A Simple User Manual for the Source Code

In the 'AQI-Prediction' pack, a user manual ('User Manual.PDF'), a main function file('MAIN_func_decentral.py') and several related folders are included.

The source code was tested with a Windows 10 64bit computer equipped with Intel i7-8700K CPU and NVIDIA GeForce GTX 1080 GPU, and it should be compatible with any other machines as long as the system requirements are fulfilled.

1 System Requirements

In order to run the code, the following software tools are required to be installed in a 64bit computer:

- Python 3.6.6 or above.

Other vital site-packages for python are also needed:

- Keras (version 2.3.1 recommended);
- TensorFlow (version 1.12.0 recommended);
- NumPy;
- SciPy;
- Matplotlib;
- Pandas;
- CSV;
- PyWavelets;
- sklearn;
- pyhht.
- ……

If any problem appears while running the program, please check whether these packages or any other basic packages are properly installed.

2 AQI-prediction

In 'AQI-prediction' folder there are:

- MAIN_func_decentral.py: This file is the main function file. This runnable python file contains all it needs to conduct a full experiment. Several parameters can be set in it to modify the settings of the training & forecasting process.
- 'dataset' folder: This folder contains the used AQI dataset of 12 observing stations. The data main function needs to train and predict is read from the CSV files in here.

- ‘Part’ folder: This folder contains two python files, which are ‘part_data_preprocessing.py’ and ‘part_evaluate.py’. ‘part_data_preprocessing.py’ file stores several functions the main functions need to process the data read from the CSV files. ‘part_evaluate.py’ file contains several functions for evaluating the forecasting performance.
- ‘Support’ folder: This folder contains several supportive files, which are essential to run the wavelet transform function and NLSTM neural network.
- ‘Model’ folder: This folder contains ‘model_major.py’ file, which contains built deep learning model in it.
- ‘result’ folder: This folder contains CSV files, in which the results of evaluation to the experiment are recorded after conducting the experiment.

3 MAIN_func_decentral.py: Main Function

File ‘MAIN_func_decentral.py’ is the main function in this project, it is contains several models for AQI time series prediction. As the only runnable file in this project, running ‘MAIN_func_decentral.py’ provides all the result the user needs.

Its ‘main’ function starts on line 1247, the size of training set and testing set is set on lines 1283-1284. The AQI record data is loaded on line 1301-1309, and the user can choose which AQI variate is to be loaded.

On line 1313, function ‘load_data_ts’ processes the loaded data into proper shape for training and testing process. On line 1314, function ‘load_data_emd’ processes the loaded data into proper shape and conducts empirical mode decomposition (EMD). On line 1315, function ‘load_data_wvlt’ processes the loaded data into proper shape and conducts wavelet transform decomposition (WT).

Lines 1333-1358 are functions of the prediction models. These functions contains training, testing and evaluating processes, thus the prediction results and the evaluations are obtained by running them. The prediction results are returned respectively by these functions, the evaluations are recorded in two global variables ‘eva_output’ and ‘result_all’. The functions include:

Function ‘Decide_Tree’ on line 1333: the method of Decision Tree.

Function ‘Random_forest’ on line 1334: the method of Random forest.

Function ‘Linear_Regression’ on line 1335: the method of Linear Regression.

Function ‘SVR’ on line 1336: the method of Support Vector Machine Regression.

Function ‘KNN’ on line 1337: the method of k-Nearest Neighbor.

Function ‘MLP’ on line 1338: the method of Multilayer Perception.
Function ‘gradient_Boosting’ on line 1339: the method of Gradient Boosting.
Function ‘extra_Tree’ on line 1340: the method of Extra Tree.
Function ‘bagging’ on line 1341: the method of Bagging.
Function ‘adaboost’ on line 1342: the method of Adaboost.

Function ‘LSTM’ on line 1344: the method of LSTM.
Function ‘EMD_LSTM’ on line 1345: the hybrid method of LSTM and EMD.
Function ‘Wavelet_LSTM’ on line 1346: the hybrid method of LSTM and WT.

Function ‘Nested_LSTM’ on line 1348: the method of NLSTM.
Function ‘EMD_NLSTM’ on line 1349: the hybrid method of NLSTM and EMD.
Function ‘Wavelet_NLSTM’ on line 1350: the hybrid method of NLSTM and WT.

Function ‘Stacked_LSTM’ on line 1352: the method of SLSTM.
Function ‘EMD_SLSTM’ on line 1353: the hybrid method of SLSTM and EMD.
Function ‘Wavelet_SLSTM’ on line 1354: the hybrid method of SLSTM and WT.

Function ‘BLSTM’ on line 1356: the method of BLSTM.
Function ‘EMD_BLSTM’ on line 1357: the hybrid method of BLSTM and EMD.
Function ‘Wavelet_BLSTM’ on line 1358: the hybrid method of BLSTM and WT.

After running this file, the evaluation is printed and saved in CSV files ‘result\\dataset_all_result.csv’ and ‘result\\dataset_all_result_backup.csv’.