# Supporting Information for "Improving machine learning-based weather forecast post-processing with clustering and transfer learning"

Yuwen Chen[1], Xiaomeng Huang[1,2,3], Yi Li[1,3], Yue Chen[1], Chi Yan Tsui[3],

Xing Huang[1,3], Mingqing Wang[1,3], Jonathon Wright[1]

[1]Ministry of Education Key Laboratory for Earth System Modeling, Department of Earth System Science, Tsinghua University, Beijing 100084, China

[2]Laboratory for Regional Oceanography and Numerical Modeling, Qingdao National Laboratory for Marine Science and Technology, Qingdao, 266237, China

[3]National Supercomputing Center in Wuxi, Wuxi, 214011, China

## Contents of this file



## Introduction

Supporting information for the manuscript "Improving machine learning based weather forecast post-processing with clustering and transfer learning" is provided here. Text S1 introduces the calculation of Average Silhouette Coefficient (ASC), Text S2 introduces

details of the LightGBM as applied in this work, and Text S3 introduces the alternative machine learning frameworks used for comparison with the LightGBM results. Figure S1 shows a flow chart of our CDT framework, Figure S2 shows an example tree structure based on LightGBM, and Figure S3 and Figure S4 illustrate the structures of the LSTM-FCN and ANN models, respectively.

**Text S1: Average Silhouette Coefficient**

In this paper, we use the average Silhouette coefficient (ASC) as a guide to find viable candidates for the clustering number $K$. The ASC is calculated via the following steps:

**(1)** For data point $i$ in cluster $C_i$, define

$$a(i) = \frac{1}{|C_i| - 1} \sum_{j \in C_i, i \neq j} d(i, j) \tag{1}$$

as the average distance from point $i$ to the other points in cluster $C_i$, where $d(i, j)$ is the distance between point $i$ and point $j$;

**(2)** Define

$$b(i) = \min_{k \neq i} \frac{1}{|C_k|} \sum_{j \in C_k} d(i, j) \tag{2}$$

as the smallest average distance of point $i$ to all points in any other cluster;

**(3)** Define the Silhouette of point $i$ as

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}; \tag{3}$$

**(4)** Define the ASC as the mean Silhouette of all points.

**Text S2: LightGBM**

We use the LightGBM model to post-process ECMWF temperature forecasts. Figure S2 provides an illustrative example of the LightGBM decision tree structure in the CDT

framework. Split features include ECMWF-predicted temperature (EC) and observations of temperature ($T$), pressure($p$), wind speed (WS) and relative humidity (RH) measured at previous time steps. The path selects different branches in sequence depending on the split conditions, with the leaf value (ovals) returned as the final result. For the illustrated decision tree (Fig. S2), the value of leaf 10 ($-0.176$) is calculated using a series of criteria from the leftmost node $EC \leq 6.445$ to the rightmost node $T(t-1) \leq -0.450$). LightGBM provides the added advantages of rapid training speed, low memory usage, high accuracy, and parallel learning capacity.

LightGBM only supports two-dimensional structured datasets. Therefore, the observations are converted from the three-dimensional shape $(n_{\text{samples}}, n_{\text{steps}}, n_{\text{features}})$ to the two-dimensional shape $(n_{\text{samples}}, n_{\text{steps}} \times n_{\text{features}})$. When combined with the ECWMF data, latitude, longitude and elevation of stations, the final training data is organized in the shape $(n_{\text{samples}}, n_{\text{steps}} \times n_{\text{features}} + 8)$. LightGBM predictions represent the results of boosting all trees. The RMSE between the predicted result and observations collected at the valid forecast time is used to evaluate the prediction. To control overfitting, we tune the maximum tree depth to eight in this paper. This hyper-parameter provides the maximum depth that each tree is allowed to have. A smaller value indicates a weaker predictor.

**Text S3: LSTM-FCN and ANN**

Long-Short-Term-Memory (LSTM) models are widely used in time series prediction. In this paper we exploit temporal auto-correlation in observational time series in constructing a two-layer LSTM. The input is formatted as a three-dimensional array of the shape

($n_{\text{samples}}, n_{\text{steps}}, n_{\text{features}}$), and the LSTM output is merged with ECMWF forecasts using a four-layer, fully connected network (FCN). This deep neural network is then used as a control model. Fig. S3 shows the structure of our LSTM-FCN model.

As the most basic neural network frameworks, Artificial Neural Networks (ANNs) are also widely used for time series prediction. In this paper, we use a four-layer ANN as the control model. The input data to the ANN are the same as those provided to the LightGBM. Fig. S4 shows the structure of our ANN model.

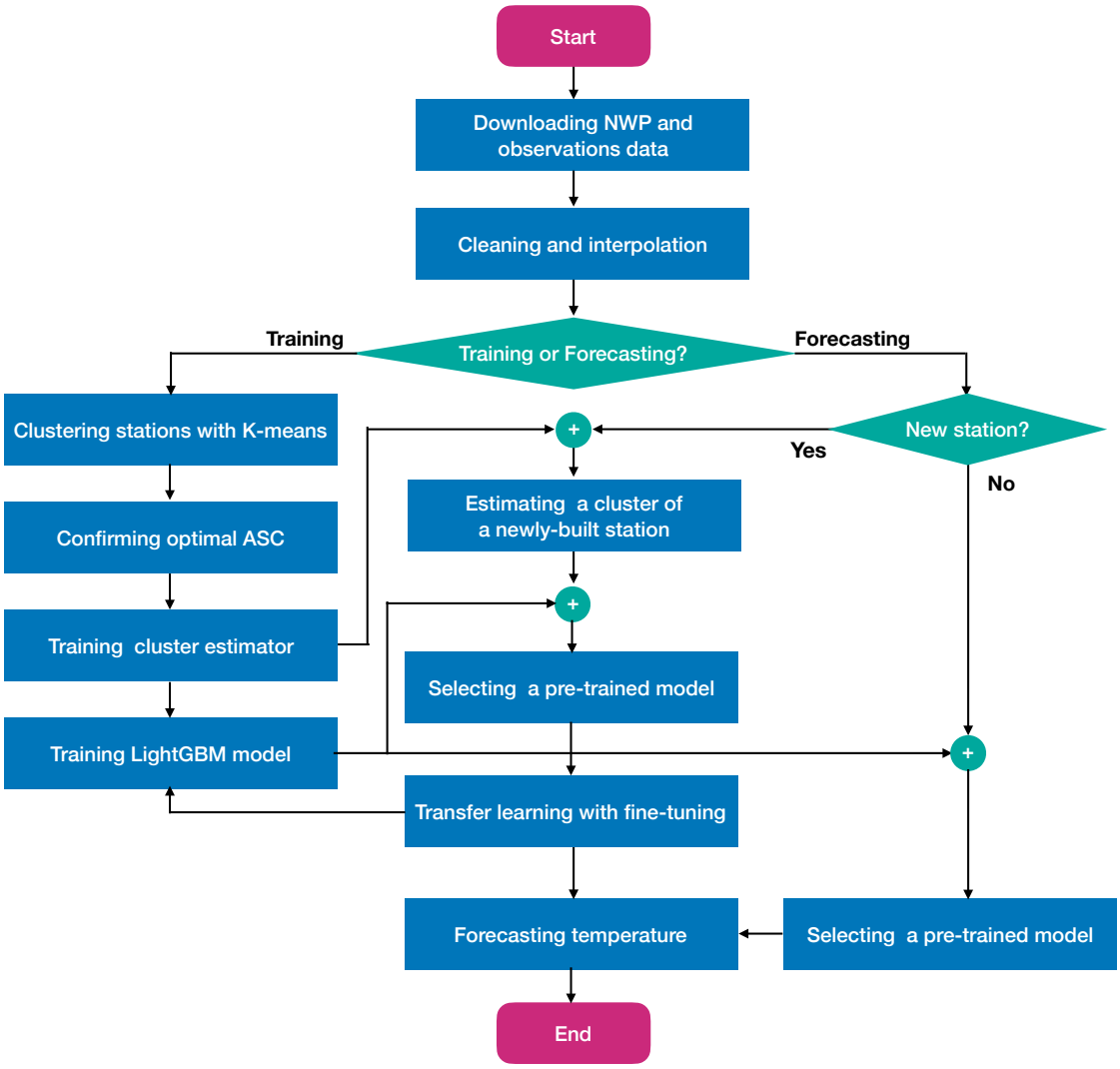


**Figure S1.** A flow chart of our CDT framework. Starting from the top, the NWP forecasts and observational data are downloaded and pre-processed. The training branch builds a 'cluster estimator' and groups the existing stations into clusters. Separate LightGBM models are then trained for each cluster and each lead time. Post-processed forecasts for existing stations are generated using the pre-trained model for the corresponding cluster. New stations are grouped into the best-fit existing cluster, after which the corresponding LightGBM model is fine-tuned to produce the final forecast.

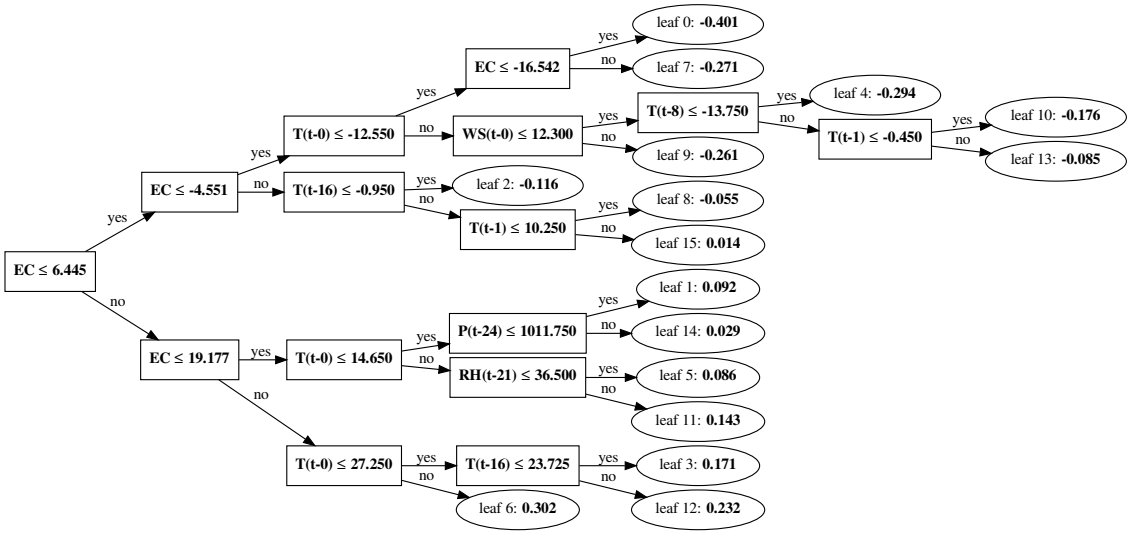


**Figure S2.** An example LightGBM decision tree. The features comprise the ECMWF result and observations of temperature ($T$), relative humidity (RH), pressure ($p$), and wind speed (WS) from a specified number of preceding time steps. Split features marked EC refer to the ECMWF prediction; $T(t-1)$ refers to the observed temperature one time step prior; $p(t-24)$ refers to the observed pressure 24 time steps prior; $\mathrm{WS}(t-0)$ refers to the observed wind speed at the current time; and $\mathrm{RH}(t-21)$ refers to the observed relative humidity 21 time steps prior. A LightGBM model consists of multiple such decision trees, and a LightGBM prediction is the result of boosting the returned leaf values from all trees.

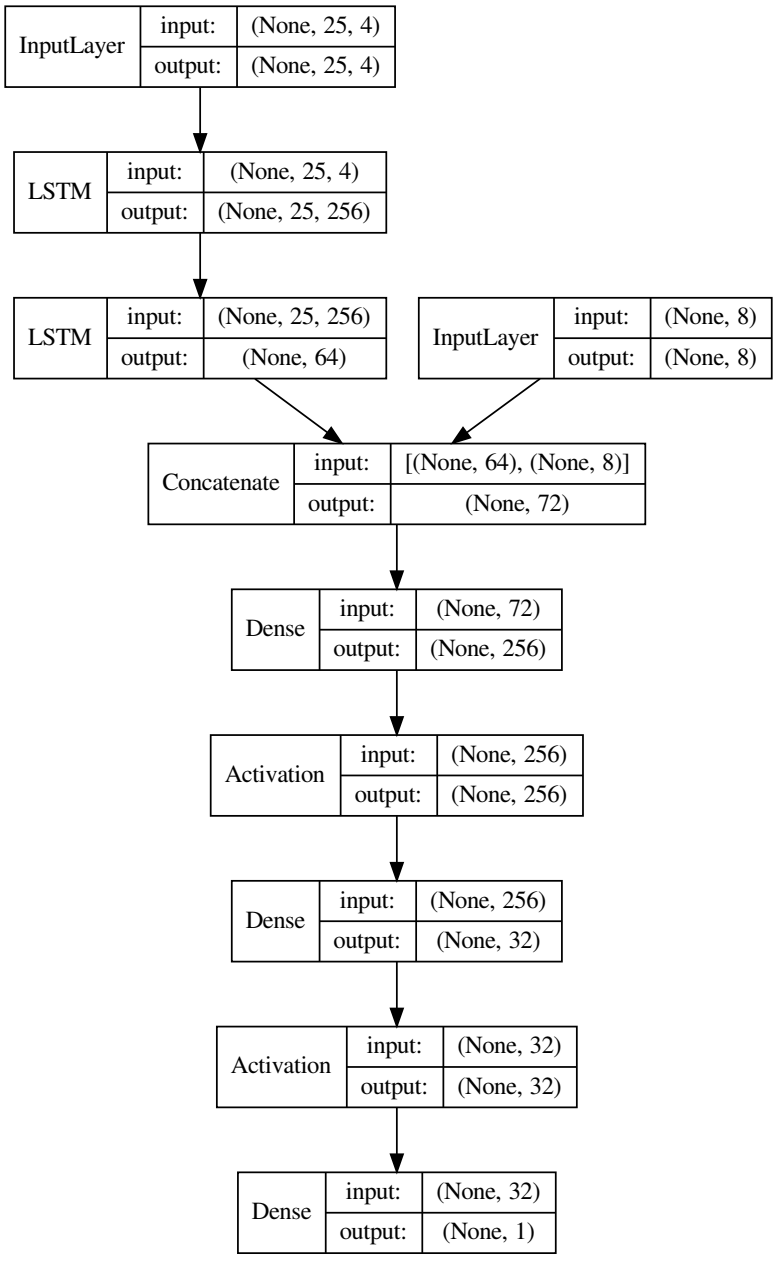


**Figure S3.** The structure of the LSTM-FCN model. The input consists of two parts, the time series of the observed data (left), and the inverse distance weighted (IDW) NWP result (right). The left input is in the shape of $(n_{\text{samples}}, n_{\text{steps}}, n_{\text{features}})$, where $n_{\text{steps}}$ equals to 25, $n_{\text{features}}$ equals to 4. The right input is in the shape of $(n_{\text{samples}}, 8)$, where the number 8 means 5 variables from ECMWF forecasts, and 3 location information of the stations (latitude, longitude and elevation).

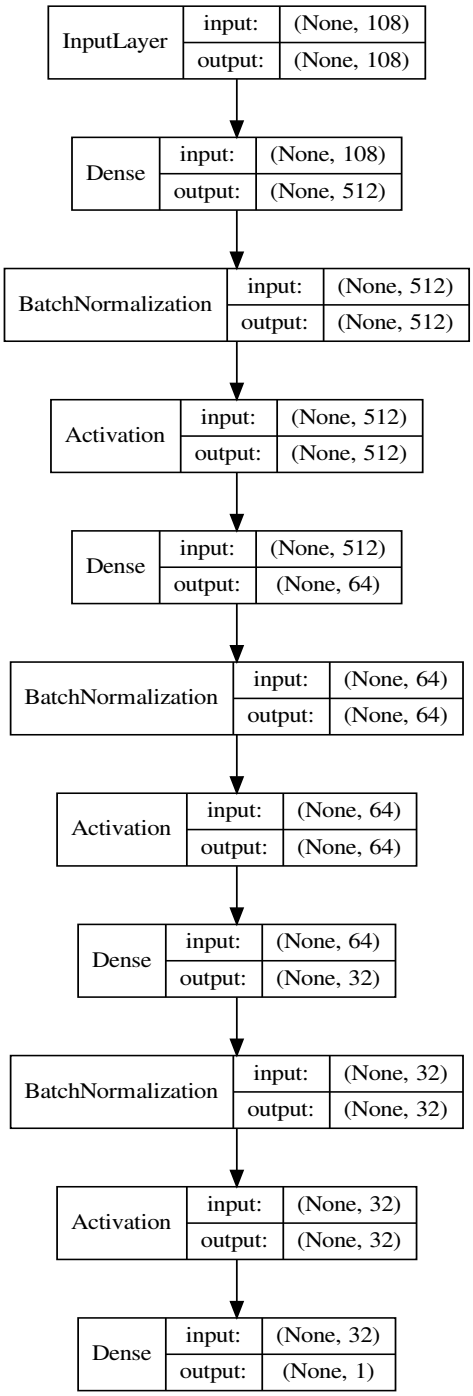


**Figure S4.** The structure of the ANN model.The input shape is $(n_{\text{samples}}, n_{\text{steps}} \times n_{\text{features}} + 8)$. Four FCN(Dense) layer with batch normalization function and Relu activation function are used to build this ANN model.