Journal of Marine Science and Application (2025) 24: 1-20
https://doi.org/10.1007/s11804-025-00683-8

REVIEW

Overview of Data-Driven Models for Wind Turbine Wake Flows

Maokun Ye[1], Min Li[1], Mingqiu Liu[1], Chengjiang Xiao[1] and Decheng Wan[1]

Received: 29 October 2024 / Accepted: 25 December 2024

Abstract

With the rapid advancement of machine learning technology and its growing adoption in research and engineering applications, an increasing number of studies have embraced data-driven approaches for modeling wind turbine wakes. These models leverage the ability to capture complex, high-dimensional characteristics of wind turbine wakes while offering significantly greater efficiency in the prediction process than physics-driven models. As a result, data-driven wind turbine wake models are regarded as powerful and effective tools for predicting wake behavior and turbine power output. This paper aims to provide a concise yet comprehensive review of existing studies on wind turbine wake modeling that employ data-driven approaches. It begins by defining and classifying machine learning methods to facilitate a clearer understanding of the reviewed literature. Subsequently, the related studies are categorized into four key areas: wind turbine power prediction, data-driven analytic wake models, wake field reconstruction, and the incorporation of explicit physical constraints. The accuracy of data-driven models is influenced by two primary factors: the quality of the training data and the performance of the model itself. Accordingly, both data accuracy and model structure are discussed in detail within the review.

Keywords Data-driven; Machine learning; Artificial neural networks; Wind turbine wake; Wake models

1 Introduction

1.1 Overview

To realize the vision of carbon neutrality by the mid-century, the deployment of wind energy, alongside other renewable energy sources, must increase severalfold compared with current levels. As a result, over the past decade and in the foreseeable future, wind farms comprising multiple wind turbines have been and will continue to be established worldwide. However, achieving this goal requires not only an increase in the total number of wind farms but also significant improvements in power generation efficiency at both the individual turbine and wind farm levels. These enhancements are crucial for maximizing the total power that can be harnessed from wind energy. To optimize the overall efficiency of wind farms, employing robust modeling approaches for the wakes generated by individual or multiple wind turbines is essential. The accuracy and complexity of these modeling methods can significantly influence critical outcomes, such as annual power production and wind farm layout design. Consequently, wind turbine wake modeling has remained a prominent research focus within the wind energy community, garnering substantial attention from researchers worldwide.

As in other areas of fluid mechanics, wind turbine wake modeling traditionally relies on two canonical approaches: analytical and numerical methods. However, with recent advancements in artificial intelligence, an emerging and rapidly evolving modeling framework leveraging modern machine learning techniques has gained increasing traction in this field. This framework, widely known as the data-driven approach, is becoming a prominent tool in wind turbine wake modeling. Data-driven models, particularly artificial neural networks (ANNs), can effectively represent high-dimensional and nonlinear flow phenomena. They can automatically identify patterns and learn from data without requiring the additional assumptions or simplifications typically introduced in analytical wake models. Moreover, they offer exceptional computational efficiency compared with numerical simulations, making them a promising alter-

Article Highlights

- Data-driven methods are increasingly popular in wind turbine wake modeling.
- The accuracy of data-driven models depends on multiple factors.
- The frameworks that have been applied to the data-driven modeling of wind turbine wake flows are summarized.
- The core ideas of different artificial neural-network models applied in the modeling of wind turbine wake flows are discussed.

✉ Decheng Wan
dcwan@sjtu.edu.cn

1 Computational Marine Hydrodynamics Lab (CMHL), School of Naval Architecture, Ocean and Civil Engineering, Shanghai Jiao Tong University, Shanghai 200240, China

native to traditional wake modeling methods. Considering this trend, the authors present a review of existing work, offering a comprehensive and clear overview of this research area for the wind energy community.

Given that the scope of this paper is to review state-of-the-art research on machine learning-based wind turbine wake modeling, the authors believe that a solid understanding of the two foundational topics—wind turbine wake modeling and machine learning—is essential for a comprehensive grasp of this work. Therefore, we will first provide concise yet sufficient introductions to these two subjects to support the reader's understanding of the paper.

1.2 Conical wind turbine wake modeling frameworks

As previously discussed, wind turbine wake modeling traditionally relies on two classic frameworks—analytical and numerical approaches—alongside the emerging data-driven approach. The data used in data-driven wake modeling is often derived from conical wake modeling methods. Readers should also bear in mind that, regardless of how advanced machine learning techniques may be, the accuracy of a data-driven wake model cannot exceed the quality of the data used to train it. Thus, the overall accuracy of these models is determined by two key factors: the quality of the input data and the performance of the model itself. To provide context, a brief overview of the canonical wake modeling approaches—analytical and numerical—will be presented in order of increasing fidelity.

1.2.1 Analytical wake models

An analytical wake model (AWM) represents the wake pattern of either a yawed or un-yawed turbine through simplified equations to estimate flow quantities, such as velocity and turbulence, within the wind turbine wake. Regardless of their complexity, AWMs aim to establish explicit formulations to describe the evolution of wind turbine wakes. However, owing to the numerous simplifications involved in their derivation, AWMs are considered low-fidelity compared with numerical simulations, which solve the governing equations of fluid dynamics. Despite this, the high computational efficiency of AWMs makes them indispensable for wind farm optimization tasks (Shakoor et al., 2016), where a large number (on the order of 1 000 to 10 000) of wind-farm-scale wake calculations are required.

Early analytical modeling work began with Jensen (Shakoor et al., 2016), who, based on the assumption of a "top-hat" distribution for cross-wind variation, leveraged the conservation of mass and momentum to develop a simple linear model for the velocity deficit. Building on the Jensen model, researchers have made significant efforts to develop more realistic wake models that better represent the wake field of wind turbines. A comprehensive literature review of analytical wake models can be found in Amiri et al. (2024), and as such, it will be omitted from the current paper.

1.2.2 Numerical wake modeling

As mentioned earlier, wind turbine wakes can be simulated by solving the complete set of fluid governing equations, namely the Navier–Stokes (N–S) equations. This approach is known as computational fluid dynamics (CFD). Owing to the multiscale nature of fluid motion, particularly turbulence, a temporal or spatial filter can be applied to the original N–S equations to obtain the well-known Reynolds-averaged Navier–Stokes (RANS) equations or the Large Eddy Simulation (LES) equations, respectively. In general, results obtained using LES are considered to be of higher fidelity than those derived from RANS simulations. In addition, different blade modeling approaches with varying levels of accuracy are used in CFD simulations of wind turbine wakes, including reduced-order modeling (ROM) and fully resolved approaches. In CFD simulations based on ROM, the blade element method (BEM) and its enhanced variants, including the actuator disc model (ADM) (Calaf et al., 2010; Helvig et al., 2021) and the actuator line model (ALM) (Troldborg, 2009; Martínez-Tossas et al., 2015; Nilsson et al., 2015; Draper et al., 2018), simplify the wind turbine blades as equivalent integral forces over two-dimensional airfoil cross-sections, thereby bypassing the need to resolve boundary-layer flow. In contrast, CFD simulations using fully resolved turbine geometries (FRGs) directly resolve the details of the wind turbine blades, with turbulent flow in their vicinity calculated explicitly (Ye et al., 2024a).

According to the above discussions, the overall accuracy, or fidelity, of numerical modeling of wind turbine wakes depends on two factors: the accuracy of the solved governing equations and the accuracy of the blade modeling methods. Within the framework of CFD simulations, this review considers results from LES simulations with FRGs to have the highest accuracy (denoted as LES/FRG), while results from RANS simulations with ROM are considered to have the lowest accuracy (denoted as RANS/ROM). All other combinations are treated as having intermediate accuracy.

1.3 Machine learning in general

Machine learning is a framework that uses algorithms to extract or learn knowledge from data without relying on prior assumptions. To facilitate this, machine learning models are typically trained on large datasets. Because this approach theoretically requires only data and not physical knowledge, the approach is often referred to as data-driven modeling in the context of physical process modeling.

Machine learning methods can generally be categorized into three types: supervised learning, unsupervised learning, and reinforcement learning (Moussaoui et al., 2023). Supervised learning is a type of machine learning where the model is trained on a labeled dataset, meaning that the training data includes both the input features and the corresponding correct output labels. The goal of supervised learning is to learn a mapping between the input features and the output labels. Examples of supervised learning tasks include classification and regression. Common algorithms for supervised learning include linear regression, logistic regression, decision trees (DTs), support vector machines (SVMs), and neural networks. In contrast, unsupervised learning models are trained without labels. The primary goal of unsupervised learning is to explore the data and uncover hidden patterns or representations, such as in fault diagnosis tasks (Gao et al., 2015). Common algorithms for unsupervised learning include k-means clustering, hierarchical clustering, and principal component analysis (PCA). In addition, reinforcement learning is a type of machine learning where an agent learns to make decisions by taking actions within an environment to achieve a specific goal. Unlike supervised learning, there are no labels provided by a teacher, and unlike unsupervised learning, the agent is guided by a clear objective it seeks to accomplish. The agent learns by receiving feedback in the form of rewards or penalties based on its actions. Over time, the agent develops a policy that maximizes cumulative reward. Reinforcement learning proves particularly useful in situations where the consequences of actions are not immediately clear, requiring the agent to learn through trial and error. This approach finds common applications in areas such as game playing, robotics, and autonomous vehicles (Kober and Peters, 2014; Isele et al., 2018).

Each of these learning paradigms has its strengths and is suited to different types of problems and data. Given the nature of wind turbine wake-related tasks, supervised learning is used in most, if not all, existing studies that apply data-driven methods. Therefore, in this review, we will focus exclusively on this category in the discussion of machine learning.

In supervised learning, regardless of the complexity of the machine learning model, the goal is to establish a mapping between the input data and the output labels, as mentioned earlier. Specifically, inspired by the function of human neurons, ANNs were developed to connect the input and output data. The efficient algorithms created for training ANNs have sparked significant growth in this field, having a profound impact on nearly every area of modern life.

1.4 Accuracy of data-driven models

According to the brief introductions above, readers should understand that the overall accuracy of data-driven models is determined by at least two main factors: the accuracy of the data used during the training process and the performance of the model itself. It is also important to note that the highest accuracy achievable by data-driven models is limited by the accuracy of the input data. Therefore, in this review, both the data-driven models and the data used will be discussed.

1.5 Structure of the current review paper

In this review, we focus on studies that apply machine learning techniques to investigate wind turbine wake-related topics, specifically wind turbine/farm power prediction and wind turbine wake modeling, with an emphasis on the latter. The remainder of the paper is organized as follows: Section 2 introduces the machine learning architectures commonly used in wind turbine wake modeling, presented in order of increasing complexity. Section 3 provides a comprehensive literature review of data-driven modeling studies on wind turbine wake, organizing the studies into four categories: wind turbine power prediction, machine learning-based analytical wake models, wake field reconstruction, and the enforcement of explicit physical constraints.

2 Definition and classification of machine learning methods

In this section, machine learning methods commonly used in data-driven wake modeling are introduced. By the end of this section, readers should have a clear understanding of the key concepts frequently discussed in machine learning-related studies. We begin by presenting an overview of non-neural network machine learning methods, which are also commonly used in data-driven wake modeling. Afterward, the dominant neural network approaches are discussed, starting with basic architectures and progressing to more advanced structures built upon these foundational models. Finally, the concept of symbolic regression (SR) is introduced, typically combining ANNs with optimization algorithms.

2.1 Non-neural network machine learning approaches

2.1.1 SVM and support vector regression

SVMs (Hearst, 1998) are a class of supervised learning algorithms commonly used for classification and regression tasks. The core concept of SVMs is to identify the optimal hyperplane that maximally separates different classes within the feature space. Support vector regression (SVR) is an extension of the SVM framework specifically designed for regression tasks. Similar to SVMs, SVR aims to find an optimal hyperplane that best fits the data. However, instead of maximizing the margin between classes, SVR focuses on minimizing prediction error while balancing the model's complexity and its ability to generalize. In SVR, the goal is to find a hyperplane that minimizes the Euclidean distance between the observed values (i. e., the ground truth) and the model predictions, subject to a tolerance level known as the epsilon-insensitive tube. Data points falling within this tube are considered correctly predicted, while those outside it are penalized. This approach makes SVR robust to outliers, as it focuses on minimizing the prediction error within a specified threshold rather than attempting to perfectly fit every data point. Despite its computational demands, particularly with large datasets, SVR remains a popular choice due to its ability to provide accurate and robust predictions, even in the presence of noise and outliers.

2.1.2 DT-based algorithms

A DT (Myles et al., 2004) is a fundamental class of supervised learning algorithms used for both classification and regression tasks. It works by recursively partitioning the feature space into regions based on the values of input features, creating a tree-like model of decisions and their possible consequences. Each internal node represents a test on an attribute, each branch represents the outcome of the test, and each leaf node represents a class label (in classification) or a predicted value (in regression). To overcome the limitations of basic DTs, more advanced algorithms based on DTs have been developed. For example, the random forest (RF) (Breiman, 2001) algorithm reduces overfitting by combining multiple tree structures, and XGBoost (Chen and Guestrin, 2016) enhances DTs by applying gradient boosting.

2.2 Basic neural network architectures

2.2.1 Multilayer perceptron

The multilayer perceptron (MLP) (Popescu et al., 2009), also frequently referred to as ANNs, backpropagation neural networks (BPNN), deep neural networks (DNN), or fully connected neural networks (FCNN) in other publications, is one of the most fundamental components of advanced NN architectures. The term ANN will be used to refer to ANNs in a general sense. For example, recurrent neural networks (RNN) and convolutional neural networks (CNN) can also be called as ANNs. An illustration of MLP is illustrated in Figure 1.

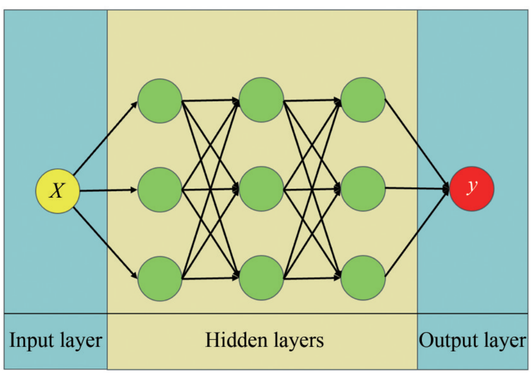

Figure 1 Illustration of an MLP model

It can be seen that an MLP consists of multiple layers of neurons, including an input layer, one or more hidden layers, and an output layer. Each neuron is weighted and connected between layers, processing input signals through activation functions to perform nonlinear transformations. The training of an MLP involves two main stages: forward propagation and backpropagation. The details of these two stages are as follows. Suppose the input to a neuron is denoted as $X\left(x_1, x_2, \cdots, x_i\right)$, where x_i represents the input "features". In forward propagation, for each neuron, the following two operations are performed:

1) Linear transformation:

$$\text{output}' = b + \sum_{i=1}^{n} w_i \cdot x_i \tag{1}$$

where w_i represents the weights of the neuron, and b denotes the bias in the linear transformation. Alternatively, this can be expressed as:

$$\text{output}' = b + \text{dot}\left(W, X\right) \tag{2}$$

where W is the weight matrix, X is the input vector, and dot denotes the dot product.

2) Nonlinear transformation:

Then, the output of the neuron is "activated" by applying a nonlinear function σ, (e.g., the tanh function) to the result of the linear transformation, allowing the model to capture the nonlinearity in the data. Note that output′ denotes the intermediate result from the linear transformation, while output refers to the final result of the neuron.

$$\text{output} = \sigma\left(\text{output}'\right) \tag{3}$$

The two operations within a single neuron are illustrated in Figure 2.

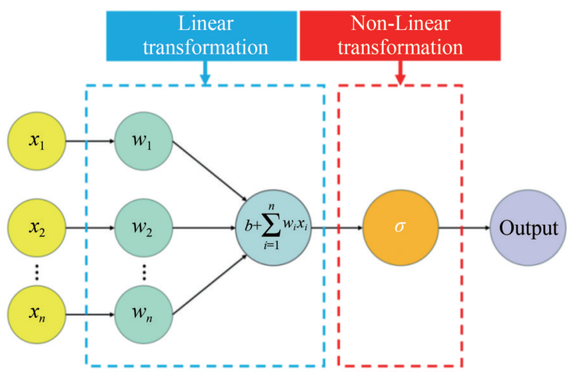

Figure 2 Illustration of a single neuron in an MLP model

In an MLP model, multiple layers of neurons are used, with the above two operations performed repeatedly. The final output of the entire MLP is denoted as y_{predict}.

In backpropagation, the loss function is first calculated by comparing the predicted output, y_{predict}, with the ground truth, y_{label}, using a specific method, such as mean square error (MSE). Then, the weights and biases—i.e., the trainable parameters—of the MLP model are adjusted, typically using the gradient descent method (Rumelhart et al., 1986), to minimize the loss function. By feeding the MLP with data, the model is trained to describe the mapping relationship between input features and output labels. This process allows the model to "learn" the underlying patterns in the data.

2.2.2 RNNs

RNNs are a class of ANNs designed to recognize patterns in sequential data, such as time series data or natural language. Unlike MLP models, which process inputs independently of sequence, an RNN model captures dependencies within a sequence by introducing an internal state. The mathematical representation of an RNN model can be written as:

$$\text{output}^T = \sigma\left[\text{dot}\left(W_i, \text{input}^T\right) + \text{dot}\left(W_s, \text{state}^T\right) + b\right] \tag{4}$$

with

$$\text{state}^T = \text{output}^{T-1} \tag{5}$$

In the above two equations, the superscript T denotes the time step T or the position in the sequence. W_i and W_s are the weight matrices for the input^T and state^T respectively. By introducing the concept of an internal state, information within a time sequence can be carried forward, enabling the model to capture dependencies within the sequence. To enhance the performance of the basic RNN structure, advanced RNN architectures such as long short-term memory (LSTM) networks (Hochreiter and Schimidhuber, 1997) and gated recurrent units (GRUs) (Chung et al., 2014) have been developed. Further details of RNN-based architectures are beyond the scope of this review, and readers are encouraged to refer to Chung et al. (2014) for more information.

2.2.3 CNNs

CNNs are a family of deep learning models designed for processing grid-like data (LeCun and Bengio, 1998), such as images. Originally developed to automatically and adaptively learn spatial hierarchies of features from visual data, CNNs typically consist of convolutional layers, pooling layers, and fully connected layers. The convolutional layers use filters that slide over the input image to detect local features, such as edges. These features are then "pooled" (e.g., using max pooling) to reduce spatial dimensions and highlight the most important features. This pooling process is repeated with increasingly complex filters, allowing the network to build a hierarchy of features. The output from the final pooling layer is flattened and passed into an MLP, which connects to the output labels. CNNs have revolutionized the field of computer vision, driving significant advances in image recognition, object detection, and image segmentation tasks (Lecun et al., 2015; Naranjo-Torres et al., 2020). Notably, due to their excellent capability for local feature extraction, CNNs are often used as local-dependency extractors in advanced ANNs. A schematic representation of a CNN model is illustrated in Figure 3.

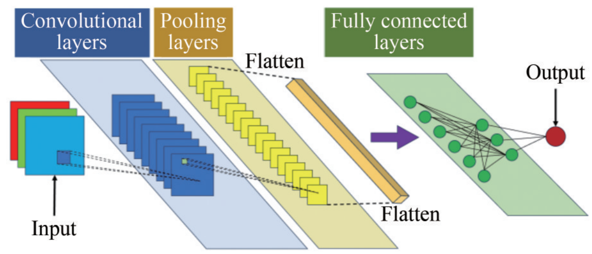

Figure 3 Illustration of a CNN model

2.2.4 Attention mechanism

The attention mechanism is a method designed to enhance the global feature extraction capabilities of ANNs, particularly when handling sequential data, such as in natural language processing (NLP). It typically involves three main components: query, key, and value. The mechanism generates a weighted representation of the values by calculating the similarity between the query and the keys. By doing so, it can effectively capture long-term dependencies, improve the model's ability to focus on key information and enable it to concentrate on the most important parts of the input. For more details on the attention mechanism, please refer to Vaswani et al. (2017).

2.2.5 Graph neural networks

Graph neural networks (GNNs) are a class of deep learning models designed to process data represented as graphs, which consist of nodes and edges that encode relationships between entities. GNNs are particularly effective at capturing the rich relational information inherent in graph-structured data, making them versatile for a wide range of tasks across different domains. GNNs operate on graph data by aggregating and transforming information at each node through a series of neural network layers. This enables the model to learn representations that encode both the features of individual nodes and the structure of the graph itself. For more information on GNNs, please refer to Scarselli et al. (2009), Xu et al. (2019), Zhou et al. (2020), and Wu et al. (2021).

2.3 Advanced ANN structures

According to the aforementioned fundamental ANN architectures, advanced ANN structures can be designed to enhance the overall performance of machine learning models. In this review paper, we categorize these approaches for improving ANN performance into three groups: enhanced feature extraction, physics-informed neural networks, and operator learning, based on their methods of improving ANNs.

2.3.1 Enhanced feature extraction

The first approach to improving ANN performance is by increasing the complexity of their structures. By stacking or connecting different fundamental ANN architectures, advanced structures can be designed with enhanced feature extraction capabilities. While the term ‘feature extraction’ can be defined from various perspectives, in this review, it refers to the process by which ANN models ‘perceive’ or ‘detect’ hidden patterns within input data. In this context, elementary ANN architectures can be considered feature extractors, and careful design of ANN structures—by combining different feature extractors to better fit a specific dataset or task—becomes possible. One of the most representative examples in this category is the U-Net structure (Ronneberger et al., 2015), which was initially designed for biomedical image segmentation and has since inspired many other advanced ANN architectures. It consists of an encoder (down-sampling path) and a decoder (up-sampling path), constructed by stacking multiple CNN layers with skip connections. An illustration of the U-Net structure is illustrated in Figure 4.

2.3.2 Physics-informed neural networks

In addition to enhancing the feature extraction capability of ANN models, the second approach to improving overall model performance involves incorporating explicit physical constraints, such as the N−S equations for fluids, into the loss functions of ANN models. This approach was first proposed by Raissi and Karniadakis (Raissi and Karniadakis, 2018) and quickly garnered attention from researchers across various science and engineering disciplines, where black-box ANN models often raise concerns regarding their lack of interpretability and explainability. By integrating governing equations into the training process, the resulting ANN models can be viewed as models that simultaneously satisfy both the observed data and the underlying physics. In this way, the concern that an ANN model may fit only the data points but fail to accurately represent the underlying physical process is mitigated. Additionally, the generalization ability (i.e. the ability to predict unseen data) of the ANN model is often improved. This method will be discussed in detail later in Section 3.4.1 and is therefore omitted here for the sake of brevity.

2.3.3 Operator networks

The third approach to improving the overall performance of ANNs is the concept of operator learning. Although the studies reviewed in this paper did not utilize operator networks (ONs) in their models, they are included here for completeness. Strictly speaking, the concepts underlying ONs, such as DeepOnets and FNOs, differ from the previously introduced ANN architectures. However, since ONs also leverage elementary ANN architectures to realize their functionality, they are presented here as part of subsection 2.3. If we treat an ANN model as a function mapping that relates input and output data, ONs can be viewed as an operator mapping that connects the functions generating the input data to those generating the output data. Theoretically, this assumption of underlying relations among observed data should be more general and offer improved performance in representing higher-dimensional data. The detailed

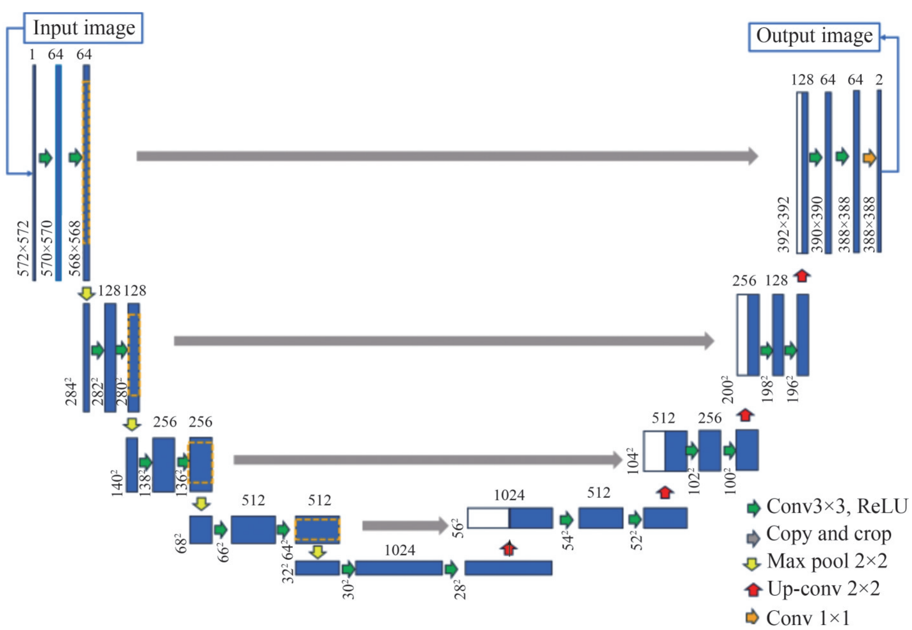

Figure 4 Illustration of the U-Net model

mathematical background of ONs can be found in Chen and Chen (1995). Illustrations of typical ON structures are illustrated in Figure 5. For more information on ONs, readers are encouraged to refer to the original papers (Lu et al., 2021).

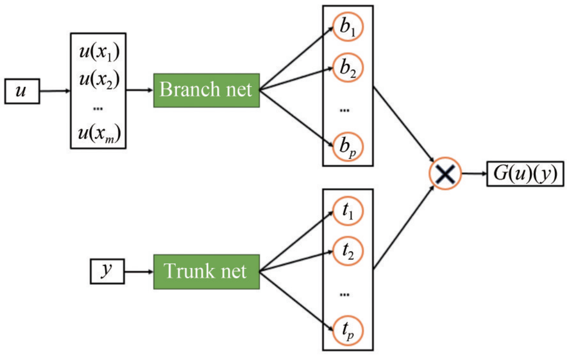

Figure 5 Illustration of ONs

2.4 SR

SR is a subset of regression analysis that aims to identify an explicit mathematical formula representing the relationship between variables in a dataset. Unlike numerical regression methods, which provide a set of coefficients for models (such as linear or polynomial regression), SR seeks to uncover the underlying functional form of the relationship. As a result, SR is considered a promising approach for tasks such as wind turbine modeling. SR is typically performed using techniques like genetic programming (GP), where a population of candidate equations is evolved over time through operations such as mutation, crossover, and selection, mimicking the process of natural selection. The advantage of SR lies in its ability to produce interpretable models that offer insights into the underlying processes generating the data. The resulting models are not only predictive but also explainable, as they can be analyzed to understand how changes in one variable affect another. However, SR also has limitations, such as the risk of overfitting the training data and the need for careful specification of the function set from which the model is derived. An illustration of SR is illustrated in Figure 6.

3 Literature review

In this review, research on data-driven modeling of wind turbine wakes is categorized into four areas based on the goal of prediction: analytical wake models, wind turbine/farm power prediction, wake reconstruction, and physics-informed neural networks (PINNs). Given the rapid rise of PINNs as a research focus in science and engineering since their introduction, we have chosen to discuss related work in the context of wind turbine wake modeling as a separate category. In this category, PINNs are considered a specific approach for incorporating physical constraints into data-driven modeling processes.

3.1 Data-driven AWMs

The first category of work implements a data-driven framework to either 1) improve the accuracy of existing wind turbine wake models or 2) derive new analytical (i.e., explicit) mathematical expressions for wind turbine wake. In the first case, certain coefficients—often highly empirical or variable across different wind scenarios, such as incoming wind velocity, turbine type, and atmospheric boundary

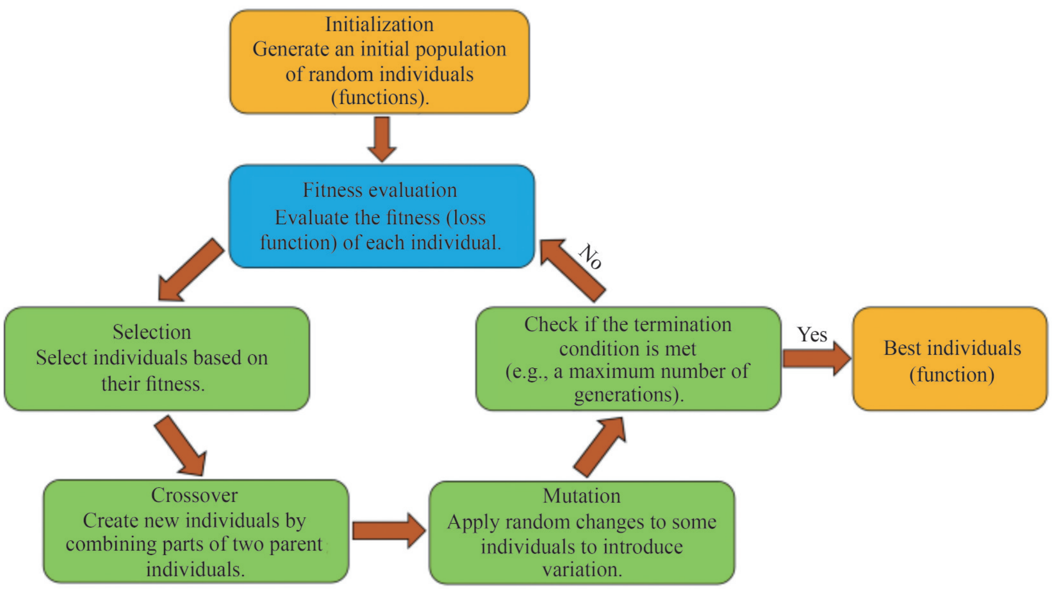

Figure 6 Flow diagram of GP-based SR

layer (ABL) stability—are replaced or improved using an ANN. This approach significantly enhances the accuracy and generalizability of the analytical model.

In particular, the wake expansion rate (Ge et al., 2019) in engineering wake models significantly influences the overall performance of these models, which has attracted the attention of researchers. Guo et al. (2022b) improved the Gaussian wake model developed by Bastankhah and Porté-Agel (2014) by incorporating local inflow information. A machine learning model using RF was trained with field SCADA data to establish the nonlinear relationship between local inflow information and the wake expansion feature. The resulting wake model was tested in real wind farms, demonstrating a 20% improvement over the original wake model. Pujari et al. (2023) incorporated the nonlinear expansion effect into the Jensen wake model using MLP. In the study, the linear wake expansion in the Jensen model was corrected using a neural network that takes downstream distance and linear expansion as input parameters. The training data were collected from field measurements. The results showed that the wake predictions made by the improved Jensen model outperformed those of the original model.

As a side note, the authors would like to further explain the concept of 1) by drawing an analogy with studies aimed at improving the accuracy of RANS turbulence models. In those studies, the complex relationship between Reynolds stresses and mean flow, which would traditionally be obtained by solving additional artificial turbulence transport equations, was replaced by ANNs. For readers familiar with data-driven turbulence modeling, the idea of 1) discussed here can be readily seen as an application of this method in wind turbine wake modeling.

Regarding the second subset, i. e., 2) mentioned above, the SR technique introduced in Section 2.4 is used to derive new analytical expressions for wind turbine wake from observed datasets. In studies following this framework, large datasets are first generated through CFD simulations or measurements, containing various input features such as inflow velocity, inflow angle, ambient turbulence level, and more. SR is then employed to generate or search for the best mathematical formulation that fits the entire dataset, typically using optimization algorithms such as GP.

Kabir et al. (2020) used a GP algorithm to derive new AWMs that account for ABL. In their study, the training data were obtained through RANS/BEM simulations. The resulting analytical wake model was then compared with traditional models using a CFD dataset, showing a clear improvement. Gajendran et al. (2023) developed an SR approach combined with a simulated annealing (SA) optimization method to obtain an explicit wake model for yawed conditions, specifically addressing velocity deficit and wake deflection. The training dataset was obtained from ALM/LES simulations, and a set of potential input parameters was fed into the algorithm for SR. The balance between accuracy and simplicity of the resulting mathematical representation was achieved using a multi-objective combinatorial optimization (MOCO) method. The new explicit wake models obtained were found to align well with the CFD data.

It is worth mentioning that while the analytical expressions of the wind wake field significantly enhance the interpretability of data-driven models and are easier to implement in wake estimations compared with black-box or gray-box models (which will be discussed later), they may be less generalizable. Therefore, greater effort should be dedicated to selecting appropriate input features.

3.2 Wind turbine/farm power prediction

The research works in this category share a common characteristic: they can all be considered classical machine learning tasks, specifically time sequence analysis (TSA), which has been a key focus in NLP for many years. Consequently, research in wind turbine/farm power prediction has been heavily inspired by advancements in NLP. The primary objective of TSA is to predict future values of a sequence based on its historical data. For example, in NLP, an ANN model can be trained to generate an abstract for an article by feeding the model the article content. Similarly, in wind turbine/farm power prediction, many concepts and techniques are either directly inherited from or strongly influenced by the field of NLP.

In this review paper, we further categorize TSA into two types for a clearer understanding of its capabilities and limitations: 1) pure TSA and 2) correlation TSA. In the case of pure TSA, for example, an ANN model can be trained to predict the temperature of a location on the 51st day by feeding it the temperature history from the 1st to the 50th day for that location. In pure TSA, the ANN model takes only the time history of the target variable as input. Readers may immediately recognize that a strong assumption must hold for a pure TSA model to work: future values of a variable should be exclusively and sufficiently determined by its history. However, this assumption is often unrealistic in the real world. Taking the temperature example again, the temperature on the 51st day can also be influenced by other variables, such as humidity, pressure, wind speed, and the temperature at other locations. Therefore, studies that include other correlated variables as inputs will be referred to as "correlation TSA" in this review paper.

3.2.1 Pure TSA

For pure TSA applied to wind turbine/farm power prediction, the goal is to establish the following mapping relationship:

$$P^t = \mathrm{ANN}\left(P^{t-1}, P^{t-2}, P^{t-3}, \cdots, P^{t-n}\right) \tag{6}$$

where P is the power output of a wind turbine or farm, t denotes the time label, and n represents the length of the input sequence.

According to the authors' experience, as discussed in a previous paper (Ye et al., 2024b), this task is extremely challenging for two main reasons: 1) as mentioned earlier, the history of the target variable may not be sufficient to predict its future values, and 2) the multi-dimensional temporal characteristics of time sequences are difficult, if not impossible, to capture. For example, consider using a 50-day power sequence to predict the power output on the 51st day. The first question that may arise is whether a 50-day history is long enough to capture the temporal patterns in the variation of power output. In fact, to accurately predict the power output of a wind turbine or farm on the 51st day, one might feel that the power output from the past several years should be considered in the time sequence. However, if longer-term variations in power output exist, such as those caused by global warming trends, how can these oscillations with contrasting time scales be effectively captured?

To address the challenges mentioned above, many efforts have been made to enhance the feature extraction and pattern recognition capabilities of ANN models in power output prediction. For example, Abdoos (2016) trained a single-layer MLP using measured data from two real wind farms in Spain and the U.S. During the training process, the variational mode decomposition technique was applied to the wind farm power output time series to improve the feature extraction performance of the MLP. However, owing to the intrinsic limitations of MLPs in predicting sequential data, studies in wind turbine/farm power prediction typically use CNN or RNN-based ANN architectures to capture the temporal or spatiotemporal correlations in the input data sequences.

Although CNN is commonly used in image processing tasks, where the input is a two-dimensional matrix, it can also be applied to TSA with the manipulation of time sequence datasets. Zhu et al. (2017) trained a CNN model for wind farm power output using real power data collected in Belgium. Wang et al. (2017) also adopted CNN for the TSA model in power prediction tasks. In their work, the wind farm power history was preprocessed using wavelet transformation into different frequency components, which enhanced the feature extraction performance of the CNN model. Later, Hong and Rioflorido (2019) extended the use of CNN for wind farm power prediction to a 24-hour ahead wind power forecast, while Yu et al. (2019b) further enhanced the CNN approach by interpreting the spatial arrangement of turbines as two-dimensional images and treating the temporal variation of wind farm power as channels.

Although using CNN for TSA tasks is possible, directly adopting RNN-based architectures, such as LSTM and GRU, appears to be a more natural option (Yu et al., 2019a; Zhang et al., 2019). These architectures can also be easily extended to bi-directional models by feeding both the time sequence and its inverse into the model (Wang et al., 2023a; Xiong et al., 2023).

However, RNN-based architectures often suffer from insufficient sequential pattern recognition and difficulty in capturing long-term dependencies. To address these issues, a more robust approach that combines both CNN and RNN architectures has been developed. In these studies, CNN is used to efficiently extract temporal or spatial correlations from time sequences (Yu et al., 2020). In addition, the hybrid CNN-RNN architecture can be fed with preprocessed data to further enhance its feature extraction capability (Zhang et al., 2022; Zhao et al., 2023; Qu et al., 2024).

3.2.2 Correlation TSA

As introduced earlier, although performing a pure TSA in wind turbine/farm power prediction tasks is possible—i.e., predicting future power values purely from historical data—it is inherently challenging. This is because the power output of a wind turbine or farm can also be influenced by other factors or variables, which means these correlated variables should also be included as input features in the ANN models. For correlation TSA applied to wind turbine/farm power predictions, the goal is to establish the following mapping relation:

$$P^t = \mathrm{ANN}\left\{\left(V_1, V_2, \cdots, V_M\right)^{t-1}, \cdots, \left(V_1, V_2, \cdots, V_M\right)^{t-N}\right\} \quad (7)$$

where V represents a variable, with subscripts used to differentiate among various correlated variables.

Typically, the velocity and direction of the incoming wind are critical factors to consider. Yan et al. (2019) trained an MLP model using wind speed and direction as input features, with the power output of wind farms (the Lillgrund wind farm in Sweden and the Nørrekær onshore wind farm in Denmark) as the output. A transfer learning technique was applied, demonstrating the model's potential for application across different wind farms. Similarly, Sun et al. (2020) trained an MLP model to predict the power generation of a real wind farm in China, incorporating wind velocity, wind direction, and turbine yaw angle as input features. Under this framework, geometrical features like blockage ratio (BR) and blockage distance (BD) (Yan, 2018) or atmospheric characteristics (Optis and Perr-Sauer, 2019) can also be easily integrated into the input data, and the overall performance of the trained model could be enhanced.

The aforementioned studies predominantly utilized the fundamental MLP architecture for predictions. However, these MLP models can also be substituted with CNN or RNN-based architectures to enhance feature extraction performance. For example, Kou et al. (2020) developed an ANN model to forecast wind speed at specific turbine locations within a wind farm. Their study introduced a hybrid

ANN architecture that combined CNN and GRU, where incoming wind characteristics such as velocity and direction were represented as sequences of three-channel images. Kisvari et al. (2021) expanded the input features to include generator and gearbox temperatures, which were fed into a GRU network. This inclusion significantly improved the ANN model's overall performance. Similarly, Gu et al. (2021) employed an LSTM model to establish a mapping relation between various input features—wind speed, wind direction, air pressure, temperature, and humidity—and the output label, i.e., wind farm power.

Recent advancements in ANN architectures designed for NLP applications, such as the self-attention mechanism, have also been adopted by researchers for wind turbine and power prediction tasks. Nascimento et al. (2023) developed a novel transformer-based ANN architecture combined with wavelet transform to predict wind speed and power for the next six hours using multiple meteorological variables as input features. Similarly, Wang et al. (2023b) enhanced the forecasting accuracy of an LSTM-based encoder-decoder model for wind farm power output by integrating static information about wind turbines with meteorological data.

In addition to the attention mechanism, GNNs have also been employed in wind turbine and wind farm power prediction tasks, leveraging their flexibility in representing spatial characteristics. Bentsen et al. (2022) integrated an attention mechanism into GNN models to predict the power production of wind turbines within wind farms. The training dataset was generated using AWMs through the open-source package FLORIS (https://github.com/NREL/floris). The model used turbine locations, wind speed, and direction as input features, with turbine power output as the output variable. Results demonstrated that the GNN-based model outperformed bi-LSTM and MLP models in accuracy. Similarly, Li (2022) applied GNNs to short-term wind power forecasting tasks, highlighting their potential in this domain. The input features of the GNN model include wind speed, air density, historical wind power, and the historical wind power of adjacent turbines. Santos et al. (2024) trained a GNN model using layout geometries and inflow conditions as input features, with the outputs being power production and fatigue loads. In this study, the training data was generated from AWMs using the FLORIS open-source package.

Non-neural network machine learning techniques have also been successfully applied to power prediction tasks. Yin and Zhao (2019) predicted both the power output and structural fatigue of an offshore wind farm using various machine learning approaches, including MLP, RF, SVM, and RNN. The training dataset was generated from AWMs using FLORIS. He et al. (2022) employed SVR to predict the fatigue load and power output of wind turbines, using the velocity and turbulence intensity of incoming wind and the turbine yaw angle as input features. Additionally, the XGBoost algorithm, introduced earlier, has been applied to power prediction tasks (Nakhchi, 2023; Cakiroglu et al., 2024), demonstrating high accuracy and significant promise in this domain.

3.3 Wake field reconstruction

The studies categorized here involve a reconstruction process of the wake field, where a complete wake field is predicted or visualized using a trained data-driven model. Within this category, two subgroups are identified and discussed separately in this review: ROM-based methods and direct field reconstruction methods.

3.3.1 ROM-based methods

In this subgroup of wake field reconstruction methods, lower-order representations of the actual flow field are employed during the training of ANNs. Typically, a mode-decomposition technique is applied to the flow field, allowing the complete spatial-temporal wake field to be represented as a combination of time-independent modes and their corresponding time-dependent coefficients. This approach simplifies the initial time-dependent wake field reconstruction task into a TSA task, where only the time-varying variables need to be predicted by the ANNs.

This approach has been successfully applied to reconstruct wake fields for simple geometries. For example, Yousif and Lim (2024) developed a reduced-order model for the turbulent wake of a finite wall-mounted square cylinder using ANNs. The training data was generated through RANS simulations incorporating an IDDES turbulence model. The proper orthogonal decomposition (POD) method was utilized to extract the time coefficients of each mode, and an LSTM architecture was subsequently employed to predict the evolution of these coefficients based on their historical sequences.

In the field of wind turbine wake prediction, this analytical framework has also been effectively utilized. Zhang and Zhao (2020) employed SOWFA to generate an unsteady wake field for nine wind turbines under neutral ABL conditions. The POD method was applied to represent the wake field as separate modes, and an LSTM model was used to predict the time coefficients of each POD mode. The POD-LSTM method exhibited a higher accuracy than AWMs. Similar studies within this framework can be found in Ali et al. (2021), Geibel and Bangga (2022), Guo et al. (2023), Zhou et al. (2023), Luo et al. (2024). An illustration of this framework is illustrated in Figure 7.

Despite effectively representing the initial complex wake field, these ROM-based methods for wake reconstruction suffer from limited generalization ability. Specifically, the trained ANN models may be highly case-specific and difficult to apply to different wind turbine scenarios.

Instead of relying on traditional mode-decomposition methods, ANNs can be used directly to reduce the order of

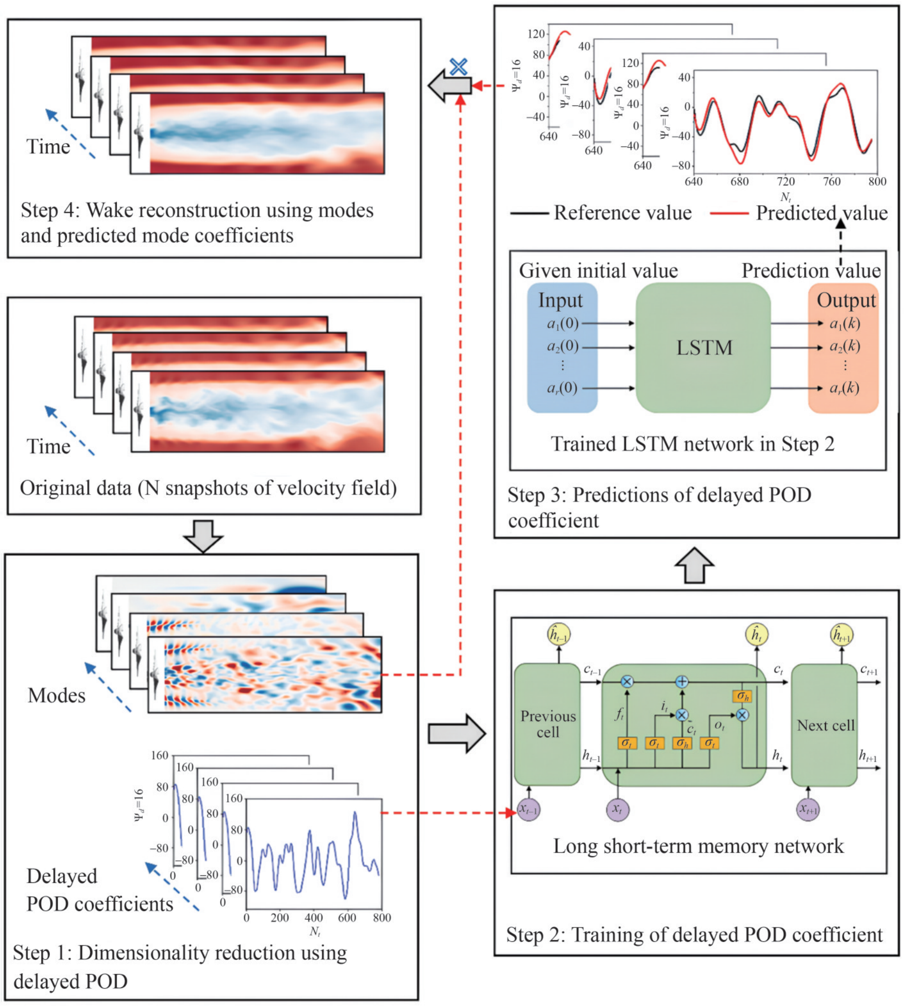

Figure 7 Illustration of the ROM-based models, specifically a POD-LSTM model, for the wake field reconstruction (Zhou et al., 2023)

the initial flow field. For example, Ashwin Renganathan et al. (2022) designed a deep convolutional autoencoder neural network to effectively map high-dimensional LiDAR measurements into a lower-dimensional space. An MLP was then used to learn the mapping relationship between the wind turbine state and the lower-dimensional features. These lower-dimensional features were subsequently transformed back into the original dimensional space using a decoder network with multiple layers of CNNs. This alternative ROM-based method, which leverages ANNs for reduced-order representation of high-dimensional data, shows great promise and warrants further investigation.

3.3.2 Direct wake reconstruction

Leveraging the capability of ANNs to learn and represent nonlinear, high-dimensional mapping relationships, direct reconstruction of wind turbine wakes is achievable by designing an ANN that maps spatial and temporal coordinates directly to flow quantities, such as velocity and turbulence. Specifically, an ANN model can be trained to use (x, y, z, t) as input and produce (U, V, W, TKE) of the flow field as output. This allows the complete wake field at any given time to be reconstructed using the trained ANN model with arbitrary (x, y, z, t) inputs. This mapping relationship can be expressed as:

$$(U, V, W, \mathrm{TKE}) = \mathrm{ANN}(x, y, z, t) \tag{8}$$

A straightforward approach for this task is to utilize an MLP architecture. Ti et al. (2020) employed an MLP model for wind turbine wake prediction, where the inflow wind velocity and turbulence intensity at the turbine hub height were selected as inputs, and the velocity deficit and added TKE in the wake field were used as outputs. The dataset for training the NN was generated using RANS simulations, with the ADM applied for rotor modeling. A specialized training technique, termed sub-model training, was employed. This technique involves using multiple ANN models (2000 in the study) to predict different regions of the wake field individually, which are subsequently merged to reconstruct the complete wake field. The mapping relationship established in this work is illustrated in Figure 8.

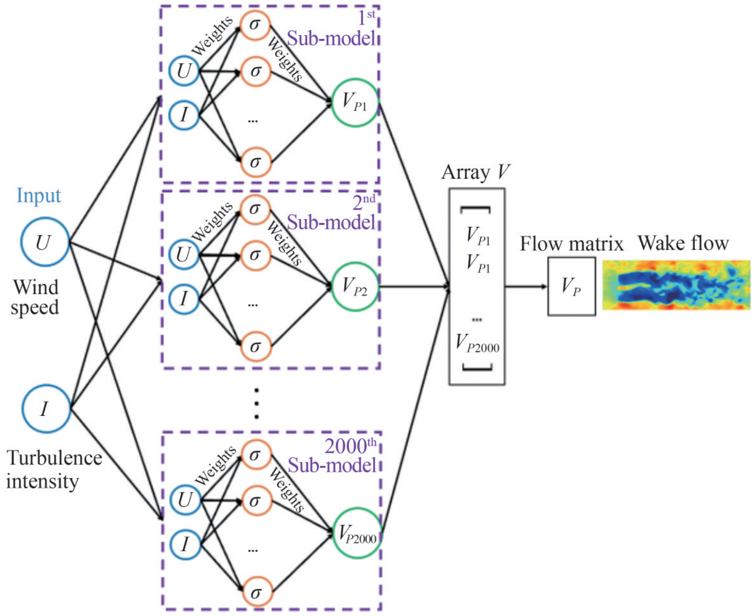

Figure 8 Illustration of the ANN architecture proposed in Ti et al. (2020) for direct wake field reconstruction

However, while this technique can effectively improve the learning efficiency and prediction accuracy of ANN models, its theoretical foundation warrants further investigation. This is because the entire wake field is fundamentally governed by a unified model, namely the N–S equations. Consequently, the use of the sub-model training technique raises questions regarding its interpretability. In addition, wake superposition models were employed in that study to predict the wake fields of multiple wind turbines.

Subsequent studies within the framework of direct wake reconstruction have mainly focused on enhancing the feature extraction capabilities of models by employing more advanced ANN architectures, such as CNNs, GANs, and encoder-decoder frameworks. However, as discussed in Ti et al. (2020) and mentioned at the outset of this review, the overall accuracy of an ANN model designed for wind turbine wake prediction is determined by at least two critical factors: 1) the accuracy of the dataset used for training the network and 2) the performance of the designed ANN architecture. A summary of studies utilizing the direct wake reconstruction framework, along with relevant information, is presented in Table 1.

3.4 Incorporation of physical knowledge

All the studies mentioned above related to wind turbine wake modeling can be classified as pure data-driven modeling. This is because they focus solely on learning the mapping relationship between input features and output labels from the data without incorporating any further information about the physical processes that govern the data generation. In other words, the trained neural networks (NNs) only capture the mapping relationship in the data but do not necessarily represent the true underlying physics. For example, in the case of incompressible fluids, the velocity field (U) at any given spatial and temporal coordinates can be obtained by numerically solving the RANS equations, provided the initial and boundary conditions are known. Thus, the relationship between (x, y, z, t) and U can be regarded as a mapping relation governed by the RANS which can be written as:

$$U = \mathrm{RANS}(x, y, z, t) \tag{9}$$

However, under the pure data-driven approach, the trained NN mapping can be expressed as:

$$U = \mathrm{NN}(x, y, z, t) \tag{10}$$

Readers can immediately recognize that the two mapping relations—RANS and NN—are not necessarily equivalent. This raises a concern that the trained ANN models may not accurately reflect the real physics, which could undermine their reliability in predicting unseen datasets. To address this issue, researchers have sought to incorporate knowledge of physical processes into NN, ensuring that the trained models not only represent the mapping relation between input and output data but also capture the underlying physical processes governing the data.

In this review paper, two distinct frameworks for incorporating physical process information are identified and discussed: the physics-guided approach and the physics-informed approach. While both frameworks integrate physical knowledge into NN, they differ fundamentally in how the physical information is incorporated—either into the input features or the output labels.

Table 1 Summary of studies under the framework of direct wake reconstruction

Literature	Training data	NN structure	Input	Output	Training technique
Ti et al. (2020) Ti et al. (2021)	RANS/ADM	MLP	Inflow wind velocity; turbulence intensity (TI) at the hub height	Velocity deficit; added TKE	Sub-model training
Zhang et al. (2021)	LES/ASM	Autoencoder CNN	Five snapshots of the instantaneous velocity field.	3D time-averaged velocity fields	
Anagnostopoulos and Piggott (2022), Anagnostopoulos et al. (2023)	AWMs: Gaussian model and cur model	CNN	Wind velocity, TI, and yaw angle	Wake velocity	Transfer learning
Li et al. (2022a)	LES/ALM	BiCNN	Historical flow fields and inflow velocity	Future flow field	
Lejeune et al. (2022)	LES/BEM	MLP	Turbine loads and operating settings	Transverse velocity component	
Zhang and Zhao (2022)	LES/ALM	Deep Convolutional Conditional Generative Adversarial Network (DC-CGAN)	Inflow wind profiles and yaw angle	Streamwise and spanwise velocity fields	
Luo et al. (2022)	LES/ADM	MLP	Inflow velocity	Wake velocity	Wind-multiplier method
Yang et al. (2022)	LES/ALM	MLP	Inflow hub-height velocity, TI, and yaw angle	Velocity deficit; TI	Sub-model training
Pawar et al. (2022)	AWMs: Gaussian model and cur model	MLP	Inflow wind speed, TI, and yaw angle	Wake velocity	
Purohit et al. (2022)	RANS/BEM	SVR, XGBoost, MLP	Inflow wind speed, thrust, turbulence intensity, and spatial coordinates	Wake velocity, TI	
Yang et al. (2023b)	RANS/ALM	MLP	Hub-height wind speed and turbulence intensity	Wake velocity, TI	
Nakhchi (2023)	ALM/LES	XGBoost, MLP	Inflow, yaw angle, TI, and CP	Wake velocity	Sub-model training
Li et al. (2023)	RANS/ADM	GNN	Inlet velocity, TI, yaw angle	Wake velocity	
Yang et al. (2023a)	RANS/ADM	MLP	Inlet velocity, TI	Wake velocity, TI	
Romero et al. (2024)	RANS/ADM	Deep convolutional hierarchical encoder-decoder neural network	2D top-view of turbine locations, undisturbed (free stream) wind velocity	Wake velocity	
Li et al. (2024b)	RANS/ADM & AWMs	Transformer-mixed conditional GAN	Inflow velocity, turbulence, and yaw angle	Wake velocity, TI	Pretraining-finetuning

3.4.1 Physics-guided NNs (PGNNs)

The physics-guided approach incorporates physical information by adding additional input features, allowing the trainable parameters of the NN model to be adjusted according to, or guided by, these features. In these studies, simplified mathematical models, such as AWMs, are often used to generate a lower-fidelity representation of the real output label. The initial input features are then combined with this lower-fidelity representation and fed into the NNs to establish the mapping relation with the real output label. In this way, physical knowledge is embedded into the resulting NN model.

Here, we use the work of Guo et al. (2022a) as an example to explain this framework in detail. Guo et al. (2022a) designed a physics-guided NN to predict the short-term wind power output of a real wind farm. For the NN, the input consisted of the 24-hour historical time sequences of incoming wind velocity (U) and wind direction (dir), while the output was the corresponding 24-hour time sequence of power output. For pure data-driven modeling approaches, this task could be achieved using an RNN-based NN architecture, as discussed in Section 3.2.2, and the mapping relation can be written as:

$$\mathrm{NN}\left(U^{t}, \mathrm{dir}^{t}, U^{t+1}, \mathrm{dir}^{t+1}, \cdots, U^{t+24}, \mathrm{dir}^{t+24}\right) = P_{\mathrm{real}}^{t}, P_{\mathrm{real}}^{t+1}, \cdots, P_{\mathrm{real}}^{t+24} \tag{11}$$

However, in this work (Guo et al., 2022a), the authors further fed the neural networks with a power sequence predicted by a physical model, using AWMs combined with wind turbine power curves. The new mapping relation can then be formulated as:

$$\mathrm{NN}\left(U^{t}, P_{\text{model}}^{t}, U^{t+1}, P_{\text{model}}^{t+1}, \cdots, U^{t+24}, P_{\text{model}}^{t+24}\right) = P_{\text{real}}^{t}, P_{\text{real}}^{t+1}, \cdots, P_{\text{real}}^{t+24} \tag{12}$$

In which the P_{model}^{t} was calculated using AWMs based on U^t and dir^t. By doing this, the physical information of the power sequence is incorporated into the NNs, and the model variables can be adjusted with the guidance of this physical knowledge during the training process. Studies that adopted the same framework are summarized in the following table.

3.4.2 Physics-informed NNs (PINNs)

Instead of incorporating physical knowledge by adding lower-fidelity models to the input features, as in PGNNs, physical knowledge or constraints can be applied to an ANN model by modifying the loss function. Specifically, in addition to the label loss, which represents the difference between the NN prediction and the true label, a physical loss can be added to the total loss. This physical loss compares the true governing equations of the physical process with the NN-reconstructed governing equations. Therefore, the total loss of the NN can be written as:

$$\text{Loss} = \text{Loss}_{\text{label}} + \text{Loss}_{\text{physics}} \tag{13}$$

Therefore, with the backpropagation of the loss function, a trained PINN can not only represent the mapping relation between the input features and output labels but also serve as an NN surrogate for the underlying physics. As a result, the likelihood of overfitting in the trained NN model is greatly reduced, while its generalization performance is significantly enhanced (Raissi and Karniadakis, 2018; Raissi et al., 2019).

It is also worth mentioning that the construction of the physical loss, often referred to as PDE loss, requires performing differential operations on the NN predictions. This, however, can be easily accomplished by leveraging the automatic differentiation (AD) functionality available in deep learning frameworks such as TensorFlow, PyTorch, and PaddlePaddle. In general, in the field of wind turbine wake reconstruction, the process of constructing the total loss can be described as follows:

1) Establish an ANN model with input variables (x, y, z, t) and output variables (U, V, W, p);

2) Perform forward propagation through the NN;

3) Calculate the label loss by directly comparing the predicted and actual values of (U, V, W, p);

4) Calculate the first and second derivatives of (U, V, W, p) with respect to (x, y, z, t) using AD;

5) Combine the derivatives to form the governing PDE, i.e., the N–S equations;

6) Move all terms of the governing equations to one side, with the residuals representing the physical losses;

7) Combine the label loss and the physical loss to obtain the total loss.

A schematic representation of this procedure is illustrated in Figure 9.

According to the previous discussion of PINNs, readers may wonder whether an NN model can be trained using only physical losses at randomly sampled points within the domain of interest. The answer is yes, which leads to the framework of solving a PDE using PINNs. However, the details of this approach are beyond the scope of this review, and readers are encouraged to refer to Ref (Raissi et al., 2019). In theory, for wind turbine wake prediction tasks, training NNs without label losses is also possible. However, based on the authors' experience, owing to the complexity of wind turbine flow, label loss is essential—at least to expedite the training process. Further research is needed to validate this assertion.

Zhang and Zhao (2021) trained a PINN model using data collected from LiDAR measurements. The data points were sparsely distributed on a horizontal plane in the wind field, and the two-dimensional N–S equations were employed to construct the physical loss in the PINN. The accuracy of the trained PINN model was then evaluated by comparing the NN-predicted wind field with LES simulations, with good agreement reported. A similar study was conducted by Wang et al. (2024), focusing on the wake trajectory under wind turbine yaw conditions. Subsequently, three-dimensional N–S equations and the effect of wind turbines were incorporated into the physical loss to further enhance the model accuracy and performance within the same framework (Zhang and Zhao, 2023). In that work, the influence of the wind turbine blades on the surrounding airflow was

Table 2 Summary of studies within the framework of PGNNs

Literature	Input	NN structure	Output
Li et al. (2022b)	Wake field generated by AWM	CNN	Wake field generated by LES+ALM
Li et al. (2024a)	Incoming wind velocity and turbulent fluctuation field	CNN	Wake field generated by LES+ADM
Zehtabiyan-Rezaie et al. (2023)	Incoming wind velocity, block ratio, and efficiency of wind turbines derived from analytical wake models	XGBoost	Efficiency of turbines calculated by RANS+ADM
Santoni et al. (2024)	Wake field generated by AWM	CNN	Wake field generated by LES+ASM

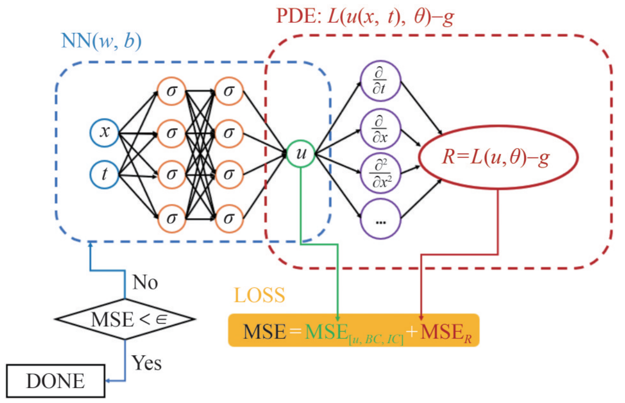

Figure 9 Schematic of loss construction in PINNs
Note that B.C. represents the boundary condition, and I.C. represents the initial condition.

modeled using the ADM, and an additional body force term was added to the physical loss. As a result, the trained PINN model could predict the wake field at the hub height of multiple wind turbines with reasonable accuracy. Sun et al. (2024) trained a PINN model to reconstruct the three-dimensional wake field behind a wind turbine. The training dataset was generated by LES/ALM simulations, and the three-dimensional N–S equations were used to derive the physical loss. During the training stage, only data from sparse spatial locations were fed into the model. The predicted wake field, obtained using the trained model, was compared with the results from MLP and LSTM models, revealing that the incorporation of the physical loss significantly enhanced the model accuracy. An illustration of this framework applied to wind turbine wake predictions is illustrated in Figure 10.

Although using the precise PDEs that govern the physical process is common practice under the framework of PINNs, the use of simplified equations is also possible. Zhou et al. (2022) designed a PINN model that incorporated AWMs in the construction of the physical loss. Although the authors referred to their model as "physics-guided", in this review it will still be considered "physics-informed", as the physical knowledge was introduced into the NNs by modifying its loss function. The study found that using AWMs also allowed the NN to regulate the optimization direction (Zhou et al., 2022). An illustration of this type of PINN model is illustrated in Figure 11.

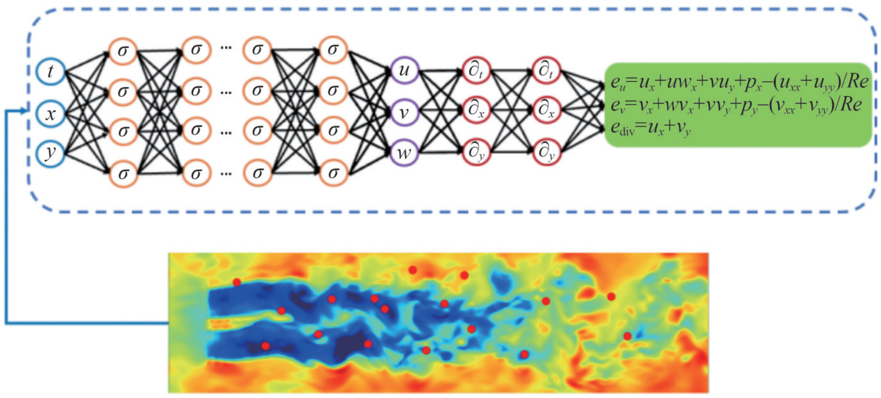

Figure 10 Application of PINNs in the wind turbine wake reconstructions using governing PDEs as physical constraints

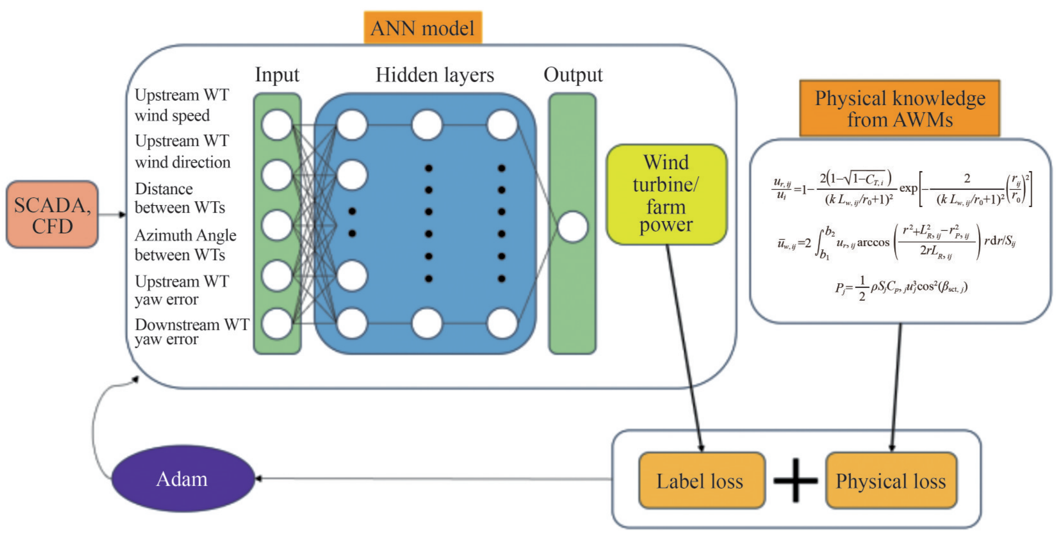

Figure 11 Application of PINNs in wind turbine wake reconstruction using AWMs as physical constraints

4 Prospects

The adoption of data-driven approaches has opened new technical pathways for wind turbine wake modeling. While researchers have achieved promising results using various machine learning frameworks, several key issues that directly impact the applicability of data-driven models remain unsolved, or at least insufficiently discussed, and thus require further investigation. Some of these issues are identified and summarized below:

1) Prediction of three-dimensional wind turbine wake: Most existing studies have focused on two-dimensional wake prediction at the turbine hub height. However, with the increasing blade length of wind turbines, three-dimensional effects resulting from large turbulence structures in the ABL are becoming increasingly important. Therefore, adopting three-dimensional wake models is crucial for accurately predicting the wake field of single or multiple wind turbines. Predicting a three-dimensional fluid field, however, remains an extremely challenging task for data-driven approaches. Although some studies (Liu et al., 2020; Kim et al., 2021; Pawar et al., 2022) have shown promising results for predicting three-dimensional flows, these are limited to simple flow scenarios. When it comes to the three-dimensional wake fields of wind turbines, the presence of rotating tip helical vortices and the interactions among various flow scales (Veers et al., 2019) make this task highly challenging. Therefore, more research efforts are needed to improve the prediction of three-dimensional wind turbine wakes using data-driven approaches.

2) NN-based wake superposition models. Most existing data-driven wind turbine wake models have been trained on datasets from a single wind turbine wake, with the wake field of multiple wind turbines then obtained by combining the single turbine model with analytical wake superposition models (AWSMs), such as linear superposition, root-sum-square superposition, and largest deficit superposition (Vogel and Willden, 2020). However, the applicability and accuracy of these AWSMs are heavily dependent on their underlying assumptions, and their performance can be enhanced by integrating data-driven approaches. For example, GNNs are well known for their ability to represent graph data. This capability can be effectively applied to wake superposition problems by training a GNN model that takes the relative locations of different wind turbines as one of the input features and outputs the complete wake field. Furthermore, developing NN-based superposition models for three-dimensional wind turbine wakes could be the next step, building on the two-dimensional AWSMs, though the complexity of this task could be significantly higher.

3) Uncertainty quantification. Another major issue that affects the overall reliability of a trained data-driven model is the uncertainty in its predictions. As mentioned at the beginning of this review, the accuracy of a trained data-driven model is influenced by at least two factors: the quality of the data and the performance of the model. While uncertainties in the data, such as those generated by CFD simulations, can be readily assessed in wake studies (Burmester et al., 2020; Ye et al., 2023a, 2023b), and the uncertainties in NN models can be partially quantified (e.g., by adopting a Bayesian NN structure) (Arbel et al., 2023), no existing framework effectively integrates these two sources of uncertainty. Therefore, further research is needed to address this issue and enhance the overall credibility of data-driven wind turbine wake models.

5 Conclusions

In this paper, existing studies on data-driven modeling of wind turbine wakes were comprehensively reviewed. After a brief introduction to the fundamental machine learning concepts, the reviewed works were classified into four categories: data-driven analytical wake models, wind turbine power prediction, wake field reconstruction, and incorporation of physical knowledge. For each of the categories, it can be summarized as follows:

• Data-driven analytical wake models: the goal of this category is to obtain explicit mathematical wake models;

• Wind turbine power prediction: two sub-categories are further identified in the current paper, i.e. "pure TSA" and "correlation TSA". For "pure TSA", it predicts the power generation of wind turbines using historical power sequences, while for "correlation TSA", the goal is to predict the power generation of wind turbines using historical sequences of multiple correlated variables;

• Wake field reconstruction: two sub-categories are identified are further identified, i. e. "ROM-based methods" and "direct wake reconstruction". The "ROM-based methods" aim to reconstruct the full wake field using lower-order representations of the initial flow field. The initial spatial-temporal prediction task is decoupled, so only the time-dependent variables need to be modeled, while the "direct wake reconstruction" attempt to establish the mapping relation between the spatial-temporal coordinates and the corresponding flow quantities is established directly using a data-driven model, such as MLP, CNN, and RF;

• Incorporation of physical knowledge: two sub-categories are identified are further identified, i.e. "Physics-guided NN" and "Physics-informed NN". For physics-guided NNs, data generated by lower-fidelity models are fed into ANNs as input features to guide the optimization of the model, while for physics-informed NNs, physical constraints, i.e., the governing PDEs, are reconstructed, and their residuals are added to the loss functions to regulate the optimization of NNs.

Abbreviations

ABL	Atmospheric boundary layer
AD	Automatic differentiation
ADM	Actuator disc method
ALM	Actuator line method
ANN	Artificial neural network, interchangeable with NN
ASM	Actuator surface method
AWM	Analytical wake model
AWSM	Analytical wake superposition models
BD	Blockage distance
BR	Blockage ratio
CFD	Computational fluid dynamics
CNN	Convolutional neural networks
GNN	Graph neural networks
GRU	Gated recurrent unit
GP	Generic programming
LES	Large eddy simulations
LSTM	Long short-term memory
MLP	Multilayer perceptron
MOCO	Multi-objective combinatorial optimization
NLP	Natural language processing
NN	Neural network, interchangeable with ANN
N–S	Navier–Stokes
ON	Operator network
PCA	Principal component analysis
PGNN	Physics-guided neural networks
PINN	Physics-informed neural networks
POD	Proper orthogonal decomposition
RANS	Reynolds-averaged Navier–Stokes
RF	Random forest
RNN	Recurrent neural networks
ROM	Reduced-order modeling
SA	Simulated annealing
SR	Symbolic regression
SVM	Support vector machine
SVR	Support vector regression
TSA	Time sequence analysis

Funding Supported by the National Natural Science Foundation of China under Grant No. 52131102.

Competing interest Decheng Wan is an editorial board member for the Journal of Marine Science and Application and was not involved in the editorial review, or the decision to publish this article. All authors declare that there are no other competing interests.

References

Abdoos AA (2016) A new intelligent method based on combination of VMD and ELM for short term wind power forecasting. Neurocomputing 203: 111-120

Ali N, Calaf M, Cal RB (2021) Cluster-based probabilistic structure dynamical model of wind turbine wake. Journal of Turbulence 22(8): 497-516

Amiri MM, Shadman M, Estefen SF (2024) A review of physical and numerical modeling techniques for horizontal-axis wind turbine wakes. Renewable and Sustainable Energy Reviews 193: 114279

Anagnostopoulos S, Piggott M (2022) Offshore wind farm wake modelling using deep feed forward neural networks for active yaw control and layout optimisation. Journal of Physics 2151: 012011

Anagnostopoulos SJ, Bauer J, Clare MCA, Piggott MD (2023) Accelerated wind farm yaw and layout optimisation with multi-fidelity deep transfer learning wake models. Renewable Energy 218: 119293

Arbel J, Pitas K, Vladimirova M, Fortuin V (2023) A primer on Bayesian neural networks: Review and debates. arXiv preprint, arXiv: 2309.16314, Sep 2023

Ashwin Renganathan S, Maulik R, Letizia S, Iungo GV (2022) Data-driven wind turbine wake modeling via probabilistic machine learning. Neural Comput & Applic 34(8): 6171-6186

Bastankhah M, Porté-Agel F (2014) A new analytical model for wind-turbine wakes. Renewable Energy 70: 116-123

Bentsen LØ, Warakagoda ND, Stenbro R, Engelstad P (2022) Wind park power prediction: Attention-based graph networks and deep learning to capture wake losses. Journal of Physics 2265: 022035

Breiman L (2001) Random forests. Machine Learning 45: 5-32

Burmester S, Vaz G, el Moctar O (2020) Towards credible CFD simulations for floating offshore wind turbines. Ocean Engineering 209: 107237

Cakiroglu C, Demir S, Ozdemir MH, Aylak BL, Sariisik G, Abualigah L (2024) Data-driven interpretable ensemble learning methods for the prediction of wind turbine power incorporating SHAP analysis. Expert Systems With Applications 237: 121464

Calaf M, Meneveau C, Meyers J (2010) Large eddy simulation study of fully developed wind-turbine array boundary layers. Physics of Fluids 22(1): 015110

Chen T, Chen H (1995) Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems. IEEE Trans Neural Netw 6(4): 911-917

Chen T, Guestrin C (2016) XGBoost: A scalable tree boosting system. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, San Francisco, 785-794

Chung J, Gulcehre C, Cho K, Bengio Y (2014) Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint, arXiv: 1412.3555, Dec 2014

Draper M, Guggeri A, Mendina M, Usera G, Campagnolo F (2018) A Large Eddy Simulation-Actuator Line Model framework to simulate a scaled wind energy facility and its application. Journal of Wind Engineering and Industrial Aerodynamics 182: 146-159

Gajendran MK, Kabir IFSA, Vadivelu S, Ng EYK (2023) Machine learning-based approach to wind turbine wake prediction under yawed Conditions. Journal of Marine Science and Engineering 11: 2111

Gao Z, Cecati C, Ding SX (2015) A survey of fault diagnosis and fault-tolerant techniques—Part I: Fault diagnosis with model-based and signal-based approaches. IEEE Trans Ind Electron 62(6): 3757-3767

Ge M, Wu Y, Liu Y, Li Q (2019) A two-dimensional model based on the expansion of physical wake boundary for wind-turbine wakes. Applied Energy 233: 975-984

Geibel M, Bangga G (2022) Data reduction and reconstruction of wind turbine wake employing data driven approaches. Energies 15: 3773

Gu B, Zhang T, Meng H, Zhang J (2021) Short-term forecasting and uncertainty analysis of wind power based on long short-term memory, cloud model and non-parametric kernel density estimation. Renewable Energy 164: 687-708

Guo NZ, Shi KZ, Bo L, Qi LW, Wu HH, Zhang ZL, Xu JZ (2022a) A physics-inspired neural network model for short-term wind power prediction considering wake effects. Energy 261: 125208

Guo NZ, Zhang MM, Li B (2022b) A data-driven analytical model for wind turbine wakes using machine learning method. Energy Conversion and Management 252: 115130

Guo Z, Xu L, Zhou G, Zhang K (2023) A non-intrusive reduced-order model for wind farm wake analysis based on SPOD-DNN. Wind Engineering 47(4): 852-866

He R, Yang H, Shilin S, Lu L, Sun H, Xiaoxia G (2022) A machine learning-based fatigue loads and power prediction method for wind turbines under yaw control. Applied Energy 326: 120013

Hearst MA (1998) Support vector machines. IEEE Intell Syst Their Appl 13(4): 18-28

Helvig S de J, Vinnes MK, Segalini A, Worth NA, Hearst RJ (2021) A comparison of lab-scale free rotating wind turbines and actuator disks. Journal of Wind Engineering and Industrial Aerodynamics 209: 104485

Hochreiter S, Schimidhuber J (1997) Long short-term memory. Neural Computation 9: 1835-1780

Hong YY, Rioflorido CLPP (2019) A hybrid deep learning-based neural network for 24-h ahead wind power forecasting. Applied Energy 250: 530-539

Isele D, Rahimi R, Cosgun A, Subramanian K, Fujimura K (2018) Navigating occluded intersections with autonomous vehicles using deep reinforcement learning. arXiv preprint, arXiv: 1705.01196v2, Feb 2018

Kabir IFSA, Safiyullah F, Ng EYK, Tam VWY (2020) New analytical wake models based on artificial intelligence and rivalling the benchmark full-rotor CFD predictions under both uniform and ABL inflows. Energy 193: 116761

Kim H, Kim J, Won S, Lee C (2021) Unsupervised deep learning for super-resolution reconstruction of turbulence. J Fluid Mech 910: A29

Kisvari A, Lin Z, Liu X0 (2021) Wind power forecasting-A data-driven method along with gated recurrent neural network. Renewable Energy 163: 1895-1909

Kober J, Peters J (2014) Reinforcement learning in robotics: A Survey. In: Learning Motor Skills. Springer International Publishing, Cham, pp 9-67

Kou P, Wang C, Liang D, Cheng S, Gao L (2020) Deep learning approach for wind speed forecasts at turbine locations in a wind farm. IET Renew Power Gener 14(13): 2416-2428

LeCun Y, Bengio Y (1998) Convolutional networks for images, speech, and time-series. In: The handbook of brain theory and neural networks. MIT Press, Cambridge, pp 255-258

Lecun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521 (7553): 436-444

Lejeune M, Moens M, Chatelain P (2022) A meandering-capturing wake model coupled to rotor-based flow-sensing for operational wind farm flow prediction. Front Energy Res 10: 884068

Li B, Ge M, Li X, Liu Y (2024a) A physics-guided machine learning framework for real-time dynamic wake prediction of wind turbines. Physics of Fluids 36(3): 035143

Li H (2022) Short-term wind power prediction via spatial temporal analysis and deep residual networks. Frontiers in Energy Research 10: 920407

Li H, Yang Q, Li T (2024b) Wind turbine wake prediction modelling based on transformer-mixed conditional generative adversarial network. Energy 291: 130403

Li R, Zhang J, Zhao X (2022a) Dynamic wind farm wake modeling based on a Bilateral Convolutional Neural Network and high-fidelity LES data. Energy 258: 124845

Li R, Zhang J, Zhao X (2022b) Multi-fidelity modeling of wind farm wakes based on a novel super-fidelity network. Energy Conversion and Management 270: 116185

Li S, Zhang M, Piggott MD (2023) End-to-end wind turbine wake modelling with deep graph representation learning. Applied Energy 339: 120928

Liu B, Tang J, Huang H, Lu XY (2020) Deep learning methods for super-resolution reconstruction of turbulent flows. Physics of Fluids 32(2): 025105

Lu L, Jin P, Pang G, Zhang Z, Karniadakis GE (2021) Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. Nat Mach Intell 3(3): 218-229

Luo Z, Luo W, Xie J, Xu J, Wang L (2022) A new three-dimensional wake model for the real wind farm layout optimization. Energy Exploration & Exploitation 40(2): 701-723

Luo Z, Wang L, Xu J, Wang Z, Yuan J, Tan ACC (2024) A reduced order modeling-based machine learning approach for wind turbine wake flow estimation from sparse sensor measurements. Energy 294: 130772

Martínez-Tossas LA, Churchfield MJ, Leonardi S (2015) Large eddy simulations of the flow past wind turbines: actuator line and disk modeling. Wind Energy 18: 1047-1060

Moussaoui H, El Akkad N, Benslimane M (2023) Reinforcement Learning: A review. IJCDS 13(1): 1465-1483

Myles AJ, Feudale RN, Liu Y, Woody NA, Brown SD (2004) An introduction to decision tree modeling. Journal of Chemometrics 18: 275-285

Nakhchi ME, Naung SW, Rahmati (2023) Wake and power prediction of horizontal-axis wind farm under yaw-controlled conditions with machine learning. Energy Conversion and Management 296: 117708

Naranjo-Torres J, Mora M, Hernández-García R, Barrientos RJ, Fredes C, Valenzuela A (2020) A review of convolutional neural network applied to fruit image processing. Applied Sciences 10: 3343

Nascimento EGS, de Melo TAC, Moreira DM (2023) A transformer-based deep neural network with wavelet transform for forecasting wind speed and wind energy. Energy 278: 127678

Nilsson K, Shen WZ, Sørensen JN, Breton S-P, Ivanell S (2015) Validation of the actuator line method using near wake measurements of the MEXICO rotor. Wind Energy 18: 499-514

Optis M, Perr-Sauer J (2019) The importance of atmospheric turbulence and stability in machine-learning models of wind farm power production. Renewable and Sustainable Energy Reviews 112: 27-41

Pawar S, Sharma A, Vijayakumar G, Bay CJ, Yellapantula S, San O (2022) Towards multi-fidelity deep learning of wind turbine wakes. Renewable Energy 200: 867-879

Popescu MC, Balas VE, Perescu-Popescu L, Mastorakis N (2009) Multilayer Perceptron and Neural Networks. Wseas Transactions on Circuits and Systems 8(7): 579-588

Pujari KN, Miriyala SS, Mitra K (2023) Jensen-ANN: A machine learning adaptation of jensen wake model. IFAC-PapersOnLine 56(2): 4651-4656

Purohit S, Ng EYK, Syed Ahmed Kabir IF (2022) Evaluation of three potential machine learning algorithms for predicting the velocity and turbulence intensity of a wind turbine wake. Renewable Energy 184: 405-420

Qu Z, Hou X, Li J, Hu W (2024) Short-term wind farm cluster power prediction based on dual feature extraction and quadratic decomposition aggregation. Energy 290: 130155

Raissi M, Karniadakis GE (2018) Hidden physics models: Machine learning of nonlinear partial differential equations. Journal of Computational Physics 357: 125-141

Raissi M, Perdikaris P, Karniadakis GE (2019) Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. Journal of Computational Physics 378: 686-707

Romero DA, Hasanpoor S, Antonini EGA, Amon CH (2024) Predicting wind farm wake losses with deep convolutional hierarchical encoder–decoder neural networks. APL Machine Learning 2(1): 016111

Ronneberger O, Fischer P, Brox T (2015) U-Net: Convolutional networks for biomedical image segmentation. arXiv preprint, arXiv: 1505.04597, May 2015

Rumelhart DE, Hintont GE, Williams RJ (1986) Learning representations by back-propagating errors. Nature 323(9): 533-536

Santoni C, Zhang D, Zhang Z, Samaras D, Sotiropoulos F, Khosronejad A (2024) Toward ultra-efficient high-fidelity predictions of wind turbine wakes: Augmenting the accuracy of engineering models with machine learning. Physics of Fluids 36(6): 065159

Santos FDN, Duthé G, Abdallah I, Réthoré P-É, Weijtjens W, Chatzi E, Devriendt C (2024) Multivariate prediction on wake-affected wind turbines using graph neural networks. J Phys: Conf Ser 2647(11): 112006

Scarselli F, Gori M, Ah Chung Tsoi, Hagenbuchner M, Monfardini G (2009) The graph neural network model. IEEE Trans Neural Netw 20(1): 61-80

Shakoor R, Hassan MY, Raheem A, Wu YK (2016) Wake effect modeling: A review of wind farm layout optimization using Jensen's model. Renewable and Sustainable Energy Reviews 58: 1048-1059

Sun H, Qiu C, Lu L, Gao X, Chen J, Yang H (2020) Wind turbine power modelling and optimization using artificial neural network with wind field experimental data. Applied Energy 280: 115880

Sun S, Cui S, He T, Yao Q (2024) An integrated deep neural network framework for predicting the wake flow in the wind field. Energy 291: 130400

Ti Z, Deng XW, Yang H (2020) Wake modeling of wind turbines using machine learning. Applied Energy 257: 114025

Ti Z, Deng XW, Zhang M (2021) Artificial neural networks based wake model for power prediction of wind farm. Renewable Energy 172: 618-631

Troldborg N (2009) Actuator line modeling of wind turbine wakes. PhD thesis, Technical University of Denmark, Copenhagen

Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. 31st Conference on Neural Information Processing Systems, NIPS 2017, Long Beach, USA

Veers P, Dykes K, Lantz E, Barth S, Bottasso CL, Carlson O, Clifton A, Green J, Green P, Holttinen H, Laird D, Lehtomäki V, Lundquist JK, Manwell J, Marquis M, Meneveau C, Moriarty P, Munduate X, Muskulus M, Naughton J, Pao L, Paquette J, Peinke J, Robertson A, Rodrigo JS, Sempreviva AM, Smith JC, Tuohy A, Wiser R (2019) Grand challenges in the science of wind energy. Science 366(6464): eaau2027

Vogel CR, Willden RHJ (2020) Investigation of wind turbine wake superposition models using Reynolds-averaged Navier-Stokes simulations. Wind Energy 23(3): 593-607

Wang D, Yang M, Zhang W (2023a) Wind power group prediction model based on multi-task learning. Electronics 12(17): 3683

Wang H, Li G, Wang G, Peng J, Jiang H, Liu Y (2017) Deep learning based ensemble approach for probabilistic wind power forecasting. Applied Energy 188: 56-70

Wang L, Chen M, Luo Z, Zhang B, Xu J, Wang Z, Tan ACC (2024) Dynamic wake field reconstruction of wind turbine through physics-informed neural network and sparse LiDAR data. Energy 291: 130401

Wang Q, Wang Y, Zhang K, Liu Y, Qiang W, Han Wen Q (2023b) Artificial intelligent power forecasting for wind farm based on multi-source data fusion. Processes 11(5): 1429

Wu Z, Pan S, Chen F, Long G, Zhang C, Yu PS (2021) A comprehensive survey on graph neural networks. IEEE Trans Neural Netw Learning Syst 32(1): 4-24

Xiong J, Peng T, Tao Z, Zhang C, Song S, Nazir MS (2023) A dual-scale deep learning model based on ELM-BiLSTM and improved reptile search algorithm for wind power prediction. Energy 266: 126419

Xu K, Hu W, Leskovec J, Jegelka S (2019) How Powerful are Graph Neural Networks? arXiv preprint, arXiv:1810.00826, Feb 2019.

Yan C (2018) Wind turbine wakes: from numerical modeling to machine learning. PhD thesis, University of Delaware, Newark

Yan C, Pan Y, Archer CL (2019) A general method to estimate wind farm power using artificial neural networks. Wind Energy 22(11): 1421-1432

Yang K, Deng X, Ti Z, Yang S, Huang S, Wang Y (2023a) A data-driven layout optimization framework of large-scale wind farms based on machine learning. Renewable Energy 218: 119240

Yang S, Deng X, Ti Z, Yan B, Yang Q (2022) Cooperative yaw control of wind farm using a double-layer machine learning framework. Renewable Energy 193: 519-537

Yang S, Yang K, Deng X, Yang J (2023b) Smart cooperative control scheme for large-scale wind farms based on a double-layer machine learning framework. Energy Conversion and Management 285: 116949

Ye M, Chen H, Wan D, Ye M, Chen H, Wan D (2024a) Quantification of numerical uncertainties of CFD-predicted wind turbine performance with overset grids. Ocean Engineering 298(December

2023): 117231
Ye M, Chen HC, Koop A (2023a) Verification and validation of CFD simulations of the NTNU BT1 wind turbine. Journal of Wind Engineering and Industrial Aerodynamics 234(February): 105336
Ye M, Chen HC, Koop A (2023b) High-fidelity CFD simulations for the wake characteristics of the NTNU BT1 wind turbine. Energy 265(November 2022): 126285
Ye M, Ma C, Wan D, Chen HC (2024b) A comparative study of different artificial neural-network architectures in the prediction of riser VIV. Proceedings of the Thirty-fourth International Ocean and Polar Engineering Conference, Rhodes, 2770-2777
Yin X, Zhao X (2019) Big data driven multi-objective predictions for offshore wind farm based on machine learning algorithms. Energy 186: 115704
Yousif MZ, Lim HC (2024) Reduced-order modeling for turbulent wake of a finite wall-mounted square cylinder based on artificial neural network. Physics of Fluids 34: 015116
Yu R, Gao J, Yu M, Lu W, Xu T, Zhao M, Zhang J, Zhang R, Zhang Z (2019a) LSTM-EFG for wind power forecasting based on sequential correlation features. Future Generation Computer Systems 93: 33-42
Yu R, Liu Z, Li X, Lu W, Ma D, Yu M, Wang J, Li B (2019b) Scene learning: Deep convolutional networks for wind power prediction by embedding turbines into grid space. Applied Energy 238: 249-257
Yu Y, Han X, Yang M, Yang J (2020) Probabilistic prediction of regional wind power based on spatiotemporal quantile regression. IEEE Trans on Ind Applicat 56(6): 6117-6127
Zehtabiyan-Rezaie N, Iosifidis A, Abkar M (2023) Physics-guided machine learning for wind-farm power prediction: Toward interpretability and generalizability. PRX Energy 2(1): 013009
Zhang J, Yan J, Infield D, Liu Y, Lien F (2019) Short-term forecasting and uncertainty analysis of wind turbine power based on long short-term memory network and Gaussian mixture model. Applied Energy 241: 229-244
Zhang J, Zhao X (2020) A novel dynamic wind farm wake model based on deep learning. Applied Energy 277: 115552
Zhang J, Zhao X (2022) Wind farm wake modeling based on deep convolutional conditional generative adversarial network. Energy 238: 121747
Zhang J, Zhao X (2021) Spatiotemporal wind field prediction based on physics-informed deep learning and LIDAR measurements. Applied Energy 288(February): 116641
Zhang J, Zhao X (2023) Digital twin of wind farms via physics-informed deep learning. Energy Conversion and Management 293: 117507
Zhang W, Lin Z, Liu X (2022) Short-term offshore wind power forecasting-A hybrid model based on Discrete Wavelet Transform (DWT), Seasonal Autoregressive Integrated Moving Average (SARIMA), and deep-learning-based Long Short-Term Memory (LSTM). Renewable Energy 185: 611-628
Zhang Z, Santoni C, Herges T, Sotiropoulos F, Khosronejad A (2021) Time-averaged wind turbine wake flow field prediction using autoencoder convolutional neural networks. Energies 15(1): 41
Zhao Z, Yun S, Jia L, Guo J, Meng Y, He N, Li X, Shi J, Yang L (2023) Hybrid VMD-CNN-GRU-based model for short-term forecasting of wind power considering spatio-temporal features. Engineering Applications of Artificial Intelligence 121: 105982
Zhou H, Qiu Y, Feng Y, Liu J (2022) Power prediction of wind turbine in the wake using hybrid physical process and machine learning models. Renewable Energy 198: 568-586
Zhou J, Cui G, Hu S, Zhang Z, Yang C, Liu Z, Wang L, Li C, Sun M (2020) Graph neural networks: A review of methods and applications. AI Open 1: 57-81
Zhou L, Wen J, Wang Z, Deng P, Zhang H (2023) High-fidelity wind turbine wake velocity prediction by surrogate model based on d-POD and LSTM. Energy 275: 127525
Zhu A, Li X, Mo Z, Wu R (2017) Wind power prediction based on a convolutional neural network. 2017 International Conference on Circuits, Devices and Systems, ICCDS, IEEE, Chengdu, 131-135