DTU Library

Adaptive expert fusion model for online wind power prediction

Wang, Renfang; Wu, Jingtong; Cheng, Xu; Liu, Xiufeng; Qiu, Hong

Published in:
Neural Networks

Link to article, DOI:
10.1016/j.neunet.2024.107022

Publication date:
2025

Document Version
Publisher's PDF, also known as Version of record

Neural Networks 184 (2025) 107022

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Full Length Article

Adaptive expert fusion model for online wind power prediction

Renfang Wang [a], Jingtong Wu [b], Xu Cheng [c], Xiufeng Liu [c,*], Hong Qiu [a,*]

[a] College of big data and software engineering, Zhejiang Wanli University, 315200 Ningbo, China
[b] College of Information, Shanghai Ocean University, 201306 Shanghai, China
[c] Department of Technology, Management and Economics, Technical University of Denmark, 2800 Kgs., Lyngby, Denmark

ARTICLE INFO

Keywords:
Wind power prediction
Expert fusion model
Dynamic ensemble technique
Real-time forecasting
Intra-hour scheduling

ABSTRACT

Wind power prediction is a challenging task due to the high variability and uncertainty of wind generation and weather conditions. Accurate and timely wind power prediction is essential for optimal power system operation and planning. In this paper, we propose a novel Adaptive Expert Fusion Model (EFM+) for online wind power prediction. EFM+ is an innovative ensemble model that integrates the strengths of XGBoost and self-attention LSTM models using dynamic weights. EFM+ can adapt to real-time changes in wind conditions and data distribution by updating the weights based on the performance and error of the models on recent similar samples. EFM+ enables Bayesian inference and real-time uncertainty updates with new data. We conduct extensive experiments on a real-world wind farm dataset to evaluate EFM+. The results show that EFM+ outperforms existing models in prediction accuracy and error, and demonstrates high robustness and stability across various scenarios. We also conduct sensitivity and ablation analyses to assess the effects of different components and parameters on EFM+. EFM+ is a promising technique for online wind power prediction that can handle nonstationarity and uncertainty in wind power generation.

1. Introduction

Wind energy has emerged as a cornerstone of sustainable energy solutions, playing a pivotal role in the global energy transition toward renewable resources. Wind energy is crucial not only due to its potential to reduce carbon emissions but also because of the unique technical and operational challenges it presents, such as intermittency and grid integration. Unlike other renewable resources, the variability of wind requires more sophisticated forecasting models to ensure effective grid stability and reliability (Belaïd, Al-Sarihi, & Al-Mestneer, 2023; Jose, Panigrahi, Patil, Fernando, & Ramakrishna, 2020). A testament to its importance is the rapid capacity growth of wind energy, with an average annual increase of 9% expected between 2021 and 2030 (Mackenzie, 2022). This growth is driven by factors such as technological advancements, policy support, and increasing awareness of environmental sustainability. Notably, wind power contributed 7.6% of global electricity demand in 2022 (Ember, 2022), and is projected to reach a substantial 20% by 2030 (Hu, Liu, Wu, Zhao, Wang, & Liu, 2024; Liu & Zhang, 2020). The significant cost reductions in wind and solar technologies, with prices dropping by over 80% since 2009, have also been a crucial factor in this expansion (Wiser, 2021).

Despite the growth, wind energy has its unique challenges. Chief among them is the inherent uncertainty and variability of wind speeds, which directly impact power generation. This variability necessitates the development of adaptive and real-time prediction models that can respond to sudden changes in wind conditions, providing accurate and timely information for power system operations. Such models are critical for maintaining grid stability, particularly when integrating a high share of renewable energy (Mora, Spelling, van der Weijde, & Pavageau, 2019). Accurate prediction of wind power generation is essential for effective planning and operation of power systems. Reliable forecasts help optimize generation scheduling, balance loads, and manage operating costs (Sarshar, Moosapour, & Joorabian, 2017). They enable utility companies to better coordinate with other energy sources, minimize excess generation, and reduce costs, thus making the power system more efficient and reliable (Aguero, Takayesu, Novosel, & Masiello, 2017). From a grid operation perspective, accurate wind power prediction enables higher penetration of wind energy into the grid. It provides the necessary lead time for grid operators to manage the variability and uncertainty of wind power, enhancing grid stability and reliability (Zhao, Wu, Hu, Xu, & Rasmussen, 2015). At the policy level, wind power prediction plays a key role in setting and achieving renewable energy targets by ensuring that the grid has sufficient flexibility to integrate a high proportion of wind energy without compromising stability (Foley et al., 2020). Furthermore, insights

* Corresponding authors.
E-mail addresses: renfang_wangac@126.com (R. Wang), w1405523055@163.com (J. Wu), xuche@dtu.dk (X. Cheng), xiuli@dtu.dk (X. Liu), qiuhong@zwu.edu.cn (H. Qiu).

https://doi.org/10.1016/j.neunet.2024.107022
Received 26 February 2024; Received in revised form 19 October 2024; Accepted 3 December 2024
Available online 10 December 2024

from prediction models help researchers and technologists address both technical challenges, such as intermittency, and economic challenges, such as cost competitiveness (Rafique & Jianhua, 2018).

A variety of wind power prediction models have been developed over the years, demonstrating the significance and complexity of this task. These models can be broadly categorized into three types: physical models, statistical models, and machine learning models (Maldonado-Correa, Solano, & Rojas-Moncayo, 2021). Physical models use atmospheric and wind turbine data to predict long-term wind power, from hours to days ahead. However, they require extensive data and computations, which can be resource-intensive and time-consuming (Kashinath et al., 2021). Despite their ability to predict far in advance, physical models' accuracy is often limited due to uncertainties in input data and physical relationships (Hu et al., 2021). On the other hand, statistical and machine learning models employ historical wind farm data to capture relationships and patterns, providing short-term predictions from minutes to hours ahead. These models have been successful in capturing the non-linear and complex patterns of wind speeds and power generation. However, these models rely on the assumption that the underlying relationships remain static over time. This assumption often fails in the face of non-linear relationships or time-varying distributions, leading to inaccurate forecasts (Tawn & Browell, 2022). Neural network models, a subtype of machine learning models, attempt to overcome these limitations through adaptive learning. They can learn complex and non-linear patterns from data and adapt to changes in these patterns over time. However, they often lack online prediction capabilities required for real-time scheduling, limiting their utility in dynamic power system operations. A critical review of the literature reveals a noticeable gap in the area of real-time wind power prediction for intra-hour scheduling. Specifically, most existing studies lack real-time adaptability, and there is limited research focusing on the challenges associated with rapid changes in wind conditions. Addressing the need for intra-hour scheduling is crucial for improving grid reliability and ensuring efficient integration of wind energy into the power system (Zhang, Li, Gao, Liu, & Ren, 2021).

Thus, online wind power prediction has come into focus, and different design approaches have emerged. The use of acoustic data and active wake control for wind power prediction (Dittmer, Sharan, & Werner, 2022; Sun, Su, & He, 2024) improves prediction accuracy by obtaining additional information from the physical environment of the wind farm. However, these methods are highly dependent on data quality and have limited generalization capabilities. Adapting the least squares method for real-time, recursive wind power prediction enables continuous model parameter updates, accommodating dynamic changes in wind speed and other influencing factors (Qiu, Shi, Wang, Zhang, Liu, & Cheng, 2024; Zhang, Li, Li, Wang and Du, 2023). But they are highly sensitive to the initial parameter values; if the parameters are chosen poorly, the model may deviate from the correct path right from the start, leading to instability or inaccuracy in subsequent updates. Ensemble learning and online techniques mainly tackle the challenge of non-stationarity in wind data, enabling models to continuously adapt to the most recent data and improve the accuracy of real-time predictions (Jin et al., 2022; Peng et al., 2022). This has become the mainstream solution in the field of wind power prediction. However, these methods also face certain challenges, such as catastrophic forgetting, slow adaptability, and concept drift.

In view of the shortcomings of ensemble models and online learning combination techniques, the exploration of adaptive expert fusion models has been promoted. The adaptive expert fusion model not only inherits the advantages of addressing the non-stationary characteristics of wind power data, but also allows for dynamic adjustment according to real-time model performance, ensuring that the ensemble is constantly updated to reflect the current situation and reducing the negative impact caused by catastrophic forgetting. These models leverage the combined strengths of multiple individual prediction models through dynamic weighting schemes that adjust based on each model's performance. This approach has shown promise in various domains, including financial forecasting, medical diagnosis, and traffic prediction (Li, Yang et al., 2023; Tian et al., 2023). Specifically for wind power prediction, several studies have demonstrated the potential of Adaptive Expert Fusion Models to enhance forecast accuracy and robustness. For instance, an adaptive hybrid model was proposed that combines physical models with machine learning models using dynamic weighting based on weather conditions, demonstrating superior performance in accuracy and reliability (Tian et al., 2023). Similarly, another ensemble model integrates multiple neural network models with a weighting scheme that considers the uncertainty of each model's predictions. This effectively captures the time-varying patterns of wind power generation, thereby improving prediction accuracy (Li, Yang et al., 2023). While these studies have shown promising results, there are still challenges in effectively applying adaptive expert ensemble models to online wind power prediction. These challenges include designing a weighting scheme that can adapt to rapidly changing wind conditions and ensuring the real-time applicability of scheduling decisions within an hour. Moreover, mitigating the negative impacts of concept drift and slow adaptability in the process of online forecasting with adaptive expert ensemble models requires further research and solutions.

Therefore, to address the aforementioned issues, our research aims to develop a more dynamic and adaptive model for wind power prediction. Unlike previous adaptive models, the proposed method focuses on real-time adaptability and intra-hour scheduling to handle rapid changes in wind conditions. We utilize a sliding window approach to specifically mitigate the negative effects of concept drift and implement a dynamic adaptive module for real-time adjustment of the weighting scheme, alleviating slow adaptability issues and enhancing the model's real-time applicability. This model not only adapts dynamically but also ensures computational efficiency in real-time scenarios. We argue that to fully optimize generation scheduling and maximize wind energy utilization, we need online prediction models that can adapt to rapid changes in wind conditions and provide accurate predictions for intra-hour scheduling. This would require the models to process and learn from the most recent wind farm data, detect changes in wind power generation patterns, and update the forecasts in real-time.

In this context, we propose a novel *Adaptive Expert Fusion Model (EFM+)* for online wind power prediction. The use of XGBoost and self-attention LSTM offers distinct advantages: XGBoost effectively captures complex, non-linear relationships between wind power generation and various influencing factors, while self-attention LSTM excels at modeling temporal dependencies and identifying relevant features in time series data. This combination allows EFM+ to provide robust and accurate forecasts even in rapidly changing wind conditions. The EFM+ consists of two individual models: XGBoost, a gradient boosting decision tree model that can capture the complex and non-linear relationships between wind power generation and various factors; and self-attention LSTM, a recurrent neural network model that can capture the temporal dependencies and long-term patterns in the time series data. The EFM+ combines the predictions from these models using a dynamic weighting scheme that assigns higher weights to models that perform better on similar samples from the recent data. This scheme enables flexible Bayesian predictive inference and real-time updates of uncertainty as new data arrive. By weighting models based on their recent predictive performance, the EFM+ can capture the most accurate insights for the current situation and time-varying conditions. The EFM+ design recognizes that different models may perform differently under various wind conditions. Some models may show higher accuracy under stable wind conditions but perform less accurately under variable conditions. Therefore, at each prediction point, the EFM+ evaluates the recent performance of each model and adjusts their predictions accordingly. By dynamically adjusting the weight of models based on their current predictive skills, the EFM+ can optimally utilize different models to adapt to changes in the time-series distribution. This makes EFM+ particularly suitable for handling nonstationarity and for online prediction.

The main contributions of this paper are three-fold:

- We present a novel system for real-time wind power prediction that can handle nonstationarity and uncertainty in wind power generation. The EFM+ is a flexible and adaptive ensemble model that combines predictions from multiple models using a dynamic weighting scheme based on their current performance and error.
- We design a novel dynamic weighting scheme that assigns higher weights to models that perform better on similar samples from the recent data. Unlike traditional ensemble techniques that rely on fixed weights or global performance metrics, our scheme incorporates a subset-based error calculation that captures the local performance of each model around the prediction point. This scheme enables flexible Bayesian predictive inference and real-time updates of uncertainty as new data arrive. It also allows the EFM+ to exploit the distinct advantages of each model and adapt as required, delivering accurate and timely forecasts for intra-hour scheduling.
- We conduct extensive experiments on a real-world wind farm dataset and compare our EFM+ with several baselines and state-of-the-art methods. The results show that our EFM+ outperforms all the other methods in terms of prediction accuracy and error, and demonstrates high robustness and stability under different scenarios. We also perform sensitivity analysis and ablation studies to evaluate the impact of different components and parameters on the performance of our EFM+.

The rest of this paper is organized as follows: Section 2 reviews relevant literature; Section 3 outlines our data collection process; Section 4 presents the proposed Expert Fusion Model; Section 5 conducts the experiments to evaluate the proposed model, and presents our experimental results; Finally, Section 6 concludes the paper and presents the future work.

2. Literature review

Wind power prediction is a crucial task for optimizing generation scheduling and maximizing wind energy utilization. However, wind power prediction is challenging due to the uncertainty and volatility of wind speed, which is affected by various factors such as weather conditions, terrain features, turbine characteristics, etc. In this section, we review the existing literature on wind power prediction models and identify the research gap that motivates our study.

2.1. Clustering and optimization

Clustering is widely used in wind prediction because of its ability to identify data patterns, simplify data, detect anomalies, optimize prediction models, and effectively handle the complexity of multi-scale wind changes.

Several studies have used K_means for wind power generation forecasting to improve model accuracy and computational efficiency. For example, Huang, Chen, Yang, and Chen (2023), who combined the K_means algorithm with the optimized weighted average integration method, the optimized integration method gives more accurate predictions than a single model. Liu, Yang, and Zhang (2021), who designed a new deep convolutional cyclic network method, namely K-shaped and K-means-guided gated cyclic unit integrated convolutional neural network, which is used for short-term multi-step advance prediction of wind turbine power generation. Zhou, Huang, Lin, Chai and Wang (2024), who proposed an improved Snake optimization short- and long-term memory (ISO-LSTM) model and Gaussian hybrid model (GMM) clustering to predict wind power from an adaptive perspective. Moreover, Ultra-short-term photovoltaic power (2024), who introduced a photovoltaic power prediction model based on similar day clustering and time convolutional network (TCN), with a two-way short- and long-term memory (BiLSTM) model. By reading a large number of relevant literatures, it is found that the clustering method has a very promising application in wind power prediction. Using clustering to process wind datasets not only reduces data complexity, but also improves the accuracy of the model.

Optimization algorithms play a key role in machine learning, enabling models to effectively minimize loss functions and improve prediction accuracy. Common methods include gradient-based methods (Li et al., 2019), particle swarm optimization (Sun, Wang and Yang, 2022), adaptive moment estimation (Chang, Zhang, & Chen, 2019), genetic algorithms (Chen et al., 2019), differential evolution algorithms (Li, He, Ma, Wang, & Zhang, 2020), etc., each of which is suitable for different problem areas. In online wind power prediction, where data is highly dynamic and models must adapt in real-time, the choice of optimization algorithm becomes crucial. Traditional batch optimization techniques, although effective on static datasets, often struggle with the real-time demands of online data streams. As wind conditions change rapidly, forecasting models need optimization algorithms that can quickly adapt to new information without waiting for a full dataset. To address the challenges of online learning, Stochastic Gradient Descent (SGD) emerges as a highly suitable optimization algorithm (Hoi, Sahoo, Lu, & Zhao, 2018). Unlike traditional methods, SGD updates model parameters after each new data point, making it ideal for real-time learning and dynamic environments. This single-sample update mechanism allows models to quickly adapt to changes in the data stream, making SGD a natural fit for online learning systems. Many scholars choose SGD to combine with different models to achieve online prediction results. For example, Huang and Kuo (2018), who used SGD multiple models for online learning. Fekri, Patel, Grolinger, and Sharma (2021), who designed adaptive RNN online based on the characteristics of SGD, a load prediction method that can continuously learn from newly arrived data and adapt to new patterns. Hu et al. (2020), who proposed a new substrate-based spatiotemporal wind power predictor (CSTWPP) and combined it with SGD to achieve online learning.

Through extensive practical evidence, it has been proven that SGD can efficiently handle real-time incoming data. By gradually updating model parameters, it is particularly suitable for online learning scenarios. Furthermore, SGD's method of updating through single samples or small batches of data significantly reduces the demand for computational resources, ensuring that the model can rapidly iterate while maintaining accuracy.

2.2. Wind power prediction models

We can categorize the existing models into two groups based on the data sources they use: meteorological data-based models and historical data-based models.

Meteorological data-based models use weather forecasts or measurements as input features to predict wind power output. These models can capture the physical relationship between wind speed and power output, and can provide long-term predictions for different time horizons. However, these models require high-quality and high-resolution meteorological data, which may not be available or reliable in some regions or scenarios. Moreover, these models may not account for the nonlinear and complex effects of other factors such as turbine characteristics, grid conditions, etc. One example of meteorological data-based models is the one proposed by Sun, Sun, Wang and Zhao (2022), who divided the meteorological processes into four types: stable, unstable, neutral, and transitional. They then established a long short-term memory (LSTM) network for each meteorological process to predict wind power output. They claimed that their model could improve the prediction accuracy by considering the different characteristics of each meteorological process. Another example is the one proposed by Zhang, Zhai and Sun (2023), who used an empirical mode decomposition-extreme learning machine (EMD-ELM) method to predict wind power output based on meteorological data. They first applied a hierarchical clustering method to group the meteorological data into different clusters based on their similarity. Then they used EMD to decompose each cluster into several intrinsic mode functions (IMFs), which represent different frequency components of the data. Finally, they used ELM to train and predict each IMF separately, and then aggregated them to obtain the final prediction result. They claimed that their method could enhance the data characteristics of similar weather conditions and improve the prediction accuracy. Other influential works in this category include Tian et al. (2023), who combined improved variational mode decomposition (IVMD) and fuzzy entropy (FE) with Informer to predict wind power output; Xinxin, Xiaopan, Xueyi, and Shijia (2023), who combined improved complete ensemble empirical mode decomposition with adaptive noise (ICEEMDAN), multiscale fuzzy entropy (MFE), LSTM and Informer to predict wind speed; and Guan et al. (2023), who used a graph convolutional network (GCN) to extract the spatial–temporal features of meteorological data and then used an attention-based LSTM to predict wind power output. Historical data-based models use past wind speed or power output data as input features to predict future wind power output. These models can capture the temporal patterns and correlations of wind speed or power output data, and can provide short-term predictions for intra-hour or hourly time horizons. However, these models may not account for the external factors that affect wind speed or power output, such as weather conditions, turbine characteristics, grid conditions, etc. Moreover, these models may suffer from insufficient or noisy historical data, which may degrade their prediction performance. Accurately modeling and handling time-varying delays in historical data is crucial for improving prediction accuracy, as demonstrated in studies applying fuzzy logic and Lyapunov-Krasovskii Functionals (LKFs) to address this challenge in related control systems (Aslam, Radhika, Chandrasekar, & Zhu, 2024). One example of historical data-based models is the one proposed by Li, Wang, Geng, Jin and Xu (2023), who used a genetic algorithm (GA) to optimize the parameters of an extreme gradient boosting (XGBoost) regression model to predict wind power output based on historical data. They claimed that their model could improve the prediction accuracy and speed by using GA to find the optimal parameter values for XGBoost. Additionally, Wu, Chen, Du, Mao, and Ju (2023) used a combined model based on autoregressive moving average-gated recurrent unit (ARMA-GRU) to predict short-term wind power output based on historical data. They first used ARMA to model the linear trend of wind power output data, then used GRU to model the nonlinear fluctuation of wind power output data, and finally used an error correction model to correct the prediction errors. They claimed that their model could improve the prediction precision by combining ARMA and GRU. Other related works in this category include Aruna et al. (2023), who used a sparrow search optimization with deep belief network (SSODBN) technique to predict wind power output; Wang et al. (2023), who used a combined model based on T-LSTNet and Markov chain to predict short-term wind power output; and Han et al. (2023), who used a hybrid model based on wavelet transform, empirical mode decomposition, and LSTM to predict wind power output. Moreover, Liu and Zhang (2022), who revisited the data-driven short-term wind power prediction (WPP) modeling process with introducing a new data usage paradigm, using SCADA data of both high and low sampling resolutions as model inputs, and a new WPP modeling framework based on bilateral branching learning has been proposed, achieving excellent results. To conclude, we have reviewed the existing literature on wind power prediction models and identified the research gap that motivates our study. We found that most existing models either use only one type of data (meteorological or historical) or a fixed weighting scheme to combine them, which may not account for the external factors that affect wind speed or power output, such as weather conditions, turbine characteristics, grid conditions, etc. Moreover, we found that most existing models are not designed for online prediction and learning, which is essential for intra-hour scheduling. Therefore, we propose a novel Adaptive Expert Fusion Model (EFM+) for online wind power prediction, which can integrate both meteorological data and historical data as input features, and dynamically adjust the weights assigned to different models based on their current performance and error. Our proposed framework can also update the models and the weights with new data as it becomes available, and provide short-term predictions using the most recent wind power generation data. This way, our framework can overcome the limitations of the existing models and provide accurate and adaptive predictions for intra-hour scheduling.

2.3. Ensemble prediction model

Ensemble model By combining the results of multiple models, the ensemble model can effectively reduce the prediction error of a single model, thereby generating more accurate predictions. It can also reduce the risk of overfitting. A single model may perform well on the training data, but often overfits on the actual test data, while the ensemble model improves the model's ability to adapt to new data by integrating the prediction results of different models. generalization ability. In addition, ensemble models can achieve a better balance between bias and variance and avoid performance problems caused by overly simple or overly complex models.

Therefore, ensemble models have gained much attention. For example, Deon et al. (2024) introduced a comprehensive approach by proposing 42 ensemble learning models specifically designed for forecasting wind power time series. Created by combining various machine learning models and leveraging different aggregation methods. Zhang et al. (2024) combined a multi-network deep integration model with the improved manta ray foraging optimization algorithm (IMRFO) and wind power prediction feature selection. Layer 1 builds an ensemble prediction model, combining Extreme Gradient Boosting (XGBoost), Gated Recurrent Units (GRU), and Temporal Convolutional Networks (TCN). In layer 2, Random Forest (RF) is employed to make predictions based on the fusion results. Xiao, Wu, He, Hu, and Yi (2024) proposed a hybrid wind power prediction model based on the "decomposition-reconstruction-integration" strategy, which consists of four main parts: decomposition, reconstruction, prediction, and integration. Zhou, Wei, Kuang and Mahfoud (2024) proposed an ensemble model named Completely Ensemble Empirical Modal Decomposition with Adaptive Noise (CEEMDAN) and Convolutional Bidirectional Long Short-Term Memory (CNN-BiLSTM), which combines data preprocessing technology, features Selection methods, deep ensemble models, and adaptive control. Zhu, Xu, Lin, Ming, and Tan (2024) proposed a multi-objective ensemble learning short-term wind speed interval prediction based on clustering, which can provide accurate and reliable wind speed interval prediction and support energy dispatch planning. Moreover, Özdemir (2024) proposed a new ensemble wind speed prediction model based on artificial neural networks, a system that uses meteorological nonlinear data for high-precision execution, and filled the gap in this field.

Considering the application of related ensemble models in wind power prediction, compared to individual models, they can improve prediction accuracy, reduce overfitting, increase robustness, and adapt to complex problems. However, there are some drawbacks including high computational cost, high model complexity, difficulty in optimization, and the risk of over-integration. EFM+ compensates for the shortcomings of traditional ensemble models when facing different data scenarios by dynamically adjusting the weights of the expert models. It can adaptively choose the most suitable expert based on the characteristics and changes of the input data, thereby enhancing the overall prediction performance.

2.4. Dynamic ensemble technique

A dynamic ensemble technique is a method that combines predictions from multiple models using a dynamic weighting scheme based on their current performance. This technique can capture the most accurate insights for the current situation and time-varying conditions (Yan, Liu, Han, Wang, & Feng, 2015). It can also adapt to nonstationarity and online prediction. A dynamic ensemble technique is particularly suitable for wind power prediction, as it can exploit the advantages of different models and handle the uncertainty and variability of wind conditions.

Several studies have applied dynamic ensemble techniques for wind power prediction, using different models and methods to construct and combine the individual predictors. For example, Hossain, Chakrabortty, Elsawah, and Ryan (2021) proposed a dynamic ensemble wind speed prediction model based on hybrid deep reinforcement learning. The model consists of real-time decomposition, ensemble learning, multi-objective optimization, and reinforcement learning to ensure effective prediction. While similar to our approach in using an ensemble of models, their method focuses on very short-term wind speed forecasting and relies on deep reinforcement learning, which may be computationally expensive. The model was tested on two real-world datasets and showed better performance than several benchmark models. Al-Dahidi, Baraldi, Zio, and Legnani (2017) proposed a dynamic weighting ensemble approach for wind energy production prediction. The approach is based on an ensemble of artificial neural networks (ANNs), which receive weather forecast variables as input and predict the wind plant energy production. The approach uses a dynamic weighting scheme that combines the individual models' outcomes proportionally to their local performances in the neighborhood of the test pattern under analysis. Their weighting scheme shares similarities with our approach in considering local performance. However, they utilize a fixed neighborhood size for evaluating local performance, while our method dynamically selects a subset of similar samples based on their distance to the prediction point using a k-nearest neighbor approach. The approach was compared with a real dataset and found to provide more accurate predictions than other methods. Liu and Zhang (2023) proposed an innovative two-party participatory data-driven modeling framework that utilizes a pre-training phase and a fine-tuning phase. In the fine-tuning phase, local data from the previous iteration and global latent features are used to dynamically adjust the model's learning of local latent features. Moreover, Bilal, Adjallah, Sava, Yetilmezsoy, and Ouassaid (2023) proposed a dynamic ensemble model based on adaptive neuro-fuzzy inference systems (ANFIS) for short-term wind power prediction. The model uses a fuzzy clustering algorithm to partition the input space into different regions and trains an ANFIS model for each region. The model then uses a weighted average method to combine the outputs of the regional ANFIS models based on their prediction errors. Their approach differs from ours using regional ANFIS models and a global error-based weighting scheme. In contrast, our EFM+ utilizes a combination of globally trained models (XGBoost and self-attention LSTM) and a subset-based error calculation for weighting, which allows for more fine-grained adaptation to local changes in the data distribution. The model was evaluated on real wind farm data and showed better results than other models.

These studies demonstrate the potential and effectiveness of dynamic ensemble techniques for wind power prediction. However, they also have some limitations and challenges. First, the selection of appropriate individual models and weighting schemes for different wind conditions and forecasting horizons. Different models may have different strengths and weaknesses under various situations, and the optimal weighting scheme may vary over time. Therefore, choosing the best combination of models and weights is a challenging task that requires careful analysis and evaluation (Martín-Vázquez, Aler, & Galván, 2018). Second, the trade-off between accuracy and computational complexity of the dynamic ensemble techniques. Dynamic ensemble techniques require more computations than single models, as they involve multiple models and dynamic weighting schemes. This may increase the computational cost and time of the prediction process, especially for large-scale datasets and online prediction scenarios. Therefore, finding a balance between accuracy and efficiency is a crucial issue for dynamic ensemble techniques (Albukhanajer, Jin, & Briffa, 2015). Third, the robustness and stability of dynamic ensemble techniques under uncertain and noisy data are crucial. Wind power prediction is inherently susceptible to various sources of uncertainty, including measurement errors, weather fluctuations, and turbine failures. Radhika, Chandrasekar, Vijayakumar, and Zhu (2023) investigated input-to-state stability for stochastic neural networks with time-varying delays, emphasizing the importance of robustness to noise disturbances, a key consideration for wind power prediction models. Further highlighting the complexity of non-stationarity, Chandrasekar, Radhika, and Zhu (2022) examined input-to-state stability for stochastic neural networks with probabilistic time-varying delays, advocating for the broader concept of mean-square exponential input-to-state stability for such systems. These factors can affect the quality and reliability of the input data and the output predictions. Therefore, ensuring the robustness and stability of the dynamic ensemble techniques under these conditions is an important challenge that requires robust data processing and error correction methods (Qin, Wang, Du, Chen, & Yan, 2023).

Therefore, there is still room for improvement and innovation in this field. In this paper, we aim to address these challenges by proposing a novel EFM for online wind power prediction, which combines predictions from multiple models using a dynamic ensemble technique with time-varying parameters, allowing for flexible Bayesian predictive inference and real-time updates as new data arrives. Our proposed EFM+ model distinguishes itself through its unique dynamic weighting scheme. This scheme employs a subset-based error calculation that captures the local performance of each model around the prediction point, enabling fine-grained adaptation to concept drift and non-stationarity in wind power data. The proposed model differentials other in using a combined system for real-time wind power prediction, introducing time-dependent model factors for nonstationary data, and supporting short-term forecasts for intra-hour scheduling.

3. Materials

This study utilizes a real-world wind farm dataset provided by the Baidu KDD Cup. The dataset encompasses spatial and temporal information from 134 wind turbines, including their power generation and weather forecast variables. Data collection occurs at 10-min intervals via the Supervisory Control and Data Acquisition (SCADA) system, which records various parameters such as wind speed, pitch angle, rotational speed, and active power output. The dataset spans one year, from January 1, 2022, to December 31, 2022.

To prepare the data for wind power prediction, we implemented several preprocessing steps to ensure data quality and reliability. Initially, missing values in the dataset were filled with zeros. Subsequently, we identified and removed abnormal wind turbine states based on SCADA data and domain knowledge. A wind turbine was considered abnormal if it exhibited any of the following conditions: zero or negative active power output; zero or negative wind speed; pitch angle outside the normal range; rotational speed outside the normal range; temperature inside the turbine nacelle above a certain threshold; or any other fault indicators reported by the SCADA system.

Feature selection and engineering were performed to identify the most relevant and influential features for wind power prediction. We utilized feature correlation heatmaps and feature importance plots based on SHAP values to analyze the relationship between each feature and the active power output. Fig. 1 illustrates an example of these plots for one of the selected wind turbines. Based on this analysis, we selected four features as input variables for our prediction models: wind speed (*Wspd*), pitch angle (*Pab*), reactive power (*Prtv*), and active power (*Patv*). We excluded four features that showed low correlation

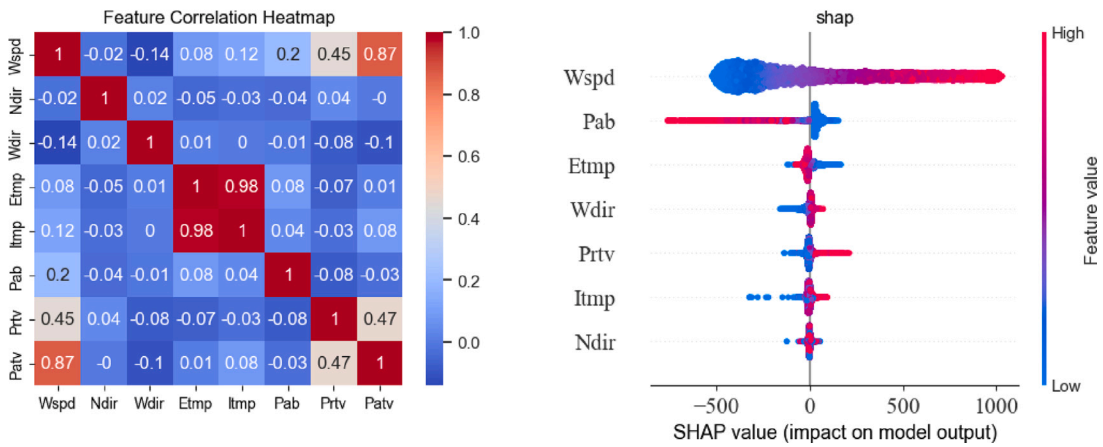

Fig. 1. Feature correlation and importance analysis for the selected dataset. The left plot shows a heatmap of the Pearson correlation coefficients between features. The right plot shows the SHAP values of each feature for the active power output. The SHAP values measure the contribution of each feature to the prediction, taking into account the interactions with other features. The features are ranked by their absolute mean SHAP value, with the most important features at the top. The color of the SHAP value indicates whether the feature has a positive (red) or negative (blue) impact on the prediction. For example, a high wind speed (*Wspd*) increases the active power output (*Patv*), while a high pitch angle (*Pab*) decreases it. These plots help us understand which features are more relevant and influential for wind power prediction.

or impact on the active power output: ambient temperature (*Etmp*), temperature inside the turbine nacelle (*Itmp*), angle between wind direction and nacelle position (*Wdir*), and nacelle direction (*Ndir*).

The dataset was further divided into different seasons based on the seasonality of the wind speed feature. We observed higher wind speeds in spring and winter and lower values in summer. Consequently, we partitioned the dataset into four seasons: spring (March–May), summer (June–August), autumn (September–November), and winter (December–February). This stratification helped reduce data interference and improve model fitting and training.

Through these preprocessing steps, we aimed to enhance the quality and reliability of the data, thereby improving the accuracy and robustness of our wind power prediction models.

4. Methodology

In this section, we first formulate the research problem of online prediction of wind power production as a time series prediction task. We then present the methodology of the proposed EFM+ for online wind power prediction in details.

4.1. Problem formulation

This paper approaches the online wind power prediction problem as a time series prediction task. This approach is particularly suitable due to the inherent characteristics of wind power data. Firstly, wind power generation exhibits strong temporal dependencies, meaning that past generation patterns significantly influence future values. Time series prediction models excel at capturing these complex dependencies to make informed forecasts. Secondly, the online nature of the prediction task necessitates models that can process sequential data as it arrives and provide updated forecasts in real time, aligning perfectly with the capabilities of time series prediction methods.

In our formulation, we aim to predict wind power generation at a 10-min horizon. This means that given a historical time series of wind power data, our model will forecast the power output for the next 10 min. More formally, let:

- y_i represent the ith data point in the wind power generation time series, which is the target variable we aim to predict.
- x_i be the ith sample containing relevant explanatory variables (features) used for prediction.
- d denote the dimension or length of each data point or pattern. For example, $d = 1$ for individual 10-min readings, while $d = 144$ would represent a daily pattern consisting of 144 ten-minute readings.
- k indicate the current time step or index in the time series.

Given an energy generation time series $y_1, y_2, \ldots, y_k$, where k is the length of the time series and each point $y_i \in \mathbb{R}^d$, at time k, we use a previous window of length d as input for prediction – i.e., $y_{k-d}, \ldots, y_{k-1}$. Using our chosen prediction algorithms, we predict the next value y_k in the sequence. The error between the actual and predicted value at time k, denoted as $err_k = |y_k - y'_k|$, where y'_k is the predicted value, serves as a measure of the prediction model's accuracy.

4.2. Overview of proposed model

The EFM+ is an innovative ensemble model that combines predictions from multiple models using a dynamic weighting scheme based on their current performance. This design allows the EFM+ to exploit the distinct advantages of each model and adapt to changing situations, delivering accurate and timely forecasts. The EFM+ introduces time-dependent model factors for nonstationary data. This feature enables the weights assigned to different models to evolve over time based on changes in the prediction environment. As a result, the EFM+ can effectively adapt to nonstationarity in wind power generation and capture complex relationships that vary over time. The time-dependent factors are data-driven, updating based on the recent performance of each model. The EFM+ is designed to provide short-term forecasts for intra-hour scheduling. It provides predictions every 10 min using the most recent 10 min of wind power generation data. This feature enables the EFM+ to detect rapid changes and fluctuations in wind power generation and provide timely and accurate forecasts for intra-hour scheduling decisions.

Fig. 2 shows the overall structure and workflow of the EFM+. The EFM+ consists of six main components, each of which plays a specific role in the online wind power prediction process. The data sources and preprocessing component is responsible for collecting and cleaning the wind farm data from the SCADA system and extracting relevant features for prediction, such as wind speed, pitch angle, reactive power, and active power. The XGBoost algorithm and the self-attention LSTM algorithm are two individual models that generate predictions based on the input features. The XGBoost algorithm is a

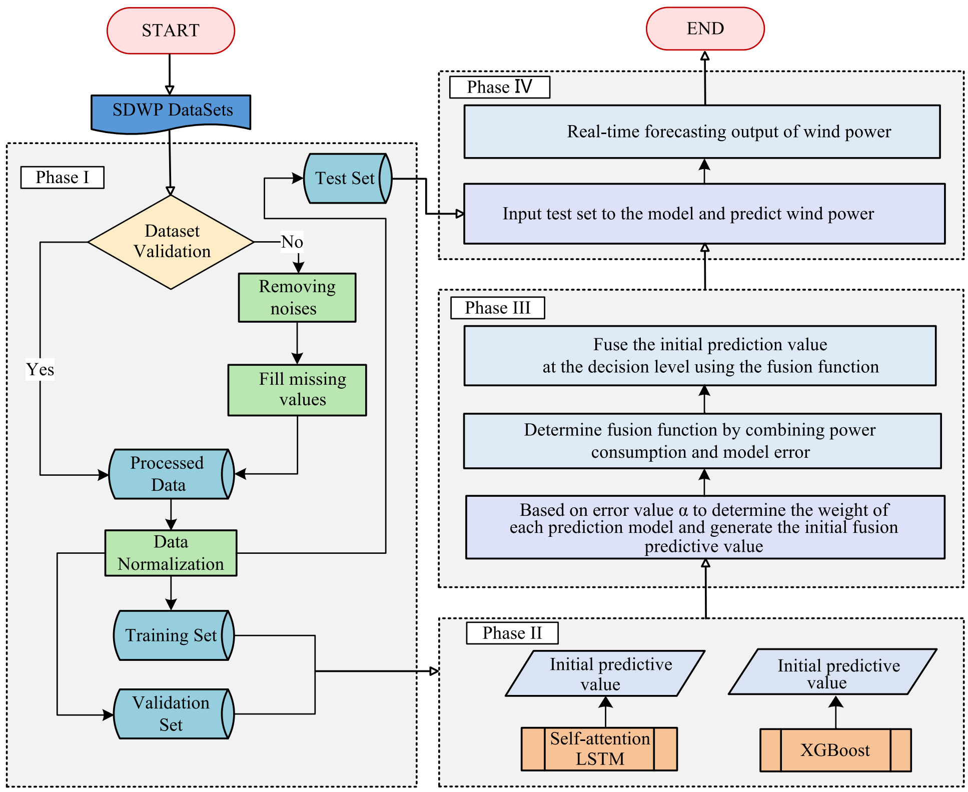

Fig. 2. Overview of the proposed EFM+.

gradient boosting decision tree model that can capture nonlinear and complex relationships between the features and the output. The self-attention LSTM algorithm is a recurrent neural network model that can learn temporal dependencies and patterns in the time series data using a self-attention mechanism. The online learning component is responsible for updating the models with new data as it becomes available and adjusting their parameters accordingly. This component enables the models to adapt to changing situations and improve their performance over time. The recurrent batch normalization component is responsible for normalizing the inputs to the inner layers of the models to improve their convergence and generalization. This component helps to reduce the internal covariate shift problem and stabilize the training process of the models. The adaptive forecasting component is responsible for combining the predictions from the individual models using a dynamic weighting scheme that depends on their current performance and error. This component allows the EFM+ to exploit the distinct advantages of each model and select the best combination of weights for each prediction situation. The final output of the EFM+ is a short-term forecast of wind power generation for intra-hour scheduling, which can be used by grid operators and energy traders to optimize their decisions and operations.

4.3. Individual models for wind power prediction

In this subsection, we describe the two individual models that we use in the EFM+ to generate predictions based on the input features. These models are XGBoost and self-attention LSTM, which are both state-of-the-art machine learning algorithms that can handle complex and nonlinear data. We explain the rationale behind choosing these models (the selection of the two models will also be supported by the experiments in Section 5.5), their main characteristics and advantages, and the formalization of the models.

4.3.1. XGBoost algorithm

XGBoost is an optimized distributed gradient boosting library designed for efficient and scalable training of machine learning models. It is an ensemble learning method that combines the predictions of multiple weak models to produce a stronger prediction. XGBoost stands for "Extreme Gradient Boosting" and it has become one of the most popular and widely used machine learning algorithms due to its ability to handle large datasets and its ability to achieve state-of-the-art performance in many machine learning tasks such as classification and regression. XGBoost is suitable for wind power prediction because it can capture the complex and non-linear relationships between wind power generation and various factors such as weather conditions, turbine states, seasonal patterns, and stochastic fluctuations. XGBoost also has several advantages over other machine learning algorithms, such as high accuracy, fast speed, scalability, and regularization.

The main idea of XGBoost is to construct a boosting tree model that learns a new function and its coefficients to fit the residuals of the previous step of the prediction. Boosting tree models are a type of ensemble learning methods that combine multiple decision trees to improve the accuracy and generalization ability of the model. The basic principle of boosting tree models is to iteratively add new trees that correct the errors made by the previous trees, until a predefined stopping criterion is met. Each tree is trained on a weighted version of the training data, where the weights are updated based on the prediction errors of the previous trees. This way, each tree focuses more on the difficult examples that are hard to predict by the previous trees. XGBoost uses a forward addition strategy where each time the model adds a decision tree, it learns a new function and its coefficients to fit the residuals of the last step of the prediction. Therefore, using the model at step m as an example, the prediction for the ith sample x_i is as Eq. (1) and the object function of step m is as Eq. (2).

$$y_i^m = y_i^{m-1} + f_m\left(x_i\right) \tag{1}$$

$$\mathrm{Obj}^{(m)}=\sum_{i=1}^{n} L\left(y_i, y^{m-1}+f_m\left(x_i\right)\right)+\Omega\left(f_m\right) \tag{2}$$

where y_i^m is the predicted value of x_i at step m, f_m is the decision tree added at step m, L is the loss function that measures the difference between the actual and predicted values, and Ω is the regularization term that penalizes the complexity of the model.

To simplify the objective function, XGBoost uses Taylor expansion to approximate it with a second-order polynomial function. This allows XGBoost to use only the first and second derivatives of the loss function in the calculation process, which reduces the computational cost and improves the efficiency of the algorithm. The objective function can be rewritten as Eq. (3).

$$\mathrm{Obj}^{(m)}\approx\sum_{i=1}^{n}\left[L\left(y_i, y^{m-1}\right)+g_i f_m\left(x_i\right)+\frac{1}{2}h_i f_m^2\left(x_i\right)\right]+\Omega\left(f_m\right) \tag{3}$$

where g_i represents the first derivative of $L\left(y_i, y^{m-1}\right)$ with respect to y^{m-1}, and h_i represents the second derivative of $\left(y_i, y^{m-1}\right)$ with respect to y^{m-1}. In addition, XGBoost merges the same function values on the same leaf node to reduce redundancy. The optimal weight for each leaf node can be calculated as Eq. (4), and the objective function can be simplified as Eq. (5).

$$w_j=-\frac{G_j}{H_j+\lambda} \tag{4}$$

$$\mathrm{Obj}^{(m)}=-\frac{1}{2}\sum_{j=1}^{T}\frac{G_j^2}{H_J+\lambda}+\gamma T+C \tag{5}$$

where w_j is the weight of the jth leaf node, G_j and H_j are the sum of the first and second derivatives of the loss function for all samples on the jth leaf node, λ is the L2 regularization parameter, γ is the complexity penalty parameter, T is the number of leaf nodes, and C is a constant.

The selection and tuning of XGBoost parameters are also crucial, as they can significantly impact the performance of the algorithm. Some of the important parameters are max_depth, which controls the maximum depth of each decision tree; learning_rate, which controls the learning rate or step size for each iteration; n_estimators, which controls the number of decision trees to be added; and subsample, which controls the fraction of samples to be used for each tree. Choosing the optimal values for these parameters requires careful analysis and evaluation, as different values may have different effects on the accuracy and speed of the algorithm. Some methods or criteria for choosing the optimal values for these parameters are cross-validation, grid search, or Bayesian optimization.

4.3.2. Self-attention LSTM algorithm

The self-attention LSTM algorithm (Cao, Lei, Li, Zhang, & Duan, 2022; Katrompas & Metsis, 2022; Xia, Feng, Lu, Fei, & Xue, 2021) is a hybrid model that combines the advantages of LSTM networks and self-attention mechanisms for time series prediction. We choose this algorithm because it can model both long-term and short-term dependencies in the wind power generation data by using a memory cell to store and update information over time. Moreover, this algorithm can focus on the most relevant parts of the input sequence and capture complex relationships that vary over time by using a self-attention mechanism that relates different positions of the sequence to each other. Self-attention can also learn from multiple levels of representation and handle long-range dependencies more effectively than LSTM networks.

The self-attention LSTM algorithm consists of two main components: an LSTM encoder and a self-attention decoder. The LSTM encoder takes the input sequence as a series of vectors and encodes it into a hidden state vector that summarizes the information of the sequence. The self-attention decoder takes the hidden state vector as input and generates an output vector that represents the prediction of the next value in the sequence. The self-attention decoder uses a query-key–value mechanism to compute an attention score for each position in the input sequence, which reflects how much each position contributes to the prediction. The self-attention layer also outputs a context vector that is a weighted sum of the input vectors, where the weights are determined by the attention scores. The context vector is then concatenated with the hidden state vector and passed through a fully connected layer to produce the output vector. The formulation of the self-attention LSTM algorithm is as follows:

Given an input sequence $x=(x_1,x_2,\ldots,x_k)$, where $x_t\in\mathbb{R}^d$ is a d-dimensional feature vector at time step t, the LSTM encoder computes the hidden state vector $h_t\in\mathbb{R}^h$ for each time step t by using the following equations:

$$f_t=\sigma\left(W_f x_t+U_f h_{t-1}+b_f\right) \tag{6}$$

$$i_t=\sigma\left(W_i x_t+U_i h_{t-1}+b_i\right) \tag{7}$$

$$o_t=\sigma\left(W_o x_t+U_o h_{t-1}+b_o\right) \tag{8}$$

$$c_t=f_t\odot c_{t-1}+i_t\odot\tanh\left(W_c x_t+U_c h_{t-1}+b_c\right) \tag{9}$$

$$h_t=o_t\odot\tanh\left(c_t\right) \tag{10}$$

where f_t, i_t, and o_t are the forget gate, input gate, and output gate vectors, respectively; c_t is the cell state vector; σ is the sigmoid activation function; $\odot$ is the element-wise multiplication operator; W, U, and b are learnable parameters.

The self-attention decoder takes the last hidden state vector h_k as input and generates an output vector $y_{k+1}\in\mathbb{R}$ that represents the prediction of the next value in the sequence. The self-attention decoder uses a query-key–value mechanism to compute an attention score e_{mj} for each position j in the input sequence, which reflects how much position j contributes to position $k+1$. The attention score is calculated by using a learned linear transformation as follows:

$$e_{mj}=v^{\top}\tanh\left(W_1 h_j+W_2 h_k+b_1\right) \tag{11}$$

where v, W_1, W_2, and b_1 are learnable parameters.

The attention score is then normalized by using a softmax function to obtain an attention weight a_{mj} for each position j in the input sequence, which represents how much attention should be paid to position j. The attention weight is calculated as follows:

$$a_{mj}=\frac{\exp\left(e_{mj}\right)}{\sum_{k=1}^{T}\exp\left(e_{mk}\right)} \tag{12}$$

The self-attention layer then outputs a context vector $c_{k+1}\in\mathbb{R}^h$ that is a weighted sum of the hidden state vectors of the input sequence, where the weights are determined by the attention weights a_{mj}. The context vector is calculated as follows:

$$c_{k+1}=\sum_{j=1}^{T} a_{mj}h_j \tag{13}$$

The context vector is then concatenated with the last hidden state vector h_k and passed through a fully connected layer to produce the output vector y_{k+1}. The output vector is calculated as follows:

$$y_{k+1}=W_3[h_k;c_{k+1}]+b_2 \tag{14}$$

where W_3 and b_2 are learnable parameters.

The objective function of the self-attention LSTM algorithm is defined as the mean squared error (MSE) between the actual and predicted values of the output sequence, which measures the accuracy of the prediction model. The objective function is optimized by using the Adam optimizer, which is a stochastic gradient descent method that adapts the learning rate for each parameter based on the first and second moments of the gradients. The Adam optimizer can handle sparse gradients and noisy data, and achieve fast convergence and robust performance.

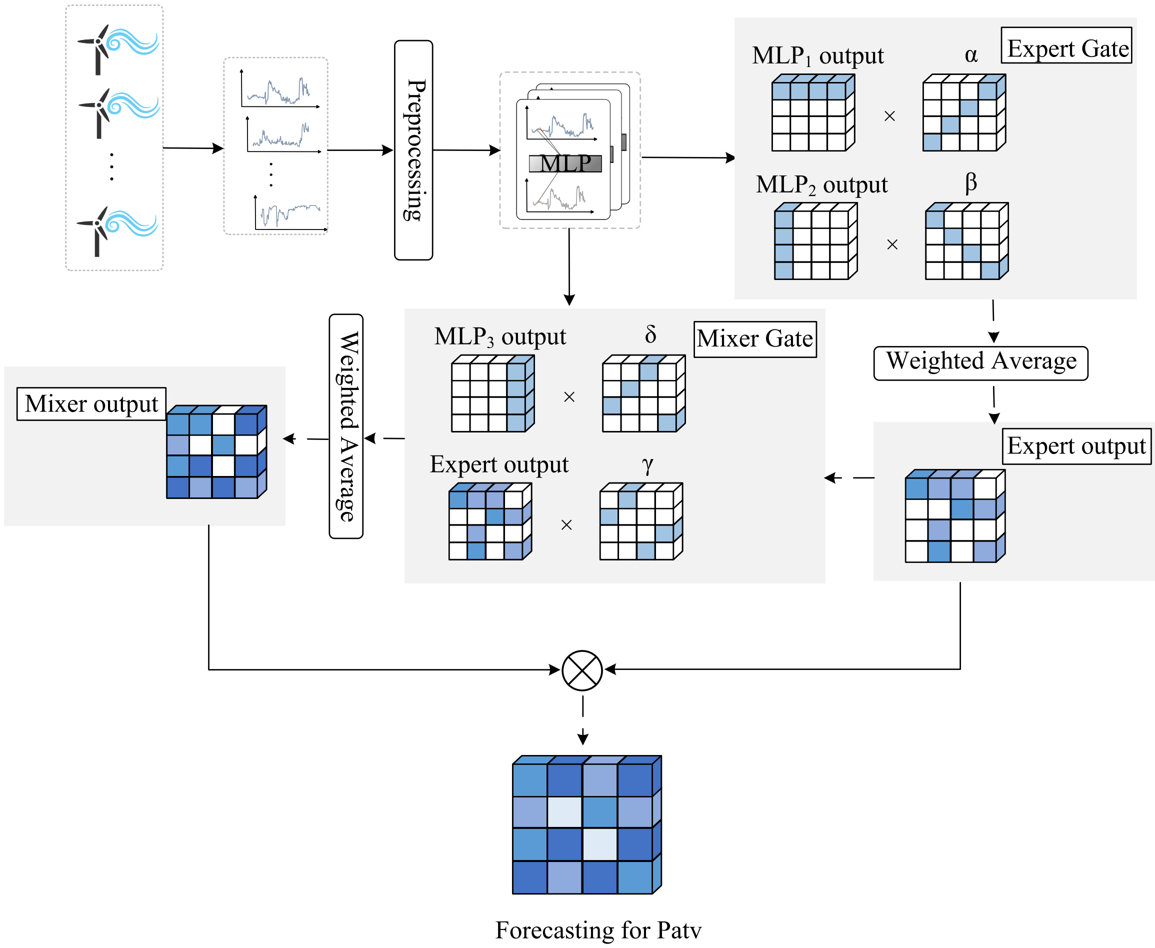

Fig. 3. The structure and workflow of the EFM+.

4.4. Adaptive expert fusion model (EFM+)

The EFM+ is an innovative ensemble model that dynamically combines the predictions from multiple models based on their current performance. The model ($\mathcal{F}$) consists of five components: data sources and preprocessing, XGBoost algorithm, self-attention LSTM algorithm, online learning, and adaptive forecasting. Fig. 3 illustrates the structure and workflow of the EFM+.

The data sources and preprocessing component collects and cleans the wind farm data and extracts relevant features for prediction, such as wind speed, pitch angle, reactive power, etc. The data is divided into training and testing sets using a sliding window method, where the window size is determined by the prediction horizon. For example, if the prediction horizon is one hour, the window size is 144, which means each input sequence contains 144 10-min readings.

The XGBoost algorithm ($\mathcal{X}$) and the self-attention LSTM algorithm ($\mathcal{L}$) are two individual models that generate predictions based on input features. Both models are trained on the training set and tested on the testing set. The XGBoost algorithm uses a boosting tree model that learns a new function and its coefficients to fit the residuals of the previous step of the prediction. The self-attention LSTM algorithm uses a hybrid model that combines the advantages of LSTM networks and self-attention mechanisms for time series prediction. Both models can capture complex non-linear relationships and temporal dependencies in the wind power generation data.

The online learning component updates the models with new data as it becomes available and adjusts their parameters accordingly. The online learning component uses a stochastic gradient descent (SGD) algorithm to minimize a loss function that measures the error between the actual and predicted values. The loss function can be either mean squared error (MSE) or mean absolute error (MAE), depending on user preference. This online learning component enables the models to improve their accuracy over time and adapt to changing conditions.

The adaptive forecasting component combines the predictions from the individual models based on their current performance and assigns different weights to them. The adaptive forecasting component uses an expert gate and a mixer gate, which are two multi-layer perceptrons (MLPs) that learn how to select and combine the outputs of the individual models. The expert gate determines the weight of each individual model based on its prediction error, while the mixer gate determines the weight of each expert gate output based on its similarity to the input sequence. The final prediction is a weighted average of the expert gate outputs, where the weights are determined by the mixer gate. The adaptive forecasting component can achieve better accuracy and robustness than the individual models by leveraging their complementary strengths and weaknesses.

The vector of weighted features ($\mathcal{V}$) is a component that enhances the input features by applying a fully connected layer with a softmax activation function on them. The vector of weighted features assigns different weights to the input features based on their relevance and importance for the prediction. The vector of weighted features can capture the nonlinear and dynamic relationships among the input features and improve the representation of the input sequence.

The final output of $\mathcal{F}$ is a short-term forecast of wind power generation for intra-hour scheduling.

$$X_t' = \mathcal{F}(X_t) = \mathcal{A}(X_t) \tag{15}$$

where X_t is a vector of input features at time t, such as wind speed, pitch angle, reactive power, etc., and X'_t is the predicted value of wind power generation at time t.

We can express $\mathcal{F}$ as a composition of three sub-functions: $\mathcal{X}(X_t)$, $\mathcal{L}(X_t)$, and $\mathcal{V}(X_t)$, which represent the predictions by the XGBoost algorithm, the self-attention LSTM algorithm, and the vector of weighted feature component respectively. Then, we have:

$$\mathcal{F}(X_t) = \mathcal{A}(X_t) = \mathcal{M}(\mathcal{E}(\mathcal{X}(X_t), \mathcal{L}(X_t)), \mathcal{V}(X_t)) \tag{16}$$

We can further decompose $\mathcal{A}$ into two sub-functions: $\mathcal{E}(X_t)$ and $\mathcal{M}(X_t)$, which represent the expert gate and the mixer gate respectively. Then, we have:

$$\mathcal{A}(X_t) = \mathcal{E}(\mathcal{X}(X_t), \mathcal{L}(X_t)) + \mathcal{M}(\mathcal{V}(X_t)) \tag{17}$$

where $\mathcal{E}$ and $\mathcal{M}$ are weighted sums of their inputs. This shows that $\mathcal{A}$ combines the predictions from the individual models and the weighted features in a nonlinear way.

Let α_t, β_t, γ_t, and δ_t be the weights assigned to $\mathcal{X}$, $\mathcal{L}$, $\mathcal{E}$, and $\mathcal{M}$ respectively at time t. These weights are time-varying parameters that depend on the input features and the prediction errors of each model. They are updated in real-time as new data arrives, allowing for flexible Bayesian predictive inference and adaptation to nonstationarity. The expert gate and the mixer gate can be defined as follows:

$$\mathcal{E}(\mathcal{X}(X_t), \mathcal{L}(X_t)) = \alpha_t \mathcal{X}(X_t) + \beta_t \mathcal{L}(X_t) \tag{18}$$

$$\mathcal{M}(\mathcal{E}(\mathcal{X}(X_t), \mathcal{L}(X_t)), \mathcal{V}(X_t)) = \gamma_t \mathcal{E}(\mathcal{X}(X_t), \mathcal{L}(X_t)) + \delta_t \mathcal{V}(X_t) \tag{19}$$

where α_t, β_t, γ_t, and δ_t are dynamic weights computed by using a fully connected layer with a softmax activation function on the inputs and outputs of each model and $\mathcal{V}$.

The parameter γ_t is introduced to control the trade-off between the expert gate output and the vector of weighted features. The parameter γ_t reflects the confidence and reliability of the expert gate output, which may vary depending on the performance and error of the individual models. If the expert gate output is more accurate and consistent than the vector of weighted features, the parameter γ_t will be higher, and vice versa. The parameter γ_t allows the adaptive forecasting component to balance the contributions of the expert gate output and the vector of weighted features and achieve the optimal prediction.

To make $\mathcal{F}$ suitable for online prediction, which means predicting the next value in the sequence as soon as new data arrives, we use an online learning algorithm such as stochastic gradient descent (SGD) to update the parameters of each individual model and $\mathcal{F}$ based on new data. This is different from offline prediction, which means predicting the next value in the sequence after collecting and processing all the data. The online learning algorithm can be written as:

$$\theta_{t+1} = \theta_t - \eta \nabla L(X_t, X'_t) \tag{20}$$

where θ_t is the vector of parameters of each individual model and $\mathcal{F}$ at time t, η is the learning rate, and $L(X_t, X'_t)$ is the loss function that measures the error between the actual and predicted values. The gradient operator ∇ computes the partial derivatives of the loss function with respect to each parameter, which indicate the direction and magnitude of the steepest descent. The gradient operator can be calculated as:

$$\nabla L(X_t, X'_t) = \left(\frac{\partial L}{\partial \theta_1}, \frac{\partial L}{\partial \theta_2}, \dots, \frac{\partial L}{\partial \theta_n} \right) \tag{21}$$

where θ_i is the ith parameter in the vector θ_t and n is the number of parameters.

4.5. Dynamic ensemble for active online forecasting

In this subsection, we explain how we employ dynamic ensemble to achieve adaptive online forecasting. EFM+ combines the predictions of multiple individual models using a weighted average method based on their current performance and error, which can adapt to concept drift and nonstationarity in wind power generation and weather conditions by updating the weights of the individual models at each time step using a subset-based error calculation.

Let $\{\mathcal{D}_t\}_{t=1}^{+\infty}$ be the stream data, where $\mathcal{D}_t = \{\mathrm{x}_{t,i}, y_{t,i}\}_{i=1}^{N_t}$ denotes the training data at time step t and $(\mathrm{x}_{t,i}, y_{t,i})$ are generated i.i.d. from the density $P_t(X, y)$. Here, $\mathrm{x}_{t,i} \in \mathbb{R}^f$ represents the data features of the ith wind turbine at time step t, and $y_{t,i} \in \mathbb{R}$ is the actual value of wind power generation. We assume that we have a pool of M individual models $\{F_m\}_{m=1}^{M}$ that are trained on the same dataset using different methods, such as XGBoost and self-attention LSTM. We denote the prediction of model F_m for sample x_t as $F_m(\mathrm{x}_t)$. We also assume that each model has a memory length of l, which means that it uses the previous l values of $\mathrm{x}_{t,i}$ to predict $y_{t,i}$. Therefore, we can write:

$$y_{t,i} = P_t(X_{t-l+1,i}, \dots, X_{t,i}) \tag{22}$$

$$F_m(x_t) = F_m(X_{t-l+1,i}, \dots, X_{t,i}) \tag{23}$$

This way, we can account for the temporal dependency of wind power production on previous data features.

To update the weight of each model at each time step, we use a dynamic weighting scheme based on their current performance and error. We define the weight of model F_m at time step t as $w_{m,t}$ and we calculate it as follows:

$$w_{m,t} = \frac{\exp\left(-\alpha \cdot e_{m,t}\right)}{\sum_{m=1}^{M} \exp\left(-\alpha \cdot e_{m,t}\right)} \tag{24}$$

where α is a hyperparameter that controls the sensitivity of the weight to the error and $e_{m,t}$ is the absolute error of model F_m at time step t, defined as:

$$e_{m,t} = |y_t - F_m(x_t)| \tag{25}$$

The weight reflects the confidence or competence of each model for predicting sample x_t. The higher the weight, the more confident or competent the model is. The weight is updated based on the error of each model on a subset of similar samples from the training set. We use a k-nearest neighbor method to find a subset of k samples from $\mathcal{D}_{t-1}$ that are closest to x_t based on some similarity measure, such as Euclidean distance. We denote this subset as $\mathcal{N}_k(\mathrm{x}_t)$. We then calculate the error of each model on this subset as:

$$e_{m,t} = \frac{1}{k} \sum_{(\mathrm{x},y) \in \mathcal{N}_k(\mathrm{x}_t)} |y - F_m(x)| \tag{26}$$

By using this subset-based error, we can capture the local performance of each model around sample x_t and assign higher weights to models that perform better on similar samples.

To combine the predictions of each model using a dynamic ensemble technique, we use a weighted average method based on their weights. The final prediction for sample x_t is given by:

$$F_t(x_t) = \sum_{m=1}^{M} w_{m,t} F_m(x_t) \tag{27}$$

This prediction is expected to be more accurate and robust than any single model or a static ensemble technique that uses fixed or equal weights.

By updating the weights and combining the predictions in this way, we can achieve dynamic and adaptive forecasting that can handle concept drift and nonstationarity in wind power prediction. Concept drift refers to the phenomenon that the underlying distribution of the data changes over time, which may cause the performance of the prediction models to deteriorate. For example, the wind speed and direction may vary due to seasonal changes, weather events, or environmental factors. The dynamic ensemble technique can exploit the advantages

Algorithm 1: Adaptive expert fusion model for online forecasting

Result: Dynamic Ensemble Technique for Online Wind Power Prediction
Data: A pool of M individual models $\{F_m\}_{m=1}^{M}$, a stream of data $\{\mathcal{D}_t\}_{t=1}^{+\infty}$, a hyperparameter α, and a similarity measure S

```
for t = 1 to +∞ do
    Receive a new sample x_t and its actual value y_t from the stream data;
    for m = 1 to M do
        Generate a prediction for sample x_t using model F_m: F_m(x_t);
        Find a subset of k samples from D_{t-1} that are closest to x_t based on S: N_k(x_t);
        Calculate the error of model F_m on the subset:
            e_{m,t} = (1/k) Σ_{(x,y)∈N_k(x_t)} |y − F_m(x)|;
        Update the weight of model F_m based on the error:
            w_{m,t} = exp(−α·e_{m,t}) / Σ_{m=1}^{M} exp(−α·e_{m,t});
    end
    Combine the predictions of each model using a weighted average method based on their weights:
        F_t(x_t) = Σ_{m=1}^{M} w_{m,t} F_m(x_t);
    Evaluate the prediction error for sample x_t: e_t = |y_t − F_t(x_t)|;
    Update the parameters and learn from sample x_t using online learning and recurrent batch normalization methods;
end
```

of different models and adjust their weights according to their current performance and error, thereby improving the accuracy and robustness of the expert fusion model for online wind power prediction. Algorithm 1 summarizes the dynamic ensemble technique for adaptive online forecasting.

4.6. Complexity analysis

This subsection analyzes the computational complexity of the EFM+ model, considering both training and prediction phases.

4.6.1. Training complexity

The training phase involves training the individual XGBoost ($\mathcal{X}$) and self-attention LSTM ($\mathcal{L}$) models, as well as updating the parameters of the expert gate ($\mathcal{E}$), mixer gate ($\mathcal{M}$), and vector of weighted features ($\mathcal{V}$).

- **XGBoost** ($\mathcal{X}$)**:** The training complexity of XGBoost is dominated by the construction of boosting trees. For a dataset with n samples and p features, the complexity is approximately $O(np\log n)$ per tree. With T trees, the overall complexity becomes $O(Tnp\log n)$.
- **Self-attention LSTM** ($\mathcal{L}$)**:** The complexity of training an LSTM network is $O(Wdh)$, where W is the number of weights, d is the input dimension, and h is the hidden state dimension. The self-attention mechanism adds a complexity of $O(n^2d)$, where n is the sequence length. Therefore, the overall complexity is $O(Wdh + n^2d)$.
- **Expert Gate** ($\mathcal{E}$), **Mixer Gate** ($\mathcal{M}$), and Vector of Weighted Features ($\mathcal{V}$)**:** These components are implemented using fully connected layers, which have a complexity of $O(I \cdot O)$, where I is the input dimension and O is the output dimension. Since these dimensions are relatively small compared to the input data size, their training complexity is negligible compared to $\mathcal{X}$ and $\mathcal{L}$.

The overall training complexity of EFM+ is dominated by the XGBoost and self-attention LSTM models, resulting in a complexity of $O(Tnp\log n + Wdh + n^2d)$.

4.6.2. Prediction complexity

The prediction phase involves generating predictions from $\mathcal{X}$ and $\mathcal{L}$, applying $\mathcal{V}$, and combining the results using $\mathcal{E}$ and $\mathcal{M}$.

- **XGBoost** ($\mathcal{X}$)**:** The prediction complexity of XGBoost is $O(T\log n)$ per sample, where T is the number of trees.
- **Self-attention LSTM** ($\mathcal{L}$)**:** The prediction complexity of the LSTM network is $O(Wdh)$, and the self-attention mechanism adds a complexity of $O(n^2d)$. Therefore, the overall prediction complexity is $O(Wdh + n^2d)$.
- **Expert Gate** ($\mathcal{E}$), **Mixer Gate** ($\mathcal{M}$), **and Vector of Weighted Features** ($\mathcal{V}$)**:** Similar to the training phase, the prediction complexity of these components is $O(I \cdot O)$ and is negligible compared to $\mathcal{X}$ and $\mathcal{L}$.

The overall prediction complexity of EFM+ is dominated by the individual models, resulting in a complexity of $O(T\log n + Wdh + n^2d)$ per sample. Since the prediction is performed online for each new data point, the complexity remains manageable for real-time applications.

5. Experiments

5.1. Experimental settings

We implemented all models using the Tensorflow v2.8.0 framework, and the corresponding code is available at https://github.com/22Jingtong/EFM. We conducted the experiments on a desktop computer with a 12th Gen Intel(R) Core(TM) i7-12700 2.10 GHz processor, 16G memory, and a NVIDIA RTX 3060Ti 8 GB GPU.

5.2. Data preparation

To prepare the data for the prediction models, we applied the following steps:

- We used a sliding window approach to generate samples for each turbine every 10 min, increasing the amount of training data and capturing the temporal dependencies of the historical data.
- We filled the missing or invalid values in the data with the respective preceding values for both training and inference purposes, avoiding the introduction of bias or distortion in the data.
- We used a min–max scaler to standardize the features to a range between zero and one, ensuring that the model can better learn the relationships between features with different scales and units.

To illustrate the effect of data preprocessing, we took turbine No. 28 as an example. Fig. 4 shows the curves for wind power data before and after preprocessing. We can see that after preprocessing, the data became smoother and more consistent, reducing the fluctuations and outliers. We also applied a k-means clustering method to divide the data into different categories based on their wind speed and direction. This step aimed to capture the heterogeneity and diversity of the wind power generation under different wind conditions. We chose the number of clusters as 5, as it was the optimal value according to the elbow curve shown in Fig. 5(a). The elbow curve plots the sum of squared errors (SSE) against the number of clusters, and shows that SSE decreases rapidly until 5 clusters, and then decreases slowly afterwards. Fig. 5(b) shows the distribution of wind power data in each cluster. We labeled each cluster as breeze (Cluster 0), moderate wind (Cluster 3), cool wind (Cluster 2), strong wind (Cluster 1), and powerful wind (Cluster 4), based on their average wind speed and direction. To measure the effects of wind speed and direction on wind power, we calculated the Pearson correlation coefficient between them before and

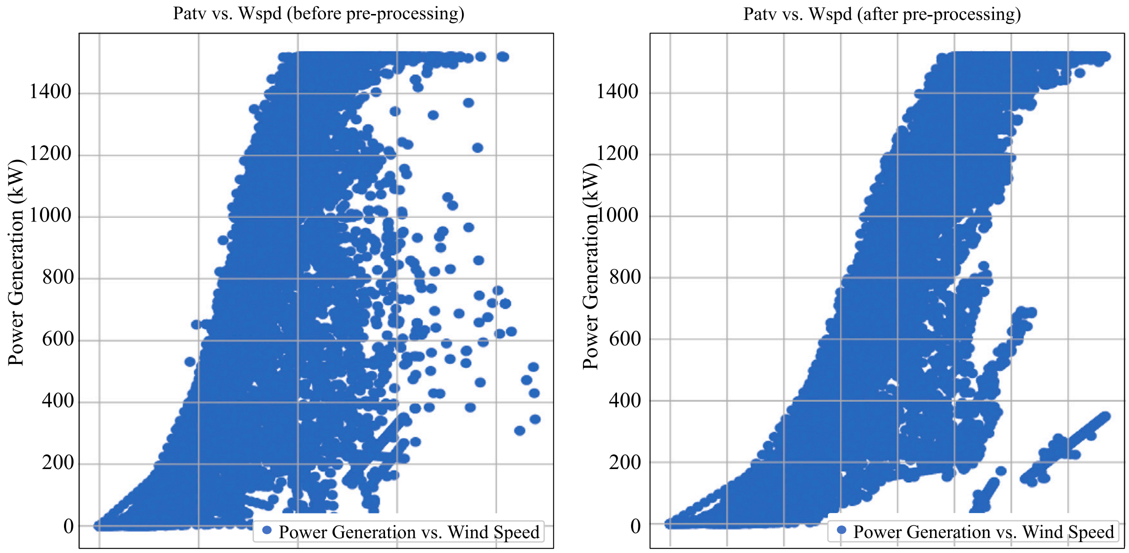

Fig. 4. Curves for wind power data: (Left) before pre-processing and (Right) after pre-processing.

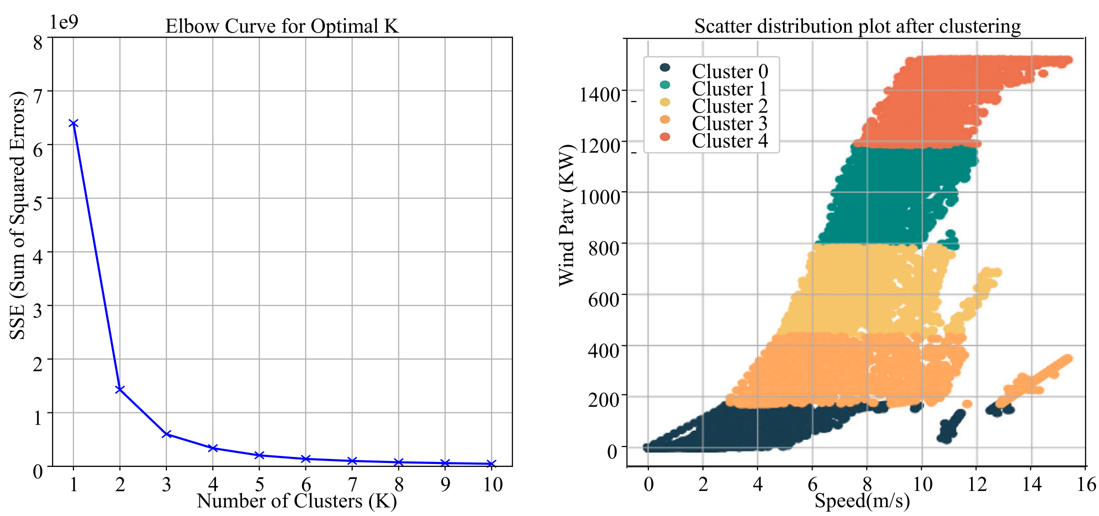

Fig. 5. Data Clustering Operation: (Left)The elbow curve for collected wind power data and (Right)Wind power data classification using a k-means method.

Table 1
Pearson correlation coefficient values between weather variables and wind power.

Variable	Before pre-processing				After pre-processing			
	Wspd	Pab	Prtv	Patv	Wspd	Pab	Prtv	Patv
Wspd	1.00	−0.12	0.57	0.89	1.00	0.08	0.47	0.92
Pab	−0.12	1.00	−0.28	−0.26	0.08	1.00	−0.09	−0.04
Prtv	0.57	−0.28	1.00	0.57	0.47	−0.09	1.00	0.45
Patv	0.89	−0.26	0.57	1.00	0.92	−0.04	0.45	1.00

after data preprocessing. Table 1 displays the correlation coefficient values for the turbine. We can see that after preprocessing, there is a noticeable increase in the correlation coefficient between wind speed and wind power, indicating a stronger correlation. There is also a slight decrease in the correlation coefficient between wind speed, *Pab*, and *Prtv*, indicating a weaker influence of these auxiliary variables on the prediction of power generation.

5.3. Evaluation metrics

We used four metrics to evaluate the accuracy of the wind power prediction models: Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE) and a final score. These metrics quantify the average magnitude of the errors, the square root of the average squared errors, and the average ratio of the errors to the actual values, respectively. They are suitable for wind power prediction because they can capture different aspects of prediction accuracy and error, such as bias, variance, and scale.

The final score is the average of RMSE and MAE over all wind turbines. The lower the values of these metrics, the more accurate the model's prediction performance. The metrics are defined as follows:

- **MAE**: The average absolute difference between the predicted and actual values.

$$\mathbf{MAE} = \frac{1}{C}\sum_{c=1}^{C}\frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \tag{28}$$

- **MAPE**: The average ratio of the absolute difference between the predicted and actual values to the actual values.

$$\mathbf{MAPE} = \frac{1}{C}\sum_{c=1}^{C}\frac{1}{N}\sum_{i=1}^{N}\frac{|y_i - \hat{y}_i|}{y_i} \tag{29}$$

- **RMSE**: The square root of the average squared difference between the predicted and actual values.

$$\textbf{RMSE} = \frac{1}{C}\sum_{c=1}^{C}\sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \tag{30}$$

- **Score**: The average of RMSE and MAE over all wind turbines.

$$\textbf{Score} = \frac{1}{2}\frac{1}{C}\sum_{c=1}^{C}\sum_{k=0}^{K}\sum_{i=1}^{W-1} RMSE^{i}_{T(k)} + MAE^{i}_{T(k)} \tag{31}$$

where C is the number of clusters, N is the number of samples in each cluster, K is the number of instances to evaluate the model with, T is a set of K timestamps to predict, and W is the window size for prediction. We calculated these metrics separately for each cluster, and then averaged them.

5.4. Comparison with baselines

In this section, we compare the proposed model with several state-of-the-art methods from different categories. We describe these methods below and report their performance on the wind power prediction task. We also evaluate the effect of the dynamic ensemble module by comparing the proposed model with and without this module. We use *EFM* to denote the model without the dynamic ensemble module, and *EFM+* to denote the model with the dynamic ensemble module. In the following subsections, we will report the results of both EFM and EFM+ for all the experiments.

The baseline methods include:

- **Simple statistical models**: Historical average and Moving average are two basic methods that use the past values of wind power generation to calculate the average value for the next time step. These methods are simple and fast, but they cannot capture the complex and dynamic patterns of wind power generation (Mehdizadeh, Kozekalani Sales, & Safari, 2020).
- **Gradient boosting framework**: LightGBM (Ju et al., 2019) is a gradient boosting decision tree model that can handle large datasets and achieve state-of-the-art performance in many machine learning tasks. LightGBM can capture the nonlinear and complex relationships between wind power generation and various factors such as weather conditions, turbine states, seasonal patterns, and stochastic fluctuations (Zhang, Yan, Infield, Liu, & Lien, 2019).
- **Deep learning architectures**: GRU (Saini, Bhardwaj, Gupta, Kumar, & Mathur, 2020) and GNN (Li, 2022) are two deep learning models that can learn temporal dependencies and spatial correlations in wind power data. GRU is a recurrent neural network model that uses a memory cell to store and update information over time. GNN is a graph neural network model that uses a graph structure to represent the spatial connections between different wind turbines. iTransformer (Liu et al., 2023), Autoformer (Wu, Xu, Wang, & Long, 2021), and FEDformer (Zhou, Ma et al., 2022) are all transformer variants, which mainly deal with long-term dependent time series prediction models.
- **Linear regression model**: MDLinear is a linear regression model that uses multiple features to predict wind power generation. This model is simple and fast, but it cannot capture the nonlinear and complex relationships between wind power generation and various factors (Li & Armandpour, 2022).
- **Graph neural network model**: XTGN is a graph neural network model that uses a graph structure to represent the spatial connections between different wind turbines. This model can capture the spatial correlations in wind power data, but it cannot capture the temporal dependencies (Li & Armandpour, 2022).

Table 2
Offline scores and inference times for the methods (window size = 288, equivalent to 288 ten-minute intervals).

Method	RMSE	MAE	MAPE	Time (s)	Score
Historical average	56.72	47.86	15.67	**121**	52.29
Moving average	61.56	50.62	16.39	127	56.09
GRU	55.13	45.77	12.73	409	50.45
GNN	55.39	47.15	11.29	245	51.27
LightGBM	53.05	44.89	15.41	4035	48.97
MDLinear	56.74	48.32	13.54	1384	52.53
XTGN	54.54	46.50	11.75	227	50.52
MDLinear + XTGN	53.74	45.86	10.83	1420	49.80
Autoformer	52.83	44.37	11.45	553	49.18
FEDformer	52.50	43.79	10.96	477	48.65
iTransformer	51.96	43.52	10.86	461	48.37
EFM	**35.69**	**30.74**	**5.77**	1557	**33.21**
EFM+	**33.37**	**28.31**	**6.13**	1858	**30.84**

- **Hybrid model**: MDLinear + XTGN is a hybrid model that combines a MDLinear and a XTGN to predict wind power generation. This model ranked sixth in the KDDcup 2022 competition (Zhou, Lu et al., 2022).

The proposed models (EFM and EFM+) apply the data preprocessing techniques described in Section 5.2 to enhance the data quality and reliability, and to mitigate the impact of noise and outliers on the prediction performance. These techniques are inspired by some previous studies that have used similar approaches for wind power prediction, such as Zhen et al. (2020) and Zhang et al. (2019). The other baselines follow their original papers.

Table 2 presents the performance of all methods (using the test set provided by the competition). The results show that the proposed models (EFM and EFM+) consistently outperform existing methods across all metrics. The EFM+ model achieves the lowest RMSE, MAE, and MAPE scores, indicating the highest prediction accuracy among all methods. The gap between the scores of EFM+ and other methods is significant, particularly due to the dynamic weighting scheme that can adapt to real-time changes in wind conditions and data distribution, further demonstrating the model's effectiveness.

These results suggest that the proposed EFM+, enhanced by both the data preprocessing techniques and the dynamic weighting scheme introduced in this paper, offers superior prediction performance compared to existing methods. However, EFM+ also has some drawbacks. Its high time cost is evident from the comparison of the average time cost per turbine in the table, and its resource consumption is relatively large, requiring 28MB of memory, which may hinder deployment on resource-constrained terminal devices. Thus, while the model excels in accuracy, there remains room for improvement in terms of computational efficiency, particularly for real-time applications.

5.5. Individual and fusion model evaluation

In this subsection, we validate the generality and robustness of the proposed EFM by applying it to different combinations of expert models. We also show the reason why we select XGBoost and self-attention LSTM as our expert models in the proposed EFM+. In this experiment, we used 288 data points for prediction, which corresponds to one day ahead hourly wind power prediction. We selected eight expert models, namely SALSTM (Self-Attention LSTM), RFR (Random Forest Regressor), XGBoost, Bayesian, GBR (Gradient Boosting Regressor), KNN (K-Nearest Neighbors), SVR (Support Vector Regressor), and LightGBM, to serve as the individual models for the fusion process. These models are selected based on their popularity and effectiveness in wind power prediction. We employed different model combination techniques to assess the dependency and robustness of the ensemble model with respect to the expert models. For each model, we trained them using pre-processed data and made predictions for the next 288

Table 3
Performance of individual models in wind power prediction (window size = 288, in fans site 28).

Method	SALSTM	XGB	RFR	GBR	LGB	Bayesian	KNN	SVR
RMSE	32.6260	33.0503	32.2731	34.8188	34.5103	33.0116	35.2254	33.7177
MAPE	5.0427	5.1449	4.9087	4.7103	5.4408	5.3018	5.5632	7.1596
MAE	25.1492	25.4590	24.9235	26.3672	26.3452	24.6239	28.2564	27.4098

Table 4
Performance comparison of EFM and EFM+ with different combinations of individual models for one-day-ahead wind power prediction (window size = 288, in fans site 28).

Combination	EFM			EFM+		
	RMSE	MAPE	MAE	RMSE	MAPE	MAE
Bayesian+ RFR	33.3562	7.1113	26.8518	**29.6700**	**4.7344**	**23.4253**
RFR+ SALSTM	32.0136	7.0514	25.8984	32.7281	6.6606	25.9286
XGB+ SALSTM	**31.9160**	**7.0617**	**26.0071**	30.1673	5.9875	24.1855
XGB+ RFR	32.8438	7.0748	26.2184	30.8441	5.8825	24.2926
Bayesian+ SALSTM	35.8086	8.8591	29.6189	30.6886	5.0621	24.5686
Bayesian+ XGB	35.9034	7.8228	29.4694	30.4833	4.8004	23.7094
GBR+ LGB	32.8269	7.0818	26.2209	34.0892	6.5927	26.9396
KNN+ SVR	33.5652	6.2741	27.2304	32.5858	6.0854	25.9872
XGB+ SALSTM+ RFR	31.7577	7.0452	25.8020	31.2117	6.2384	25.2575
XGB+ SALSTM+ Bayesian	33.4543	5.6225	27.0214	**30.8728**	**5.0903**	**23.9847**
XGB+ SALSTM+ SVR	32.9956	7.1647	26.7579	31.8834	6.4121	25.6269
XGB+ SALSTM+ LGB	**31.7554**	**7.0300**	**25.6162**	31.6515	6.3396	25.6641
XGB+ SALSTM+ GBR	34.0273	7.5346	28.0303	31.9123	6.0736	25.4867
XGB+ SALSTM+ KNN	34.6912	7.5491	28.4708	33.3769	6.1374	26.2090
XGB+ SALSTM+ RFR+ GBR	31.8050	7.0446	25.9472	32.0559	6.1434	25.4126
XGB+ SALSTM+ RFR+ Bayesian	**31.7080**	**7.0483**	**25.9276**	**31.3705**	**5.0538**	**24.1300**
XGB+ SALSTM+ RFR+ LGB	32.1402	7.0616	25.9803	32.5384	6.3447	25.7752
XGB+ SALSTM+ RFR+ KNN	32.0897	7.0370	26.1203	33.8600	6.2596	26.7632
XGB+ SALSTM+ RFR+ SVR	32.8057	6.8526	26.6779	33.5695	6.4153	26.5840
XGB+ SALSTM+ RFR+ Bayesian+ LGB	32.0168	7.0573	25.9385	**30.9928**	**4.9650**	**24.1171**
XGB+ SALSTM+ RFR+ Bayesian+ GBR	**31.6906**	**7.0406**	**25.9081**	31.4490	5.0197	24.07405
XGB+ SALSTM+ RFR+ Bayesian+ SVR	32.7322	6.8519	26.6714	32.2037	5.0010	24.7740
XGB+ SALSTM+ RFR+ Bayesian+ KNN	31.9101	7.0307	26.0606	32.4526	5.169	25.2992
XGB+ SALSTM+ RFR+ GBR+ LGB+ SVR	32.7109	6.8534	26.6536	34.6317	5.1607	27.0012
XGB+ SALSTM+ RFR+ GBR+ Bayesian+ KNN	**31.9038**	**7.0329**	**26.0484**	33.7717	5.2225	26.1822
XGB+ SALSTM+ RFR+ GBR+ Bayesian+ SVR	32.7109	6.8534	26.6536	33.7696	5.0963	26.1896
XGB+ SALSTM+ RFR+ GBR+ Bayesian+ LGB	31.9997	7.0523	25.9232	**33.3148**	**5.1013**	**26.1630**
ALL	32.7706	6.9953	26.6972	34.8802	5.3413	27.0951

data points, while recording their performance metrics. To select the best combination of models, we randomly combined two expert models and observed their performance metrics. We then selected the best-performing combination as an ensemble model for the next round of random model combinations. We repeated this process incrementally, gradually increasing the number of models in the ensemble. We used EFM and EFM+ as the fusion methods for the ensemble models. The results are presented in the following two tables.

Table 3 shows the RMSE, MAPE, and MAE scores for each expert model. The table reveals that RFR has the lowest RMSE and MAE scores, indicating the smallest prediction error, while GBR has the lowest MAPE score, indicating the highest prediction accuracy among all the expert models. The table suggests that the expert models have different strengths and weaknesses in wind power prediction, and that combining them may improve the prediction performance.

The effectiveness of adaptive expert fusion often relies on the diversity among models. When there are fewer sub-models, the differences between them may not be enough to support the effective work of the dynamic adaptive module, resulting in weakened synergy between models. When there are too many sub-models, some sub-models may not have valid information or may even introduce noise. When the dynamic adaptation module attempts to synthesize the output of these sub-models, it may be affected by noise, resulting in overall performance degradation. Table 4 shows the RMSE, MAPE, and MAE scores for each expert model combination, along with the fusion method used (EFM or EFM+). Verifying the previous speculation, the performance of the ensemble model depends on the selection and number of expert models. Some combinations of expert models achieve better results than others, and adding more expert models does not necessarily improve performance. When there are fewer expert models, the integration effect is more unstable, and after passing through the dynamic adaptive module, the effect will be worse. When there are many expert models, the integration effect shows a downward trend, and the effect is worse than that of a single expert model. Relative to EFM+, the best number of expert models is 3, and EFM+ is generally better than EFM, which illustrates the effectiveness of the dynamic integration module. In addition to the poor fitting effect of KNN, it has also been greatly improved compared to the single expert model. The best performing expert model combination is XGBoost+SALSTM+Bayesian, which achieves the lowest RMSE, MAPE and MAE scores on EFM+. This combination of expert models can capture the complex and non-linear relationships between wind power generation and various factors, as well as the temporal and spatial dependencies in wind power data. The dynamic integration module can adjust the weight of the expert model according to real-time changes in wind conditions and data distribution, further improving prediction accuracy. XGBoost is good at processing nonlinear relationships and high-dimensional features and can efficiently capture complex relationships between features; SALSTM can effectively process time series data through the combination of self-attention mechanism and long-short-term memory network, especially in dehydration prediction. Capture the characteristics of long-term arousal and critical moments; the Bayesian model provides dynamic uncertainty estimation and can adaptively adjust the parameters of the model according to changes in data to ensure that the model maintains stability and accuracy in complex scenarios. Adapting to the fusion mechanism, the advantages of different expert models can be fully utilized, which improves the overall prediction effect and improves the robustness of the model.

These results demonstrate that the proposed EFM can be generalized to other expert models and reduce the dependence on the performance of a single model. They also justify our choice of XGBoost and self-attention LSTM as our expert models in the proposed EFM+, as they have shown to be the most effective and stable expert models for wind power prediction. We also compared the performance of the ensemble models with the expert models and the existing methods.

5.6. Ablation study

We now perform ablation studies to evaluate the performance of two separate models in EFM+ (Self-Attentive LSTM (SALSTM) and XGBoost), EFM, and EFM +. We use preprocessed datasets with or without $C = 5$ k-means clustering applied. The prediction window size for the above experiments was 288, and we now use 144 as the window length, which gives an indication of the model's ability to adapt under different prediction windows. Table 5 shows the performance results for clustered and unclustered data. For clustered data, we also report the average performance across all clusters. We now conduct the ablation study to evaluate the performance of the two individual models in EFM+ (self-attention LSTM (SALSTM) and XGBoost), EFM, and EFM+. We use the pre-processed data set with or without applying k-means clustering with $C = 5$. These models are used to forecast future wind power generation data with a window size of $W = 144$. Table 5 shows their performance results for both clustered and un-clustered data. For the clustered data, we also report the average performance across all clusters.

The results show that EFM+ achieves the best performance among all the models for both clustered and un-clustered data. EFM+ has the lowest values of RMSE, MAE, MAPE, and score for both cases, indicating that it can provide more accurate and robust predictions than the other models. The results also show that clustering the data can improve the performance of all the models, as the values of RMSE, MAE, MAPE, and score are lower for clustered data than for unclustered data. This indicates that clustering can reduce data interference and enhance model fitting and training. From the results, we could observe that EFM and EFM+ have better performance than the individual models (SALSTM and XGBoost) for both clustered and un-clustered data. This indicates that using an ensemble technique can take advantage of the complementary strengths of different models and improve overall prediction accuracy and stability.

To further illustrate the prediction performance of EFM and EFM+, we plot their prediction errors in Fig. 6. The figure shows the prediction errors of EFM and EFM+ for a sample wind turbine over a period of two days (288 data points). The prediction errors are calculated as the absolute difference between the predicted and actual power output values. The figure shows that EFM+ has lower prediction errors than EFM for most of the time, especially when there are sudden changes or spikes in power output. This demonstrates that EFM+ can capture the dynamic and nonstationary patterns of wind power generation better than EFM, and can adapt to changes in wind conditions and data distribution more quickly and effectively.

Therefore, the results demonstrate the effectiveness and superiority of EFM+ for online wind power prediction. EFM+ can leverage the strengths of multiple models and use a dynamic weighting scheme to combine their predictions based on their current performance and error. EFM+ can also handle nonstationarity and uncertainty in wind power generation by updating the weights of the individual models at each time step using a subset-based error calculation. EFM+ can provide accurate and timely forecasts for intra-hour scheduling, which is essential for optimal power system operation and planning.

Table 5
Ablation study using clustered and un-clustered data (window size = 144, in fans site 28).

Method	Cluster	RMSE	MAE	MAPE	Score
SALSTM	0	7.87	9.72	12.89	8.79
	1	41.88	47.80	4.12	44.84
	2	26.41	31.10	4.71	28.76
	3	18.89	25.00	6.52	21.94
	4	27.22	36.50	1.89	31.86
	Average	**24.45**	**30.02**	**6.03**	**27.24**
	Un-clustered	36.96	46.26	24.85	41.61
XGBoost	0	9.63	12.50	16.49	11.06
	1	33.74	40.90	3.33	37.32
	2	32.45	37.30	5.86	29.89
	3	15.62	18.10	5.31	16.86
	4	20.14	25.60	1.39	22.62
	Average	**22.31**	**27.06**	**6.48**	**24.69**
	Un-clustered	31.93	43.48	15.67	37.71
EFM	0	7.87	9.72	12.89	8.79
	1	41.88	47.80	4.12	44.84
	2	26.41	31.10	4.71	28.76
	3	18.89	25.00	6.52	21.94
	4	27.22	36.50	1.89	31.86
	Average	**24.45**	**30.02**	**6.03**	**27.24**
	Un-clustered	36.95	46.25	24.84	41.60
EFM+	0	7.45	9.24	11.98	8.35
	1	31.46	37.10	3.14	34.28
	2	32.87	39.00	5.96	29.87
	3	17.97	22.30	5.83	23.10
	4	19.66	25.60	1.53	19.66
	Average	**21.94**	**26.65**	**5.65**	**24.30**
	Un-clustered	32.06	43.71	15.36	37.89

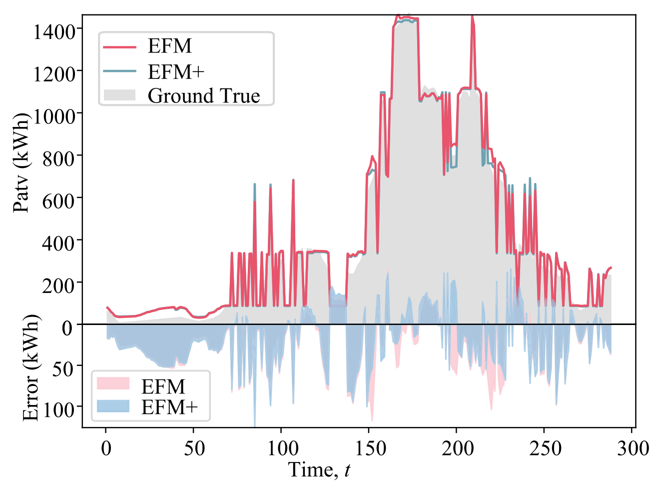

Fig. 6. Prediction results of EFM and EFM+.

5.7. Comparison of methods

This section provides a comprehensive comparison of the proposed EFM+ model, evaluating its performance using different clustering and ensemble methods. The analysis focuses on how these methods impact the predictive accuracy and computational efficiency of EFM+, validating its overall superiority in wind power prediction.

5.7.1. Clustering comparison

Five clustering methods – hierarchical clustering, spectral clustering, Gaussian mixture model, DBSCAN, and K-means – were assessed in terms of RMSE, MAPE, MAE, and clustering time. As shown in Table 6, K-means clustering consistently outperforms the other methods when combined with both EFM and EFM+, offering a strong balance between prediction accuracy and computational efficiency.

Table 6
Comparing the effect of EFM and EFM+ and the time required for clustering under different clustering methods (window size = 288, in fans site 28).

Clustering method	EFM			EFM+			Clustering time (s)
	RMSE	MAPE	MAE	RMSE	MAPE	MAE	
hierarchical clustering	38.5206	7.8241	31.9290	33.3861	2.0874	31.0830	38.6505
DBSCAN	53.8371	22.4005	45.5394	36.5899	17.3445	31.2041	**0.2008**
Gaussian mixture model	43.0089	7.6038	33.1120	31.1673	5.9875	27.1855	0.6309
Spectral clustering	87.3722	38.5643	75.9714	82.4936	32.3623	69.8425	720.3677
K_means	**33.4511**	**5.6247**	**27.0215**	**30.8728**	**5.0903**	**23.9847**	0.2503

Table 7
Comparison of the effect of different integration models, each integration model consists of three sub-models (window size = 288, in fans site 28).

Ensemble method	RMSE	MAPE	MAE
Voting regressor	33.3243	4.8053	27.4636
Adaptive weighted average	37.1845	**3.1009**	30.6960
Stacking regressor	39.4692	7.4634	30.5439
EFM	33.4511	5.6247	27.0215
EFM+	**30.8728**	5.0903	**23.9847**

Hierarchical clustering and Gaussian mixture models also show competitive accuracy; however, their higher clustering times make them less suitable for real-time applications. On the other hand, DBSCAN, while computationally efficient, struggles with prediction accuracy, particularly with capturing complex wind patterns. Spectral clustering performs the worst in terms of clustering time, making it impractical for scenarios requiring fast, real-time forecasting.

The results suggest that K-means is the most suitable clustering method for EFM+, as it combines accurate predictions with relatively low clustering time. This makes it ideal for real-time wind power prediction, where both accuracy and speed are crucial.

5.7.2. Ensemble comparison

In addition to clustering methods, we compared the performance of EFM and EFM+ against traditional ensemble techniques, including voting regressor, adaptive weighted average, and stacking regressor (Table 7). The results demonstrate that EFM+ significantly outperforms these traditional methods, consistently achieving better accuracy across all evaluation metrics.

While the adaptive weighted average method shows good performance under specific conditions, its overall flexibility and adaptability are limited compared to EFM+. The voting regressor and stacking regressor, although useful in typical ensemble contexts, fall short in handling the real-time data variability encountered in wind power forecasting. The dynamic weighting mechanism employed by EFM+ enables it to adjust more effectively to changing wind conditions, providing superior predictive accuracy and robustness.

The findings highlight that EFM+, through its adaptive expert fusion approach, is better suited to handle the complex, non-linear relationships present in wind power data. This flexibility allows EFM+ to outperform traditional ensemble models in real-time prediction tasks, ensuring both accuracy and efficiency.

5.8. Model sensitivity analysis

In this subsection, we conduct the sensitivity analysis for the proposed adaptive expert fusion model (EFM+) using three methods: (1) Bayesian optimization (Nguyen, 2019), (2) adding Gaussian noise to training data, and (3) varying number of features.

First, we only use Bayesian optimization to study how the performance of the EFM+ model changes with respect to different parameter values. Bayesian optimization is a technique for optimizing complex and noisy objective functions, such as the performance metrics of the EFM+ model. This technique uses a probabilistic model, such as a Gaussian process, to estimate the posterior distribution of the objective function given the observed data. It also uses an acquisition function, such as expected improvement, to balance the trade-off between exploration and exploitation. It iteratively selects the next exploration point that maximizes the acquisition function, and evaluates the objective function at that point. We use RMSE and MAE as the objective functions and use the learning rate, the number of hidden units, and the weight decay as the parameters to be tuned. We use 288 data points for prediction. The experimental results in Fig. 7 show that the EFM+ model has a lower RMSE and MAE compared to the EFM model. This means that the EFM+ model can achieve higher prediction accuracy and lower prediction error than the EFM model. The figure also shows that the EFM+ model is more robust and stable than the EFM model, as it has less variation in the RMSE and MAE values across different sensitivity points. The sensitivity points are the values of the parameters that are dynamically explored by the Bayesian optimization algorithm based on the acquisition function and the probabilistic model of the objective function. They represent the samples of the parameter space that are used to evaluate the performance of the model. This means that the EFM+ model can adapt to the changes in the wind conditions and data distribution better than the EFM model.

Second, we further add Gaussian noise to the training data to study how the performance of the EFM+ model changes with respect to different levels of noise. Gaussian noise is a type of noise that follows a normal distribution and can be used to simulate the uncertainty and variability in the wind power data. We add Gaussian noise from two to three standard deviations to the training data, and we evaluate the model performance on the test data. The experimental results in Fig. 8 show that the EFM+ model has a lower RMSE and MAE compared to EMF. This means that the EFM+ model can achieve higher prediction accuracy and lower prediction error than the EMF. The figure also shows that the EFM+ model is more robust and stable than EMF, as it has less variation in the RMSE and MAE values across different levels of noise. This means that the proposed model with dynamic weighting scheme can handle the uncertainty and variability in the wind power data in better.

Third, we vary the number of features to investigate how the performance of the model varies with different sets of features. Features are variables or attributes that describe wind power generation data, such as wind speed (`wspd`), pitch angle (`pab`), reactive power (`prtv`), wind direction (`wdir`), internal temperature (`itmp`), and nacelle orientation (`ndir`). For the correlation between the different features and the effect on the target feature (`patv`) can be observed from Fig. 1, `wspd` and `prtv` have a strong positive correlation effect on `patv`. Since prtv is used to maintain voltage stability in the power system, `prtv` can play an auxiliary role to `wspd` to complement the dynamic uncertainty of wspd and improve the stability of the data. Although other features with low correlation between them are prone to generate unnecessary noise, most of the effects on `patv` are negative and small. We select features based on their correlation with wind power generation, using 1 to 6 features per set. We evaluate the EFM+ model on test data using different sets of features. The results in Table 8 show that the EFM+ model generally outperforms the EFM model in terms of RMSE, MAPE, MAE, and Score, which suggests that the EFM+ model has higher prediction accuracy and lower prediction error. The results also show that the EFM+ model is more robust and stable than the EFM model because it has less variation in performance metrics across feature sets. In addition, the results show that the combination of `wspd`, `pab` and

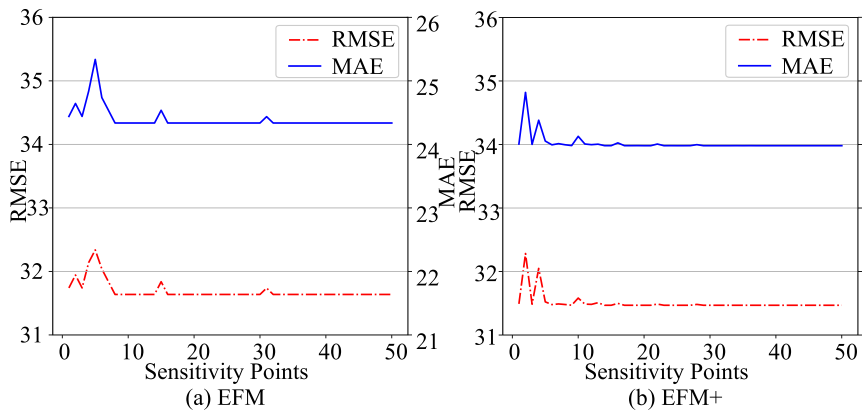

Fig. 7. Model sensitivity analysis(window size = 288, in fans site 28).

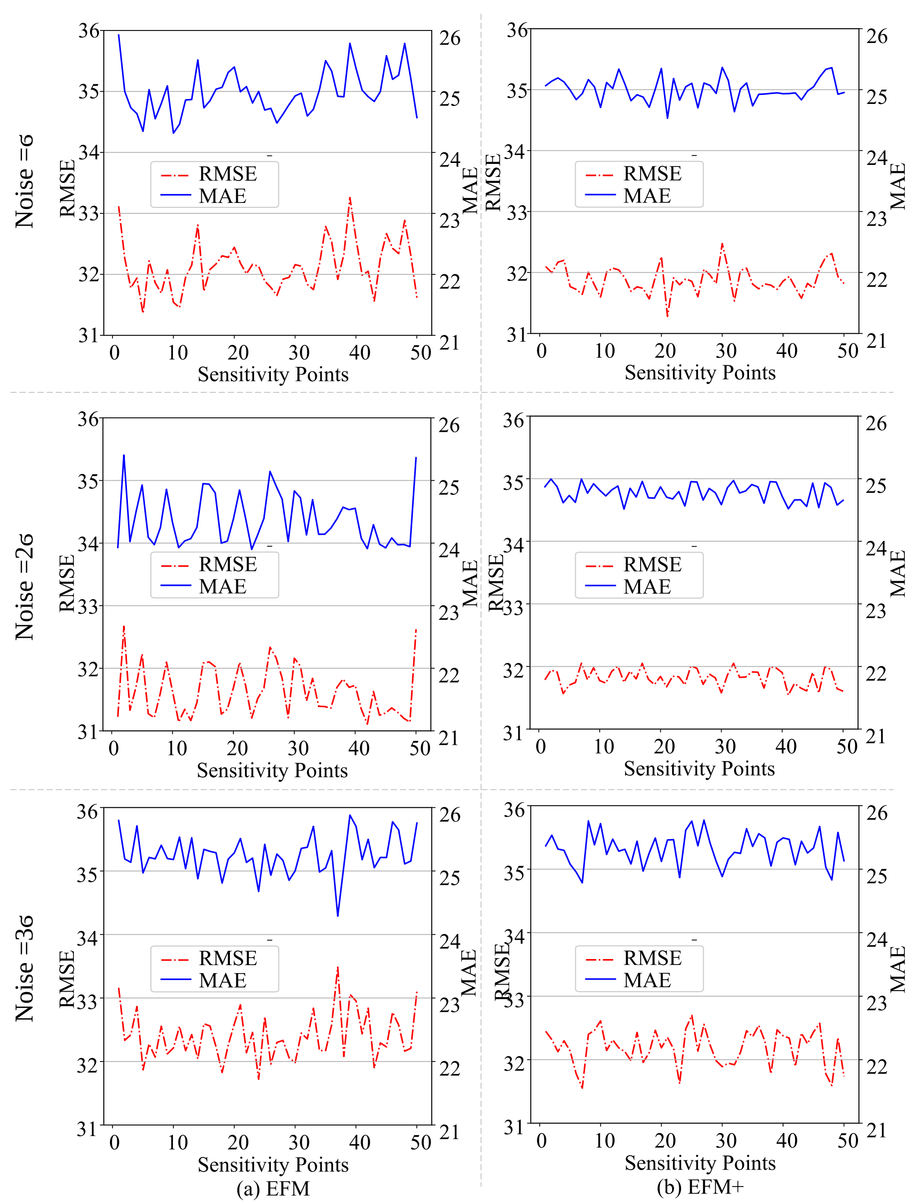

Fig. 8. Sensitivity analysis by adding Gaussian noise(window size = 288, in fans site 28).

Table 8
Sensitivity analysis by varying number of features (window size = 288, in fan site 28).

Feature	EFM				EFM+			
	RMSE	MAPE	MAE	Score	RMSE	MAPE	MAE	Score
wspd	34.24	5.91	27.39	30.81	35.11	5.96	27.90	31.50
wspd+pab	34.54	6.02	27.72	31.13	33.35	5.80	26.60	29.97
wspd+prtv	34.29	5.79	28.01	31.15	34.11	5.69	27.52	30.81
wspd+pab+prtv	**33.45**	**5.62**	**27.02**	**30.23**	**30.87**	**5.09**	**23.98**	**28.14**
wspd+wdir+ndir	38.62	6.69	31.39	35.00	36.52	6.06	29.55	33.03
wspd+etmp+itmp	37.08	6.62	30.02	33.55	35.97	6.13	29.03	32.50
wspd+pab+prtv+etmp+itmp	34.37	5.76	27.80	31.08	31.47	5.26	25.44	28.46
wspd+pab+prtv+wdir+ndir	35.06	5.83	28.43	31.74	33.49	5.62	27.22	30.36
all	33.93	5.89	27.26	30.59	30.93	5.51	24.73	28.68

`prtv` achieves the best performance, which validates the effectiveness of our feature selection.

To summarize, the sensitivity analysis shows that the EFM+ model can adapt its performance to different parameter values, noise levels, and feature sets. It also confirms our selection of XGBoost and self-attention LSTM as the expert models in the EFM+, as they are the most effective and stable models for wind power prediction. Furthermore, it validates the feature set that we use in our model, which consists of wind speed, pitch angle, and reactive power.

6. Conclusions and future work

Wind power prediction is a vital task to optimize the planning and operation of power systems, as well as to integrate more renewable energy sources into the grid. In this paper, we proposed a novel Adaptive Expert Fusion Model (EFM+) for online wind power prediction. The EFM+ is an innovative ensemble model that combines predictions from multiple models using a dynamic weighting scheme based on their current performance. The EFM+ can capture complex and non-linear relationships between wind power generation and various factors, adapt to changes in wind conditions and data distribution over time, and provide accurate predictions for intra-hour scheduling. We conducted extensive experiments on a real-world wind farm dataset and compared our EFM+ with state-of-the-art methods within different categories. The results showed that our EFM+ outperformed all the other methods in terms of prediction accuracy and error, and demonstrated strong robustness and stability under different scenarios. We also performed ablation studies and sensitivity analyses to evaluate the impact of different components and parameters on the performance of our model.

The proposed model has several advantages. First, it is dynamically adaptable and capable of updating model parameters in real time in response to changing input data. This flexibility makes our model more robust in real-world applications, especially in online learning scenarios. Furthermore, by integrating multiple expert models such as XGBoost, SALSTM, and Bayesian regression, our model leverages the strengths of different algorithms, improving overall prediction performance. Adaptive expert gating and hybrid gating enable the model to adjust the weights of different models dynamically, further enhancing accuracy. However, there are some limitations, such as the computational complexity during online learning, which can increase resource consumption, especially with large-scale data. Additionally, the choice of Euclidean distance as the similarity measure may not be optimal in all cases, and model performance is also dependent on hyperparameter selection, requiring fine-tuning for different datasets to achieve the best results.

For future work, we plan to extend our EFM+ to other renewable energy sources, such as solar power and hydro power, and explore the possibility of integrating them into a unified framework for multi-source energy prediction. We also intend to incorporate more features and data sources, such as weather forecasts, grid load, and market prices, to improve prediction performance and provide more insight for energy management. Furthermore, we aim to develop more efficient and scalable algorithms for online learning and adaptive forecasting, as well as more effective and interpretable methods for model evaluation and explanation.

CRediT authorship contribution statement

Renfang Wang: Writing – review & editing, Writing – original draft, Software, Methodology, Formal analysis. **Jingtong Wu:** Writing – original draft, Software, Methodology, Data curation. **Xu Cheng:** Writing – review & editing, Writing – original draft, Methodology, Formal analysis. **Xiufeng Liu:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Formal analysis, Conceptualization. **Hong Qiu:** Writing – review & editing, Writing – original draft, Resources, Methodology, Funding acquisition, Formal analysis.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 61906170), the Project of the Science and Technology Plan for Zhejiang Province, China (No. LGF21F020023), the Plan Project of Ningbo Municipal Science and Technology, China (Nos. 2022Z233, 2021Z050, 2022S002 and 2023J403), and the Marie Skłodowska-Curie Postdoctoral Individual Fellowship project ANSWER under Grant No. 101111188.

Data availability

The authors do not have permission to share data.

References

Aguero, J. R., Takayesu, E., Novosel, D., & Masiello, R. (2017). Modernizing the grid: Challenges and opportunities for a sustainable future. *IEEE Power and Energy Magazine*, *15*(3), 74–83.

Al-Dahidi, S., Baraldi, P., Zio, E., & Legnani, E. (2017). A dynamic weighting ensemble approach for wind energy production prediction. In *2017 2nd international conference on system reliability and safety* ICSRS, (pp. 296–302). IEEE.

Albukhanajer, W. A., Jin, Y., & Briffa, J. A. (2015). Trade-off between computational complexity and accuracy in evolutionary image feature extraction. In *2015 IEEE congress on evolutionary computation* CEC, (pp. 2412–2419). IEEE.

Aruna, M., Narayana, P. B., Kumar, S. N., Walid, A. A., Patra, J. P., & Kumar, B. S. (2023). Sparrow search optimization with deep belief network based wind power prediction model. In *2023 international conference on intelligent data communication technologies and internet of things (iDCIoT)* (pp. 765–770). IEEE.

Aslam, M. S., Radhika, T., Chandrasekar, A., & Zhu, Q. (2024). Improved event-triggered-based output tracking for a class of delayed networked T–S fuzzy systems. *International Journal of Fuzzy Systems*, 1–14.

Belaïd, F., Al-Sarihi, A., & Al-Mestneer, R. (2023). Balancing climate mitigation and energy security goals amid converging global energy crises: The role of green investments. *Renewable Energy*, *205*, 534–542.

Bilal, B., Adjallah, K. H., Sava, A., Yetilmezsoy, K., & Ouassaid, M. (2023). Wind turbine output power prediction and optimization based on a novel adaptive neuro-fuzzy inference system with the moving window. *Energy*, *263*, Article 126159.

Cao, X., Lei, Z., Li, Y., Zhang, M., & Duan, X. (2022). Prediction method of equipment remaining life based on self-attention long short-term memory neural network. *Journal of Shanghai Jiaotong University (Science)*, 1–13.

Chandrasekar, A., Radhika, T., & Zhu, Q. (2022). Further results on input-to-state stability of stochastic Cohen–Grossberg BAM neural networks with probabilistic time-varying delays. *Neural Processing Letters*, 1–23.

Chang, Z., Zhang, Y., & Chen, W. (2019). Electricity price prediction based on hybrid model of adam optimized LSTM neural network and wavelet transform. *Energy*, *187*, Article 115804.

Chen, G., Shan, J., Li, D. Y., Wang, C., Li, C., Zhou, Z., et al. (2019). Research on wind power prediction method based on convolutional neural network and genetic algorithm. In *2019 IEEE Innovative Smart Grid Technologies-Asia (ISGT Asia)* (pp. 3573–3578).

Deon, S., de Lima, J. D., Dranka, G. G., Ribeiro, M. H. D. M., dos Anjos, J. C. S., de Paz Santana, J. F., et al. (2024). Ensemble learning models for wind power forecasting. In *International conference on disruptive technologies, tech ethics and artificial intelligence* (pp. 15–27). Springer.

Dittmer, A., Sharan, B., & Werner, H. (2022). Data-driven adaptive model predictive control for wind farms: A koopman-based online learning approach. In *2022 IEEE 61st conference on decision and control* CDC, (pp. 1999–2004). IEEE.

Ember (2022). Global electricity review 2022. URL https://ember-climate.org/insights/research/global-electricity-review-2022/.

Fekri, M. N., Patel, H., Grolinger, K., & Sharma, V. (2021). Deep learning for load forecasting with smart meter data: Online adaptive recurrent neural network. *Applied Energy*, *282*, Article 116177.

Foley, A. M., McIlwaine, N., Morrow, D. J., Hayes, B. P., Zehir, M. A., Mehigan, L., et al. (2020). A critical evaluation of grid stability and codes, energy storage and smart loads in power systems with wind generation. *Energy*, *205*, Article 117671.

Guan, S., Wang, Y., Liu, L., Gao, J., Xu, Z., & Kan, S. (2023). Ultra-short-term wind power prediction method combining financial technology feature engineering and XGBoost algorithm. *Heliyon*.

Han, X., Zheng, S., He, Z., Huang, J., Che, L., & Li, L. (2023). Short-term wind power prediction model based on improved ACA-GRU neural network. *vol 2527*, In *Journal of physics: conference series*. IOP Publishing, Article 012078.

Hoi, S. C. H., Sahoo, D., Lu, J., & Zhao, P. (2018). Online learning: A comprehensive survey. *Neurocomputing*, *459*, 249–289, URL https://api.semanticscholar.org/CorpusID:3619626.

Hossain, M. A., Chakrabortty, R. K., Elsawah, S., & Ryan, M. J. (2021). Very short-term forecasting of wind power generation using hybrid deep learning model. *Journal of Cleaner Production*, *296*, Article 126564.

Hu, Y., Liu, H., Wu, S., Zhao, Y., Wang, Z., & Liu, X. (2024). Temporal collaborative attention for wind power forecasting. *Applied Energy*, *357*, Article 122502.

Hu, T., Wu, W., Guo, Q., Sun, H., Shi, L., & Shen, X. (2020). Very short-term spatial and temporal wind power forecasting: A deep learning approach. *CSEE Journal of Power and Energy Systems*, *6*(2), 434–443. http://dx.doi.org/10.17775/CSEEJPES.2018.00010.

Hu, S., Xiang, Y., Zhang, H., Xie, S., Li, J., Gu, C., et al. (2021). Hybrid forecasting method for wind power integrating spatial correlation and corrected numerical weather prediction. *Applied Energy*, *293*, Article 116951.

Huang, C.-M., Chen, S.-J., Yang, S.-P., & Chen, H.-J. (2023). One-day-ahead hourly wind power forecasting using optimized ensemble prediction methods. *Energies*, *16*(6), 2688.

Huang, C.-J., & Kuo, P.-H. (2018). A short-term wind speed forecasting model by using artificial neural networks with stochastic optimization for renewable energy systems. *Energies*, *11*(10), 2777.

Jin, H., Li, Y., Wang, B., Yang, B., Jin, H., & Cao, Y. (2022). Adaptive forecasting of wind power based on selective ensemble of offline global and online local learning. *Energy Conversion and Management*, *271*, Article 116296.

Jose, R., Panigrahi, S. K., Patil, R. A., Fernando, Y., & Ramakrishna, S. (2020). Artificial intelligence-driven circular economy as a key enabler for sustainable energy management. *Materials Circular Economy*, *2*, 1–7.

Ju, Y., Sun, G., Chen, Q., Zhang, M., Zhu, H., & Rehman, M. U. (2019). A model combining convolutional neural network and LightGBM algorithm for ultra-short-term wind power forecasting. *Ieee Access*, *7*, 28309–28318.

Kashinath, K., Mustafa, M., Albert, A., Wu, J., Jiang, C., Esmaeilzadeh, S., et al. (2021). Physics-informed machine learning: case studies for weather and climate modelling. *Philosophical Transactions of the Royal Society, Series A*, *379*(2194), Article 20200093.

Katrompas, A., & Metsis, V. (2022). Enhancing LSTM models with self-attention and stateful training. *vol. 1*, In *Intelligent systems and applications: proceedings of the 2021 intelligent systems conference* (intelliSys), (pp. 217–235). Springer.

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

Li, J., & Armandpour, M. (2022). Deep spatio-temporal wind power forecasting. In *ICASSP 2022-2022 IEEE international conference on acoustics, speech and signal processing* ICASSP, (pp. 4138–4142). IEEE.

Li, N., He, F., Ma, W., Wang, R., & Zhang, X. (2020). Wind power prediction of kernel extreme learning machine based on differential evolution algorithm and cross validation algorithm. *Ieee Access*, *8*, 68874–68882.

Li, C., Tang, G., Xue, X., Chen, X., Wang, R., & Zhang, C. (2019). Deep interval prediction model with gradient descend optimization method for short-term wind power prediction. arXiv preprint arXiv:1911.08160.

Li, X., Wang, J., Geng, Z., Jin, Y., & Xu, J. (2023). Short-term wind power prediction method based on genetic algorithm optimized XGBoost regression model. *vol. 2527*, In *Journal of physics: conference series*. IOP Publishing, Article 012061.

Li, D., Yang, F., Miao, S., Gan, Y., Yang, B., & Zhang, Y. (2023). An adaptive spatiotemporal fusion graph neural network for short-term power forecasting of multiple wind farms. *Journal of Renewable and Sustainable Energy*, *15*(1).

Liu, Y., Hu, T., Zhang, H., Wu, H., Wang, S., Ma, L., et al. (2023). Itransformer: Inverted transformers are effective for time series forecasting. arXiv preprint arXiv:2310.06625.

Liu, X., Yang, L., & Zhang, Z. (2021). Short-term multi-step ahead wind power predictions based on a novel deep convolutional recurrent network method. *IEEE Transactions on Sustainable Energy*, *12*(3), 1820–1833. http://dx.doi.org/10.1109/TSTE.2021.3067436.

Liu, X., & Zhang, Y. (2020). Wind power generation: A review of worldwide trends. *Renewable and Sustainable Energy Reviews*, *134*, Article 110371.

Liu, H., & Zhang, Z. (2022). A bilateral branch learning paradigm for short term wind power prediction with data of multiple sampling resolutions. *Journal of Cleaner Production*, *380*, Article 134977.

Liu, H., & Zhang, Z. (2023). A bi-party engaged modeling framework for renewable power predictions with privacy-preserving. *IEEE Transactions on Power Systems*, *38*(6), 5794–5805. http://dx.doi.org/10.1109/TPWRS.2022.3224006.

Mackenzie, W. (2022). Global installed wind power capacity set to grow by 9% to 2030. URL https://www.woodmac.com/press-releases/global-installed-wind-power-capacity-set-to-grow-by-9-to-2030/.

Maldonado-Correa, J., Solano, J., & Rojas-Moncayo, M. (2021). Wind power forecasting: A systematic literature review. *Wind Engineering*, *45*(2), 413–426.

Martín-Vázquez, R., Aler, R., & Galván, I. M. (2018). Wind energy forecasting at different time horizons with individual and global models. In *Artificial intelligence applications and innovations: 14th IFIP wG 12.5 international conference, AIAI 2018, rhodes, Greece, May 25–27, 2018, proceedings 14* (pp. 240–248). Springer.

Mehdizadeh, S., Kozekalani Sales, A., & Safari, M. J. S. (2020). Estimating the short-term and long-term wind speeds: implementing hybrid models through coupling machine learning and linear time series models. *SN Applied Sciences*, *2*, 1–15.

Mora, E. B., Spelling, J., van der Weijde, A. H., & Pavageau, E.-M. (2019). The effects of mean wind speed uncertainty on project finance debt sizing for offshore wind farms. *Applied Energy*, *252*, Article 113419.

Nguyen, V. (2019). Bayesian optimization for accelerating hyper-parameter tuning. In *2019 IEEE second international conference on artificial intelligence and knowledge engineering* AIKE, (pp. 302–305). IEEE.

Özdemir, M. E. (2024). A novel ensemble wind speed forecasting system based on artificial neural network for intelligent energy management. *IEEE Access*.

Peng, X., Li, C., Jia, S., Zhou, L., Wang, B., & Che, J. (2022). A short-term wind power prediction method based on deep learning and multistage ensemble algorithm. *Wind Energy*, *25*(9), 1610–1625.

Qin, R., Wang, L., Du, X., Chen, X., & Yan, B. (2023). Dynamic ensemble selection based on deep neural network uncertainty estimation for adversarial robustness. arXiv preprint arXiv:2308.00346.

Qiu, H., Shi, K., Wang, R., Zhang, L., Liu, X., & Cheng, X. (2024). A novel temporal–spatial graph neural network for wind power forecasting considering blockage effects. *Renewable Energy*, *227*, 120499.

Radhika, T., Chandrasekar, A., Vijayakumar, V., & Zhu, Q. (2023). Analysis of Markovian jump stochastic Cohen–Grossberg BAM neural networks with time delays for exponential input-to-state stability. *Neural Processing Letters*, *55*(8), 11055–11072.

Rafique, S. F., & Jianhua, Z. (2018). Energy management system, generation and demand predictors: a review. *IET Generation, Transmission & Distribution*, *12*(3), 519–530.

Saini, V. K., Bhardwaj, B., Gupta, V., Kumar, R., & Mathur, A. (2020). Gated recurrent unit (gru) based short term forecasting for wind energy estimation. In *2020 international conference on power, energy, control and transmission systems* ICPECTS, (pp. 1–6). IEEE.

Sarshar, J., Moosapour, S. S., & Joorabian, M. (2017). Multi-objective energy management of a micro-grid considering uncertainty in wind power forecasting. *Energy*, *139*, 680–693.

Sun, B., Su, M., & He, J. (2024). Wind power prediction through acoustic data-driven online modeling and active wake control. *Energy Conversion and Management*, *319*, Article 118920. http://dx.doi.org/10.1016/j.enconman.2024.118920, URL https://www.sciencedirect.com/science/article/pii/S0196890424008616.

Sun, S., Sun, S., Wang, Q., & Zhao, P. (2022). Short-term wind power prediction based on meteorological process division. *vol. 2401*, In *Journal of physics: conference series*. IOP Publishing, Article 012025.

Sun, Y., Wang, X., & Yang, J. (2022). Modified particle swarm optimization with attention-based LSTM for wind power prediction. *Energies*, *15*(12), 4334.

Tawn, R., & Browell, J. (2022). A review of very short-term wind and solar power forecasting. *Renewable and Sustainable Energy Reviews*, *153*, Article 111758.

Tian, Y., Wang, D., Zhou, G., Wang, J., Zhao, S., & Ni, Y. (2023). An adaptive hybrid model for wind power prediction based on the ivmd-fe-ad-informer. *Entropy*, *25*(4), 647.

(2024). Ultra-short-term photovoltaic power prediction based on similar day clustering and temporal convolutional network with bidirectional long short-term memory model: A case study using DKASC data. *Applied Energy*, *375*, Article 124085. http://dx.doi.org/10.1016/j.apenergy.2024.124085, URL https://www.sciencedirect.com/science/article/pii/S0306261924014685.

Wang, Y., Wu, Y., Xu, H., Chen, Z., Gao, J., Xu, Z., et al. (2023). A combination predicting methodology based on T-LSTNet_Markov for short-term wind power prediction. *Network: Computation in Neural Systems*, 1–23.

Wiser, R. (2021). Falling costs of wind and solar energy technologies: Implications and opportunities.

Wu, X., Chen, N., Du, Q., Mao, S., & Ju, X. (2023). Short-term wind power prediction model based on ARMA-GRU-QPSO and error correction. *vol. 2427*, In *Journal of physics: conference series*. IOP Publishing, Article 012028.

Wu, H., Xu, J., Wang, J., & Long, M. (2021). Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in Neural Information Processing Systems*, *34*, 22419–22430.

Xia, J., Feng, Y., Lu, C., Fei, C., & Xue, X. (2021). LSTM-based multi-layer self-attention method for remaining useful life estimation of mechanical systems. *Engineering Failure Analysis*, *125*, Article 105385.

Xiao, Y., Wu, S., He, C., Hu, Y., & Yi, M. (2024). An effective hybrid wind power forecasting model based on "decomposition-reconstruction-ensemble" strategy and wind resource matching. *Sustainable Energy, Grids and Networks*, *38*, Article 101293.

Xinxin, W., Xiaopan, S., Xueyi, A., & Shijia, L. (2023). Short-term wind speed forecasting based on a hybrid model of ICEEMDAN, MFE, LSTM and informer. *PLoS One*, *18*(9), Article e0289161.

Yan, J., Liu, Y., Han, S., Wang, Y., & Feng, S. (2015). Reviews on uncertainty analysis of wind power forecasting. *Renewable and Sustainable Energy Reviews*, *52*, 1322–1330.

Zhang, F., Li, P.-C., Gao, L., Liu, Y.-Q., & Ren, X.-Y. (2021). Application of autoregressive dynamic adaptive (ARDA) model in real-time wind power forecasting. *Renewable Energy*, *169*, 129–143.

Zhang, F., Li, N., Li, L., Wang, S., & Du, C. (2023). A local semi-supervised ensemble learning strategy for the data-driven soft sensor of the power prediction in wind power generation. *Fuel*, *333*, Article 126435.

Zhang, C., Tao, Z., Xiong, J., Qian, S., Fu, Y., Ji, J., et al. (2024). Research and application of a novel weight-based evolutionary ensemble model using principal component analysis for wind power prediction. *Renewable Energy*, *232*, Article 121085.

Zhang, J., Yan, J., Infield, D., Liu, Y., & Lien, F.-s. (2019). Short-term forecasting and uncertainty analysis of wind turbine power based on long short-term memory network and Gaussian mixture model. *Applied Energy*, *241*, 229–244.

Zhang, J., Zhai, W., & Sun, Q. (2023). The wind power prediction based on hierarchical clustering method. *vol. 12594*, In *Second international conference on electronic information engineering and computer communication (EIECC 2022)* (pp. 99–104). SPIE.

Zhao, H., Wu, Q., Hu, S., Xu, H., & Rasmussen, C. N. (2015). Review of energy storage system for wind power integration support. *Applied Energy*, *137*, 545–553.

Zhen, H., Niu, D., Yu, M., Wang, K., Liang, Y., & Xu, X. (2020). A hybrid deep learning model and comparison for wind power forecasting considering temporal-spatial feature extraction. *Sustainability*, *12*(22), 9490.

Zhou, Y., Huang, R., Lin, Q., Chai, Q., & Wang, W. (2024). Probabilistic optimization based adaptive neural network for short-term wind power forecasting with climate uncertainty. *International Journal of Electrical Power & Energy Systems*, *157*, Article 109897.

Zhou, J., Lu, X., Xiao, Y., Su, J., Lyu, J., Ma, Y., et al. (2022). SDWPF: A dataset for spatial dynamic wind power forecasting challenge at KDD cup 2022. arXiv:2208.04360.

Zhou, T., Ma, Z., Wen, Q., Wang, X., Sun, L., & Jin, R. (2022). Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *International conference on machine learning* (pp. 27268–27286). PMLR.

Zhou, Y., Wei, F., Kuang, K., & Mahfoud, R. J. (2024). Research on a deep ensemble learning model for the ultra-short-term probabilistic prediction of wind power. *Electronics*, *13*(3), 475.

Zhu, Q., Xu, Y., Lin, Q., Ming, Z., & Tan, K. C. (2024). Clustering-based short-term wind speed interval prediction with multi-objective ensemble learning. *IEEE Transactions on Emerging Topics in Computational Intelligence*.