**Response to Reviewers**

**Preview Manuscript ID: EGY-D-24-04910**

**Title: Integrating Domain Knowledge into Transformer for Short-Term Wind Power Forecasting**

The authors would like to sincerely thank the editor for handling the review of our paper and the reviewers for providing valuable comments and suggestions, which have been very helpful towards improving the quality of the paper. The figure numbers and paragraph numbers in this response refer to those in the revised manuscript, unless otherwise stated. The comments from the reviewers have been given in red-colored font, whereas our point-to-point responses are in black-colored font and examples are in *black-colored* *italic* font. In the manuscript, the main revisions are marked in blue-colored font.

The main changes are summarized as follows:

1. The Section 1 has been revised by incorporating relevant literature on the processing of numerical weather forecast features.
2. Figure 2 has been redrawn to enhance clarity in depicting the input, output, and overall process.
3. Details explanations of the advantage of denoising module have been provided in Subsection 3.1.1.
4. Details explanations of the clipping module and boundary module have been provided in Subsection 3.1.2 and Subsection 3.1.3, respectively.
5. The specific implementation details of transfer learning have been provided in Subsection 3.2.
6. A comprehensive analysis of the data normalization method has been provided in Subsection 4.1.2.
7. The correlation analysis between denoised wind speed and wind power has been provided in Subsection 4.1.3.
8. The evaluation metric has been supplemented with appropriate units.
9. The detailed process of hyper-parameters selection and the training/inference time of DKFormer have been provided in Subsection 4.2.1.
10. A visualization diagram for multi-step forecasting has been added in Subsection 4.3.2.
11. Visualization results of ablation experiments have been provided in Subsection 4.4.
12. A module compatible evaluation section has been provided in Section 4.5.
13. The conclusion and discussion section has been carefully revised.
14. A more detailed explanation is provided on how the boundary module utilizes a cubic polynomial to fit the upper and lower boundary functions in the Appendix.
15. The paper has been thoroughly proofread to correct mistakes and typographical errors.

**Response to the Editor**

Editor’s comments: The review of your paper is now complete, the Reviewers' reports are below. As you can see, one Reviewer recommends rejection, and the others recommend possible reconsideration after major revisions. They have presented important points of criticism and a series of recommendations. We kindly ask you to consider all comments and revise the paper accordingly in order to respond fully and in detail to the Reviewers' recommendations. If this process is completed thoroughly, the paper will be acceptable for a second review. If you choose to revise your manuscript it will be due into the Editorial Office by the Aug 01, 2024.

**Response:** Thank you for your valuable comments. We have clarified the technical issues raised by the reviewers, and a significant revision has been made with all responses in the format of point-to-point included in this document. The novelties and contributions of this work have been emphasized in **the seventh paragraph of Section 1**:

*“...we propose a DL model, namely domain-knowledge integrated Transformer (DKFormer), for short-term wind power forecasting. The incorporation of domain knowledge has been taken into account during the development of DKFormer. Furthermore, historical wind power and NWP data are jointly leveraged to guide the forecasting outcomes of DKFormer. Additionally, transfer learning technique is employed to tackle the issue of model training and further enhance the accuracy and stability of DKFormer. The primary contributions of this research can be summarized as follows.*

*(1) We initially propose the DKFormer forecasting model, which integrates domain knowledge through three constraint modules that are crucial in data pre-processing, model training, and forecasting stages.*

*(2) Unlike conventional sequential extrapolation models that solely rely on historical data for forecasting, our proposed DKFormer model incorporates NWP data’s lead time to guide the forecasting results by constructing a cubic polynomial boundary constraint.*

*(3) We combine transfer learning technique with the DKFormer to enhance the forecasting performance of our model.*

*(4) The proposed domain knowledge constraint modules demonstrate great portability and can be incorporated into various DL baseline models such as LSTM and GRU, resulting in improved forecasting accuracy.*

**Response to Reviewer #1**

**Reviewer #1:** This article proposes a short-term wind power prediction method based on DKFormer. The author demonstrates that the proposed model has a certain degree of accuracy and has certain significance in practical engineering applications by experimental results. But there are a lot of unclear wording and logical errors in the article that the author needs to pay attention to:

(1) The EEMD decomposition method has certain drawbacks, which can cause modal aliasing between components. Why not use other decomposition methods to decompose the features of numerical weather forecasts? In addition, using data decomposition methods poses a risk of future information leakage results in lower credibility of the final prediction results. Has the author taken data leakage prevention measures? What are the specific measures taken?

**Response:** Thank you for the valuable comment. Inevitably, all decomposition methods have certain drawbacks, but we believe that EEMD is more suitable for wind power forecasting. EEMD’s strength lies in its ability to effectively analyze and process non-stationary and non-linear signals by iteratively decomposing signals with added Gaussian white noise, enhancing independence and distinctiveness of the resulting intrinsic mode functions and mitigating modal aliasing issues. Please refer to **the first paragraph of Subsection 3.1.1:**

*“…To mitigate the impact of noise, we employ ensemble empirical mode decomposition (EEMD) [42], which is a suitable technique for analyzing and processing non-stationary and non-linear signals, to denoise and extract features from raw wind speed signals. This approach involves conducting multiple empirical mode decomposition (EMD) by superimposing Gaussian white noise onto the signal. By leveraging the statistical properties of Gaussian white noise with uniformly distributed frequencies, different iterations introduce white noise with equal amplitude that modifies extreme points in the signal. Subsequently, averaging intrinsic mode functions (IMFs) obtained from multiple iterations effectively cancels out the added white noise. Introducing randomness in each iteration helps disrupt the patterns present in the original signal, rendering each IMF more independent and distinct, thereby addressing the issue of modal aliasing in EMD.”*

In addition, to further validate the effectiveness of EEMD, additional experiments for correlation analysis have been conducted. The results demonstrate that the denoised wind speed data obtained through EEMD exhibits a higher Pearson correlation coefficient of 0.82 with wind power, surpassing the correlation of 0.78 observed between wind power and the original wind speed. This finding further emphasizes the effectiveness of EEMD in noise reduction. This has been stated in **the second paragraph of Subsection 4.1.3:**

*“According to the results, Figure 8 shows a correlation coefficient of 0.78 between wind speed (ws) and wind power (wp), surpassing the threshold of 0.7. Hence, disregarding future NWPws data due to their significant association with future wp would be imprudent. This robust correlation between ws and wp further substantiates the rationality and effectiveness of the boundary module in DKFormer. Upon applying the denoising module to NWP wind speed data, there was an increase in the correlation coefficient between denoised wind speed (ws') and wp to 0.82. This provides additional evidence that the EEMD-based denoising module can effectively capture pertinent information in wind speed closely related to wind power. While other variables exhibit weak correlation with wp, they will also be included as inputs for the DKFormer model.”*

The issue of potential future information leakage is addressed by employing Empirical Mode Decomposition (EEMD) on NWP data, where both historical and forthcoming NWP values are readily available in practical scenarios. Consequently, the risk of data leakage is effectively mitigated. This has been stated in **the last paragraph of Subsection 3.1.1:**

*“The EEMD decomposition is applied to the NWP data obtained from the wind cluster, with the iteration count and symmetric extension count set at 150 and 4 respectively, following previous research [42]. Figure 4 illustrates the normalized NWP signal of length 1500 samples alongside the denoised signal obtained after the decomposition. In practical scenarios, when training the DKFormer model, we can access to both its historical and future values. Therefore, applying the denoising module directly to the entire NWP series does not result in data leakage as long as explicit splitting into a training set and a test set is done.”*

(2) In Figure 2, where does the (y1, y2,... yn) matrix of the prediction part of the model proposed by the author come from? Is this matrix the result of the previous module's processing? And is it a newly introduced feature variable? In addition, besides the decomposition and data normalization parts described by the author, the description of other modules in this model is relatively vague.

**Response:** Thank you for the valuable comment. In the revised manuscript, these ambiguous statements have been amended. Figure 2 has been redrawn to enhance the overall clarity of the model’s flow and please refer to **Figure 2**. In addition, the clipping module and boundary module have been further elaborated upon. Please refer to **Subsection 3.1.2** and **Subsection 3.1.3.**

(3) The upper and lower bounds of the wind speed-power curve are shown in Figure 4. What contribution did the author make to the prediction work in this section?

**Response:** Thank you for the valuable comment. The upper and lower bounds of the wind speed-power curve function as constraints that guide the forecasting process of the DKFormer model. When the predicted wind power value exceeds these boundaries, DKFormer corrects it to the nearest boundary point. This correction mechanism ensures that the model’s predictions align with the expected range of wind power values, thereby enhancing its ability to accurately track and predict wind power over multiple steps using future NWP data. This has been statedin **the third and fourth paragraphs of Subsection 3.1.3:**  
*“**...Utilizing the obtained boundaries, DKFormer integrates the single-step forecasting result Yi into the input of the subsequent step’s forecasting Xi+1, while also considering the constraint Ci+1 imposed by NWP wind speed. The boundary module rectifies the predicted wind power Yi through the following formula:*

*In this process, any predicted result Yi+1 that exceeds the boundaries will be adjusted to the nearest boundary point in order to improve the model’s ability to track multi-step predictions using future NWP data.”*

(4) Section 3.2 of the article is about "transfer learning", but it seems unrelated to the model proposed in the article. The author should focus more on describing this part.

**Response:** Thank you for the valuable comment. Leveraging data from other power plants, transfer learning can effectively enhance the prediction accuracy at the target power plant. This has been stated in **the first paragraph of Subsection 3.2:**

*“Transfer learning is a pivotal machine learning technique that addresses the inherent challenge of limited training data. The pre-training phase facilitates the extraction of intrinsic patterns within wind power sequences from extensive data collected across multiple wind clusters. Subsequently, the fine-tuning phase focuses on capturing the specific characteristics of an individual wind cluster by utilizing its unique dataset. Leveraging data from multiple wind farms for transfer learning offers significant advantages in terms of enhancing both accuracy and stability in model predictions.”*

To further enhance the performance, we incorporated transfer learning into the DKFormer model using our experimental dataset consisting of multiple wind farms, resulting in the proposed TL+DKFormer model. The TL+DKFormer model exhibits superior forecasting capabilities compared to the standalone DKFormer model. Please refer to **the last** **paragraph of Subsection 4.3.2:**

*“To further enhance the prediction accuracy and stability of the model, additional experiments on transfer learning are conducted. We incorporate transfer learning into the DKFormer model to construct the TL+DKFormer model. In this process, one wind cluster is randomly selected as the target data, while the data from the remaining six wind clusters are utilized as the source data. The pre-training phase involves training DKFormer on the source data, which consists of a total of 52,704 data pairs. According to Table 2 results, compared to basic DKFormer without transfer learning, the constructed TL+DKFormer model demonstrates improved forecasting accuracy and stability.”*

(5) The prediction method mentioned in the article is short-term prediction. As far as I know, the standard short-term prediction in China generally refers to 1-3 days, and within 24 hours is Intra-day prediction; The experimental part of the article seems to lean more towards intra-day predictions rather than short-term predictions.

**Response:** Thank you for the valuable comment. In this study, we meticulously replicated the experimental setup and writing style employed in previous works [22, 46] to conduct a comprehensive series experiments encompassing forecasting intervals ranging from 1 hour to 48 hours. In addition, explicit explanations for both intra-day (less than 8 hours) and short-term (8 to 48 hours ahead) forecasting intervals to enhance the clarity of our predictions. Please refer to **the** **first paragraph of Subsection 4.3.2:**

*“Building upon previous studies [22, 46], the multi-step wind power forecasting experiment encompasses both intra-day (less than 8 hours) and short-term (8 to 48 hours ahead) prediction horizons. This approach facilitates an extensive evaluation of the proposed DKFormer model’s performance across various time scales. As shown in Table 2, the results clearly demonstrates that the DKFormer model outperforms other models in terms of accuracy and stability for all forecast horizons.”*

(6) The evaluation indicators section of section 4.1.3, Please refer to the relevant standards of the National Energy Administration of In Multi-step Forecasting experiment, China for the calculation of RMSE and MAE, and these three evaluation indicators are three formulas and cannot be combined into one number.

**Response:** Thank you for the valuable comment. In the revised manuscript, the standards set by the National Energy Administration of China for calculating the R-squared (R2), root mean squared error (RMSE), and mean absolute error (MAE) have been clarified. Units for these three evaluation indicators have been included. the unit for R2 is dimensionless, while the units for MAE and RMSE are kilowatt (kW). Detailed formulas for these indicators have been presented **in the first paragraph of Subsection 4.1.4:**

*“The performance of wind power forecasting models is evaluated in this experiment using commonly employed metrics. The mean absolute error (MAE) and root mean square error (RMSE) are measured in kilowatt (kW), and the R-squared (R2) is dimensionless. The formulas for these metrics are provided as follow:”*

(7) In section 4.2.1, the author used greedy algorithms for hyper-parameter selection. Please provide a more detailed description of this section.

**Response:** Thank you for the valuable comment. The greedy algorithm is a widely employed approach in the selection and validation of hyper-parameters for identifying the appropriate combination within the hyper-parameter space. Detailed implementation steps have been provided in **the first and second paragraphs of Subsection 4.2.1:**

*“The proposed model is trained using the Adam optimizer and the mean squared error (MSE) loss function in this experiment. It should be emphasized that the meticulous selection of hyper-parameters significantly influences the model’s performance and different choices can lead to substantial variations in its effectiveness. To enhance the efficiency of hyper-parameter selection, a greedy line search strategy is employed, building upon a reference baseline configuration described by [30]. In this approach, each hyper-parameter is individually adjusted within predefined ranges while keeping the other parameters constant. The values of hyper-parameters yielding the lowest prediction errors are separately selected. Variation curves are analyzed to identify optimal performance-inducing hyper-parameter settings.*

*The DKFormer model necessitates determination of six pivotal adjustable hyper-parameters: learning rate (α), sequence length (L), number of transformer encoders (Nencoder) and decoders (Ndecoder), dimension of MLP layer (dmlp), and dropout rate (r). The outcomes obtained from our greedy experiment are presented in Figure 8, while the determined values of these hyper-parameters are presented in Table 1. To ensure precise future wind power forecasting, the sequence length (L) is adjusted to 48, enabling the model to assimilate wind power data from the preceding 48 hours.”*

(8) In section 4.2.2, the author analyzed the Pearson correlation coefficients between various numerical weather forecast information and wind power, but did not directly reflect the significance of this work?

**Response:** Thank you for the valuable comment. The Pearson correlation matrix, a widely utilized statistical approach, serves as an effective tool for assessing the linear relationship among variables. The purpose of conducting correlation analysis is to determine the strength of the association between input feature variables and output variable, thereby facilitating feature selection and analysis. This has been stated in **the first and second paragraphs of Subsection 4.1.3**:

*“It is crucial to evaluate the associations between different variables by utilizing the Pearson correlation coefficient (PCC). The PCC matrix, a widely utilized statistical approach, serves as an effective tool for assessing the linear relationship among variables. Represented in a symmetrical matrix format, each element represents the PCC denoting the association between two variables. Ranging from -1 to 1, this coefficient indicates both the strength and direction of the linear correlation. In this study, data from seven wind clusters are integrated to compute the PCC matrix for the entire dataset, as depicted in Figure 8. Typically, a PCC equal to or exceeding 0.7 is considered indicative of a strong linear correlation.*

*According to the results, Figure 8 illustrates a correlation coefficient of 0.78 between wind speed (ws) and wind power (wp), surpassing the threshold of 0.7. Hence, disregarding the future NWPws data due to their significant association with future wp would be imprudent. This robust correlation between ws and wp further substantiates the rationality and effectiveness of DKFormer’ boundary module. Upon applying the denoising module to NWP wind speed data, the correlation coefficient between denoised wind speed (ws') and wp increased to 0.82, providing additional evidence that the EEMD based denoising module can effectively capture the pertinent information in wind speed closely linked to wind power generation. Although other variables exhibit weak correlations with wp, they will also be included as inputs for the DKFormer model.”*

(9) Through Tables 2 and 3, it is found that the prediction models are different. What is the significance of the author’s setting in this way? If the prediction model is different, can it reflect the progressiveness of the model in single step prediction and multi-step prediction?

**Response:** Thank you for the important comment. The purpose of the experiments presented in the original Table 2 is to assess the efficacy of the RELU1 module and ascertain its comparability with other models. In response to your feedback, a new experimental section titled “Module Compatible Evaluation” (Section 4.5) has been conducted not only for RELU1 but also for other modules. This has been stated in **the first paragraph of** **Subsection 4.5:**

*“In order to evaluate the compatibility of the proposed portable modules with other DL models, a series of module compatibility assessments are conducted. The denoising, clipping, and boundary modules are individually integrated into LSTM and GRU models to assess their effectiveness. Following the parameter settings in Section 4.2, the capabilities of these models on both the one-step and day-ahead forecasting scales are evaluated. The results presented in Table 4 clearly demonstrate that incorporating any of these modules leads to significant performance improvements. Particularly, the denoising modules exhibit remarkable efficacy by substantially enhancing performance in both (1-step) and longer-term (day-ahead) forecasting tasks. On the other hand, the boundary module primarily functions effectively when dealing with extended time steps due to potential time delay issues leading to predictions exceeding boundaries. The day-ahead forecasting outcomes suggest that incorporating the boundary module effectively mitigates such time delays.”*

(10) Section 4.4 is an ablation experiment, but the author's writing on this section lacks depth. In addition, the author should add a schematic diagram of the results of the wind power prediction in the ablation experiment section and analyze it.

**Response:** Thank you for the valuable comment. In the revised manuscript, a schematic diagram **(**Figure 14**)** illustrating the outcomes of the wind power prediction in the ablation experiment has been provided, accompanied by a comprehensive and intricate analysis. This has been stated in **the third and fourth paragraphs of Subsection 4.4:**

*“The results of the ablation experiment are presented in Table 3 and Figure 13. Eliminating any of the physical knowledge constraint modules from the DKFormer leads to a deterioration in all three performance metrics of the model, indicating that each module plays a distinct role by effectively incorporating domain knowledge constraints to the data pre-processing, model training, and forecasting output stages. Consequently, the DL model's decisions become more rationalized. The boundary module is primarily designed to address the issue of error accumulation in multi-step forecasting, thus its effectiveness becomes more pronounced with an increase in forecasting stride.*

*In addition, we further illustrate the outcomes of ablation experiments in Figure 14. The figure clearly demonstrates that the phenomenon of time delay is exacerbated when any module is eliminated from DKFormer. This can be attributed to the fact that each module in DKFormer fulfills a distinct role, and removing a module disrupts the seamless functioning of the model, leading to either malfunction or adverse effects on other modules.”*

(11) As a newly proposed prediction model, the running time is also worth considering, and the author should add the training and running time of the model.

**Response:** Thank you for the valuable comment. In the revised manuscript, the running time of models have been discussed. Specifically, DKFormer demonstrates comparable training time to the Transformer model, with each epoch taking 34 seconds, and achieves a swift prediction speed of 30 milliseconds per step, experiencing only a negligible 1-millisecond delay attributed to the boundary module. However, DKFormer outperforms significantly in terms of the forecasting performance. This has been stated in **the last paragraph of Subsection 4.2.1:**

*“When training DKFormer with these parameters, each epoch takes approximately 34 seconds, which is comparable to the training time of the basic Transformer model. In addition, DKFormer achieves a prediction speed of 30 milliseconds per step. Although the inclusion of the boundary module slightly increases the prediction time by 1 millisecond compared to the Transformer model, this marginal difference can be considered negligible in practical scenarios.”*

(12) The conclusion section of the paper is relatively chaotic, without highlighting the key points and specific experimental conclusions of the paper.

**Response:** Thank you for the valuable comment. In the revised manuscript, the conclusion section has been reorganized and the key points have been emphasized. Please refer to **Section 5:**

*“Wind power generation forecasting is crucial for making deployment decisions of wind power plants, especially when faced with limited data. In this paper, we proposed DKFormer, a transformer-based forecasting model that integrates DL with domain knowledge through three portable modules: denoising, clipping, and boundary. These modules played essential roles in data pre-processing, model training, and forecasting stages, respectively. Transfer learning methods were employed to address the issue of limited training data and enhance the accuracy and stability of the forecasts. Furthermore, DKFormer utilized the cubic boundary curves to leverage the future NWP data to reduce time delay and increase prediction accuracy.*

*The experiments conducted on real-life datasets have demonstrated that DKFormer outperforms conventional statistical and DL models in both intra-day and short-term forecasting, particularly for cases with larger time steps. We have observed significant improvements in MAE compared to the vanilla Transformer model, with reductions of 22.0% for 24-hour forecasting. Furthermore, our ablation experiments highlight the indispensability and effectiveness of the three domain knowledge constraint modules within DKFormer. Each module played a unique and vital role in incorporating domain knowledge, thereby further enhancing the forecasting performance of the model.*

*In the future, the integration of the proposed three domain knowledge injection modules into other base models holds great promise due to their high portability and potential for further research exploration. In addition, our paper introduced a pioneering approach in wind power generation models by concurrently leveraging historical wind power data and future NWP wind speed data for forecasting purposes. However, considering the limited forecast horizon of NWP data (typically 48 hours), future research could investigate concurrent prediction of both wind power and wind speed to enable long-term forecasting applications, thus advancing the field.”*

(13) The introduction of the paper only summarizes the prediction model, and there is almost no mention of relevant papers on the processing of numerical weather forecast features.

**Response:** Thank you for the valuable comment. In the revised manuscript, an overview of pertinent literature on feature processing in numerical weather prediction has been included in Introduction Please refer to **the seventh paragraph of Section 1:**

*“Second, there is untapped potential for improvement in harnessing numerical weather prediction (NWP) data and historical wind power information [22, 24, 28–30]. Existing DL models used for time series forecasting primarily rely on sequential extrapolation techniques to predict future wind power based solely on input features of equal lengths. However, the limitations imposed by their model architectures restrict methods like LSTM and Transformer to consider only historical wind power or NWP as inputs. Nevertheless, incorporating NWP data into DL models for sequential extrapolation enables them to learn from past patterns and trends while leveraging additional insights provided by NWP data. Another crucial aspect is the presence of significant noise in NWP data; hence finding a suitable denoising algorithm becomes essential. Wang et al. [35] applied wavelet transform (WT) to the original wind power sequence, resulting in an approximation series and three detailed series through nine vanishing moments and three decomposition levels. However, traditional WT has limitations when it come to handling input signal sizes [36]. Abedinia et al. [37] utilized variational mode decomposition (VMD) method and demonstrated that VMD is prone to mode mixing problems in wind power forecasting tasks. Some studies [36, 38] have successfully employed empirical mode decomposition (EMD) for denoising; however, challenges related to endpoint effect and mode mixing still persist.”*

**Response to Reviewer #2**

**Reviewer #2:**

(1) How do cubic polynomial boundary constraints introduce lead time of NWP data into prediction results? What role does it play in predicting performance improvements?

**Response:** Thank you for the valuable comment. The cubic polynomial boundary function serves as constraints that guide the forecasting process of the DKFormer model. When the predicted wind power value exceeds these boundaries, DKFormer corrects it to the nearest boundary point. This correction mechanism ensures that the model's predictions align with the expected range of wind power values, thereby enhancing its ability to accurately track and predict wind power over multiple steps using future NWP data. This has been stated **in the third and fourth paragraphs of Subsection 3.1.3:**  
*“...Utilizing the obtained boundaries, DKFormer integrates the single-step forecasting result Yi into the input of the subsequent step’s forecasting Xi+1, while also considering the constraint Ci+1 imposed by NWP wind speed. The boundary module rectifies the predicted wind power Yi through the following formula:*

*In this process, any predicted result Yi+1 that exceeds the boundaries will be adjusted to the nearest boundary point in order to improve the model’s ability to track multi-step predictions using future NWP data.”*

(2) In practice, what are the principles for setting EEMD parameters, such as the number of iterations and the number of symmetric extensions? How to affect the denoising effect and prediction accuracy.

**Response:** Thank you for the valuable comment. In practical applications, the selection of EEMD parameters should be adjusted flexibly based on the specific context to ensure a more professional and academically rigorous approach. Previous studies [42] have suggested that setting the iteration count to 150 and the symmetric extension count to 4 in EEMD applied to NWP data is a commonly adopted and suitable choice. Please refer to **the last paragraph of Subsection 3.1.1:**

*“The EEMD decomposition is applied to the NWP data obtained from the wind cluster, with the iteration count and symmetric extension count set at 150 and 4 respectively, following previous research [42].”*

In addition, EEMD improves forecasting outcomes by effectively denoising and extracting features from raw wind speed signals, thereby mitigating inaccuracies caused by noise. This is achieved through an iterative noise-aided processing method that decomposes non-stationary and non-linear signals into independent components. The role of the EEMD as a denoising module has been explained in **the first paragraph of Subsection 3.1.1:**

*“The denoising module, recognized as one of the domain-specific knowledge constraints, is employed to tackle the issue of noise reduction during the data pre-processing stage. Numerical weather prediction wind speed signals often exhibit substantial noise and irregular fluctuations due to factors such as variations in wind direction, terrain obstruction, and sensor errors. These disturbances can significantly undermine the accuracy of forecasting models. To mitigate the impact of noise, we employ ensemble empirical mode decomposition (EEMD) [42], which is a suitable technique for analyzing and processing non-stationary and non-linear signals, to denoise and extract features from raw wind speed signals. This approach involves conducting multiple empirical mode decomposition (EMD) by superimposing Gaussian white noise onto the signal. By leveraging the statistical properties of Gaussian white noise with uniformly distributed frequencies, different iterations introduce white noise with equal amplitude that modifies extreme points in the signal. Subsequently, averaging intrinsic mode functions (IMFs) obtained from multiple iterations effectively cancels out the added white noise. Introducing randomness in each iteration helps disrupt the patterns present in the original signal, rendering each IMF more independent and distinct, thereby addressing the issue of modal aliasing in EMD.”*

(3) In the Boundary Module, how to select boundary points for historical data and ensure that outliers are excluded to ensure the accuracy of the model.

**Response:**Thank you for the valuable comment. The boundary module employs the DBSCAN algorithm to detect outliers in NWP data, excluding them and fitting cubic polynomials with constraints using “cftool” toolbox in MATLAB to ensure accurate model outcomes. Further details on the selection method for boundary points have been provided in the Appendix. Please refer to **the Appendix:**  
*“According to the analysis conducted in Section 3.1.3, the upper and lower boundary functions in the boundary module can be expressed as:*

*To pre-process the data, the widely-used DBSCAN clustering algorithm is employed for outlier removal [47, 48]. The DBSCAN algorithm relies on two key parameters: , which defines the radius of a point's neighborhood, and minPts, which indicates the minimum number of points required for cluster formation. Consistent with previous studies [47, 48], we set as 3 and minPts as 2 in this study.*

*Afterwards, the upper and lower boundary functions, and , can be easily fitted onto the target dataset using the “cftool” toolbox in MATLAB. According to Equation 11, it is worth noting that and should be positive after polynomial expansion. Therefore, a minimum value of 0 is set for both variables and during the auto-fitting process. Additionally, an exponential term has been incorporated as a correction factor to mitigate the impact of forecasting errors. The upper and lower boundary functions can be determined through the following steps.*

1. *Run the DBSCAN algorithm in Matlab with parameters = 3 and minPts = 2. The algorithm will classify each point as either belonging to a cluster or being an outlier.*
2. *Based on the labels generated by the algorithm, eliminate all points identified as outliers (typically labeled -1).*
3. *Obtain the NWP data with outliers removed.*
4. *Import the remaining NWP data points into “cftool” toolbox, select cubic polynomial fitting, and specify a constraint that and should not be less than 0.*

*Consequently, the final upper and lower boundary functions for the target wind cluster are presented in Table A.1, while the fitting result is depicted in Figure 5.”*

(4) It is recommended that the authors explain in more detail the pre-training and fine-tuning process of the global model, including the specific algorithms and parameter Settings used.

**Response:** Thank you for the important suggestion. In the revised manuscript, more details about the pre-training and fine-tuning procedures of the global model have been provided. Please refer to **the last paragraph of Subsection 3.2:**

*“The process of transfer learning on different wind cluster datasets is depicted in Figure 6. Initially, DKFormer employs a large-scale dataset from the source domain for model pre-training. Subsequently, the global model is fine-tuned using data from a specific wind farm. To ensure parameter and size alignment, both the pre-training and fine-tuning processes of the global model adopt identical architecture and hyper-parameter selection as discussed in Section 4.2.1 to mitigate overfitting or underfitting issues. Finally, the resulting model is utilized for predicting wind power generation in the target wind farm for the next 1-48 hours with an aim to enable capturing similarities in data distribution among various wind farms while also capturing unique patterns specific to each target wind farm. Our findings demonstrate that this transfer learning setup offers advantages in terms of prediction accuracy and stability compared to utilizing data solely from a single wind farm only.”*

Furthermore, the data setting of pre-training and fine-tuning process of the global model have been stated in **the last paragraph of Subsection 4.3.2:**

*“To further enhance the prediction accuracy and stability of the model, additional experiments on transfer learning are conducted. We integrated transfer learning into the DKFormer model to construct the TL+DKFormer model. In this process, one wind cluster is randomly selected as the target data, while the data from the remaining six wind clusters are utilized as the source data. The pre-training phase involves training DKFormer on the source data, which consists of a total of 52,704 data pairs. According to Table 2 results, compared to basic DKFormer without transfer learning, the constructed TL+DKFormer model demonstrates improved forecasting accuracy and stability.”*

(5) In the transfer learning section, how to ensure that the model can capture the unique characteristics of the target wind farm, and how to avoid overfitting or underfitting problems?

**Response:** Thank you for the valuable comment. In the transfer learning section, DKFormer is initially trained on a large-scale source dataset and subsequently fine-tuned on the target dataset, enabling efficient acquisition of features specific to the target wind farm. In addition, our approach encompasses appropriate model sizes and parameter settings, as well as consistent model configurations throughout both pre-training and fine-tuning stages of transfer learning. This uniformity in model setup facilitates consistency across different training phases while minimizing risks associated with overfitting or underfitting. This has been stated in **the last paragraph of Subsection 3.2:**

*“The process of transfer learning on different wind cluster datasets is depicted in Figure 6. Initially, DKFormer employs a large-scale dataset from the source domain for model pre-training. Subsequently, the global model is fine-tuned using data from a specific wind farm. To ensure parameter and size alignment, both the pre-training and fine-tuning processes of the global model adopt identical architecture and hyper-parameter selection as discussed in Section 4.2.1 to mitigate overfitting or underfitting issues. Finally, the resulting model is utilized for predicting wind power generation in the target wind farm for the next 1-48 hours with an aim to enable capturing similarities in data distribution among various wind farms while also capturing unique patterns specific to each target wind farm. Our findings demonstrate that this transfer learning setup offers advantages in terms of prediction accuracy and stability compared to utilizing data solely from a single wind farm only.”*

(6) What are the reasons for using the min-max normalization method in the experiment, and are other normalization techniques considered and how they compare to the proposed method?

**Response:** Thank you for the valuable comment. The min-max normalization technique is selected due to its extensive utilization in previous studies and its capability to standardize data within the range of [0, 1]. This approach is favored for its advantages in facilitating the clipping module to eliminate implausible predictions and ensuring adherence of data to established constraints, where wind power and wind speed must remain above 0 and below their theoretical maximum. This has been stated in **the first paragraph of Subsection 4.1.2:**

*“To address variations in data scales and expedite the convergence speed of the model, a normalization technique is employed. In this study, min-max normalization is chosen due to its extensive utilization in previous research [13, 22, 30]. The data, including wind power and wind speed, is normalized within the range of [0, 1]. This normalization approach offers several advantages such as facilitating the application of the proposed clipping module that effectively eliminates unreasonable predictions. Additionally, it ensures adherence to established constraints, where wind power and wind speed are always greater than 0 and less than their theoretical maximum value.”*

(7) In the Ablation Experiment, are there specific metrics to measure the impact of removing each module on the overall performance of the model?

**Response:** Thank you for the valuable comment. In the ablation experiment, the performances of wind power forecasting models are evaluated using commonly employed metrics, including mean absolute error (MAE), root mean square error (RMSE), and R-squared (R2). The results have been stated in **the third paragraph of Subsection 4.4:**

*“The results of the ablation experiment are presented in Table 3 and Figure 13. Eliminating any of the physical knowledge constraint modules from the DKFormer leads to a deterioration in all three performance metrics of the model, indicating that each module plays a distinct role by effectively incorporating domain knowledge constraints into the data pre-processing, model training, and forecasting output stages. Consequently, the decisions made by the DL model become more rationalized. The boundary module is primarily designed to address error accumulation in multi-step forecasting, thus its effectiveness becomes more pronounced with an increase in forecasting stride.”*

(8) Does the author further discuss the portability of the three domain knowledge constraint modules of DKFormer model? Are there plans to apply these modules to other base models?

**Response:** Thank you for the valuable comment. The module compatible evaluation has been conducted in Subsection 4.5 to assess the compatibility of the portable modules with other DL models. Each module has been integrated into LSTM and GRU models for evaluation purposes. The results presented in Table 4 demonstrate significant performance improvements when any of these modules are included for multi-step forecasting. Please refer to **Subsection 4.5.**

**Response to Reviewer #3**

**Reviewer #3:** In this study, a domain-knowledge integrated Transformer (DKFormer) model is proposed for short-term wind power forecasting. The proposed model integrates domain knowledge of wind power generation through three portable modules. The following comments should be solved.

(1) Elaborate on why EMD is adopted and its advantages over other decomposition methods for wind power forecasting.

**Response:** Thank you for the valuable comment. Inevitably, all decomposition methods have certain drawbacks, but we believe that EEMD is more suitable for wind power forecasting. EEMD’s strength lies in its ability to effectively analyze and process non-stationary and non-linear signals by iteratively decomposing signals with added Gaussian white noise, enhancing independence and distinctiveness of the resulting intrinsic mode functions and mitigating modal aliasing issues. Please refer to **the first paragraph of Subsection 3.1.1:**

*“…To mitigate the impact of noise, we employ ensemble empirical mode decomposition (EEMD) [42], which is a suitable technique for analyzing and processing non-stationary and non-linear signals, to denoise and extract features from raw wind speed signals. This approach involves conducting multiple empirical mode decomposition (EMD) by superimposing Gaussian white noise onto the signal. By leveraging the statistical properties of Gaussian white noise with uniformly distributed frequencies, different iterations introduce white noise with equal amplitude that modifies extreme points in the signal. Subsequently, averaging intrinsic mode functions (IMFs) obtained from multiple iterations effectively cancels out the added white noise. Introducing randomness in each iteration helps disrupt the patterns present in the original signal, rendering each IMF more independent and distinct, thereby addressing the issue of modal aliasing in EMD.”*

(2) Explain the rationale for introducing the ReLU1 function and how it ensures that the output does not exceed the maximum power limit.

**Response:** Thank you for the valuable comment. Based on domain knowledge, it is well-established that the power generation of a wind cluster at any given moment should be a non-negative value within the maximum power limit [43]. To ensure compliance with this constraint, the ReLU1 function, which incorporates a maximum clipping mechanism to prevent output from exceeding the specified power limit, is formulated as an extension of the ReLU function. This has been stated in **the first paragraph of Subsection 3.1.2:**

*“**The clipping module integrated into DKFormer plays a crucial role during model training by introducing a new activation function ReLU1 for normalized wind power forecasting, which is inspired by the ReLU function and clips the output to a reasonable range before computing the loss. Based on domain knowledge of wind power generation, it is well-established that the power generation of a wind cluster at any given moment should be a non-negative value within the maximum power limit [43]. Therefore, the predicted output should be constrained by the following equation:*

*where stand for the input of DKFormer and is the original output from the neural network. As depicted in Figure 2, following the acquisition of preliminary predicted values during the model training process, these values are subsequently subjected to the clipping module. This module applies Equation 9 to ensure their adherence to the constraints, specifically being greater than or equal to 0 and less than the maximum observed power of the wind farm. Subsequently, by comparing these updated values with ground truth values, the loss is calculated and model parameters are updated through back-propagation. This approach enables DKFormer to effectively acquire domain knowledge and generate predictions that align with the physical laws.”*

(3) Provide more details on how the boundary module utilizes a cubic polynomial to fit the upper and lower boundary functions. Include the mathematical formulation and steps involved in the fitting process. Explain how historical data is used to determine the upper and lower bounds of the constraint function, and how outliers are handled to ensure the accuracy and reliability of the constraint function.

**Response:** Thank you for the valuable comment. The boundary module employs the DBSCAN algorithm to detect outliers in NWP data, excluding them and fitting cubic polynomials with constraints using “cftool” toolbox in MATLAB to ensure accurate model outcomes. Further details on the selection method for boundary points have been provided in the Appendix. Please refer to **the Appendix:**  
*“According to the analysis conducted in Section 3.1.3, the upper and lower boundary functions in the boundary module can be expressed as:*

*To pre-process the data, the widely-used DBSCAN clustering algorithm is employed for outlier removal [47, 48]. The DBSCAN algorithm relies on two key parameters: , which defines the radius of a point's neighborhood, and minPts, which indicates the minimum number of points required for cluster formation. Consistent with previous studies [47, 48], we set as 3 and minPts as 2 in this study.*

*Afterwards, the upper and lower boundary functions, and , can be easily fitted onto the target dataset using the “cftool” toolbox in MATLAB. According to Equation 11, it is worth noting that and should be positive after polynomial expansion. Therefore, a minimum value of 0 is set for both variables and during the auto-fitting process. Additionally, an exponential term has been incorporated as a correction factor to mitigate the impact of forecasting errors. The upper and lower boundary functions can be determined through the following steps.*

1. *Run the DBSCAN algorithm in Matlab with parameters = 3 and minPts = 2. The algorithm will classify each point as either belonging to a cluster or being an outlier.*
2. *Based on the labels generated by the algorithm, eliminate all points identified as outliers (typically labeled -1).*
3. *Obtain the NWP data with outliers removed.*
4. *Import the remaining NWP data points into “cftool” toolbox, select cubic polynomial fitting, and specify a constraint that and should not be less than 0.*

*Consequently, the final upper and lower boundary functions for the target wind cluster are presented in Table A.1, while the fitting result is depicted in Figure 5.”*

(4) Clearly state the specific application scenarios of the source task (TS) and target task (TT) in DKFormer and how they are interconnected. Describe in detail how DKFormer utilizes data from a specific wind farm for fine-tuning to capture its unique patterns.

**Response:** Thank you for the valuable comment. In DKFormer, The target task (TT) specifically focuses on the data from a randomly selected wind cluster for fine-tuning purposes. The source task (TS) involves utilizing data from the remaining six wind clusters to pre-training the model. These tasks are interconnected through a transfer learning approach where DKFormer is initially pre-trained on the source data to acquire general patterns and subsequently fine-tuned on the target data to adapt and specialize its knowledge, resulting in improved forecasting accuracy and stability in the TL+DKFormer model. This has been stated in **the last paragraph of Subsection 4.3.2:**

*“To further enhance the prediction accuracy and stability of the model, additional experiments on transfer learning are conducted. We incorporate transfer learning into the DKFormer model to construct the TL+DKFormer model. In this process, one wind cluster is randomly selected as the target data, while the data from the remaining six wind clusters are utilized as the source data. The pre-training phase involves training DKFormer on the source data, which consists of a total of 52,704 data pairs. According to Table 2 results, compared to basic DKFormer without transfer learning, the constructed TL+DKFormer model demonstrates improved forecasting accuracy and stability.”*

The same architecture and hyper-parameter selection are adopted for the pre-training and fine-tuning processes of the global model. This has been stated in **the last paragraph of Subsection 3.2:**

*“The process of transfer learning on different wind cluster datasets is depicted in Figure 6. Initially, DKFormer employs a large-scale dataset from the source domain for model pre-training. Subsequently, the global model is fine-tuned using data from a specific wind farm. To ensure parameter and size alignment, both the pre-training and fine-tuning processes of the global model adopt identical architecture and hyper-parameter selection as discussed in Section 4.2.1 to mitigate overfitting or underfitting issues. Finally, the resulting model is utilized for predicting wind power generation in the target wind farm for the next 1-48 hours with an aim to enable capturing similarities in data distribution among various wind farms while also capturing unique patterns specific to each target wind farm. Our findings demonstrate that this transfer learning setup offers advantages in terms of prediction accuracy and stability compared to utilizing data solely from a single wind farm only.”*

(5) Explain the rationale for selecting the Second Wind Clustering Dataset as the target dataset, and elaborate on the choice of the 6:1:3 ratio for dividing it into training, validation, and test sets.

**Response:** Thank you for the important comment. According to Figure 7 and Figure 9, the data distributions across all seven datasets exhibit similarities. Consequently, in this study, one of the seven clusters was randomly designated as the target dataset. Furthermore, we adopted a 6:1:3 ratio for dividing it into training, validation, and test sets following the same approach employed in previous studies. These have been stated in **the first paragraph of Subsection 4.1.1:**

*“The adopted datasets comprise seven wind clusters, each characterized by its own unique data distribution, as illustrated in Figure 7. The historical power measurements are available at an hourly resolution, and the wind power data demonstrates high availability across all wind clusters during the specified period. To ensure confidentiality of commercial secrets, the wind power data is normalized by dividing it by the nominal capacity of each respective wind cluster. Additionally, the provided NWP data includes various meteorological variables, such as longitudinal wind speed (u), radial wind speed (v), total wind speed (ws), and wind direction angle (wd). These weather variables are available from July 1, 2009, to December 31, 2012, with an hourly resolution. For performance evaluation of different models, one wind cluster is randomly selected as the target dataset in accordance with experimental requirements. Following previous work [22], this dataset is partitioned into training, validation, and testing sets in a ratio of 6:1:3 respectively to facilitate the model development and evaluation. Each model undergoes training and validation using distinct data distributions for ten iterations, while being evaluated on a consistent test set.”*

(6) Provide a detailed explanation of how the greedy algorithm is applied for hyper-parameter optimization, outlining the steps involved in tuning and recording the process, which should be elaborated on in Chapter 3.

**Response:** Thank you for the valuable comment. The greedy algorithm is a widely employed approach in the selection and validation of hyper-parameters for identifying the appropriate combination within the hyper-parameter space. Detailed implementation steps have been provided in **the first and second paragraphs of Subsection 4.2.1:**

*“The proposed model is trained using the Adam optimizer and the mean squared error (MSE) loss function in this experiment. It should be emphasized that the meticulous selection of hyper-parameters significantly influences the model’s performance and different choices can lead to substantial variations in its effectiveness. To enhance the efficiency of hyper-parameter selection, a greedy line search strategy is employed, building upon a reference baseline configuration described by [30]. In this approach, each hyper-parameter is individually adjusted within predefined ranges while keeping the other parameters constant. The values of hyper-parameters yielding the lowest prediction errors are separately selected. Variation curves are analyzed to identify optimal performance-inducing hyper-parameter settings.*

*The DKFormer model necessitates determination of six pivotal adjustable hyper-parameters: learning rate (α), sequence length (L), number of transformer encoders (Nencoder) and decoders (Ndecoder), dimension of MLP layer (dmlp), and dropout rate (r). The outcomes obtained from our greedy experiment are presented in Figure 8, while the determined values of these hyper-parameters are presented in Table 1. To ensure precise future wind power forecasting, the sequence length (L) is adjusted to 48, enabling the model to assimilate wind power data from the preceding 48 hours.”*

(7) Specify the specific purpose and significance of using the Pearson correlation coefficient in this study, particularly how it is utilized to evaluate the association of wind cluster data.

**Response:** Thank you for the valuable comment. The Pearson correlation matrix, a widely utilized statistical approach, serves as an effective tool for assessing the linear relationship among variables. The purpose of conducting correlation analysis is to determine the strength of the association between input feature variables and output variable, thereby facilitating feature selection and analysis. This has been stated in **the first and second paragraphs of Subsection 4.1.3**:

*“It is crucial to evaluate the associations between different variables by utilizing the Pearson correlation coefficient (PCC). The PCC matrix, a widely utilized statistical approach, serves as an effective tool for assessing the linear relationship among variables. Represented in a symmetrical matrix format, each element represents the PCC denoting the association between two variables. Ranging from -1 to 1, this coefficient indicates both the strength and direction of the linear correlation. In this study, data from seven wind clusters are integrated to compute the PCC matrix for the entire dataset, as depicted in Figure 8. Typically, a PCC equal to or exceeding 0.7 is considered indicative of a strong linear correlation.*

*According to the results, Figure 8 illustrates a correlation coefficient of 0.78 between wind speed (ws) and wind power (wp), surpassing the threshold of 0.7. Hence, disregarding the future NWPws data due to their significant association with future wp would be imprudent. This robust correlation between ws and wp further substantiates the rationality and effectiveness of DKFormer’ boundary module. Upon applying the denoising module to NWP wind speed data, the correlation coefficient between denoised wind speed (ws') and wp increased to 0.82, providing additional evidence that the EEMD based denoising module can effectively capture the pertinent information in wind speed closely linked to wind power generation. Although other variables exhibit weak correlations with wp, they will also be included as inputs for the DKFormer model.”*

(8) In Figure 11, there is an obvious time delay in the prediction results. The author should explain the reasons for this situation and try to solve the situation.

**Response:** Thank you for the important comment. The time delay in the prediction results is inevitable. Figures 11, 12, and 14 respectively present the outcomes of single-step prediction, multi-step prediction, and ablation experiments. Notably, DKFormer demonstrates a reduced time delay compared to other models due to the incorporation of our proposed domain knowledge modules. Moreover, visual results from the ablation experiments have been provided as additional evidence supporting the significant contributions of all three proposed modules in addressing this issue. This has been stated in **the last paragraph of Subsection 4.4:**

*“In addition, we further illustrate the outcomes of ablation experiments in Figure 14. The figure clearly demonstrates that the phenomenon of time delay is exacerbated when any module is eliminated from DKFormer. This can be attributed to the fact that each module in DKFormer fulfills a distinct role, and removing a module disrupts the seamless functioning of the model, leading to either malfunction or adverse effects on other modules.”*

(9) The authors only present the evaluation results of multi-step prediction results. However, the authors did not present the graph of the multi-step prediction results. The time delay phenomenon is obvious in the single-step prediction results. Then, does the time delay still exist in multi-step prediction? Please give the corresponding prediction curve results and analysis.

**Response:** Thank you for the valuable comment. The presentation has been supplemented with a graph illustrating the results of multi-step predictions as **Figure 12**. In multi-step forecasting, the presence of time delay is common, which becomes more apparent as the number of time steps increases. This phenomenon is inevitable due to the volatility and complexity of wind power data. However, the results indicate that DKFormer exhibits better tracking performance compared to other models, even when dealing with longer time steps. This has been stated in **the fifth paragraph of** **Subsection 4.3.2:**

*“Additionally, the multi-step forecasting results are visualized in Figure 12, showcasing the pronounced advantages in multi-step forecasting. It demonstrates remarkable responsiveness to abrupt changes in wind speed data while exhibiting a significantly reduced time delay effect compared to other baseline models. In contrast, other models merely capture the overall average trend of wind speed variations.”*

(10) Wind energy is highly volatile. The introduction of this paper highlights the good prediction performance of the proposed method in the presence of drastic changes in input data. However, there are no experimental results corresponding to this conclusion in the results of the paper.  
**Response:** Thank you for the valuable comment. Figures 12 and 14 present data with significant fluctuations, along with the corresponding prediction results of each model. The results demonstrate that DKFormer outperforms other models in terms of tracking performance, even when dealing with longer time intervals, thereby resulting in reduced latency. This has been stated in **the fifth paragraph of Subsection 4.3.2:**

*“Additionally, the multi-step forecasting results are visualized in Figure 12, showcasing the pronounced advantages in multi-step forecasting. It demonstrates remarkable responsiveness to abrupt changes in wind speed data while exhibiting a significantly reduced time delay effect compared to other baseline models. In contrast, other models merely capture the overall average trend of wind speed variations.”*

(11) The description of each module in the article is independent. The author should draw a general flowchart of the prediction method. Explain in detail how each module cooperate with each other and the data transmission of each module.

**Response:** Thank you for the valuable comment. In the revised manuscript, the ambiguous statements have been rectified. Figure 2 has been redrawn to enhance the overall clarity of the model’s flow. Additional elaboration has been provided for both the clipping module and boundary module. Please refer to **Subsection 3.1.2** and **Subsection 3.1.3**. Additionally, the extraction process of training samples, forecasting samples, and their corresponding constraints in DKFormer has been illustrated **in the first paragraph of Subsection 3.1:**

*“The architecture of the DKFormer model is depicted in Figure 2. Prior to model training, the raw wind power data undergoes pre-processing using the denoising module. Throughout the training process, constraints are enforced by means of the clipping module, while leveraging NWP data, the boundary module refines and enhances prediction accuracy. The extraction of training samples, forecasting samples, and their corresponding constraints in DKFormer is illustrated in Figure 3.”*

(12) Contributions 4 and 5 are not substantive. They should be in Chapter 5.

**Response:** Thank you for the important suggestion. In the revised manuscript, we have reorganized the contribution section and made revisions to Section 5. Please refer to **the seventh paragraph of Section 1** and **Section 5**.

*“(1) We initially propose the DKFormer forecasting model, which integrates domain knowledge through three constraint modules that are crucial in data pre-processing, model training, and forecasting stages.*

*(2) Unlike conventional sequential extrapolation models that solely rely on historical data for forecasting, our proposed DKFormer model incorporates NWP data’s lead time to guide the forecasting results by constructing a cubic polynomial boundary constraint.*

*(3) We combine transfer learning technique with the DKFormer to enhance the forecasting performance of our model.*

*(4) The proposed domain knowledge constraint modules demonstrate great portability and can be incorporated into various DL baseline models such as LSTM and GRU, resulting in improved forecasting accuracy.”*

*“Wind power generation forecasting is crucial for making deployment decisions of wind power plants, especially when faced with limited data. In this paper, we proposed DKFormer, a transformer-based forecasting model that integrates DL with domain knowledge through three portable modules: denoising, clipping, and boundary. These modules played essential roles in data pre-processing, model training, and forecasting stages, respectively. Transfer learning methods were employed to address the issue of limited training data and enhance the accuracy and stability of the forecasts. Furthermore, DKFormer utilized the cubic boundary curves to leverage the future NWP data to reduce time delay and increase prediction accuracy.*

*The experiments conducted on real-life datasets have demonstrated that DKFormer outperforms conventional statistical and DL models in both intra-day and short-term forecasting, particularly for cases with larger time steps. We have observed significant improvements in MAE compared to the vanilla Transformer model, with reductions of 22.0% for 24-hour forecasting. Furthermore, our ablation experiments highlight the indispensability and effectiveness of the three domain knowledge constraint modules within DKFormer. Each module played a unique and vital role in incorporating domain knowledge, thereby further enhancing the forecasting performance of the model.*

*In the future, the integration of the proposed three domain knowledge injection modules into other base models holds great promise due to their high portability and potential for further research exploration. In addition, our paper introduced a pioneering approach in wind power generation models by concurrently leveraging historical wind power data and future NWP wind speed data for forecasting purposes. However, considering the limited forecast horizon of NWP data (typically 48 hours), future research could investigate concurrent prediction of both wind power and wind speed to enable long-term forecasting applications, thus advancing the field.”*