome constant errors caused by numerical model settings and inputs, which would limit the range of conditions used during training the ML model. Here, two examples of such errors are seen to transfer into the surrogate model and the resultant spatiotemporal model framework. In addition, diffraction of wave systems arriving from the North Atlantic and resolution of the wave systems in the English Channel reduce the accuracy of physics-based models when estimating wave conditions at FabTest. This is present both in the physics-based model and the trained model, which relies on an accurate representation of the physics to predict wave conditions.

These results highlight the benefit of using an accurate and reliable hindcast model and the increased potential of the proposed model framework to achieve high-accuracy predictions when trained on best-in-class hindcast datasets. The proposed model also raises the possibility of using simulated data to ensure the full range of conditions are met, including defining spectral shapes and considering local wind conditions to

represent key physical processes accurately. Though the ML model is a tool for using existing computational model setups in a novel way, there is also potential for reviewing the physical processes in the wave model such as white-capping parameterizations, air-sea temperature difference, accuracy of bathymetry, etc., in the context of both training and operation of the machine learning models. Importantly, however, this framework is able to be reconfigured to use an alternative spatial nowcast model such as that based on hindcast data provided by global national weather centers.

5. Conclusions

The paper proposes an ML framework for multivariate multi-step and multioutput spatiotemporal forecasting for nearshore ocean wave characteristics relying on buoy observations and a regional numerical hindcast model. This framework is shown to offer half-hourly forecasts of wave parameters across a spatial domain for up to 12 h ahead with an equivalent level of accuracy with the numerical model.

TABLE 9. Validation and benchmarking of proposed model framework with Met Office forecast and in situ measurements at FabTest.

FabTest 2020		Proposed model framework				Met Office forecast			
Parameter	Horizon	RMSE	MAE	MAAPE	R^2	RMSE	MAE	MAAPE	R^2
H_s	0 h	0.3204	0.2313	0.3142	0.7702	0.2853	0.2137	0.2784	0.8134
	1 h	0.3406	0.2437	0.3234	0.7409	0.2829	0.2100	0.2711	0.8163
	6 h	0.3558	0.2563	0.3486	0.7172	0.2842	0.2111	0.2747	0.8150
	12 h	0.3678	0.2599	0.3540	0.6978	0.2871	0.2144	0.2791	0.8114
T_z	0 h	1.2680	1.1192	0.2433	-1.023	0.7441	0.5334	0.1228	0.3189
	1 h	1.2189	1.0604	0.2293	-0.866	0.7374	0.5276	0.1220	0.2895
	6 h	1.2092	1.0164	0.2157	-0.836	0.7708	0.5478	0.1258	0.2692
	12 h	1.2129	1.0065	0.2119	-0.847	0.7952	0.5678	0.1302	0.2229
Dir	0 h	38.624	24.951	0.1619	0.1149	35.0475	20.7617	0.1288	0.5502
	1 h	37.663	23.761	0.1576	-0.065	34.1053	20.8190	0.13335	0.5746
	6 h	39.516	25.969	0.1734	-0.461	35.3537	20.8904	0.1303	0.5494
	12 h	40.356	26.858	0.1797	-0.743	35.9345	21.5279	0.1350	0.5370

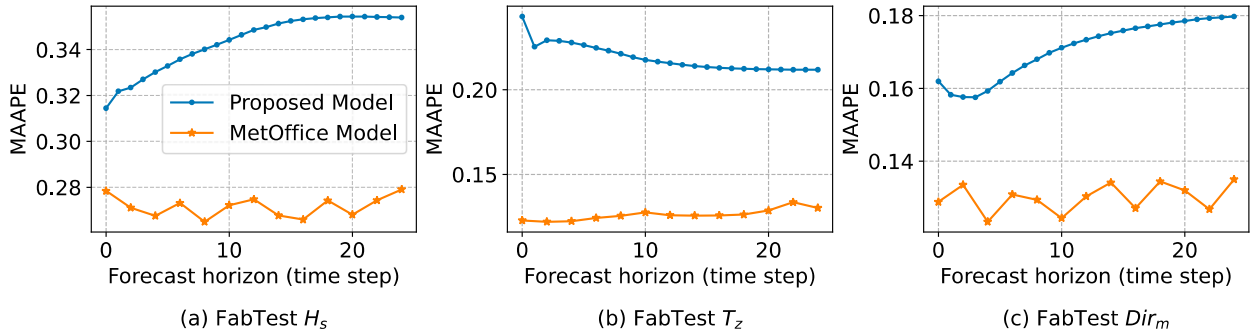


FIG. 14. Accuracy vs forecast horizon of proposed model framework and Met Office operational forecasting model compared at FabTest buoy location in 2020.

The proposed model framework was validated against wave buoy measurements and benchmarked against the Met Office operational forecasting model at two potential renewable energy sites in the southwest United Kingdom: Wave-Hub and FabTest. Results were found to have very similar errors with respect to RMSE, MAE, and MAAPE when benchmarked against a physics-based forecast. However, the proposed model framework was able to achieve this using significantly less computational power and required less than 30 s to generate half-hourly, 12-h spatial forecasts over 2 years.

The results presented highlighted the sensitivity of the proposed model framework to the underlying physics-based hindcast used to train the ML approach. In particular, errors in the initial hindcast model setup were seen to transfer to the surrogate spatial model, indicating that the ML incorporated these errors. This reduced accuracy when compared to the UKMO model and the observations. Future work will explore further development of the model architecture considering further input features from other sensors (i.e., other met-ocean-related parameters such as wind) or other measurement resources (e.g.,

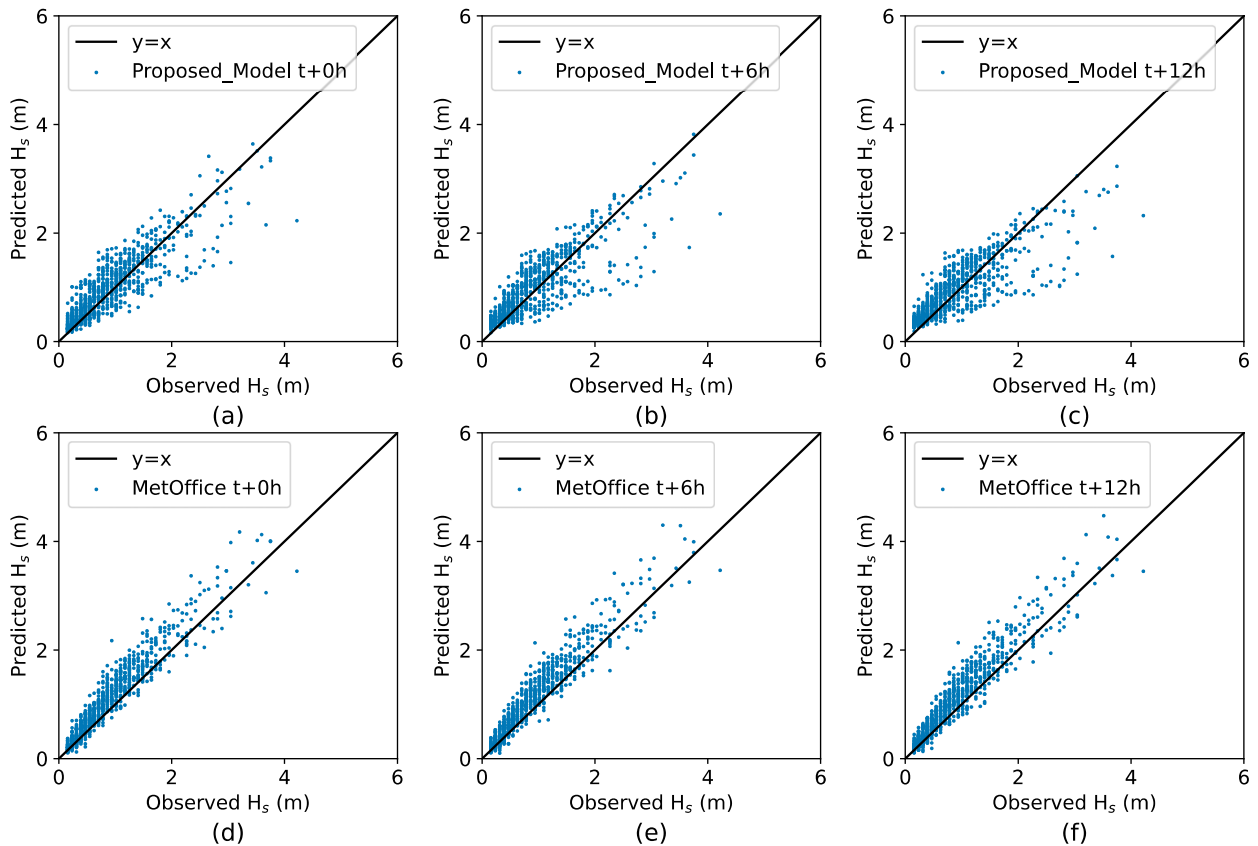


FIG. 15. Scatterplots of model (y) vs observed (x) H_s using buoy observations at FabTest site: (a) proposed model framework at 0-h horizon, (b) proposed model framework at 6-h horizon, (c) proposed model framework at 12-h horizon, (d) Met Office model at 0-h horizon, (e) Met Office model at 6-h horizon, and (f) Met Office model at 12-h horizon.

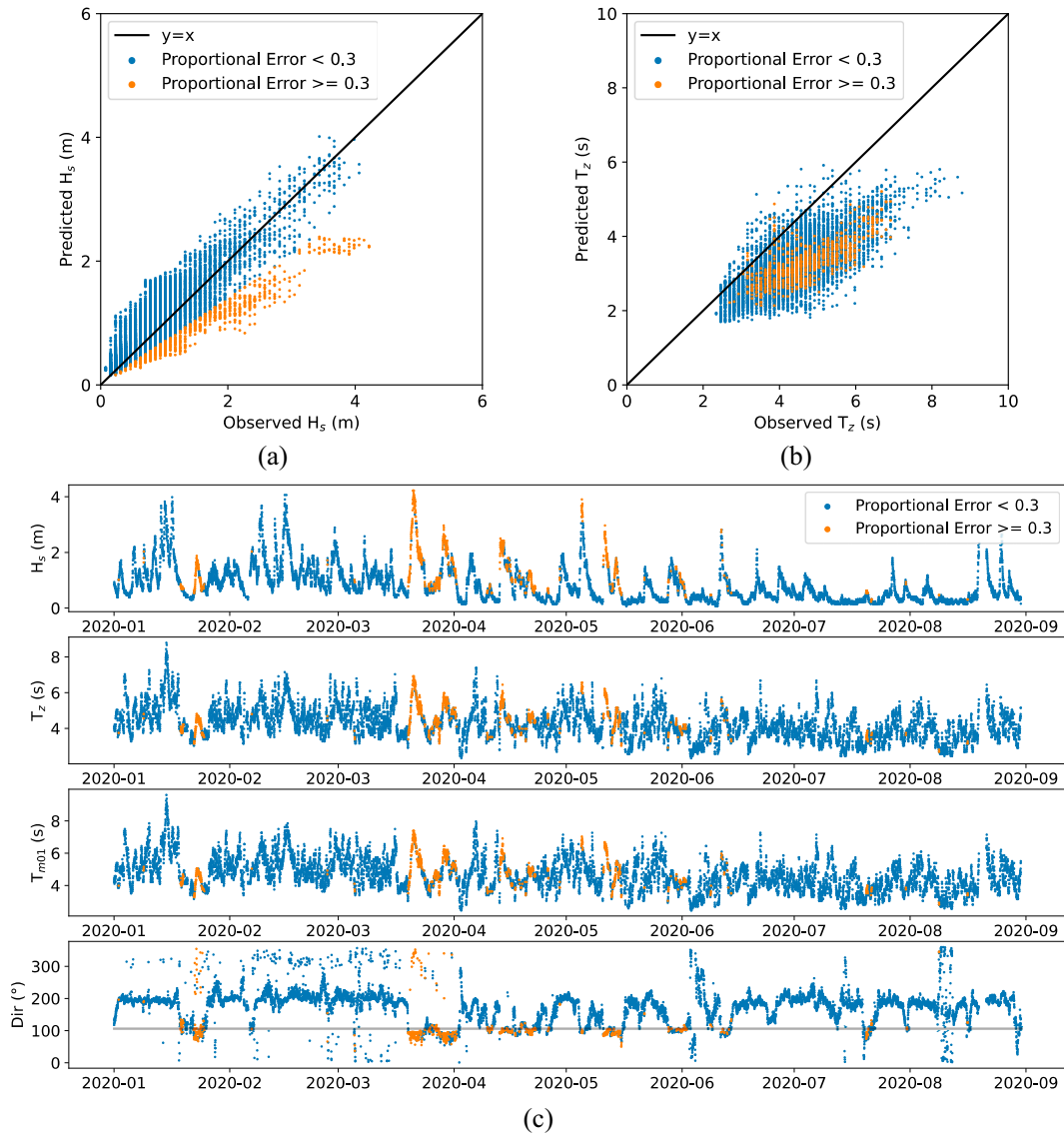


FIG. 16. Filtering forecast scatters of H_s with proportional error: yellow points represent scatters with proportional error > 0.3 , and blue points illustrate scatters with proportional error ≤ 0.3 . (a),(b) The scatters are filtered for H_s and T_z with respect to H_s forecast proportional error. (c) The real observation time series plots for H_s , T_z , T_{m01} , and Dir are filtered with respect to H_s forecast proportional error.

satellite data and mobile autonomous systems) to mitigate these errors in order to demonstrate its robustness and generic application to a full range of physics-based wave models. Further work is also necessary to ensure the generalizability of the proposed methodology to other sites, to global scales, and to extend beyond bulk wave parameters to consider spectral parameters. Performance under extreme and storm-event conditions must also be explored.

The proposed model framework is highly flexible and has potential for offering a low-cost, low computational resource approach for the provision of short-term forecasts and can operate with existing, widely used methodologies. The fact that the current application of the framework can achieve respectable levels

of accuracy compared to a leading physics-based numerical weather prediction suggests that these ideas have significant potential for providing a new class of rapidly updating met-ocean capability.

Acknowledgments. This work was funded by the EPSRC-funded SuperGen Offshore Renewable Energy Hub (Grant EP/S000747/1) under the Machine Learning for Low-Cost Offshore Modeling (MaLCOM) flexible fund project. A. C. Pillai acknowledges support from the Royal Academy of Engineering under the Research Fellowship scheme (Award RF202021\20\175). We are grateful to Teil Howard (Met Office) for her feedback on the final draft of the manuscript. Author contributions are as follows:

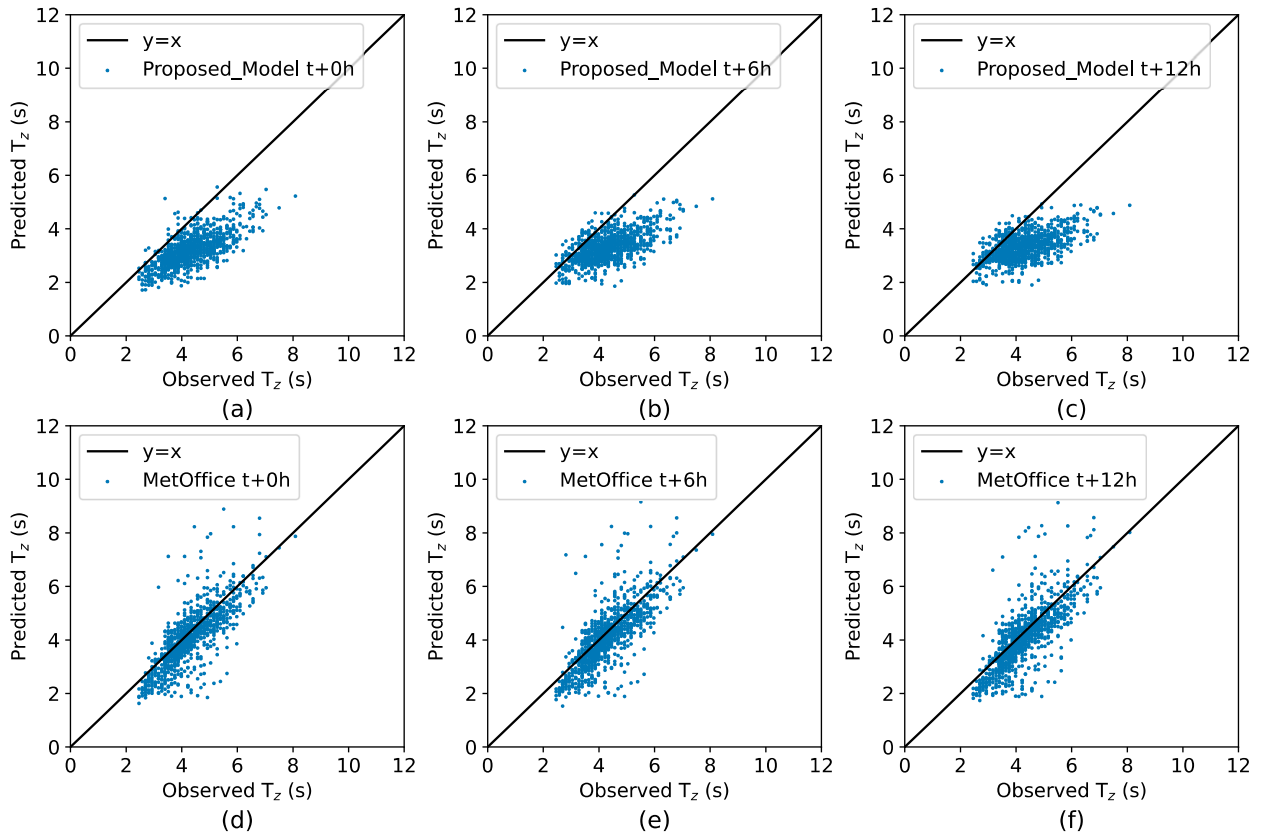


FIG. 17. Scatterplots of model (y) vs observed (x) T_z using buoy observations at FabTest site: (a) proposed model framework at 0-h horizon, (b) proposed model framework at 6-h horizon, (c) proposed model framework at 12-h horizon, (d) Met Office model at 0-h horizon, (e) Met Office model at 6-h horizon, and (f) Met Office model at 12-h horizon.

J. Chen: Conceptualization, methodology, software, validation, formal analysis, investigation, data curation, writing of original draft, writing review and editing, and visualization. Ian G.C. Ashton: Conceptualization, methodology, formal analysis, investigation, writing review and editing, supervision, and funding acquisition. Edward C. C. Steele: Methodology, data curation, and writing review and editing. Ajit C. Pillai: Conceptualization, methodology, formal analysis, investigation, writing review and editing, supervision, and funding acquisition.

Data availability statement. In situ wave data collected using Datawell Directional Wave Rider Mk III buoys and operated by the Channel Coastal Observatory ([Channel Coastal Observatory 2021](#)) were used in the model development described in this manuscript. The spatial surrogate model that the present work builds on is attributed to [Chen et al. \(2021\)](#) for which the underlying physics-based numerical model is attributed to [van Nieuwkoop et al. \(2013\)](#). The benchmark UKMO regional wave forecast is an instance of

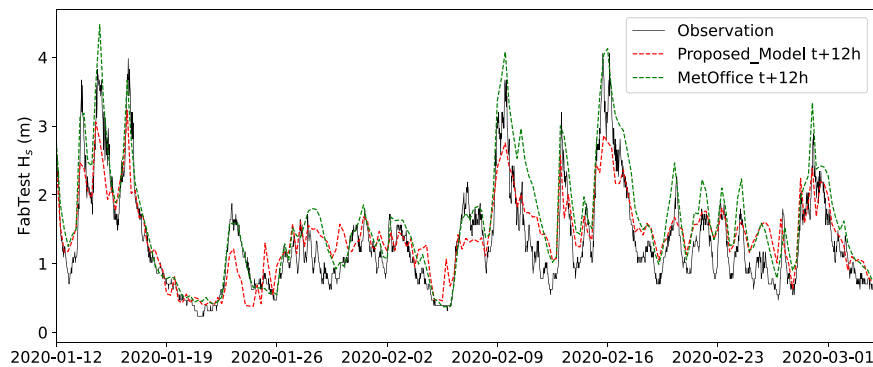


FIG. 18. The 12-h forecasts of H_s during storm events for FabTest.

TABLE A1. Nomenclature.

Acronym	Full form
BPTT	Back propagation through time
\mathbf{C}_v	Context vector
D_i	Input feature dimension
Dir	Wave direction
Dir_m	Mean wave direction
Dir_p	Peak wave direction
DLNN	Deep learning neural network
D_o	Output feature dimension
D_{Total}	All feature dimension
ElasticNetCV	Elastic net algorithm with cross validation
H_{max}	Maximum observed wave height
H_s	Significant wave height
LRTC-TNN	Low-rank tensor completion model with a truncated nuclear norm
LSTM	Long short-term memory neural network
ML	Machine learning
n_b	Lookback time steps
n_f	Forecast time steps
n_s	Number of samples
NWP	Numerical weather prediction
RNN	Recurrent neural network
S_a	Alignment score
seq2seq	Sequence to sequence
Spd	Wave direction spread
SST	Sea surface temperature
SWAN	Simulating Waves Nearshore
T_p	Peak wave period
T_z	Spectrally derived mean zero-crossing wave period
X_{Sp}	Spatial model input data array
\mathbf{X}_{Te}	Temporal model input data array
y_{Sp}	Spatial model output data array
\mathbf{y}_{Te}	Temporal model output data array

the WAVEWATCH III model, whose domain covers the seas on the northwest European continental shelf, forced by 10-m winds from the UKMO atmospheric global Unified Model (Walters et al. 2011), with lateral wave boundary conditions and surface current inputs from the UKMO global wave forecast (Saulter et al. 2016) and UKMO Atlantic Margin Model ocean physics forecast (Tonani et al. 2019), respectively. The open-

source ML library scikit-learn (Pedregosa et al. 2011) and deep learning framework TensorFlow (Abadi et al. 2016) in Python were used to implement the models.

APPENDIX A

Nomenclature

Table A1 lists definitions of the terms used in this paper.

APPENDIX B

LSTM Network

The key feature in an LSTM network is the memory cell. These memory cells are connected along the entire chain and enable the addition or removal of information through their implementation of structures called “gates.” The gates are composed of a sigmoid neural net layer and a pointwise multiplication operation (see Fig. B1). Due to the use of the gates in each cell, data can be disposed, filtered, or added for the next cells, which can therefore ensure the long-term dependency. The standard LSTM equations (Hochreiter and Schmidhuber 1997; Lipton et al. 2015) for time step t are as follows:

$$\text{Forget gate : } \mathbf{f}_t = \sigma(\mathbf{W}_{f_x} \mathbf{x}_t + \mathbf{W}_{f_h} \mathbf{h}_{t-1}), \quad (\text{B1})$$

$$\text{Input gate : } \mathbf{i}_t = \sigma(\mathbf{W}_{i_x} \mathbf{x}_t + \mathbf{W}_{i_h} \mathbf{h}_{t-1}), \quad (\text{B2})$$

$$\text{Output gate : } \mathbf{o}_t = \sigma(\mathbf{W}_{o_x} \mathbf{x}_t + \mathbf{W}_{o_h} \mathbf{h}_{t-1}), \quad (\text{B3})$$

$$\begin{aligned} \text{Cell memory state : } \mathbf{c}_t = & \mathbf{f}_t \odot \mathbf{c}_{t-1} \\ & + \mathbf{i}_t \odot \tanh(\mathbf{W}_{c_x} \mathbf{x}_t + \mathbf{W}_{c_h} \mathbf{h}_{t-1}), \end{aligned} \quad (\text{B4})$$

$$\text{Hidden state : } \mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t), \quad (\text{B5})$$

where $\mathbf{x}_t \in \mathbb{R}^d$, $\mathbf{f}_t \in \mathbb{R}^d$, $\mathbf{i}_t \in \mathbb{R}^d$, $\mathbf{o}_t \in \mathbb{R}^d$, $\mathbf{c}_t \in \mathbb{R}^d$, and $\mathbf{h}_t \in \mathbb{R}^d$ for all $t = 1, 2, 3, \dots, T$ are the input vector, forget gate, input gate, output gate, cell memory state, and hidden state for the LSTM network, respectively, while t represents the time step and \mathbf{W}_{ij} is the weight matrix corresponding to the dimensions of the gate vectors \mathbf{i} and \mathbf{j} .

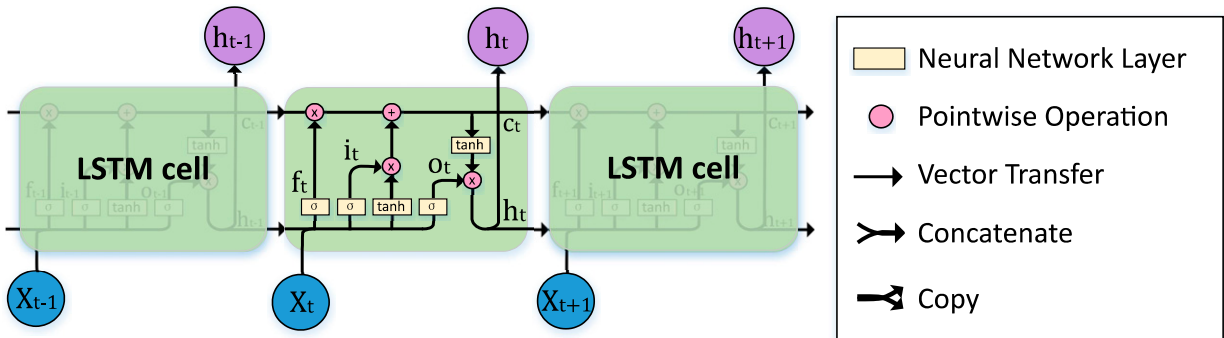


FIG. B1. Three sequential LSTM cells for time steps $t - 1$, t , and $t + 1$. The terms σ and \tanh are the sigmoid and tangent hyperbolic activation functions, respectively; the \times operators indicate elementwise (Hadamard) matrix multiplication; \mathbf{f} , \mathbf{i} , and \mathbf{o} , are forget, input, and output gates at time step t , respectively; and hidden and state cells propagate through the network.

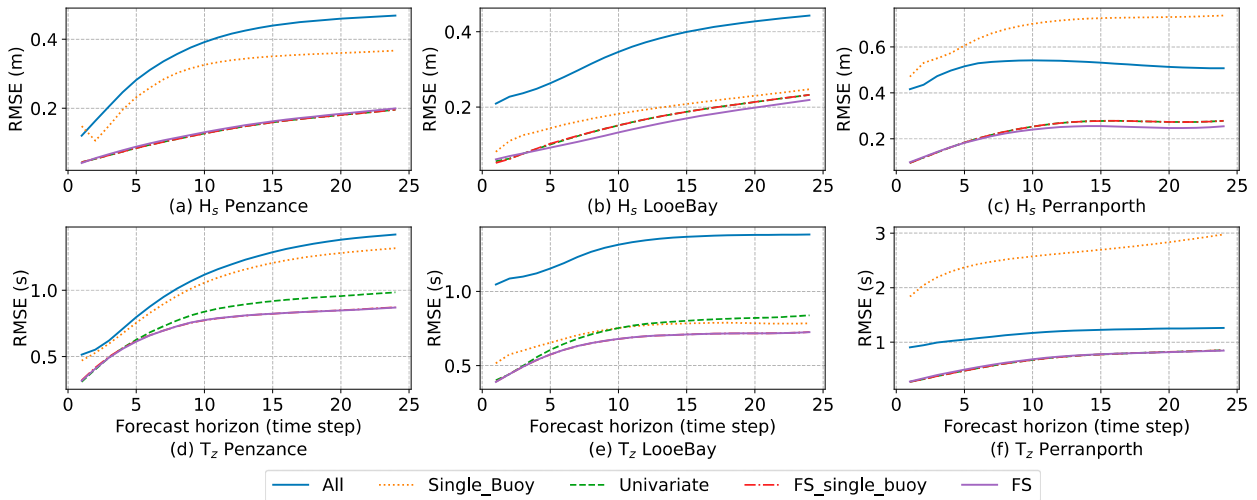


FIG. C1. Feature selection experiments comparison of 24-step single-variate (H_s , T_z) forecasting results in terms of RMSE. Training data: 0000:00 UTC 2 Feb 2010–2330:00 UTC 6 May 2010. Test data: 0000:00 UTC 7 May 2010–2330:00 UTC 7 Jun 2010. Recurrent time steps = 48 and forecast time steps = 24.

The sigmoid function σ is given by $\sigma(x) = 1/(1 + e^{-x})$, $\forall x \in \mathbb{R}$, which returns the value in the open interval of $[0, 1]$. The return values represent the amount of data allowed to pass through the cell, i.e., a value of zero implies that “nothing passes through,” and a value of one implies that “everything passes through.” The notation \odot represents elementwise (Hadamard) matrix multiplication, which only exists if the matrices are of the same dimensions. The internal gates allow the model to be trained using backpropagation through time (BPTT) and avoid the vanishing gradients problem.

and T_z from the three buoys is used in evaluating the value of feature selection.

In all cases, the RMSE values increase with forecast lead time. Using all available features from all three buoys or using all features from each individual buoy generally performs worse than using fewer features strategically selected via the feature selection process (see Fig. C1). The best performance is obtained when using features selected from all the available features across all the buoys (purple lines in Fig. C1).

APPENDIX C

Impact of Feature Selection on Temporal Model

Feature selection has been explored using the Elastic-NetCV for the single-parameter prediction. In this section, five input feature cases representing the range of possible input features are compared:

- 1) all available features across all buoys (i.e., no feature selection): All;
- 2) all available features from a single buoy (i.e., no feature selection for the single buoy): Single_buoy;
- 3) univariate feature, i.e., the target feature itself: Univariate;
- 4) feature selection from a single buoy’s features: FS_single_buoy; and
- 5) feature selection from all available features across all available buoys: FS.

Due to the computational complexity, a shorter dataset from 0000:00 UTC 2 January 2010 to 2330:00 UTC 7 June 2010, corresponding to 7500 samples, was used to explore the impact of feature selection. In this case, the first 6000 samples are treated as the training dataset, and the later 1500 samples are treated as the test dataset. RMSE for H_s ,

REFERENCES

- Abadi, M., and Coauthors, 2016: Tensorflow: A system for large-scale machine learning. *Proc. 12th USENIX Symp. On Operating Systems Design and Implementation (OSDI 16)*, Savannah, GA, USENIX, 265–283, <https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf>.
- Ardente, F., M. Beccali, M. Cellura, and V. Lo Brano, 2008: Energy performances and life cycle assessment of an Italian wind farm. *Renewable Sustainable Energy Rev.*, **12**, 200–217, <https://doi.org/10.1016/j.rser.2006.05.013>.
- Ashton, I. G. C., and L. Johanning, 2014: Wave energy testing in Cornwall: Were the waves during winter 2013/14 exceptional? *Proc. ASRANet Int. Conf. on Offshore Renewable Energy*, Glasgow, United Kingdom, ASRANet, <https://ore.exeter.ac.uk/repository/handle/10871/128473>.
- Bahadori, M. T., Q. Yu, and Y. Liu, 2014: Fast multivariate spatio-temporal analysis via low rank tensor learning. *Proc. 27th Int. Conf. on Neural Information Processing Systems*, Montreal, QC, Canada, ACM, 3491–3499, <https://dl.acm.org/doi/10.5555/2969033.2969216>.
- Bailey, K., and Coauthors, 2019: Coastal mooring observing networks and their data products: Recommendations for the next decade. *Front. Mar. Sci.*, **6**, 180, <https://doi.org/10.3389/fmars.2019.00180>.
- Balog, I., P. M. Ruti, I. Tobin, V. Armenio, and R. Vautard, 2016: A numerical approach for planning offshore wind farms from

- regional to local scales over the Mediterranean. *Renewable Energy*, **85**, 395–405, <https://doi.org/10.1016/j.renene.2015.06.038>.
- Bishop, C. M., 1995: *Neural Networks for Pattern Recognition*. Oxford University Press, 498 pp.
- Bondur, V. G., V. A. Dulov, A. B. Murynin, and V. Y. Ignatiev, 2016: Retrieving sea-wave spectra using satellite-imagery spectra in a wide range of frequencies. *Izv., Atmos. Oceanic Phys.*, **52**, 637–648, <https://doi.org/10.1134/S0001433816060049>.
- Booij, N., R. C. Ris, and L. H. Holthuijsen, 1999: A third-generation wave model for coastal regions: 1. Model description and validation. *J. Geophys. Res.*, **104**, 7649–7666, <https://doi.org/10.1029/98JC02622>.
- Boy, F., J.-D. Desjonquères, N. Picot, T. Moreau, and M. Raynal, 2017: CryoSat-2 SAR-mode over oceans: Processing methods, global assessment, and benefits. *IEEE Trans. Geosci. Remote Sens.*, **55**, 148–158, <https://doi.org/10.1109/TGRS.2016.2601958>.
- Centurioni, L. R., 2018: Drifter technology and impacts for sea surface temperature, sea-level pressure, and ocean circulation studies. *Observing the Oceans in Real Time*, R. Venkatesan et al., Eds., Springer, 37–57.
- Channel Coastal Observatory, 2021: National Network of Regional Coastal Monitoring Programmes. <https://coastalmonitoring.org/>.
- Chen, J., A. C. Pillai, L. Johannang, and I. Ashton, 2021: Using machine learning to derive spatial wave data: A case study for a marine energy site. *Environ. Modell. Software*, **142**, 105066, <https://doi.org/10.1016/j.envsoft.2021.105066>.
- , I. G. C. Ashton, and A. C. Pillai, 2022: Wave record gap-filling using a low-rank tensor completion model. *ASME 41st Int. Conf. on Ocean, Offshore and Arctic Engineering (OMAE2022)*, Hamburg, Germany, ASME, OMAE2022-79897, <https://doi.org/10.1115/OMAE2022-79897>.
- Chen, X., J. Yang, and L. Sun, 2020: A nonconvex low-rank tensor completion model for spatiotemporal traffic data imputation. *Transp. Res.*, **117C**, 102673, <https://doi.org/10.1016/j.trc.2020.102673>.
- Desouky, M. A. A., and O. Abdelkhalik, 2019: Wave prediction using wave rider position measurements and NARX network in wave energy conversion. *Appl. Ocean Res.*, **82**, 10–21, <https://doi.org/10.1016/j.apor.2018.10.016>.
- DNVGL, 2018: N001 marine operations and marine warranty. DNVGL, <https://www.dnv.com/training/marine-operations-standard-dnv-st-n001-173725>.
- Du, S., T. Li, Y. Yang, and S.-J. Horng, 2020: Multivariate time series forecasting via attention-based encoder–decoder framework. *Neurocomputing*, **388**, 269–279, <https://doi.org/10.1016/j.neucom.2019.12.118>.
- Effrosynidis, D., and A. Arampatzis, 2021: An evaluation of feature selection methods for environmental data. *Ecol. Inform.*, **61**, 101224, <https://doi.org/10.1016/j.ecoinf.2021.101224>.
- Gentry, R. R., S. E. Lester, C. V. Kappel, C. White, T. W. Bell, J. Stevens, and S. D. Gaines, 2017: Offshore aquaculture: Spatial planning principles for sustainable development. *Ecol. Evol.*, **7**, 733–743, <https://doi.org/10.1002/ece3.2637>.
- Global Wind Energy Council, 2020: Global Offshore Wind Report 2020. GWEC Rep., 102 pp., <https://gwec.net/wp-content/uploads/2020/12/GWEC-Global-Offshore-Wind-Report-2020.pdf>.
- Glorot, X., and Y. Bengio, 2010: Understanding the difficulty of training deep feedforward neural networks. *Proc. 13th Int. Conf. on Artificial Intelligence and Statistics*, Sardinia, Italy, PMLR, 249–256, <https://proceedings.mlr.press/v9/glorot10a.html>.
- Greff, K., R. K. Srivastava, J. Koutnik, B. R. Steunebrink, and J. Schmidhuber, 2017: LSTM: A search space odyssey. *IEEE Trans. Neural Networks Learn. Syst.*, **28**, 2222–2232, <https://doi.org/10.1109/TNNLS.2016.2582924>.
- Günther, H., S. Hasselmann, and P. A. Janssen, 1992: The WAM model cycle 4. Tech. Rep. DKRZ-TR-4(REV.ED.). Deutsches Klimarechenzentrum, 109 pp., https://inis.iaea.org/search/search.aspx?orig_q=RN:26000788.
- Guşatı, L. F., S. Menegon, D. Depellegrin, C. Zuidema, A. Faaij, and C. Yamu, 2021: Spatial and temporal analysis of cumulative environmental effects of offshore wind farms in the North Sea basin. *Sci. Rep.*, **11**, 10125, <https://doi.org/10.1038/s41598-021-89537-1>.
- Hochreiter, S., and J. Schmidhuber, 1997: Long short-term memory. *Neural Comput.*, **9**, 1735–1780, <https://doi.org/10.1162/neco.1997.9.8.1735>.
- Hoerl, A. E., and R. W. Kennard, 1970: Ridge regression: Applications to nonorthogonal problems. *Technometrics*, **12**, 69–82, <https://doi.org/10.1080/00401706.1970.10488635>.
- Hu, H., A. J. van der Westhuysen, P. Chu, and A. Fujisaki-Manome, 2021: Predicting Lake Erie wave heights and periods using XGBoost and LSTM. *Ocean Modell.*, **164**, 101832, <https://doi.org/10.1016/j.ocemod.2021.101832>.
- James, R. W., 1957: *Application of Wave Forecasts to Marine Navigation*. U.S. Naval Hydrographic Office, 854 pp.
- James, S. C., Y. Zhang, and F. O'Donncha, 2018: A machine learning framework to forecast wave conditions. *Coastal Eng.*, **137**, 1–10, <https://doi.org/10.1016/j.coastaleng.2018.03.004>.
- Johnston, P., and M. Poole, 2017: Marine surveillance capabilities of the AutoNaut wave-propelled unmanned surface vessel (USV). *Proc. OCEANS 2017*, Aberdeen, United Kingdom, Institute of Electrical and Electronics Engineers, 1–46, <https://doi.org/10.1109/OCEANSE.2017.8084782>.
- Komen, G. J., L. Cavaleri, M. Donelan, K. Hasselmann, S. Hasselmann, and P. Janssen, 1996: *Dynamics and Modelling of Ocean Waves*. Cambridge University Press, 554 pp., <https://doi.org/10.1017/CBO9780511628955>.
- LeCun, Y., Y. Bengio, and G. Hinton, 2015: Deep learning. *Nature*, **521**, 436–444, <https://doi.org/10.1038/nature14539>.
- Lipton, Z. C., J. Berkowitz, and C. Elkan, 2015: A critical review of recurrent neural networks for sequence learning. arXiv, 1506.00019v4, <https://doi.org/10.48550/arXiv.1506.00019>.
- Luong, M.-T., H. Pham, and C. D. Manning, 2015: Effective approaches to attention-based neural machine translation. arXiv, 1508.04025v5, <http://arxiv.org/abs/1508.04025>.
- Meyer, H., C. Reudenbach, T. Hengl, M. Katurji, and T. Nauss, 2018: Improving performance of spatio-temporal machine learning models using forward feature selection and target-oriented validation. *Environ. Modell. Software*, **101**, 1–9, <https://doi.org/10.1016/j.envsoft.2017.12.001>.
- Moon, I.-J., 2005: Impact of a coupled ocean wave–tide–circulation system on coastal modeling. *Ocean Modell.*, **8**, 203–236, <https://doi.org/10.1016/j.ocemod.2004.02.001>.
- Musial, W., P. Spitsen, P. Beiter, P. Duffy, M. Marquis, A. Cooperman, R. Hammond, and M. Shields, 2021: Offshore wind market report: 2021 edition. U.S. Department of Energy, 119 pp.
- O'Donncha, F., Y. Zhang, B. Chen, and S. C. James, 2018: Ensemble model aggregation using a computationally lightweight machine-learning model to forecast ocean waves. arXiv, 1812.00511v2, <https://doi.org/10.48550/arXiv.1812.00511>.

- Oh, J., and K.-D. Suh, 2018: Real-time forecasting of wave heights using EOF-wavelet-neural network hybrid model. *Ocean Eng.*, **150**, 48–59, <https://doi.org/10.1016/j.oceaneng.2017.12.044>.
- Pedregosa, F., and Coauthors, 2011: Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.*, **12**, 2825–2830.
- Perotto, H., L. Farina, and N. Violante-Carvalho, 2020: Spatio-temporal regularization of global ocean waves obtained from satellite and their graphical representation. arXiv, 2012.07102v2, <https://doi.org/10.48550/arXiv.2012.07102>.
- Pirhooshyaran, M., and L. V. Snyder, 2020: Forecasting, hindcasting and feature selection of ocean waves via recurrent and sequence-to-sequence networks. *Ocean Eng.*, **207**, 107424, <https://doi.org/10.1016/j.oceaneng.2020.107424>.
- , K. Scheinberg, and L. V. Snyder, 2020: Feature engineering and forecasting via derivative-free optimization and ensemble of sequence-to-sequence networks with applications in renewable energy. *Energy*, **196**, 117136, <https://doi.org/10.1016/j.energy.2020.117136>.
- Reikard, G., B. Robertson, and J.-R. Bidlot, 2017: Wave energy worldwide: Simulating wave farms, forecasting, and calculating reserves. *Int. J. Mar. Energy*, **17**, 156–185, <https://doi.org/10.1016/j.ijome.2017.01.004>.
- Ren, Z., A. S. Verma, Y. Li, J. J. E. Teuwen, and Z. Jiang, 2021: Offshore wind turbine operations and maintenance: A state-of-the-art review. *Renewable Sustainable Energy Rev.*, **144**, 110886, <https://doi.org/10.1016/j.rser.2021.110886>.
- Ris, R. C., L. H. Holthuijsen, and N. Booij, 1999: A third-generation wave model for coastal regions: 2. Verification. *J. Geophys. Res.*, **104**, 7667–7681, <https://doi.org/10.1029/1998JC900123>.
- Sadeghifar, T., M. Nouri Motlagh, M. Torabi Azad, and M. Mohammad Mahdizadeh, 2017: Coastal wave height prediction using recurrent neural networks (RNNs) in the South Caspian Sea. *Mar. Geod.*, **40**, 454–465, <https://doi.org/10.1080/01490419.2017.1359220>.
- Salcedo-Sanz, S., L. Cornejo-Bueno, L. Prieto, D. Paredes, and R. García-Herrera, 2018: Feature selection in machine learning prediction systems for renewable energy applications. *Renewable Sustainable Energy Rev.*, **90**, 728–741, <https://doi.org/10.1016/j.rser.2018.04.008>.
- Saulter, A., C. Bunney, and J. Li, 2016: Application of a refined grid global model for operational wave forecasting. Met Office Forecasting Research Tech. Rep. 614, 48 pp., https://www.metoffice.gov.uk/binaries/content/assets/metofficegovuk/pdf/research/weather-science/frtr_614_2016p.pdf.
- , —, R. R. King, and J. Waters, 2020: An application of NEMOVAR for regional wave model data assimilation. *Front. Mar. Sci.*, **7**, 579834, <https://doi.org/10.3389/fmars.2020.579834>.
- Schultz, M. G., C. Betancourt, B. Gong, F. Kleinert, M. Langguth, L. H. Leufen, A. Mozaffari, and S. Stadler, 2021: Can deep learning beat numerical weather prediction? *Philos. Trans. Roy. Soc.*, **A379**, 20200097, <https://doi.org/10.1098/rsta.2020.0097>.
- Serras, P., G. Ibarra-Berastegi, J. Sáenz, and A. Ulazia, 2019: Combining random forests and physics-based models to forecast the electricity generated by ocean waves: A case study of the Mutriku wave farm. *Ocean Eng.*, **189**, 106314, <https://doi.org/10.1016/j.oceaneng.2019.106314>.
- Stott, P., 2016: How climate change affects extreme weather events. *Science*, **352**, 1517–1518, <https://doi.org/10.1126/science.aaf7271>.
- Sutskever, I., O. Vinyals, and Q. V. Le, 2014: Sequence to sequence learning with neural networks. *NIPS'14: Proceedings of the 27th International Conference on Neural Information Processing Systems*, Vol. 2, MIT Press, 3104–3112, <https://dl.acm.org/doi/10.5555/2969033.2969173>.
- The Swamp Group, 2013: *Ocean Wave Modeling*. Springer, 256 pp.
- Tibshirani, R., 1996: Regression shrinkage and selection via the Lasso. *J. Roy. Stat. Soc.*, **58B**, 267–288, <https://doi.org/10.1111/j.2517-6161.1996.tb02080.x>.
- Tolman, H. L., 2009: User manual and system documentation of WAVEWATCH III TM version 3.14. Tech. Note, 220 pp.
- , B. Balasubramanian, L. D. Burroughs, D. V. Chalikov, Y. Y. Chao, H. S. Chen, and V. M. Gerald, 2002: Development and implementation of wind-generated ocean surface wave models at NCEP. *Wea. Forecasting*, **17**, 311–333, [https://doi.org/10.1175/1520-0434\(2002\)017<0311:DAIOWG>2.0.CO;2](https://doi.org/10.1175/1520-0434(2002)017<0311:DAIOWG>2.0.CO;2).
- Tonani, M., and Coauthors, 2019: The impact of a new high-resolution ocean model on the Met Office north-west European shelf forecasting system. *Ocean Sci.*, **15**, 1133–1158, <https://doi.org/10.5194/os-15-1133-2019>.
- van Nieuwkoop, J. C. C., H. C. M. Smith, G. H. Smith, and L. Johanning, 2013: Wave resource assessment along the Cornish coast (UK) from a 23-year hindcast dataset validated against buoy measurements. *Renewable Energy*, **58**, 1–14, <https://doi.org/10.1016/j.renene.2013.02.033>.
- Walters, D. N., and Coauthors, 2011: The Met Office unified model global atmosphere 3.0/3.1 and JULES global land 3.0/3.1 configurations. *Geosci. Model Dev.*, **4**, 919–941, <https://doi.org/10.5194/gmd-4-919-2011>.
- WAMDI Group, 1988: The WAM model—A third generation ocean wave prediction model. *J. Phys. Oceanogr.*, **18**, 1775–1810, [https://doi.org/10.1175/1520-0485\(1988\)018<1775:TWMTGO>2.0.CO;2](https://doi.org/10.1175/1520-0485(1988)018<1775:TWMTGO>2.0.CO;2).
- Wilks, D. S., 2019: *Statistical Methods in the Atmospheric Sciences*. Vol. 100, Academic Press, 704 pp.
- Zou, H., and T. Hastie, 2005: Regularization and variable selection via the elastic net. *J. Roy. Stat. Soc.*, **67B**, 301–320, <https://doi.org/10.1111/j.1467-9868.2005.00503.x>.