Dear Reviewer,

Thank you for your effort on review of my submission. Your comments and suggestions are helpful for my current submission and future research. Now, I respond to your comments item-by-item. Your comments in blue and my response in black, yellow represents the modified parts in the manuscript.

1- In Section 2.3 Data Preprocessing, the topic sentence states that K-NN is used to interpolate missing values in environmental driving variables. However, lines 164 – 165 suggest that KNN imputation is applied to estimate missing values in sensible and latent heat fluxes. Clarifying the intended use would improve consistency.

Thank you for your meticulous review of our paper and your valuable comments. We take your concern about the inconsistency in the use of the K-NN method very seriously and would like to clarify this point.

In our study, the K-Nearest Neighbors (K-NN) method is employed in two different scenarios:

Imputation of Missing Values in Environmental Driving Variables: Firstly, we utilize the K-NN method to fill in missing data within the environmental driving variables. These variables are crucial for the training of subsequent machine learning models, and ensuring their completeness enhances the models' performance. We selected K-NN due to its distance-based weighting mechanism, which allows observations most similar to the missing data in feature space to have the greatest impact on the imputation results. Specifically, we chose three neighbors (K=3) to achieve a balance between computational efficiency and imputation accuracy.

As a Comparative Model for Imputing Missing Values of Turbulent Heat Fluxes: Subsequently, we also applied the K-NN method to impute missing values of sensible and latent heat fluxes. In this scenario, K-NN serves as a baseline model against which we compare the performance of other machine learning models such as Support Vector Regression (SVR), XGBoost, Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), Transformer, and Transformer_CNN. Through this comparison, we evaluate the effectiveness of different models in filling the missing values of turbulent heat fluxes. The results indicate that the Transformer model performs the best.

The reason for employing the K-NN method in two different scenarios is that we aim to fully leverage its advantages in handling missing data. In the first scenario, K-NN helps ensure the completeness of the environmental driving variables, providing a reliable data foundation for subsequent models. In the second scenario, K-NN serves as a comparative model to help us assess the relative performance of more advanced models.

2- Line 175, a brief explanation of how random forests rank the contributions of different variables would be useful for readers.

Thank you for your insightful comment regarding Line 175. We agree that providing a brief explanation of how random forests rank the contributions of different variables would be beneficial for readers. In response, we have revised the manuscript to include a concise explanation of the variable importance ranking mechanism used in random forests.

Random forests assess the contribution of each variable to model predictive performance through importance ranking, with variables that contribute significantly to predictive performance receiving higher rankings. This importance is typically calculated by measuring the decrease in node impurity (e.g., Gini impurity or entropy) brought about by splits on each variable across all trees in the forest. Specifically, for each decision tree, the algorithm sums the impurity decrease from splitting on each variable and then averages this decrease over all trees to obtain an overall importance score for each variable. By sorting features based on their importance, random forests select the optimal feature combination, not only effectively reducing the dimensionality of input features but also aiding in the selection of variables within machine learning models. In this study, the number of trees for the random forest model was set to 159 based on ten-fold cross-validation and grid search algorithms, meaning the model consists of 159 decision trees. Since bootstrapping (sampling with replacement) is used to generate random decision trees, not all samples participate in the tree generation process. The unused samples are referred to as out-of-bag (OOB) samples, which can be used to evaluate the accuracy of the trees. OOB scores provide an unbiased estimate of the model's generalization ability by effectively assessing the model’s capability to predict unknown data. The higher the OOB score, the stronger the model's generalization capability (Wang et al., 2023)

3- Line 315-317. Could the authors expand on the concept of multi-scale interactions and long-distance dependencies? For instance, are these related to temporal dimensions, or do they involve correlations between driving variables?

Thank you for your thoughtful comment regarding Lines 315-317. We appreciate the opportunity to elaborate on the concepts of multi-scale interactions and long-distance dependencies in our study.

Multi-Scale Interactions:
Multi-Scale in the Temporal Dimension: The variations in turbulent heat fluxes are influenced by physical processes occurring at different temporal scales, including instantaneous moments, diurnal cycles, seasonal changes, and interannual variations. For example, the diurnal variation of solar radiation affects surface temperatures, which in turn influence sensible and latent heat fluxes; seasonal changes in vegetation can alter surface characteristics, impacting turbulent exchange processes. Capturing these variations across different temporal scales is crucial for accurately predicting turbulent heat fluxes.

Long-Term Dependencies:
Long-Term Dependencies in Time Series: This refers to the phenomenon in time series data where the current turbulent heat flux is influenced by states at earlier time points. For instance, changes in soil moisture can affect latent heat fluxes over subsequent days or even longer periods. Traditional models may struggle to capture such dependencies over extended time intervals.

Correlation with Driving Variables:
Interactions Across Variables: Turbulent heat fluxes are not solely influenced by individual variables but result from complex nonlinear interactions among multiple environmental driving variables (such as wind speed, temperature, humidity, and radiation). Nonlinear and higher-order correlations may exist among these variables, and these correlations may manifest differently at various temporal scales.

How the Transformer_CNN Model Captures These Characteristics:
Transformer Component: Utilizing the self-attention mechanism, the Transformer effectively captures long-term dependencies in time series data. It can dynamically focus on and weigh information from different past time points when predicting the current turbulent heat flux, thereby capturing influences over long time spans (Vaswani et al., 2017).

CNN Component: Convolutional Neural Networks (CNNs) excel at extracting local features and can capture short-term patterns and local variations in the data, such as diurnal changes and seasonal cycles (Krizhevsky et al., 2012). Through convolution operations, CNNs can efficiently identify locally correlated patterns in time series.
Advantages of Model Fusion: Combining the Transformer and CNN allows simultaneous capture of multi-scale temporal features and complex interactions across variables in turbulent heat fluxes. This fusion approach helps the model to more comprehensively understand the influence of driving variables such as wind speed, temperature, and radiation on turbulent heat fluxes, thereby improving prediction accuracy and the model's generalization capability.

4- Line 225, which traditional statistical method was used to generate the test dataset? Additionally, in line 359, if a traditional statistical method serves as a reference, is it appropriate to compare it with machine learning models, and why do the machine learning approaches outperform the reference dataset?

Thank you for your insightful comments and for highlighting these important points. We apologize for any confusion caused by our previous wording, and we have revised the manuscript to clarify these issues.

As for comparing the two, the primary reasons are as follows:

Baseline Comparison: Traditional statistical methods serve as a baseline to evaluate the performance of more advanced models. By comparing machine learning methods with traditional approaches, we can quantify the improvements and demonstrate the benefits of using more complex models, which aligns with the suggestions of previous reviewers.

Enhanced Performance: Machine learning models, such as neural networks, have the ability to capture nonlinear relationships and interactions between variables that traditional linear models may overlook. This enables them to more effectively model the inherent complexities present in soil turbulent heat flux data.

Handling Data Complexity: Soil turbulent heat fluxes are influenced by various environmental factors that interact in complex and nonlinear ways. Machine learning models are better equipped to handle this complexity, leading to improved imputation accuracy and predictive performance.

Empirical Results: Our results show that machine learning methods outperform the traditional statistical method in key evaluation metrics such as Mean Squared Error (MSE) and the coefficient of determination (R^2). This indicates that machine learning models provide a more accurate and reliable approach for imputing missing data and predicting soil turbulent heat fluxes.

By providing this comparison, our aim is to highlight the superiority of machine learning models over traditional methods in time series imputation and prediction. We believe this strengthens the case for adopting advanced modeling techniques in environmental data analysis.

5- Table 4, could the statistics for the Transformer_CNN be included?

We appreciate your suggestion to include the statistics for the Transformer_CNN model in Table 4. Our intention with Table 4 was to demonstrate that the Transformer model significantly outperforms the other baseline models. By focusing on this comparison, we aim to highlight the substantial improvement achieved by the Transformer architecture over other models.

We chose not to include the Transformer_CNN model's statistics in Table 4 to maintain the coherence and logical flow of the manuscript. Our rationale is as follows:

Progressive Presentation of Results: By first establishing the effectiveness of the Transformer model compared to existing baseline models, we set a foundation for introducing our proposed enhancement—the Transformer_CNN model—in subsequent sections.

Emphasizing Incremental Improvements: Presenting the Transformer_CNN results separately allows us to clearly demonstrate the additional benefits gained by integrating the CNN component with the Transformer architecture. This stepwise progression helps readers appreciate the incremental advancements and the specific contribution of our proposed model.

In the subsequent sections, we provide a comprehensive evaluation of the Transformer_CNN model, including its performance metrics and a comparison with the Transformer model. This allows us to thoroughly discuss how the addition of the CNN component leads to further improvements in imputation accuracy.

We hope that this explanation clarifies our reasoning for the current presentation format. However, we are open to reconsidering if you feel that including the Transformer_CNN statistics in Table 4 would significantly enhance the clarity or value of our findings.

6- Fig 5, the panels are not labeled.

Thank you very much for your feedback. We have corrected this figure.
The black dashed line represents the diagonal (1:1 line). The red line denotes the fitted regression line between the actual and estimated values. The color bar represent the density of data points; areas with redder hues correspond to regions of higher data density.

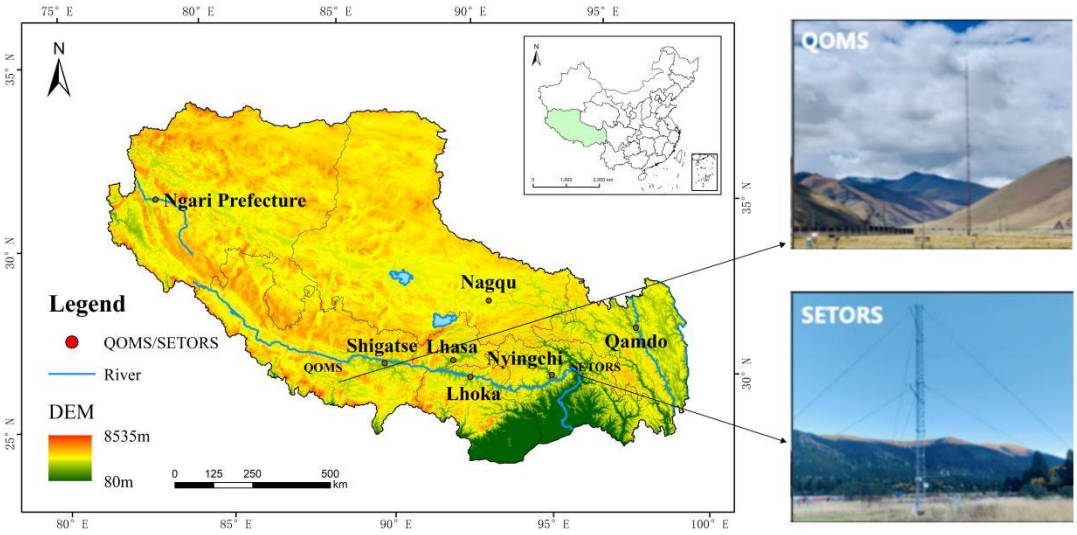
7- For the published data, it would be helpful to have a QC indicator of measured and estimated values.

Thank you for your valuable suggestion regarding the inclusion of quality control (QC) indicators for the measured and estimated values in the published data.In our study, we imputed the missing values of sensible heat flux and latent heat flux at the QOMS station. We have provided quality control indicators—Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and the coefficient of determination (R^2)—for the measured and estimated values in Figure 5 of the manuscript. These metrics demonstrate the accuracy and reliability of our imputation methods by quantifying the agreement between the estimated values and the actual measurements.

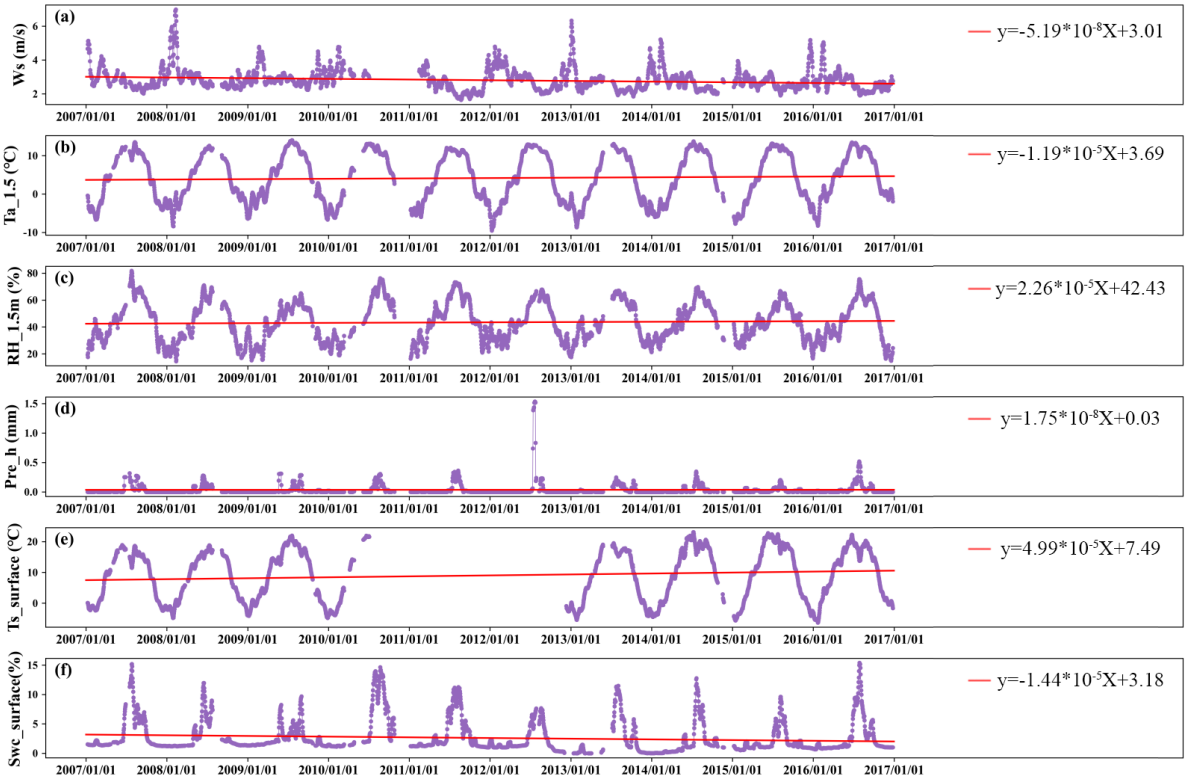
For the published dataset, we have retained all the original measured values and have imputed the missing data to create a continuous, high-quality, and complete time series of sensible and latent heat fluxes. By including both the original measurements and the imputed values, along with the associated QC indicators, we provide users with comprehensive information to assess the quality and reliability of the data.

8- Figure1 is not clear. I would suggest recreating it. The authors could use some open-source library to denote the location of the site and the elevation of the Tibet area.

Thank you very much for your feedback. We have corrected this figure.



9- Figure2. Add the description of the horizontal line, which could be the linear trend or the MK equation as in Table 1 for each variable. It would be helpful to denote the slope (e.g., degree per decade). As for Table 1, what's the fitting equation of the MK test? Is MK test only a statistical test without providing the equations? Are these equations estimated from least squares regression or Theil–Sen estimator? Please add more clarification here.



The Mann-Kendall (MK) test is a non-parametric statistical method used to detect the presence and significance of monotonic trends in time series data without assuming any specific data

distribution. The MK test itself does not provide a fitting equation or estimate the slope of the trend line; it primarily assesses whether a statistically significant trend exists.

To quantify the magnitude of the trends identified by the MK test, we concurrently applied least squares linear regression to calculate the slope and intercept for each variable. This approach allows us to obtain specific linear equations representing the trends by minimizing the sum of the squared differences between the observed values and the values predicted by the linear model.

10- Line 184. “Before training the model…”. Please clarify which model. KNN and Random forest are also ML models.

In Line 184, when we stated "Before training the model...", we were referring to the training process for all the machine learning models utilized in our study. This includes the K-Nearest Neighbors (KNN), Support Vector Regression (SVR), XGBoost, Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), Transformer, Convolutional Neural Network (CNN), and Transformer_CNN models.

Our intention was to indicate that certain preprocessing steps were performed uniformly before training each of these models. These steps ensure data consistency and optimal model performance across the different algorithms.

11- Following the comments from previous reviewers, the use of KNN and RandomForest requires further justification. If my understanding is correct, KNN is first used to fill missing values (line 157) with the number of nearest neighbors setting to 3. Then it is used as a comparison to all the other DL models (e.g., LSTM, Transformer). This seems like a two-step KNN. Please clarify it in the text. In addition, LSTM handles time-series as input, whereas RandomForest does not take time dependency into consideration (theoretically we can input a time series into RandomForest, but from Figure 3, it seems each variable is used for one time step). How can the feature importance from this RF can be used to compare with LSTM (e.g., if Prec_h has a strong effect but with a time lag, it will not show in Figure3 but actually is important for LSTM)?

1. Clarification on the Two-Step Use of KNN:

You are correct in your understanding that KNN is employed in two distinct steps in our research:

First Application: We use KNN (with K=3) to impute missing values in the environmental driving variables. These variables are critical inputs for our models, and ensuring their completeness is essential for accurate modeling. We chose K=3 to strike a balance between imputation accuracy and computational efficiency.

Second Application: KNN is also used as a comparative model for imputing the missing values of turbulent heat flux data (sensible and latent heat fluxes). In this context, KNN serves as a baseline model against which we compare the performance of other machine learning and deep learning

models, such as Support Vector Regression (SVR), XGBoost, Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), Transformer, and Transformer_CNN.

We acknowledge that this two-step use of KNN may cause confusion. To address this, we will revise the manuscript to explicitly clarify the distinct roles of KNN in both steps. This clarification will help readers understand that the first use of KNN is for data preprocessing (imputing missing environmental variables), while the second use is as a baseline model in our comparative analysis.

2. Use of Random Forest for Feature Selection and Its Relation to Time-Series Models:

In our study, Random Forest is utilized primarily for feature selection rather than for direct comparison with models like LSTM. The reasons for using Random Forest for feature selection are:

Dimensionality Reduction: Random Forest helps reduce the number of input features by identifying those that have the most significant impact on the model's predictive performance. This reduction in feature dimensionality improves training efficiency and computational speed for subsequent models.

Mitigating Overfitting: By removing unimportant or redundant features, Random Forest decreases the risk of overfitting, enhancing the model's generalization ability on unseen data.

Understanding Feature Importance: Random Forest provides estimates of feature importance by evaluating how each variable contributes to reducing prediction error. This insight deepens our understanding of the factors influencing turbulent heat fluxes.

12- In addition, if RandomForest has already chosen to predict heat fluxes, why not use it as a benchmark model for the following comparisons? If its skill is not optimal, why would we trust its feature importance results?

1. Use of Random Forest for Feature Selection vs. Benchmarking:

Our study primarily aims to investigate the imputation effects of deep learning techniques on turbulent heat fluxes. While Random Forest is a robust machine learning method, we did not include it as a benchmark model for direct comparison with the deep learning models. The reasons are as follows:

Focus on Deep Learning Techniques: Our objective was to explore and highlight the capabilities of deep learning models (such as LSTM, GRU, Transformer, and Transformer_CNN) in handling the complex, nonlinear, and temporal dynamics of turbulent heat flux data.

Scope of the Study: Including every possible machine learning method was beyond the scope of this paper. Instead, we selected a representative set of models to compare, including traditional methods like KNN and SVR, and advanced models like XGBoost and various deep learning architectures.

Future Work: We acknowledge that including Random Forest as a benchmark could provide additional insights. We plan to consider this in future research to further enrich the comparative analysis.

2. Trusting Feature Importance Results from Random Forest:

Despite not using Random Forest as a predictive benchmark, we employed it for feature selection due to its strengths in this area:

Effective Dimensionality Reduction: Random Forest excels at evaluating feature importance, allowing us to identify the most influential variables that contribute significantly to model performance. This reduces the dimensionality of the input data, leading to more efficient training and faster convergence in deep learning models.

Mitigating Overfitting: By eliminating unimportant or redundant features, we decrease the risk of overfitting, thereby enhancing the generalization capability of our deep learning models on unseen data.

Robust Feature Importance Measures: Random Forest provides reliable estimates of feature importance through mechanisms like Gini importance or permutation importance. These measures are valuable even if Random Forest is not the optimal predictor in terms of overall performance metrics like RMSE or R^2 .

Complementarity with Deep Learning Models: While Random Forest may not capture temporal dependencies as effectively as models like LSTM or Transformer, it offers a different perspective by evaluating the immediate impact of features on the target variable. This information complements the deep learning models, which can then focus on learning complex temporal patterns with a refined set of input features.

The method of using Random Forest for feature selection has been validated and demonstrated in the study by Wang et al(Wang et al., 2023).

13- Line 216. Is SVM used as a classifier or a regressor here?

We used Support Vector Regression (SVR), which is the regression variant of SVM, to model and predict the turbulent heat fluxes.

14- Line 269. Is the CNN layer also used before LSTM and GRU? Please clarify it in the model description section.

Thank you for your valuable comment and for highlighting the need for clarification regarding the use of the CNN layer in our models.

Clarification on the Use of the CNN Layer:

In our study, the CNN layer is only combined with the Transformer model, resulting in the Transformer_CNN architecture. We did not use the CNN layer before the LSTM or GRU models.

The reason for combining CNN and Transformer is logical:

Initial Model Evaluation: We first evaluated several models—including KNN, SVR, XGBoost, LSTM, GRU, and Transformer—to determine their effectiveness in imputing missing values of turbulent heat fluxes.

Performance of the Transformer Model: The Transformer model demonstrated superior performance compared to the other models, exhibiting higher accuracy and better generalization capabilities.

Enhancing the Transformer Model: To further improve the performance of the Transformer model, we integrated a CNN layer before the Transformer encoder. The CNN component is adept at capturing local temporal patterns and extracting high-level features from the input data, which complements the Transformer's strength in modeling long-range dependencies.

15- I may miss some parts, but what’ s the input time window size to the ML and DL models

(i.e., how many time steps are used as input? What’s the sequence length)? For traditional ML models (SVM etc.,), do they take input variables as a time series, or at a concurrent time step? If the latter, they have less information than the RNN models, which is not a fair comparison.

Thank you for your insightful comments and for giving us the opportunity to clarify these aspects of our study.

To ensure a fair comparison, we standardized the input data across all models. Specifically, both traditional ML models (such as Support Vector Machines [SVM] and Random Forest) and DL models (including Long Short-Term Memory [LSTM], Gated Recurrent Unit [GRU], and Transformer) received the same input features, which consisted only of the variables at the current time step. We did not include data from previous time steps in the input features for any of the models. This approach means that each sample is an independent feature vector with a sequence length of one.

By doing so, we ensured that all models operated under identical conditions and had access to the same amount of information. This methodology allows us to directly compare the models' abilities to capture complex relationships between the input features and the target variable at a single time step, without the influence of differing input sequences.

Similar methodologies have been applied and validated in previous studies, demonstrating the effectiveness of DL models even when the sequence length is one (Bai et al., 2018; Borovykh et al., 2017; Schuster & Paliwal, 1997).

16- Following the previous comment, are we looking at hourly predictions in Fig 6 and daily in Fig 8? Please clarify it in the caption.

Sorry, I didn't understand. Are you referring to Figure 7
Figure 6 displays the model's predictions at an hourly resolution.
Figure 7 presents the predictions aggregated at a daily resolution.
We agree that clarifying the temporal resolution in the figure captions will enhance the readers' understanding. We will update the captions accordingly in the revised manuscript to explicitly state the time scale of the predictions shown in each figure.

17- Fig4. What does “view” represent here? Does it mean the model has two outputs? In the responses, the authors mentioned “contrastive learning”, which is not in the main text. If this is “contrastive learning”, a reference is needed here.

Thank you for your question regarding the term "view" in Figure 4. In our model, "views" refer to two outputs generated during the forward propagation process—specifically, F1 (primary view) and F2 (contrast view). This occurs because the model's forward function is invoked twice in each training iteration, and due to the stochastic nature of dropout layers, these outputs may differ. Although the model ultimately produces two views, they are not independent outputs but are used within a contrastive learning framework to enhance the model's robustness. Our loss function is the Smooth L1 Loss, comprising three components: the loss between F1 and the true values, the loss between F2 and the true values, and a regularization term (multiplied by 0.1) representing the distance between F1 and F2. During inference, we retain dropout to account for uncertainty and invoke the model twice to generate F1 and F2. The final prediction is obtained by averaging these two outputs, which helps reduce the prediction uncertainty caused by dropout and stabilizes the results. We have updated the manuscript to clarify the definition of "view," explained how

contrastive learning is applied in our model, and provided an appropriate reference to address your concerns.

The forward propagation process involves invoking the feed-forward function twice during each training iteration, resulting in two different outputs due to the stochastic nature of dropout layers. These outputs are referred to as two data "views": F1 (primary view) and F2 (contrast view).

18- In the authors responses, a cross-validation test is added. However, instead of separating the predictions by years, the authors should use the concatenated prediction to assess the performance (i.e., predictions from all years vs target from all years).

Thank you for your insightful comment and for giving us the opportunity to clarify our evaluation methodology.Regarding your suggestion to use concatenated predictions from all years to assess the model's performance, we would like to explain our approach:

Following the recommendations from previous reviewers, we have adopted a standard practice in deep learning by splitting our dataset into training, validation, and test sets. Specifically, the validation set is not involved in the training process but is used to fine-tune hyperparameters and prevent overfitting, thereby providing a more accurate assessment of the model's performance on unseen data.

Our data partitioning is as follows:
Training Set: Data from the years 2007 – 2011 and 2013 – 2016, with 10% of this data randomly selected to form the validation set.
Validation Set: Used exclusively for model tuning and selection without influencing the training process.
Test Set: Data from the year 2012, reserved for evaluating the final performance of the model.
We selected 2012 as the test set to simulate the model's performance in practical applications and to evaluate its generalization capability on data from an unseen year. This approach aligns with best practices in deep learning and mirrors real-world scenarios where models are often applied to future or previously unseen time periods.

By structuring our data split in this manner, we aim to provide a robust evaluation of the model's performance, ensuring that the results are both reliable and reflective of its ability to generalize to new data.We believe that our current data partitioning and evaluation methodology more accurately reflect the model's performance in practical settings. However, we appreciate your suggestion and are open to discussing alternative evaluation strategies. If you have further questions or recommendations, we would be happy to consider them and adjust our approach accordingly.

19- not sure I understand here, as the data is from 2007-2016, what is the relationship for this experiment in 2012?

I apologize for any confusion caused. The experiment initiated in 2012 incorporated data from previous trials, thereby providing data support for this study.I will also modify it.

20- also put the average here (not only in the text) for the better reading

	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	average
gap_H	39.4%	10.3%	22.2%	9.8%	32.2%	29.6%	11.7%	10.4%	18.1%	33.3%	21.7%
gap_LE	37.65%	8.28%	21.30%	8.48%	23.63%	28.52%	9.90%	8.84%	33.57%	33.34%	21.4%

21- this is also a part of the machine learning method, right?

Yes, that's correct. Recurrent Neural Networks (RNNs) are indeed a part of machine learning methods.

22- neutral networks (LSTM, GRU) and deep learning model are also part of the machine learning methods? right?

Thank you for your insightful comment. You are absolutely correct that neural networks, including Long Short-Term Memory (LSTM) networks, Gated Recurrent Units (GRU), and deep learning models like the Transformer, are all part of machine learning methods. In our manuscript, our intention was to distinguish between traditional machine learning algorithms and deep learning models due to their differing architectures and capacities for handling complex data patterns.

Once again, we sincerely appreciate your insightful comments, which have undoubtedly strengthened the quality of our work. We have made the necessary revisions based on your suggestions, and the improved manuscript now better meets the journal's requirements.

Best regards,

Sincerely

Quanzhe Hou, Zhiqiu Gao, Zexia Duan, and Minghui Yu

October 20th, 2024

References:

Wang, L., Wan, B., Zhou, S., Sun, H., and Gao, Z.: Forecasting tropical cyclone tracks in the northwestern Pacific based on a deep-learning model, Geosci. Model Dev., 16, 2167 – 2179, https://doi.org/10.5194/gmd-16-2167-2023, 2023.

Bai, S., Kolter, J. Z., and Koltun, V.: An empirical evaluation of generic convolutional and recurrent networks for sequence modeling, arXiv preprint, arXiv:1803.01271, https://arxiv.org/abs/1803.01271, 2018.

Borovykh, A., Bohte, S., and Oosterlee, C. W.: Conditional time series forecasting with convolutional neural networks, arXiv preprint, arXiv:1703.04691, https://arxiv.org/abs/1703.04691, 2017.

Schuster, M., and Paliwal, K. K.: Bidirectional recurrent neural networks, IEEE Trans. Signal Process., 45, 2673 – 2681, https://doi.org/10.1109/78.650093, 1997.