Wind Energy

WILEY

RESEARCH ARTICLE OPEN ACCESS

Predicting Wind Turbine Blade Tip Deformation With Long Short-Term Memory (LSTM) Models

Shubham Baisthakur | Breiffni Fitzgerald

Department of Civil, Structural and Environmental Engineering, Trinity College Dublin, Dublin, Ireland

Correspondence: Breiffni Fitzgerald (breiffni.fitzgerald@tcd.ie)

Received: 21 February 2024 | **Revised:** 15 April 2025 | **Accepted:** 28 April 2025

Funding: This study was supported by the Science Foundation Ireland (SFI) grant no. 20/FFP-P/8702, the SFI project NexSys grant no. 547 21/SPP/3756 and the Sustainable Energy Authority of Ireland (SEAI) grant no. RDD/604.

Keywords: feature selection | long short-term memory (LSTM) | multivariate time series prediction | surrogate modelling | virtual sensor

ABSTRACT

Driven by the challenges in measuring blade deformations, this study presents a novel machine learning methodology to predict blade tip deformation using inflow wind data and operational parameters. Using a long short-term memory (LSTM) model and a novel feature selection approach based on mutual information and recursive feature addition, this study presents a robust framework for multivariate time series prediction. The developed model offers significant computational cost reductions compared to full-dynamic simulations and also allows virtual sensing. This work empowers efficient and reliable wind turbine operation by providing an accurate and computationally efficient blade response prediction tool that can assist in improved wind turbine management, site-specific analysis and fatigue assessment.

1 | Introduction

Over the last two decades, wind turbines have dramatically increased in size. The growing demand for sustainable energy paired with technological advancements has led to the development of multimegawatt turbines featuring taller hubs and extended blades to enhance energy capture [1]. While scaling up these structures has the benefit of increased energy capture, it also causes a decrease in the stiffness of the towers and blades. As a result, modern wind turbines are more flexible, presenting challenges such as increased fatigue loads, bend-twist coupling and high-frequency vibrations [1–3]. Maintaining consistent reliability is crucial for the safe operation of these structures. According to current IEC design practices, wind turbine models are designed assuming a specific site class, which prescribes the wind conditions to be used for design calculations. Nonetheless, wind and other environmental factors can vary significantly across different locations, resulting in varied load envelopes experienced by these structures. Consequently, turbines designed to IEC standards may not maintain a uniform safety factor across different installation location. In this context, it is important to assess the site-specific load distribution to achieve the desired level of reliability at each installation location.

The high cost of performing dynamic analysis of wind turbines using numerical models, coupled with the requirement of running extensive simulations, prohibits such precise site-specific investigations. To address this, the authors propose a methodology to develop a surrogate model for predicting the dynamic response of wind turbines subjected to turbulent wind inflow

Abbreviations: Adam, adaptive moment estimation; BEM, blade element momentum; IEA, International Energy Agency; IEC, International Electrotechnical Commission; LIDAR, light detection and ranging; LDA, linear discriminant analysis; LSTM, long short-term memory; MIQ, mutual information quotient; MRMR, minimum redundancy maximum relevance; NREL, National Renewable Energy Laboratory; PCA, principal component analysis; PCE, polynomial chaos expansion; RCA, recursive feature addition; RCE, recursive feature elimination; RFA, recursive feature addition; RFE, recursive feature elimination; RMSE, root mean square error; RNN, recurrent neural network; SCADA, supervisory control and data acquisition.

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2025 The Author(s). *Wind Energy* published by John Wiley & Sons Ltd.

using a machine learning-based sequence modelling algorithm. This study specifically focuses on modelling the response of wind turbine blades due to the more challenging nature of this element involving twisted-curved geometry, use of composite materials and relatively lower stiffness. The rotation of the blades adds further complexity. This study aims to predict the blade deformation response using aerodynamic forces at blade nodes as the input. A feature sensitivity study is performed to minimise the number of inputs required for accurate response prediction. The findings from this analysis also provide useful guidelines for the sensor installation. The surrogate model developed in this study can predict the dynamic response of the blade at a fraction of the computational cost required for full-dynamic simulation of the detailed wind turbine model and can make rapid estimations of fatigue damage and load distribution computationally feasible.

During the operational phase, it is important to monitor the behaviour of these structures to derive a quantitative estimate of their performance. However, not all operational parameters are accurately measurable with physical sensors [4]. Additionally, deploying a large number of sensors increases operational costs. Blade-mounted sensors are particularly problematic as their installation disrupts the airfoil's continuity and may negatively impact their aerodynamic efficiency [5]. Thus, there remains a challenge in efficiently monitoring wind turbine performance with a minimum number of sensors without affecting the operations. This study extends the surrogate model approach to function as a virtual sensor, enabling digital measurement of blade deformations. Virtual sensors can optimise the collection of operational parameter data by minimising reliance on physical sensors. A comprehensive framework for developing a machine learning model that functions as a virtual sensor and predicts the dynamic behaviour of blade deformation is presented in this manuscript.

The use of surrogate modelling for site-specific load estimation and fatigue analysis has been widely researched by the wind turbine community using statistical analysis [6, 7], parametric models [8–10] and response surface method [11, 12]. Out of the different surrogate modelling techniques, the majority of the case studies are presented using polynomial chaos expansion (PCE) and Kriging. These methods (PCE and Kriging) offer superior flexibility and accuracy in capturing complex, nonlinear relationships and spatial correlations, outperforming traditional surrogate modelling techniques that rely on simpler, parametric assumptions or linear regression. [13] has presented the state-of-the-art methods in multifidelity surrogate modelling using PCE and has applied them for predicting the fore-aft bending moment in wind turbine towers. [14] used PCE to predict blade loads and [15] has combined PCE with nonnegative matrix factorisation, a dimensionality reduction technique, to predict the long-term degradation of composite blades due to fatigue. PCE has also been used to predict aerodynamic loads on wind turbine rotors [16] and uncertainty propagation [17]. Kriging method has been found effective in similar areas of research, such as [18], who applied Kriging for fatigue damage prediction and reliability analysis [19, 20], who performed reliability-based design optimisation of the tripod substructure for an offshore wind turbine considering dynamic response requirement. The Kriging approach is also applied for reliability analysis [10] and damage detection [21]. Surrogate modelling has been used in diverse applications. [22] has compared the performance of commonly used surrogate modelling techniques in estimating site-specific loads as a function of environmental variables. The authors concluded that PCE is the best approach considering the computational time and performance achieved.

A typical feature found in many of the studies mentioned above is their reliance on predictions centred around aggregated statistics, leading to the development of static models that often fall short of addressing the system's dynamic aspects. While trying to establish a surrogate model linking aggregated statistics, these models frequently fail to effectively represent dynamics occurring at shorter time scales. In addition, although the methods discussed here have been very promising and have shown excellent results, as with any statistical mapping, the accuracy of these models depends on the quality of the data the model is being trained on. The training data should be representative of the actual conditions the model will experience during the operations. Although once the model is trained, the computational cost of subsequent applications is insignificant, generating data for training the model requires significant computational efforts. As such, the statistical surrogate model development approach is not entirely free of computational limitations; instead, the computational cost is shifted from the analysis stage to the model development stage. As an alternative to this approach, this manuscript proposes a long short-term memory (LSTM) model-based approach, which aims to learn the dynamics of the system rather than the statistical mapping. For training these models, the training data generation is independent of the probability distribution of the uncertain variables. Instead, these distributions are used to define the practical bounds of the variables used for training. After training, these models can be used to generate a large amount of data at much less computational cost, which can be used to develop a statistical mapping surrogate. To this end, the presented LSTM model is not a replacement for the statistical surrogate modelling technique but rather a complementary approach to further reduce computational efforts.

The LSTM method has shown its effectiveness in solving a wide range of problems involving sequential data. LSTM models have been successfully implemented in wind turbines for power forecasting [23–25] and damage detection [26–28]. Using a similar approach, this study focuses primarily on predicting the deformation response of the tip of a wind turbine blade. Blade tip deformation is chosen as the target variable, as the actual measurement of blade deformation during wind turbine operation is very challenging; These challenges in blade deformation measurement are highlighted by [29]. [30] in their study further addresses this issue and highlights the lack of studies addressing the challenges in measurement and prediction of blade deformations. [30] have predicted blade deformation using the LSTM method, where the LSTM model is treated as a virtual sensor using the inputs from the supervisory control and data acquisition (SCADA) data and the load measurement system. To get the best performance out of an LSTM model, the optimum selection of input features and network architecture is of paramount importance. In their

10991824, 2025, 6, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/we.70027 by Breifni Fitzgerald - Trinity College Dublin, Wiley Online Library on [16/05/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

study, [30] have used a wrapper-type trial and error-based qualitative approach for feature selection, which is computationally prohibitive when a large number of input features are available, and network architecture based on experience and previous studies has been used, which may not be optimal for the problem being addressed. In addition, the effect of the stochastic nature of the inflow for a set of environmental parameters is not explicitly discussed. In addition, the selection of features for multivariate time series is not a trivial task and has been studied extensively. In this context, the current study presents a mutual information-based minimum redundancy maximum relevance algorithm to select the optimum set of input features. The performance of the commonly used recursive feature selection method with exhaustive feature search is compared, and insights into the computational cost of the respective methods and achieved accuracy are also presented. A combination of wrapper-type and filter-type approaches for optimal feature selection is presented. After feature selection, the Bayesian optimisation model is used to tune the model hyperparameters. A learning rate scheduling approach based on validation loss plateaus is also proposed, which ensures optimal convergence. This study only uses inputs as obtainable from Light Detection and Ranging (LIDAR) and SCADA systems to avoid the uncertainty encountered during load system measurements.

The manuscript is structured as follows: Section 2 discusses the problem formulation and objectives of this study, while Section 3 presents the theoretical formulation of the numerical model used to generate training data. Section 4 deals with the choice of modelling algorithm, variable selection and training data generation. Feature selection methods are discussed in Section 5, followed by hyperparameter tuning in Section 6. Section 7 presents the numerical results, with key findings discussed in Section 8. Finally, conclusions are drawn in Section 9.

2 | Problem Formulation

As described in Section 1, quick prediction of the blade deformation response and its accurate measurement is a challenging task. In this study, the authors propose a surrogate modelling approach that alleviates the computational cost of predicting the dynamic time history of blade tip deformations. Traditionally, numerical methods, such as time-stepping algorithms, are used to solve a numerical model to compute the deformation response. Time-stepping algorithms often require small step sizes for convergence, resulting in lengthy computational times. Consequently, the evaluation of the deformation response contributes significantly to the overall computational time required to analyse the numerical model. In addition, the computational cost increases significantly while performing a large number of simulations. For a wind turbine, this higher computational cost makes optimisation and site suitability analysis computationally exhaustive. To address this challenge, the authors propose employing a data-driven LSTM model to predict blade deformation. LSTM models are particularly well-suited for capturing short- and long-term dependencies within sequential data, making them ideal for time series analysis. LSTM networks can effectively learn complex temporal patterns by adjusting fixed and recurrent weights. LSTM-based predictions significantly reduce computation time compared to complete dynamic analysis, improving efficiency without compromising accuracy.

Typically, when dealing with a large number of features, the recursive feature addition (RFA) method is often used. This approach continuously adds features to the model until their inclusion no longer enhances performance. Similarly, the recursive feature elimination (RFE) method starts with a full set of features and progressively removes them until the drop of a feature significantly affects the prediction accuracy. However, both methods assume linear dependence and prioritise individual feature importance, overlooking feature interactions. Although these methods have delivered good results in many applications, they may yield suboptimal results for nonlinear systems where feature combinations heavily influence system response. To this end, exhaustive feature combinations are tested to effectively study the impact of interactions among input features on prediction accuracy. A comparative analysis of the performance and computational costs of RFA is performed, and an exhaustive feature search for feature selection is also conducted. Additionally, Bayesian optimisation is implemented for hyperparameter tuning, and a loss-based learning rate scheduling approach based on validation loss plateaus is proposed to ensure optimal convergence. This paper offers a comprehensive approach to multivariate time series modelling, addressing various aspects of model development to achieve optimal performance.

3 | Numerical Model of IEA-15-MW Wind Turbine

This study aims to develop a surrogate model for estimating blade tip deformation response in wind turbines. As direct measurement during operation proves challenging, this work relies on the results of a numerical model. The numerical model used in this study is developed using the multibody dynamics approach leveraging the principles of Kane's dynamics [31]. Kane's method was chosen specifically for its efficient handling of the complex interactions between various turbine components, enabling an accurate representation of the overall dynamics. This method reduces the effort required to derive equations of motion and offers simpler computer implementation compared to classical approaches like Euler–Lagrange and D'Alembert's principle. A total of 22° of freedom are considered to capture the dynamics of various components of wind turbine. Six degrees of freedom are used to model the foundation, which includes three translation motions and three rotational motions. The tower is modelled using the modal summation method, using four principal mode shapes representing the tower motion in fore-aft and side-to-side direction. The axial shortening and twisting of the tower from the external load application are not considered. The Generator azimuth and twisting of the low-speed shaft are considered for capturing the rotor speed variations. The blades are modelled as flexible members using the modal summation method where three mode shapes for each blade are considered, two modes representing the blade deformations in flapwise direction and one mode to represent the deformation in edgewise direction. Various reference frames are defined to describe the motion of different components of the system

10991824, 2025, 6, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/we.70027 by Breifni Fitzgerald - Trinity College Dublin , Wiley Online Library on [16/05/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

and to define their orientation with respect to the other members. Using this model representation, the equilibrium equations for a simple holonomic multibody system using Kane's approach are given by the following:

$$F_r + F_r^* = 0 \tag{1}$$

where F_r stands for the generalised active forces and F_r^* represents the inertia force. These forces can be expressed in terms of the kinematic quantities as follows:

$$F_r = \sum_{i=1}^{n} {}^{E}v_r^{X_i} \cdot F^{X_i} + {}^{E}\omega_r^{N_i} \cdot M^{N_i} \tag{2}$$

$$F_r^* = -\sum_{i=1}^{n} {}^{E}v_r^{X_i}\left(m^{N_i}\, {}^{E}a^{X_i}\right) - {}^{E}\omega_r^{N_i} \cdot {}^{E}\dot{H}^{N_i} \tag{3}$$

where F^{X_i} is a force vector acting on the centre of mass of point X_i and M^{N_i} is the moment vector acting on the rigid body N_i. ${}^{E}v_r^{X_i}$ and ${}^{E}\omega_r^{N_i}$ are the partial linear and partial angular velocity of the point X_i and rigid body N_i, respectively. ${}^{E}\dot{H}^{N_i}$ is the time derivative of the angular momentum of the rigid body N_i about its centre of mass X_i in the inertial frame, given by the following equation

$${}^{E}\dot{H}^{N_i} = \overline{\overline{I}}^{N_i} \cdot {}^{E}\alpha^{N_i} + {}^{E}\omega^{N_i} \times \overline{\overline{I}}^{N_i} \cdot {}^{E}\omega^{N_i} \tag{4}$$

The final equation of the system is of the form

$$M(q,t)\ddot{q} + f(q,\dot{q},t) = 0 \tag{5}$$

where $M(q,t)$ is the inertia matrix and $\ddot{q}$ is the acceleration vector, whereas $f(q,\dot{q},t)$ is the force vector consisting of the external and restoring forces acting on the structure. This system of equations is solved using numerical methods. The authors used the fourth-order Runge–Kutta method in this study to solve the system's governing equation. A complete derivation of these equations of motion is beyond the scope of this study, and interested readers can refer to the work of [32] for more insights. The research undertaken in this study is performed using the specifications of the International Energy Agency (IEA) 15-MW wind turbine, which is an International Electrotechnical Commission (IEC) Class 1B direct-drive machine with a rotor diameter of 240 m and a hub height of 150 m. A representative model of the IEA 15-MW wind turbine is shown in Figure 1, reproduced from the technical report on the definition of the IEA 15-megawatt reference wind turbine [33], and the details of operating parameters are presented in Table 1.

Wind turbines experience random turbulent wind inflow during operation, influencing both power output and loads on the turbine components. Therefore, producing a representative wind inflow model is essential for the accuracy of the simulation results. This research employs TurbSim [34] to generate the wind fields acting on a wind turbine. TurbSim is a stochastic tool for simulating turbulent wind patterns, providing a comprehensive model of wind inflow. Using a statistical framework, it generates a time series of three-dimensional

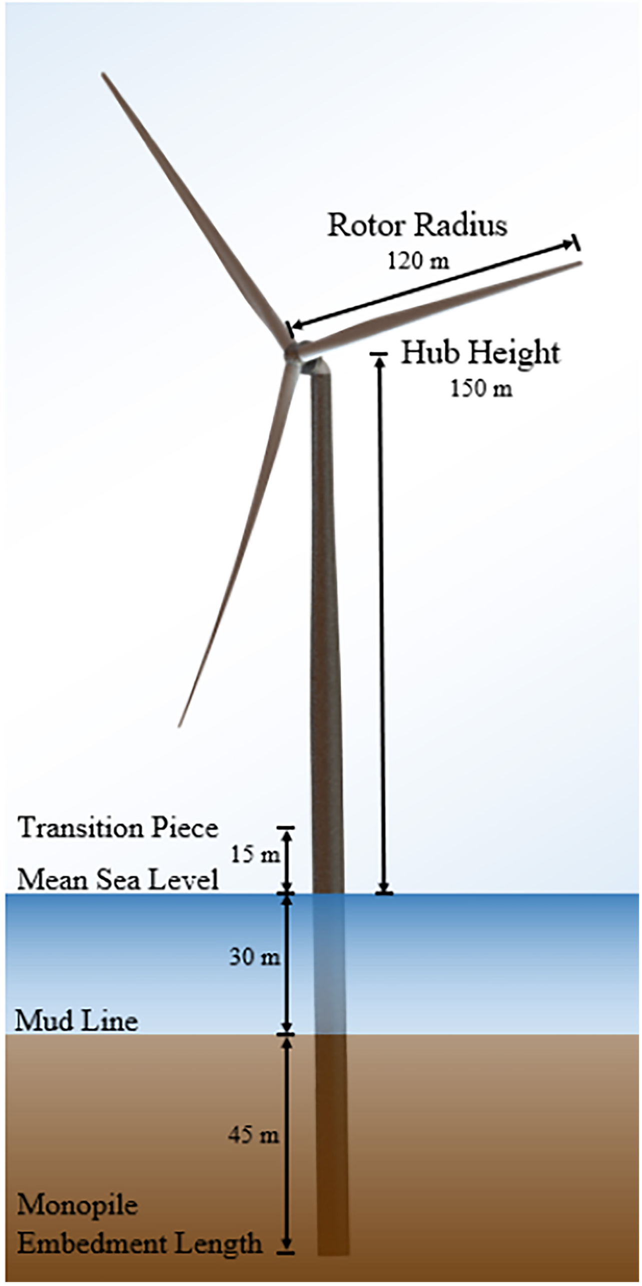


FIGURE 1 | The IEA 15-MW reference wind turbine.

TABLE 1 | Key parameters of the IEA-15-MW wind turbine.

Parameter	Value
Hub height	150 m
Rotor diameter	240 m
Cut-in wind speed	3.00 m/s
Rated wind speed	10.59 m/s
Cut-out wind speed	25.00 m/s
Minimum rotor speed	5.00 rpm
Maximum rotor speed	7.56 rpm

10991824, 2025, 6, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/we.70027 by Breifni Fitzgerald - Trinity College Dublin , Wiley Online Library on [16/05/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

wind speed vectors over a fixed two-dimensional vertical grid that remains stationary throughout the simulation. To create wind fields adapted to specific wind turbine configurations and site conditions, TurbSim offers adjustable parameters related to turbine structure and meteorological conditions that define the wind speed field. For an in-depth discussion of these parameters, the readers are referred to the TurbSim user guide [34].

This wind turbine is modelled using the methodology presented in the previous paragraph. The multibody model developed in this study has been benchmarked against the National Renewable Energy Laboratory's (NREL) multibody dynamic wind turbine model, OpenFast [35], a widely used open-source model. OpenFast has been benchmarked against the experimental results and thus very closely represents the actual behaviour of wind turbines. Benchmarking the numerical model in this study against OpenFast is a way to ascertain the accuracy of our model. The results from the validation exercise are presented in Figure 2.

A good agreement between OpenFast and the numerical model derived in this study is observed. This model is further used to generate the data required for training the machine learning model.

4 | Surrogate Model Development

This study aims to estimate the time-dependent deformation patterns at the blade tip utilising a machine learning technique. The LSTM model is selected to capture the force-deformation dynamics based on training data generated using the numerical model presented in Section 3. The choice of the LSTM model is based on the fundamental architecture of the algorithm, which effectively models time dependency by selectively retaining and discarding information through multiple gates present in its unit. These gates consist of fixed and recurrent weights that are adjusted to appropriately weigh the input data at a given time step and the hidden state from the previous time step. Such models have demonstrated exemplary performance in simulating time series responses of numerical models [36–38]. Readers interested in LSTM cell architecture and its core mathematical formulation are referred to the seminal work of [39].

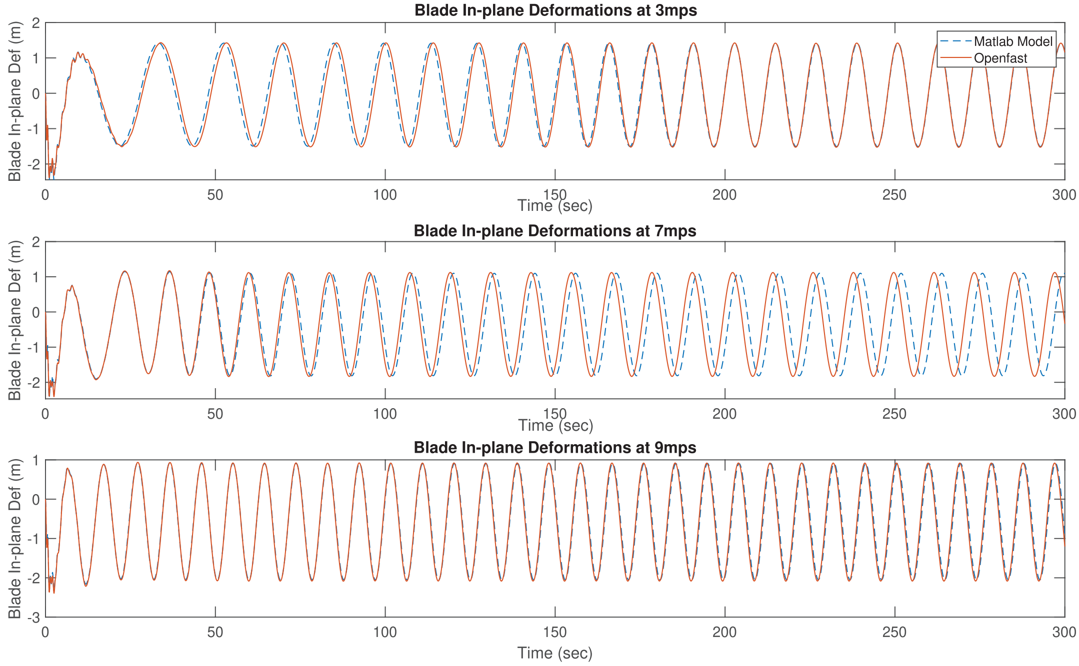
The LSTM model learns the force-deformation dynamics from training data; thus, to ensure its generalisability and accuracy, it is important to simulate the force time history covering a range of potential values and deformation patterns encountered during operations. In a wind turbine, the primary dynamic forces result from aerodynamic loads acting on the system. Variations in other load components, like gravity loads, are relatively consistent. It is assumed that the impact of gravity forces will produce a predictable deformation pattern, which the algorithm will capture. Among various operational parameters, rotor speed, generator azimuth angle and blade pitch angle are considered as input features to capture the deformation response. These features are chosen because they significantly influence the deformations experienced by the wind turbine blade and are easily measurable during operation, supporting the model's application as a virtual sensor. In addition to these operational quantities, aerodynamic forces are considered as input features for predicting deformation.

The aerodynamic forces acting on the blade are a function of the turbulent inflow and the aerodynamic properties of the wind turbine blade. In this study, the blade geometry and aerodynamic characteristics are assumed to remain constant during the operation. Therefore, the variation in aerodynamic forces can be captured by generating various environmental conditions. [22] has studied the effect of environmental variables on aerodynamic loading, especially the fatigue response. It has been found that the major factors governing the aerodynamic loads are the mean wind speed, turbulence intensity and the shear exponent. Also, these variables uniquely define an IEC site class, so selecting these input variables ensures that this model will be applicable to all the different IEC site classes. While sampling these variables, mean wind speed is chosen as the only independent input variable, and the turbulence intensity (TI) and shear exponent (α) are modelled as the dependent variable of mean wind speed. The mean wind speed values are first generated using the Sobol sequence and using the equations presented in Table 2, the bounds for TI and α are determined. The random samples for these variables are simulated following these bounds using the Sobol sequence. The sampled values of these variables are passed as input to the TurbSim application, which generates a unique wind flow pattern for each combination of these values. A total of 50 unique combinations of these environmental variables are generated using the Sobol sequence. During the operation of wind turbines, the major source of uncertainty is contributed by the stochastic nature of wind inflow. To take this into account, a total of 15 random seeds are considered for each combination of the environmental variables. Since the environmental variable defines the statistical properties of the wind field, random seeds are used to produce different turbulence structures having the same statistical properties. Generating multiple random seed ensures that uncertainty due to the turbulent nature of the inflow is taken into account during the model training. To this end, out of the 15 turbulent wind fields generated for each combination of environmental variables, nine are used for training, and six are used for testing and validation.

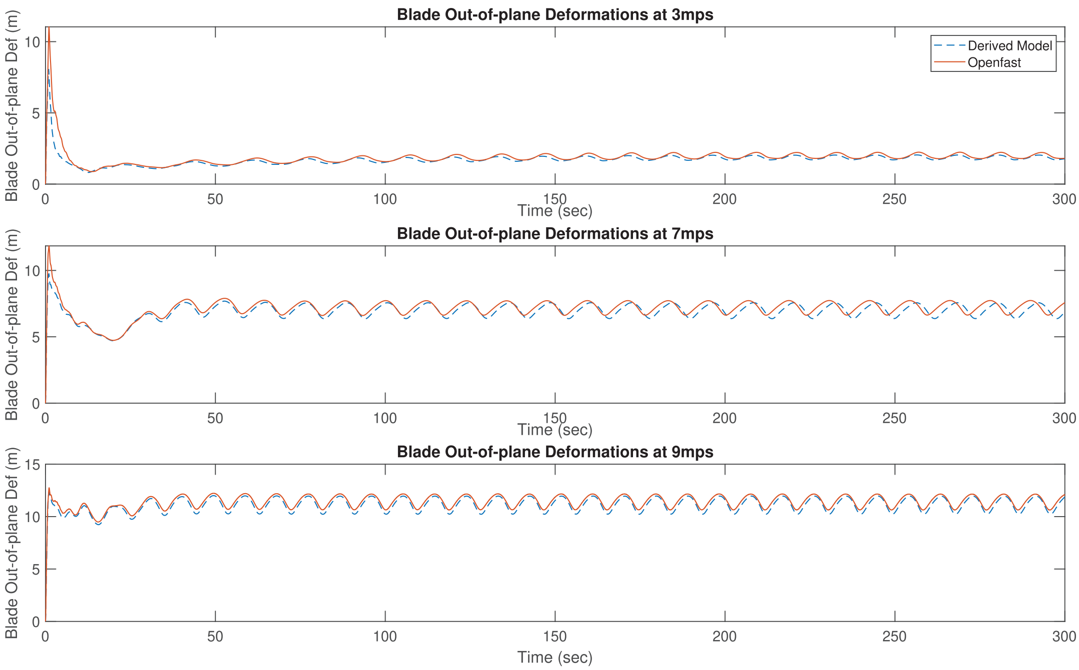
5 | Feature Selection

Optimum feature identification is of paramount importance in developing a robust machine learning surrogate. A carefully selected feature set is crucial in improving model performance, reducing chances of overfitting and enhancing the interpretability of the model. By selecting a subset of informative features, computational resources are conserved, and model complexity is reduced, leading to more efficient and effective machine learning models. Two primary approaches for feature identification are feature extraction and feature selection. Feature extraction involves manipulating available input data through various operators to derive a set of features in alternative parameter spaces. Techniques such as principal component analysis (PCA) [40], linear discriminant analysis (LDA) [41] and Autoencoders [42] are commonly employed in feature extraction. For features identified through feature

10991824, 2025, 6, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/we.70027 by Breifni Fitzgerald - Trinity College Dublin , Wiley Online Library on [16/05/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License



(a) Blade in-plane deformations



(b) Blade out of plane deformations

FIGURE 2 | Model verification: Comparison of blade response.

extraction methods that rely on specific operations or calculations involving input data from all channels, it is necessary to record data from all input channels after model deployment. Conversely, feature selection involves ranking available input features based on a scoring criterion within the same coordinate space. Based on the procedure used for feature ranking, these methods are classified into filter-type and wrapper-type methods. Filter-type methods utilise statistical properties such as correlation coefficient, covariance and chi-square statistics to evaluate the relevance of each input feature with reference to the target variable. The input features are ranked based on their relevance. Conversely, wrapper-type methods integrate

10991824, 2025, 6, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/we.70027 by Breiffni Fitzgerald - Trinity College Dublin, Wiley Online Library on [16/05/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

TABLE 2 | Uncertain environmental variables and their limits.

Variable	Lower bound	Upper bound
Mean wind speed (m/s)	3	25
Turbulence intensity (%)	0.025	$\frac{0.18}{U}\left(6.8+0.75\cdot U+3\left(\frac{10}{U}\right)^{2}\right)$
Shear exponent	0	$\alpha_{ref,UB}+0.4\left(\frac{R}{z}\right)\left(\frac{U_{max}}{U}\right)$

the machine learning model into the selection process, training it on various feature subsets to identify the subset yielding the best performance. While wrapper-type methods often yield better results, these methods are computationally intensive due to the need to train the model for each feature subset. This study employs a hybrid approach combining filter and wrapper methods to achieve optimal input feature combinations. This study uses a mutual information-based minimum redundancy and maximum relevance (MRMR) algorithm to rank the features based on their relevance. Based on a thresholding criterion, the first few features are selected. A wrapper-type approach is implemented on these selected features to arrive at the feature subset yielding the best performance. The hybrid approach of combining filter and wrapper methods leverages the strengths of both methods, exploiting statistical relationships between features while also considering the performance of the machine learning model. By balancing computational efficiency with model performance, the hybrid approach enables the identification of optimal feature combinations at a reasonable computational cost. The details of this method are presented in the next section.

5.1 | MRMR Algorithm

MRMR algorithm [43] selects a subset of input features such that each selected feature provides unique information about the target variable. This method is particularly useful in cases where a large set of correlated input features is available. In this study, mutual information is used as the criterion to quantify the relevance of an input feature with respect to the target variable. Mutual information is based on the concept of entropy, which measures the uncertainty associated with a random variable. The entropy of a discrete random variable X with probability mass function $p(X)$ is given by Equation (6).

$$H(X)=-\sum_{x}p(x)\log p(x) \tag{6}$$

Here X is used to represent random variable and x is used to represent its realisation. Similarly, conditional entropy is a measure of average uncertainty present in a variable given the outcome of another random variable. Conditional entropy is defined in Equation (7).

$$H(X|Y)=-\sum_{x,y}p(x,y)\log p(x|y) \tag{7}$$

where $p(x,y)$ is a joint distribution of random variables x and y and $p(x|y)$ is a conditional distribution of x given y. Using these concepts, intuitively, mutual information quantifies how much information one random variable gives about another. The formal definition of mutual information between two random variables X and Y is defined in Equation (8).

$$I(X,Y)=\sum_{x,y}p(x,y)\log\frac{p(x,y)}{p(x)p(y)} \tag{8}$$

Mutual information can be expressed in terms of entropy as presented in Equation (9).

$$I(X,Y)=H(X)-H(X|Y) \tag{9}$$

In the context of feature selection, assuming X represents an input feature and Y represents the target variable, the MRMR algorithm's aim is to identify a subset of input features X_s which maximises the mutual information between X_s and Y and minimises the mutual information between the components of X_s. Mathematically, MRMR algorithm is defined in Equation (10).

$$\max X_s=\left\{\frac{1}{|X_s|}\sum_{i=1}^{|X_s|}I\left(X_{s_i},Y\right)-\frac{1}{|X_s|(|X_s|-1)}\sum_{i=1}^{|X_s|}\sum_{j=1,_{j\neq i}}^{|X_s|}I\left(X_{s_i},X_{s_j}\right)\right\} \tag{10}$$

where $|X_s|$ represents the number of features selected in subset X_s, $I\left(X_{s_i},Y\right)$ represents the mutual information between each individual feature X_{s_i} and target variable Y and $I\left(X_{s_i},X_{s_j}\right)$ represents the mutual information between pairs of selected features in subset X_s, with $i\neq j$ to avoid calculating mutual information of a feature with itself. The MRMR algorithm iteratively selects features based on their relevance to the target variable and their redundancy with respect to other selected features, ultimately resulting in a subset of input features that maximise predictive performance while minimising redundancy.

For the implementation in this study, *fsrmrmr* feature in Matlab is used. In this algorithm in Matlab®, each feature is assigned a score using the mutual information quotient (MIQ) presented in Equation (11), such that the feature with higher score corresponds to more relevance with the target variable.

$$MIQ_x=\frac{I(x,y)}{\frac{1}{|x|}\sum_{z\in S}I(x,z)} \tag{11}$$

In this algorithm, a reduction in the feature score represents improved confidence in the feature selection. Using this property, the difference in scores of the features is computed and the number of features with a threshold of reduction in feature score are selected for the further process of model training. Using this approach, a small subset ($|X_s|$) of features is identified. However, these features are selected based on the statistical properties and may not lead to the optimum performance. To ensure an optimum performance, the subset of feature is evaluated using wrapper-type method to select the final

10991824, 2025, 6, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/we.70027 by Breifni Fitzgerald - Trinity College Dublin , Wiley Online Library on [16/05/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

optimum subset most suitable for the specific machine learning algorithm considered for the study. Two possible ways of wrapper type method based on the combination of subsets are presented in the next section.

5.2 | Recursive Feature Selection

The concept of the recursive feature selection method is briefly discussed in Section 1. These iterative techniques operate by systematically exploring the feature space, to identify an optimal subset of input features that maximise the predictive performance of the machine learning model. The starting point can be either an empty set or the complete set of features, and the selection process operates through sequential addition (forward selection) or removal (backward elimination) of features. Forward selection, also known as recursive feature addition (RFA), starts with an empty feature set (denoted as X_s) and iteratively incorporates features based on their individual relevance to the target variable (Y). This relevance is assessed by evaluating the impact of each feature on a chosen performance metric, such as the root mean squared error (RMSE) employed in this study. The feature leading to the lowest RMSE is then added to X_s. This process continues by evaluating all remaining features in combination with the chosen ones, exploring the combination that further enhances the model performance. These steps are repeated iteratively until a predefined stopping criterion is met, such as reaching a desired number of features or achieving a satisfactory level of model performance. The method terminates when adding extra features to X_s does not improve the performance. Conversely, backward elimination, also known as recursive feature elimination (RFE), starts with the complete set of features (X_s) and iteratively removes the least informative features such that the removal does not significantly compromise model performance. RFE identifies and removes the least impactful members, ultimately retaining the core group of features that delivers optimal results. By formalising these concepts, the forward selection process can be expressed as $X_s = X_{max}$, where X_s denotes the evolving feature set, and X_{max} symbolises the set of most impactful features. In contrast, RFE's iterative refinement is captured by$X_s = X_s - X_{min}$, signifying the removal of the least informative feature set (X_{min}) from the complete set X_s. While RFS methods are effective in identifying an optimal subset leading to a robust feature identification, a major limitation of these models is the computational efforts required. Using RFA, the number of simulations, N_{sim}, required to select n number of features from N available features is given by Equation (12)

$$N_{sim} = \frac{n(2N - n + 1)}{2} \tag{12}$$

The computational cost increases with increase in available features 'N' and final required number of features 'n'. In this context, pairing this method with MRMR algorithm and operating on a subset X_s, $|X_s| << |X|$ can significantly bring down the computational cost by reducing the total number of features 'N' from $|X|$ to $|X_s|$. One of the major disadvantages of this method is that this method focuses on the individual feature importance and does not take into account feature interaction, potentially leading to sub-optimal solutions. This limitation can lead to a suboptimal performance for nonlinear systems where feature interactions can significantly affect the system dynamics. To further investigate the impact of neglecting feature interactions, the performance of RFA on a reduced feature set (X_s) will be compared with an exhaustive evaluation of all possible feature combinations within X_s. This will lead a total number of simulations given by Equation (13)

$$N_{sim} = \sum_{k=1}^{n} C(N,k);\ C(N,k) = \frac{N!}{k!(N-k)!} \tag{13}$$

While exhaustive evaluation could be computationally intensive for larger datasets, pairing this method with MRMR reduces the total number of simulations required. Further, this approach enables a direct comparison and highlights the potential performance loss due to neglecting interactions while remaining computationally feasible. The findings of our study demonstrate the broader applicability of RFA to nonlinear systems.

6 | Hyperparameter Tuning and Model Training

LSTMs offer exceptional flexibility and power in sequential data modelling, but their true potential hinges on carefully selecting architectural settings known as hyperparameters. Hyperparameters are key architectural parameters of a machine learning model defined prior to model training, directly affecting the model performance and generalisation ability. Effective hyperparameter tuning is, therefore, paramount for realising an efficient and well-performing LSTM model. The hyperparameter tuning process aims to identify the optimal combination of hyperparameter values, potentially leading to greater accuracy, faster convergence, and enhanced generalisation of unseen data. Mathematically, this process is defined in Equation (14), where θ represents a set of hyperparameters and $h(\theta)$ represents the objective function being minimised, which is a model performance metric such that minimising the objective function results in greater model performance.

$$\theta^* = \underset{\theta \in \Theta}{argmin}\, h(\theta) \tag{14}$$

In Equation (14), Θ represents the domain of possible values the hyperparameters can assume. For each set of hyperparameters, the model must be trained to compute the value of the objective function. Since this domain usually represents a large set of points and model training is computationally intensive, defining the boundaries of this domain and selecting a suitable optimisation algorithm is of paramount importance to ensure efficient hyperparameter optimisation. The study leverages the *Bayesian optimisation* technique as implemented in Matlab. This approach constructs a probabilistic surrogate model of the objective function, guiding informed decisions about generating new parameter combinations within a predefined range. The Bayesian optimisation process continually updates the surrogate probability model after each objective function evaluation. This sequential evaluation of the hyperparameter ensures that the optimisation learns

10991824, 2025, 6, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/we.70027 by Breifni Fitzgerald - Trinity College Dublin , Wiley Online Library on [16/05/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

from the previous objective function evaluation and uses that information to guide the optimisation process to track the minimum value in a minimum number of model evaluations. The optimisation process employs an acquisition function, '*expected improvement plus*', strategically balancing exploring new regions with exploiting promising regions identified thus far.

Prior to hyperparameter tuning, a two-layer LSTM architecture was established through preliminary trials. This configuration was chosen based on its initial performance and suitability for capturing temporal dependencies within the data. This study considers the number of hidden units in each LSTM layer, drop-out probability and batch size in the hyperparameter tuning process. The number of hidden units in each LSTM layer directly governs the model learning performance and its capability to capture the relationship between input and output variables. The drop-out probability refers to the percentage of data that would be randomly set to zero; this parameter governs the model's generalisation capability and helps avoid overfitting. Batch size, controlling the data fed per training iteration, presents a trade-off between frequent updates, faster training with less memory (smaller batches) and smoother gradient updates with higher memory requirements (larger batches). The hyperparameter tuning process chooses an optimum combination of these parameters to achieve the best performance (minimum RMSE). A total of 40 hyperparameter combinations are evaluated to arrive at the optimum hyperparameter combination, leading to a minimum RMSE in the validation data set.

Having established the optimal architecture, the LSTM model is trained using the adaptive moment estimation (Adam) algorithm [44]. The Adam optimiser tackles the limitations of Stochastic Gradient Descent by incorporating decaying averages of past gradients, enabling it to escape local minima and converge towards the global optimum. A plateauing validation loss-based learning rate schedule is developed to further fine-tune the learning process. In this approach, the learning rate was reduced by a factor of δ if the validation loss remained stagnant for n_p consecutive epochs, ensuring progress towards the minimum. A minimum learning rate was also set to prevent stagnation and overly prolonged training. Interestingly, after m reductions in the learning rate, if performance plateaued again, the learning rate is increased by a factor of ξ. This step ensures that the model does not get stuck in saddle points, which are plateaus in the loss landscape that are not the global minimum. Based on empirical observations, this strategy often produces superior results compared to a constant learning rate approach. In addition, during training, especially in deep networks, gradients can become very small, slowing down the learning process. A controlled increase in learning rate can counteract this issue and accelerate progress, particularly when approaching the minimum. The performance of the presented learning rate schedule strategy depends on the values chosen for m, n_p, ξ, and δ for this study the values considered of 5, 10, 1.5 and 0.9 give good results. These values should be selected based on the nature of the problem and the training data. The complete process of developing an LSTM surrogate for sequence-to-sequence regression is presented in Algorithm 1.

Algorithm 1 Machine learning model development algorithm

Input: Raw data (X, y) **Output:** Optimised machine learning model

1. Data Preprocessing
Outlier Removal: Apply a suitable outlier detection and removal technique
Data Downsampling: Use an appropriate downsampling technique to reduce the quantity of the data
2. Feature Selection
Normalisation: Normalise features so that feature ranking is not impacted by the absolute value of the data
2.1 MRMR Feature Ranking:
for $X_i \in X$ **do**
    Calculate mutual information $I(X_i, y)$ between X_i and y using Equation (8).
    for $X_j \in X$ **do**
        if $X_i \neq X_j$ **then**
            Calculate mutual information $I\left(X_i, X_j\right)$ between X_i and X_j using Equation (8).
        end if
    end for
    Calculate MIQ score MIQ_x for X_i using Equation (11).
end for
Select a subset X_s of features based on the highest MIQ scores, accounting for a drop in scores for confidence.
2.2 Initial Hyperparameter Tuning
Define the LSTM model architecture.
Select hyperparameters to tune
Perform initial hyperparameter tuning on X_s using *Bayesian optimisation.*
2.3 Wrapper-based Feature Selection:
Recursive Feature Addition (RFA):
$X_s^{temp} = \emptyset$
while improvement criteria met **do**
    for $X_i \in X_s$ **do**
        Evaluate model performance with X_s.
    end for
    Select the X_i that yields the best performance.
    Add the selected X_i to X_s^{RFA}.
end while
Exhaustive Feature Search:
for all possible subsets of X_s **do**
    Evaluate model performance with the subset.
end for
Select the subset X_s^{EPS} that yields the best performance.
Choose the final refined subset: $X_s^* = \operatorname{argmax}\left(\text{performance}\left(X_s^{RFA}\right), \text{performance}\left(X_s^{EPS}\right)\right)$.
3. Hyperparameter Tuning (optional)
Further refine hyperparameter tuning on the selected subset X_s^*.
4. Model Training
Train the model using the selected hyperparameters and Adam optimiser with a proposed learning rate scheduling approach.
5. Model Evaluation
Evaluate the performance of the trained model on unseen data using appropriate metrics.

Ensure: Optimised machine learning model ready for deployment

10991824, 2025, 6, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/we.70027 by Breifni Fitzgerald - Trinity College Dublin, Wiley Online Library on [16/05/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

7 | Numerical Results

Using the numerical model in combination with the training data generation approach, the numerical model is used to evaluate the wind turbine response for the turbulent wind inflow generated using Turbsim [34]. The time history of aerodynamic forces acting at each node of the wind turbine blade and the blade tip deformations are recorded. The aerodynamic forces are the input features, and the blade response represents the target variable.

Following the details presented in Algorithm 1, the first step was to perform the data preprocessing. Since this study uses the output coming from a numerical model, extensive preprocessing or outlier detection is not implemented. The only preprocessing carried out in this study is the downsampling of data. The numerical model uses a time step of 0.005 s to achieve the convergence; however, such fine resolution may not be required in the development of the surrogate model. The downsampling frequency should be chosen such that the dynamics of the system are sufficiently captured without omitting any significant frequency. In blade deformation response, the blade rotations and the low-frequency component of wind would contribute to the lower frequency response. As per the operating parameters of the 15-MW wind turbine, the cut-in rotor speed is 3 rpm which corresponds to 0.05 Hz. So the downsampled data should have a frequency more than 0.05 Hz to sufficiently capture the response at this frequency. From the model training and memory management perspective, downsampling reduces the total quantity of data, leading to faster training time and lesser memory requirements. The downsampling frequency also depends on the model deployment conditions, as for these models, the data should be fed at the same frequency as that of the data considered during training. Assuming a hub-mounted light

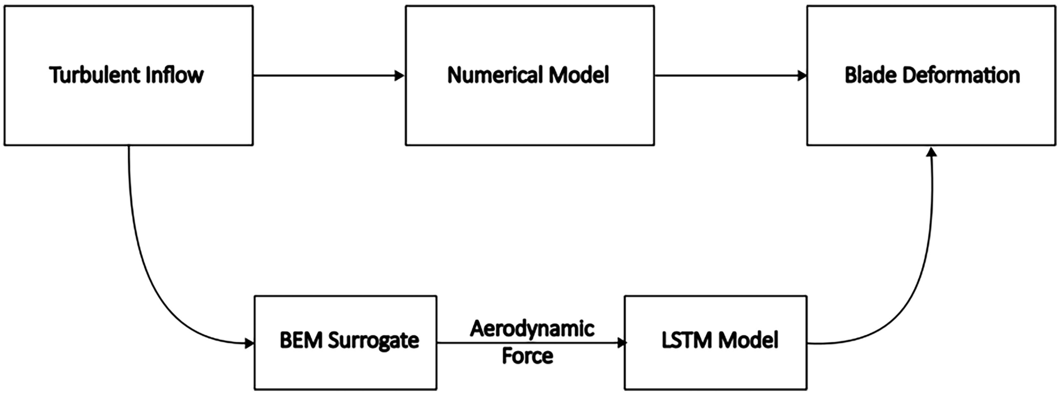


FIGURE 3 | Graphical representation of computing blade deformations from turbulent inflow using a surrogate modelling approach. Details of BEM surrogate can be found in Ref. [46].

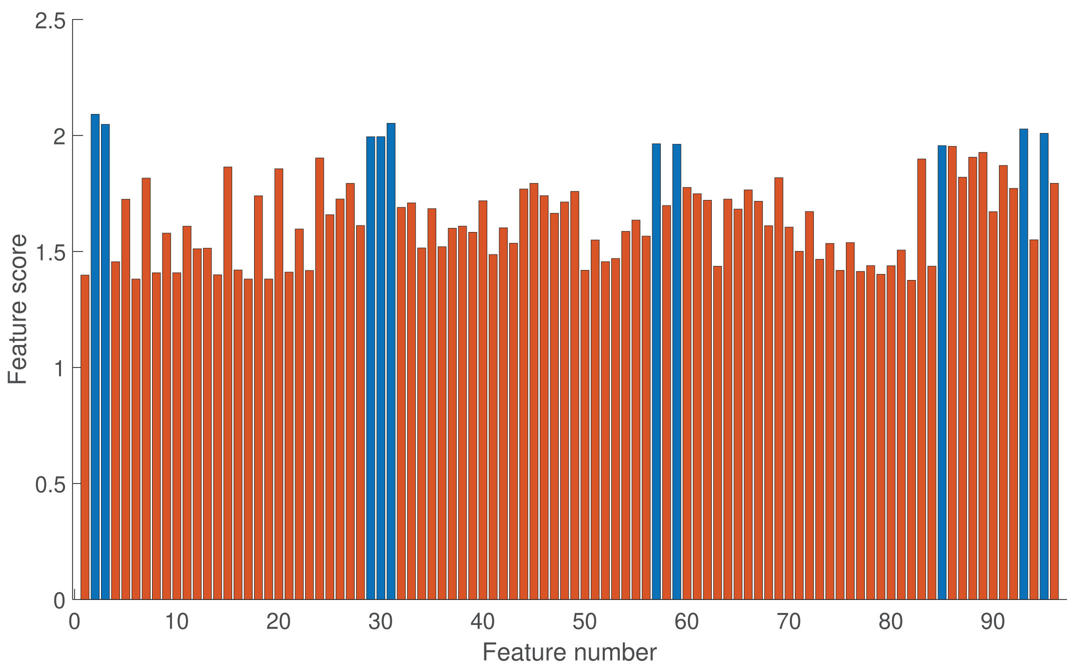
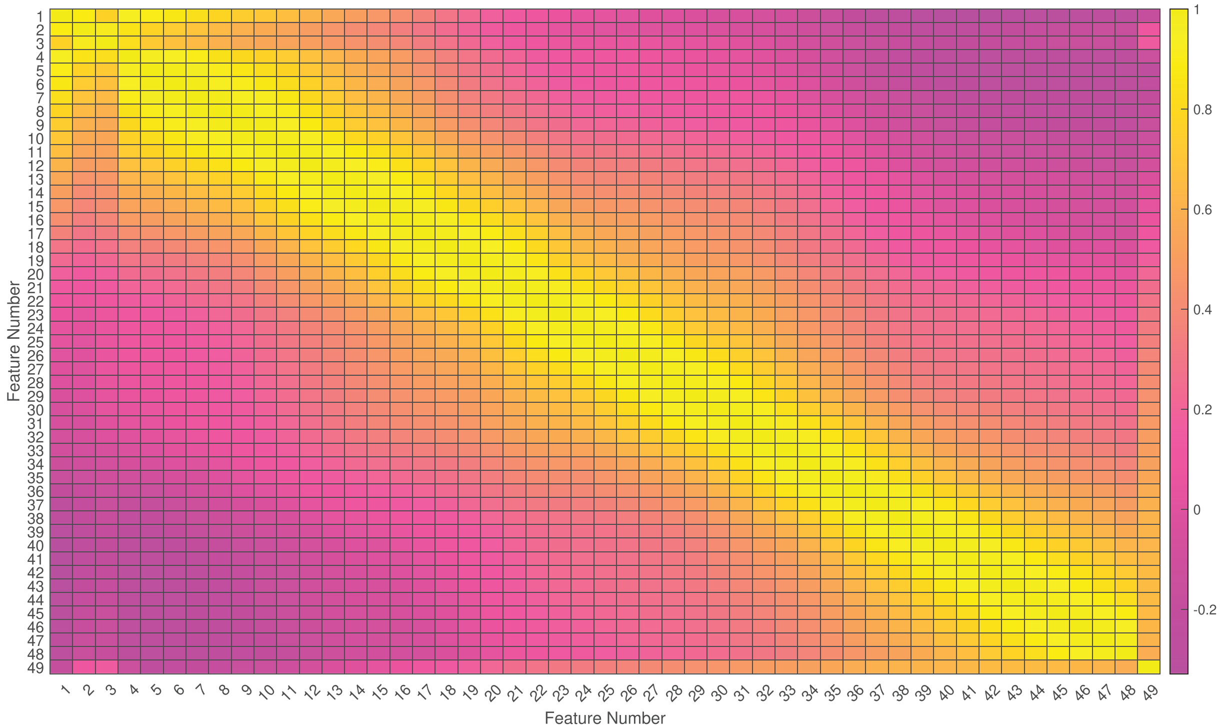
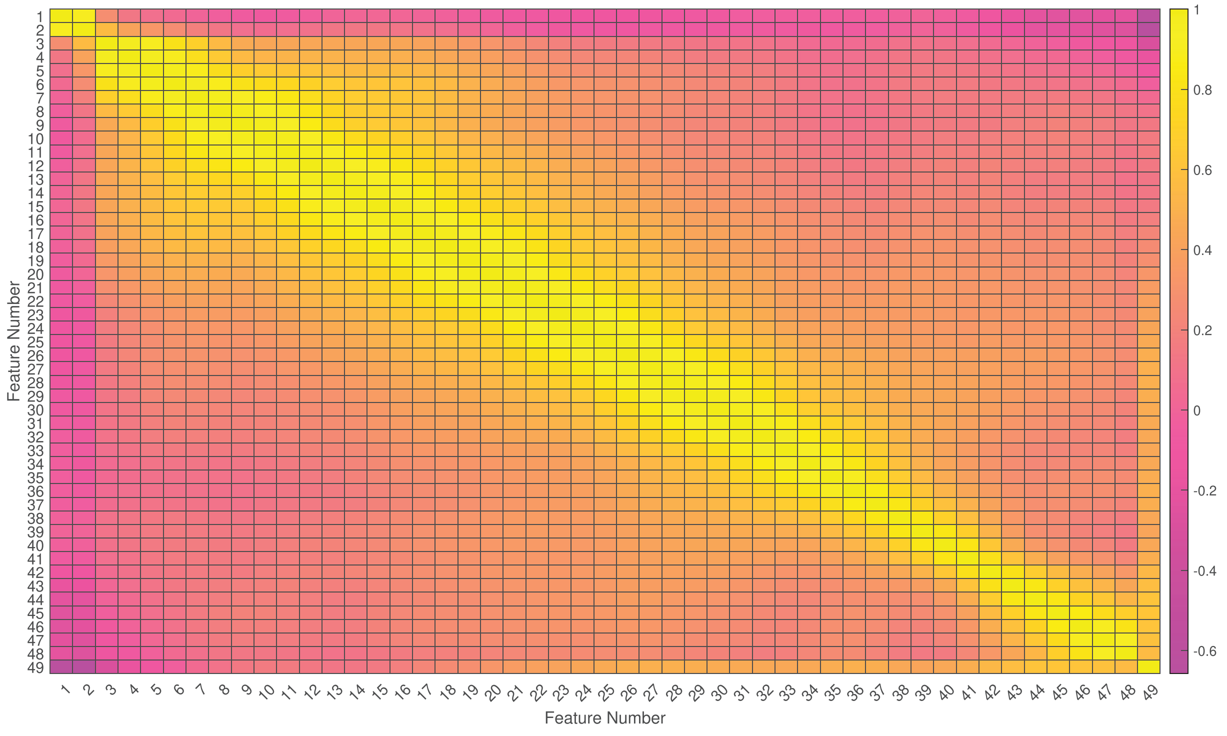


FIGURE 4 | Feature score prediction using MRMR algorithm.

10991824, 2025, 6, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/we.70027 by Breiffni Fitzgerald - Trinity College Dublin, Wiley Online Library on [16/05/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License



(a) Correlation matrix for forces normal to rotor plane



(b) Correlation matrix for forces tangential to rotor plane

FIGURE 5 | Correlation matrix showing high correlation in feature set.

detection and ranging (LIDAR) system to measure the incoming wind speed, the frequency of data used during training should match the frequency of data measurement. With reference to NREL's research on LIDAR wind speed measurements of evolving wind fields [45], the LIDAR sampling frequency is assumed to be 50 Hz. The data from the numerical model are sampled to 50 Hz.

To extend the applicability of the proposed model to an operational wind turbine with a LIDAR system, this study only uses variables as measurable by the LIDAR system. The input features for the LSTM model are regarded as the aerodynamic forces. Since these forces cannot be measured directly on site, it is necessary to derive a method to estimate them from the measured wind speed. Earlier, the authors proposed a surrogate-based

10991824, 2025, 6, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/we.70027 by Breiffni Fitzgerald - Trinity College Dublin, Wiley Online Library on [16/05/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

approach to estimate aerodynamic forces using wind speed data [46]. This approach uses a feed-forward neural network (FFNN) to predict the angle of attack using wind speeds, rotor speed and pitch angle. Given that the blade element momentum (BEM) equations are not explicitly dependent on the past values of the governing variables, an FFNN algorithm was utilised to develop the surrogate model, as FFNN operates on each input data point individually. A physics-informed neural network (PINN) and a data-driven neural network (DDNN) surrogate were constructed and validated under various operational conditions. While a detailed discussion of this algorithm is outside the scope of our study, interested readers can find more information in Ref. [46]. Based on this work, it is assumed that LIDAR data can be used with the neural network-based surrogate model of BEM [46] to predict aerodynamic forces. These predictions can then be integrated into the LSTM model developed in this study to predict blade tip deformations. Integrating these two surrogate models can allow a rapid estimation of blade tip deformations using LIDAR measurements. Figure 3 illustrates this process graphically.

Now, assuming that the aerodynamic forces at each blade node is known, the aim is to identify the subset of these data which can accurately predict the blade deformations. For the 15-MW wind turbine blade, a total of 50 such nodes are present, and at each node, the aerodynamic forces normal to the rotor plane and tangential to the rotor plane are recorded. For the purpose of surrogate model development, the first and the last blade nodes are not considered. Therefore, the complete set of aerodynamic forces considered for this study has a total of 96 input features. These features are passed through the MRMR algorithm to compute the MIQ quotient using Equation (11). The results of feature scores predicted by the algorithm are presented in Figure 4.

Using the drop in predicted feature score as the criterion, the MRMR algorithm presents 10 features which represent the subset X_s defined in Algorithm 1. The relevant nodes chosen by the MRMR algorithm are well distributed along the blade length, as shown in Figure 4. The forces at these nodes can capture a unique component of the blade deformation response as the nodes placed close to each other will exhibit a high correlation, leading to less unique information. The correlation matrix for the raw data and the data picked by the MRMR algorithm is presented in Figures 5 and 6, respectively. These figures show that the correlation between the features chosen through the MRMR algorithm is very low, implying that these features will provide unique information about the response.

Using the features identified using the MRMR algorithm, hyperparameter tuning is performed to determine the best combination of hyperparameters. The *Bayesian optimisation* algorithm in Matlab is used for this tuning process. The bounds of values for this optimisation are chosen based on the commonly adopted values in literature and are presented in Table 3.

In the next step, using the set of optimised parameters, the model is trained using the recursive feature addition (RFA) method. In all the model training exercises for the below-rated wind speeds, the azimuthal position of the blade and rotor speed are kept as the constant features of a subset X_s, while for above-rated operating conditions, the azimuthal position of the blade and the pitching angle are the constant features. For the RFA approach,

TABLE 3 | Hyperparameter bounds considered for tuning.

Hyperparameters	Lower bound	Upper bound
Number of hidden layers in LSTM	50	200
Drop-out probability	0.1	0.6
Batch size	1	15

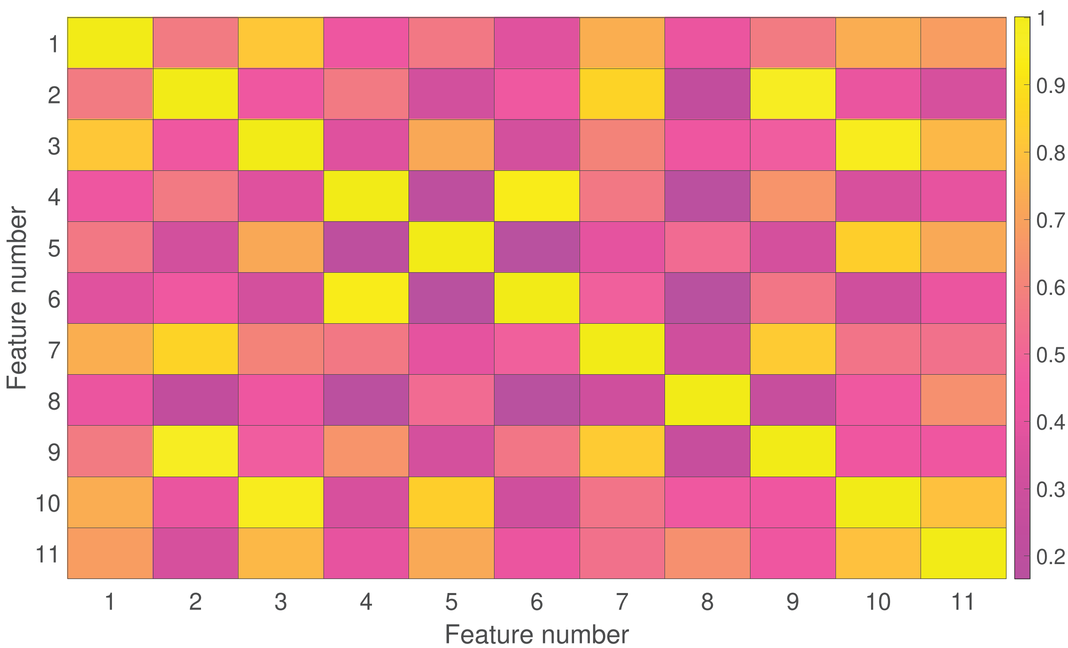


FIGURE 6 | Correlation matrix of features identified using MRMR algorithm.

10991824, 2025, 6, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/we.70027 by Breifni Fitzgerald - Trinity College Dublin , Wiley Online Library on [16/05/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

the average RMSE value across all the wind speeds is considered as the performance metric. This algorithm keeps adding the features until the addition of the features leads to a reduction in RMSE. The general process for this approach is presented in Algorithm 1. The performance of the model developed using the features identified using the MRMR algorithm is presented in Figures 7 and 8. Figure 7 shows a qualitative representation of model performance for different subsets of the input feature set identified by the MRMR algorithm, where the subset size is linearly increased by adding the features based on their feature scores such that the feature with the highest score is added first and then the next feature with the second highest score and so

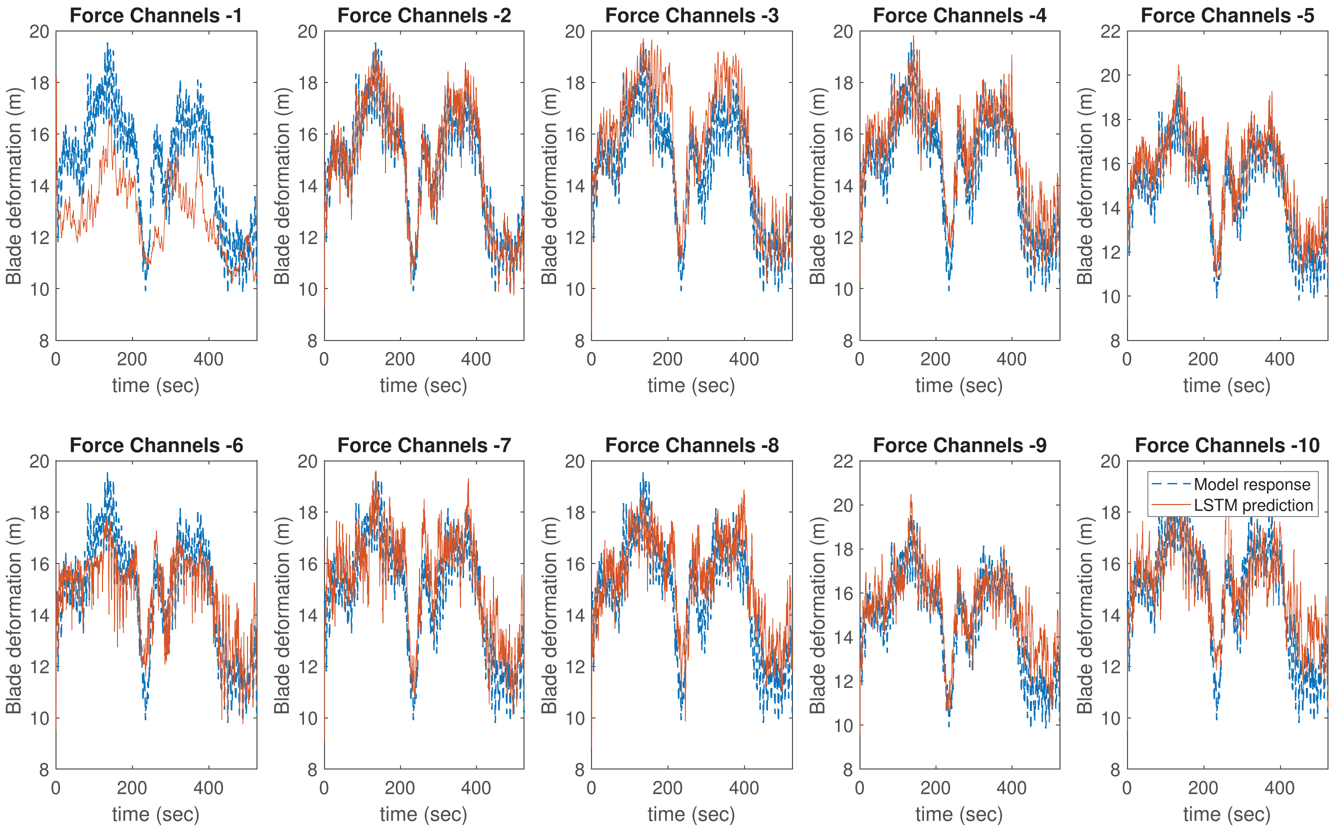


FIGURE 7 | Performance of LSTM model trained using features identified through MRMR algorithm.

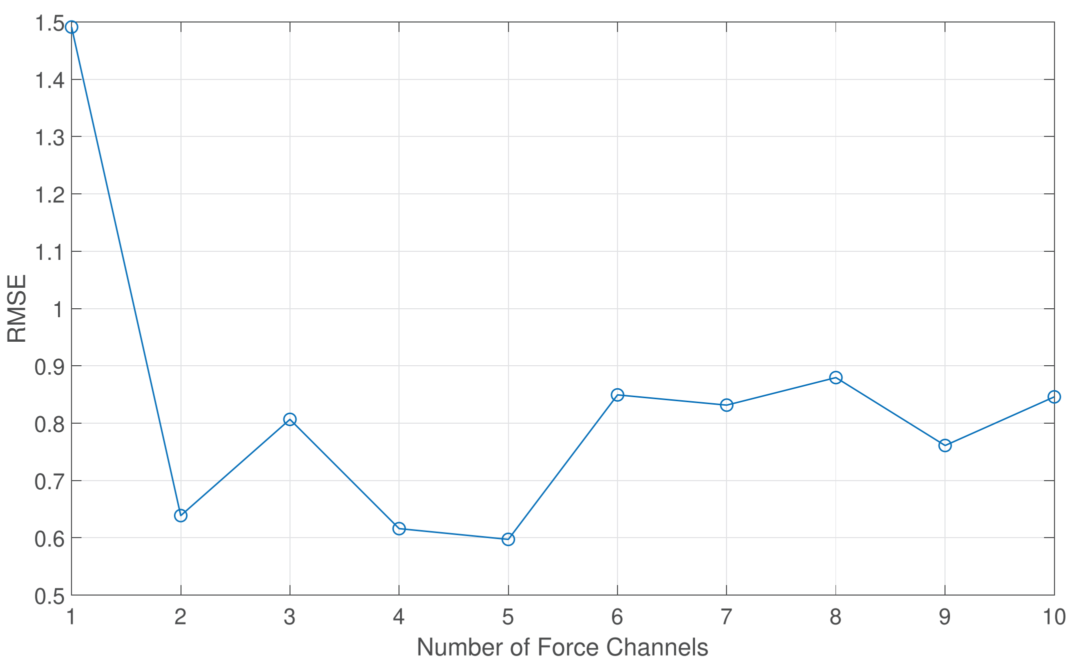


FIGURE 8 | Effect of increasing input feature space on model performance.

10991824, 2025, 6, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/we.70027 by Breifni Fitzgerald - Trinity College Dublin, Wiley Online Library on [16/05/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

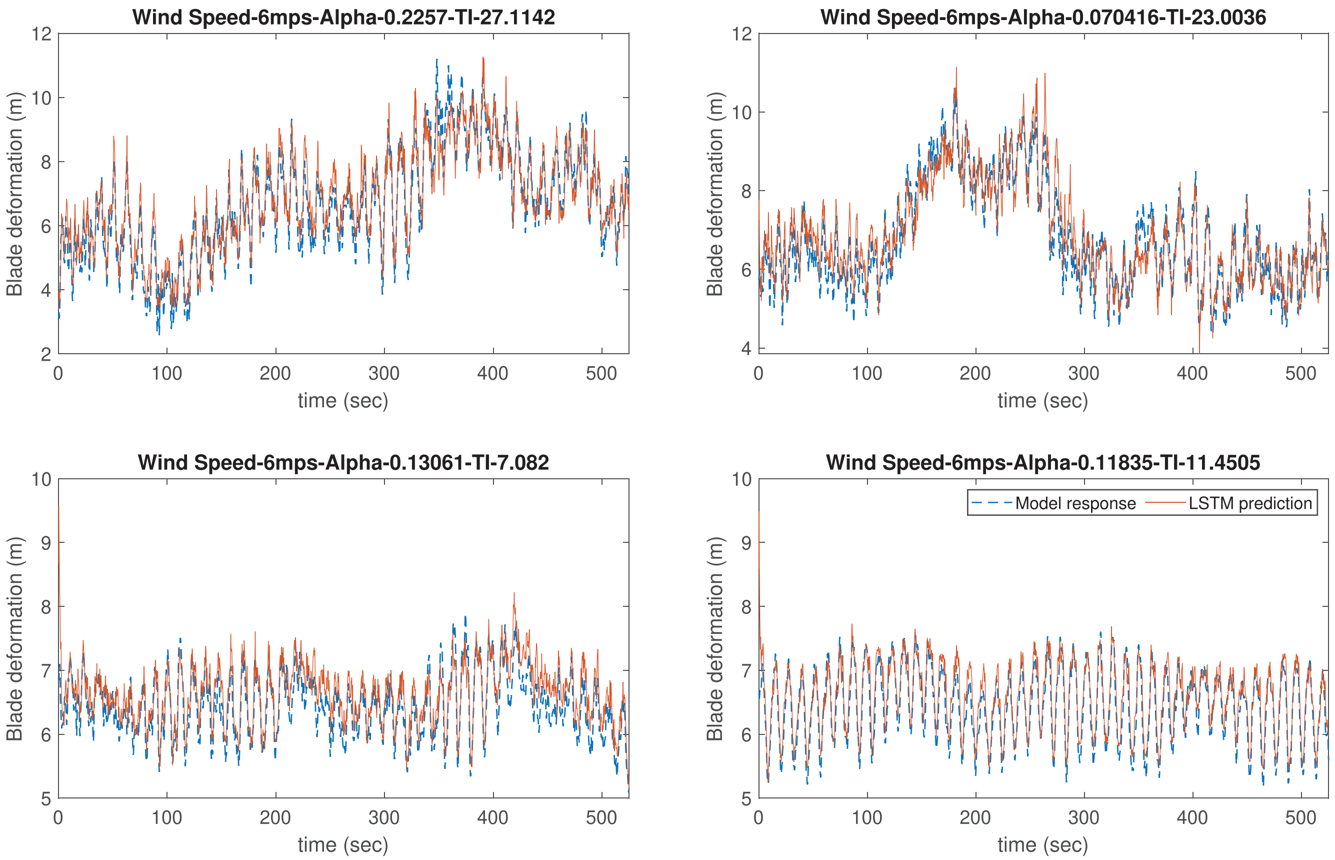


FIGURE 9 | Best performance achieved from LSTM model trained using features identified through MRMR algorithm across different stochastic realisations.

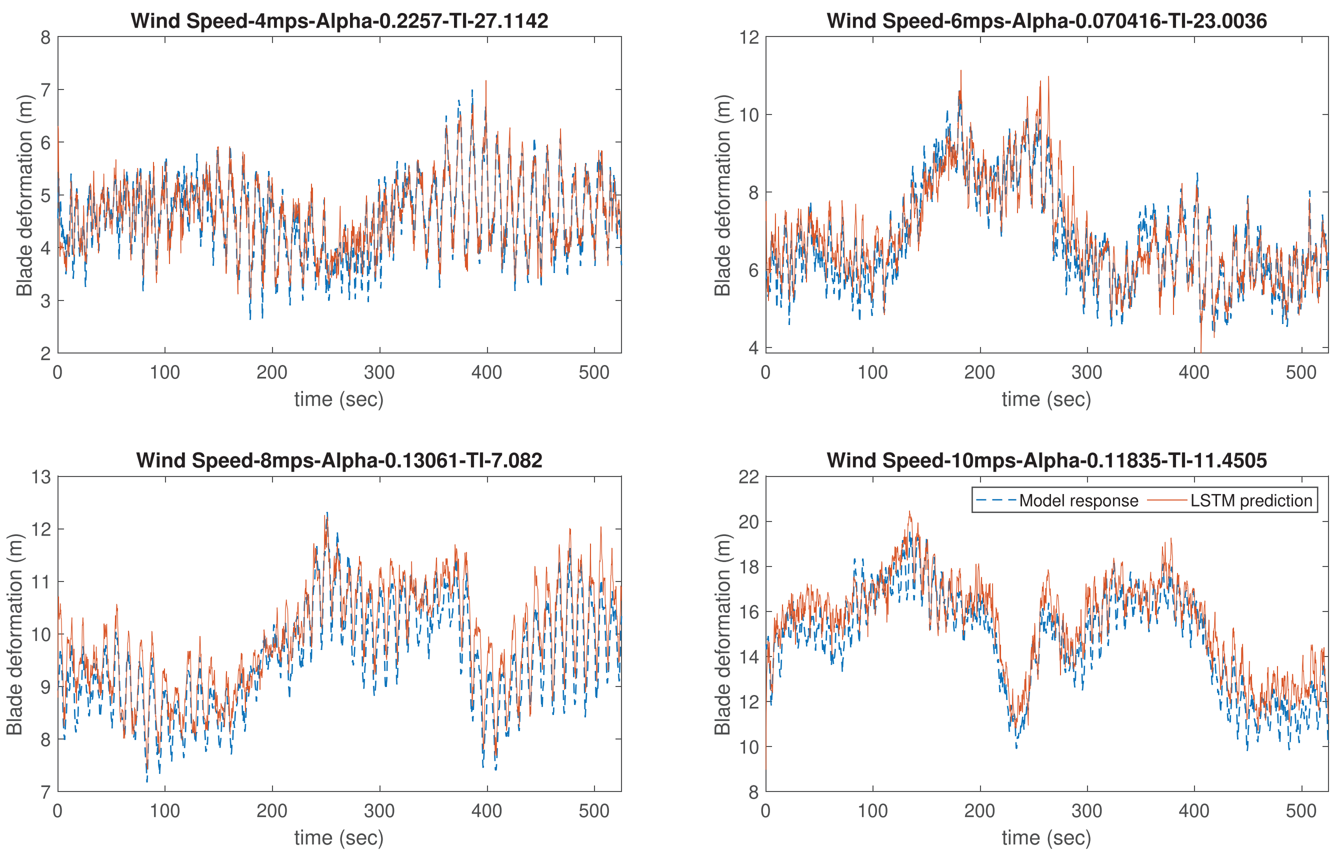
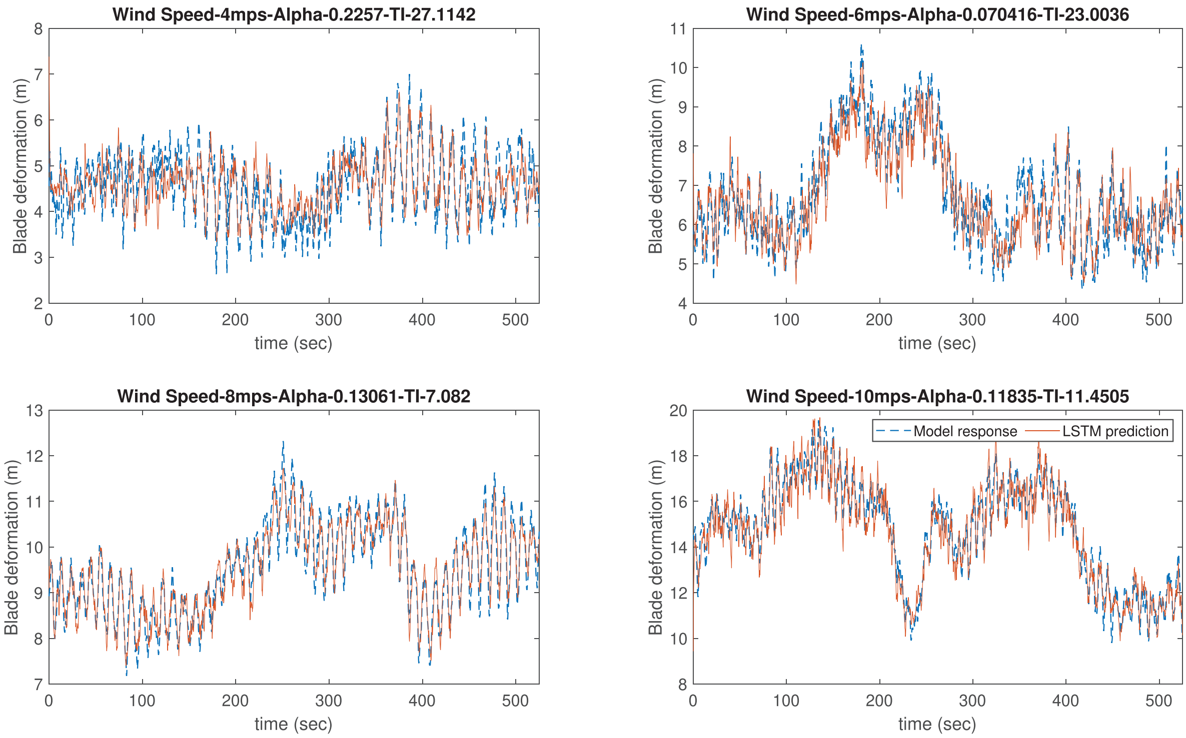
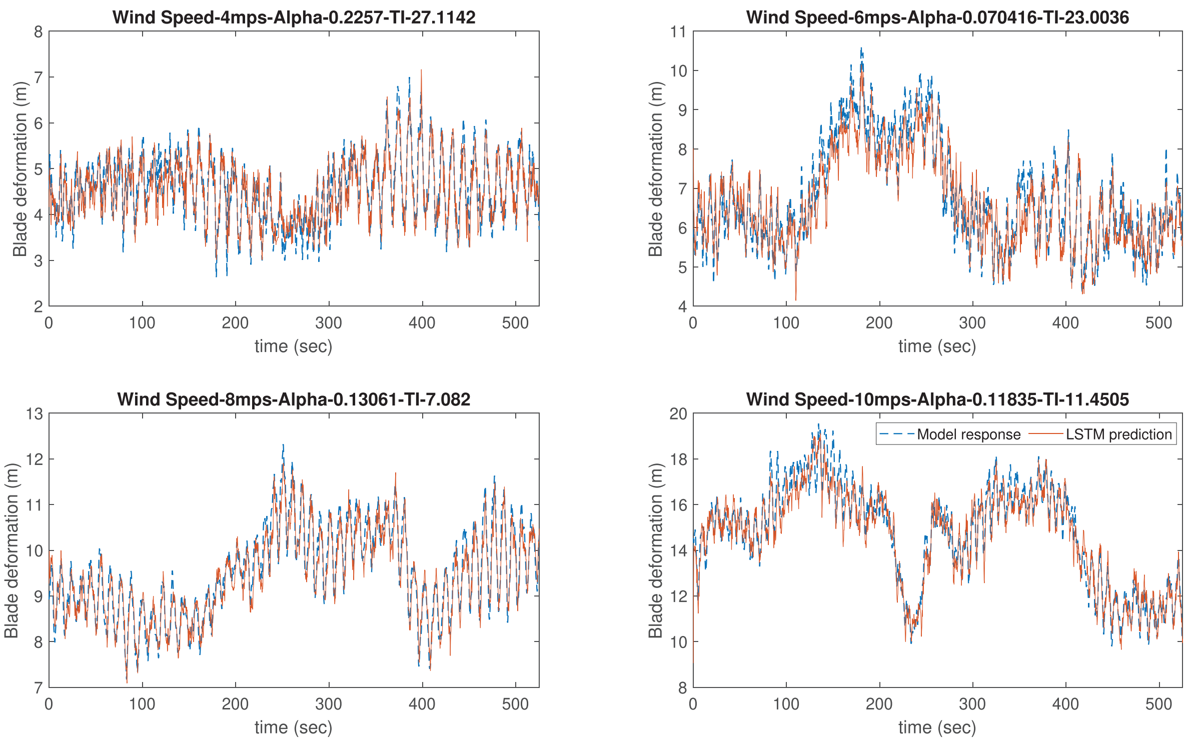


FIGURE 10 | Best performance achieved from LSTM model trained using features identified through MRMR algorithm across different mean wind speeds.

10991824, 2025, 6, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/we.70027 by Breifni Fitzgerald - Trinity College Dublin, Wiley Online Library on [16/05/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License



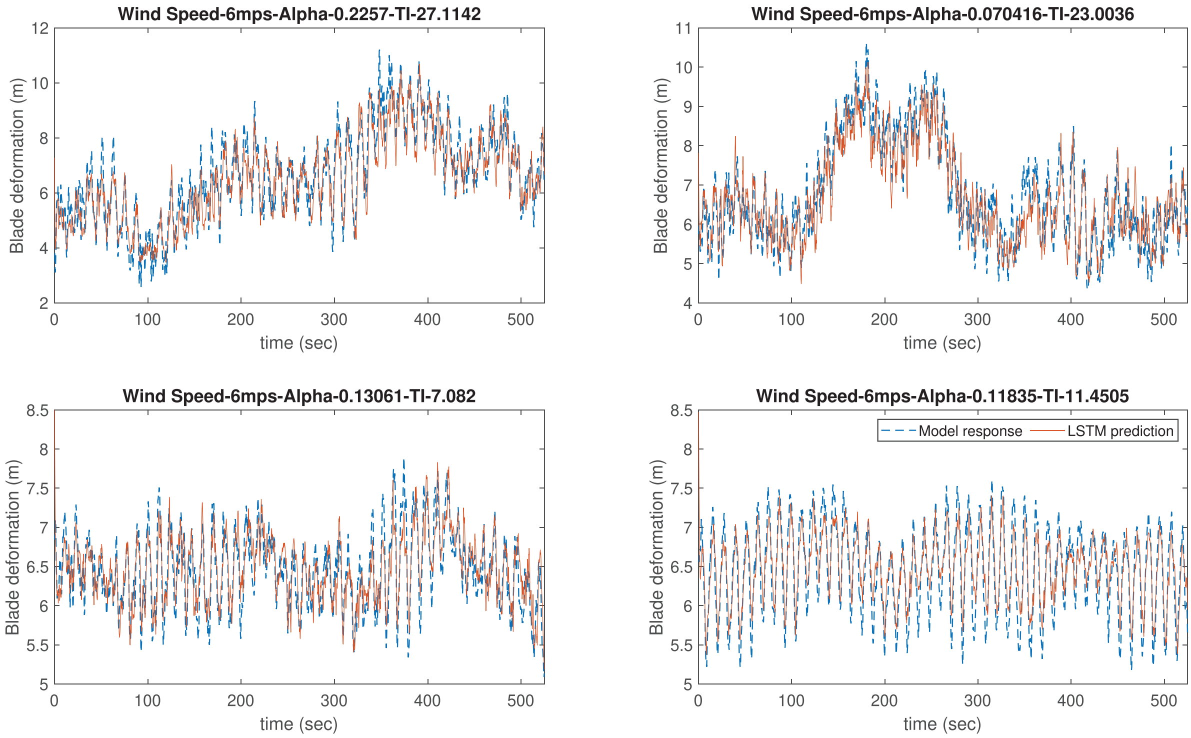
(a) Predictions of LSTM model trained using information from 1 node



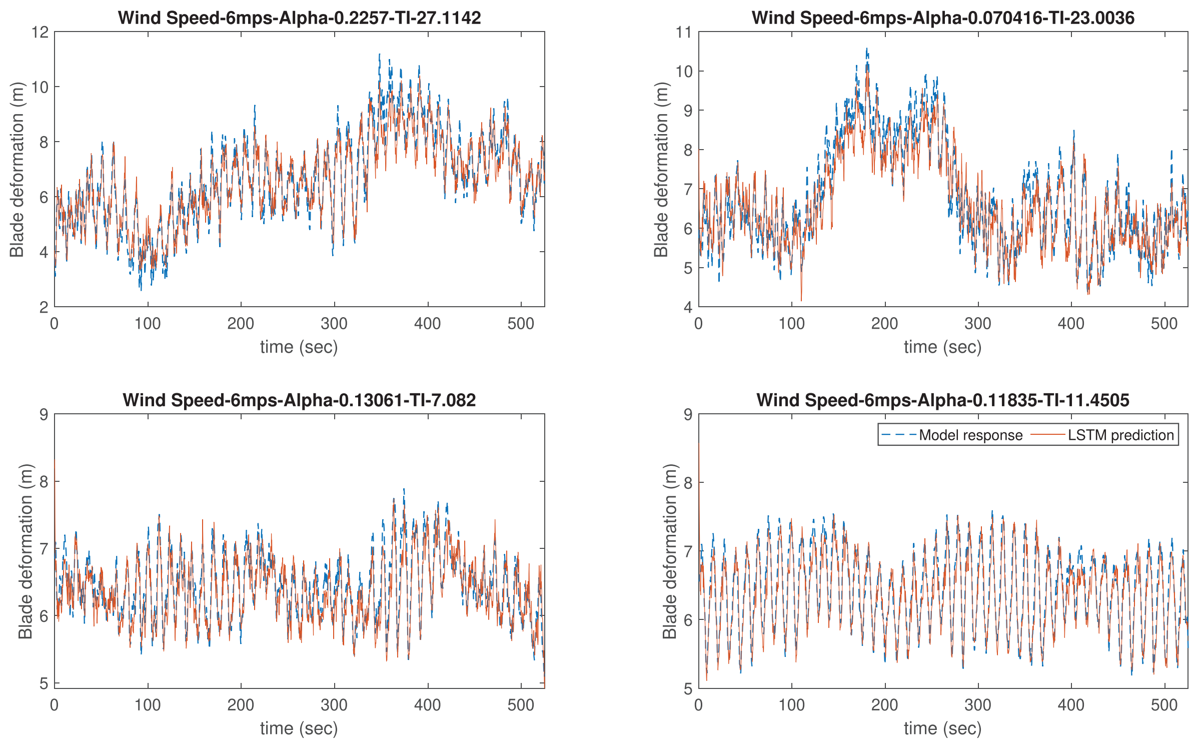
(b) Predictions of LSTM model trained using information from 2 nodes

FIGURE 11 | Performance of LSTM model across different wind speeds.

10991824, 2025, 6, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/we.70027 by Breifni Fitzgerald - Trinity College Dublin , Wiley Online Library on [16/05/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License



(a) Predictions of LSTM model trained using information from node 31



(b) Predictions of LSTM model trained using information from nodes 31 and 47

FIGURE 12 | Performance of LSTM model across different stochastic realisations.

on. However, increasing the number of input features may not always lead to improved performance. This fact is highlighted through Figure 8, where the achieved RMSE is plotted against the number of input features considered.

From Figure 8, it is evident that the best model performance is achieved when a total five aerodynamic input features are considered. The results for the model trained on these features are presented in Figures 9 and 10.

10991824, 2025, 6, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/we.70027 by Breiffni Fitzgerald - Trinity College Dublin, Wiley Online Library on [16/05/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

However, the wrapper-based feature selection methods are required to ensure that an optimum subset of the input features is identified. In wrapper-based selection, first, the RFA method is performed on the subset of input features identified using the MRMR algorithm. The performance of the final model developed using this method is presented in Figures 11 and 12.

Following the RFA approach, the best model performance is observed by considering the following input features:

- Normal force at Node 31
- Tangential force at Node 47
- Rotor speed
- Blade azimuth angle

Figure 11 shows that the model can capture the trend of the blade tip deformation using the information from the normal force at node 31. The results are further improved when the tangential force at node 47 is considered. The effect of adding more channels in terms of the average RMSE value is presented in Table 4. Quantitative analysis of the loss values shows that the addition of information from node 47 results in a 18.91% reduction in the RMSE. Since the model developed using information from only nodes is able to capture the trend well, the need for the addition of further nodes depends on the expected performance of the model and the resources available at the installation location.

TABLE 4 | Effect of number of input force channels on RMSE.

Number of nodes	RMSE
1	0.5658
2	0.4758

To study the effect of feature interaction, the model is trained using the exhaustive feature search method. The results of the exhaustive feature search are presented in Figures 13 and 14. The best performance for the exhaustive feature search method is observed for the following combination of input features:

- Normal force at Node 30
- Tangential force at Node 45
- Rotor speed
- Blade azimuth angle

To compare the performance of the three feature selection methods presented in this study, the final RMSE obtained by the LSTM model trained on the best feature subset identified by these methods is presented in Table 5. It can be seen that the exhaustive feature search method achieves better results but at the cost of an increased computational resource.

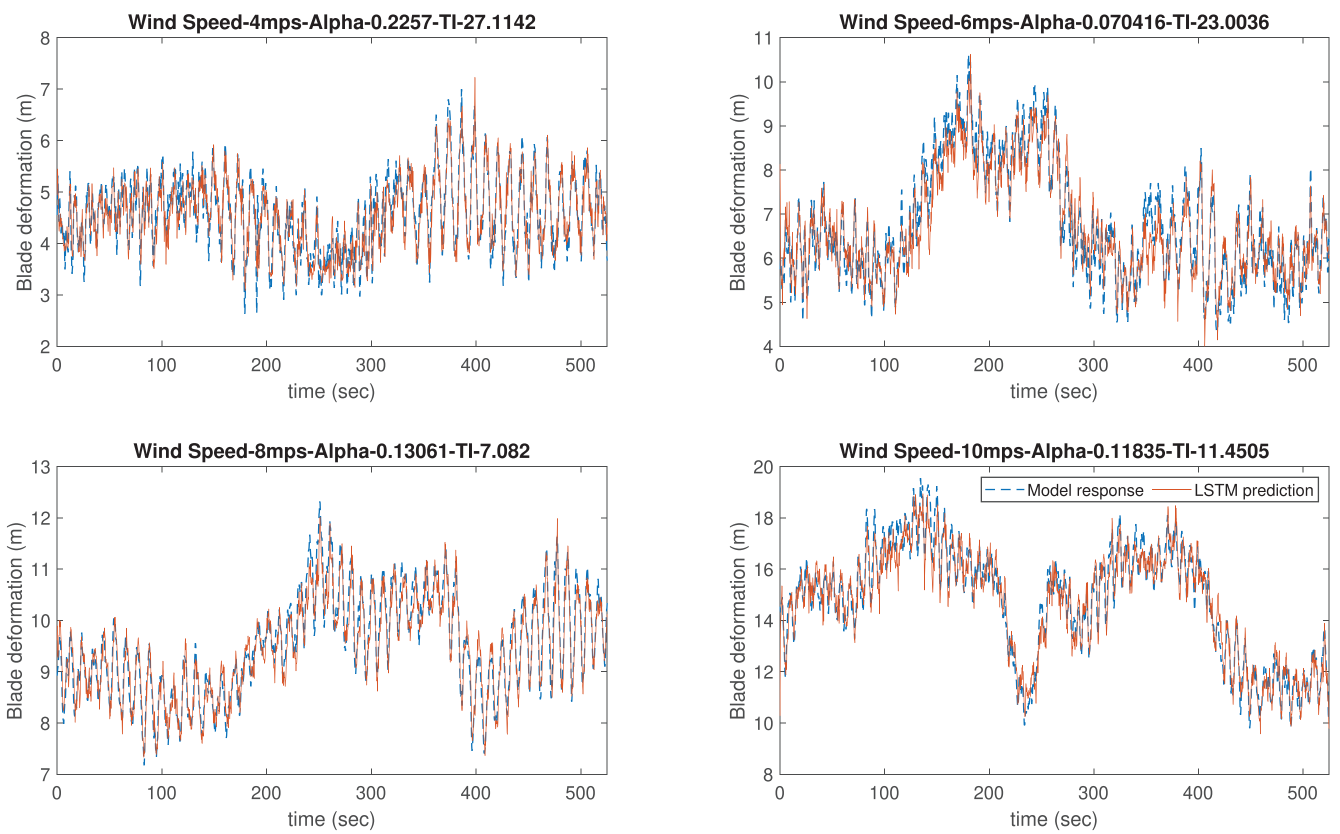


FIGURE 13 | Performance of LSTM model trained using exhaustive feature search across different mean wind speeds.

10991824, 2025, 6, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/we.70027 by Breifni Fitzgerald - Trinity College Dublin, Wiley Online Library on [16/05/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

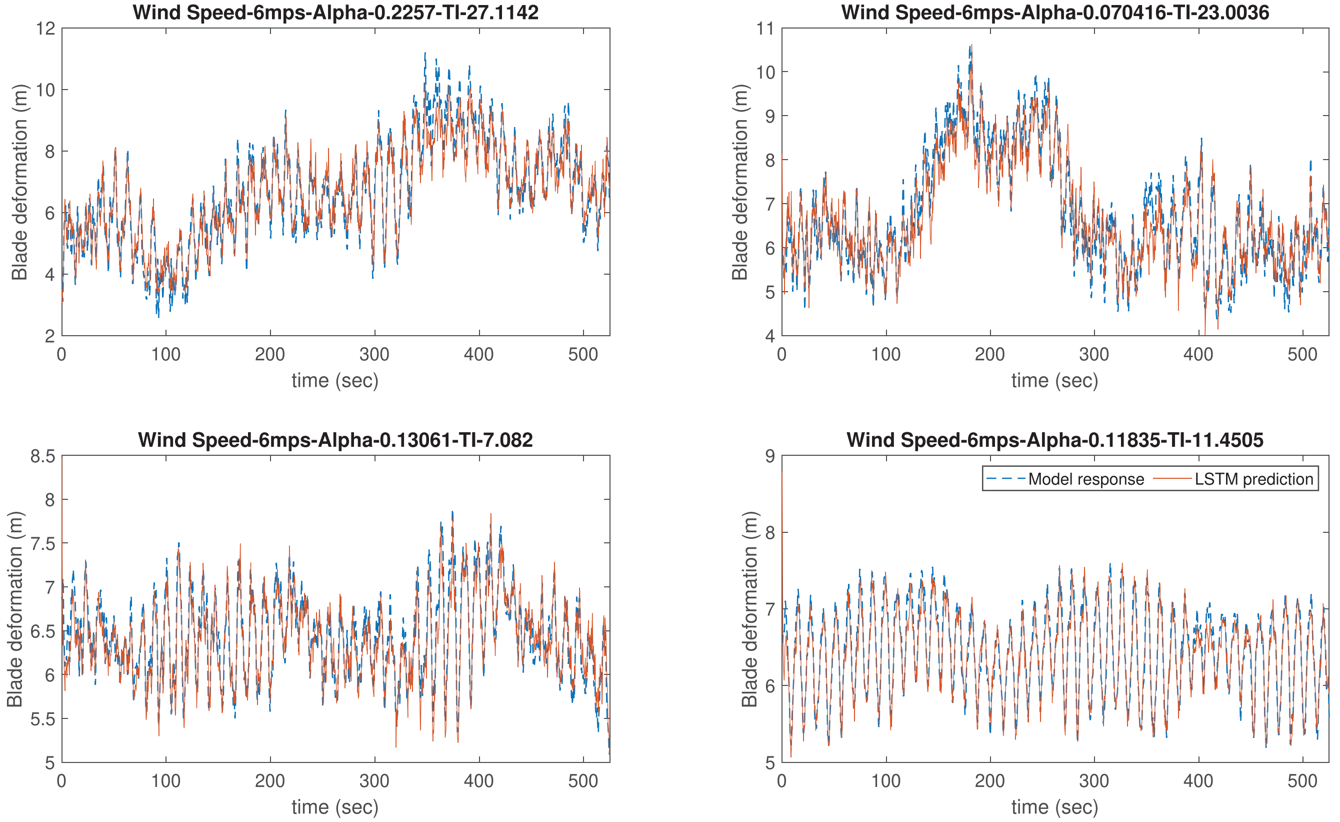


FIGURE 14 | Performance of LSTM model trained using exhaustive feature search across different stochastic realisations.

TABLE 5 | Feature selection method and obtained RMSE metric.

Feature selection method	Validation data	Test data
MRMR	0.5765	0.7983
MRMR + RFA	0.4758	0.5247
MRMR + Exhaustive feature identification	0.4218	0.3691

7.1 | Dealing With Noisy Input Signals

One of the possible applications of the LSTM model developed in this study is as a virtual sensor for analysing blade response in operating wind turbines. Due to its ability to handle sequential data, it is possible that this model could work online in conjunction with wind turbine controllers or be used for offline applications. In such scenarios, where the model works with real-world measurements, the model inputs will often be subjected to uncertainty due to measurement noise or sensitivity limitations. To evaluate the model's robustness against this uncertainty, synthetic Gaussian noise is introduced into the simulated wind velocity data.

The noise levels chosen (10%, 20% and 30%) represent scenarios with a high uncertainty of the measurement. While generic, these values are chosen to be consistent with typical error ranges considered in the literature. However, a more accurate representation would require accounting for the propagation of instrument-specific uncertainties in wind speed measurements through aerodynamic force calculations. This detailed uncertainty analysis is beyond the scope of this study but remains an important consideration for real-world deployment.

The model's performance is evaluated under these noise levels by comparing its predictions with the noise-free ground truth. Figure 15 visually shows the impact of noise on predictions at a mean wind speed of 8 m/s, and Table 6 summarises the corresponding root mean square error (RMSE) and the percentage change observed in the mean and maximum value of predictions with noisy data and the corresponding values with the zero noise case.

To ensure that the comparison of model performance provides statistically significant results, 10 independent noise realisations are simulated for each noise level, and the average of RMSE, mean and maximum values is used to calculate the deviations presented in Table 6. The results demonstrate good overall robustness to uncertainty in the input signal. The acceptable level of uncertainty in the input signals depends on the specific application domain. For applications such as fatigue analysis, where results rely primarily on aggregate statistics, comparatively high noise levels could be considered acceptable. However, applications focussing on extreme values, such as structural health monitoring, might be significantly affected by high noise levels. Further detailed analysis can be performed based on the expected application of the model.

10991824, 2025, 6, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/we.70027 by Breiffni Fitzgerald - Trinity College Dublin, Wiley Online Library on [16/05/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

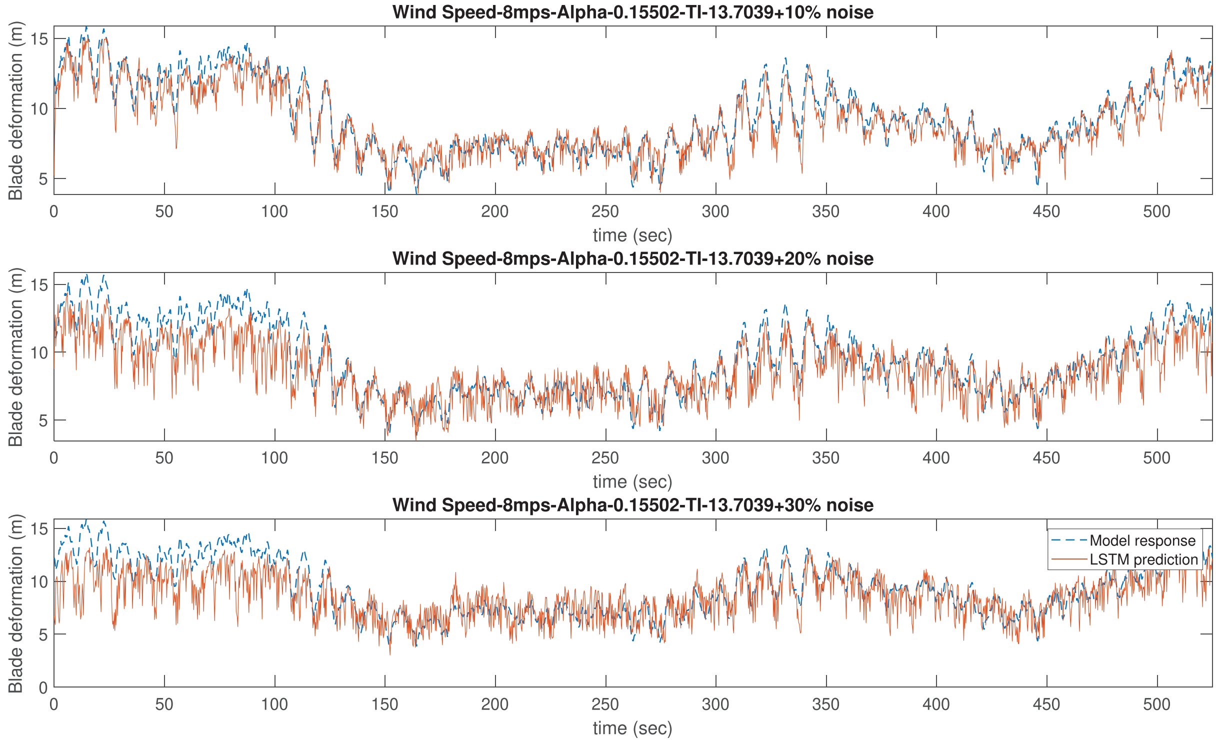


FIGURE 15 | LSTM model predictions for noisy input data.

TABLE 6 | Effect of noise on LSTM model performance.

Noise level	Achieved RMSE	% Change in mean	% Change in maxima
10%	0.9495	0.07	0.65
20%	1.4742	0.94	2.63
30%	2.0776	3.53	5.54

8 | Discussion

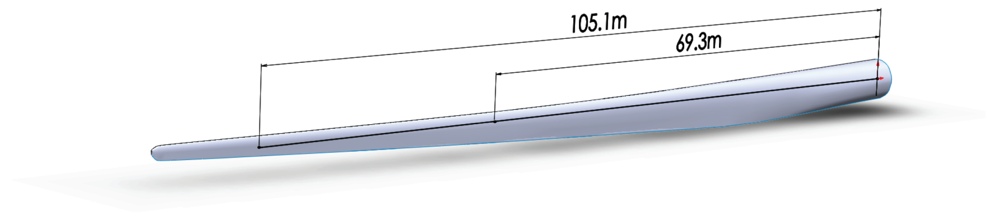
The results presented in this study highlight several key findings on feature selection, model performance and noise handling capabilities. The filter type feature selection method, the MRMR algorithm in particular, provides an efficient approach to identify a most informative feature subset based on statistical properties of the data, drastically reducing the input feature size from 96 to a mere 10 feature subset (Figures 4 and 6). The surrogate model trained on this subset further highlights that increasing the number of input features may not lead to better performance (Figure 8), highlighting the need to further refine the set of input features to extract pertinent information while maintaining efficiency. Further refinement through recursive feature addition (RFA) led to better results, with fewer input features. Although exhaustive feature search yielded the best performance, it comes at the cost of increased computational burden. Therefore, the best choice between these methods is based on specific application requirements and resource constraints. This trade-off is crucial in practical applications where resource limitations cannot be ignored.

Analysing the locations of the chosen features on the blade offered valuable information. The locations of the features identified by these feature selection methods on the blade are represented in Figure 16.

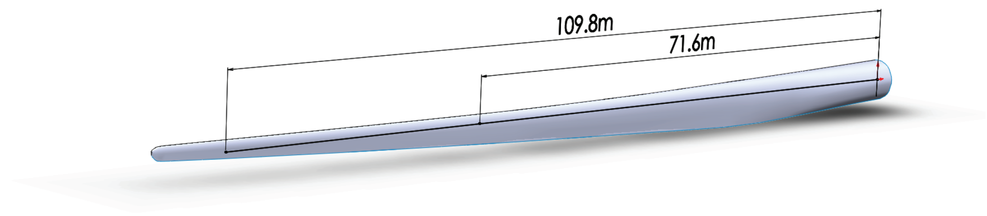
This localisation of features through the wrapper-type methods reveals the blade locations that capture most information about the blade tip deformation. A further analysis of these locations from a structural engineering perspective shows that the locations identified through the feature selection approach represent areas with lower stiffness and higher levels of deformation. Naturally, these regions will contribute substantially to the blade tip deformations. These regions and the corresponding mode shape and rigidity are presented in Figure 17, where the areas are shaded using a rectangular patch.

The performance of the surrogate model has been demonstrated by showcasing the predictive capability of the model for a range of different mean wind speed values (Figures 9, 11 and 13). These results show that the model can predict the blade response over a range of wind velocities encountered during the wind turbine operation. Further, as the stochastic nature of the inflow wind is the most significant source of uncertainty encountered during wind turbine operation, the performance of the model across different stochastic realisations further highlights the efficiency of the surrogate in handling the turbulent inflow encountered during the operation (Figures 12 and 14). The results presented comprehensively demonstrate the capability of this model to predict the blade response at a minimal computational cost. A numerical simulation of wind turbine blades using the model described in Section 3 for a duration of 600 s requires approximately 300 s of CPU time. In contrast, the LSTM model estimates the deformation response in less than a second, demonstrating the computational superiority of the LSTM surrogate model. It is important to note that the

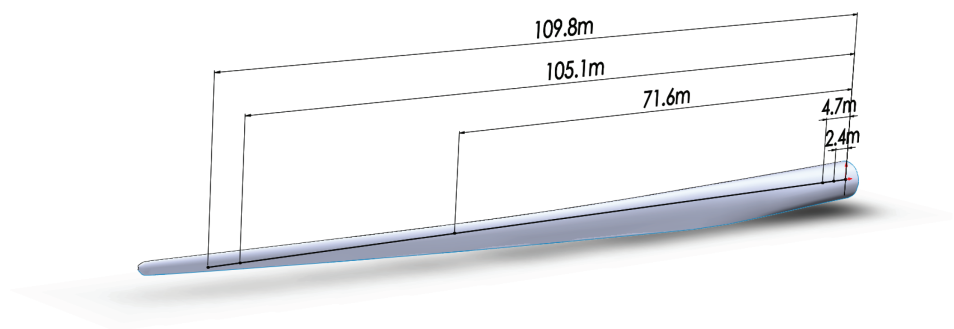
10991824, 2025, 6, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/we.70027 by Breiffni Fitzgerald - Trinity College Dublin, Wiley Online Library on [16/05/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License



(a) Nodes identified through exhaustive feature search: 30 and 45



(b) Nodes identified through recursive feature addition: 31 and 47



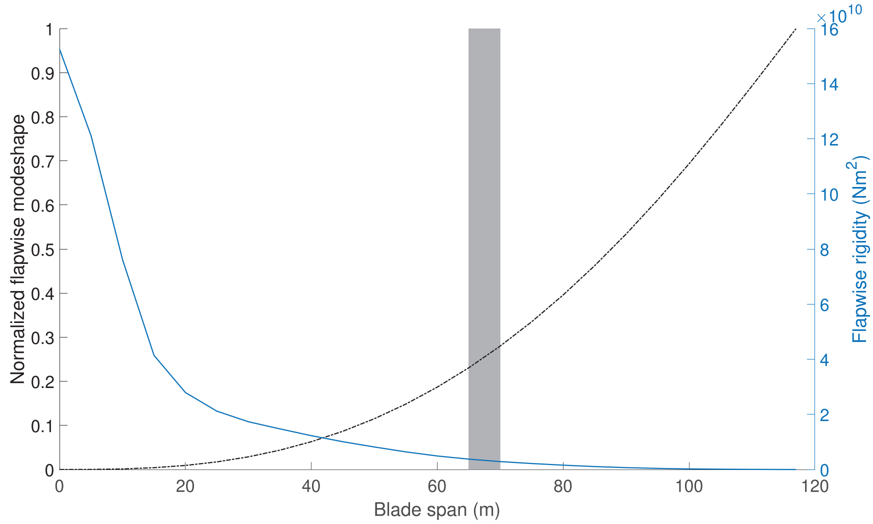
(c) Nodes identified through MRMR: 2, 3, 31, 45 and 47

FIGURE 16 | Nodes identified through different feature identification approaches.

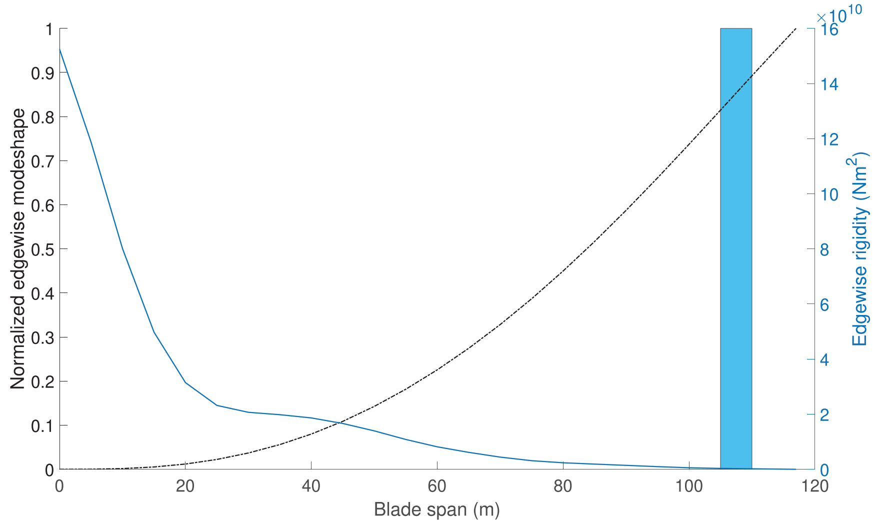
computational advantage does not consider the time invested in generating training data and developing the model; however, the advantages of employing such models for recurrent tasks like site-specific analysis or fatigue evaluation are crucial. For evaluating the computational improvements, all simulations were conducted on a system equipped with an 8-core Intel Xeon CPU operating at a clock speed of 3.8 GHz, utilising 32 GB RAM and running on Microsoft Windows 10 Pro. Advanced techniques like Gated Recurrent Units (GRU), Transformers, and Neural ODE offer potential solutions to the problem examined in this study. However, the LSTM model provided satisfactory results, so this study did not investigate these methods further. Future research could aim to compare the performance of these techniques with the LSTM model.

One of the major applications of this model could be as a virtual sensor for digitally measuring the blade deformation response due to measured inflow conditions. Since this application will inherently require a model that can produce good results under noisy measurements, the performance of the model under noisy input is investigated (Figure 15). The model exhibited robustness against simulated Gaussian noise, even at levels as high as 30%. Interestingly, the noise had a more pronounced effect on extreme values than the response's mean values. This aligns

10991824, 2025, 6, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/we.70027 by Breifni Fitzgerald - Trinity College Dublin , Wiley Online Library on [16/05/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License



(a) Region identified by wrapper type method for aerodynamic force normal to plane



(b) Region identified by wrapper method for force tangential to rotor plane

FIGURE 17 | Region of interest on blade span identified by wrapper type method.

with our intuition, as fluctuations in extreme values are naturally more susceptible to distortion by noise. However, the acceptable noise level depends on the intended application. For instance, fatigue analysis, primarily concerned with aggregate statistics, can tolerate higher noise levels than applications like structural health monitoring, which rely heavily on precise extreme value estimations. Further, delving deeper into the specific application can inform a more detailed analysis of noise tolerance. While this study investigates robustness using simulated noise, it is crucial to acknowledge the limitations of this approach. Real-world noise characteristics can differ from assumed distributions and exhibit dynamic variations. Future work should explore incorporating uncertainty quantification techniques and calibration methods to bridge the gap between simulated and real-world scenarios. Additionally, investigating performance with actual sensor data from wind turbines will provide a more comprehensive understanding of the model's robustness in practical applications.

9 | Conclusion

The authors demonstrated the application of LSTM in predicting the blade tip deformation response time series. A detailed algorithm to develop a machine learning surrogate for sequence prediction addressing the most important steps in model development is presented. A comprehensive analysis of the model's accuracy and robustness against measurement uncertainty is demonstrated. The following major conclusions can be drawn from this study:

- Following the methodology presented in this study, robust LSTM surrogate models can be developed for multivariate time series regression applications.
- The comparative analysis of feature selection methods provides a comprehensive analysis to identify the important features based on the required model performance and the available computational resources.

10991824, 2025, 6, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/we.70027 by Breifni Fitzgerald - Trinity College Dublin , Wiley Online Library on [16/05/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

- The learning rate scheduling approach presented in this study leads to optimum parameter convergence.
- The application of the surrogate for wind turbine monitoring and sensing applications can reduce the number of sensors required and can produce satisfactory performance.

Acknowledgments

This work is supported by the Science Foundation Ireland (SFI) grant no. 20/FFP-P/8702, SFI project NexSys grant no. 21/SPP/3756 and Sustainable Energy Authority of Ireland (SEAI) grant no. RDD/604. Open access funding provided by IReL.

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

The data presented in this study are available on request from the corresponding author. The data are not publicly available because it also forms part of an ongoing study.

References

1. P. Veers, C. Bottasso, L. Manuel, et al., "Grand Challenges in the Design, Manufacture, and Operation of Future Wind Turbine Systems," *Wind Energy Science Discussions* 2022 (2022): 1–102.

2. R. McKenna, P. O. Vd Leye, and W. Fichtner, "Key Challenges and Prospects for Large Wind Turbines," *Renewable and Sustainable Energy Reviews* 53 (2016): 1212–1221.

3. B. Fitzgerald and B. Basu, "Vibration Control of Wind Turbines: Recent Advances and Emerging Trends," *International Journal of Sustainable Materials and Structural Systems* 4, no. 2-4 (2020): 347–372.

4. M. Bezziccheri, P. Castellini, P. Evangelisti, C. Santolini, and N. Paone, "Measurement of Mechanical Loads in Large Wind Turbines: Problems on Calibration of Strain Gage Bridges and Analysis of Uncertainty," *Wind Energy* 20, no. 12 (2017): 1997–2010.

5. J. G. Schepers and S. J. Schreck, "Aerodynamic Measurements on Wind Turbines," *Wiley Interdisciplinary Reviews: Energy and Environment* 8, no. 1 (2019): e320.

6. T. Kashef and S. R. Winterstein, "Relating Turbulence to Wind Turbine Blade Loads: Parametric Study With Multiple Regression Analysis," *Journal of Solar Energy Engineering* 121 (1999): 156–161.

7. L. D. Nelson, L. Manuel, H. J. Sutherland, and P. S. Veers, "Statistical Analysis of Inflow and Structural Response Data From the List Program," in *Wind Energy Symposium*, Vol. 75944, (2003): 270–282.

8. F. Mouzakis, E. Morfiadakis, and P. Dellaportas, "Fatigue Loading Parameter Identification of a Wind Turbine Operating in Complex Terrain," *Journal of Wind Engineering and Industrial Aerodynamics* 82, no. 1-3 (1999): 69–88.

9. L. Manuel, P. S. Veers, and S. R. Winterstein, "Parametric Models for Estimating Wind Turbine Fatigue Loads for Design," *Journal of Solar Energy Engineering* 123, no. 4 (2001): 346–355.

10. P. J. Moriarty, W. E. Holley, and S. Butterfield, "Effect of Turbulence Variation on Extreme Loads Prediction for Wind Turbines," *Journal of Solar Energy Engineering* 124, no. 4 (2002): 387–395.

11. H. S. Toft, L. Svenningsen, W. Moser, J. D. Sørensen, and M. L. Thøgersen, "Assessment of Wind Turbine Structural Integrity Using Response Surface Methodology," *Engineering Structures* 106 (2016): 471–483.

12. K. Müller, M. Dazer, and P. W. Cheng, "Damage Assessment of Floating Offshore Wind Turbines Using Response Surface Modeling," *Energy Procedia* 137 (2017): 119–133.

13. M. Berchier, "Multi-Fidelity Surrogate Modelling With Polynomial Chaos Expansions," (2016), ETH Zurich, Zurich, Switzerland.

14. N. Ganesh and S. Gupta, "Estimating the Rain-Flow Fatigue Damage in Wind Turbine Blades Using Polynomial Chaos," in *Proceedings of the International Symposium on Engineering under Uncertainty: Safety Assessment and Management (ISEUSAM-2012)*, (Springer, 2013): 711–727.

15. C. Mylonas, I. Abdallah, and E. Chatzi, "Surrogate Modelling for Fatigue Damage of Wind-Turbine Blades Using Polynomial Chaos Expansions and Non-Negative Matrix Factorization," in *Iabse Symposium 2017: Engineering the Future*, Vol. 109, (International Association for Bridge and Structural Engineering (IABSE), 2017): 809–816.

16. R. Haghi and C. Crawford, "Surrogate Models for the Blade Element Momentum Aerodynamic Model Using Non-Intrusive Polynomial Chaos Expansions," *Wind Energy Science* 7, no. 3 (2022): 1289–1304.

17. J. P. Murcia, P.-E. Réthoré, N. Dimitrov, et al., "Uncertainty Propagation Through an Aeroelastic Wind Turbine Model Using Polynomial Surrogates," *Renewable Energy* 119 (2018): 910–922.

18. R. Teixeira, M. Nogal, A. O'Connor, J. Nichols, and A. Dumas, "Stress-Cycle Fatigue Design With Kriging Applied to Offshore Wind Turbines," *International Journal of Fatigue* 125 (2019): 454–467.

19. R. Teixeira, A. O'Connor, M. Nogal, N. Krishnan, and J. Nichols, "Analysis of the Design of Experiments of Offshore Wind Turbine Fatigue Reliability Design With Kriging Surfaces," *Procedia Structural Integrity* 5 (2017): 951–958.

20. H. Yang, Y. Zhu, Q. Lu, and J. Zhang, "Dynamic Reliability Based Design Optimization of the Tripod Sub-Structure of Offshore Wind Turbines," *Renewable Energy* 78 (2015): 16–25.

21. M. Awadallah and A. El-Sinawi, "Effect and Detection of Cracks on Small Wind Turbine Blade Vibration Using Special Kriging Analysis of Spectral Shifts," *Measurement* 151 (2020): 107076.

22. N. Dimitrov, M. C. Kelly, A. Vignaroli, and J. Berg, "From Wind to Loads: Wind Turbine Site-Specific Load Estimation With Surrogate Models Trained on High-Fidelity Load Databases," *Wind Energy Science* 3, no. 2 (2018): 767–790.

23. A. Banik, C. Behera, T. V. Sarathkumar, and A. K. Goswami, "Uncertain Wind Power Forecasting Using LSTM-Based Prediction Interval," *IET Renewable Power Generation* 14, no. 14 (2020): 2657–2667.

24. R. Yu, J. Gao, M. Yu, et al., "LSTM-EFG for Wind Power Forecasting Based on Sequential Correlation Features," *Future Generation Computer Systems* 93 (2019): 33–42.

25. S. Woo, J. Park, and J. Park, "Predicting Wind Turbine Power and Load Outputs by Multi-Task Convolutional LSTM Model," in *2018 IEEE Power & Energy Society General Meeting (PESGM)*, (IEEE, 2018): 1–5.

26. D.-E. Choe, H.-C. Kim, and M.-H. Kim, "Sequence-Based Modeling of Deep Learning With LSTM and GRU Networks for Structural Damage Detection of Floating Offshore Wind Turbine Blades," *Renewable Energy* 174 (2021): 218–235.

27. L. Xiang, P. Wang, X. Yang, A. Hu, and H. Su, "Fault Detection of Wind Turbine Based on SCADA Data Analysis Using CNN and LSTM With Attention Mechanism," *Measurement* 175 (2021): 109094.

28. H. Chen, H. Liu, X. Chu, Q. Liu, and D. Xue, "Anomaly Detection and Critical SCADA Parameters Identification for Wind Turbines Based on LSTM-AE Neural Network," *Renewable Energy* 172 (2021): 829–840.

29. J. Liew, W. H. Lio, A. M. Urbán, J. Holierhoek, and T. Kim, "Active Tip Deflection Control for Wind Turbines," *Renewable Energy* 149 (2020): 445–454.

10991824, 2025, 6, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/we.70027 by Breifni Fitzgerald - Trinity College Dublin , Wiley Online Library on [16/05/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

30. N. Dimitrov and T. Göçmen, "Virtual Sensors for Wind Turbines With Machine Learning-Based Time Series Models," *Wind Energy* 25, no. 9 (2022): 1626–1645.

31. T. R. Kane and D. A. Levinson, *Dynamics, Theory and Applications* (McGraw Hill, 1985).

32. S. Sarkar and B. Fitzgerald, "Use of Kane's Method for Multi-Body Dynamic Modelling and Control of Spar-Type Floating Offshore Wind Turbines," *Energies* 14, no. 20 (2021): 6635.

33. E. Gaertner, J. Rinker, L. Sethuraman, et al., "IEA Wind TCP Task 37: Definition of the IEA 15-Megawatt Offshore Reference Wind Turbine," National Renewable Energy Lab. (NREL), Golden, CO (United States), (2020).

34. B. J. Jonkman, "Turbsim User's Guide: Version 1.50," National Renewable Energy Lab. (NREL), Golden, CO (United States), (2009).

35. B. Jonkman, A. Platt, R. M. Mudafort, et al., *Openfast/Openfast: v4.0.0* (Zenodo, 2024), https://doi.org/10.5281/zenodo.14553563.

36. R. Chen, X. Jin, S. Laima, Y. Huang, and H. Li, "Intelligent Modeling of Nonlinear Dynamical Systems by Machine Learning," *International Journal of Non-Linear Mechanics* 142 (2022): 103984.

37. R. Zhang, Z. Chen, S. Chen, J. Zheng, O. Büyüköztürk, and H. Sun, "Deep Long Short-Term Memory Networks for Nonlinear Structural Seismic Response Prediction," *Computers & Structures* 220 (2019): 55–68.

38. H. Peng, J. Yan, Y. Yu, and Y. Luo, "Time Series Estimation Based on Deep Learning for Structural Dynamic Nonlinear Prediction," in *Structures*, Vol. 29, (Elsevier, 2021): 1016–1031.

39. S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation* 9, no. 8 (1997): 1735–1780.

40. S. Wold, K. Esbensen, and P. Geladi, "Principal Component Analysis," *Chemometrics and Intelligent Laboratory Systems* 2, no. 1-3 (1987): 37–52.

41. R. A. Fisher, "The Use of Multiple Measurements in Taxonomic Problems," *Annals of Eugenics* 7, no. 2 (1936): 179–188.

42. M. A. Kramer, "Nonlinear Principal Component Analysis Using Autoassociative Neural Networks," *AIChE Journal* 37, no. 2 (1991): 233–243.

43. H. Peng, F. Long, and C. Ding, "Feature Selection Based on Mutual Information Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy," *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27, no. 8 (2005): 1226–1238.

44. D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," (2014), arXiv preprint arXiv:1412.6980.

45. E. Simley, L. Pao, N. Kelley, B. Jonkman, and R. Frehlich, "Lidar Wind Speed Measurements of Evolving Wind Fields," in *50th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, (2012): 656.

46. S. Baisthakur and B. Fitzgerald, "Physics-Informed Neural Network Surrogate Model for Bypassing Blade Element Momentum Theory in Wind Turbine Aerodynamic Load Estimation," *Renewable Energy* 2024 (2024): 120122.

10991824, 2025, 6, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/we.70027 by Breiffni Fitzgerald - Trinity College Dublin , Wiley Online Library on [16/05/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License