Journal of Intelligent Monitoring and Control Systems (JIMCS)

DOI: 10.65098/jimcs.01.2026.01.34



RESEARCH ARTICLE

Wind Speed Prediction Methods and Applications: A Review and Roadmap

Yihe Zou[1], Xuefang Xu[1,*], Guoqian Jiang[1], Yan Wang[2], Wenxian Yang[3], Hamid Reza Karimi[4], Peiming Shi[1]

1 School of Electrical Engineering, Yanshan University, Qinhuangdao 066004, China
2 School of Green Energy and Storage, Lanzhou University of Technology, Lanzhou 730050, China
3 School of Computing and Engineering, University of Huddersfield, Huddersfield HD1 3DH, UK
4 Department of Mechanical Engineering, Politecnico di Milano, Milan 20156, Italy
****Corresponding Author Email: xuefangxu@ysu.edu.cn***

This is an open access article distributed under the Creative Commons Attribution License CC BY 4.0, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

ARTICLE DETAILS

Article History:

Received: 03 Jun 2026
Accepted: 02 Sep 2026
Available online: 30 Oct 2026

Online Article Code

ABSTRACT

Background: Wind speed forecasting (WSF) is critical to the planning and operation of power and energy systems. Despite the growing body of research, no comprehensive review has systematically mapped the state of the art and charted a path forward. This paper fills that gap by providing a holistic review and a strategic roadmap that traces the development of WSF and outlines future research directions. Review Methods: This study systematically surveys the historical evolution of WSF methodologies, tracking the progression from early physical and statistical methods-which relied heavily on expert knowledge-through traditional machine learning (which began to reduce human intervention), to the current dominance of deep learning. The analysis focuses on how these methods have increasingly shifted toward end-to-end prediction pipelines that directly link forecasting models to large-scale wind speed data. Main Findings: The review reveals a clear trajectory: early methods depended on manual feature extraction, while the advent of deep learning in the late 2010s fundamentally reshaped the field by further automating the prediction pipeline. Future research is expected to focus on lightweight real-time models and deep reinforcement learning-driven ensemble methods to address practical engineering challenges. A detailed roadmap is presented to highlight promising research avenues and remaining difficulties. An appendix also provides definitions of forecast time horizons, common evaluation metrics, and relevant publication titles for quick reference. Conclusions and Roadmap Value: The research concludes that wind speed prediction techniques will undoubtedly continue to improve. The proposed roadmap and comprehensive analysis serve as a valuable guide for researchers and engineers, offering a clear vision for the future implementation of WSF technologies in complex engineering settings.

KEYWORDS

Wind Speed Forecasting; Review and Roadmap; Physical and Statistical Methods; Machine Learning; Deep Learning

1. INTRODUCTION

The depletion of fossil resources has led to the utilization of clean energy a sharp increase in emphasis on clean energy (Guo et al., 2024). As clean and renewable energy sources, wind energy and solar energy have reached 12% of the global supply and are continuously increasing, according to statistics from Ember in Global Power Review 2022 (Ember, 2022), as depicted in Figure 1.

Solar energy utilization is limited by battery materials and manufacturing technology, resulting in low conversion efficiency and poor cost-effectiveness. Unlike solar energy, wind energy has fewer limitations, and wind energy generation has higher conversion efficiency. With these advantages, wind energy maintains rapid growth according to GWEC's "2022 Global Wind Energy Report" (Global Wind Energy Council, 2022), as shown in Figure 2.

However, this rapid growth will bring a negative impact on the stable operation of the power grid due to the high randomness and volatility of wind speed (Zhang et al., 2014). Therefore, it is necessary to accurately predict wind speed for scheduling the daily operation of wind power systems, which is crucial for effectively utilizing wind energy.

In recent years, wind speed forecasting has attracted much attention from academic researchers and industrial engineers (Lu et al., 2021). The number of publications related to Wind speed forecasting (WSF) is counted based on the search results of Web of Science (Clarivate, 2024) and displayed in Figure 3. According to the results on the topic of WSF, this quantity has rapidly increased since the late 2010s. Therefore, we roughly divide the research on WSF into three periods as follows.

In the past, the methods for wind speed prediction mainly consisted of physical methods, statistical methods, and traditional machine learning methods, among which traditional machine learning methods were used from their initial development until approximately 2018. Physical models usually make long-term predictions at mesoscale or large scale based on given boundary and initial conditions by establishing control equations describing the physical equations of atmospheric motion, of which numerical weather prediction (NWP) (Al-Yahyai et al., 2010) and weather research and forecasting (WRF) (Done et al., 2004) are most representative. In addition, these methods are affected by specific geographical conditions, surrounding environment, wind turbine operating conditions, uncertainty of model input, parameters, and structures, resulting in poor generalization ability. The statistical methods initially focused on autoregressive (AR) analysis (Jenkins, 1954), which was later improved to AR integrated moving

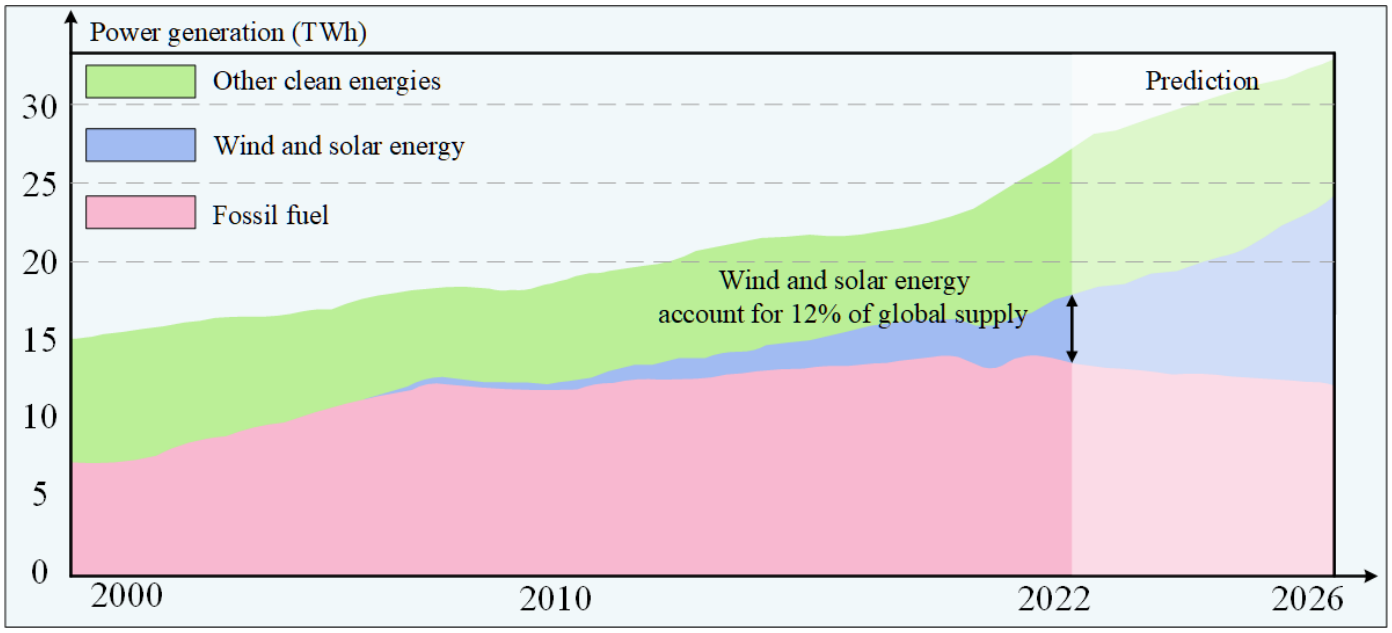


Figure 1 The Growth of Sustainable Energy Including Wind Energy, Solar Energy and Other Clean Energies

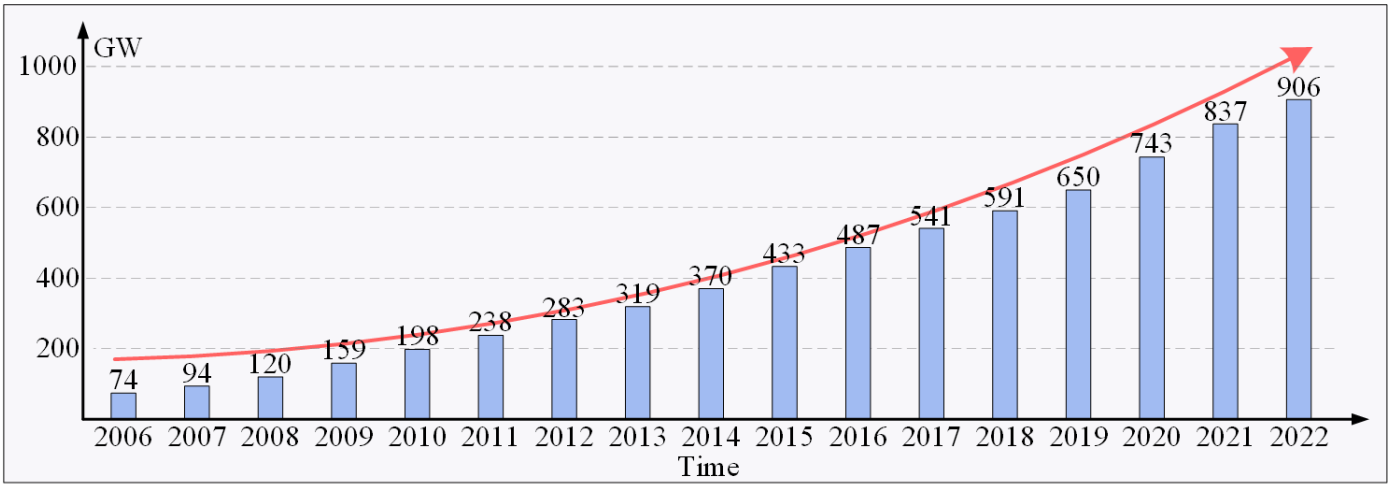


Figure 2 The Growth of Wind Energy from 2006 to 2022

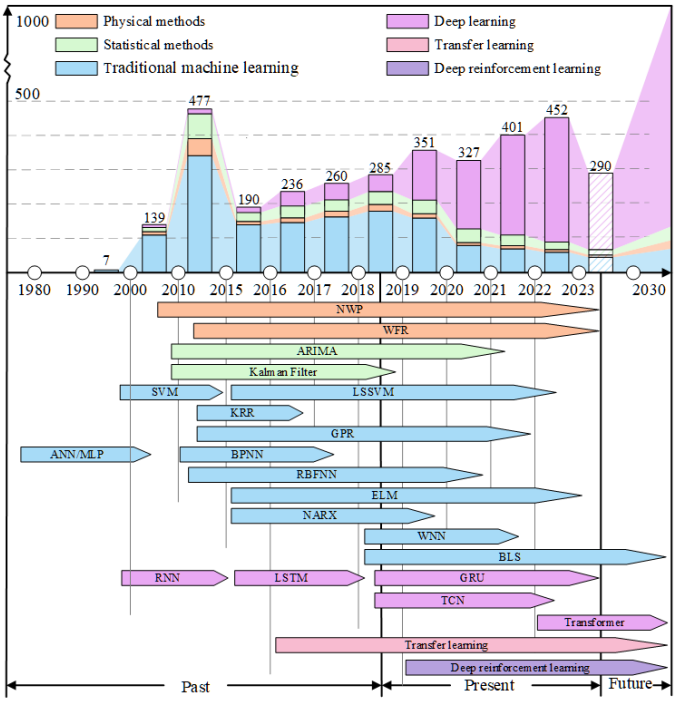


Figure 3 Development and Milestones of WSF

average (ARIMA) (Box & Pierce, 1970) and became more representative. In addition, there are the Kalman filter methods (Kalman, 1960). Different from physical models, these statistical methods are suitable for small-scale prediction and only use historical data (Soman et al., 2010). However, their prediction performance shows poor scalability and large uncertainty and usually fails to yield good performance. The earliest research on traditional machine learning can be traced back to the 1930s, and boosted to become an important development of artificial intelligence in the 1980s (Crafts, 2021). It was not until the early 21st century that these methods were widely applied in the WSF. These traditional machine learning methods are divided into four categories in this article. (a) Methods based on kernel techniques. These methods map the wind speed data into a high-dimensional feature space through nonlinear transformation, and then perform a linear regression in this feature space for WSF, including support vector regression (SVR) (Cortes & Vapnik, 1995), least squares SVR (LSSVR) (Suykens et al., 2002), kernel ridge regression (KRR) (Hoerl & Kennard, 1970), gaussian process regression (GPR) (Rasmussen & Nickisch, 2010). (b) Methods based on neural networks. These methods are based on neural networks for wind speed prediction, consisting of artificial neural networks (ANN) (Rosenblatt, 1958), back propagation neural network (BPNN) (Rumelhart et al., 1986), extreme learning machine (ELM) (Huang et al., 2006), wavelet neural network (WNN) (Zhang et al., 1995), nonlinear AR exogenous (NARX) (Williams et al., 1970), radial basis function neural networks (RBFNN) (Moody, 1988). (c) Methods based on the broad learning system (BLS). These methods are based on BLS (Chen & Liu, 2017) and have the ability to perform incremental learning for wind speed prediction. (d) Methods based on probabilistic graphical models. These methods have the ability to handle uncertainty problems utilizing Bayesian theory, including Bayesian-based methods (Bouckaert, 1995; Castelletti

& Consonni, 2020) and Markov-based methods (Eddy, 1996). They can automatically extract features from historical wind speed data and make predictions, alleviating the dominance of expert knowledge in feature engineering. However, their key parameters mainly depend on manual selection, resulting in limited scalability for large data sets.

In the present, the number of deep learning methods has started to grow explosively in the WSF field. Initially, deep learning was a new topic in the field of machine learning, first proposed by Hinton et al. in 2006 (Hinton et al., 2006). After a period of development, deep learning methods based on recurrent neural network (RNN) (Elman, 1990) became a hot spot and were widely used in the WSF field in the late 2010s. Furthermore, novel methods emerged in WSF, such as long short-term memory (LSTM) (Hochreiter & Schmidhuber, 1997), Gated recurrent unit (GRU) (Chung et al., 2014), temporal convolutional network (TCN) (Lea et al., 2016), echo state network (ESN) (Jaeger, 2002), Transformer (Vaswani et al., 2017), etc. These deep learning methods can extract nonlinear features from a large amount of historical wind speed data, attempting to provide end-to-end predictive models when dealing with growing data. It is expected that these models will directly link the original monitoring data with the corresponding prediction status, thereby constructing an automatic wind speed prediction system.

In the future, two paths lightweight (Wang et al., 2022) and ensemble methods (Dietterich, 2000) may be utilized to solve problems of WSF in engineering scenarios. Although deep learning has achieved significant success in WSF, these successes ignore engineering scenarios that are limited by devices. For edge devices, model simplification and operational efficiency are extremely important in industrial scenarios; hence, lightweight models are expected to receive more attention. These lightweight methods that have a more compact network structure and maintain the same performance as before, may be a way to solve the timeliness problem. In addition, there is a problem that the neural network must first train the model and then obtain the prediction results. Considering this problem, as an incremental learning method, BLS can get the prediction structure without additional training, which may also be a solution. For ensemble devices, with the continuous development of the big data era, the performance requirements will inevitably be demanded to improve in WSF. Currently, to achieve higher performance, more ensemble methods have been used to improve the adaptability and robustness of the model, which means more model parameters are required. However, a large number of model parameters will lead to overfitting of the prediction methods and poor prediction performance. Fortunately, deep reinforcement learning (François-Lavet et al., 2018), as a method of interaction between agents and the environment, is expected to achieve outstanding results in the integrated optimization process of multiple model parameters. Nevertheless, these ensemble methods based on deep reinforcement learning are currently confronted with a serious challenge that there are not sufficient label data to train the prediction model. To solve this issue, three solutions are proposed in this paper, including generative adversarial network (GAN) (Creswell et al., 2018), meteorological numerical simulation (Milliez & Carissimo, 2007), and transfer learning (Torrey & Shavlik, 2010).

To summarize the research on WSF, many review articles have been published, as shown in Table 1. These reviews are outdated and do not cover recently developed methods such as Transformer, TCN, and GRU. Specifically, these recent papers have not introduced new prediction methods for WSF in recent years, such as Transformer, TCN, GRU, etc. (b) They have not reported a roadmap for WSF to predict future trends yet. But for review papers, readers prefer to be interested in potential trends of WSF in the next five or ten years. Therefore, it needs a novel review to summarize the current research progress in WSF.

To overcome the aforementioned shortcomings, this paper systematically reviews the development of WSF and presents a roadmap for this field. The contributions of this review are refined as follows:

(1) The development of WSF is summarized into three periods. In the past, deep learning was not prevalent in the WSF field, and researchers attempted to use physical methods, statistical methods, and traditional machine learning methods to predict wind speed. In the present, deep learning focuses on further enhanced benefits in WSF. In the future, WSF will develop methods driven by lightweight real-time methods and ensemble methods based on deep reinforcement learning.

(2) A roadmap of WSF is pictured in this review. The roadmap includes potential research trends and provides valuable guidelines for researchers for future work.

Table 1 A Summary of Published Reviews

Authors	Date	Main research contents
Wu et al.	2007	A discussion based on NWP, statistical, and ANN methods.
Monteiro et al.	2009	A review on the definition, usage, and case studies of NWP.
Lei et al.	2009	A review based on types, including physical, statistical, spatial correlation, and ANN methods.
Soman et al.	2010	NWP, statistical approaches, ANN, and hybrid techniques.
Haque et al.	2012	A performance analysis using BPNN, RBFNN, and an adaptive neuro-fuzzy inference system (ANFIS).
Shi et al.	2012	An evaluation of hybrid approaches utilizing ARIMA, ANN, and Support vector machine (SVM).
Chandra et al.	2013	An introduction based on types, including persistence, physical, heuristic, learning, and hybrid methods.
Chang et al.	2014	A review based on method categories, including persistence, physical, statistical, spatial correlation, artificial intelligence, and hybrid methods.
Lawan et al.	2014	A review based on prediction horizons, including persistence, NWP, statistical, and ANN methods.
Samet et al.	2014	Nonlinear analyses and evaluation of some statistical prediction methods.
Goh et al.	2016	An evaluation review based on the Mycielski algorithm, K-means clustering, and Weibull distribution.
Bali et al.	2019	A review based on deep learning.
Bokde et al.	2019	A review on hybrid empirical mode decomposition models.
Valdivia-Bautista et al.	2023	A review on pre-processing techniques, prediction horizon, and regression metrics for wind speed forecasting.

(3) Some common metrics, definitions of prediction time horizon, and relevant publication titles in the WSF field are listed for reference by researchers in the appendix.

The rest of this review is organized as follows. In Section 2, we focus on the development of WSF in the past, including applications of physical methods, statistical methods, and shallow neural networks based on some algorithms. Section 3 reviews the applications of deep learning theories, which are considered the present period in the development of WSF. Section 4 discusses applications of lightweight methods and ensemble methods to WSF, including the motivation, the definitions, and some exploratory works. In Section 5, we further display a roadmap, when combined with the challenges of WSF. Conclusions are enclosed in Section 6. The appendix is placed in Section 7.

2. PAST: WSF USING PHYSICAL METHODS, STATISTICAL METHODS, AND TRADITIONAL MACHINE LEARNING METHODS

This section summarizes the applications of physical methods, statistical methods, and traditional machine learning methods to WSF, respectively. First, we introduce their theory and prediction procedure for WSF. Second, the achievements of these methods are reviewed.

2.1 Overview

In ancient times, WSF was often related to mystical activities such as divination and seeking rain, belonging to metaphysics. For wind speed forecasting, observing clouds, distinguishing winds, and identifying celestial phenomena are less useful due to the randomness and volatility of wind speed. With the help of mathematical methods and physical theory, the accuracy of wind speed prediction is expected to be enhanced. In the past period of WSF, physical methods, such as NWP and WRF, were applied to forecast wind speed. Then, some statistical methods based on mathematical methods, such as ARIMA and the Kalman filter, were utilized to predict wind speed. Furthermore, some traditional machine learning methods were used in WSF, including SVR, LSSVR, KRR, GPR, ANN, BPNN, ELM, WNN, NARX, RBFNN, and BLS, etc.

2.2 Physical Methods

2.2.1 A Brief Overview of Physical Methods

Physical models usually make long-term wind speed predictions at mesoscale or large scale based on given boundary and initial conditions by establishing control equations describing the physical laws of atmospheric motion (Al-Deen et al., 2006), mainly including NWP (Al-Yahyai et al., 2010) and WRF (Done et al., 2004). NWP utilizes mathematical equations and initial observation data to describe the physical processes of the atmosphere for forecasting wind speed, as shown in Figure 4, where these equations are based on physical laws such as mass conservation, momentum conservation, and thermodynamic principles, etc.

Specifically, first, observational data (including meteorological stations, radars, satellites, etc.) are imported into a numerical model of NWP. Second, this numerical model divides the Earth into a three-dimensional grid and calculates the physical processes of the atmosphere based on primitive equations of atmospheric motion at each grid point, including atmospheric motion, radiation transfer, turbulent mixing, etc. Finally, this numerical model utilizes the evolution of time and space to predict weather conditions in the coming hours, days, and even weeks, as well as to predict the development and evolution trends of large-scale weather systems. In addition, WRF is a mesoscale NWP model originally developed in collaboration with a series of US research institutions such as the Center for Environmental Prediction and the National Center for Atmospheric Research (Lo et al., 2008). Subsequently, it was used for meteorological prediction and numerical meteorological simulation.

2.2.2 Applications of Physical Methods

The publications about applications of physical methods to WSF are listed in Table 2.

Initially, physical methods were used more often than improved methods and were commonly used for regional weather forecasting. For example, Speer et al. utilized NWP to model the wind field in Australia and study its weather (Speer, 1997). Feng et al. used WRF to predict weather for North America (Feng et al., 1884). Subsequently, to further improve the accuracy of forecasting, some improved physical methods have been proposed, which are mainly divided into three categories. The first type is to improve the algorithm of physical methods, which is to reduce the minimum scale or provide more parameter information. For instance, Allen et al. proposed a boundary layer scaling technique for estimating near-surface wind energy (Allen et al., 2017). Cheng et al. assimilated anemometer wind speed observations from wind farm turbines into an NWP system, which enhances the accuracy of initial conditions (Cheng et al., 2017). Dupre et al. developed two downscaling approaches to improve the performance of NWP (Dupré et al., 2020). Optis et al. added 24 different settings, such as atmospheric pressure and sea surface temperature, to the WRF to enhance the accuracy (Optis et al., 2021). The second type utilizes other algorithms to optimize physical models. For example, computational fluid dynamics is introduced into NWP to improve performance (Durán et al., 2019; Zajaczkowski et al., 2011). Cassola et al. utilized Kalman filtering to reduce the deviation, which makes the prediction results of NWP more satisfactory (Cassola & Burlando, 2012). Zhao et al. proposed a CS-FS-WRF-E model based on WRF, fuzzy system, and cuckoo search (CS) algorithm (Zhao et al., 2016). Wang et al. presented a sequence transfer correction algorithm for NWP wind speed (Wang et al., 2019). The third type is to combine physical models with other models to form a hybrid model. Wang et al. combined NWP and SVM models to establish a correction model for WSF (Wang et al., 2017). Yang et al. proposed a dual physical model utilizing NWP and WRF (Yang et al., 2018).

Based on the above analysis, NWP and WRF are beneficial for a longer prediction horizon at mesoscale or large-scale, but they also exhibit some weaknesses. (1) It is difficult to capture prediction results at small scales. (2) It is not easy to make accurate short-term predictions. (3) High computational complexity leads to low time efficiency.

2.3 Statistical Methods

Statistical methods aim at establishing links between the present and the past to forecast wind speed by utilizing a constructed model and the estimation of its parameters according to historical data (Lei et al., 2009). Compared with physical methods, statistical models are easy to model and cheaper to develop. In particular, ARIMA and the Kalman filter are currently more popular in statistical methods.

2.3.1 A Brief Overview of ARIMA

The ARIMA method has five standard models, including the AR (Jenkins, 1954), moving average (MA) (Hunter, 1986), auto-regressive MA (ARMA) (Hannan & Kavalieris, 1984), and ARIMA (Jenkins, 1954). It is based on the principle of transforming the data into stationary data by differencing, and then forecasting by regressing the dependent variable only on its lagged series and the present and lagged series of the random error term. Firstly, some groundwork related to ARIMA is elaborated below.

(1) AR. AR was first proposed by British statistician Yule in 1927 (Jenkins, 1954). Let denote the input wind speed series. AR is usually utilized to predict a stationary series and is abbreviated $AR(p)$, as shown in Equation (1).

$$x_t = a_1 x_{t-1} + a_2 x_{t-2} + \ldots + a_p x_{t-p} + \varepsilon_t \quad (1)$$

where x_t indicates the wind speed value at moment t, P denotes the order of AR, $a = (a_0, a_1, \ldots a_p)^T$ represents the AR coefficient, and ε_t is random interference in the case x_t. This model reflects the existence of a linear relationship between the target value at moment t and the first p target values before.

(2) MA. It was first developed by Walker in 1931 (Walker, 1962), and refers to MA(q) and can be calculated by Equation (2) utilizing stationary historical data.

$$x_t = \mu + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} \ldots - \theta_q \varepsilon_{t-q} \quad (2)$$

where $\theta = (\theta_1, \theta_2 \ldots \theta_q)^T$ indicates the error term, and $\varepsilon = (\varepsilon_{t-1}, \varepsilon_{t-2} \ldots \varepsilon_{t-q})^T$ denotes random interference in the case θ, μ represents the mean value of the wind speed series X.

(3) ARMA. It was also proposed by Walker (Walker, 1962), consisting of AR and MA. ARMA can only forecast stationary series, abbreviated as ARMA(p, q), as calculated in Equation (3).

$$x_t = a_1 x_{t-1} + a_2 x_{t-2} + \ldots + a_p x_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} \ldots - \theta_q \varepsilon_{t-q} \quad (3)$$

Specifically, the flow chart is displayed in Figure 5. Firstly, it is necessary to determine whether the input historical wind speed series is a stationary series. If it is stationary, proceed directly to the next step. If it is unstable, perform differential data processing. Second, the autocorrelation coefficient and partial autocorrelation coefficient are calculated to determine the order and recognition model of ARMA. Third, the parameters of the model are estimated. Fourth, the adaptability of the model is tested. If the test passes, the prediction phase begins. If the inspection fails, repeat

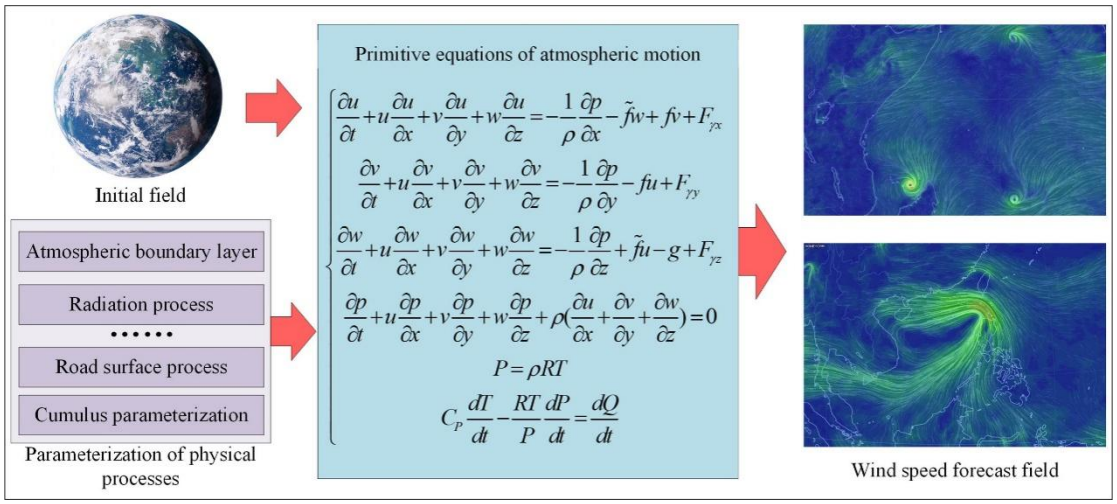


Figure 4 The Structure of NWP

Table 2 Summary of Applications of Physical Methods to WSF

References	Methodologies	Forecast time scale
Speer (1997); Allen et al. (2017); Cheng et al. (2017); Dupré et al. (2020); Zajaczkowski et al. (2011); Durán et al. (2019); Cassola & Burlando (2012); Wang et al. (2019); Wang et al. (2017); Yang et al. (2018)	NWP	Medium to long-term (hours to days)
Feng et al. (1884), Optis et al. (2021); Zhao et al. (2016)	WRF	Medium-term (hours to days)

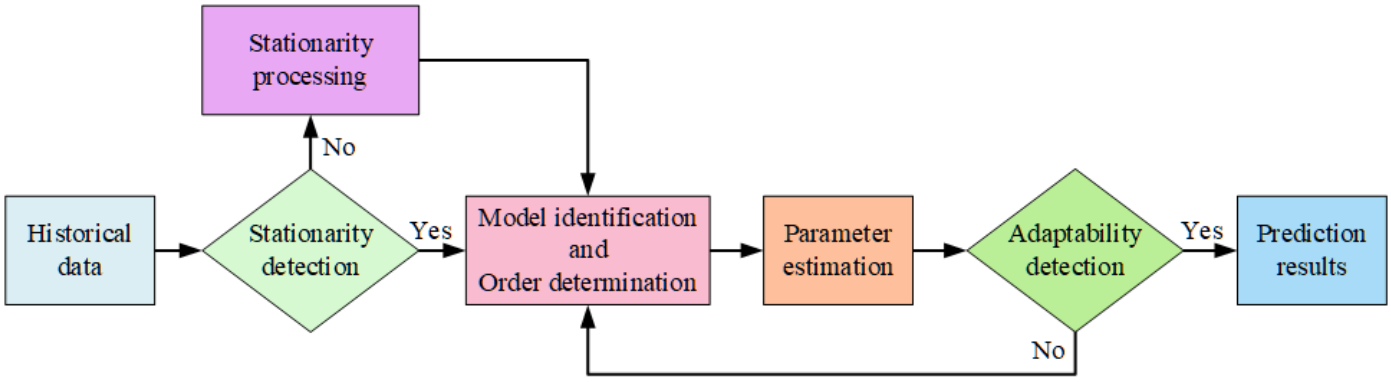


Figure 5 ARMA Flow Chart

the second and third steps. Finally, a highly adaptable model is utilized for prediction and output prediction results.

(4) ARIMA. It is a famous time series prediction method proposed by Box and Jenkins in the early 1970s (Jenkins, 1954); hence, it is also known as the Box-Jenkins model or the Box-Jenkins method. According to Equation (3), the differential operation is introduced to eliminate the fluctuations of the wind speed series based on ARMA, as in Equation (4).

$$\Delta^d x_t = a_1\Delta^d x_{t-1} + a_2\Delta^d x_{t-2} + \ldots + a_p\Delta^d x_{t-p} + \varepsilon_t - \theta_1\varepsilon_{t-1} - \theta_2\varepsilon_{t-2} \ldots - \theta_q\varepsilon_{t-q} \quad (4)$$

where the operator Δ corresponds to a discrete derivative and is defined as in Equation (5).

$$\Delta x_t = x_t - x_{t-1} \quad (5)$$

The modeling of wind speed by ARIMA is divided into a three-step iterative process as follows. (1) Step 1. The identification of model order to stabilize the input wind speed series: Given the differencing order d to stabilize the series by differencing the signal time series. Then, the autocorrelation and partial autocorrelation of the differenced series are investigated. Finally, the values of p and q are estimated by analyzing the autocorrelation function (ACF) and partial autocorrelation function (PACF) (Hannan & Kavalieris, 1984). (2) Step 2. The Estimation of the model: The maximum likelihood method of model estimation was used to estimate the model. (3) Step 3. The diagnostic check: The residuals or prediction errors will be checked to assess the estimation performance and fitting performance of the model. For an outstanding fitting model, the prediction error corresponding to white noise will be small. Also, for an excellent estimation model, the ACF of the prediction error will be very low.

2.3.2 A Brief Overview of Kalman Filter

Kalman filter was first presented by Kalman (Kalman, 1960) which is an algorithm for making state inference in a state-space model of a linear dynamical system. The linear dynamical system can be calculated by Eqs. (4, 5).

$$x_t = A * x_{t-1} + B\mu_t + \omega_{t-1} \quad (6)$$

$$z_t = H * x_t + \upsilon_t \quad (7)$$

where x_t denotes the state vector, A represents the state transition matrix, B indicates the control-input matrix, ω_{t-1} is the process noise at time step $t-1$, z_t denotes the observation vector, H indicates the observation transition matrix, and υ_t represents the observation noise. ω_{t-1} and υ_t are independent Gaussian white noises, which can be described by Eqs. (8, 9),

$$p(\omega) \sim N(0, Q) \quad (8)$$

$$p(v) \sim N(0, R) \tag{9}$$

where their covariance denotes Q and R, respectively. Furthermore, the computational and probabilistic origins of the Kalman filter, parameters, and tuning have been introduced in (Maybeck, 1990; Simon, 2001); therefore, its details are no longer described in this paper.

2.3.3 Other Statistical Methods

In addition to ARIMA and Kalman filter, some other statistical methods have been utilized in WSF, such as Mycielski approach (Hocaoğlu et al., 2009), stochastic differential equation (Arenas-López & Badaoui, 2020; Iversen et al., 2016; Møller et al., 2016; Zárate-Miñano et al., 2013), Weibull distribution (Kaplan & Temiz, 2017; Ozay & Celiktas, 2016; Wang et al., 2011), Markov model (Fidan et al., 2012; Li et al., 2021; Zhang et al., 2019), Bayesian methodology (Miranda & Dunn, 2006), and grey prediction (El-Fouly et al., 2006; Riahy & Abedi, 2008).

2.3.4 Applications of Statistical Methods

Currently, many statistical methods have been applied in WSF, as shown in Table 2.

In the development of ARIMA methods, IAR was first used to forecast wind speed (Schlink & Tetzlaff, 1998), but the prediction results are unsatisfying. To solve the problem of randomness, the MA algorithm was introduced into AR to obtain ARMA methods for wind speed prediction. For instance, Torres et al. utilized ARMA to forecast hourly average wind speed (Torres et al., 2005). Subsequently, some improved ARMA methods were proposed to enhance the prediction performance. Erdem et al. proposed four approaches based on ARMA and other algorithms for forecasting wind speed (Erdem & Shi, 2011). Li et al. used ARMA combined with wavelet multi-resolution analysis to predict wind speed (Ling-ling et al., 2011). Liu et al. developed an ARMA-generalized AR conditional heteroscedasticity approach for modeling the mean and volatility of wind speed (Liu et al., 2011). Lydia et al. proposed linear and non-linear ARMA models with and without external variables (Lydia et al., 2016). Han et al. developed non-parametric hybrid models based on ARMA for probabilistic wind speed forecasting (Han et al., 2017). Zhang et al. presented a VMD-PRBF-ARMA-E model combining two algorithms, including variational mode decomposition (VMD), principal component analysis (PCA), and radial basis functions (RBF) (Zhang et al., 2020). Furthermore, to overcome the non-stationarity of wind speed series, differencing was added to the ARMA methods, developing a popular ARIMA method. Among them, some variant ARIMA methods were presented, including fractional-ARIMA (f-ARIMA) (Kavasseri & Seetharaman, 2009; Kiplangat et al., 2016), Taylor-Kriging ARIMA (Liu et al., 2010), EMD-recursive ARIMA (Liu et al., 2015), ARIMA with exogenous variables (ARIMAX) (Zhao et al., 2015), nested ARIMA (Sim et al., 2018), wavelet transform (WT)-ARIMA (Singh & Mohapatra, 2019), and seasonal ARIMA (Liu et al., 2021). In addition, considering low data quality caused by the severe scenarios of wind turbines, Kalman filtering was utilized to improve the robustness of the prediction model. For instance, Xiu et al. forecasted wind speed using a chaotic operator network based on the Kalman Filter (Xiu & Guo, 2013). Zuluaga et al. proposed a short-term wind speed prediction model based on robust Kalman Filtering (Zuluaga et al., 2015). Besides, some other statistical methods were employed in WSF. For example, Hocaoglu et al. utilized the Mycielski approach to forecast wind speed (Hocaoğlu et al., 2009). Iversen et al. (2016), Zárate-Miñano et al. (2013), Møller et al. (2016), and Arenas-López and Badaoui (2020) utilized stochastic differential equations for wind speed forecasting. Kaplan and Temiz (2017), Wang et al. (2011), and Ozay and Celiktas (2016) proposed some prediction methods based on the Weibull distribution for wind speed prediction. Li et al. developed a Markov model for short term wind speed prediction by integrating the wind acceleration information (Li et al., 2021). Miranda et al. achieved one-hour-ahead wind speed prediction using a Bayesian methodology (Miranda & Dunn, 2006). Riahy and Abedi (2008) and Huang et al. utilized the grey forecasting method for WSF (Huang et al., 2011).

Based on the above literature, these statistical methods are suitable for small-scale prediction and only use historical data to provide timely predictions. However, it is not based on any predefined mathematical models and require different modeling methods for different historical data, which shows poor scalability and large uncertainty. In addition, it is difficult to accurately establish models and extract wind speed features when inputting complex historical wind speed data, which leads to unsatisfactory prediction results.

2.4 Traditional Machine Learning Methods

Traditional machine learning methods are usually based on the application of mathematical methods such as statistics, linear algebra, and optimization algorithms to learn and construct predictive models from historical data for wind speed forecasting (Sarker, 2021). These methods are mainly divided into three categories based on their theoretical basis, including methods based on kernel techniques (Schaback & Wendland, 2006), methods based on neural networks, and methods based on probabilistic graphical models (Koller & Friedman, 2009). The prediction procedure of traditional machine learning methods includes three steps: data collection, model training, and wind speed prediction, as shown in Figure 6.

In the step of data collection, the data, including wind speed, wind direction, atmospheric pressure, and air humidity etc., are usually collected from the sensors of wind turbines. Model training includes two steps. First, due to the complex application scenarios of wind turbines, the quality of these data is low, and improvements need to be made through the utilization of preprocessing approaches, such as EMD (Zeller et al., 2010), PCA (Wold et al., 1987), seasonal-trend decomposition (STL) (Cleveland et al., 1990), and dynamic time warping (DTW) (Müller, 2007), etc. Second, some methods, such as ANN, SVR, BPNN, etc., are utilized to extract features from these improved data to obtain prediction information. In the step of wind speed prediction, some loss functions, such as mean absolute error (MAE) (Error, 2016), mean square error (MSE) (Chai & Draxler, 2014), Huber loss (Meyer, 2021), mean absolute percentage error (MAPE) (De Myttenaere et al., 2016), etc., are usually employed to optimize the model to obtain accurate prediction results.

2.4.1 Methods Based on Kernel Techniques

(1) A brief overview of SVR

SVR is an application of SVM to regression prediction problems and was first proposed by Cortes et al. (Cortes & Vapnik, 1995). Specifically, it has the ability to make regression predictions based on statistical learning theory and the structural risk minimization principle. Its basic principle is to map the data into a high-dimensional feature space via nonlinear mapping, and to perform a linear regression in this feature space. The regression formula of SVR can be expressed as Equation (10),

$$f(x) = \sum_{i=1}^{D} W_i^S \varphi_i(x) + b \tag{10}$$

where $\{\varphi_i\}(x)_{i=1}^{D}$ denotes features, b indicates the bias term, and $\{W_i^S\}_{i=1}^{D}$ represents weight vectors estimated from the data. The coefficients $\{W_i^S\}_{i=1}^{D}$ can be obtained from the data as shown in Equation (11).

$$\begin{aligned} &\min_{(w,b,\xi)} \frac{1}{2}\|\{W^S\|^2 + C_S \sum_{i=1}^{n}(\xi_i + \xi_i^*) \\ &s.t.|yi - \langle W^S \cdot \Phi(x)\rangle - b| \le \varepsilon + \xi_i \\ &\xi_i \ge 0, \xi_i^* \ge 0, i = 1, 2, \dots, n \end{aligned} \tag{11}$$

where ξ_i denotes a slack variable and C_S indicates a constant. Then, according to Equation (11), the function can be derived, as follows Equation (12),

$$f(x, \alpha, \alpha^*) = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)K^S(x_i, x) + b \tag{12}$$

where the α_i and α_i^* are the Lagrange multipliers, and the kernel function $K^S(x_i, x)$ represents the inner product in the D-dimensional feature space.

(2) A brief overview of LSSVR

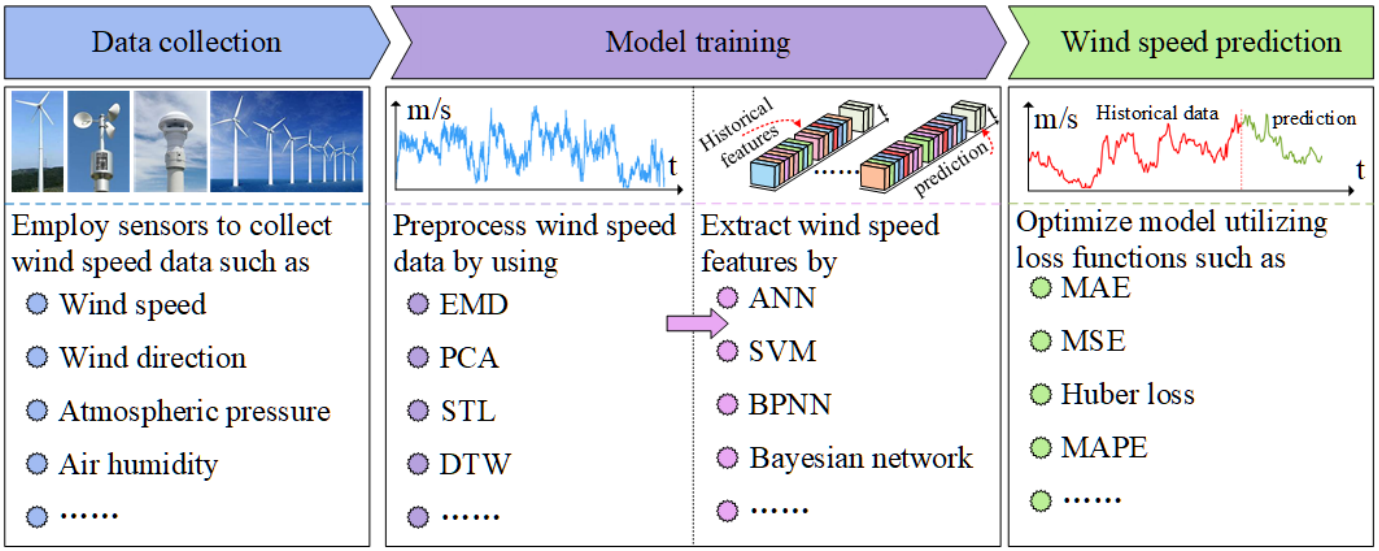


Figure 6 The Prediction Procedure of Traditional Machine Learning Methods

LSSVR is an improved SVR proposed by Suykens et al. (Suykens et al., 2002), which is based on the least-squares method (Björck, 1990). The regression formula of LSSVR is constructed as follows Equation (13),

$$f(x) = W^{LT}\Phi(x) + b \tag{13}$$

where W^L denotes the weight vector, b indicates the bias term, $\Phi(x)$ represents the nonlinear mapping function that transfers the input to a higher-dimensional feature space. Specifically, the weight vector W^L can be calculated by Eqs. (14, 15),

$$C_L = \frac{1}{2}W^{LT}W^L + \frac{1}{2}\gamma\sum_{i=1}^{n} e_i^2 \tag{14}$$

$$f(x_i)(W^{LT}\Phi(x_i) + b) = 1 - e_i \tag{15}$$

where e_i denotes the error variable for time t, γ represents a regulation constant. Furthermore, based on Eqs. (14, 15), the Lagrange function can be constructed, as follows Equation (16),

$$L(W^L, b, e, \alpha) = \frac{1}{2}\|W^L\|^2 + \gamma\frac{1}{2}\sum_{i=1}^{n} e_i^2 - \sum_{i=1}^{n}\alpha_i(W^{LT}\Phi(x_i) + b + e_i - f(x_i)) \tag{16}$$

where α_i indicates Lagrange multipliers. If the function meets the Mercer condition (Lyu, 2005), a kernel function is defined, as displayed in Equation (17),

$$K^L(x_i, x_j) = \Phi(x_i)\Phi(x_j) \tag{17}$$

Finally, taking the derivative of Equation (16) and combining Equation (17), the nonlinear regression function can be derived in Equation (18),

$$f(x) = \sum_{i=1}^{n}\alpha_i K^L(x_i, x_j) + b \tag{18}$$

(3) A brief overview of KRR

KRR was developed by Hoerl et al. (Hoerl & Kennard, 1970), and is a nonlinear regression approach that utilizes the kernel trick to transform time series data into high-dimensional feature spaces for prediction. The regression model of KRR is indicated as $f(x_i) = W^{KT}\phi(x_i)$, where $\phi(x_i)$ denotes a nonlinear mapping, and W^K represents the weight vector and is given by $W^K = [W_1^K, W_2^K, \dots, W_n^K]^T$. The weight vector W^K can be obtained by using regularized least squares according to the following objective function Equation (19):

$$\mathrm{Min}L_{KRR} = \frac{1}{2}\|W^k\|^2 + \frac{1}{2}C_K\sum_{i=1}^{N}(\hat{f}(x_i) - W^{KT}\phi(x_i))^2 \tag{19}$$

where C_K denotes a regularization parameter. Then, by taking the derivative of L_{KRR} and the representer theorem[121], $f(x_i)$ can be obtained by Eqs. (20, 21),

$$f(x_i) = \begin{bmatrix}\phi(x_i)\phi(x_1)^T \\ \phi(x_i)\phi(x_2)^T \\ \vdots \\ \phi(x_i)\phi(x_N)^T\end{bmatrix}(K + \frac{I}{C_K})^{-1}\hat{f}(x) \tag{20}$$

$$K = \begin{bmatrix}\phi(x_1)\phi(x_1)^T & \phi(x_1)\phi(x_2)^T & \cdots & \phi(x_1)\phi(x_N)^T \\ \phi(x_2)\phi(x_1)^T & \phi(x_2)\phi(x_2)^T & \cdots & \phi(x_2)\phi(x_N)^T \\ \vdots & \vdots & \ddots & \vdots \\ \phi(x_N)\phi(x_1)^T & \phi(x_N)\phi(x_2)^T & \cdots & \phi(x_N)\phi(x_N)^T\end{bmatrix}_{N\times N} \tag{21}$$

where I is an identity matrix of $N \times N$ dimension, and K denotes the kernel matrix. Finally, a new test sample x_i is input into this model, and then the prediction output can be obtained utilizing the double weights and similarity between the test sample x_i and all the training samples used to forecast wind speed.

(4) A brief overview of GPR

GPR was first presented by Rasmussen et al. (Rasmussen & Nickisch, 2010). It is a regression technique based on non-parametric Bayesian modelling, which has strong capability to solve the problem of stochastic uncertainty and non-stationarity. GPR is a computationally expensive algorithm commonly used for low-dimensional and small-sample regression problems (Dou et al., 2016). Let us create a function that meets $f(x_i) = \varphi(x_i) + \varepsilon_i$, where ε indicates normally distributed additive noise, and $\varphi(x)$ is the gaussian distribution, and its mean is usually zero. Therefore, $\varphi(x)$ can be described by Equation (22),

$$\varphi(x) \sim N(0, \Sigma) \tag{22}$$

where Σ indicates a covariance function. Then, let kernel $K^G(x_i, x_j)$ equal to Σ_{ij} for computing the covariance function. Finally, the distribution of GPR is given by Equation (23),

$$f(x) \sim N(0, K^G(x_i, x_j)) + \varepsilon \tag{23}$$

(5) Applications of methods based on kernel techniques to WSF

The publications about applications of methods based on kernel techniques to WSF are listed in Table 3, which is divided into four categories, including SVR-based methods, LSSVR-based methods, KRR-based methods, and GPR-based methods.

Table 3 Summary of Applications of Statistical Methods to WSF

References	Methodologies	Forecast time scale
Schlink and Tetzlaff (1998)	AR	Short-term
Torres et al. (2005), Erdem and Shi (2011), Ling-ling et al. (2011), Liu et al. (2011), Lydia et al. (2016), Han et al. (2017), Zhang et al. (2020)	ARMA	Short-term
Kavasseri and Seetharaman (2009), Kiplangat et al. (2016), Liu et al. (2010), Liu et al. (2015), Zhao et al. (2015), Sim et al. (2018), Singh and Mohapatra (2019), Liu et al. (2021).	ARIMA	Short to medium-term
Xiu and Guo (2013), Zuluaga et al. (2015).	Kalman Filter	Very short to short-term
Hocaoğlu et al. (2009).	Mycielski approach	Short-term
Iversen et al. (2016), Zárate-Miñano et al. (2013), Møller et al. (2016), Arenas-López and Badaoui (2020).	Stochastic differential equation	Short to medium-term
Kaplan and Temiz (2017), Wang et al. (2011), Ozay and Celiktas (2016)	Weibull distribution	Long-term
Li et al. (2021), Fidan et al. (2012), Zhang et al. (2019).	Markov model	Very short-term to short-term
Miranda and Dunn (2006)	Bayesian methodology	Short-term
Riahy and Abedi (2008), Huang et al. (2011).	Grey prediction	Long-term

These methods usually combine optimization algorithms and preprocessing approaches to enhance performance in WSF. Initially, Mohandes et al. verified that SVM had better performance than ANN in WSF (Mohandes et al., 2004). Therefore, SVM has been extensively studied. In these methods based SVR, many proposed methods are based on various algorithms to optimize them, Such as ensemble empirical mode decomposition (EEMD)-SVM (Hu et al., 2013), unscented Kalman filter (UKF)-SVR (Chen & Yu, 2014), generalized PCA-SVR (Hu et al., 2014), SVM based on RBF (Petković et al., 2014), wavelet decomposition and artificial bee colony algorithm (ABC)-based relevance vector machine (Fei & He, 2015), particle swarm optimization (PSO)-reduced SVM (Kong et al., 2015), cuckoo optimization (CO)-SVR (Wang et al., 2015), genetic algorithms (GA)-SVR (Santamaría-Bonfil et al., 2016), EMD-SVR (Ren et al., 2014), multi-objective differential evolution (MODE)-SVM (Shrivastava et al., 2016), seasonal information extraction, wavelet decomposition algorithm and genetic algorithm (SIE-WDA-GA)-SVR (Wang & Li, 2018), singular spectrum analysis and locality-sensitive hashing (SSA-LSH)-SVR (Liu et al., 2018), stacked de-noising autoencoder (SDA)-UKF-SVR (Cai et al., 2019), Twin-SVR (Dhiman et al., 2019), backward bat algorithm (BBA)-SVM (Li et al., 2019), SSA-EMD-CNN-SVM (Mi et al., 2019), SSA-VMD-SVM (Natarajan & Subramaniam Nachimuthu, 2020), Jaya-SVM (Liu et al., 2020), EMD and local mean decomposition (LMD)-SVM (Tian & Chen, 2021), whale optimization algorithm (WOA)-SVR (Xian & Che, 2022), slime mould algorithm (SMA)-VMD-TSVR (Fu et al., 2023), etc. Similarly, Xiaolan and Hui (2009), Hu et al. (2015), Jiang and Huang (2017), Jiang et al. (2018), Jiang et al. (2019), Kang et al. (2017), Li and Jin (2018), Adnan et al. (2019), Xiang et al. (2019), and Wang et al. (2023) utilized some optimization algorithms to improve the predictive ability of Least squares support vector machine (LSSVM) in WSF. In addition, Douak et al. utilized KRR for wind speed prediction (Douak et al., 2013). Naik et al. presented a series of KRR-based methods in WSF, including a low-rank multi-KRR (Naik et al., 2018), EMD-KRR (Naik et al., 2018), and multi-KRR (Naik et al., 2019). Yang et al. proposed a hybrid wind speed prediction method combining EMD and Bayesian ridge regression (Yang & Yang, 2020). Recently, Carneiro et al. proposed a ridge regression ensemble of machine learning models for solar and wind forecasting (Carneiro et al., 2022). Different from the above methods, GPR not only utilizes the kernel trick but also a regression technique based on non-parametric Bayesian modelling, and is commonly improved in algorithms to apply to WSF. Yu et al. firstly utilized GPR for long-term wind speed prediction (Yu et al., 2013). Subsequently, some improved GPR methods were developed for wind speed prediction, such as empirical wavelet transform (EWT)-GPR (Hu & Wang, 2015), AR-GPR (Zhang et al., 2016), EWT-expectation propagation (EP)-GPR-t Observation model (Hu et al., 2017), GPR combined NWP (Hoolohan et al., 2018), Multi-Task GPR (Cai et al., 2020), light gradient boosting machine (LGB)-GPR (Liu et al., 2022), CEEMDAN-GPR (Zhang et al., 2022).

The above methods utilize the kernel trick to characterize the input data into high-dimensional nonlinear features, which effectively enhances the ability of the prediction models in WSF. However, the above methods have two drawbacks. First, similar to statistical models, they require a theoretical framework to derive their methods, but there are usually deep-seated relationships behind prediction tasks, and simple modeling can lead to inaccurate predictions. Second, due to kernel techniques mapping input data to high-dimensional features, this leads to low efficiency and overfitting of prediction models when there is a large amount of input data.

2.4.2 Methods Based on Neural Network

(1) A brief overview of ANN

Initially, a formal neuron model was proposed by Warren McCulloch and Walter Pitts in 1943, modeling biological neurons. Subsequently, the perceptron and related learning algorithms were invented by Frank Rosenblatt (Rosenblatt, 1958), which was the first practical application of ANN and demonstrated its powerful capabilities in the late 1950s. Specifically, ANN is a parallel and distributed information processing system, applied in modeling nonlinear problems for wind speed forecasting (Here, ANN is only the benchmark artificial neural network or multi-layer perceptron (MLP)). ANN can learn from given sample examples by constructing an input-output mapping to perform predictions of future samples, as shown in Figure 7. ANN consists of an input layer, hidden layer, and output layer. The relationship between the output and input is the following Equation (24).

$$f(x) = \varphi(\sum_{j=1}^{n} W_j^A x_j - \theta) \tag{24}$$

where $f(x)$ indicates the output, x_j, for $j = 1, 2, \dots, n$ denote the input, W_j^A represents the weight associated with the $j-$th input, θ is a threshold, and φ is the transfer function.

(2) A brief overview of BPNN

BPNN is a multi-layer feedforward neural network trained utilizing the error back-propagation algorithm, proposed by Rumelhart and McClelland in 1986 (Rumelhart et al., 1986). It consists of three layers: the input layer, the hidden layer, and the output layer, as shown in Figure 8. The hidden layer usually includes one or more hidden layers, allowing the network to construct complex functions. BPNN mainly utilizes forward information propagation and error backpropagation to achieve information transfer for WSF, as follows. (25, 26),

$$f(x_j) = \varphi_I(W_j + \sum_{m=t-n}^{t-1} W_{jm} f(x_m)) \tag{25}$$

$$f(x_t) = \varphi_o(W_o + \sum_{j=1}^{I} W_{oj} f(x_j)) \tag{26}$$

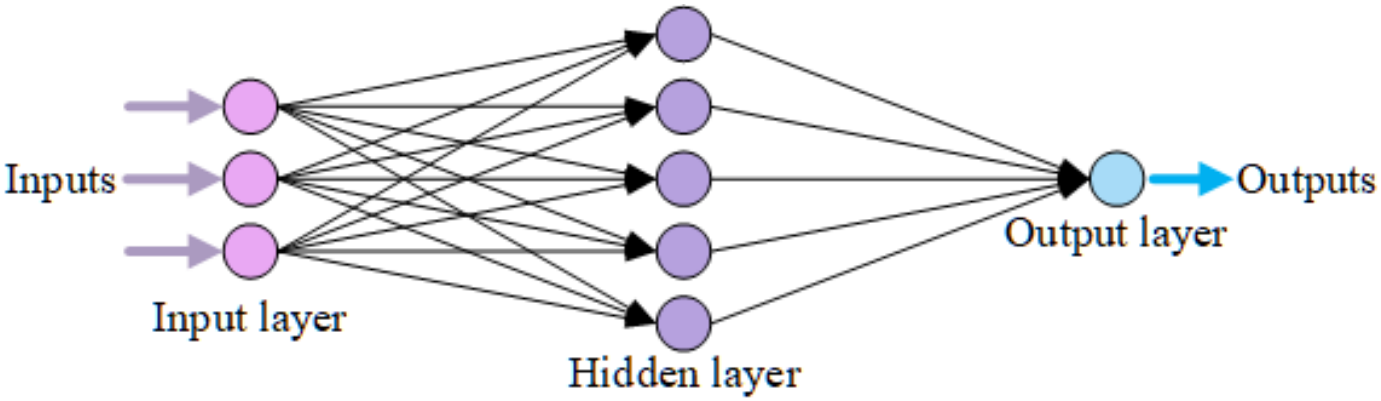


Figure 7 Structure of a Single-Layer ANN

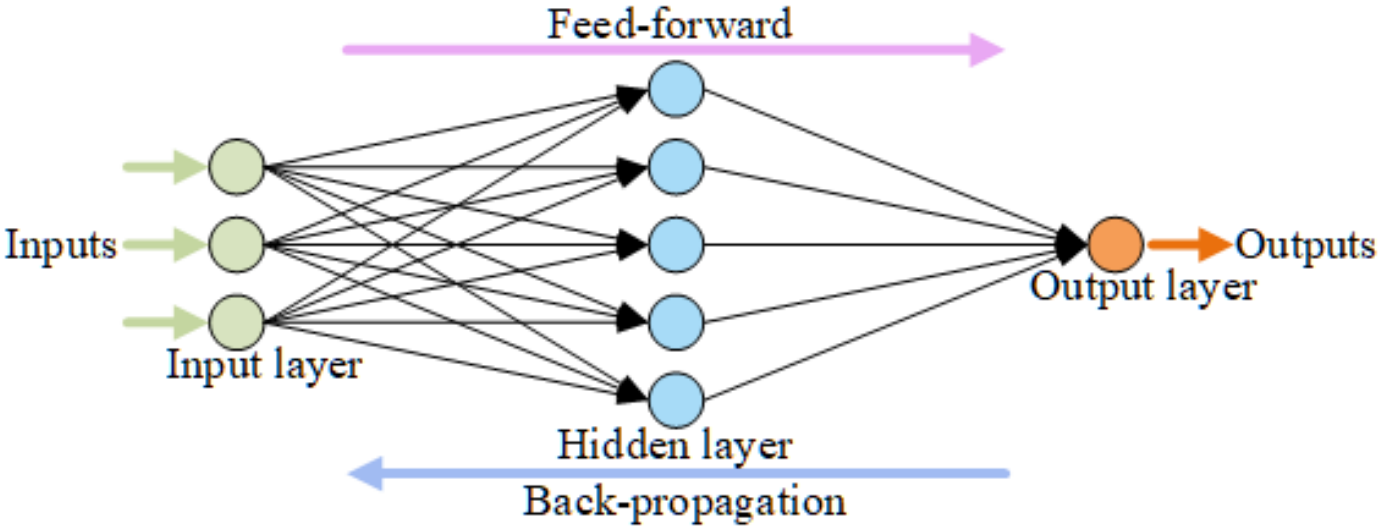


Figure 8 Structure of a Single-Layer BPNN

where $f(x_m)$ denotes the input of the input layer, $f(x_j)$ indicates the input of the hidden layer, $f(x_t)$ represents the prediction value in time step t, W_{jm} is the weight of the hidden layer, W_{oj} denotes the weight of the output layer, W_j represents the threshold value of the hidden layer, W_o indicates the threshold value of the output layer. φ_I and φ_o are the activation functions of the hidden and output layers, respectively. Usually, logical and hyperbolic functions are utilized as the hidden layer activation functionsφ_I, while a linear function is used as the output layer activation functionφ_o.

(3) A brief overview of ELM

In order to improve learning efficiency and simplify the setting of learning parameters, ELM was proposed by Huang et al. in 2004 (Huang et al., 2006). It is a feedforward neural network with outstanding generalization performance and extremely fast learning ability. The standard structure of ELM is depicted in Figure 9. Assuming a dataset with N samples $\{(x_i, f(x_i))\}_{i=1}^{N}$, where $x_i \in \mathbb{R}^N$ denotes the input vector, $f(x_i) \in \mathbb{R}^m$ represents the output vector, set $\varphi(x)$ as the activation function of the hidden layer with Q nodes, and the process of the standard ELM can be described as follows in Equation (27):

$$f(x_i) = \sum_{q=1}^{Q} W_q^O \varphi(W_q^I x_i + b_q) \tag{27}$$

where b_q denotes the bias of the $q-$th hidden node. W_q^I and W_q^O indicate the input and output weights, respectively. Simplify Equation (27) as a linear system in Equation (28):

$$HW^E = \mathbb{Y} \tag{28}$$

where H denotes the output of the hidden layers, as follows in Equation (29),

$$H = \begin{bmatrix} \varphi(W_1 \cdot X_1 + b_1) & \cdots & \varphi(W_Q \cdot X_1 + b_Q) \\ \vdots & \ddots & \vdots \\ \varphi(W_1 \cdot X_N + b_1) & \cdots & \varphi(W_Q \cdot X_N + b_Q) \end{bmatrix}_{N \times Q} \tag{29}$$

W^E represents the output weight matrix between the hidden layer and output layer described in Equation (30):

$$W^E = [W_1^{ET}, W_2^{ET}, \ldots, W_Q^{ET}]_{Q \times m}^T \tag{30}$$

$\mathbb{Y}$ indicates the target output results of the output layer, as follows in Equation (31):

$$\mathbb{Y} = [\mathbb{Y}_1^T, \mathbb{Y}_2^T, \ldots, \mathbb{Y}_N^T]_{N \times m}^T \tag{31}$$

Then, the optimal least-squares estimate of the output weight matrix $\tilde{W}^E$ can be obtained by minimizing the empirical risk, as in Equation (32):

$$\tilde{W}^E = (H^T H)^{-1} H^T \mathbb{Y} \tag{32}$$

Finally, the prediction results of ELM can be expressed as Equation (33):

$$\tilde{\mathbb{Y}} = H\tilde{W}^E \tag{33}$$

(4) A brief overview of WNN

WNN is a neural network utilizing wavelet basis functions as hidden layer nodes and excitation function error backpropagation, proposed by Zhang et al. firstly (Zhang et al., 1995). It has a very effective approximation ability and fault tolerance, as displayed in Figure 10. The formulas for the WNN are as follows Eqs. (34, 35):

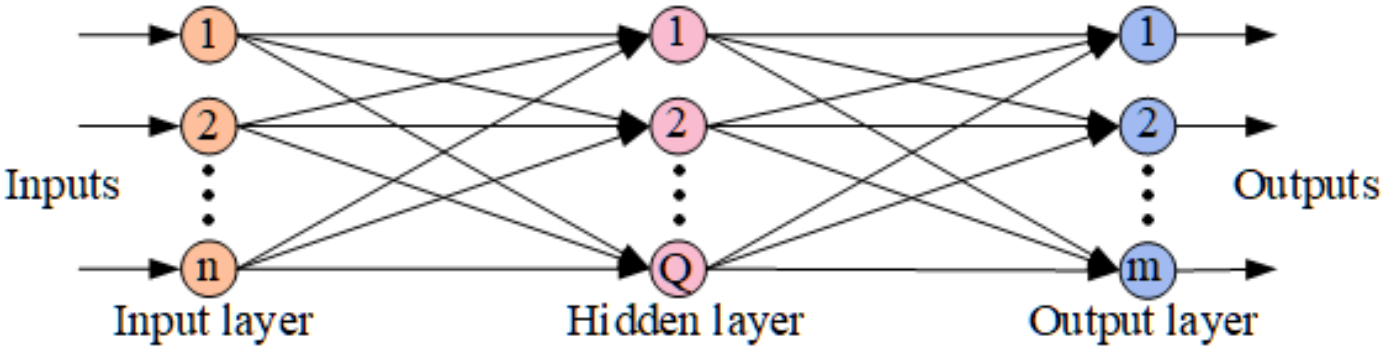


Figure 9 A Standard Structure of ELM

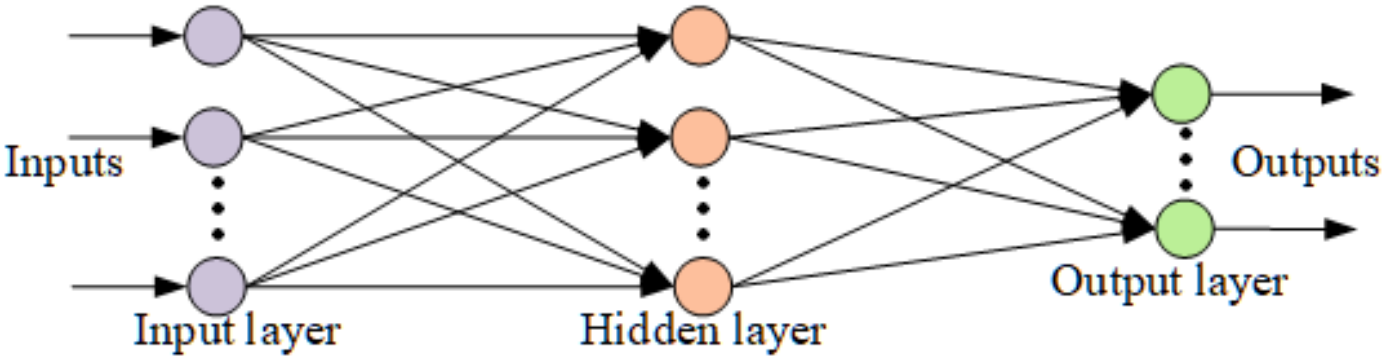


Figure 10 Structure of a Single-Layer WNN

$$h(j) = h_j\left(\frac{\sum_{i=1}^{k} W_{ij}^{W} x_i - b_j}{a_j}\right), j = 1, 2, \cdots, l \tag{34}$$

$$f(x_k) = \sum_{j}^{l} W_{jk}^{W} h(i), k = 1, 2, \cdots, m \tag{35}$$

where h denotes the hidden layer output, W_{ij}^{W} represents the weight of the input to the hidden layer, W_{jk}^{W} indicates the weight of the hidden to output layer, a_j is the wavelet basis function expansion factor, b_j denotes the wavelet basis function translation factor. l and m indicate the hidden and output layer nodes, respectively. The weight and parameter correction algorithm of WNN is similar to that of BPNN.

(5) A brief overview of NARX

NARX is a nonlinear AR neural network (NAR) with external inputs, first utilized by Williams et al. (Williams et al., 1970). It has strong nonlinear prediction ability to handle complex nonlinear relationships in non-stationary environments. Firstly, a benchmark model, NAR, is described in Equation (36),

$$f(t) = \varphi(x(t-1), \ldots, x(t-d)) \tag{36}$$

where x denotes a data series, $\varphi(\cdot)$ indicates the function and is not known prior to training. During training, this function can be approximately determined by optimizing the neural network weights and neuron biases. The structure of NAR is displayed in Figure 11.

The d features $(x(t-1), \ldots, x(t-d))$ are feedback delays, such as 1:28 (for d=28). In the hidden layer, W^N denotes the weight matrix and b indicates the bias vector.

Different from NAR, NARX allows the utilization of more information that corresponds to the data set to be predicted, such as ambient pressure and temperature, as shown in Figure 12. NARX is constructed to forecast regressive time series as in Equation (37),

$$f(t) = f(x(t-1), \ldots, x(t-d), \mathcal{X}(t-1), \ldots, \mathcal{X}(t-d)) \tag{37}$$

where $\mathcal{X}(t)$ represents another external input series and can be single- or multi-dimensional inputs.

(6) A brief overview of RBFNN

RBFNN is a three-layer feedforward network based on kernel technology, first presented by Moody et al. (Moody, 1988). The first layer is composed of signal source nodes in the input layer. The second layer is the hidden layer, and the transformation function of its hidden units is a non-negative nonlinear radial basis function. It's the hidden layer nodes represent a data cluster that is centered at a particular point with a given radius. The third layer is the output layer, which is a linear combination of outputs from hidden layer neurons. Its structure is displayed in Figure 13. The formula of the RBFNN is described as follows in Equation (38),

$$f(x) = \sum_{k=1}^{N} W_k^R \varphi_k(\|x - c(k)\|, \sigma_k) \tag{38}$$

where $x \in \mathbb{R}^{m\times 1}$ indicates an input wind speed series, N is the number of neurons, $W^R = [w_1, w_2, \ldots, w_N]^T \in \mathbb{R}^{N\times 1}$ represents the weight vector, and $\varphi_k(\cdot)$ is the basis function. $c(k) = [c_{k1}, c_{k2}, \ldots, c_{km}]^T \in \mathbb{R}^{m\times 1}$ denotes the center vector, σ_k indicates the bandwidth of the basis function, and $\|\cdot\|$ represents the Euclidean distance.

(7) Applications of NN-based methods to WSF

The publications about applications of NN-based methods to WSF are listed in Table 4, which are divided into six categories, including ANN, BPNN, ELM, WNN, NARX, RBFNN.

Initially, Flores et al. utilized ANN combined with a control algorithm for wind speed prediction (Flores et al., 2005). Subsequently, these ANN-based methods based on ANN are mainly improved in two structures. First, a series of algorithms are performed to improve the features extraction capability of ANN, such as, ANN- recursive prediction error algorithm (Barbounis & Theocharis, 2006), ANN-resilient propagation (RP) learning algorithm (Bilgili et al., 2007), EMD-feed-forward neural network (FNN) (Guo et al., 2012), EMD-ANN (Liu et al., 2012; Karakuş et al., 2017), self-organizing maps-MLP (Sheela & Deepa, 2013), GA-MLP (Liu et al., 2013), WD-ANN (Liu et al., 2013), k-NN-ANN (Yesilbudak et al., 2013), ANN-genetic expression programming (GEP) (Ghorbani et al., 2016), Fast EEMD-GA-MLP (Liu et al., 2015), WPD-FEEMD-Elman (Liu et al., 2015), ANN and ANFIS (Zhang et al., 2016), Improved WT (IWT)-ENN (Ruiz-Aguilar et al., 2021), etc. Second, some other models are used to further improve the accuracy of the prediction results, such as ARIMA-ANN (Cadenas & Rivera, 2007; Maatallah et al., 2015), ANN- MM5 model (Salcedo-Sanz et al., 2008; Salcedo-Sanz et al., 2009a, 2009b), ANN-MLR (Verma et al., 2023). Furthermore, the backpropagation (BP) algorithm was added to the ANN

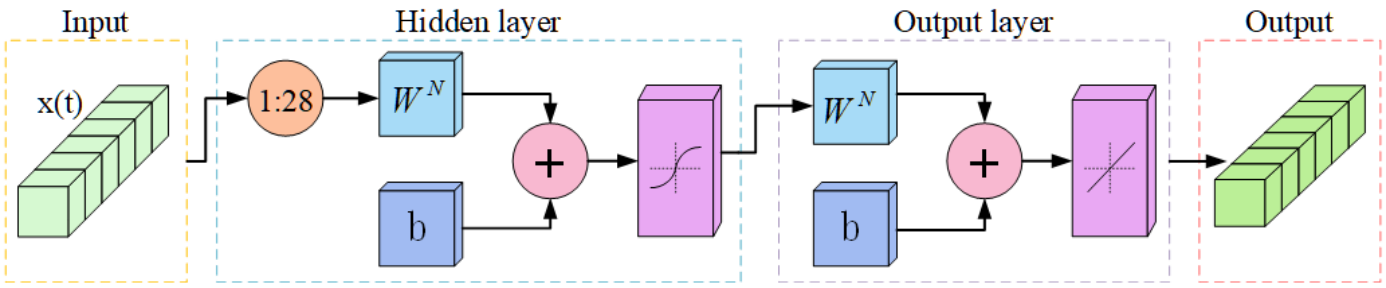


Figure 11 The Structure of NAR

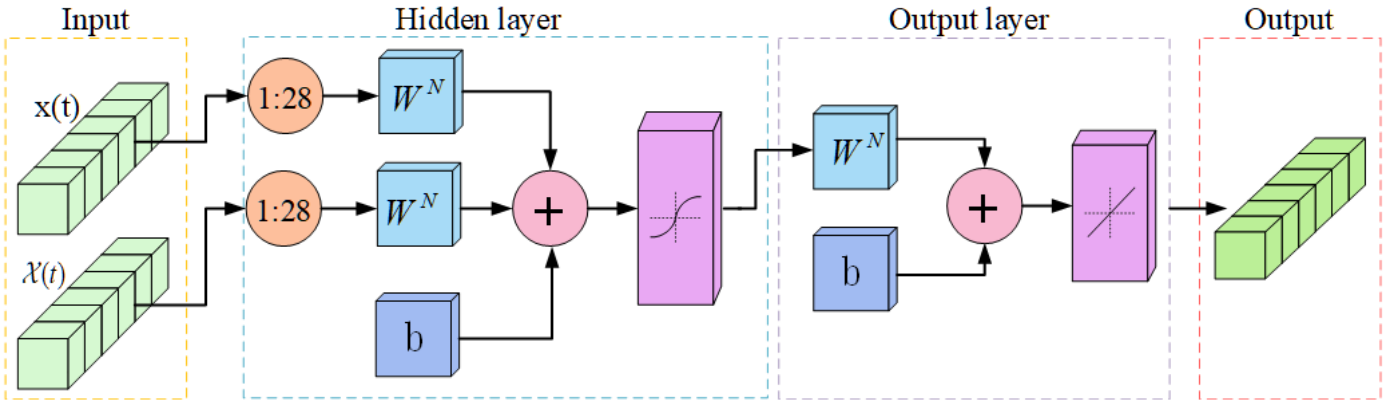


Figure 12 The Structure of NARX

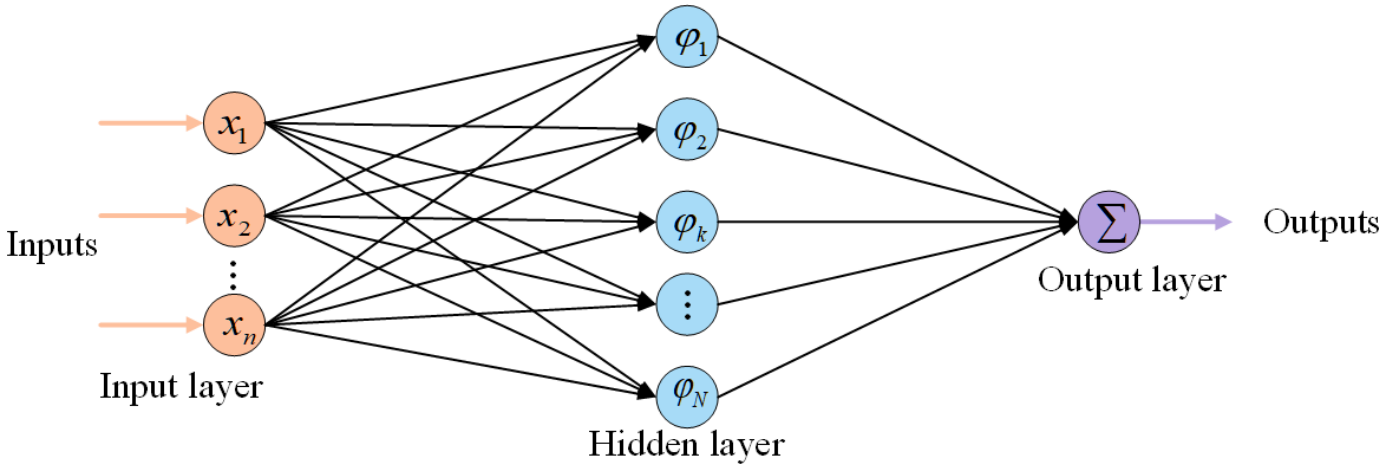


Figure 13 Structure of a Single-Layer RBFNN

to improve model accuracy in nonlinear processing, which allows BPNN to capture the public's attention. Many methods based on BPNN were proposed, such as, seasonal exponential adjustment (SEA)-BPNN (Guo et al., 2011), IS-PSO-BP (Ren et al., 2014), BPNN with crisscross optimization (Meng et al., 2016), BPNN with flower pollination based on bat search algorithm (Qu et al., 2017), CEEMD-BPNN (Yang & Wang, 2018; Qu et al., 2019), MLR-BP (Barhmi et al., 2020), EMD-SSA-ARIMA-BP (Ding & Meng, 2020). In addition, ELM was a research hotspot, as it only utilized simple structures to maintain prediction accuracy. Some decomposition approaches were performed to improve the predictive performance of ELM, such as, EMD (Li et al., 2019; Liu et al., 2018), EWT (Guo et al., 2023; Liu et al., 2018), EEMD (Tian et al., 2020; Wang et al., 2020), VMD (Chen et al., 2021; Fu et al., 2023; Hua et al., 2022; Nie et al., 2020; Wang et al., 2018), and complete ensemble empirical mode decomposition (CEEMD) (Fu et al., 2021; Wang et al., 2020), etc. Safavieh et al. proposed an integrated approach for very short-term wind speed forecasting (Azad et al., 2014), which is the earliest study on using WNNs for wind speed prediction. Then, some improved WNNs were extensively developed, such as WNN with multi-resolution analysis (Cadenas et al., 2016), WNN based on multi-objective interval prediction (Doucoure et al., 2016), and WNN utilizing EEMD (Safavieh et al., 2007). Furthermore, Azad et al. utilized NARX for long-term wind speed forecasting in WSF (Santhosh et al., 2018). Cadenas et al. combined univariate ARIMA and a multivariate NARX for wind speed forecasting (Shi et al., 2018). Jamil et al. compared and analyzed the NARX model with other ANN models in WSF (Blanchard & Samanta, 2020). Jawad et al. proposed a GA-NARX to solve the issue of uncertainty modelling for short-term and medium-term wind speed forecasting (Jamil & Zeeshan, 2019). Blanchard et al. used NAR and NARX for wind speed prediction (Jawad et al., 2018). Sfetsos et al. utilized RBFNN for the first time to predict wind speed in various comparison studies (Madhiarasan, 2020). Wang et al. presented a hybrid model based on the seasonal adjustment method (SAM), exponential smoothing method (ESM), and RBFN for hourly wind speed prediction (Sfetsos, 2000). Zhang et al. developed a hybrid model based on VMD-WT and PCA-BP-RBF neural network for short-term wind speed forecasting (Wang et al., 2014). Madhiarasan et al. proposed a recursive RRBFNN to forecast wind speed at different horizons (Zhang et al., 2019).

Due to the high self-learning capability, NN-based methods could automatically predict wind speed from input data by optimizing the NAR. Furthermore, they can easily forecast multi-step wind speed by utilizing the extracted features. However, there are two disadvantages. First, the complexity of the prediction models would greatly increase with the increase in input data. The increasing model parameters lower the training efficiency and further result in overfitting, which reduces the diagnostic accuracy of the prediction models. Second, the NN-based prediction models are black-boxed due to the lack of rigorous theoretical support. As a result, they are subject to low interpretability.

Table 4 Summary of Applications of Regression-Based Methods to WSF

References	Methodologies	Forecast time scale
Mohandes et al. (2004), Hu et al. (2013), Chen and Yu (2014), Hu et al. (2014), Petković et al. (2014), Fei and He (2015), Kong et al. (2015), Wang et al. (2015), Ren, Suganthan, and Srikanth (2014), Santamaría-Bonfil, Reyes-Ballesteros, and Gershenson (2016), Shrivastava, Lohia, and Panigrahi (2016), Wang and Li (2018), Liu et al. (2018), Cai et al. (2019), Dhiman, Deb, and Guerrero (2019), Li et al. (2019), Mi, Liu, and Li (2019), Natarajan and Subramaniam Nachimuthu (2020), Liu et al. (2020), Tian and Chen (2021), Xian and Che (2022), Fu et al. (2023)	SVR	Short-term
Xiaolan and Hui (2009), Guo et al. (2011), Hu, Wang, and Ma (2015), Jiang and Huang (2017), Jiang et al. (2018), and Jiang et al. (2019), Kang et al. (2017), Li and Jin (2018), Adnan et al. (2019), Xiang et al. (2019), Wang et al. (2023)	LSSVR	Short to medium-term
Douak et al. (2013), Naik, Bisoi, and Dash (2018); Naik, Satapathy, and Dash (2018); and Naik, Dash, and Dhar (2019), Yang and Yang (2020), Carneiro et al. (2022)	KRR	Short-term
Yu et al. (2013), Hu and Wang (2015), Zhang et al. (2016), and Hu, Wang, and Xiao (2017), Hoolohan, Tomlin, and Cockerill (2018), Cai et al. (2020), Liu, Wang, Qin, et al. (2022), and Zhang, Peng, and Nazir (2022)	GPR	Short to long-term

2.4.3 Methods Based on BLS

BLS is a representative incremental learning method, which refers to a learning system that can continuously learn new knowledge from new samples and can save most of the knowledge that has been previously gained, as shown in Figure 14. It was proposed for the first time by Chen et al. and became one of the most popular networks (Chen & Liu, 2017).

The above three categories of methods each exhibit inherent limitations that collectively motivated the transition to deep learning. Physical models (NWP, WRF) are computationally expensive, cannot resolve small-scale wind dynamics due to coarse grid resolution, and perform poorly on short-term forecasting horizons. Statistical methods (ARIMA, Kalman filter) are computationally efficient but require bespoke model selection for each dataset, cannot capture the non-linear and non-stationary characteristics of real wind speed series, and rely on manually designed lag structures that limit feature extraction capacity. Traditional machine learning methods (kernel-based methods, shallow neural networks, and BLS) introduced non-linear modeling and partial automation, yet still depend on manually selected kernels or network architectures, scale poorly to large datasets due to quadratic kernel matrix growth or overparameterization, and remain black-box models with low interpretability. Deep learning emerged as a direct response to these gaps: its hierarchical architectures automatically extract features from raw data without manual engineering, stochastic gradient-based training scales linearly with data, and end-to-end pipelines eliminate the need for task-specific preprocessing or model selection, thereby addressing the scalability, non-linearity, and automation limitations that earlier methods could not overcome.

The process of BLS is explained in three stages by us. In the first stage of BLS, the original wind speed data with linear mapping is input to the mapped feature layer, and the mapped feature node is imported to the enhancement node layer by nonlinear mapping. Assume the mapped features as $X \in \mathbb{R}^{l\times d}$, where l is the number of wind speed samples and d is the dimension of features. BLS has n groups of mapping feature nodes, with each group containing k nodes. The output of the ith group of the mapping feature nodes can be denoted as in Equation (39),

$$z_i = \varphi_i(XW_{ei}^B + \beta_{ei}), i = 1, 2, \dots, n \tag{39}$$

where W_{ei}^B denotes the weighting matrix, and β_{ei} indicates the bias term that is randomly initialized with the proper dimensions. Then, the mapped feature nodes and enhancement nodes are concatenated as hidden layers and connected to the output layer by weights. After that, BLS only requires performing the calculation of the pseudoinverse to rapidly obtain the weights from the hidden layer to the output layer, ensuring the efficiency of model training. In detail, these randomly generated mapping feature groups will be collected together, which can be represented as $Z_n = [z_1, z_2, \cdots, z_n]$. Furthermore, these randomly generated mapping feature groups are utilized to expand the enhancement layer through a nonlinear activation function, as depicted in Equation (40),

$$h_j = \xi(Z_n W_{hj}^B + \beta_{hj}), j = 1, 2, \dots, m \tag{40}$$

where $\xi(\cdot)$ denotes the nonlinear activation function. In the second stage of BLS, the weights between the hidden and output layers will be calculated. The output of the enhancement nodes can be represented as $H_m = [h_1, h_2, \dots h_m]$. The mapping feature nodes and enhancement nodes are concatenated into a matrix, and finally BLS can be denoted by Equation (41),

$$Y = [z_1, z_2, \dots, z_n | h_1, h_2, \dots, h_m] W^{Bm} = [Z^n | H^m] W^{Bm} \tag{41}$$

In the third stage of BLS, the weights will be updated. The pseudoinverse algorithm is utilized to compute the weights of the output layers as in Equation (42),

$$W^{Bm} = [Z^n | H^m]^+ Y[0, 1] \tag{42}$$

Moreover, an alternative approach is used to solve the pseudoinverse in Equation (43),

$$argmin(W^B) = \|AW - Y\|_v^{\sigma_1} + \lambda\|_u^{\sigma_2}\| \tag{43}$$

where $\sigma_1 > 0$ and $\sigma_2 > 0$, and v is norm regularization. A constraint term λ is added to the original least squares estimate $A = [Z^n | H^m]$ to make it possible to find the pseudoinverse when the original generalized inverse is ill-conditioned. Finally, the connection weights of the BLS can be approximated as in Equation (44),

$$W^{Bm} = (\lambda I + AA^T)^{-1} A^T Y \tag{44}$$

We specifically have Equation (45),

$$A+ = \lim_{\lambda \to 0} (\lambda I + AA^T)^{-1} A^T \tag{45}$$

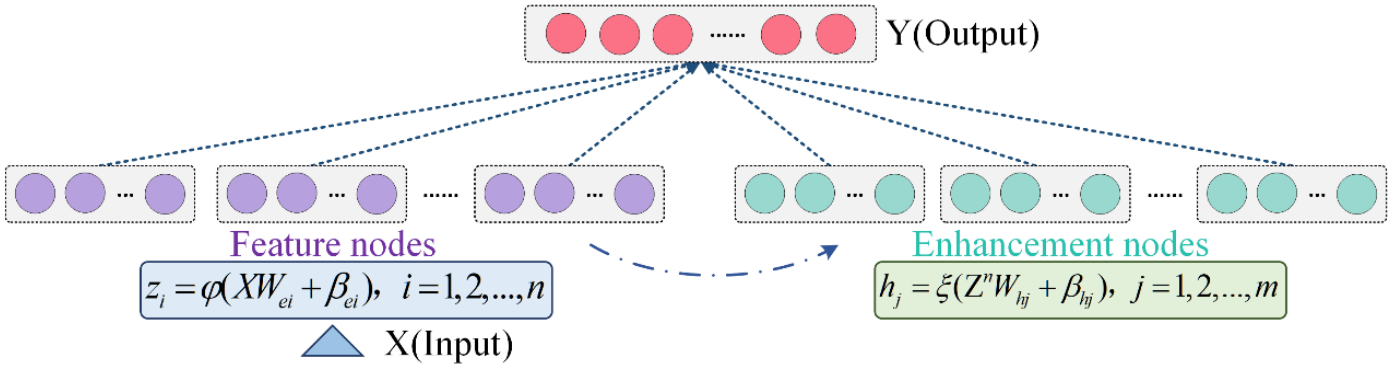


Figure 14 Structure of BLS

Currently, BLS has received great attention in WSF owing to the superiority of fewer parameters and incremental learning. Zhu et al. utilized BLS for the first time to predict wind speed and achieved satisfactory results Zhu & Lian, 2019; Zhu et al., 2020). Recently, Xu et al. have presented a wind speed prediction model based on phase space reconstruction and BLS (Jiao et al., 2023). Jiao et al. proposed an error correction-based VMD and BLS for wind speed prediction (Xu et al., 2023). In general, BLS has less operation time and computational complexity, but its prediction accuracy is less than that of deep learning.

2.4.4 Methods Based on Probabilistic Graphical Models

In addition to the approaches in the aforementioned sections, some methods are also concerned with WSF, including Bayesian network and Markov-based methods.

(1) Bayesian-based methods

Bayesian-based methods, also known as belief networks (Bouckaert, 1995) or directed acyclic graphical models (Castelletti & Consonni, 2020), are probabilistic graphical models. These methods have a strong ability to handle uncertainty problems. They express the correlation between input information elements using conditional probability and can learn and predict knowledge under limited, incomplete, and uncertain information conditions. For example, Miranda et al. utilized a Bayesian approach to characterize the wind resource for wind speed forecasting (Miranda & Dunn, 2006). Ye et al. proposed a Bayesian emulator with a Gaussian process prior for probabilistic forecasting of wind speed (Ye et al., 2021). Jiang et al. presented a method based on Bayesian theory and structural break modeling, which could incorporate domain knowledge about wind speed as a prior (Jiang et al., 2013). However, building Bayesian networks efficiently and accurately based on related knowledge has been a hot and challenging topic in the past decade.

(2) Markov-based methods

For Markov-based methods, the hidden Markov model (Eddy, 1996) is the simplest structured dynamic Bayesian network and can predict subsequent states by estimating the transition probability between discrete states in the observation system through past experience. For instance, Sahin et al. (Sahin & Sen, 2001), Shamshad et al. (Shamshad et al., 2005), and Song et al. (Song et al., 2014) were based on Markov chains to forecast wind speed. Kani et al. (Kani & Riahy, 2008; Kani & Ardehali, 2011) combined ANN and Markov chain for forecasting wind speed on a very short-term time scale. However, when constructing Markov methods, the assumed probability is usually constant and independent, which is perhaps difficult to meet the actual prediction needs.

3. PRESENT: WSF USING DEEP LEARNING METHODS

This section summarizes the applications of deep learning in WSF. First, the prediction procedures for WSF are introduced based on deep learning theories. Second, the achievements of deep learning-based approaches.

3.1 Overview

With the rapid development of internet technologies and the Internet of things (IoT), the volume of collected data has increased dramatically compared to ever before (Omolara et al., 2022). The increasingly large data bring more sufficient information to WSF so that it is more possible to provide accurate prediction results. Unfortunately, wind speed prediction based on traditional machine learning theories in the past is not appropriate for such big data scenarios. It is necessary to develop some advanced WSF methods.

Deep learning, derived from the research of neural networks, employs deep hierarchical architectures to represent the abstract features automatically, and further establish the relationship between the learned features and the target output directly (Aggarwal, 2018). The deep learning-based prediction procedure, as shown in Figure 15, consists of two steps, i.e., big data collection and deep learning-based prediction.

Big data has always been a popular term in modern industry and other application scenarios (Nazir et al., 2019). It includes four characteristics: quantity, speed, diversity, and accuracy. The characteristics are summarized as follows.

(1) Large volume. The volume of collected data sustainably grows during the long-term operation of machines, especially for large wind farm clusters.

(2) Low value density. The collected big data is usually of low quality, such as containing missing or outlier values, due to the complex operating conditions of machines.

(3) Multi-source and heterogeneous data. Multi-source data are collected by different kinds of sensors in machines, such as wind speed, atmospheric pressure, and cloud density.

(4) Monitoring data stream. The high-speed transmission channels are able to collect the data from machines immediately for real-time wind speed prediction.

Deep learning-based wind speed prediction methods automatically learn features from the input monitoring data and simultaneously forecast wind speed according to the learned features. These methods employ NAR frameworks, such as RNNs and TCNs, to learn features from the input big data. During the training process, the error between the actual output and the target is minimized by using the BP algorithm (Leung & Haykin, 1991) to update the training parameters of the prediction models. This subsection reviews typical deep learning methods and their applications in WSF, including RNN-based methods, TCN-based methods, and Transformer-based methods.

3.2 RNN-Based Methods

RNN-based methods are a type of RNN that takes sequence data as input, recurses in the direction of sequence evolution, and all loop units are connected in a chain, such as RNN (Elman, 1990), LSTM (Hochreiter & Schmidhuber, 1997), GRU (Chung et al., 2014), ESN (Jaeger, 2002). Then, the achievements of RNN-based methods are reviewed.

3.2.1 A Brief Overview of RNN

As the benchmark model for RNN-based methods, RNN has the ability to forecast wind speed series, first proposed by Elman et al. (Elman, 1990). In each loop unit of an RNN, there are two main steps to obtain the output,

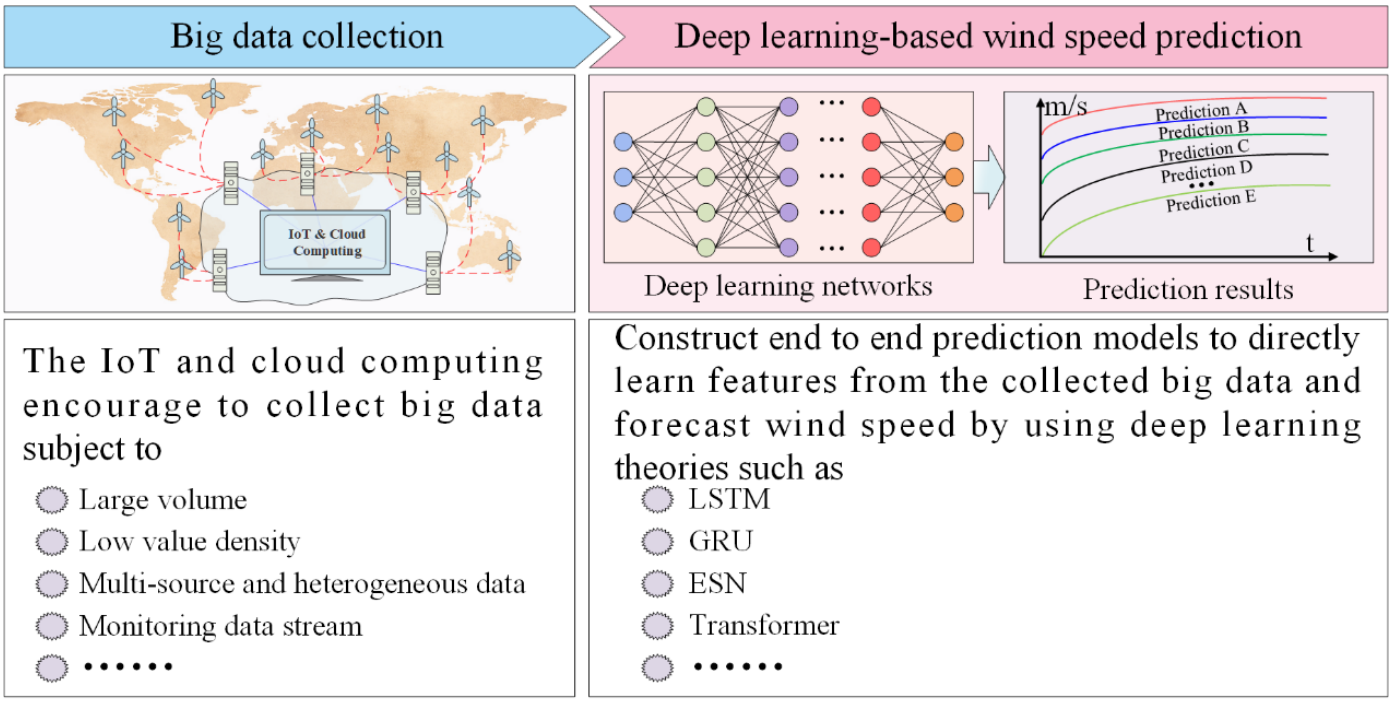


Figure 15 The Prediction Procedure of Deep Learning Methods

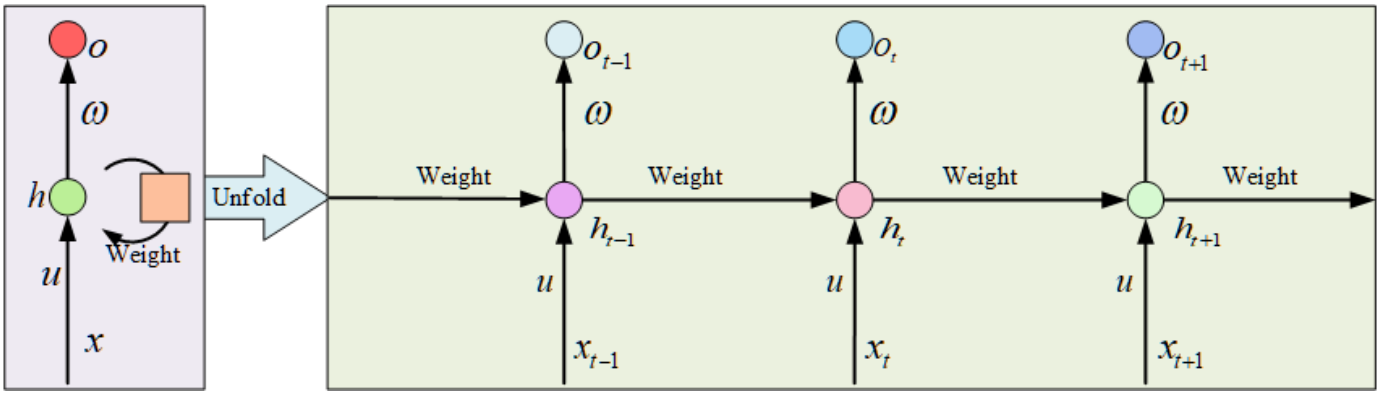


Figure 16 Structure of RNN

as depicted in Figure 16. Firstly, let the hidden state carry the network memory, as follows in Equation (46),

$$h_t = \sigma(\mu_h x_t + \omega_h h_{t-1} + b_s) \tag{46}$$

where h_{t-1} denotes the previous hidden state, x_t indicates the input for time t, b_s is the bias vector, and $\sigma(\cdot)$ denotes the activation function. μ_h and ω_h represent the weight matrices of x_t and h_{t-1}, respectively. Then, the current state is calculated as in Equation (47),

$$o_t = \omega_o h_t + b_o \tag{47}$$

where ω_o represents the weight matrices, and b_o denotes the bias vector.

3.2.2 A Brief Overview of LSTM

LSTM was first presented by Hochreiter et al. to overcome the problem of vanishing and exploding gradients in model training (Hochreiter & Schmidhuber, 1997), and is a more advanced RNN than a standard RNN. It consists of the input gate, output gate, and forget gate, as shown in Figure 17. The derivation formulas of LSTM are shown as follows in Eqs. (48-53):

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{48}$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{49}$$

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{50}$$

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \tag{51}$$

$$O_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{52}$$

$$h_t = o_t \cdot \tanh(C_t) \tag{53}$$

where h_{t-1} is the output from the previous, x_t indicates the input, f_t denotes ineffective information, $\sigma(\cdot)$ and $\tanh(\cdot)$ represent the activation function, i_t is updated information, $\tilde{C}_t$ denotes information of the hidden state in the memory cell, C_{t-1} indicates information of cell state from the previous, C_t represents information of cell at present, O_t indicates the output information of the hidden state. In addition, W_f, W_i, W_c,W_o represent the relevant weight matrices, respectively. b_f, b_i, b_c, b_o the relevant bias vector, separately.

3.2.3 A Brief Overview of GRU

GRU has a simpler structure than LSTM, firstly presented by Chung et al. (Chung et al., 2014), consisting of the update gate and the reset gate, as displayed in Figure 18. The update gate is utilized to update information in the current state that is stored in the previous state, and to accept information from the candidate hidden state. The reset gate can control

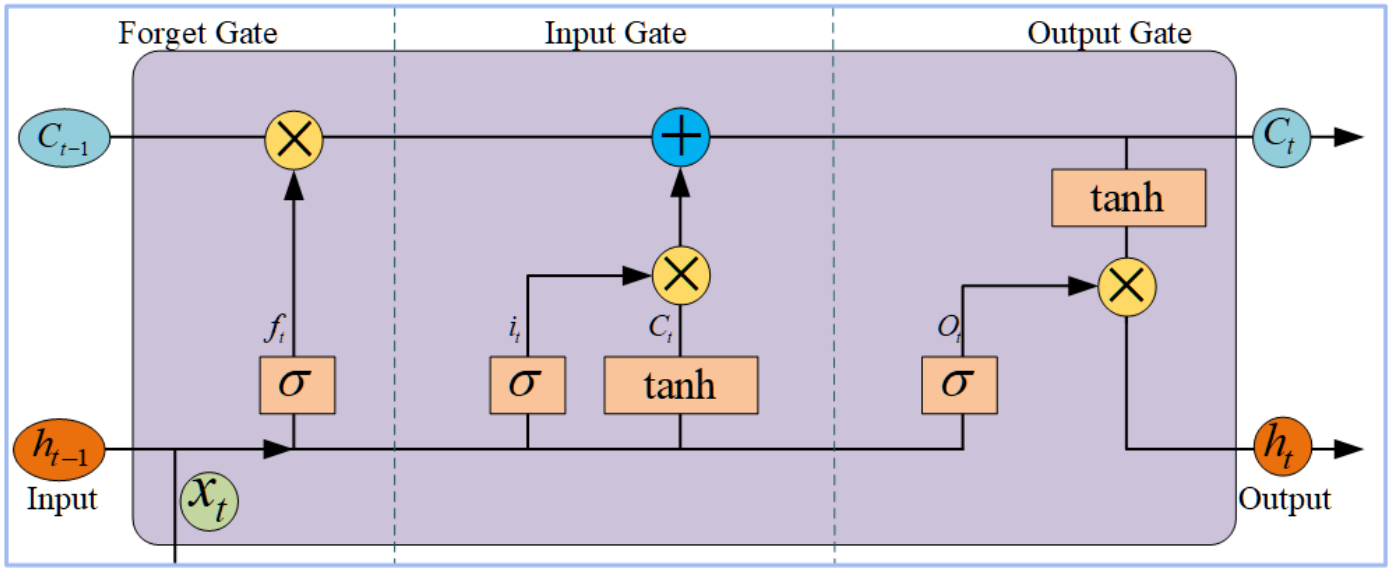


Figure 17 Structure of LSTM

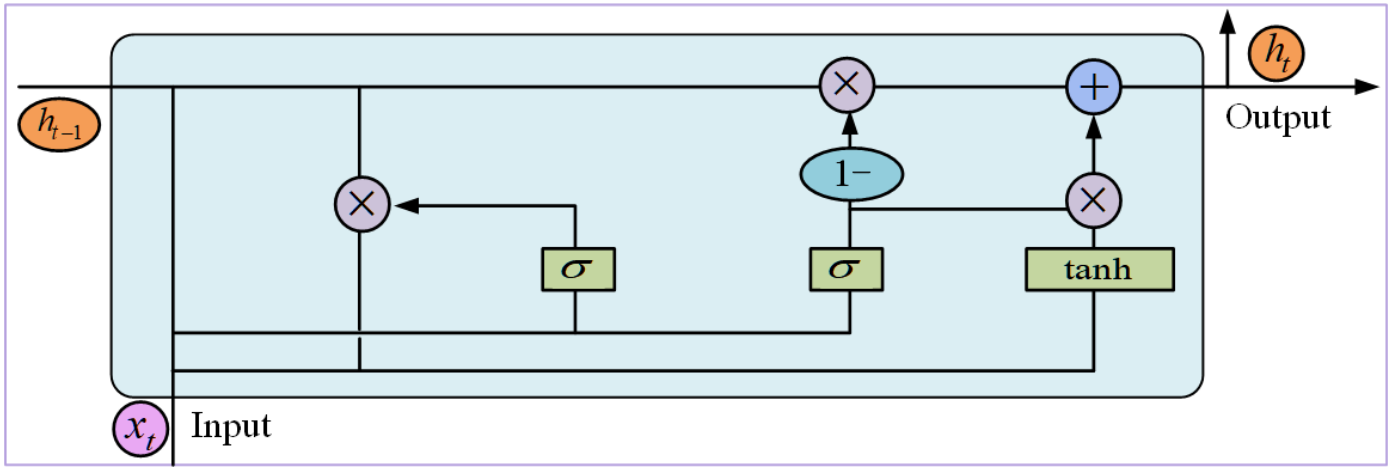


Figure 18 Structure of GRU

whether the calculation of the candidate hidden state is dependent on the previous state. The process can be expressed in Eqs. (54-57),

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \tag{54}$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \tag{55}$$

$$\tilde{h}_t = \tanh(W_h x_t + U_h(r_t h_{t-1}) + b_h) \tag{56}$$

$$h_t = z_t h_{t-1} + (1 - z_t)\tilde{h}_t \tag{57}$$

where z_t indicates information about the update gate, r_t denotes information of the reset gate, and $\tilde{h}_t$ represents new memory information. Moreover, W_z, W_r, W_h, U_z, U_r, U_h denote the relevant weight matrices, respectively. b_z, b_r, b_h represent the relevant bias vector, separately.

3.2.4 A Brief Overview of ESN

ESN, also known as reserve pool calculation, firstly was presented by Jaeger et al. (Jaeger, 2002). It utilizes a reservoir composed of neurons that are obtained from a random internal weight matrix as the hidden layer, then mapping it into a high-dimensional space to characterize features, as depicted in Figure 19. Its dynamics equations are defined as follows Eqs. (58–60),

$$x(t + \Delta t) = (1 - \alpha)x(t) + \alpha \tanh(\omega_{res} x(t) + \omega_{in} u(t)) \tag{58}$$

$$u(t) = [u_1(t), u_2(t), \dots, u_M](t)^T \tag{59}$$

$$x(t) = [x_1(t), x_2(t), \dots, x_M](t)^T \tag{60}$$

where α represents the leakage rate, $x(t)$ denotes a reservoir state vector, $u(t)$ is the input series at time t, and ω_{res}, ω_{in} indicate the parameter matrix corresponding to and $u(t)$.

3.2.5 Applications of RNN-Based Methods to WSF

Researchers developed studies on applications of RNN-based methods, including RNN, LSTM, GRU, and ESN, as depicted in Table 5.

As a baseline method in RNNs, RNN was developed to achieve wind speed series tasks. Barbounis et al. firstly utilized a diagonal RNN network for long-term wind speed forecasting (Barbounis et al., 2006). Cao et al. presented a comparative analysis of the wind speed forecasting accuracy of RNN with ARIMA counterparts (Cao et al., 2012). Yu et al. (2018) and Duan et al. (2021) developed a framework based on RNN and other models for wind speed prediction, respectively. Saeed et al. proposed an approach for short-term wind speed interval prediction based on RNN (Saeed et al., 2022). However, the problem of gradient vanishing occurs in RNNs owing to the increasing input length of time series data. To alleviate this problem, LSTM was presented that utilizes the input gate, forget gate, and output gate to control feature flow, thus achieving the preservation and transmission of time series information on a large scale. As a result, a large amount of the improved LSTM-based methods was proposed to predict wind speed, such as EnsemLSTM (Chen et al., 2018), BiLSTM (Biswas & Sinha, 2021; Moharm et al., 2020; Jaseena K U & Kovoor B C, 2021; Liang et al., 2021; Han et al.,

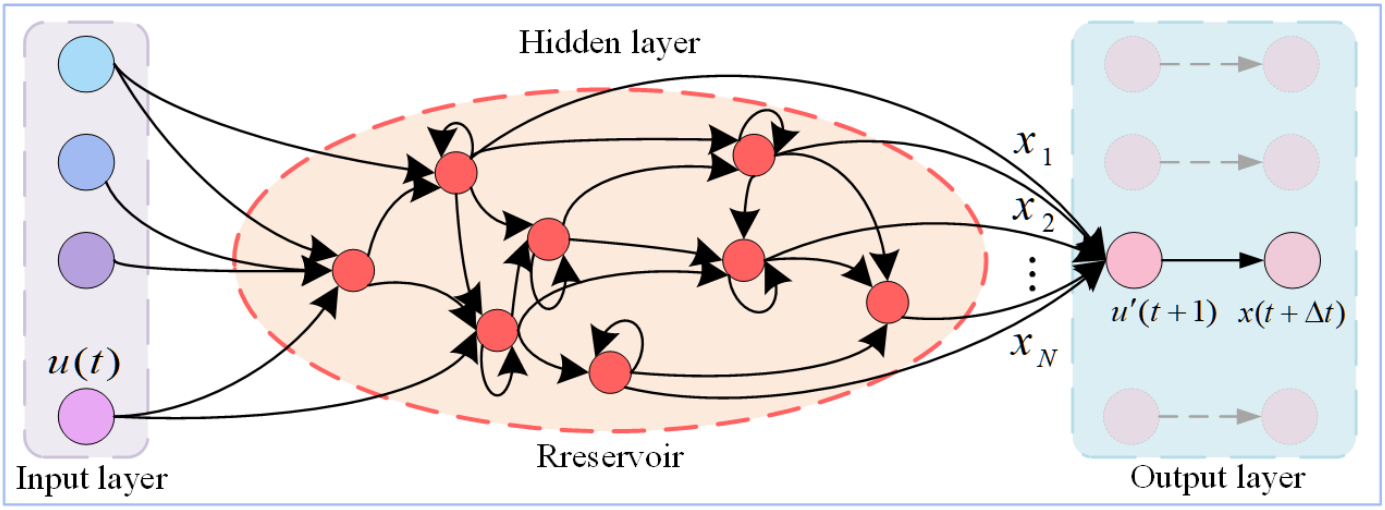


Figure 19 Structure of ESN

Table 5 Summary of Applications of Methods Based on NN to WSF

References	Methodologies	Forecast time scale
Barbounis and Theocharis (2006), Bilgili, Sahin, and Yasar (2007), Flores, Tapia, and Tapia (2005), Guo, Zhao, Lu, and Wang (2012), Liu et al. (2012a, 2012b, 2013a, 2013b, 2015a, 2015b), Cadenas and Rivera (2007)Ghorbani et al. (2016)Karakuş et al. (2017)Ruiz-Aguilar et al. (2021), Sheela and Deepa (2013), Yesilbudak et al. (2013), Yu et al. (2017), Zhang et al. (2016), Maatallah et al. (2015), Salcedo-Sanz et al. (2008, 2009a, 2009b), and Sanz et al. (2008), Verma et al. (2023)	ANN	Short-term
Guo et al. (2011), Meng et al. (2016), Ren et al. (2014), Barhmi et al. (2020), Qu et al. (2017, 2019), and Yang and Wang (2018)	BPNN	Short to medium-term
Liu et al. (2018a, 2018b), Wang et al. (2018), Chen et al. (2021), Fu et al. (2021, 2023), Guo et al. (2023), Hua et al. (2022), Li et al. (2019), Nie et al. (2020), Tian et al. (2020), Wang et al. (2020), and Zhang et al. (2019)	ELM	Very short to short-term
Doucoure et al. (2016), Safavieh et al. (2007), Santhosh et al. (2018), and Shi et al. (2018)	WNN	Short-term
Azad et al. (2014), Blanchard and Samanta (2020), Cadenas et al. (2016), Jamil and Zeeshan (2019), and Jawad et al. (2018)	NARX	Short to medium-term
Madhiarasan (2020), Sfetsos (2000), Wang et al. (2014), and Zhang et al. (2019)	RBFNN	Short-term

2022; Li et al., 2022), ConvLSTM (Chen et al., 2020; Ibrahim et al., 2020; Yan et al., 2023), EvLSTM (Huang et al., 2023), time-frequency recurrent neural network (TFR) (Yu et al., 2023), etc. Moreover, some subfields of WSF were beginning to emerge, including spatial-temporal (Khodayar & Wang, 2018; Shen et al., 2022; Wang et al., 2022; Yu et al., 2023; Zhu et al., 2021), probability interval (Wang et al., 2022; Liu et al., 2020; Zhu et al., 2019), and multimodal (Li et al., 2019; Zhang et al., 2019) wind speed prediction. More hybrid LSTM methods (Tyass et al., 2023; Zhang et al., 2019; Li et al., 2024; Yang et al., 2024; Saeed et al., 2025) are listed in Table 5.

Furthermore, GRU, a simple method based on LSTM, was constructed to improve computing efficiency. Cheng et al. first proposed a probabilistic wind speed forecasting approach based on GRU in WSF. Similar with methods based on LSTM, these methods were mainly utilized in the spatial-temporal (Chen et al., 2023; Peng et al., 2020; Ren et al., 2023; Wang & He, 2022; Xu et al., 2023; Ye et al., 2022; Chen et al., 2024; Ullah et al., 2025; Yao and Xu, 2026), probability interval (Afrasiabi et al., 2019; Afrasiabi et al., 2020; Tang et al., 2020; Wang et al., 2020; Xiang et al., 2020), and multimodal wind speed prediction(Afrasiabi et al., 2019). Besides, a method based on reservoir computing, ESN, was presented to avoid the issue of non-convergence and high computational costs by using the least-squares method. Liu et al. investigated a wind speed forecasting method based on ESN, which was the first time it was proposed in WSF (Zhang & Liu, 2020). In addition, some hybrid ESN methods were developed to extract the nonlinear features of wind speed (Bai et al., 2021; Chen et al., 2019; Chitsazan et al., 2017; Chitsazan et al., 2019; Ding et al., 2023; Hu et al., 2021; Tang et al., 2019; Tian et al., 2021).

These methods, including LSTM, GRU, and ESN, are developed to obtain higher generalization performance on the basis of the architecture of RNN. Currently, these prediction methods have been fully developed for wind speed forecasting and can obtain high prediction accuracy.

3.3 TCN-Based Methods

3.3.1 A Brief Overview of TCN

TCN was first proposed by Lea et al. (Lea et al., 2016), which is a generic convolutional network developed for sequence modeling tasks with a causal constraint. In a series modeling task, given the input series $\{\hat{x}_0, \hat{x}_1, \dots, \hat{x}_T\}$, the corresponding output series denotes $\{\hat{y}_0, \hat{y}_1, \dots, \hat{y}_T\}$, where only those previous observations $x_0, x_1, \dots, x_t$ can be taken as inputs to forecast the output y_t at time t. Accordingly, a nonlinear function is designed to represent TCN, as depicted in Equation (61),

$$\hat{y}_0, \dots, \hat{y}_T = f(x_0, \dots, x_T) \tag{61}$$

Under the causal constraint and to increase the receptive field, TCN utilizes dilated causal convolutions, which ensure that there is no leakage of information from the future to the past, as displayed in Figure 20 (a). The dilated convolution operation F on element s of the $1-D$ series $x \in R^n$ for a filter $f: \{0, 1, \dots, k-1\} \to R$ is formulated as Equation (62),

$$F(s) = (x *_d f)(s) = \sum_{i=1}^{k-1} f(i) \cdot x_{s-d \cdot i} \tag{62}$$

where k denotes the filter size, d indicates the dilation factor, and $*$ represents the convolution calculation. In addition, each layer in TCN utilizes a universal residual module for feature extraction. As shown in Figure 20 (b), the residual block consists of two layers of dilated causal convolution and nonlinearity, where the rectified linear unit serves as the activation function. Subsequently, the convolutional filter adopts batch normalization and adds spatial loss after each dilated convolution for regularization in each layer. Finally, an extra 1×1 convolution is used to ensure that elements are added using a tensor with a consistent shape due to input-output width incompatibility.

3.3.2 Applications of TCN-Based Methods to WSF

The applications of TCN-based methods to wind speed forecasting are categorized in Table 6. According to the architecture of CNN, TCN is constructed for application in time series forecasting. In order to construct a TCN-based prediction model and obtain high performance, researchers adopted some effective approaches. Gan et al. were the first to utilize TCN for wind speed interval prediction in WSF (Gan et al., 2021). Then, some hybrid TCN methods were proposed for the application of wind speed forecasting (Li et al., 2022; Kosana et al., 2022; Wang et al., 2023a, 2023b; Zheng et al., 2021; G. Zhang et al., 2024; Mi et al., 2025) (Table 7).

Due to the ability of convolutional networks to parallelize data computation, TCN can achieve better performance than RNN-based methods while utilizing convolutional networks to avoid common defects of recursive models, such as gradient explosion and vanishing. However, this convolutional network also results in poor generalization ability. Specifically, in different scenarios, the input data required for model prediction may be different, but the input of convolutional networks usually cannot be set arbitrarily; otherwise, it will lead to unsatisfactory prediction results.

3.4 Transformer-Based Methods

3.4.1 A Brief Overview of the Transformer

Transformer firstly was proposed by Vaswani et al. for natural language processing (NLP) (Vaswani et al., 2017), which relies entirely on the self-attention mechanism to learn features between input and output. It consists of the encoder and decoder blocks, whose structure is depicted in Figure 21.

(1) Encoder block. The encoder block is constructed by stacking several identical layers to learn the features in the input series. Two adjacent identical layers are linked by a residual connection, and then layer normalization is performed. Each layer consists of two sublayers: a multi-head self-attention and a fully connected feed-forward network. The multi-head attention is based on self-attention, which can deeply learn the internal feature relationships of the input series, as following Eqs. (63–66),

$$Q = \omega_Q X \tag{63}$$

$$K = \omega_K X \tag{64}$$

$$V = \omega_V X \tag{65}$$

$$Attn(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d}})V \tag{66}$$

where X denotes the input matrix, Q, K and V indicate the query matrix, key matrix, and value matrix, respectively. ω_Q, ω_K and ω_V are the weight matrices corresponding to Q, K and V, respectively. $\sqrt{d}$ represents a scale factor. Whereby, the multi-head attention can be derived from Eqs. (67, 68),

$$\text{MHA}(Q, K, V) = \text{Concat}(Head_1, \dots, Head_H)\omega^O \tag{67}$$

$$\text{Head}_i = \text{Attn}(X\omega_i^Q, X\omega_i^K, X\omega_i^V) \tag{68}$$

where ω_i^Q, ω_i^K and ω_i^V represent the weight matrices corresponding to Q, K and V, in $Head_i$. $Concat$ is utilized to splice the output of each $Head$, and ω^O is the parameter matrix. The fully connected feed-forward network can be expressed as a linear transformation given by Equation (69),

$$FFN = \max(0, X\omega_1 + b_1)\omega_2 + b_2 \tag{69}$$

where ω_1 and ω_2 denote the weights, and b_1, b_2 are the bias terms.

(2) Decoder block. Similar to the encoder block, the decoder block is also formed by stacking several identical layers. Each decoder layer consists of three sublayers: a masked multi-head self-attention, a fully connected feed-forward network, and a multi-head attention. The masked multi-head attention is the same as multi-head attention, except for the design of self-attention. In this design of self-attention, the weight is calculated by Equation (70),

$$\omega_{mask} = softmax(\frac{QK^T \odot M}{\sqrt{d}}) \tag{70}$$

where $\odot$ represents the element-wise multiplication operation, M denotes a lower triangular matrix. Then, the masked multi-head self-attention is derived by Equation (71),

$$MMHA(Q, K, V) = Contact(MHead_1, \dots, MHead_H)\omega^O \tag{71}$$

where $MHead_i = MAttn(X\omega_{mask}{}^i X\omega_{mask}{}^i X\omega_{mask}{}_i^V)$, and $MAttn(Q, K, V) = \omega_{mask}V$. Finally, a linear layer is performed to obtain the prediction.

3.4.2 Applications of Transformer-Based Methods to WSF

Some publications have introduced Transformer and its variants into WSF, as depicted in Table 8.

Bommidi et al. utilized Transformer and Improved Complete Ensemble EMD (ICEEMDAN) to complete the wind speed forecasting tasks, which was one of the earliest studies in WSF applications of Transformer (Bommidi et al., 2023). In addition, these methods are divided into hybrid Transformer methods and improved Transformer methods. Specifically, these hybrid Transformer methods are based on a Transformer model with some modules added, such as integrating ICEEMDAN (Bommidi et al., 2023), GCN (Pan et al., 2022), EMD (Li et al., 2023; Wu et al., 2022), GNN (Bentsen et al., 2023; Wu et al., 2022), VMD (Zhang et al., 2022), WT (Nascimento et al., 2023), CNN (Vo, 2023), Mixformer(Wu and Ling, 2024a), GAOformer(Lin et al., 2024), Evolutionary PatchTST(Ma et al., 2025). Different from these hybrid methods, these improved Transformer methods have improvements in the encoder or decoder block in order to make them available for wind speed forecasting (Bi et al., 2023; Deng et al., 2022; Lin et al., 2023; Wang et al., 2023; Zha et al., 2023). For example, Zha et al. introduced the convolutional layer into the encoder block for enhancing the ability of feature extraction in spatiotemporal wind speed forecasting (Zha et al., 2023). Furthermore, Bi et al. proposed a large-scale Pangu model based on a Transformer, which is the first large-scale model used in WSF (Bi et al., 2023). Wu and Ling proposed STELLM, a spatio-temporally enhanced wind speed forecasting model based on a pre-trained Transformer language model (Wu & Ling, 2024b). Liu et al. developed a diffusion Transformer for probabilistic multi-location wind speed forecasting (Liu et al., 2026). (Xia et

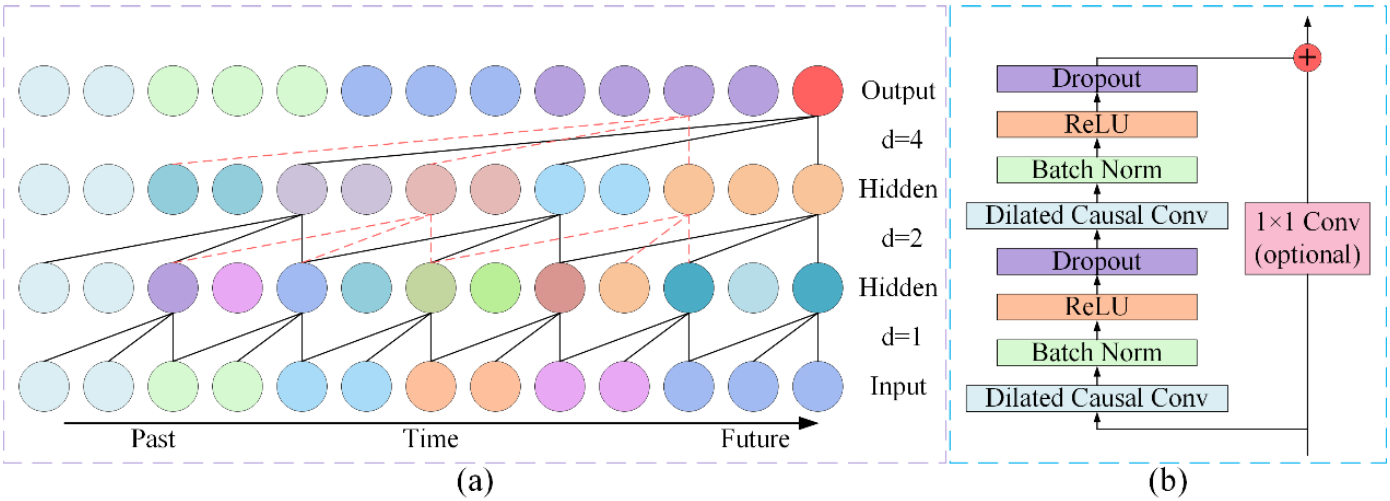


Figure 20 Structure of TCN: (a) A Dilated Causal Convolution; (b) A 1×1 Convolution is Added When Residual Input and Output Have Different Dimensions

Table 6 Summary of Applications of RNN-based Methods to WSF

References	Methodologies	Forecast time scale
Barbounis et al. (2006), Cao et al. (2012), Yu et al. (2018), Duan et al. (2021), and Saeed et al. (2022)	RNN	Short-term
Biswas and Sinha (2021), Chen et al. (2018), Chen et al. (2020), Han et al. (2022), Huang et al. (2023), Ibrahim et al. (2020), Jaseena and Kovoor (2021), Khodayar and Wang (2018), Li et al. (2022), Liang et al. (2021), Moharm et al. (2020), Saeed et al. (2022), and Yan et al. (2023), Gangwar et al. (2020), Hu and Chen (2018), Li et al. (2019), Liu et al. (2020), Shen et al. (2022), Wang et al. (2022), Wang et al. (2022), Zhang et al. (2019), Zhu et al. (2019), and Zhu et al. (2021), Li et al. (2018), Li et al. (2019), Liu et al. (2018), Liu et al. (2018), Liu et al. (2018), Pei et al. (2019), Son et al. (2019), Wang and Li (2018), and Wu et al. (2019), Chen et al. (2020), Fu et al. (2020), Geng and Wu (2020), Ibrahim et al. (2020), Li et al. (2020), Mi and Zhao (2020), Saeed et al. (2020), Xiang et al. (2020), Zhang et al. (2019), and Zhang et al. (2019), Chen et al. (2021), Elsaraiti and Merabet (2021), Elsaraiti and Merabet (2021), Geng et al. (2021), Nie et al. (2021), Shahid et al. (2021), Shao et al. (2021), Tian (2021), and Zhu et al. (2019), Chen et al. (2022), Huang et al. (2023), Hossain et al. (2023), Lv and Wang (2022), Sibtain et al. (2022), Tyass et al. (2023), Wang et al. (2021), Xu et al. (2021), Xiong et al. (2021), Zhang et al. (2022), Zhang et al. (2022), and Zhang and Chen (2021), Li et al. (2024), and Yang et al. (2024), Saeed et al. (2025)	LSTM	Very short to medium-term
Chen et al. (2023), Peng et al. (2020), Ren et al. (2023), Wang and He (2022), Xu et al. (2023), and Ye et al. (2022), Afrasiabi et al. (2019, 2020), Cheng et al. (2018), Ding et al. (2019), Ji et al. (2022), Kou et al. (2020), Liu et al. (2019), Liu et al. (2021), Lv et al. (2023), Meng et al. (2023), Tang et al. (2020), Wang et al. (2020, 2021), Wang and Gui (2022), Wu et al. (2022), Xiang et al. (2020), Yang et al. (2022), Yao et al. (2023), Yaghoubirad et al. (2023), Ye et al. (2022), Yu et al. (2022), Zhang and Liu (2020), Zhang et al. (2022), and Zou et al. (2020), Chen et al. (2024); Ullah et al. (2025), and Yao and Xu (2026)	GRU	Very short to short-term
Bai et al. (2021), Chen et al. (2019), Chitsazan et al. (2017, 2019), Ding et al. (2023), Hu et al. (2021), Liu et al. (2015), Tang et al. (2019), and Tian et al. (2021)	ESN	Short-term

al., 2026) systematically evaluated Transformer-based time-series models for post-processing WRF-simulated wind speeds.

Compared to other deep learning methods, Transformer-based methods achieve higher accuracy for wind speed forecasting. However, these Transformer-based methods are confronted with some challenges, which concern three aspects: first, due to the attention mechanism is high

Table 7 Summary of the Applications of TCN to WSF

References	Methodologies	Forecast time scale
Gan et al. (2021), Kosana et al. (2022), Li et al. (2022), Wang et al. (2023a, 2023b), and Zheng et al. (2021), Mi et al. (2025), G. Zhang et al. (2024)	TCN	Short-term

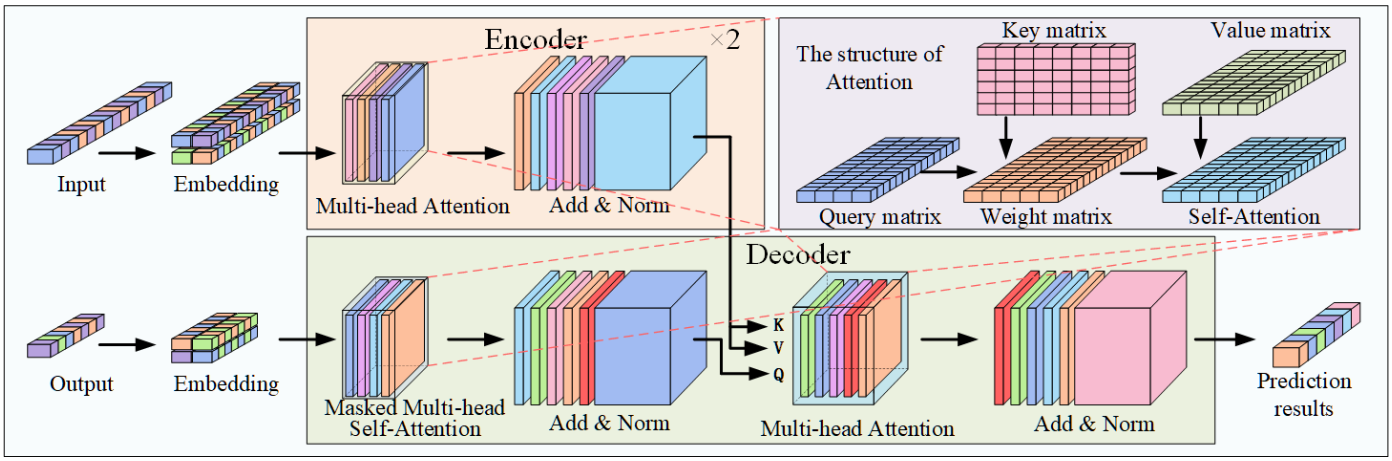


Figure 21 Structure of Standard Transformer

Table 8 Summary of the Applications of Transformer to WSF

References	Methodologies	Forecast time scale
Bentsen et al. (2023), Bommidi et al. (2023), Li et al. (2023), Nascimento et al. (2023), Pan et al. (2022), Vo (2023), Wu et al. (2022), Wu et al. (2022), and Zhang et al. (2022), Lin et al. (2024), Ma et al. (2025), Xia et al. (2026)	The hybrid Transformer	Short-term
Bi et al. (2023), Deng et al. (2022), Lin et al. (2023), Wang et al. (2023), and Zha et al. (2023), Liu et al. (2026), and Wu and Ling (2024), Wu and Ling (2024)	The improved Transformer	Short-term

computational complexity, Transformer has a long running time and high requirements for computer hardware. Second, Transformer has a larger number of parameters than other methods; therefore, it and its hybrid d methods are prone to overfitting, resulting in unsatisfactory results. Third, Transformers are also neural networks whose connections of neurons and high nonlinearity make them difficult to explain.

4. FUTURE: WSF USING TECHNOLOGIES DRIVEN BY LIGHTWEIGHT REAL-TIME METHODS AND ENSEMBLE METHODS-BASED DEEP REINFORCEMENT LEARNING

This section forecasts two of the most promising research prospects, i.e., applications of lightweight methods and ensemble methods, as shown in Figure 22.

In the lightweight methods, there are three subsections: (1) why use lightweight methods, (2) what lightweight means, (3) how to achieve lightweight. The first subsection focuses on the motivation for why lightweight methods have become a promising topic in future research on WSF. In the second subsection, lightweight methods are defined in WSF. In the third subsection, a few exploratory studies are reviewed according to the latest developments in WSF. In the ensemble methods, four subsections are analyzed: (1) why utilize ensemble methods, (2) what ensemble methods, (3) what are the challenges for ensemble methods, (4) how to achieve ensemble methods. The first section focuses on the motivation for ensemble methods to become a promising topic in future WSF research. Ensemble methods are defined in the second subsection. In the third subsection, challenges are discussed, and corresponding solutions are proposed. In the fourth subsection, some exploratory studies are reviewed.

4.1 Lightweight Methods for Real-Time Forecasting

4.1.1 Motivation of Applying Lightweight Methods in WSF

Currently, resource-constrained edge terminal devices, such as mobile phones and edge devices, have become more prevalent (Shapsough et al., 2018). Lots of real-world engineering scenarios have an imperative need for prediction capabilities on these devices, such as off-grid small wind generators and portable weather stations. However, limited computing power and storage hinder the development of deep learning in the IoT and the concurrent on-device applications of edge devices. Therefore, reducing parameters and generating compact models are crucial for enabling these edge devices to implement application scenarios that support deep learning under different constraints. The application of lightweight methods to remove redundant model parameters and connections not only enormously improves the popularity of deep learning models, but also greatly saves computational cost and energy consumption. As a result, lightweight methods are expected to expand from academic research to engineering scenarios in the future.

4.1.2 Definition of Lightweight Methods in WSF

Lightweight methods represent procedures for compressing the deep NAR into a more compact network while maintaining the same performance as before, which is applicable to execution at the edge terminal due to the limited resources and computing capacity of the edge device. Among the lightweight methods, the classic lightweight methods are SqueezeNet (Iandola et al., 2016), MobileNet (Sinha & El-Sharkawy, 2019), and ShuffleNet (Zhang et al., 2018), among which SqueezeNet is recognized as one of the earliest works of lightweight methods. In recent years, many achievements have been made for wind speed forecasting in the application of lightweight methods, and a variety of NAR lightweight algorithms have been designed, which can be divided into four categories, including pruning (Blalock et al., 2020), quantization (Gray & Neuhoff, 1998), knowledge distillation (Gou et al., 2021), and low-rank decomposition (Bach & Jordan,

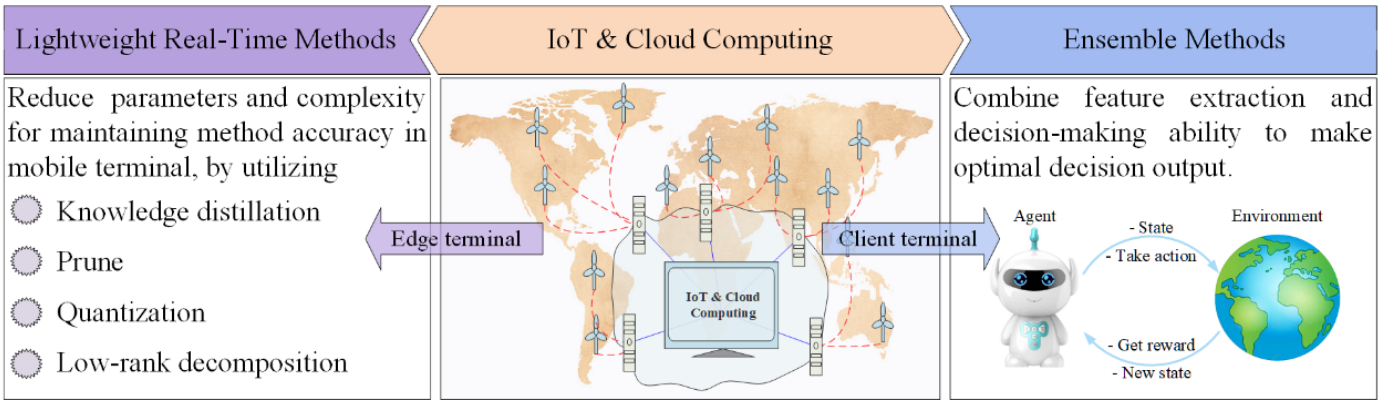


Figure 22 Technologies of Lightweight Real-Time Methods and Ensemble Methods-Based Deep Reinforcement Learning

2005).

(1) Pruning

Pruning is the procedure of removing redundant connections in a neural network, which makes these connections no longer participate in the forward or backward operations of the network to reduce the network computation. For instance, the neurons and corresponding connections are removed to reduce the storage of the model, as depicted in Figure 23.

In this process, an originally dense neural network becomes sparse due to the removal of partial connections. Pruning is usually performed on trained neural networks. The importance of each connection is defined according to some evaluation criteria after the trained model is obtained. A widely used evaluation criterion is the absolute value of the connection weight. The smaller the value, the smaller the importance of the corresponding neuron on the network output results. It is an unimportant neuron and should be removed, and the corresponding connection should also be cut.

(2) Quantization

Quantization is the process of compressing neural networks by reducing the number of bits representing each weight. Since the success of AlexNet in the ImageNet challenge in 2012, the upsurge in using quantization has become popular (Krizhevsky et al., 2012). Specifically, the weight parameters of the model are usually saved in the form of 32-bit floating-point numbers, which will occupy a large storage space since neural networks have tens of millions of parameters. Therefore, if the 32-bit floating-point number is quantized as an 8-bit fixed-point number when storing model parameters, the parameter size can be reduced to 1/4 of the original size, and the size of the entire model can also be reduced to 1/4 of the original size. Extremely, binary quantization only uses (-1) and (+1) to represent the weight value. In this way, a weight value only occupies one bit, and the parameter model is compressed to only 1/32 of the original. In addition, ± 1 has natural computing advantages in multiplication and addition operations compared with ordinary floating-point weight values, which leads to extremely high computing efficiency, as shown in Figure 24.

(3) Knowledge distillation

Knowledge distillation (KD) refers to a method of training a smaller network (i.e., student network) using a large, in-advance-trained network (i.e., teacher network), which transfers information from cumbersome models to smaller models that are more suitable for deployment. In particular, knowledge distillation is not a compression technique, but a method for training lightweight models (Gou et al., 2021). Knowledge distillation usually includes two types: a KD algorithm using a weighted average of two objective functions, and a TAS algorithm with multistage knowledge distillation. Specifically, the KD algorithm first trains the complex model with a higher temperature T, and then trains the simple model, as depicted in Figure 25.

At the same temperature T, calculate the cross-entropy of logits from the complex model and the simple model as the first objective function. Then, in the case of T1, calculate the cross-entropy of logits output by the simple model and the correct tag as the second objective function. Finally, the weighted average of the two objective functions is used as the total objective function of the simple model for network training. In the TAS algorithm, knowledge distillation utilizes a medium-sized network (i.e., teacher assistant) to bridge the gap between students and teachers.

(4) Low-rank decomposition

Low-rank decomposition is a method that decomposes the parameter matrix into a series of combinations of small matrices by using linear algebra, so that the combination of small matrices is basically consistent with the original convolution layer in terms of expressive ability. This method can maintain a certain accuracy of the model and greatly reduce the space occupied by parameter storage.

4.1.3 Application Scenarios of Lightweight Methods in WSF

Currently, lightweight methods have attracted the attention of some researchers in WSF. For example, SqueezeNet was introduced into BiGRU to construct Mixed-SqueezeNet-BiGRU for wind speed forecasting (Xu et al., 2022). In addition to using this lightweight network for wind speed prediction, lots of models utilize lightweight methods to improve existing models. The most commonly used lightweight method is pruning, such as in Informer (Zhou et al., 2021). Zenkner et al. proposed a flexible and lightweight model, which was based on quantization, reducing the model's parameters and enhancing efficiency (Zenkner & Navarro-Martinez, 2023). Recently, Pangu, a 3D NAR, was constructed to predict wind speed, which is based on low-rank decomposition theory, reducing the computational complexity of 3D attention using sliding windows (Bi et al., 2023). Beyond explicit pruning and quantization, computational efficiency can also be improved through compact hybrid design: (Xiong et al., 2024) couple decomposition with a multivariable grey model, whereas (Cheng and Shi,2026) use RestrictedCrossformer after multi-strategy MVMD to constrain attention computation.

In addition, unlike these lightweight methods based on deep learning, BLS has been used to solve the problem of model lightweight in WSF. This method is based on incremental learning and is much faster than deep learning, because it has no layer-to-layer connections in a NAR and does not need to use gradient descent to update the weight values. Nevertheless, there is still little research on BLS. Presently, Zhu et al. first utilized BLS for wind speed prediction, but the prediction results seem not satisfying (Zhang & Liu, 2020). Then, Xu et al. proposed a natural phase space reconstruction-based BLS for offshore wind speed forecasting, which significantly improved the accuracy and runtime of the model (Wang et al., 2021). Recently, Hu et al. presented an incremental forecaster using the C-C algorithm for phase space reconstruction and a broad learning network, which has made improvements based on the natural phase space reconstruction-based BLS (Hu et al., 2024).

However, the above methods only consider the running time of the model as an indicator and have not yet delved into a systematic exploration of running memory and running time. Both lightweight deep models and BLS aim to reduce computational overhead for edge-device deployment, but they differ fundamentally in their training paradigm. The key distinction lies in the use of BP. Lightweight deep models (e.g., SqueezeNet, MobileNet, ShuffleNet) are compressed variants of deep neural networks trained via iterative BP. They require layer-by-layer gradient computation over multiple epochs. For instance, Mixed-SqueezeNet-BiGRU (Xu et al., 2022) demands full forward and backward passes in every training epoch. BLS, in contrast, eliminates BP entirely by using random weights for feature and enhancement nodes and computing output weights via one-shot pseudo-inversion. As demonstrated by Xu et al. (2023), their natural phase space reconstruction-based BLS achieved training times several orders of magnitude faster than BP-trained models (BPNN, LSTM, GRU)—on the order of less than 0.1 s per training instance compared to tens of seconds for

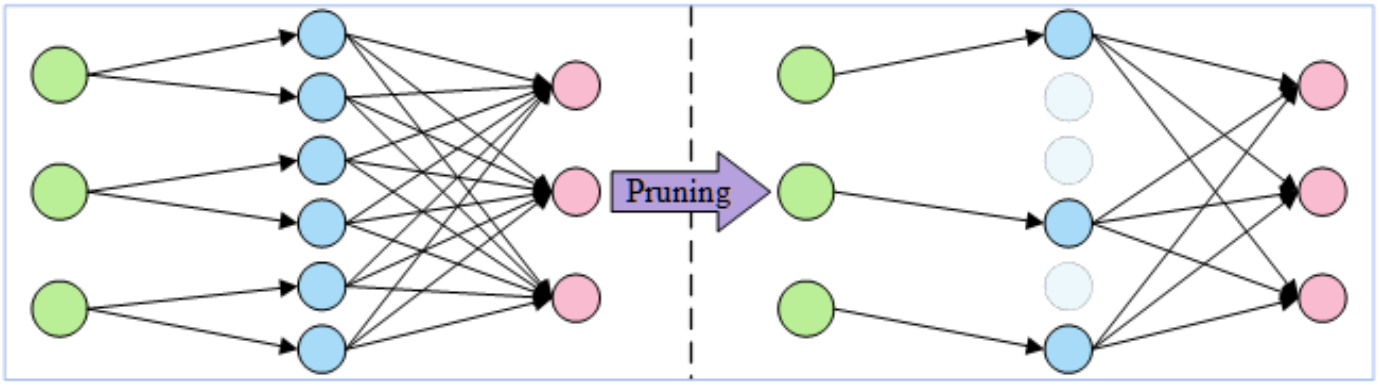


Figure 23 The Procedures of Pruning in a Neural Network Layer

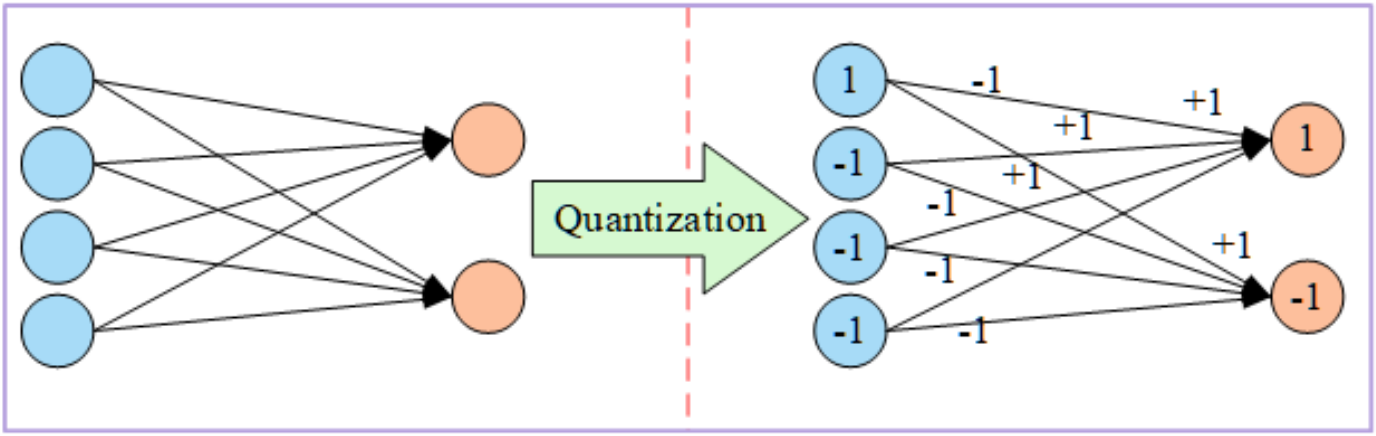


Figure 24 The Procedures of Binary Quantization Neural Network Layer

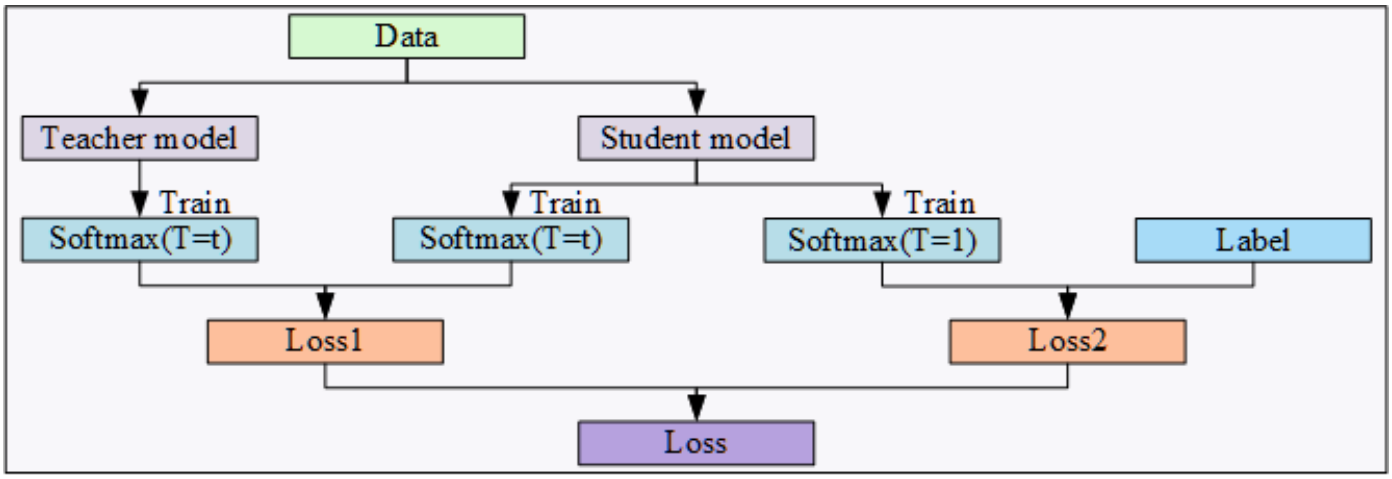


Figure 25 The Procedures of Knowledge Using Knowledge Distillation Algorithm

BP-based methods—while maintaining competitive single-step accuracy. However, lightweight deep models retain an advantage on multi-step-ahead tasks where deeper temporal modeling is essential. The trade-off is thus between speed (BLS) and accuracy on complex tasks (lightweight deep models).

4.2 Ensemble Methods Driven by Big Data

4.2.1 Motivation of Applying Ensemble Methods in WSF

Driven by the development of big data, deep learning methods have developed rapidly. Some hybrid methods have widely utilized the stacking ensemble strategy in WSF, which regards the prediction results of base models as features and trains another high-level model to combine these features. These hybrid methods enhance the performance of wind speed prediction, but still have limitations. (a) Focusing on the construction of a single foundation model difficultly for learning diversified characteristics. (b) Relying on the same combination weights for different wind speed features making methods learning a static process. Therefore, the combination weights of the basic model are reasonably changed according to the time-varying characteristics of wind speed, which may improve the predictive performance of the ensemble prediction model.

Fortunately, some deep reinforcement learning methods have been developed by combining traditional theories and deep learning in the field of reinforcement learning (Arulkumaran et al., 2017), which have made obvious breakthroughs in many AI tasks, such as earliest Deepmind (Powles & Hodson, 2017), AlphaGo (Wang et al., 2016), AlphaStar (Arulkumaran et al., 2019), and OpenAI Five (Berner et al., 2019). These applications of deep reinforcement learning have inspired the formation of ensemble methods. Therefore, these ensemble methods can utilize reinforcement learning theory to optimize the combined weights of diverse heterogeneous deep learning neural networks in the sublayer, which can effectively characterize the time-varying features of wind speed and improve the accuracy of wind speed prediction.

4.2.2 Definition of Ensemble Methods in WSF

In this paper, the ensemble method is defined as a deep reinforcement learning method that combines reinforcement learning and deep learning in WSF. Reinforcement learning proposes a formal framework, as shown in Figure 26, where artificial agents can learn by interacting with the environment, similar to biological agents.

The state S_t denotes the agent's observation of its environment, the action A_t represents the approaches that the agent uses to interact with the environment, and the reward R_t indicates the bonus that the agent gets once it takes an action in the environment at the given time step$t$. Furthermore, deep reinforcement learning utilizes the powerful representation ability of neural networks to fit strategies of reinforcement learning. Common algorithms for deep reinforcement learning include deep Q-networks (DQN), deep deterministic policy gradient (DDPG), and proximal policy optimization (PPO) (Arulkumaran et al., 2017). For wind speed forecasting, deep reinforcement learning is usually utilized to determine the weights of the neural network in order to improve the ability of learning features. DRL optimizes ensemble weights by formulating the problem as an MDP. The state s_t encodes current wind conditions, including recent speed observations, variance, gradient, the forecast errors of each base model over a look-back window, and contextual features such as time of day, season, and turbulence intensity. The action a_t is a weight vector w_t = [w_1, w_2, ..., w_n] where each w_i ranges from 0 to 1 and all weights sum to 1, representing each base model's contribution to the ensemble prediction y_hat_t = sum_i w_i times f_i(x_t). The reward r_t equals the negative Root mean square error (RMSE) of the ensemble prediction, providing immediate feedback to guide learning.

Through repeated interaction with historical data, the DRL agent learns a policy pi(a_t|s_t) that dynamically shifts weights in response to changing wind regimes. Consider an ensemble of three BP-trained models: BPNN, LSTM, and GRU. Without DRL, standard practice is to average their outputs with fixed weights such as one-third each. With DRL, the agent discovers that the BPNN tends to overfit on volatile segments while the GRU adapts more quickly to regime shifts. The learned policy therefore assigns lower weight to the BPNN during high-volatility periods and shifts weight toward the GRU and LSTM. Similarly, if one model consistently outperforms others under specific conditions, for instance, LSTM during ramp events, the policy up-weights it automatically. This adaptive weighting typically reduces ensemble RMSE by 5 to 15 percent compared to fixed-weight averaging.

Common DRL algorithms for this task include DQN for discrete weight configurations, DDPG for continuous weight assignment, and PPO for stable policy updates in high-dimensional weight spaces.

4.2.3 Challenges for Ensemble Methods in WSF

Although ensemble methods can effectively improve the accuracy of wind speed prediction, this requires a large amount of data to train a NAR. To solve this problem, some effective methods can be referred to, such as GAN (Creswell et al., 2018), meteorological manifold simulation (Milliez & Carissimo, 2007), and Transfer learning(Torrey & Shavlik, 2010).

GAN

Introduction to GAN. It is inspired by the two-person zero-sum game in game theory, which was pioneered by Goodfellow et al. (Creswell et al., 2018), and includes a generative model G and a discriminative model D. The generation model captures the distribution of sample data, and determines whether the model is a two classifier, which determines whether the input is real data or generated samples. The optimization process of this model is a minimax two-player game problem. When training, fix one side, update the parameters of the other model, and iterate alternately to maximize the error of the other side. Finally, the generative model G can estimate the distribution of sample data. Its algorithm is depicted in Table 9.

Currently, some researchers have utilized GANs as a tool for data augmentation in time series prediction, such as TTS-GAN (Li et al., 2022), MTSS-GAN (Snow, 2020), TimeVAE (Desai et al., 2021), PSA-GAN (Jeha et al., 2021), GAPGAN (Liu et al., 2024). It is worth noting that these methods can model long sequence data points with irregular time relationships to learn features, which has the potential to generate data that accurately represents the original data, while also flexibly generating data with novel and unique relationships. Specifically, this will be a possible solution to the following two scenario problems: (a) the quality problem of noise and missing wind speed data, (b) the quantity problem of insufficient wind speed data.

Meteorological Numerical Simulation

A brief overview of meteorological numerical simulation. It denotes that the computer uses a numerical model to simulate the state and change of the atmosphere by controlling the combination of physical factors of the model and the structure of the initial field, which studies the change law of a certain part or the whole atmosphere. In particular, it can simulate atmospheric movement utilizing the physical factors of the target wind farms and the structure of the initial wind farms, thereby obtaining simulated wind speed information, including wind speed, wind direction, atmospheric pressure, etc.

At present, the popular meteorological numerical simulation methods include WRF (Done et al., 2004) and UM (Uranchimeg et al., 2017). Generally, when the physical mode and initial field of the wind field can be obtained, the meteorological numerical simulation will be able to overcome some problems: (a) the quality problems of noise and missing multimodal wind speed data, (b) the relevance problems between cluster wind speeds in the wind farm, (c) the quantity problems of insufficient wind speed data.

Transfer Learning

A brief overview of Transfer learning. It is different from the above solutions, in which the knowledge from one or more prediction tasks can be reused for other related but different ones. For transfer learning in WSF, wind speed prediction is expected to be reused from one or more prediction tasks (source domain) to other related but different prediction tasks (target domain), which denotes the concept of domain and task. The domain usually is indicated as a pair of $D = \{X, P(X)\}$, where $X = \{x_i\}_{i=1}^n$ is the dataset, and $P(X)$ denotes its marginal probability distribution. Then, the prediction task is represented as $T = \{Y, f(\cdot)\}$ consisting of the label space $Y = \{y_i\}_{i=1}^n$ and the prediction model $f(\cdot)$. The prediction model $f(\cdot)$ can be trained with the labeled data $\{x_i, y_i\}_{i=1}^n$. After that, the model is further endowed with learned knowledge, i.e., the relationship between the input data and their corresponding prediction results. Given the source domain D^s, the target domain D^t, the prediction task T^s, and the transfer task T^t, transfer problems aim to apply the prediction knowledge from task T^s to improve the performance of the prediction model $f(\cdot)$ on related task T^t, where $D^s \neq D^t$, and $T^s \neq T^t$.

Currently, the application of transfer learning has existed but has not been popularized in WSF. The existing transfer methods focus on the research of wind speed evaluation for building a new wind farm or the wind farm location problem (Hu et al., 2016; Liang et al., 2021). In the future, the application scenario of transfer learning should focus on the following aspects: (a) The training label problem of wind speed big data. Specifically, in model training, it usually relies on labeled data, but labeling data is very cumbersome. It is noteworthy that transfer learning can make the model learn knowledge from richly labeled data and then only utilize a small amount of prediction data to fine-tune the model, so that the model can forecast the target wind speed of a small amount more accurately. (b) The missing value problems of a wind turbine in large wind farms. For example, the complex working conditions of the turbine may break down, leading to data loss, and other turbines in the same wind farm are similar in their working conditions; therefore, data from other turbines can be transferred to the failed turbines will improve the prediction accuracy.

4.2.4 Application Scenarios of Ensemble Methods in WSF

Presently, some deep reinforcement learning methods have been utilized in WSF. Liu et al. first utilized deep reinforcement learning and proposed an ensemble method combining LSTM, deep belief networks (DBN), and ESN for forecasting wind speed (Liu et al., 2020). Chen et al. presented a deep reinforcement learning framework based on bias-variance analysis to predict wind speed (Chen & Liu, 2021). Malik et al. presented a first effort utilizing fuzzy reinforcement learning for WSF (Malik & Yadav, 2021). Rui et al. established an ensemble model consisting of data adaptive decomposition, reinforcement learning ensemble, and improved error correction (Yang et al., 2022).

Compared with deep learning, these ensemble methods have advantages for wind speed prediction in the following two aspects: (a) They can adaptively learn the optimal strategy without the need for artificially

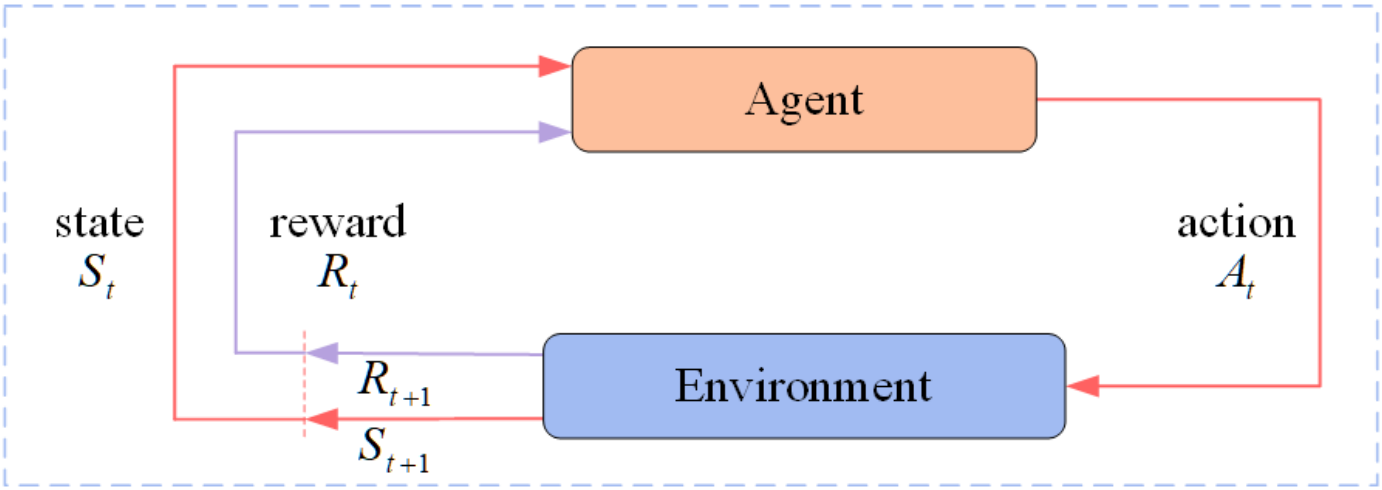


Figure 26 The Procedures of Reinforcement Learning

Table 9 The Algorithm Flow of GAN

Algorithm 1: Minibatch SGD training of GANs

Require: k (steps to apply to discriminator, hyperparameter; we used k=1),
m (minibatch size),
learning rates (any standard rule; we used momentum)
for number of training iterations do
for k steps do
Sample minibatch of m noise samples {z^(1),...,z^(m)} from p_g(z)
Sample minibatch of m examples {x^(1),...,x^(m)} from p_data(x)
Update discriminator by ascending its stochastic gradient:
∇_{θ_d} (1/m) Σ_{i=1}^m [log D(x^(i)) + log(1 - D(G(z^(i))))]
end for
Sample minibatch of m noise samples {z^(1),...,z^(m)} from p_g(z)
Update generator by descending its stochastic gradient:
∇_{θ_g} (1/m) Σ_{i=1}^m [log(1 - D(G(z^(i))))]
end for
Note: Gradient-based updates can use any standard learning rule (e.g., momentum).

designed features. For instance, the weights of various heterogeneous benchmark models can be adaptively set to the optimal value, which can maintain the dynamics and optimality of the model. (b) It can perform transfer learning in different tasks and environments to improve learning efficiency. Significantly, ensemble methods can combine transfer learning methods, transfer abundant data to various tasks, and use deep reinforcement learning to solve the problem of insufficient task data and achieve better predictive performance. However, there are still some unresolved issues to promote ensemble methods, mainly including two issues. (a) A large amount of data and computing resources. (b) The interpretability of results. In general, ensemble methods based on deep reinforcement learning are still a subject that has not been thoroughly studied in WSF.

5. DISCUSSIONS: FUTURE CHALLENGES AND A ROADMAP IN WSF

5.1 How to Provide High-Quality Big Data for Training WSF Methods Based on Deep Learning?

In the present period of WSF, the volume of collected data is expanding rapidly. Nevertheless, the quality of the data collected is not satisfactory, since a portion of it might be of poor quality. These poor-quality data with inaccurate, uncertain, incomplete, and not time-stamped, are defined as incorrect data. These incorrect data will result in unreliable prediction results when they are directly utilized to train the prediction models. Clustering algorithms and reasoning models may help separate data and further improve the quality of collected data, such as K-means (Ahmed et al., 2020), Gaussian mixture model (GMM) (Bond et al., 2001), hierarchical clustering (Murtagh & Contreras, 2012), and GAN.

decision-making. (b) Attention mechanism (Niu et al., 2021). The attention causal relationships in data, thus improving the accuracy of prediction and (An et al., 2022). Deep learning is adept at extracting complex and abstract high-dimensional features in data. Therefore, combining deep learning with causal networks can help us better understand and use

5.2 How to Construct Deep Learning-Based WSF Methods Subject to Special Issues in Big Data Revolution?

With the revolution of big data, a special problem arises that results in negative effects on WSF using deep learning, i.e., analytics for data streams. The monitoring data stream is regarded as an increasing data stream in continuous time series. Through analysis, WSF can provide real-time prediction results. This problem has been concerned by researchers for many years, but due to unreliable data transmission and bandwidth constraints, the monitoring data stream cannot reach the destination in an uninterrupted sequence. In addition, inefficient computing power hinders the success of data stream analysis. Therefore, WSF using deep learning is only implemented based on offline historical data. With the emergence of the IoT, broadband Internet, and cloud computing, monitoring data streams can be collected and effectively processed. Therefore, the development of online WSF is encouraged to make real-time decisions on grid dispatching of wind farms. Incremental learning is expected to promote the use of deep learning online WSF and continuously acquire prediction knowledge from the monitoring data stream.

5.3 How to Improve the Interpretability of the Deep Learning-Based WSF Methods?

Although wind speed prediction models based on deep learning have achieved state-of-the-art results, the open problem of the black box in deep networks still confuses researchers in the academic community. It is difficult to understand how these prediction models learn wind speed features from historical data. Therefore, the prediction model is constructed through experimental tests again and again rather than a strict theoretical background. In order to make up for this problem, it is suggested to focus on two research interests. (a) Causal network

mechanism in deep learning is a method of imitating human visual and cognitive systems, which allows neural networks to focus on relevant parts when processing input data. By introducing the attention mechanism, the NAR can automatically learn and selectively focus on the important information in the input, and improve the performance and generalization ability of the model.

5.4 Computational Burden Across WSF Methods

Beyond accuracy, the computational burden of WSF methods is a critical factor for practical deployment, especially in edge computing, online prediction, and large-scale wind farm scenarios. This subsection provides a systematic analysis of the computational complexity, memory requirements, and incremental learning capability of each method category discussed in this review. Table 9 presents a comprehensive comparison across all major WSF approaches.

Several key observations emerge from Table 10. First, kernel-based methods (GPR, SVR) carry the heaviest computational burdens, with GPR requiring O(N3) operations and O(N2) memory that make it impractical for large-scale WSF. Empirical evidence from offshore wind farm studies confirms this: a GPR model required over 11,800 seconds for a single update cycle (Xu et al., 2023), which is three orders of magnitude slower than BLS alternatives.

Second, deep learning methods (LSTM, GRU, Transformer) suffer from a fundamental architectural limitation: they lack native incremental learning capability. When new wind speed data become available, these models must be retrained from scratch on the entire historical dataset, a process requiring 3,000 to 5,500 seconds per update (Xu et al., 2023). In real-world scenarios where data are collected at intervals of minutes or even seconds, this retraining overhead renders online prediction infeasible.

Third, BLS with incremental learning offers a unique combination of low computational burden and high prediction accuracy. By computing output weights via ridge regression approximation rather than iterative gradient descent, BLS avoids local minima trapping and enables rapid model updates at only 2.18 seconds per new data batch (Xu et al., 2023). This makes BLS the only method that can satisfy both the accuracy and timeliness requirements of online WSF simultaneously.

Fourth, shallow networks such as ELM and BPNN exhibit fast training times (0.12 s and 98.41 s, respectively) but fail to extract the deep nonlinear features present in wind speed data, resulting in significantly larger prediction errors. This accuracy-efficiency trade-off underscores the importance of architectural design choices that balance model capacity with computational feasibility.

Finally, future methods in the lightweight and ensemble categories must carefully consider computational costs. While knowledge distillation and network pruning can reduce model size, deep reinforcement learning ensembles multiply the training cost by the ensemble size K, and GAN-based data augmentation requires adversarial training with two interacting networks. These computational demands remain open challenges that the roadmap in the following subsections must address.

Table 10 The Definition of Time Horizon in WSF

Class	Memory	Incremental	Online feasibility	Relative burden
NWP/WRF (Physical)	Very high (HPC)	No	No	Critical
ARIMA (Statistical)	Low (~MB)	No	Yes	Negligible
GPR (Kernel)	O(N2) ~GB	No	No	Critical
SVR/LSSVR (Kernel)	O(N2)	No	Partial	High
ELM (Shallow)	Low (~10 MB)	No	Yes	Minimal
BPNN (Shallow)	Low (~10 MB)	No	Yes	Low
BLS (Broad)	Moderate (~100 MB)	Yes	Yes	Very Low
LSTM/GRU (Deep)	High (~GB)	No	No	High
TCN (Deep)	Moderate (~500 MB)	No	Partial (GPU)	Moderate
Transformer (Deep)	Very high (GB+)	No	No	Very High
ESN (Reservoir)	Moderate	No	Yes	Low
DRL Ensemble	Very high	No	No	Critical
GAN	Very high	No	No	Very High

6. CONCLUSION

In this paper, the applications of WSF have been reviewed, which are roughly divided into three periods. In the past, WSF was implemented based on physical and statistical methods. By utilizing traditional machine learning theories, the prediction methods are able to automatically forecast future wind speed. However, the above literature reviews show that feature extraction still relies on expert knowledge. In recent years, the emergence of deep learning has brought positive effects and higher benefits. Deep learning theories construct end-to-end prediction methods that automatically learn features from large-scale historical wind speed data and subsequently obtain the prediction results. In the future, it should be noted that the success of deep learning prediction models based on big data depends on the performance of the devices. Due to device limitations, it is bound to develop in two directions: lightweight and ensemble. Therefore, technologies driven by lightweight real-time methods and ensemble methods based on deep reinforcement learning are promising for constructing prediction models, which these models can be applied in scenarios of edge or IoT devices, respectively. Finally, we discuss the challenges and further depict a roadmap. This review is expected to systematically present the development of WSF and provide valuable guidelines for future research in this field.

7. APPENDIX

7.1 The Common Metrics

Some common metrics have been organized and displayed in Table 11, where y and $\hat{y}$ respectively denote the observed and predicted wind speeds.

Table 11 The Common Metrics

The metrics	The computational formulas
MSE	$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$
RMSE	$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$
The MAE	$MAE = \frac{1}{n}\sum_{i=1}^{n}\lvert y_i - \hat{y}_i\rvert$
The MAPE	$MAPE = \frac{1}{n}\sum_{i=1}^{n}\lvert\frac{y_i - \hat{y}_i}{y_i}\rvert \times 100\%$
Symmetric mean absolute percentage error (SMAPE)	$SMAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{2\lvert y_i - \hat{y}_i\rvert}{\lvert y_i\rvert + \lvert\hat{y}_i\rvert} \times 100\%$
Symmetric mean absolute percentage error (SMAPE)	$SSE = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$
The coefficient of determination (R2)	$R^2 = 1 - \frac{1}{n}\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \overline{y}_i)^2}$
Theil inequality coefficient (TIC)	$TIC = \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}\hat{y}_i^2} + \sqrt{\frac{1}{n}\sum_{i=1}^{n}y_i^2}}$
Pearson correlation coefficient (PCC)	$PCC = \frac{\sum_{i=1}^{n}(y_i - \overline{y}_i)(\hat{y}_i - \hat{y}_i)}{\sqrt{\sum_{i=1}^{n}(y_i - \overline{y}_i)^2 \times \sum_{i=1}^{n}(\hat{y}_i - \hat{y}_i)^2}}$

7.2 The Definitions of Time Horizon in WSF

The definitions of the time horizon in WSF are presented in Table 12.

Table 12 The Definitions Time Horizon in WSF

Time horizon	Range
Ultra-short-term forecasting	Few seconds to 1 hour ahead
Short-term forecasting	1 hour to several hours ahead
Medium-term forecasting	several hours to 1 week ahead
Long-term forecasting	1 week to 1 year or more ahead

ACKNOWLEDGMENTS

This research is supported by the National Natural Science Foundation of China under Grant 62403406; in part by the Natural Science Foundation of Hebei Province under Grant F2025203081.

REFERENCES

A. Desai, C. Freeman, Z. Wang (2021). Timevae: a variational auto-encoder for multivariate time series generation. arxiv preprint arxiv:2111.08095.

A. Sfetsos (2000). A comparison of various forecasting techniques applied to mean hourly wind speed time series. *Renewable Energy*, *21*(1), 23–35.

Adnan, R. M., Liang, Z., Yuan, X (2019). Comparison of lssvr, m5rt, nf-gp, and NF-SC models for predictions of hourly wind speed and wind power based on cross-validation. *Energies*, *12*(2), 329.

Afrasiabi, M., Mohammadi, M., Rastegar, M (2020). Advanced deep learning approach for probabilistic wind speed forecasting. *IEEE Transactions on Industrial Informatics*, *17*(1), 720–727.

Aggarwal, C. C (2018). Neural networks and deep learning. springer.

Al-Deen, S., Yamaguchi, A., Ishihara, T (2006). A physical approach to wind speed prediction for wind energy forecasting. *Journal of Wind Engineering*, *108*, 349–352.

Allen, D. J., Tomlin, A. S., Bale, C. S. E (2017). A boundary layer scaling technique for estimating near-surface wind energy using numerical weather prediction and wind map data. *Applied Energy*, *208*, 1246–1257.

Al-Yahyai, S., Charabi, Y., & Gastli, A (2010). Review of the use of numerical weather prediction (nwp) models for wind energy assessment. *Renewable and Sustainable Energy Reviews*, *14*(9), 3192–3198.

An, W., Beauvile, R., & Rosche, B (2022). Causal network analysis. *Annual Review of Sociology*, *48*, 23–41.

Arenas-López J P, & Badaoui M (2020). Stochastic modelling of wind speeds based on turbulence intensity. *Renewable Energy*, *155*, 10–22.

B. Doucoure, K. Agbossou, & A. Cardenas (2016). Time series prediction using artificial wavelet neural network and multi-resolution analysis: application to wind speed data. *Renewable Energy*, *92*, 202–211.

Bach, F. R., & Jordan, M. I (2005). Predictive low-rank decomposition for kernel methods. *In Proceedings of the 22nd International Conference on Machine Learning*, *33–40*.

Bai, Y., Liu, M. D., Ding, L (2021). Double-layer staged training echo-state networks for wind speed prediction using variational mode decomposition. *Applied Energy*, *301*, 117461.

Bali, V., Kumar, A., & Gangwar, S (2019). Deep learning based wind speed forecasting-a review. *In 2019 9th international conference on cloud computing, data science & engineering (confluence)*, *426*–431. IEEE.

Barbounis, T. G., & Theocharis, J. B (2006). Locally recurrent neural networks for long-term wind speed and power prediction. *Neurocomputing*, *69*(4–6), 466–496.

Barbounis, T. G., Theocharis, J. B., Alexiadis, M. C (2006). Long-term wind speed and power forecasting using local recurrent neural network models. *IEEE Transactions on Energy Conversion*, *21*(1), 273–284.

Barhmi, S., Elfatni, O., & Belhaj, I (2020). Forecasting of wind speed using multiple linear regression and artificial neural networks. *Energy Systems*, *11*, 935–946.

Bentsen, L. Ø., Warakagoda, N. D., Stenbro, R., & Engelstad, P (2023). Spatio-temporal wind speed forecasting using graph networks and novel transformer architectures. *Applied Energy*, *333*, 120565.

Bi, K., Xie, L., Zhang, H., & Chen, X (2023). Accurate medium-range global weather forecasting with 3d neural networks. *Nature*, *619*(7970), 533–538.

Bilgili, M., Sahin, B., & Yasar, A (2007). Application of artificial neural networks for the wind speed prediction of target station using reference stations data. *Renewable Energy*, *32*(14), 2350–2360.

Biswas, S., & Sinha, M (2021). Performances of deep learning models for indian ocean wind speed prediction. *Modeling Earth Systems and Environment*, *7*, 809–831.

Björck, Å (1990). Least squares methods. *Handbook of Numerical Analysis*, *1*, 465–652.

Blalock, D, & Gonzalez Ortiz, J. J., Frankle, J., & Guttag, J (2020). What is the state of neural network pruning? proceedings of machine learning and systems, 2, 129–146.

Bokde, N., Feijóo, A., Villanueva, D (2019). A review on hybrid empirical mode decomposition models for wind speed and wind power prediction. *Energies*, *12*(2), 254.

Bommidi, B. S., Teeparthi, K., & Kosana, V (2023). Hybrid wind speed forecasting using ICEEMDAN and transformer model with novel loss function. *Energy*, *265*, 126383.

Bond, S. R., Hoeffler, A., & Temple, J. R. W (2001). GMM estimation of empirical growth models. available at SSRN 290522.

Bouckaert, R. R (1995). Bayesian belief networks: from construction to inference. [technical report].

Box, G. E. P., & Pierce, D. A (1970). Distribution of residual autocorrelations in autoregressive-integrated moving average time series models. *Journal of the American Statistical Association*, *65*(332), 1509–1526.

C. Berner, G. Brockman, B. Chan (2019). Dota 2 with large scale deep reinforcement learning. arxiv preprint arxiv:1912.06680.

C. Chen & H. Liu (2021). Dynamic ensemble wind speed prediction model based on hybrid deep reinforcement learning. *Advanced Engineering Informatics*, *48*, 101290.

C. Huang, H. R. Karimi, P. Mei (2023). Evolving long short-term memory neural network for wind speed forecasting. *Information Sciences*, *632*, 390–410.

C. Zhang, H. Ma, L. Hua (2022). An evolutionary deep learning model based on tvfemd, improved sine cosine algorithm, CNN and bilstm for wind speed prediction. *Energy*, *254*, 124250.

Cadenas, E., & Rivera, W (2007). Wind speed forecasting in the south coast of oaxaca, mexico. *Renewable Energy*, *32*(12), 2116–2128.

Cai, H, Jia, X, & Feng, J (2019). A combined filtering strategy for short term and long term wind speed prediction with improved accuracy. *Renewable Energy*, *136*, 1082–1090.

Cai, H., Jia, X., Feng, J (2020). Gaussian process regression for numerical wind speed prediction enhancement. *Renewable Energy*, *146*, 2112–2123. https://doi.org/10.1016/j.renene.2019.07.045

Cao, Q., Ewing, B. T., & Thompson, M. A (2012). Forecasting wind speed with recurrent neural networks. *European Journal of Operational Research*, *221*(1), 148–154.

Carneiro, T. C., Rocha, P. A. C., Carvalho, P. C. M (2022). Ridge regression ensemble of machine learning models applied to solar and wind forecasting in brazil and spain. *Applied Energy*, *314*, 118936. https://doi.org/10.1016/j.apenergy.2022.118936

Cassola, F., & Burlando, M (2012). Wind speed and wind energy forecast through kalman filtering of numerical weather prediction model output. *Applied Energy*, *99*, 154–166.

Castelletti, F., & Consonni, G (2020). Discovering causal structures in bayesian gaussian directed acyclic graph models. *Journal of the Royal Statistical Society Series A: Statistics in Society*, *183*(4), 1727–1745.

Chai T, & Draxler R R (2014). Root mean square error (rmse) or mean absolute error (mae). *Geoscientific Model Development Discussions*, *7*(1), 1525–1534.

Chandra, D. R., Kumari, M. S., & Sydulu, M (2013). A detailed literature review on wind forecasting. 2013 international conference on power, energy and control (icpec), 630–634.

Chang, W. Y (2014). A literature review of wind forecasting methods. *Journal of Power and Energy Engineering*, *2*(04), 161.

Chen, G, Li, L, & Zhang, Z (2020). Short-term wind speed forecasting with principle-subordinate predictor based on conv-lstm and improved BPNN. *IEEE Access*, *8*, 67955–67973. https://doi.org/10.1109/ACCESS.2020.2985660

Chen K, & Yu J (2014). Short-term wind speed prediction using an unscented kalman filter based state-space support vector regression approach. *Applied Energy*, *113*, 690–705.

Chen, C. L. P., & Liu, Z (2017). Broad learning system: an effective and efficient incremental learning system without the need for deep architecture. *IEEE Transactions on Neural Networks and Learning Systems*, *29*(1), 10–24.

Chen, G., Li, L., Zhang, Z (2020). Short-term wind speed forecasting with principle-subordinate predictor based on conv-lstm and improved BPNN. *IEEE Access*, *8*, 67955–67973.

Chen, J., Zeng, G. Q., Zhou, W (2018). Wind speed forecasting using nonlinear-learning ensemble of deep learning time series prediction and extremal optimization. *Energy Conversion and Management*, *165*, 681–695.

Chen, X., Li, Y., Zhang, Y (2021). A novel hybrid model based on an improved seagull optimization algorithm for short-term wind speed forecasting. *Processes*, *9*(2), 387.

Chen, Y., Dong, Z., Wang, Y (2021). Short-term wind speed predicting framework based on EEMD-GA-LSTM method under large scaled wind history. *Energy Conversion and Management*, *227*, 113559.

Chen, Y., He, Z., Shang, Z (2019). A novel combined model based on echo state network for multi-step ahead wind speed forecasting: a case study of NREL. *Energy Conversion and Management*, *179*, 13–29.

Chen, Z., Zhang, B., Du, C., Meng, W., & Meng, A (2024). A novel dynamic spatio-temporal graph convolutional network for wind speed interval prediction. *Energy*, *294*, 130930.

Cheng, L., Zang, H., Ding, T (2018). Ensemble recurrent neural network based probabilistic wind speed forecasting approach. *Energies*, *11*(8), 1958.

Cheng, R., & Shi, J (2026). Collaborative wind speed forecasting based on multi-strategy MVMD and restrictedcrossformer. *Energy*, *140163*.

Cheng, W. Y. Y., Liu, Y., Bourgeois, A. J (2017). Short-term wind forecast of a data assimilation/weather forecasting system with wind turbine anemometer measurement assimilation. *Renewable Energy*, *107*, 340–351.

Chitsazan, M. A., Fadali, M. S., & Trzynadlowski, A. M (2019). Wind speed and wind direction forecasting using echo state network with nonlinear functions. *Renewable Energy*, *131*, 879–889.

Chitsazan, M. A., Fadali, M. S., Nelson, A. K (2017). Wind speed forecasting using an echo state network with nonlinear output functions. *In 2017 American Control Conference (ACC)*, *5306*–5311. IEEE.

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

Cleveland, R B, Cleveland, W S, & McRae, J E (1990). STL: a seasonal-trend decomposition. j. off. *Stat*, *6*(1), 3–73.

Cortes, C., & Vapnik, V (1995). Support vector machine. *Machine Learning*, *20*(3), 273–297. https://doi.org/10.1007/BF00994018

Crafts, N (2021). Artificial intelligence as a general-purpose technology: an historical perspective. *Oxford Review of Economic Policy*, *37*(3). https://doi.org/10.1093/oxrep/grab012

Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., & Bharath, A. A (2018). Generative adversarial networks: an overview. *IEEE Signal Processing Magazine*, *35*(1), 53–65.

D. Snow (2020). MTSS-GAN: multivariate time series simulation generative adversarial networks. SSRN electronic journal. https://doi.org/10.2139/ssrn.3616557

D. Xiong, W. Fu, K. Wang (2021). A blended approach incorporating tvfemd, psr, nnct-based multi-model fusion and hierarchy-based merged optimization algorithm for multi-step wind speed prediction. *Energy Conversion and Management*, *230*, 113680.

De, Myttenaere A, Golden, B, & Le, Grand B (2016). Mean absolute percentage error for regression models. *Neurocomputing*, *192*, 38–48.

Deng, B., Wu, Y., Liu, S., & Zhang, J (2022). Wind speed forecasting for wind power production based on frequency-enhanced transformer. *In 2022 4th International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI)*, *151*–155. IEEE.

Dhiman, H S, & Deb D, & Guerrero J M (2019). Hybrid machine intelligent SVR variants for wind forecasting and ramp events. *Renewable and Sustainable Energy Reviews*, *108*, 369–379.

Dietterich, T. G (2000). Ensemble methods in machine learning. *In International Workshop on Multiple Classifier Systems*, *1*–15. Springer Berlin Heidelberg.

Ding, L., Bai, Y. L., Fan, M. H (2023). Serial-parallel dynamic echo state network: a hybrid dynamic model based on a chaotic coyote optimization algorithm for wind speed prediction. *Expert Systems with Applications*, *212*, 118789.

Ding, M., Zhou, H., Xie, H (2019). A gated recurrent unit neural networks based wind speed error correction model for short-term wind power forecasting. *Neurocomputing*, *365*, 54–61.

Ding, W., & Meng, F (2020). Point and interval forecasting for wind speed based on linear component extraction. *Applied Soft Computing*, *93*, 106350.

Done, J., Davis, C. A., & Weisman, M (2004). The next generation of nwp: explicit forecasts of convection using the weather research and forecasting (wrf) model. *Atmospheric Science Letters*, *5*(6), 110–117.

Dou, Q, Wei, L, & Magee, D R (2016). Real-time hyperbola recognition and fitting in GPR data. *IEEE Transactions on Geoscience and Remote Sensing*, *55*(1), 51–62.

Douak, F., Melgani, F., & Benoudjit, N (2013). Kernel ridge regression with active learning for wind speed prediction. *Applied Energy*, *103*,

328–340.

Duan, J., Zuo, H., Bai, Y (2021). Short-term wind speed forecasting using recurrent neural networks with error correction. *Energy*, *217*, 119397.

Dupré, A., Drobinski, P., Alonzo, B (2020). Sub-hourly forecasting of wind speed and wind energy. *Renewable Energy*, *145*, 2373–2379.

Durán, P., Meißner, C., Rutledge, K (2019). Meso-microscale coupling for wind resource assessment using averaged atmospheric stability conditions. meteorologische zeitschrift.

E. Cadenas, W. Rivera, R. Campos-Amezcua (2016). Wind speed prediction using a univariate ARIMA model and a multivariate NARX model. *Energies*, *9*(2), 109.

E. Safavieh, A. J. Ardakani, A. K. Laviani (2007). A new integrated approach for very short-term wind speed prediction using wavelet networks and pso. in proceedings of the international conference on power systems.

Eddy, S. R (1996). Hidden markov models. *Current Opinion in Structural Biology*, *6*(3), 361–365.

El-Fouly, T H M, & El-Saadany E F, & Salama M M A (2006). Grey predictor for wind energy conversion systems output power prediction. *IEEE Transactions on Power Systems*, *21*(3), 1450–1452.

Elman, J. L (1990). Finding structure in time. *Cognitive Science*, *14*(2), 179–211.

Elsaraiti, M., & Merabet, A (2021). A comparative analysis of the ARIMA and LSTM predictive models and their effectiveness for predicting wind speed. *Energies*, *14*(20), 6782.

Elsaraiti, M., & Merabet, A (2021). Application of long-short-term-memory recurrent neural networks to forecast wind speed. *Applied Sciences*, *11*(5), 2387.

Ember (2022). Global electricity review 2022.

Erdem, E., & Shi, J (2011). ARMA based approaches for forecasting the tuple of wind speed and direction. *Applied Energy*, *88*(4), 1405–1414.

Error, M A (2016). Mean absolute error. *Retrieved September 19*, *2016*.

F. Y. Wang, J. J. Zhang, X. Zheng (2016). Where does alphago go: from church-turing thesis to alphago thesis and beyond. *IEEE/CAA Journal of Automatica Sinica*, *3*(2), 113–120.

Fei S, & He Y (2015). Wind speed prediction using the hybrid model of wavelet decomposition and artificial bee colony algorithm-based relevance vector machine. *International Journal of Electrical Power & Energy Systems*, *73*, 625–631.

Feng, S., Lauvaux, T., Barkley, Z. R (2021). ACT-America: wrf-chem baseline simulations for north america, 2016–2019. ORNL DAAC.

Fidan M, Hocaoğlu F O, & Gerek Ö N (2012). Improved synthetic wind speed generation using modified mycielski approach. *International Journal of Energy Research*, *36*(13), 1226–1237.

Flores, P., Tapia, A., & Tapia, G (2005). Application of a control algorithm for wind speed prediction and active power generation. *Renewable Energy*, *30*(4), 523–536. https://doi.org/10.1016/j.renene.2004.07.045

Francois-Lavet, V., Henderson, P., Islam, R., Bellemare, M. G., & Pineau, J (2018). An introduction to deep reinforcement learning. *Foundations and Trends® in Machine Learning*, *11*(3–4), 219–354.

Fu, X, Feng, Z, & Yao, X (2023). A novel twin support vector regression model for wind speed time-series interval prediction. *Energies*, *16*(15), 5656.

Fu, W., Fu, Y., Li, B (2023). A compound framework incorporating improved outlier detection and correction, vmd, weight-based stacked generalization with enhanced DESMA for multi-step short-term wind speed forecasting. *Applied Energy*, *348*, 121587.

Fu, W., Wang, K., Tan, J (2020). A composite framework coupling multiple feature selection, compound prediction models and novel hybrid swarm optimizer-based synchronization optimization strategy for multi-step ahead short-term wind speed forecasting. *Energy Conversion and Management*, *205*, 112461.

G. Chen, B. Tang, X. Zeng (2022). Short-term wind speed forecasting based on long short-term memory and improved BP neural network. *International Journal of Electrical Power & Energy Systems*, *134*, 107365.

G. Tang, Y. Wu, C. Li (2020). A novel wind speed interval prediction based on error prediction method. *IEEE Transactions on Industrial Informatics*, *16*(11), 6806–6815.

G. Zenkner & S. Navarro-Martinez (2023). A flexible and lightweight deep learning weather forecasting model. *Applied Intelligence*, *53*(21), 24991–25002.

Gan, Z., Li, C., Zhou, J (2021). Temporal convolutional networks interval prediction model for wind speed forecasting. *Electric Power Systems Research*, *191*, 106865.

Gangwar, S., Bali, V., & Kumar, A (2020). Comparative analysis of wind speed forecasting using LSTM and SVM. *EAI Endorsed Transactions on Scalable Information Systems*, *7*(25), e1–e1.

Geng, D., Zhang, H., & Wu, H (2020). Short-term wind speed prediction based on principal component analysis and LSTM. *Applied Sciences*, *10*(13), 4416.

Geng, X., Xu, L., He, X (2021). Graph optimization neural network with spatio-temporal correlation learning for multi-node offshore wind speed forecasting. *Renewable Energy*, *180*, 1014–1025.

Ghorbani, M. A., Khatibi, R., FazeliFard, M. H., Naghipour, L., & Makarynskyy, O (2016). Short-term wind speed predictions with machine learning techniques. *Meteorology and Atmospheric Physics*, *128*, 57–72.

Global, Wind Energy Council (2022). Global wind report 2022.

Goh, H. H., Lee, S. W., Chua, Q. S (2016). Wind energy assessment considering wind speed correlation in malaysia. *Renewable and Sustainable Energy Reviews*, *54*, 1389–1400.

Gou, J., Yu, B., Maybank, S. J., & Tao, D (2021). Knowledge distillation: a survey. *International Journal of Computer Vision*, *129*, 1789–1819.

Gray, R. M., & Neuhoff, D. L (1998). Quantization. *IEEE Transactions on Information Theory*, *44*(6), 2325–2383.

Guo, Z, Zhao, J, & Zhang, W (2011). A corrected hybrid approach for wind speed prediction in hexi corridor of china. *Energy*, *36*(3), 1668–1679.

Guo, C., Zhang, X., & Iqbal, S (2024). Does oil price volatility and financial expenditures of the oil industry influence energy generation intensity? implications for clean energy acquisition. *Journal of Cleaner Production*, *434*, 139907. https://doi.org/10.1016/j.jclepro.2023.139907

Guo, Z., Wu, J., Lu, H (2011). A case study on a hybrid wind speed forecasting method using BP neural network. *Knowledge-Based Systems*, *24*(7), 1048–1056.

Guo, Z., Zhao, W., Lu, H., & Wang, J (2012). Multi-step forecasting for wind speed using a modified emd-based artificial neural network model. *Renewable Energy*, *37*(1), 241–249.

H. B. Azad, S. Mekhilef, & V. G. Ganapathy (2014). Long-term wind speed forecasting and general pattern recognition using neural networks. *IEEE Transactions on Sustainable Energy*, *5*(2), 546–553.

H. Chen, S. Wei, W. Yang (2023). Input wind speed forecasting for wind turbines based on spatio-temporal correlation. *Renewable Energy*, *216*, 119075.

H. Liu, C. Yu, H. Wu (2020). A new hybrid ensemble deep reinforcement learning model for wind speed short term forecasting. *Energy*, *202*, 117794.

H. Malik & A. K. Yadav (2021). A novel hybrid approach based on relief algorithm and fuzzy reinforcement learning approach for predicting wind speed. *Sustainable Energy Technologies and Assessments*, *43*, 100920.

H. Zhou, S. Zhang, J. Peng (2021). Informer: beyond efficient transformer for long sequence time-series forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, *35*(12), 11106–11115.

Han, Y, Mi, L, & Shen, L (2022). A short-term wind speed prediction method utilizing novel hybrid deep learning algorithms to correct numerical weather forecasting. *Applied Energy*, *312*, 118777.

Han, Q., Meng, F., Hu, T (2017). Non-parametric hybrid models for wind speed forecasting. *Energy Conversion and Management*, *148*, 554–568.

Hannan, E J, & Kavalieris, L (1984). A method for autoregressive-moving average estimation. *Biometrika*, *71*(2), 273–280.

Haque, A. U., Mandal, P., Kaye, M. E., Meng, J., Chang, L., & Senjyu, T (2012). A new strategy for predicting short-term wind speed using soft computing models. *Renewable and Sustainable Energy Reviews*, *16*(7),

4563–4573.

Hinton, G. E., Osindero, S., & Teh, Y. W (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, *18*(7), 1527–1554.

Hocaoğlu F O, & Fidan M, Gerek Ö N (2009). Mycielski approach for wind speed prediction. *Energy Conversion and Management*, *50*(6), 1436–1443.

Hochreiter, S., & Schmidhuber, J (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780.

Hoerl, A. E., & Kennard, R. W (1970). Ridge regression: biased estimation for nonorthogonal problems. *Technometrics*, *12*(1), 55–67. https://doi.org/10.1080/00401706.1970.10488634

Hoolohan, V., Tomlin, A. S., & Cockerill, T (2018). Improved near surface wind speed predictions using gaussian process regression combined with numerical weather predictions and observed meteorological data. *Renewable Energy*, *126*, 1043–1054. https://doi.org/10.1016/j.renene.2018.04.045

Hu, J, & Wang J, & Ma K (2015). A hybrid technique for short-term wind speed prediction. *Energy*, *81*, 563–574.

Hu, J, & Wang J, & Zeng G (2013). A hybrid forecasting approach applied to wind speed time series. *Renewable Energy*, *60*, 185–194.

Hu, Q, Su, P, & Yu, D (2014). Pattern-based wind speed prediction based on generalized principal component analysis. *IEEE Transactions on Sustainable Energy*, *5*(3), 866–874.

Hu, H., Wang, L., & Tao, R (2021). Wind speed forecasting based on variational mode decomposition and improved echo state network. *Renewable Energy*, *164*, 729–751.

Hu, J., & Wang, J (2015). Short-term wind speed prediction using empirical wavelet transform and gaussian process regression. *Energy*, *93*, 1456–1466.

Hu, J., Wang, J., & Xiao, L (2017). A hybrid approach based on the gaussian process with t-observation model for short-term wind speed forecasts. *Renewable Energy*, *114*, 670–685.

Hu, Y. L., & Chen, L (2018). A nonlinear hybrid wind speed forecasting model using LSTM network, hysteretic ELM and differential evolution algorithm. *Energy Conversion and Management*, *173*, 123–142.

Hua, L., Zhang, C., Peng, T (2022). Integrated framework of extreme learning machine (elm) based on improved atom search optimization for short-term wind speed prediction. *Energy Conversion and Management*, *252*, 115102.

Huang, C, Karimi, H R, & Mei, P (2023). Evolving long short-term memory neural network for wind speed forecasting. *Information Sciences*, *632*, 390–410.

Huang, C. Y., Liu, Y. W., Tzeng, W. C (2011). Short term wind speed predictions by using the grey prediction model based forecast method. *In 2011 IEEE Green Technologies Conference (IEEE-Green)*, *1*–5. IEEE.

Huang, G. B., Zhu, Q. Y., & Siew, C. K (2006). Extreme learning machine: theory and applications. *Neurocomputing*, *70*(1–3), 489–501. https://doi.org/10.1016/j.neucom.2005.12.126

Hunter, J. S (1986). The exponentially weighted moving average. *Journal of Quality Technology*, *18*(4), 203–210.

I. Tyass, T. Khalili, R. Mohamed (2023). Wind speed prediction based on statistical and deep learning models. *International Journal of Renewable Energy Development*, *12*(2), 288.

Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., & Keutzer, K (2016). SqueezeNet: alexnet-level accuracy with 50x fewer parameters and <0.5 MB model size. arxiv preprint arxiv:1602.07360.

Ibrahim, M, Alsheikh, A, & Al-Hindawi, Q (2020). Short-time wind speed forecast using artificial learning-based algorithms. *Computational Intelligence and Neuroscience*, *2020*.

Iversen, E B, Morales, J M, & Møller J K, & Madsen H (2016). Short-term probabilistic forecasting of wind speed using stochastic differential equations. *International Journal of Forecasting*, *32*(3), 981–990.

J. Powles & H. Hodson (2017). Google deepmind and healthcare in an age of algorithms. *Health and Technology*, *7*(4), 351–367.

J. Wang, W. Zhang, J. Wang (2014). A novel hybrid approach for wind speed prediction. *Information Sciences*, *273*, 304–318.

J. Wang, Y. Wang, Z. Li (2020). A combined framework based on data preprocessing, neural networks and multi-tracker optimizer for wind speed prediction. *Sustainable Energy Technologies and Assessments*, *40*, 100757.

Jaeger, H (2002). Adaptive nonlinear system identification with echo state networks. *Advances in Neural Information Processing Systems*, *15*.

Jaseena K U, & Kovoor B C (2021). Decomposition-based hybrid wind speed forecasting model using deep bidirectional LSTM networks. *Energy Conversion and Management*, *234*, 113944.

Jenkins, G. M (1954). Tests of hypotheses in the linear autoregressive model: 1. null hypothesis distributions in the yule scheme. *Biometrika*, *41*(3/4), 405–419.

Ji, L., Fu, C., Ju, Z (2022). Short-term canyon wind speed prediction based on cnn–gru transfer learning. *Atmosphere*, *13*(5), 813.

Jiang Y, & Huang G (2017). Short-term wind speed prediction: hybrid of ensemble empirical mode decomposition, feature selection and error correction. *Energy Conversion and Management*, *144*, 340–350.

Jiang, Y, Huang, G, & Peng, X (2018). A novel wind speed prediction method: hybrid of correlation-aided dwt, LSSVM and GARCH. *Journal of Wind Engineering and Industrial Aerodynamics*, *174*, 28–38.

Jiang, Y, Zhao, N, & Peng, L (2019). A new hybrid framework for probabilistic wind speed prediction using deep feature selection and multi-error modification. *Energy Conversion and Management*, *199*, 111981.

Jiang, Y., Song, Z., & Kusiak, A (2013). Very short-term wind speed forecasting with bayesian structural break model. *Renewable Energy*, *50*, 637–647.

K. Arulkumaran, A. Cully, & J. Togelius (2019). Alphastar: an evolutionary computation perspective. *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, *314*–315.

K. Arulkumaran, M. P. Deisenroth, M. Brundage (2017). Deep reinforcement learning: a brief survey. *IEEE Signal Processing Magazine*, *34*(6), 26–38.

Kalman, R. E (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, *82*(1), 35–45.

Kang, A., Tan, Q., Yuan, X (2017). Short-term wind speed prediction using EEMD-LSSVM model. *Advances in Meteorology*, *2017*.

Kani, S. A. P., & Ardehali, M. M (2011). Very short-term wind speed prediction: a new artificial neural network–markov chain model. *Energy Conversion and Management*, *52*(1), 738–745.

Kani, S. A. P., & Riahy, G. H (2008). A new ann-based methodology for very short-term wind speed prediction using markov chain approach. *In 2008 IEEE Canada Electric Power Conference*, *1*–6. IEEE.

Kaplan O, & Temiz M (2017). A novel method based on weibull distribution for short-term wind speed prediction. *International Journal of Hydrogen Energy*, *42*(28), 17793–17800.

Karakuş, O., Kuruoğlu, E. E., & Altınkaya, M. A (2017). One-day ahead wind speed/power prediction based on polynomial autoregressive model. *IET Renewable Power Generation*, *11*(11), 1430–1439.

Kavasseri, R. G., & Seetharaman, K (2009). Day-ahead wind speed forecasting using f-arima models. *Renewable Energy*, *34*(5), 1388–1393.

Khodayar M, & Wang J (2018). Spatio-temporal graph deep neural network for short-term wind speed forecasting. *IEEE Transactions on Sustainable Energy*, *10*(2), 670–681. https://doi.org/10.1109/TSTE.2018.2844102

Kiplangat, D. C., Asokan, K., & Kumar, K. S (2016). Improved week-ahead predictions of wind speed using simple linear models with wavelet decomposition. *Renewable Energy*, *93*, 38–44.

Koller, D., & Friedman, N (2009). Probabilistic graphical models: principles and techniques. MIT press.

Kong, X, Liu, X, & Shi, R (2015). Wind speed prediction using reduced support vector machines with feature selection. *Neurocomputing*, *169*, 449–456.

Kosana, V., Teeparthi, K., & Madasthu, S (2022). A novel and hybrid framework based on generative adversarial network and temporal convolutional approach for wind speed prediction. *Sustainable Energy Technologies and Assessments*, *53*, 102467.

Kou, P., Wang, C., Liang, D (2020). Deep learning approach for wind speed forecasts at turbine locations in a wind farm. *IET Renewable Power Generation*, *14*(13), 2416–2428.

Krizhevsky, A., Sutskever, I., & Hinton, G. E (2012). Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, *25*.

L. Wang & Y. He (2022). M2STAN: multi-modal multi-task spatiotemporal attention network for multi-location ultra-short-term wind power multi-step predictions. *Applied Energy*, *324*, 119672.

L. Wang, J. Li, A. Hu (2020). Deterministic and probabilistic multi-step forecasting for short-term wind speed based on secondary decomposition and a deep learning method. *Energy Conversion and Management*, *220*, 113098.

L. Zhu & C. Lian (2019). Wind speed forecasting based on a hybrid EMD-BLS method. *In 2019 Chinese Automation Congress (CAC)*, *2191*–2195. IEEE.

L. Zhu, C. Lian, Z. Zeng (2020). A broad learning system with ensemble and classification methods for multi-step-ahead wind speed prediction. *Cognitive Computation*, *12*, 654–666.

Lawan, S. M., Abidin, W., Chai, W. Y., Baharun, A., & Masri, T (2014). Different models of wind speed prediction; a comprehensive review. *International Journal of Scientific & Engineering Research*, *5*(1), 1760–1768.

Lea, C., Vidal, R., Reiter, A., & Hager, G. D (2016). Temporal convolutional networks: a unified approach to action segmentation. *In Computer Vision–ECCV 2016 Workshops*, *47*–54. Springer International Publishing.

Lei, M., Shiyan, L., Chuanwen, J., Hongling, L., & Yan, Z (2009). A review on the forecasting of wind speed and generated power. *Renewable and Sustainable Energy Reviews*, *13*(4), 915–920.

Leung, H., & Haykin, S (1991). The complex backpropagation algorithm. *IEEE Transactions on Signal Processing*, *39*(9), 2101–2104.

Li, J, Song, Z, & Wang, X (2022). A novel offshore wind farm typhoon wind speed prediction model based on pso–bi-lstm improved by VMD. *Energy*, *251*, 123848.

Li, W, Jia, X, Li, X, & Zhang Y, & Wang H (2021). A markov model for short term wind speed prediction by integrating the wind acceleration information. *Renewable Energy*, *164*, 242–253.

Li, Z L, Xia, J, & Liu, A (2019). States prediction for solar power and wind speed using bba-svm. *IET Renewable Power Generation*, *13*(7), 1115–1122.

Li, C., Zhu, Z., Yang, H (2019). An innovative hybrid system for wind speed forecasting based on fuzzy preprocessing scheme and multi-objective optimization. *Energy*, *174*, 1219–1237.

Li, D., Jiang, F., Chen, M (2022). Multi-step-ahead wind speed forecasting based on a hybrid decomposition method and temporal convolutional networks. *Energy*, *238*, 121981.

Li, F., Ren, G., & Lee, J (2019). Multi-step wind speed prediction based on turbulence intensity and hybrid deep neural networks. *Energy Conversion and Management*, *186*, 306–322.

Li, M., Zhang, Z., Ji, T (2020). Ultra-short term wind speed prediction using mathematical morphology decomposition and long short-term memory. *CSEE Journal of Power and Energy Systems*, *6*(4), 890–900.

Li, N., Dong, J., Liu, L., & Li, H (2023). A novel EMD and causal convolutional network integrated with transformer for ultra short-term wind power forecasting. *International Journal of Electrical Power & Energy Systems*, *154*, 109470.

Li, Q., Wang, G., Wu, X., Gao, Z., & Dan, B (2024). Arctic short-term wind speed forecasting based on CNN-LSTM model with CEEMDAN. *Energy*, *299*, 131448.

Li, R., & Jin, Y (2018). A wind speed interval prediction system based on multi-objective optimization for machine learning method. *Applied Energy*, *228*, 2207–2220. https://doi.org/10.1016/j.apenergy.2018.07.070

Li, Y., Wu, H., & Liu, H (2018). Multi-step wind speed forecasting using EWT decomposition, LSTM principal computing, RELM subordinate computing and IEWT reconstruction. *Energy Conversion and Management*, *167*, 203–219.

Liang, T, Zhao, Q, & Lv, Q (2021). A novel wind speed prediction strategy based on bi-lstm, MOOFADA and transfer learning for centralized control centers. *Energy*, *230*, 120904.

Lin, S., Wang, J., Xu, X., & Chen, Y (2023). SWSA transformer: a forecasting method of ultra-short-term wind speed from an offshore wind farm using global attention mechanism. *Journal of Renewable and Sustainable Energy*, *15*(4).

Lin, S., Wang, S., Xu, X., Li, R., & Shi, P (2024). GAOformer: an adaptive spatiotemporal feature fusion transformer utilizing GAT and optimizable graph matrixes for offshore wind speed prediction. *Energy*, *292*, 130404.

Ling-ling, L., Li, J. H., He, P. J (2011). The use of wavelet theory and ARMA model in wind speed prediction. *In 2011 1st International Conference on Electric Power Equipment-Switching Technology*, *395*–398. IEEE.

Liu, L, Ji, T, & Li, M (2018). Short-term local prediction of wind speed and wind power based on singular spectrum analysis and locality-sensitive hashing. *Journal of Modern Power Systems and Clean Energy*, *6*(2), 317–329.

Liu, M, Cao, Z, & Zhang, J (2020). Short-term wind speed forecasting based on the jaya-svm model. *International Journal of Electrical Power & Energy Systems*, *121*, 106056.

Liu, D., Wang, J., & Wang, H (2015). Short-term wind speed forecasting based on spectral clustering and optimised echo state networks. *Renewable Energy*, *78*, 599–608.

Liu, G., Wang, C., Qin, H (2022). A novel hybrid machine learning model for wind speed probabilistic forecasting. *Energies*, *15*(19), 6942.

Liu, H., Chen, C., Tian, H., & Li, Y (2012). A hybrid model for wind speed prediction using empirical mode decomposition and artificial neural networks. *Renewable Energy*, *48*, 545–556.

Liu, H., Erdem, E., & Shi, J (2011). Comprehensive evaluation of arma–garch (-m) approaches for modeling the mean and volatility of wind speed. *Applied Energy*, *88*(3), 724–732.

Liu, H., Liu, J., Hu, T., & Ma, H (2025). Spatio-Temporal probabilistic forecasting of wind speed using transformer-based diffusion models. IEEE transactions on sustainable energy.

Liu, H., Mi, X., & Li, Y (2018). An experimental investigation of three new hybrid wind speed forecasting models using multi-decomposing strategy and ELM algorithm. *Renewable Energy*, *123*, 694–705.

Liu, H., Mi, X., & Li, Y (2018). Smart deep learning based wind speed prediction model using wavelet packet decomposition, convolutional neural network and convolutional long short term memory network. *Energy Conversion and Management*, *166*, 120–131.

Liu, H., Mi, X., & Li, Y (2018). Smart multi-step deep learning model for wind speed forecasting based on variational mode decomposition, singular spectrum analysis, LSTM network and ELM. *Energy Conversion and Management*, *159*, 54–64.

Liu, H., Mi, X., & Li, Y (2018). Wind speed forecasting method based on deep learning strategy using empirical wavelet transform, long short term memory neural network and elman neural network. *Energy Conversion and Management*, *156*, 498–514.

Liu, H., Mi, X., Li, Y (2019). Smart wind speed deep learning based multi-step forecasting model using singular spectrum analysis, convolutional gated recurrent unit network and support vector regression. *Renewable Energy*, *143*, 842–854.

Liu, H., Shi, J., & Erdem, E (2010). Prediction of wind speed time series using modified taylor kriging method. *Energy*, *35*(12), 4870–4879.

Liu, H., Tian, H., & Li, Y (2012). Comparison of two new ARIMA-ANN and arima-kalman hybrid methods for wind speed prediction. *Applied Energy*, *98*, 415–424.

Liu, H., Tian, H., & Li, Y (2015). An emd-recursive ARIMA method to predict wind speed for railway strong wind warning system. *Journal of Wind Engineering and Industrial Aerodynamics*, *141*, 27–38.

Liu, H., Tian, H., Chen, C., & Li, Y (2013). An experimental investigation of two wavelet-mlp hybrid frameworks for wind speed prediction using GA and PSO optimization. *International Journal of Electrical Power & Energy Systems*, *52*, 161–173.

Liu, H., Tian, H., Liang, X., & Li, Y (2015). New wind speed forecasting approaches using fast ensemble empirical model decomposition, genetic algorithm, mind evolutionary algorithm and artificial neural networks. *Renewable Energy*, *83*, 1066–1075.

Liu, H., Tian, H., Liang, X., & Li, Y (2015). Wind speed forecasting approach using secondary decomposition algorithm and elman neural networks. *Applied Energy*, *157*, 183–194.

Liu, H., Tian, H., Pan, D., & Li, Y (2013). Forecasting models for wind speed using wavelet, wavelet packet, time series and artificial neural networks. *Applied Energy*, *107*, 191–208.

Liu, H., Wu, H., & Li, Y (2018). Smart wind speed forecasting using EWT decomposition, GWO evolutionary optimization, RELM learning and IEWT reconstruction. *Energy Conversion and Management*, *161*, 266-283.

Liu, M. D., Ding, L., & Bai, Y. L (2021). Application of hybrid model based on empirical mode decomposition, novel recurrent neural networks and the ARIMA to wind speed prediction. *Energy Conversion and Management*, *233*, 113917.

Liu, X., Lin, Z., & Feng, Z (2021). Short-term offshore wind speed forecast by seasonal ARIMA-A comparison against GRU and LSTM. *Energy*, *227*, 120492.

Liu, X., Yu, J., Gong, L., Liu, M., & Xiang, X (2024). A gcn-based adaptive generative adversarial network model for short-term wind speed scenario prediction. *Energy*, *294*, 130931.

Liu, Y., Qin, H., Zhang, Z (2020). Probabilistic spatiotemporal wind speed forecasting based on a variational bayesian deep learning model. *Applied Energy*, *260*, 114259.

Lo, J. C. F., Yang, Z. L., & Pielke Sr, R. A (2008). Assessment of three dynamical climate downscaling methods using the weather research and forecasting (wrf) model. *Journal of Geophysical Research: Atmospheres*, *113*(D9).

Lu, P., Ye, L., Zhao, Y (2021). Review of meta-heuristic algorithms for wind power prediction: methodologies, applications and challenges. *Applied Energy*, *301*, 117446.

Lv, S., Wang, L., & Wang, S (2023). A hybrid neural network model for short-term wind speed forecasting. *Energies*, *16*(4), 1841.

Lydia, M., Kumar, S. S., Selvakumar, A. I (2016). Linear and non-linear autoregressive models for short-term wind speed forecasting. *Energy Conversion and Management*, *112*, 115–124.

Lyu, S (2005). Mercer kernels for object recognition with local features. *In 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, *223*–229. IEEE.

M. A. Hossain, E. Gray, J. Lu (2023). Optimized forecasting model to improve the accuracy of very short-term wind power prediction. IEEE transactions on industrial informatics.

M. Afrasiabi, M. Mohammadi, M. Rastegar (2019). Probabilistic deep neural network price forecasting based on residential load and wind speed predictions. *IET Renewable Power Generation*, *13*(11), 1840–1848.

M. Ahmed, R. Seraj, & S. M. S. Islam (2020). The k-means algorithm: a comprehensive survey and performance evaluation. *Electronics*, *9*(8), 1295.

M. Jamil & M. Zeeshan (2019). A comparative analysis of ANN and chaotic approach-based wind speed prediction in india. *Neural Computing and Applications*, *31*, 6807–6819.

M. Jawad, S. M. Ali, B. Khan (2018). Genetic algorithm-based non-linear auto-regressive with exogenous inputs neural network short-term and medium-term uncertainty modelling and prediction for electrical load and wind speed. *The Journal of Engineering*, *2018*(8), 721–729.

M. Madhiarasan (2020). Accurate prediction of different forecast horizons wind speed using a recursive radial basis function neural network. *Protection and Control of Modern Power Systems*, *5*(1), 1–9.

M. Santhosh, C. Venkaiah, & D. M. V. Kumar (2018). Ensemble empirical mode decomposition based adaptive wavelet neural network method for wind speed prediction. *Energy Conversion and Management*, *168*, 482–493.

M. Sibtain, H. Bashir, M. Nawaz (2022). A multivariate ultra-short-term wind speed forecasting model by employing multistage signal decomposition approaches and a deep learning network. *Energy Conversion and Management*, *263*, 115703.

Ma, C., Zhang, C., Yao, J., Zhang, X., Nazir, M. S., & Peng, T (2025). Enhancement of wind speed forecasting using optimized decomposition technique, entropy-based reconstruction, and evolutionary patchtst. *Energy Conversion and Management*, *333*, 119819.

Maatallah, O. A., Achuthan, A., Janoyan, K., & Marzocca, P (2015). Recursive wind speed forecasting based on hammerstein auto-regressive model. *Applied Energy*, *145*, 191–197.

Maybeck, P S (1990). The kalman filter: an introduction to concepts. *In Autonomous Robot Vehicles*, *194*, –204. Springer New York.

Meng, A., Ge, J., Yin, H (2016). Wind speed forecasting based on wavelet packet decomposition and artificial neural networks trained by crisscross optimization algorithm. *Energy Conversion and Management*, *114*, 75–88.

Meng, A., Xian, Z., Yin, H (2023). A novel network training approach for solving sample imbalance problem in wind power prediction. *Energy Conversion and Management*, *283*, 116935.

Meyer, G P (2021). An alternative probabilistic interpretation of the huber loss. *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, *5261*–5269.

Mi, X, & Liu H, & Li Y (2019). Wind speed prediction model using singular spectrum analysis, empirical mode decomposition and convolutional support vector machine. *Energy Conversion and Management*, *180*, 196–205.

Mi, L., Han, Y., Long, L., Chen, H., & Cai, C. S (2025). A physics-informed temporal convolutional network-temporal fusion transformer hybrid model for probabilistic wind speed predictions with quantile regression. *Energy*, *326*, 136302.

Mi, X., & Zhao, S (2020). Wind speed prediction based on singular spectrum analysis and neural network structural learning. *Energy Conversion and Management*, *216*, 112956.

Milliez, M., & Carissimo, B (2007). Numerical simulations of pollutant dispersion in an idealized urban area, for different meteorological conditions. *Boundary-Layer Meteorology*, *122*(2), 321–342.

Miranda M S, & Dunn R W (2006). One-hour-ahead wind speed prediction using a bayesian methodology. *In 2006 IEEE Power Engineering Society General Meeting*, *6*. IEEE.

Mohandes, M A, Halawani, T O, & Rehman, S (2004). Support vector machines for wind speed prediction. *Renewable Energy*, *29*(6), 939–947.

Moharm, K., Eltahan, M., & Elsaadany, E (2020). Wind speed forecast using LSTM and bi-lstm algorithms over gabal el-zayt wind farm. *In 2020 International Conference on Smart Grids and Energy Systems (SGES)*, 922–927. IEEE.

Møller, J K, & Zugno M, & Madsen H (2016). Probabilistic forecasts of wind power generation by stochastic differential equation models. *Journal of Forecasting*, *35*(3), 189–205.

Monteiro, C., Bessa, R., Miranda, V., Botterud, A., Wang, J., & Conzelmann, G (2009). Wind power forecasting: state-of-the-art 2009. argonne national lab. (anl).

Moody, J (1988). Fast learning in multi-resolution hierarchies. *Advances in Neural Information Processing Systems*, *1*.

Müller, M (2007). Dynamic time warping. *Information Retrieval for Music and Motion*, *69*–84.

Murtagh, F., & Contreras, P (2012). Algorithms for hierarchical clustering: an overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *2*(1), 86–97.

Naik, J., Bisoi, R., & Dash, P. K (2018). Prediction interval forecasting of wind speed and wind power using modes decomposition based low rank multi-kernel ridge regression. *Renewable Energy*, *129*, 357–383.

Naik, J., Dash, P. K., & Dhar, S (2019). A multi-objective wind speed and wind power prediction interval forecasting using variational modes decomposition based multi-kernel robust ridge regression. *Renewable Energy*, *136*, 701–731.

Naik, J., Satapathy, P., & Dash, P. K (2018). Short-term wind speed and

wind power prediction using hybrid empirical mode decomposition and kernel ridge regression. *Applied Soft Computing*, *70*, 1167–1188.

Nascimento, E. G. S., de Melo, T. A. C., & Moreira, D. M (2023). A transformer-based deep neural network with wavelet transform for forecasting wind speed and wind energy. *Energy*, *278*, 127678.

Natarajan Y J, & Subramaniam Nachimuthu D (2020). New SVM kernel soft computing models for wind speed prediction in renewable energy applications. *Soft Computing*, *24*, 11441–11458.

Nazir, S., Nawaz, M., Adnan, A (2019). Big data features, applications, and analytics in cardiology—a systematic literature review. *IEEE Access*, *7*, 143742–143771.

Nie, Y., Bo, H., Zhang, W (2020). Research on hybrid wind speed prediction system based on artificial intelligence and double prediction scheme. *Complexity*, *2020*, 1–22.

Nie, Y., Liang, N., & Wang, J (2021). Ultra-short-term wind-speed bi-forecasting system via artificial intelligence and a double-forecasting scheme. *Applied Energy*, *301*, 117452.

Niu, Z., Zhong, G., & Yu, H (2021). A review on the attention mechanism of deep learning. *Neurocomputing*, *452*, 48–62.

Omolara, A. E., Alabdulatif, A., Abiodun, O. I., Alawida, M., Alabdulatif, A., & Arshad, H (2022). The internet of things security: a survey encompassing unexplored areas and new insights. *Computers & Security*, *112*, 102494.

Optis, M., Kumler, A., Brodie, J (2021). Quantifying sensitivity in numerical weather prediction-modeled offshore wind speeds through an ensemble modeling approach. *Wind Energy*, *24*(9), 957–973.

Ozay C, & Celiktas M S (2016). Statistical analysis of wind speed using two-parameter weibull distribution in alaçatı region. *Energy Conversion and Management*, *121*, 49–54.

P. Jeha, M. Bohlke-Schneider, P. Mercado (2021). PSA-GAN: progressive self attention gans for synthetic time series. international conference on learning representations.

Pan, X., Wang, L., Wang, Z (2022). Short-term wind speed forecasting based on spatial-temporal graph transformer networks. *Energy*, *253*, 124095.

Pei, S., Qin, H., Zhang, Z (2019). Wind speed prediction method based on empirical wavelet transform and new cell update long short-term memory network. *Energy Conversion and Management*, *196*, 779–792.

Petković D, Shamshirband, S, & Anuar, N B (2014). An appraisal of wind speed distribution prediction by soft computing methodologies: a comparative study. *Energy Conversion and Management*, *84*, 133–139.

Q. Hu, R. Zhang, & Y. Zhou (2016). Transfer learning for short-term wind speed prediction with deep neural networks. *Renewable Energy*, *85*, 83–95.

Qu, Z., Mao, W., Zhang, K (2019). Multi-step wind speed forecasting based on a hybrid decomposition technique and an improved back-propagation neural network. *Renewable Energy*, *133*, 919–929.

Qu, Z., Zhang, K., Mao, W (2017). Research and application of ensemble forecasting based on a novel multi-objective optimization algorithm for wind-speed forecasting. *Energy Conversion and Management*, *154*, 440–454.

R. Yang, H. Liu, N. Nikitas (2022). Short-term wind speed forecasting using deep reinforcement learning with improved multiple error correction approach. *Energy*, *239*, 122128.

R. Ye, X. Li, Y. Ye (2022). DynamicNet: a time-variant ODE network for multi-step wind speed prediction. *Neural Networks*, *152*, 118–139.

Rasmussen, C. E., & Nickisch, H (2010). Gaussian processes for machine learning (gpml) toolbox. *The Journal of Machine Learning Research*, *11*, 3011–3015.

Ren, Y, & Suganthan P N, & Srikanth N (2014). A novel empirical mode decomposition with support vector regression for wind speed forecasting. *IEEE Transactions on Neural Networks and Learning Systems*, *27*(8), 1793–1798.

Ren, C., An, N., Wang, J (2014). Optimal parameters selection for BP neural network based on particle swarm optimization: a case study of wind speed forecasting. *Knowledge-Based Systems*, *56*, 226–239.

Riahy G H, & Abedi M (2008). Short term wind speed forecasting for wind turbine applications using linear prediction method. *Renewable Energy*, *33*(1), 35–41.

Rosenblatt, F (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, *65*(6), 386.

Ruiz-Aguilar, J. J., Turias, I., González-Enrique, J., & Urda, D (2021). A permutation entropy-based emd–ann forecasting ensemble approach for wind speed prediction. *Neural Computing and Applications*, *33*(7), 2369–2391.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J (1986). Learning representations by back-propagating errors. *Nature*, *323*(6088), 533–536.

S. Hu, X. Xu, M. Li (2024). Incremental forecaster using c–c algorithm to phase space reconstruction and broad learning network for short-term wind speed prediction. *Engineering Applications of Artificial Intelligence*, *128*, 107461.

S. Uranchimeg, H. H. Kwon, & K. W. Kim (2017). Assessing forecast accuracy of the UM numerical weather model for the hydrological application. *Proceedings of the Korea Water Resources Association Conference*, *233*.

S. Wang, J. Wang, H. Lu (2021). A novel combined model for wind speed prediction–combination of linear model, shallow neural networks, and deep learning approaches. *Energy*, *234*, 121275.

S. X. Lv & L. Wang (2022). Deep learning combined wind speed forecasting with hybrid time series decomposition and multi-objective parameter optimization. *Applied Energy*, *311*, 118674.

Saeed, A., Li, C., Danish, M (2020). Hybrid bidirectional LSTM model for short-term wind speed interval prediction. *IEEE Access*, *8*, 182283–182294.

Saeed, A., Li, C., Gan, Z (2022). A simple approach for short-term wind speed interval prediction based on independently recurrent neural networks and error probability distribution. *Energy*, *238*, 122012.

Saeed, A., Li, C., Rubaiee, S., Danish, M., & Anwar, S (2025). Enhanced wind speed forecasting for sustainable power systems: a deep learning framework unifying deterministic predictions and uncertainty quantification. *Energy*, *137979*.

Sahin, A. D., & Sen, Z (2001). First-order markov chain approach to wind speed modelling. *Journal of Wind Engineering and Industrial Aerodynamics*, *89*(3–4), 263–269.

Salcedo-Sanz, S., Perez-Bellido, A. M., Ortiz-García, E. G (2009). Accurate short-term wind speed prediction by exploiting diversity in input data using banks of artificial neural networks. *Neurocomputing*, *72*(4–6), 1336–1341.

Salcedo-Sanz, S., Pérez-Bellido, Á. M., Ortiz-García, E. G., Portilla-Figueras, A., Prieto, L., & Paredes, D (2009). Hybridizing the fifth generation mesoscale model with artificial neural networks for short-term wind speed prediction. *Renewable Energy*, *34*(6), 1451–1457.

Samet, H., & Marzbani, F (2014). Quantizing the deterministic nonlinearity in wind speed time series. *Renewable and Sustainable Energy Reviews*, *39*, 1143–1154.

Santamaría-Bonfil, G, & Reyes-Ballesteros A, & Gershenson C (2016). Wind speed forecasting for wind farms: a method based on support vector regression. *Renewable Energy*, *85*, 790–809.

Sanz, S. S., Perez-Bellido, A., Ortiz-Garcia, E., & Salcedo-Sanz, S (2008). Short-term wind speed prediction by hybridizing global and mesoscale forecasting models with artificial neural networks. *In 2008 Eighth International Conference on Hybrid Intelligent Systems*, *608*–612. IEEE.

Sarker, I. H (2021). Machine learning: algorithms, real-world applications and research directions. *SN Computer Science*, *2*(3), 160.

Schaback, R., & Wendland, H (2006). Kernel techniques: from machine learning to meshless methods. *Acta Numerica*, *15*, 543–639.

Schlink U, & Tetzlaff G (1998). Wind speed forecasting from 1 to 30 minutes. *Theoretical and Applied Climatology*, *60*, 191–198.

Schölkopf, B, & Herbrich R, & Smola A J (2001). A generalized representer theorem. *In International Conference on Computational Learning Theory*, *416*–426. Springer Berlin Heidelberg.

Shahid, F., Zameer, A., & Iqbal, M. J (2021). Intelligent forecast engine for short-term wind speed prediction based on stacked long short-term

memory. *Neural Computing and Applications*, *33*(20), 13767–13783.

Shamshad, A., Bawadi, M. A., Hussin, W. M. A. W., Majid, T. A., & Sanusi, S. A. M (2005). First and second order markov chain models for synthetic generation of wind speed time series. *Energy*, *30*(5), 693–708.

Shao, B., Song, D., Bian, G (2021). Wind speed forecast based on the LSTM neural network optimized by the firework algorithm. *Advances in Materials Science and Engineering*, *2021*, 1–13.

Shapsough, S., Aloul, F., & Zualkernan, I. A (2018). Securing low-resource edge devices for iot systems. *In 2018 International Symposium in Sensing and Instrumentation in IoT Era (ISSI)*, *1*–4. IEEE.

Sheela, K. G., & Deepa, S. N (2013). Neural network based hybrid computing model for wind speed prediction. *Neurocomputing*, *122*, 425–429.

Shen, Z., Fan, X., Zhang, L (2022). Wind speed prediction of unmanned sailboat based on CNN and LSTM hybrid neural network. *Ocean Engineering*, *254*, 111352.

Shi, J., Guo, J., & Zheng, S (2012). Evaluation of hybrid forecasting approaches for wind speed and power generation time series. *Renewable and Sustainable Energy Reviews*, *16*(5), 3471–3480.

Shrivastava, N A, & Lohia K, & Panigrahi B K (2016). A multiobjective framework for wind speed prediction interval forecasts. *Renewable Energy*, *87*, 903–910.

Sim, S. K., Maass, P., & Lind, P. G (2018). Wind speed modeling by nested ARIMA processes. *Energies*, *12*(1), 69.

Simon, D (2001). Kalman filtering. *Embedded Systems Programming*, *14*(6), 72–79.

Singh, S. N., & Mohapatra, A (2019). Repeated wavelet transform based ARIMA model for very short-term wind speed forecasting. *Renewable Energy*, *136*, 758–768.

Sinha, D., & El-Sharkawy, M (2019). Thin mobilenet: an enhanced mobilenet architecture. *In 2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, *0280*–0285. IEEE.

Soman, S. S., Zareipour, H., Malik, O (2010). A review of wind power and wind speed forecasting methods with different time horizons. *In North American Power Symposium 2010*, *1*–8. IEEE.

Son, N., Yang, S., & Na, J (2019). Hybrid forecasting model for short-term wind power prediction using modified long short-term memory. *Energies*, *12*(20), 3901.

Song, Z., Jiang, Y., & Zhang, Z (2014). Short-term wind speed forecasting with markov-switching model. *Applied Energy*, *130*, 103–112.

Speer, M. S (1997). Observational and modelling studies of australian severe weather doctoral dissertation, UNSW sydney.

Suykens, J. A. K, & De Brabanter, J., Lukas, L (2002). Weighted least squares support vector machines: robustness and sparse approximation. *Neurocomputing*, *48*(1–4), 85–105.

T. Blanchard & B. Samanta (2020). Wind speed forecasting using neural networks. *Wind Engineering*, *44*(1), 33–48.

Tang, Z. H., Zhao, G., Cao, S (2019). Wind speed prediction based on CEEMDAN-ESN and error correction strategy. *In 2019 Chinese Automation Congress (CAC)*, *1568*–1571. IEEE.

Tian Z, & Chen H (2021). Multi-step short-term wind speed prediction based on integrated multi-model fusion. *Applied Energy*, *298*, 117248.

Tian, Z (2021). Modes decomposition forecasting approach for ultra-short-term wind speed. *Applied Soft Computing*, *105*, 107303.

Tian, Z., & Chen, H (2021). A novel decomposition-ensemble prediction model for ultra-short-term wind speed. *Energy Conversion and Management*, *248*, 114775.

Tian, Z., Li, H., & Li, F (2021). A combination forecasting model of wind speed based on decomposition. *Energy Reports*, *7*, 1217–1233.

Tian, Z., Li, S., & Wang, Y (2020). A prediction approach using ensemble empirical mode decomposition-permutation entropy and regularized extreme learning machine for short-term wind speed. *Wind Energy*, *23*(2), 177–206.

Torres, J L, Garcia, A, & De Blas M, & De Francisco A (2005). Forecast of hourly average wind speed with ARMA models in navarre (spain). *Solar Energy*, *79*(1), 65–77.

Torrey, L., & Shavlik, J (2010). Transfer learning. *In Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques*, *242*–264. IGI Global.

Ullah, S., Chen, X., Han, H., Wu, J., Dong, J., Liu, R., ... & Yu, P. L (2025). A novel hybrid ensemble approach for wind speed forecasting with dual-stage decomposition strategy using optimized GRU and transformer models. *Energy*, *329*, 136739.

Valdivia-Bautista, S. M., Domínguez-Navarro, J. A., Pérez-Cisneros, M (2023). Artificial intelligence in wind speed forecasting: a review. *Energies*, *16*(5), 2457.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, *30*.

Verma, M., Ghritlahre, H. K., & Chandrakar, G (2023). Wind speed prediction of central region of chhattisgarh (india) using artificial neural network and multiple linear regression technique: a comparative study. *Annals of Data Science*, *10*(4), 851–873.

Vo, T (2023). An enhancement of transformer-based architecture with randomized regularization for wind speed prediction. *Journal of Intelligent & Fuzzy Systems*, *1*–17.

W. Fu, K. Zhang, K. Wang (2021). A hybrid approach for multi-step wind speed forecasting based on two-layer decomposition, improved hybrid DE-HHO optimization and KELM. *Renewable Energy*, *164*, 211–229.

W. Xu, P. Liu, L. Cheng (2021). Multi-step wind speed prediction by combining a WRF simulation and an error correction strategy. *Renewable Energy*, *163*, 772–782.

Walker, A M (1962). Large-sample estimation of parameters for autoregressive processes with moving-average residuals. *Biometrika*, *49*(1/2), 117–131.

Wang J, & Li Y (2018). Short-term wind speed prediction using signal preprocessing technique and evolutionary support vector regression. *Neural Processing Letters*, *48*, 1043–1061.

Wang, J, Zhou, Q, & Jiang, H (2015). Short-term wind speed forecasting using support vector regression optimized by cuckoo optimization algorithm. *Mathematical Problems in Engineering*, *2015*.

Wang, R, & Li W, & Bagen B (2011). Development of wind speed forecasting model based on the weibull probability distribution. *In 2011 International Conference on Computer Distributed Control and Intelligent Environmental Monitoring*, *2062*–2065. IEEE.

Wang, C. H., Huang, K. Y., Yao, Y., & Chiu, J. C (2022). Lightweight deep learning: an overview. IEEE consumer electronics magazine.

Wang, F., Chen, P., Zhen, Z (2022). Dynamic spatio-temporal correlation and hierarchical directed graph structure based ultra-short-term wind farm cluster power forecasting method. *Applied Energy*, *323*, 119579.

Wang, H., Han, S., Liu, Y (2019). Sequence transfer correction algorithm for numerical weather prediction wind speed and its application in a wind power forecasting system. *Applied Energy*, *237*, 1–10.

Wang, H., Li, Y., Xiong, M (2023). A combined wind speed prediction model based on data processing, multi-objective optimization and machine learning. *Energy Reports*, *9*, 413–421.

Wang, H., Yan, J., Liu, Y (2017). Multi-step-ahead method for wind speed prediction correction based on numerical weather prediction and historical measurement data. *Journal of Physics: Conference Series*, *926*(1), 012007.

Wang, J., & Li, Y (2018). Multi-step ahead wind speed prediction based on optimal feature extraction, long short term memory neural network and error correction strategy. *Applied Energy*, *230*, 429–443.

Wang, J., Feng, C., & He, J (2021). Wind speed prediction in the hexi corridor region of china using an enhanced multiscale nonlinear ensemble model. *Energy Technology*, *9*(12), 2100700.

Wang, J., Guo, H., Li, Z (2023). Quantile deep learning model and multi-objective opposition elite marine predator optimization algorithm for wind speed prediction. *Applied Mathematical Modelling*, *115*, 56–79.

Wang, J., Wang, S., Zeng, B (2022). A novel ensemble probabilistic forecasting system for uncertainty in wind speed. *Applied Energy*, *313*, 118796.

Wang, J., Wang, Y., & Li, Y (2018). A novel hybrid strategy using three-phase feature extraction and a weighted regularized extreme learning machine for multi-step ahead wind speed prediction. *Energies*, *11*(2), 321.

Wang, X., Li, X., & Su, J (2023). Distribution drift-adaptive short-term wind speed forecasting. *Energy*, *273*, 127209.

Wang, Y., & Gui, R (2022). A hybrid model for GRU ultra-short-term wind speed prediction based on tsfresh and sparse PCA. *Energies*, *15*(20), 7567.

Wang, Y., Xu, H., Song, M., & Zhang, L (2023). A convolutional transformer-based truncated gaussian density network with data denoising for wind speed forecasting. *Applied Energy*, *333*, 120601.

Williams, D., Evans, J. T., Gomm, J. B., & Jones, D. I (1970). Application of an artificial neural network to on-line control of a process. *WIT Transactions on Information and Communication Technologies*, *1*.

Wold, S, & Esbensen K, & Geladi P (1987). Principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, *2*(1–3), 37–52.

Wu Y K, & Hong J S (2007). A literature review of wind forecasting technology in the world. 2007 IEEE lausanne power tech, 504–509.

Wu, H., Meng, K., Fan, D (2022). Multistep short-term wind speed forecasting using transformer. *Energy*, *261*, 125231.

Wu, J., Li, N., Zhao, Y (2022). Usage of correlation analysis and hypothesis test in optimizing the gated recurrent unit network for wind speed forecasting. *Energy*, *242*, 122960.

Wu, Q., Zheng, H., Guo, X (2022). Promoting wind energy for sustainable development by precise wind speed prediction based on graph neural networks. *Renewable Energy*, *199*, 977–992.

Wu, T., & Ling, Q (2024a). Mixformer: mixture transformer with hierarchical context for spatio-temporal wind speed forecasting. *Energy Conversion and Management*, *299*, 117896.

Wu, T., & Ling, Q (2024b). STELLM: spatio-temporal enhanced pre-trained large language model for wind speed forecasting. *Applied Energy*, *375*, 124034.

Wu, Y. X., Wu, Q. B., & Zhu, J. Q (2019). Data-driven wind speed forecasting using deep feature extraction and LSTM. *IET Renewable Power Generation*, *13*(12), 2062–2069.

X. Guo, C. Zhu, J. Hao (2023). Multi-step wind speed prediction based on an improved multi-objective seagull optimization algorithm and a multi-kernel extreme learning machine. *Applied Intelligence*, *53*(13), 16445–16472.

X. Jiao, D. Zhang, D. Song (2023). Wind speed prediction based on VMD-BLS and error compensation. *Journal of Marine Science and Engineering*, *11*(5), 1082.

X. Li, V. Metsis, H. Wang (2022). Tts-gan: a transformer-based time-series generative adversarial network. *International Conference on Artificial Intelligence in Medicine*, *133*–143.

X. Xu, G. Yang, J. Luo (2022). A multi-location short-term wind speed prediction model based on spatiotemporal joint learning. *Renewable Energy*, *183*, 148–159.

X. Xu, S. Hu, H. Shao (2023). A spatio-temporal forecasting model using optimally weighted graph convolutional network and gated recurrent unit for wind speed of different sites distributed in an offshore wind farm. *Energy*, *284*, 128565.

X. Xu, S. Hu, P. Shi (2023). Natural phase space reconstruction-based broad learning system for short-term wind speed prediction: case studies of an offshore wind farm. *Energy*, *262*, 125342.

Xia, X., Luo, Y., Li, P., Chang, R., Liao, Z., & Huang, L (2026). Systematic evaluation of transformer-based time series forecasting models for post-processing wrf-simulated wind speed and predicting short-term power output. *Applied Energy*, *403*, 127070.

Xian H, & Che J (2022). Unified whale optimization algorithm based multi-kernel SVR ensemble learning for wind speed forecasting. *Applied Soft Computing*, *130*, 109690.

Xiang, J., Qiu, Z., Hao, Q (2020). Multi-time scale wind speed prediction based on wt-bi-lstm. *MATEC Web of Conferences*, *309*, 05011. https://doi.org/10.1051/matecconf/202030905011

Xiang, L., Deng, Z., & Hu, A (2019). Forecasting short-term wind speed based on IEWT-LSSVM model optimized by bird swarm algorithm. *IEEE Access*, *7*, 59333–59345.

Xiang, L., Li, J., Hu, A (2020). Deterministic and probabilistic multi-step forecasting for short-term wind speed based on secondary decomposition and a deep learning method. *Energy Conversion and Management*, *220*, 113098.

Xiaolan W, & Hui L (2009). One-month ahead prediction of wind speed and output power based on EMD and LSSVM. *In 2009 International Conference on Energy and Environment Technology*, *439*–442. IEEE.

Xiong, Z., Yao, J., Huang, Y., Yu, Z., & Liu, Y (2024). A wind speed forecasting method based on EMD-MGM with switching QR loss function and novel subsequence superposition. *Applied Energy*, *353*, 122248.

Xiu, C. B., & Guo, F. H (2013). Wind speed prediction by chaotic operator network based on kalman filter. *Science China Technological Sciences*, *56*, 1169–1176.

Xu, X., Hu, S., Shi, P., Shao, H., Li, R., & Li, Z (2023). Natural phase space reconstruction-based broad learning system for short-term wind speed prediction: case studies of an offshore wind farm. *Energy*, *262*, 125342.

Y. Ren, Z. Li, L. Xu (2023). The data-based adaptive graph learning network for analysis and prediction of offshore wind speed. *Energy*, *267*, 126590.

Y. Zhang, B. Chen, G. Pan (2019). A novel hybrid model based on VMD-WT and PCA-BP-RBF neural network for short-term wind speed forecasting. *Energy Conversion and Management*, *195*, 180–197.

Y. Zhang, Y. Zhao, X. Shen (2022). A comprehensive wind speed prediction system based on monte carlo and artificial intelligence algorithms. *Applied Energy*, *305*, 117815.

Yaghoubirad, M., Azizi, N., Farajollahi, M (2023). Deep learning-based multistep ahead wind speed and power generation forecasting using direct method. *Energy Conversion and Management*, *281*, 116760.

Yan, B, Shen, R, & Li, K (2023). Spatio-temporal correlation for simultaneous ultra-short-term wind speed prediction at multiple locations. *Energy*, *284*, 128418.

Yang, D., Li, M., Guo, J. E., & Du, P (2024). An attention-based multi-input LSTM with sliding window-based two-stage decomposition for wind speed forecasting. *Applied Energy*, *375*, 124057.

Yang, J., Astitha, M, & Delle Monache, L (2018). An analog technique to improve storm wind speed prediction using a dual NWP model approach. *Monthly Weather Review*, *146*(12), 4057–4077.

Yang, K., Wang, B., Qiu, X (2022). Multi-step short-term wind speed prediction models based on adaptive robust decomposition coupled with deep gated recurrent unit. *Energies*, *15*(12), 4221.

Yang, Y., & Yang, Y (2020). Hybrid prediction method for wind speed combining ensemble empirical mode decomposition and bayesian ridge regression. *IEEE Access*, *8*, 71206–71218.

Yang, Z., & Wang, J (2018). A combination forecasting approach applied in multistep wind speed forecasting based on a data processing strategy and an optimized artificial intelligence algorithm. *Applied Energy*, *230*, 1108–1125.

Yao, H., Tan, Y., Hou, J (2023). Short-term wind speed forecasting based on the EEMD-GS-GRU model. *Atmosphere*, *14*(4), 697.

Yao, X., & Xu, H (2025). A hybridgcn-gru approach for high-accuracy wind speed forecasting via multi-graph feature fusion and dynamic modeling. *Information Sciences*, *122805*.

Ye, R., Li, X., Ye, Y (2022). DynamicNet: a time-variant ODE network for multi-step wind speed prediction. *Neural Networks*, *152*, 118–139.

Ye, X. W., Ding, Y., & Wan, H. P (2021). Probabilistic forecast of wind speed based on bayesian emulator using monitoring data. *Structural Control and Health Monitoring*, *28*(1), e2650.

Yesilbudak, M., Sagiroglu, S., & Colak, I (2013). A new approach to very short term wind speed prediction using k-nearest neighbor classification. *Energy Conversion and Management*, *69*, 77–86.

Yu, C, Li, Y, & Zhao, L (2023). A novel time-frequency recurrent network and its advanced version for short-term wind speed predictions. *Energy*, *262*, 125556.

Yu, C., Li, Y., & Zhang, M (2017). An improved wavelet transform using singular spectrum analysis for wind speed forecasting based on elman neural network. *Energy Conversion and Management*, *148*, 895–904.

Yu, C., Li, Y., Bao, Y (2018). A novel framework for wind speed prediction based on recurrent neural networks and support vector machine. *Energy Conversion and Management*, *178*, 137–145.

Yu, C., Li, Y., Chen, Q (2022). Matrix-based wavelet transformation embedded in recurrent neural networks for wind speed prediction. *Applied Energy*, *324*, 119692.

Yu, J., Chen, K., Mori, J (2013). A gaussian mixture copula model based localized gaussian process regression approach for long-term wind speed prediction. *Energy*, *61*, 673–686.

Z. Peng, S. Peng, L. Fu (2020). A novel deep learning ensemble model with data denoising for short-term wind speed forecasting. *Energy Conversion and Management*, *207*, 112524.

Z. Shi, H. Liang, & V. Dinavahi (2018). Wavelet neural network based multiobjective interval prediction for short-term wind speed. *IEEE Access*, *6*, 63352–63365.

Zajaczkowski, F. J., Haupt, S. E., & Schmehl, K. J (2011). A preliminary study of assimilating numerical weather prediction data into computational fluid dynamics models for wind prediction. *Journal of Wind Engineering and Industrial Aerodynamics*, *99*(4), 320–329.

Zárate-Miñano, R, & Anghel M, & Milano F (2013). Continuous wind speed models based on stochastic differential equations. *Applied Energy*, *104*, 42–49.

Zeller, A, Faltermeier, R, & Keck, I R (2010). Empirical mode decomposition-an introduction. *In The 2010 international joint conference on neural networks (IJCNN)*, *1*–8. IEEE.

Zha, W., Jin, Y., Sun, Y., & Li, Z (2023). A wind speed vector-wind power curve modeling method based on data denoising algorithm and the improved transformer. *Electric Power Systems Research*, *214*, 108838.

Zhang, S, Kang, Z, & Zhang Z, & Li J (2019). A hybrid model for forecasting traffic flow: using layerwise structure and markov transition matrix. *IEEE Access*, *7*, 26002–26012.

Zhang, C., Ji, C., Hua, L (2022). Evolutionary quantile regression gated recurrent unit network based on variational mode decomposition, improved whale optimization algorithm for probabilistic short-term wind speed prediction. *Renewable Energy*, *197*, 668–682.

Zhang, C., Peng, T., & Nazir, M. S (2022). A novel hybrid approach based on variational heteroscedastic gaussian process regression for multi-step ahead wind speed forecasting. *International Journal of Electrical Power & Energy Systems*, *136*, 107717.

Zhang, C., Wei, H., Zhao, J., Liu, T., Zhu, T., & Zhang, K (2016). Short-term wind speed forecasting using empirical mode decomposition and feature selection. *Renewable Energy*, *96*, 727–737.

Zhang, C., Wei, H., Zhao, X (2016). A gaussian process regression based hybrid approach for short-term wind speed prediction. *Energy Conversion and Management*, *126*, 1084–1092.

Zhang, D., Peng, X., Pan, K (2019). A novel wind speed forecasting based on hybrid decomposition and online sequential outlier robust extreme learning machine. *Energy Conversion and Management*, *180*, 338–357.

Zhang, G., & Liu, D (2020). Causal convolutional gated recurrent unit network with multiple decomposition methods for short-term wind speed forecasting. *Energy Conversion and Management*, *226*, 113500.

Zhang, G., Zhang, Y., Wang, H., Liu, D., Cheng, R., & Yang, D (2024). Short-term wind speed forecasting based on adaptive secondary decomposition and robust temporal convolutional network. *Energy*, *288*, 129618.

Zhang, J., Walter, G. G., Miao, Y., & Lee, W. N. W (1995). Wavelet neural networks for function learning. *IEEE Transactions on Signal Processing*, *43*(6), 1485–1497.

Zhang, K., Li, X., & Su, J (2022). Variable support segment-based short-term wind speed forecasting. *Energies*, *15*(11), 4067.

Zhang, X., Zhou, X., Lin, M., & Sun, J (2018). Shufflenet: an extremely efficient convolutional neural network for mobile devices. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, *6848*–6856.

Zhang, Y., Wang, J., & Wang, X (2014). Review on probabilistic forecasting of wind power generation. *Renewable and Sustainable Energy Reviews*, *32*, 255–270.

Zhang, Y., Zhao, Y., Kong, C (2020). A new prediction method based on VMD-PRBF-ARMA-E model considering wind speed characteristic. *Energy Conversion and Management*, *203*, 112254.

Zhang, Z., Qin, H., Liu, Y (2019). Long short-term memory network based on neighborhood gates for processing complex causality in wind speed prediction. *Energy Conversion and Management*, *192*, 37–51.

Zhang, Z., Qin, H., Liu, Y (2019). Wind speed forecasting based on quantile regression minimal gated memory network and kernel density estimation. *Energy Conversion and Management*, *196*, 1395–1409.

Zhang, Z., Ye, L., Qin, H (2019). Wind speed prediction method using shared weight long short-term memory network and gaussian process regression. *Applied Energy*, *247*, 270–284.

Zhao, E., Zhao, J., Liu, L (2015). Hybrid wind speed prediction based on a self-adaptive ARIMAX model with an exogenous WRF simulation. *Energies*, *9*(1), 7.

Zhao, J., Guo, Z. H., Su, Z. Y (2016). An improved multi-step forecasting model based on WRF ensembles and creative fuzzy systems for wind speed. *Applied Energy*, *162*, 808–826.

Zheng, L., Zhou, B., Or, S. W (2021). Spatio-temporal wind speed prediction of multiple wind farms using capsule network. *Renewable Energy*, *175*, 718–730.

Zhu, Q., Chen, J., Shi, D (2019). Learning temporal and spatial correlations jointly: a unified framework for wind speed prediction. *IEEE Transactions on Sustainable Energy*, *11*(1), 509–523.

Zhu, S., Yuan, X., Xu, Z (2019). Gaussian mixture model coupled recurrent neural networks for wind speed interval forecast. *Energy Conversion and Management*, *198*, 111772.

Zhu, X., Liu, R., Chen, Y (2021). Wind speed behaviors feather analysis and its utilization on wind speed prediction using 3d-cnn. *Energy*, *236*, 121523.

Zou, F., Fu, W., Fang, P (2020). A hybrid model based on multi-stage principal component extraction, GRU network and KELM for multi-step short-term wind speed forecasting. *IEEE Access*, *8*, 222931–222943.

Zuluaga, C. D., Alvarez, M. A., & Giraldo, E (2015). Short-term wind speed prediction based on robust kalman filtering: an experimental comparison. *Applied Energy*, *156*, 321–330.