Atmospheric Environment 214 (2019) 116885

Contents lists available at ScienceDirect

Atmospheric Environment

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

Improving air quality prediction accuracy at larger temporal resolutions using deep learning and transfer learning techniques

Jun Ma[a], Jack C.P. Cheng[a], Changqing Lin[a,b], Yi Tan[c], Jingcheng Zhang[d,*]

[a] Department of Civil and Environmental Engineering, The Hong Kong University of Science and Technology, Hong Kong, China
[b] Division of Environment and Sustainability, The Hong Kong University of Science and Technology, Hong Kong, China
[c] Sino-Australia Joint Research Center in BIM and Smart Construction, Shenzhen University, Shenzhen, China
[d] School of Engineering, The Hong Kong University of Science and Technology, Hong Kong, China

ARTICLE INFO

Keywords:
Air quality prediction
Large temporal resolution
Deep learning
Long short-term memory
Transfer learning

ABSTRACT

As air pollution becomes more and more severe, air quality prediction has become an important approach for air pollution management and prevention. In recent years, a number of methods have been proposed to predict air quality, such as deterministic methods, statistical methods as well as machine learning methods. However, these methods have some limitations. Deterministic methods require expensive computations and specific knowledge for parameter identification, while the forecasting performance of statistical methods is limited due to the linear assumption and the multicollinearity problem. Most of the machine learning methods, on the other hand, cannot capture the time series patterns or learn from the long-term dependencies of air pollutant concentrations. Furthermore, there is a lack of methods that could generate high prediction accuracy for air quality forecasting at larger temporal resolutions, such as daily and weekly or even monthly. This paper, therefore, proposes a deep learning-based method namely transferred bi-directional long short-term memory (TL-BLSTM) model for air quality prediction. The methodology framework utilizes the bi-directional LSTM model to learn from the long-term dependencies of $PM_{2.5}$, and applies transfer learning to transfer the knowledge learned from smaller temporal resolutions to larger temporal resolutions. A case study is conducted in Guangdong, China to test the proposed methodology framework. The performance of the framework is compared with other commonly seen machine learning algorithms, and the results show that the proposed TL-BLSTM model has smaller errors, especially for larger temporal resolutions.

1. Introduction

In the last decades, along with the rapid development of the industrialization and urbanization, the increasing level of air pollutant concentrations has been a growing concern globally. According to the World Health Organization (WHO), 9 out of 10 people in the world are living with polluted air (Zhou et al., 2019). Commonly seen air pollutants include *CO*, *SO*, O_3, PM_{10}, $PM_{2.5}$, etc. They will not only cause environmental issues, such as soil acidification, fog and haze, but also cause health problems like heart attacks and lung diseases (Chen et al., 2019; Li et al., 2019; Zhou et al., 2019). Millions of people die every year as a result of exposure to ambient air pollutants.

In regard to the damage of air pollution, governments have been working with research institutions to introduce efficient controlling policies. A number of air monitoring stations have been established to monitor and collect the air pollution data for further research. Using the routine observations, air pollutant concentrations in the next hour or the next day can be forecasted. Based on the forecasting, people can adjust their activities and prepare air pollutants prevention in advance, and therefore, mitigate the impact of air pollution on health and economy. Nowadays, when forecasting the air pollutant concentrations, deterministic methods, statistical methods, and machine learning (ML) methods are three widely used approaches (Kwok et al., 2017; Li et al., 2017; Singh et al., 2012). Deterministic methods conduct air quality prediction by constructing a simulation model of the dispersion and transport process of atmospheric chemistry. Commonly seen deterministic methods include Chemical Transport Models (CTMs) (Stern et al., 2008), Weather Research and Forecasting (WRF) models (Saide et al., 2011), Operational Street Pollution Models (OSPM) (Berkowicz, 2000), Nested Air Quality Prediction Modeling System (NAQPMS) (Wang et al., 2001), etc. However, although these deterministic methods have been proved to provide valuable insights into the mechanisms of

* Corresponding author.
E-mail address: zhangjingcheng0306@outlook.com (J. Zhang).

https://doi.org/10.1016/j.atmosenv.2019.116885
Received 10 April 2019; Received in revised form 2 July 2019; Accepted 2 August 2019
Available online 06 August 2019

pollutants diffusion, they are computationally expensive and their prediction results might be inaccurate due to the use of default parameters and the lack of real observations (Catalano and Galatioto, 2017; Kukkonen et al., 2003; Suleiman et al., 2018).

Statistical methods are another kind of approaches to forecasting air pollutant concentrations. They overcome the limitations of deterministic methods by using a large amount of observed data. Among the existing statistical methods, Autoregressive Integrated Moving Average (ARIMA), Generalized Additive Models (GAMs), Multi-layer Regression (MLR), Geographically Weighted Regression (GWR), have been widely adopted in the field of air quality prediction (Ma and Cheng, 2016a, 2016b). For example, Jian et al. (2012) applied the ARIMA model to submicron particle concentrations at a busy broadside in Hangzhou, China. Slini et al. (2002) developed a stochastic ARIMA model for maximum ozone concentration forecasts in Athens, Greece. Davis and Speckman (1999) adopted the GAM approach to predict one day in advance both in maximum and 8-h average ozone for Houston. Hu et al. (2013) used GWR to estimate ground-level $PM_{2.5}$ concentrations in the southeastern U.S. However, it is notable that most of the mentioned statistical methods assumed the relationships between the variables and the target label was linear. This is obviously inconsistent with the non-linearity of the real world. Therefore, the prediction performance of these methods is limited.

To address the problem, researchers started to adopt non-linear machine learning (ML) models as the alternative methods for air quality prediction. Methods such as Support Vector Machine (SVM) (Ma and Cheng, 2017; Osowski and Garanty, 2007; Sun and Sun, 2017), Artificial Neural Networks (ANNs) (Alimissis et al., 2018; Cheng and Ma, 2015a; Yang and Wang, 2017), Fuzzy Logic (FL) (Lin and Cobourn, 2007; Neagu et al., 2002), Random Forest (RF) (Rubal and Kumar, 2018) have been applied in a lot of literatures. Among these methods, ANNs have been one of the most popular methods for air quality prediction. For example, Niska et al. (2004) adopted a feed-forward neural network for forecasting hourly concentration of nitrogen dioxide at a traffic station. Gardner and Dorling (1999) trained Multi-layer Perceptron (MLP) neural networks to model hourly NO_x and NO_2 pollutant concentrations in Central London. Kim et al. (2010) investigated the possibility of using Recurrent Neural Network (RNN) to predict the air quality at a platform of a subway station. Kang and Qu (2017) adopted Back Propagation Neural Network (BPNN) to forecast the air quality in Lanzhou.

On the other hand, as the rapid development of AI and deep learning techniques, the model performance of traditional machine learning and shallow neural networks is no longer state-of-art. Different kinds of deep learning models are proposed to improve the prediction performance of air quality. For example, some literature proposed the implementation of deep RNN, Gated Recurrent Units (GRU) and Long Short-Term Memory (LSTM) models (B. Liu et al., 2019; Bui et al., 2018; Freeman et al., 2018; Greff et al., 2016; İ. Kök et al., 2017; Reddy et al., 2018). These models are reported to be better at modeling long-short temporal dependency. Some studies further improved the model performance by integrating additional layers that considered external features like metrological features, terrain features, and hidden spatial features (P. Soh et al., 2018; Qi et al., 2019, 2018). The back-propagations could help to integrate and optimize these features smoothly in the deep networks. Other advanced deep learning models would apply other types of networks that could better learn the spatial features for air quality such as convolutional neural networks (CNN) (Ayturan et al., 2018; Laptev et al., 2018a; Qi et al., 2019; Stojov et al., 2018).

However, although the aforementioned ANN and deep learning models have exhibited excellent performance in handling prediction problems, most of them focus on the hourly resolution and are usually one-step-ahead prediction. Predicting the air quality in the next hour is useful, but in fact, most citizens and governments are more interested to know the air quality in the next day or the next week. However, limited studies have considered the difference and difficulties in predicting air quality at different temporal resolutions (Kang and Qu, 2017; Kim et al., 2010; Wen et al., 2019; Zhenghua and Zhihui, 2017). The prediction accuracy at larger resolutions was generally lower than the hourly resolution. For example, Wang et al. (2012) studied the urban air quality for the Yangtze River Delta region. One of their experiments compared the observed and predicted concentrations of three air pollutants in Nanjing at hourly and daily resolutions. Their experimental results showed that the errors at daily resolution were much larger than that at hourly resolution. One important reason behind such a phenomenon might be the relatively smaller number of samples for the prediction at larger temporal resolutions. Given the same length of training records, the number of samples at weekly resolution was 1/(24*7) of the number of samples at hourly resolution. As a result, the prediction model may not learn enough knowledge from the historical data to generate accurate forecasting. Therefore, a forecasting model which can mitigate the limitation and improve the prediction accuracy at larger resolutions is necessarily needed.

To overcome the limitations and fill these research gaps, this paper proposes a methodology framework that integrates the bi-directional Long Short-Term Memory (LSTM) neural network and the transfer learning technique to improve the forecasting accuracy for air pollutant concentrations at different time resolutions. Bi-directional LSTM is an improved RNN and is capable of learning from the long-term dependencies of time series data from both forward and backward sequences. Transfer learning, on the other hand, is able to help the model learn and store knowledge from samples at smaller temporal resolutions and improve the prediction performance for samples at larger temporal resolutions. In this paper, the prediction performance of BLSTM is compared with other commonly seen machine learning methods. The effectiveness of transfer learning in improving prediction accuracy is evaluated. To verify the methodology framework, this paper selected PM2.5 as the air pollutant and conducted a case study in Guangdong, China. The experimental results show that our methodology can effectively improve the prediction accuracy for larger temporal resolutions. Details of the methodology framework are introduced in Section 2. Section 3 shows data collection and time series modeling. Section 4 applies the model and analyzes the results. Conclusions are drawn in Section 5.

2. Methodology framework

Fig. 1 presents the methodology framework proposed in this paper. It is composed of three parts, (1) data collection and preprocessing, (2) model application, and (3) result analysis. The first part preprocesses the raw data and constructs the time series samples for the next step. The second part is a major part of the methodology framework. In this part, the prediction performance of BLSTM is trained to predict the air qualities. Its performance will be compared with other commonly seen models. The BLSTM models at hourly, daily and weekly temporal resolutions are trained and explored. Transfer learning is then

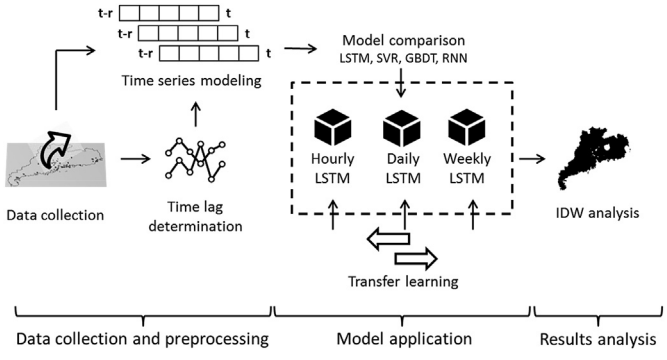

Fig. 1. Methodology framework.

implemented to improve the prediction accuracy. The improvements are evaluated in the third part. The inverse distance weighted (IDW) techniques are implemented to interpolate and visualize the prediction errors within the whole studied area before and after the application of transfer learning. Based on the IDW diagrams, the effectiveness of the methodology framework is discussed.

2.1. LSTM and Bi-directional LSTM

As shown in Fig. 1, LSTM is an important component of the methodology framework. Long Short-Term Memory (LSTM) network is a special Recurrent Neural Network (RNN) architecture proposed by Hochreiter and Schmidhuber (1997). Development of LSTM can be traced back to ANNs. In traditional ANNs, it is assumed that all input variables and output variables are independent with each other. Therefore, they fail to model time series data, of which the information of the previous moment is related to the next moment. To address the issue, RNN is designed. It can stimulate the temporal dynamic behavior for a time sequence by taking the output of the previous moment as the input of the next moment. Theoretically, RNN is able to handle information in arbitrarily long sequences. Unfortunately, in practice, RNN can look back only a few steps due to the vanishing and exploding gradient problem. This limits its performance in modeling sequential problems. Therefore, to overcome this, the LSTM network was proposed.

The specialty in LSTM is that it adds self-connected units which allow a value (forward pass) or gradient (backward pass) that flows into the unit to be preserved and subsequently retrieved at the required time step. This is achieved using the unique self-recurrent cells added in the structure. The architecture of an LSTM cell is presented in Fig. 2. It consists of three gates and two nodes. Three gates mean the forget gate $f_c^{(t)}$, the input gate $i_c^{(t)}$ and the output gate $o_c^{(t)}$. The two nodes represent the input node $g_c^{(t)}$ and the cell state node $s_c^{(t)}$. $f_c^{(t)}$ determines how much information of the last cell can be reserved for this cell. $i_c^{(t)}$ determines how much input information at this moment can flow into the cell and

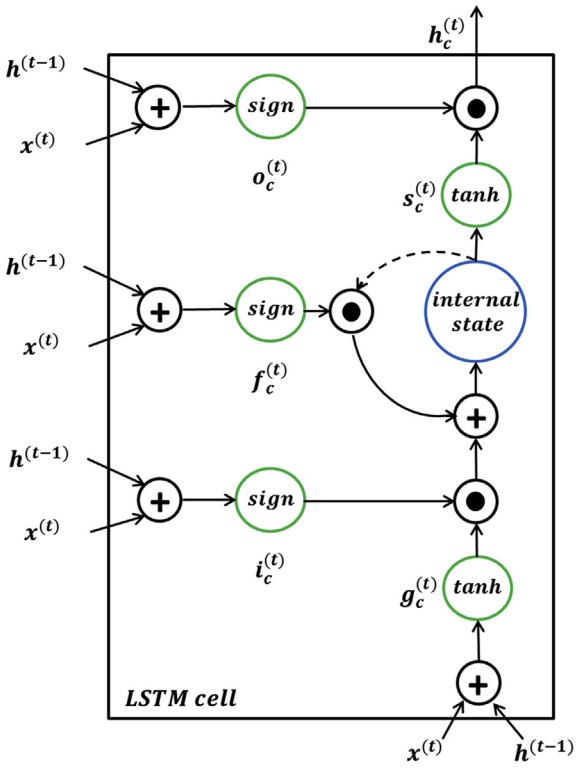

Fig. 2. Structure of an LSTM cell.

$o_c^{(t)}$ determines how much information this cell can output. These three gates jointly control the state of the cell and enable the cell to store information that lies dozen of time steps in the past.

Calculation of the outcome $h^{(t)}$ of one LSTM cell at time t can be formulated as follows:

$$g^{(t)} = \tanh(W_g h^{(t-1)} + U_g x^{(t)} + b_g) \quad (1)$$

$$i^{(t)} = \sigma(W_i h^{(t-1)} + U_i x^{(t)} + b_i) \quad (2)$$

$$f^{(t)} = \sigma(W_f h^{(t-1)} + U_f x^{(t)} + b_f) \quad (3)$$

$$o^{(t)} = \sigma(W_o h^{(t-1)} + U_o x^{(t)} + b_o) \quad (4)$$

$$s^{(t)} = g^{(t)} \odot i^{(t)} + s^{(t-1)} \odot f^{(t)} \quad (5)$$

$$h^{(t)} = \tanh(s^{(t)}) \odot o^{(t)} \quad (6)$$

where tanh () represents the activation function, σ represents the logistic sigmoid function, ⊙ represents the Hadamard product, W represents the output weight, U represents the input weight and b represents the bias vector. The calculation of σ is presented in Equation (7). When σ is 0, the gate is closed and no information can flow in or out. When σ is 1, the gate is open and all the information will flow in or out.

$$\sigma(x) = \begin{cases} 0 \text{ if } x \le t_l \\ ax + b \text{ if } x \in (t_l, t_h) \\ 1 \text{ if } x \ge t_h \end{cases} \quad (7)$$

Bi-directional LSTM (BLSTM) is an upgraded LSTM network proposed by Graves and Schmidhuber (2005). Compared with traditional LSTM, it considered the information contained in later time series to adjust the modeling and calculation. In simple words, this algorithm doubled the LSTM units in the original LSTM network. The original units were built and optimized in the original forward series, while the doubled units were built in a reverse sequence based on the reversed series inputs. These two groups of LSTM units will calculate and output two different $\overrightarrow{h}$, and $\overleftarrow{h}$. As shown in Equation (8), the eventual outputs of a BLSTM unit are the sum of these two predictions.

$$h_{bi}^{(t)} = \overrightarrow{h^{(t)}} + \overleftarrow{h^{(t)}} \quad (8)$$

2.2. Transfer learning

Transfer learning is another important technique used in this paper in order to improve the prediction accuracy at larger temporal resolutions. Transfer learning uses the similarities between two different but related datasets, tasks or models to transfer the knowledge learned from the base domain to the new domain (Laptev et al., 2018b, 2018c; Pan and Yang, 2010; Ye and Dai, 2018). Normally, it is used when the number of samples is limited or when the modeling process is too complexed and computationally expensive. A number of recent studies have taken advantage of transfer learning and obtained state-of-the-art results (Yosinski et al., 2014). For example, Hu et al. (2016) tried transferring the information obtained from data-rich wind farms to a newly-built wind farm for prediction. Wang et al. (2018) conducted experiments on cross-city transfer learning using cases from bike sharing.

However, few studies in atmospheric environment have applied transfer learning in related topics. Lv et al. (2019) implemented transfer learning to study the urban and non-urban air quality, but it is more on the spatial transfer problem, and they did not integrate the state-of-art deep learning technologies. This study, on the other hand, proposed a transfer learning framework based on deep learning techniques to temporally transfer the knowledge from smaller resolutions to larger ones.

Based on the learning patterns, transfer learning can be divided into

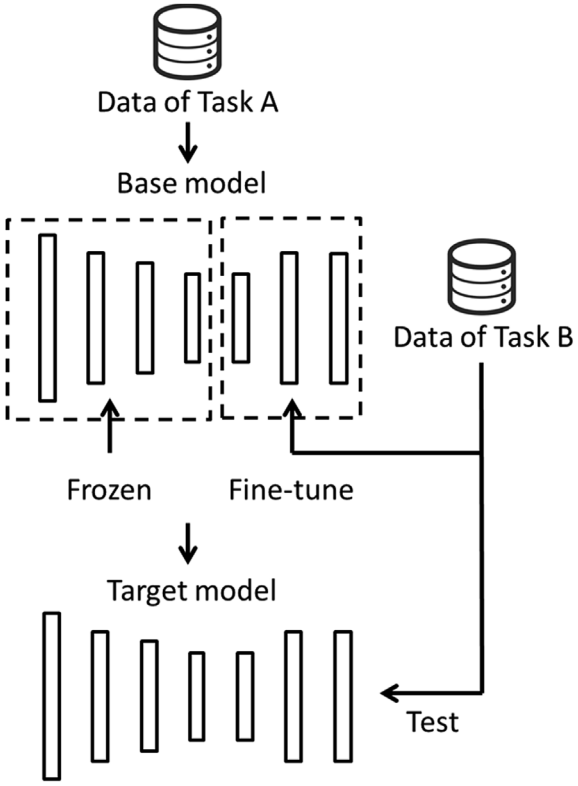

Fig. 3. Structure of the model transfer learning.

sample transfer, feature transfer, model transfer, and relation transfer. In this paper, the model transfer is adopted. A commonly used model transfer approach is shown in Fig. 3. It includes three steps. Firstly, pre-train a base model using the data of Task A. Store the network structure and weights of the model for the next step. Secondly, freeze the weights of the first few layers of the base model and fine-tune the weights of the remained layers using the data of Task B. The number of frozen layers varies from case to case. Thirdly, test the fine-tuned model using the data of Task B.

2.3. IDW interpolation

After the LSTM models are constructed and transfer learning is applied, Inverse Distance Weighting (IDW) is adapted to interpolate and visualize the prediction error within the studied area. This can give a more straightforward view of the improvements in predicting larger temporal resolutions and assess the generalization of the proposed model. Its general idea assumes that the attribute value of an unknown point is the weighted average of known values within the neighborhood, and the weights are inversely related to the distance between the prediction location and the sampled locations. Calculation of the attribute value of the unknown location is shown in Equation (9).

$$Z(x) = \frac{\sum_{i=1}^{n} Z(x_i)\cdot w_i}{\sum_{i=1}^{n} w_i} \quad (9)$$

where $Z(x)$ is the attribute value of the predicted location x, $Z(x_i)$ is the attribute value of the i^{th} point x_i within the neighborhood, n represents the number of points within the neighborhood and w_i is the weight assigned to x_i.

In summary, to predict the air quality in different temporal resolutions and improve the prediction accuracy for larger resolutions, this paper proposes a methodology framework taking advantage of the BLSTM neural network and the transfer learning technique. The methodology is capable of learning from long-term dependencies and achieving great performance even when the number of input data is limited. To validate the effectiveness of the proposed methodology, a case study is carried out in the following sections.

Table 1
Data summary.

Attribute	Content
Pollutant	$PM_{2.5}$
District	Guangdong, China
Stations	100
Time Range	2015.1–2017.12
Resolution	Hour
Unit	$\mu g/m^3$
Mean	31.80
Standard Deviation	22.14
Min	1
Max	905

3. Case study

3.1. Data collection

The case study was conducted in Guangdong, China. Three-year air quality data of all the monitoring stations in the Guangdong province were collected. Each station records 26,304 hourly $PM_{2.5}$ concentration data. The data summary of the collected data is shown in Table 1. The distribution of the stations in Guangdong is presented in Fig. 4, and it is not evenly distributed. Most of the stations distribute in the central and southern of Guangdong and the coastal areas while there are relatively fewer stations in the north.

3.2. Rolling window modeling

After the raw data are collected, data preprocessing should be conducted. Since the collected data do not contain any missing value, sample modeling remains to be the most essential step. The data need to be transformed in a more friendly way for temporal sequential models like LSTM. To achieve this, the rolling window method is commonly used (Zivot and Wang, 2006).

Rolling window constructs one sample for each time record t. The sample for t_0 is constructed using the values within $[t_0 - \Delta t, t_0)$ as the features, and the value at t_0 as the label or target. Δt is called the window size. An example of how to construct the time series samples using the rolling window is shown in Fig. 5. Assume there are 10 time-series records in the dataset, including T1, T2, …, T9 and T10. If $\Delta t = 6$, then for Sample 1, it has T1, T2, T3, T4, T5 and T6 as its

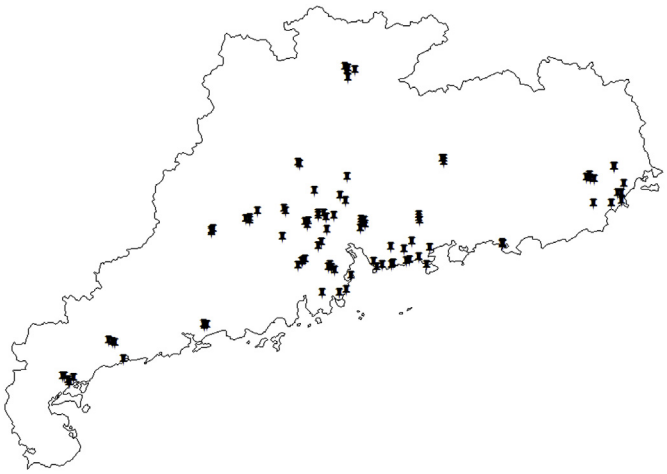

Fig. 4. Distribution of the air quality monitoring stations in Guangdong.

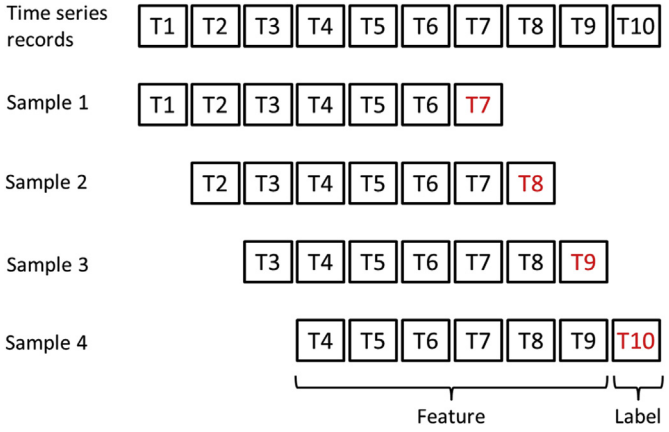

Fig. 5. An example of how to construct the time series samples.

features and T7 as its label. For Sample 2, it has T2, T3, T4, T5, T6 and T7 as its features and T8 as its label. Similarly, Sample 3 and Sample 4 are given. As a result, four time-series samples are built when the time series records are 10 and the window size is 6. If $\Delta t = 7$, then for Sample 1, it will have T1, T2, T3, T4, T5, T6 and T7 as its features and T8 as its label. Three time-series samples can be built. It can be seen that the value of window size will influence the number of time series samples and the features in a sample. For a given dataset, smaller window size means more samples but fewer features while larger window size means fewer samples but more features.

In our dataset, each station has 26,304 hourly records. For example, if we define the window size Δt as 6, then there will be 26,298 time-series samples in total for a station. The optimal window size for the case study will be identified later.

4. Model application and results analysis

After the time series samples were constructed, they would be input into the model. A series of experiments were then conducted, including (1) determination of the window size and the network parameters, (2) identification of the optimal window size at different time resolutions, (3) comparison of models and (4) application of transfer learning. To evaluate the model performance, three indicators are used. They are Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE). Calculation of these three evaluation metrics are presented in Equations (10)–(12).

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - y_i^*)^2} \tag{10}$$

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - y_i^*| \tag{11}$$

$$\mathrm{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - y_i^*|}{y_i} \tag{12}$$

where n represents the number of samples, y_i represents the observed value of the ith sample and y_i^* represents the predicted value of the ith sample. Lower values of these three indicators mean higher prediction accuracy and better performance of the models.

4.1. Determination of window size and network parameters

As mentioned above, the number of features in time series models is determined by the window size. A smaller window size cannot guarantee enough long-term memory inputs for the model while a larger window size will increase unrelated inputs and the computation complexity (Li et al., 2017). Therefore, it is necessary to identify the optimal

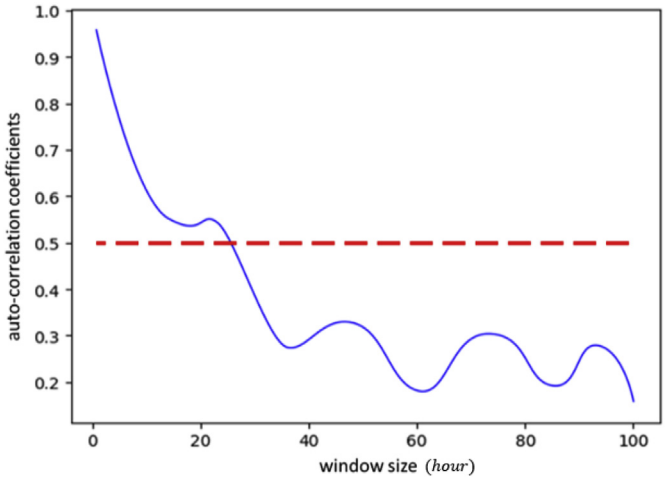

Fig. 6. Window size determination.

window size to help achieve the best prediction performance of the model. In this paper, the auto-correlation function is used to identify the most appropriate window size (Brockwell et al., 2002). Auto-correlation coefficient helps determine the time correlation among the time series data itself. Larger coefficients mean stronger time correlations and stronger lagged effects. Calculation of the auto-correlation coefficient is shown in Equation (13).

$$\rho_{\Delta t} = \frac{\mathrm{Cov}(y(t), y(t+\Delta t))}{\sqrt{\sigma_{y(t)}\sigma_{y(t+\Delta t)}}} \tag{13}$$

where $y(t)$ and $y(t + \Delta t)$ represent the time series at time t and $t + \Delta t$, respectively, $Cov(\cdot)$ represents the covariance, and $\sigma(\cdot)$ represents the variance.

To identify the most appropriate window size, auto-correlation coefficients of each station with different values of window size are calculated. The average value of the coefficients of all the stations for different Δt is then computed. Results are shown in Fig. 6. It can be seen that the coefficient decreases with as the window size increases. This proves that the earlier events have a weaker impact on the current state. It also can be observed that when Δt is smaller than 24, average auto-correlation coefficient of all the stations is higher than 0.5, which indicates the time correlation is stronger. Therefore, we narrow the range of Δt to (1, 24) for further treatments. Optimal window sizes at different time resolutions will be identified later.

Besides the window size, the network structure of the stacked LSTM network will also influence the prediction performance a lot. This paper followed parts of the structures and parameters used in the studies conducted by Yosinski et al. (2014), Salman et al. (2018) and Hu et al. (2016). In their research, an eight-layer neural network with the first seven layers as Bi-directional LSTM layers and the last layer as the fully connected layer was designed. Epochs were set as 200 while the learning rate was set as 0.015. The number of neurons in each LSTM layer was set as two times the window size in the first layer. The parameters are optimized by the integration of the Mini-Batch Gradient Descent (MBGD) algorithm, dropout neuron algorithm and L2 regularization algorithm. The batch size is set as 128. Relu was chosen as the activation function.

4.2. Optimal window size at different time resolutions

As mentioned in the introduction, the prediction accuracy at different time resolutions could be different even when using the same method and the same dataset. Especially at larger resolutions, the prediction accuracy might be low due to the smaller number of samples. In regard to this, this paper evaluates the possibility of using BLSTM and transfer learning to address the problem. Before that, the optimal

Table 2
Hourly prediction performance of LSTM with respect to different window size.

Window size	RMSE	MAE	MAPE (%)	Window size	RMSE	MAE	MAPE (%)
1	9.6606	6.8475	25.0654	13	9.6802	6.0076	24.0050
2	9.1508	5.4979	23.1150	14	10.2604	6.5570	24.5551
3	9.5607	5.8475	23.6958	15	8.3604	5.1777	23.7551
4	9.1602	5.7670	23.9451	16	8.8007	5.4773	24.0252
5	8.0402	4.7969	22.8649	17	9.7809	6.0972	24.0555
6	8.2808	5.1377	22.9750	18	9.7407	6.0176	23.7750
7	10.6603	6.8470	25.0654	19	10.1506	6.4473	24.3958
8	8.8507	5.2973	23.6150	20	8.8203	5.5772	24.1353
9	9.5606	5.8475	23.6954	21	10.4603	6.6673	24.6252
10	8.9707	5.5470	23.3950	22	8.6203	5.4576	24.0654
11	9.7303	6.0870	24.5653	23	9.9107	6.3570	24.5055
12	9.6010	6.0072	23.8755	24	8.8711	5.6878	23.7558

window size at different time resolutions needs to be determined to build hourly, daily and weekly time series samples. For simplicity, this paper takes the data of monitoring station 1356 A as an example to present the process of window size determination. 70% of the data are used as training sets and the remaining 30% as testing sets.

Based on the results of the last section, we set the range of window size as (1, 24) for hourly resolution. Prediction performance of LSTM with respect to different window size is presented in Table 2. It can be seen that when the window size varies from 1 to 24, RMSE varies around 8 to 11, MAE around 4 to 7 and MAPE around 22 to 26. To better observe the change of scores of the indicators, Fig. 7 presents the scores in a line chart. The blue line represents the scores of MAPE, the red line RMSE and the yellow line MAE. It can be observed that the trends of these three lines are similar. When the window size is 5, all the three lines reach their lowest points. This means when window size equals 5, LSTM has the best prediction performance at hourly resolution. Therefore, the window size is set as 5 for hourly resolution.

To optimize the window size at daily and weekly resolutions, daily and weekly $PM_{2.5}$ concentrations data are estimated from hourly data first. They are given by calculating the mean values of daily concentrations data. 1096 daily $PM_{2.5}$ concentrations data and 156 weekly $PM_{2.5}$ concentrations data are given. By using the same procedure in Section 4.1, the window size for daily and weekly resolution is pre-set as (1, 7). Similarly, the forecasting performances of LSTM at daily and weekly resolutions with different window size are assessed. It is also found that when the window size is 5, indicators at daily and weekly resolutions reach the smallest values. Therefore, the window size is set as 5 for both daily and weekly resolutions.

Note that the window size of the target resolution will affect the inputs of the base model. In this experiment, the window size for hourly, daily, and weekly resolutions are the same, so it will not cause problems for later transfer learning. But if the window sizes are different, the proposed method will use the size of the target resolution to train the base model. For example, if the optimal window size for daily resolution is 4 instead of 5, then when training the base model using the hourly resolution, the window size would be 4.

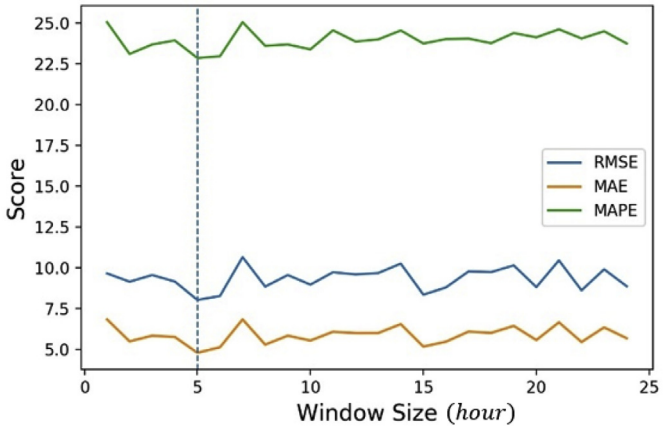

Fig. 7. Scores of three indicators with different window size for the base model on the hourly resolution.

4.3. Comparison of prediction models

This paper selects BLSTM as the major regression algorithm. To prove that BLSTM is a reasonable choice for our methodology framework, its prediction performance is compared with several other commonly used models, including Autoregressive Integrated Moving Average (ARIMA), Support Vector Regression (SVR), Gradient Boost Decision Tree (GBDT), Recurrent Neural Network (RNN), Gated Recurrent Units (GRU), ordinary LSTM and Convolutional Neural Network-LSTM (CNN-LSTM). ARIMA is one of the most typical linear methods for time series predictions. SVR and GBDT are two typical high-performance machine learning algorithms (Cheng and Ma, 2015b; Jun and Cheng, 2017). RNN has been introduced in section 2. CNN-LSTM is an integration of convolutional neural network and LSTM, and is reported to have positive performance in some literature (Bui et al., 2018; Cheng and Ma, 2015b; Wen et al., 2019). Since the inputs in this experiment are from one station, so 1-D CNN-LSTM is implemented here.

Hourly $PM_{2.5}$ concentrations data of station 1356 A are used. 70% of the data are taken as the training set and the remained 30% are taken as the testing set. The RMSE, MAE, and MAPE scores of these models are presented in Table 3. The parameters of the algorithms are all fine-tuned. Compared with the linear method and traditional machine learning algorithms, neural network-based models like RNN, GRU, LSTM and Bi-LSTM perform better. This is because those models are particularly designed for time series predictions and are better at capturing the temporal dependency. In addition, although CNN-LSTM is reported to be a more advanced algorithm (Jun and Cheng, 2017; Qi et al., 2019; Wen et al., 2019), our experiment did not support its priority. It performs better than BLSTM only at MAPE, but with higher values in RMSE and MAE. This may because the inputs in this experiment are only one-dimension time series. If multiple station inputs are included, and the values can be gridded into fishnets, the performance of CNN-LSTM may become better.

The performance of the algorithms on their training sets is also presented in Table 3. That performance is generally better than the testing results (lower error). This is quite understandable, because the results on training sets are usually overfitted (Ma and Cheng, 2016c), and cannot be used to represent the model performance. Therefore, the numbers and results in the following sections are all based on the testing results.

Overall, two of the three indicators show that BLSTM has the best performance in our study. So the following transfer learning experiments will go with BLSTM and explore the influence on the predictions

Table 3
Prediction performance of different algorithms and networks on the base model on the hourly resolution.

Indicators	ARIMA	SVR	GBDT	RNN	GRU	LSTM	CNN-LSTM	BLSTM
RMSE	12.3926	10.6739	9.1559	8.8864	8.5459	8.20,156	8.1326	8.0402
MAE	7.8525	6.4286	5.7570	6.1858	5.5494	5.2165	5.0569	4.7969
MAPE (%)	30.5160	27.4611	25.8674	24.1483	23.8528	24.958	22.0225	22.8649
RMSE (training)	10.8634	8.0634	7.7463	8.9062	8.0629	7.9481	7.7523	6.9663

Table 4
Prediction performance of BLSTM at hourly, daily and weekly resolutions.

Indicators	Hourly	Daily	Weekly
RMSE	8.0402	9.6877	7.2316
MAE	4.7969	6.9469	5.6805
MAPE (%)	22.8649	30.8274	27.7610
RMSE (adding up hourly multi-step predictions)	8.0402	10.0666	8.9966

at larger resolutions.

For comparison in later sections, we first calculated the performance of ordinary BLSTM models on daily and weekly $PM_{2.5}$ data. 1096 daily data and 156 weekly data are generated. 70% of the data are taken as training sets and the remained 30% are taken as testing sets. Table 4 presents the comparison of prediction performance of BLSTM at hourly, daily and weekly resolutions. The error indicators increased from hourly to daily, and decreased from daily to weekly, but the values did not change too much. This reflected the powerful modeling capability of BLSTM. When having much lower data in daily and weekly modeling, the performance of the model did not drop too much. The error decrease from daily to weekly may because of the data variance. Also, predictions at larger resolutions are expected to have lower error because the data records are the average value of smaller resolutions, and this helped filter out some outliers and noise.

Due to the powerful capability of multi-step prediction in LSTM or BLSTM. One of the reviewers suggested exploring the possibility of predicting $PM_{2.5}$ at larger resolutions by directly averaging the prediction results of multi-step hourly BLSTM predictions. To explore this, we calculated the multi-step predictions of BLSTM for 24 h and 168 h by setting the neurons of the output layers to those numbers. Then the predictions are averaged to later calculate the RMSEs for daily and weekly predictions. It can be seen from Table 4 that although such a way of calculation can also provide similar results than regenerating the samples in daily and weekly, the RMSE indicators still favor a specific model building procedure for different resolutions, especially for weekly resolution.

4.4. Prediction performance of the transfer learning-based BLSTM model

To improve the prediction accuracy of the BLSTM models at larger temporal resolutions, this study proposes the transfer learning-based BLSTM (TL-BLSTM) model. It helps the model learn the knowledge from hourly $PM_{2.5}$ concentrations data and adjust it using the daily and weekly $PM_{2.5}$ concentrations. Hourly data are used to pre-train a base model. For model adjustment and testing, daily and weekly data both are divided into two parts. One part accounts for 70% and is used to fine-tune the pre-trained model. The other part accounts for 30% and is used to test the TL-BLSTM model.

4.4.1. Identifying the number of frozen layers

Before applying the transfer learning-based BLSTM (TL-BLSTM) model to daily and weekly predictions, the number of frozen layers of the base model needs to be determined first. Freezing the first few layers aims at storing the knowledge learned from the base data. If the number of frozen layers is too small, the model cannot learn and store enough knowledge from the base data. If the number of frozen layers is too large, overfitting might occur. Therefore, to ensure the performance of the TL-BLSTM model at longer-term predictions, the optimal number of frozen layers needs to be identified. Daily $PM_{2.5}$ concentrations data are firstly used as an example. Prediction performances of the TL-BLSTM model with different numbers of frozen layers are evaluated. The results are presented in Table 5. It can be seen that when the number of frozen layers is 3, RMSE, MAE, and MAPE have the lowest scores. Therefore, in the following treatments for daily resolutions, the first 3 layers are frozen in the pre-trained model.

Table 5
Prediction performance of the transfer learning-based LSTM model with different number of frozen layers for daily resolution.

Number of Frozen Layers	RMSE	MAE	MAPE
1	10.9583	7.9859	33.4293
2	9.6281	7.3156	30.3259
3	8.6529	6.1840	27.9090
4	9.9598	7.9582	29.6584
5	11.2616	8.1565	34.3619
6	12.6592	8.5619	36.1845

4.4.2. Transfer results at daily resolution

After the number of frozen layers is determined, $PM_{2.5}$ concentrations at daily resolution are evaluated using TL-BLSTM. Fig. 8 shows the comparison of the observed values and the predicted values of the BLSTM models before and after applying the transfer learning. Blue lines represent the real values and orange lines represent the predicted values. Compared with the ordinary BLSTM model, the predicted values of TL-BLSTM are more consistent with the observed values.

To compare the prediction performance of the two models more straightforward, indicator scores of the two models are listed in Table 6. It can be observed that compared with the ordinary BLSTM model, the TL-BLSTM model has lower indicator scores. The prediction performance improved by around 10%.

Furthermore, all the experiments conducted above are based on the

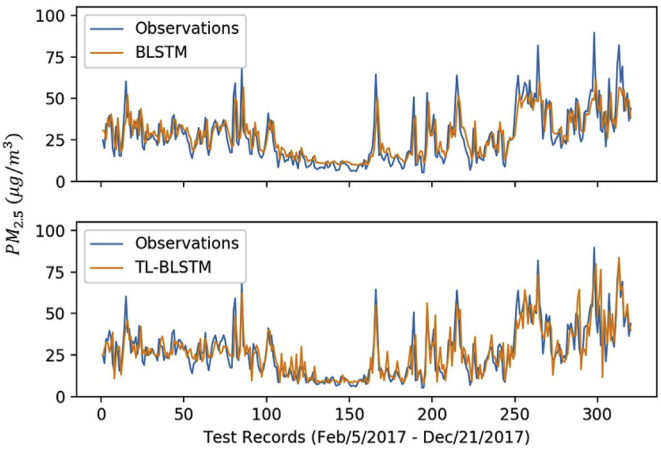

Fig. 8. Comparison of the observed values and the predicted values of the ordinary BLSTM model and the transfer learning-based BLSTM model at the daily resolution.

Table 6
Daily prediction performance of the ordinary BLSTM model and the transfer learning-based BLSTM model.

Indicators	BLSTM	TL-BLSTM	Improvement
RMSE	9.6877	8.6529	10.68%
MAE	6.9469	6.1840	10.98%
MAPE (%)	30.8274	27.9090	9.47%

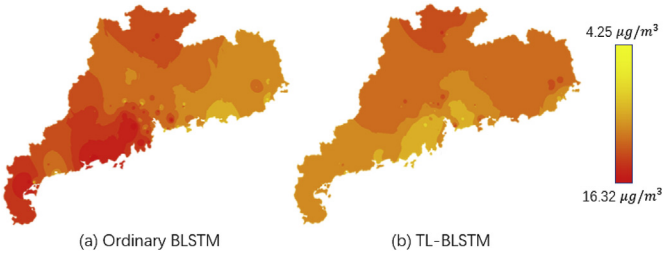

Fig. 9. The distribution of RMSE of the ordinary BLSTM model and the transfer learning-based BLSTM model at the daily resolution.

data of one station for an example. To further evaluate the generalization of the proposed TL-BLSTM model, it is applied to all the other stations within the studied area. The daily $PM_{2.5}$ concentrations for all the monitoring stations in Guangdong are predicted using the ordinary BLSTM and TL-BLSTM models. Their scores of RMSE are calculated based on the 30% of the daily data which are used to test the models. Then IDW is utilized to visualize and interpolate the RMSE values within the studied area to make errors more straightforward. IDW diagrams based on the two models are presented in Fig. 11. Yellow means lower level and smaller prediction error while red means higher level and larger prediction error. The TL-BLSTM map appears to have more yellow, and this suggests that transfer learning can efficiently minify the prediction error and improve the prediction accuracy of BLSTM at the daily resolution.

4.4.3. Transfer results at weekly resolution

Similarly, weekly $PM_{2.5}$ concentrations are estimated by the TL-BLSTM model using hourly and weekly data. The number of frozen layers in the BLSTM model is also identified using the same procedure in Section 4.4.1, and is also found to be 3 for weekly resolutions. Comparison of the observed values and the predicted values of the ordinary BLSTM model and the TL-BLSTM model are presented in Fig. 10. Blue lines represent the real values and red lines represent the predicted values. Compared with the ordinary BLSTM model, the curve from the TL-BLSTM model also fits better. Table 7 shows the weekly prediction performance of the two models. After applying the transfer learning, the three error indicators all improved by more than 50%.

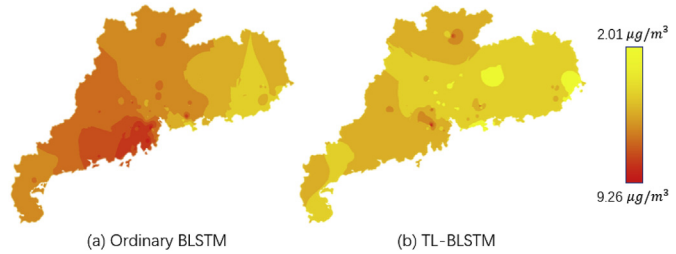

Fig. 11. The distribution of RMSE of the ordinary BLSTM model and the transfer learning-based BLSTM model at the weekly resolution.

Table 7
Weekly prediction performance of the ordinary BLSTM model and the transfer learning-based BLSTM model.

Indicators	Ordinary BLSTM	TL-BLSTM	Improvement
RMSE	7.2316	2.9777	58.82%
MAE	5.6805	2.2715	60.01%
MAPE (%)	27.7610	9.1007	67.22%

Table 8
Prediction performance of the ordinary BLSTM model and the self-transfer learning-based BLSTM model.

Indicators	Ordinary BLSTM	Self-TL-BLSTM
RMSE	8.0402	8.5426
MAE	4.7969	4.9586
MAPE (%)	22.8649	22.3261

Compared with Table 6, the improvements for the weekly resolution is much higher than that for the daily resolution. This may because the data shortage problem is more severe for the weekly resolution, and the transfer learning helped more in the modeling and prediction.

In addition, the generalization of the TL-BLSTM model at weekly resolution is examined as well. The weekly $PM_{2.5}$ concentrations for all the stations in Guangdong are predicted using the ordinary BLSTM model and the TL-BLSTM model. Their scores of RMSE are calculated. Fig. 9 presents the distribution of the RMSE of the two models in Guangdong using IDW. RMSE of the TL-BLSTM model is clearly to be lower with more yellow areas. This implies that the TL-BLSTM model can provide predicted values with a lower error at larger temporal resolutions.

4.4.4. Self-transfer learning

The two TL-BLSTM models developed above both learned knowledge from smaller temporal resolutions. Experimental results have

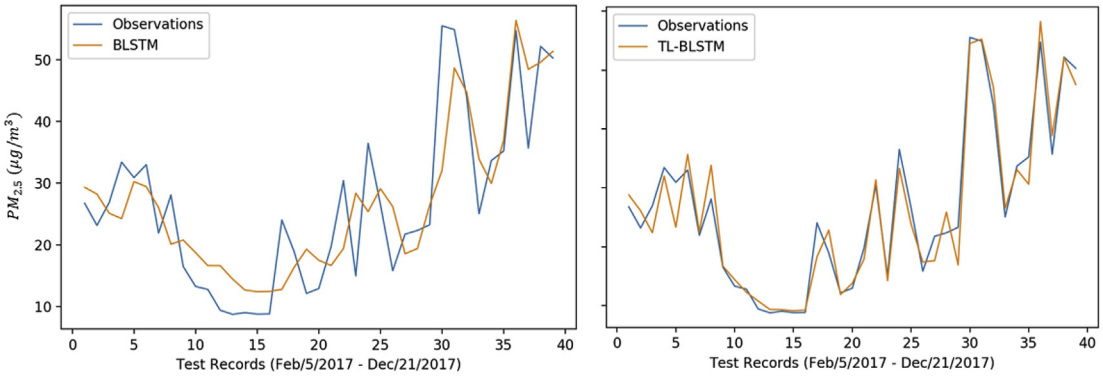

Fig. 10. Comparison of the observed values and the predicted values of the ordinary BLSTM model and the transfer learning-based BLSTM model at the weekly resolution.

shown that they could effectively reduce the prediction error. Besides, this study also investigated whether self-transfer learning can further promote the prediction accuracy of BLSTM. For the ordinary BLSTM model, the way of data partition is as presented previously. 70% of the hourly data are used as the training set and the rest 30% are used as the testing set. For the self-TL-BLSTM model, the 70% training data were separated into two equal halves from the middle. The first half is used to pre-train the model. The second half is used as fine-tuning sets. Note that this way of partition is to keep the same test data for both groups. The number of frozen layers is set as 3.

Table 8 presents the prediction performance of the models before and after applying self-transfer learning. The self-transfer learning-based BLSTM model cannot surpass the ordinary BLSTM model based on the error indicators. This is because the base model for TL-BLSTM only learns the knowledge of $PM_{2.5}$ data for the first half, while the ordinary model learns more knowledge from the data for the whole set. Also, the transfer learning part only updates the parameters in the fourth and subsequent layers. Therefore, it can be concluded that for self-transfer, the transfer learning method cannot further improve and even might negatively influence the prediction performance of LSTM.

5. Conclusions

To conclude, this paper proposes a methodology framework to investigate the effectiveness of BLSTM model and transfer learning in improving the prediction accuracy of air pollutant concentrations. Prediction performance of BLSTM is compared with other models and at different time resolutions. Transfer learning helps to improve the prediction accuracy of BLSTM at larger resolutions. Hourly $PM_{2.5}$ concentrations data of all the monitoring stations in Guangdong in 2017 are collected to test the effectiveness of the proposed methodology. Main contributions of this study can be summarized as follows:

- This paper is one of the few pioneering studies that integrated deep learning and transfer learning techniques in predicting air quality. Especially for the predictions at larger temporal resolutions. This could help governments and researchers generate and analyze more accurate trends for long-term air quality analysis.
- It is found that transfer learning can effectively lower the prediction error of BLSTM for $PM_{2.5}$ at larger temporal resolutions, with 36.85% lower RMSE at daily resolution and 42.58% lower at weekly resolution. It is expected that for larger granularizes, transfer learning performs better as it has less data for training.
- For base models used in transfer learning, BLSTM based algorithms outperform other machine learning and deep learning models in modeling $PM_{2.5}$ in our experiment.
- Self-transfer learning contributes little to improve, or even worsens the prediction performance of LSTM.

Still, this paper can be further improved. Due to the availability of the data, other influential factors of $PM_{2.5}$ concentrations, such as meteorology and geography, are not taken into consideration in this paper. Those factors are expected to help improve the model performance, and should be included in the future work. Also, the experiments in this study are all base on the $PM_{2.5}$ in Guangdong, China, 2017, and the discoveries may fit the studied area only. Although the proposed method is expected to be applicable to other places and other pollutants, future studies need to be conducted to verify the reliability and generalizability. In addition, although the proposed method provides a better prediction for the data with a larger temporal resolution, it requires more time in training. Since the base model or the base-resolution data usually has a larger data size than the target resolution, the training can consume more than twice of non-transfer learning way of modeling. Future work can be conducted to address this problem.

Other future directions of this study can be expanded to explore the applicability of transfer learning technique in other related problems. For example, is it possible to transfer the knowledge from other domain, such as meteorological features, directly to $PM_{2.5}$ concentrations. This could possibly benefit the prediction of air quality in areas without monitoring stations.

Declaration of competing interest

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

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.atmosenv.2019.116885.

References

Alimissis, A., Philippopoulos, K., Tzanis, C.G., Deligiorgi, D., 2018. Spatial estimation of urban air pollution with the use of artificial neural network models. Atmos. Environ. 191, 205–213. https://doi.org/10.1016/j.atmosenv.2018.07.058.

Ayturan, Y.A., Ayturan, Z.C., Altun, H.O., 2018. Air pollution modelling with deep learning: a review. Int. J. Environ. Pollut. Environ. Model. 1, 58–62.

Berkowicz, R., 2000. OSPM — a parameterised Street pollution model. In: Sokhi, R.S., San José, R., Moussiopoulos, N., Berkowicz, R. (Eds.), Urban Air Quality: Measurement, Modelling and Management: Proceedings of the Second International Conference on Urban Air Quality: Measurement, Modelling and Management Held at the Computer Science School of the Technical University of Madrid 3–5 March 1999. Springer Netherlands, Dordrecht, pp. 323–331. https://doi.org/10.1007/978-94-010-0932-4_35.

Brockwell, P.J., Davis, R.A., Calder, M.V., 2002. Introduction to Time Series and Forecasting. Springer.

Bui, T.-C., Le, V.-D., Cha, S.-K., 2018. A Deep Learning Approach for Air Pollution Forecasting in South Korea Using Encoder-Decoder Networks & LSTM. ArXiv Prepr. ArXiv180407891.

Catalano, M., Galatioto, F., 2017. Enhanced transport-related air pollution prediction through a novel metamodel approach. Transp. Res. Part Transp. Environ. 55, 262–276. https://doi.org/10.1016/j.trd.2017.07.009.

Chen, Z., Cui, L., Cui, X., Li, X., Yu, K., Yue, K., Dai, Z., Zhou, J., Jia, G., Zhang, J., 2019. The association between high ambient air pollution exposure and respiratory health of young children: a cross sectional study in Jinan, China. Sci. Total Environ. 656, 740–749. https://doi.org/10.1016/j.scitotenv.2018.11.368.

Cheng, J.C.P., Ma, L.J., 2015a. A data-driven study of important climate factors on the achievement of LEED-EB credits. Build. Environ. 90, 232–244. https://doi.org/10.1016/j.buildenv.2014.11.029.

Cheng, J.C.P., Ma, L.J., 2015b. A non-linear case-based reasoning approach for retrieval of similar cases and selection of target credits in LEED projects. Build. Environ. 93, 349–361. https://doi.org/10.1016/j.buildenv.2015.07.019.

Davis, J.M., Speckman, P., 1999. A model for predicting maximum and 8h average ozone in Houston. Atmos. Environ. 33, 2487–2500. https://doi.org/10.1016/S1352-2310(98)00320-3.

Freeman, B.S., Taylor, G., Gharabaghi, B., Thé, J., 2018. Forecasting air quality time series using deep learning. J. Air Waste Manag. Assoc. 68, 866–886.

Gardner, M.W., Dorling, S.R., 1999. Neural network modelling and prediction of hourly NOx and NO2 concentrations in urban air in London. Atmos. Environ. 33, 709–719. https://doi.org/10.1016/S1352-2310(98)00230-1.

Graves, A., Schmidhuber, J., 2005. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. IJCNN 18, 602–610. 2005. https://doi.org/10.1016/j.neunet.2005.06.042.

Greff, K., Srivastava, R.K., Koutník, J., Steunebrink, B.R., Schmidhuber, J., 2016. LSTM: a search space odyssey. IEEE Trans. Neural Netw. Learn. Syst. 28, 2222–2232.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9, 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735.

Hu, X., Waller, L.A., Al-Hamdan, M.Z., Crosson, W.L., Estes, M.G., Estes, S.M., Quattrochi, D.A., Sarnat, J.A., Liu, Y., 2013. Estimating ground-level PM2.5 concentrations in the southeastern U.S. using geographically weighted regression. Environ. Res. 121, 1–10. https://doi.org/10.1016/j.envres.2012.11.003.

Hu, Q., Zhang, R., Zhou, Y., 2016. Transfer learning for short-term wind speed prediction with deep neural networks. Renew. Energy 85, 83–95. https://doi.org/10.1016/j.renene.2015.06.034.

Jian, L., Zhao, Y., Zhu, Y.-P., Zhang, M.-B., Bertolatti, D., 2012. An application of ARIMA model to predict submicron particle concentrations from meteorological factors at a busy roadside in Hangzhou, China. Sci. Total Environ. 426, 336–345. https://doi.org/10.1016/j.scitotenv.2012.03.025.

Jun, M.A., Cheng, J.C.P., 2017. Selection of target LEED credits based on project information and climatic factors using data mining techniques. Adv. Eng. Inf. 32, 224–236. https://doi.org/10.1016/j.aei.2017.03.004.

Kang, Z., Qu, Z., 2017. Application of BP neural network optimized by genetic simulated annealing algorithm to prediction of air quality index in Lanzhou. In: 2017 2nd IEEE International Conference on Computational Intelligence and Applications (ICCIA),

pp. 155–160. Presented at the 2017 2nd IEEE International Conference on Computational Intelligence and Applications (ICCIA). https://doi.org/10.1109/CIAPP.2017.8167199.
Kim, M.H., Kim, Y.S., Lim, J., Kim, J.T., Sung, S.W., Yoo, C., 2010. Data-driven prediction model of indoor air quality in an underground space. Korean J. Chem. Eng. 27, 1675–1680. https://doi.org/10.1007/s11814-010-0313-5.
Kök, İ., Şimşek, M.U., Özdemir, S., 2017. A deep learning model for air quality prediction in smart cities. In: 2017 IEEE International Conference on Big Data (Big Data). Presented at the 2017 IEEE International Conference on Big Data (Big Data), pp. 1983–1990. https://doi.org/10.1109/BigData.2017.8258144.
Kukkonen, J., Partanen, L., Karppinen, A., Ruuskanen, J., Junninen, H., Kolehmainen, M., Niska, H., Dorling, S., Chatterton, T., Foxall, R., Cawley, G., 2003. Extensive evaluation of neural network models for the prediction of NO2 and PM10 concentrations, compared with a deterministic modelling system and measurements in central Helsinki. Atmos. Environ. 37, 4539–4550. https://doi.org/10.1016/S1352-2310(03)00583-1.
Kwok, L.K., Lam, Y.F., Tam, C.-Y., 2017. Developing a statistical based approach for predicting local air quality in complex terrain area. Atmos. Pollout. Res. 8, 114–126. https://doi.org/10.1016/j.apr.2016.08.001.
Laptev, N., Yu, J., Rajagopal, R., 2018a. Deepcast: Universal Time-Series Forecaster.
Laptev, N., Yu, J., Rajagopal, R., 2018b. Applied Timeseries Transfer Learning.
Laptev, N., Yu, J., Rajagopal, R., 2018c. Reconstruction and Regression Loss for Time-Series Transfer Learning.
Li, X., Peng, L., Yao, X., Cui, S., Hu, Y., You, C., Chi, T., 2017. Long short-term memory neural network for air pollutant concentration predictions: method development and evaluation. Environ. Pollut. 231, 997–1004. https://doi.org/10.1016/j.envpol.2017.08.114.
Li, X., Zhang, X., Zhang, Z., Han, L., Gong, D., Li, J., Wang, T., Wang, Y., Gao, S., Duan, H., Kong, F., 2019. Air pollution exposure and immunological and systemic inflammatory alterations among schoolchildren in China. Sci. Total Environ. 657, 1304–1310. https://doi.org/10.1016/j.scitotenv.2018.12.153.
Lin, Y., Cobourn, W.G., 2007. Fuzzy system models combined with nonlinear regression for daily ground-level ozone predictions. Atmos. Environ. 41, 3502–3513. https://doi.org/10.1016/j.atmosenv.2006.11.060.
Liu, B., Yan, S., Li, J., Qu, G., Li, Y., Lang, J., Gu, R., 2019. A sequence-to-sequence air quality predictor based on the n-step recurrent prediction. IEEE Access 7, 43331–43345. https://doi.org/10.1109/ACCESS.2019.2908081.
Lv, M., Li, Y., Chen, L., Chen, T., 2019. Air quality estimation by exploiting terrain features and multi-view transfer semi-supervised regression. Inf. Sci. 483, 82–95.
Ma, J., Cheng, J.C.P., 2016a. Identifying the influential features on the regional energy use intensity of residential buildings based on Random Forests. Appl. Energy 183, 193–201. https://doi.org/10.1016/j.apenergy.2016.08.096.
Ma, J., Cheng, J.C.P., 2016b. Data-driven study on the achievement of LEED credits using percentage of average score and association rule analysis. Build. Environ. 98, 121–132. https://doi.org/10.1016/j.buildenv.2016.01.005.
Ma, J., Cheng, J.C.P., 2016c. Estimation of the building energy use intensity in the urban scale by integrating GIS and big data technology. Appl. Energy 183, 182–192. https://doi.org/10.1016/j.apenergy.2016.08.079.
Ma, J., Cheng, J.C.P., 2017. Identification of the numerical patterns behind the leading counties in the U.S. local green building markets using data mining. J. Clean. Prod. 151, 406–418. https://doi.org/10.1016/j.jclepro.2017.03.083.
Neagu, C.-D., Avouris, N., Kalapanidas, E., Palade, V., 2002. Neural and neuro-fuzzy integration in a knowledge-based system for air quality prediction. Appl. Intell. 17, 141–169. https://doi.org/10.1023/A:1016108730534.
Niska, H., Hiltunen, T., Karppinen, A., Ruuskanen, J., Kolehmainen, M., 2004. Evolving the neural network model for forecasting air pollution time series. Eng. Appl. Artif. Intell. Intell. Control Signal Process. 17, 159–167. https://doi.org/10.1016/j.engappai.2004.02.002.
Osowski, S., Garanty, K., 2007. Forecasting of the daily meteorological pollution using wavelets and support vector machine. Eng. Appl. Artif. Intell. 20, 745–755. https://doi.org/10.1016/j.engappai.2006.10.008.
Pan, S.J., Yang, Q., 2010. A survey on transfer learning. IEEE Trans. Knowl. Data Eng. 22, 1345–1359. https://doi.org/10.1109/TKDE.2009.191.
Qi, Z., Wang, T., Song, G., Hu, W., Li, X., Zhang, Z., 2018. Deep air learning: interpolation, prediction, and feature analysis of fine-grained air quality. IEEE Trans. Knowl. Data Eng. 30, 2285–2297.
Qi, Y., Li, Q., Karimian, H., Liu, D., 2019. A hybrid model for spatiotemporal forecasting of PM2.5 based on graph convolutional neural network and long short-term memory. Sci. Total Environ. 664, 1–10. https://doi.org/10.1016/j.scitotenv.2019.01.333.
Reddy, V., Yedavalli, P., Mohanty, S., Nakhat, U., 2018. Deep Air: Forecasting Air Pollution in Beijing, China.
Rubal, Kumar, D., 2018. Evolving Differential evolution method with random forest for prediction of Air Pollution. Procedia Comput. Sci. Int. Conf. Comput. Intell. Data Sci. 132, 824–833. https://doi.org/10.1016/j.procs.2018.05.094.
Saide, P.E., Carmichael, G.R., Spak, S.N., Gallardo, L., Osses, A.E., Mena-Carrasco, M.A., Pagowski, M., 2011. Forecasting urban PM10 and PM2.5 pollution episodes in very stable nocturnal conditions and complex terrain using WRF–Chem CO tracer model. Atmos. Environ. 45, 2769–2780. https://doi.org/10.1016/j.atmosenv.2011.02.001.
Salman, A.G., Heryadi, Y., Abdurahman, E., Suparta, W., 2018. Single layer & multi-layer long short-term memory (LSTM) model with intermediate variables for weather forecasting. In: Procedia Comput. Sci., the 3rd International Conference on Computer Science and Computational Intelligence (ICCSCI 2018) : Empowering Smart Technology in Digital Era for a Better Life, vol. 135. pp. 89–98. https://doi.org/10.1016/j.procs.2018.08.153.
Singh, K.P., Gupta, S., Kumar, A., Shukla, S.P., 2012. Linear and nonlinear modeling approaches for urban air quality prediction. Sci. Total Environ. 426, 244–255. https://doi.org/10.1016/j.scitotenv.2012.03.076.
Slini, T., Karatzas, K., Moussiopoulos, N., 2002. Statistical analysis of environmental data as the basis of forecasting: an air quality application. Sci. Total Environ. 288, 227–237. https://doi.org/10.1016/s0048-9697(01)00991-3.
Soh, P., Chang, J., Huang, J., 2018. Adaptive deep learning-based air quality prediction model using the most relevant spatial-temporal relations. IEEE Access 6, 38186–38199. https://doi.org/10.1109/ACCESS.2018.2849820.
Stern, R., Builtjes, P., Schaap, M., Timmermans, R., Vautard, R., Hodzic, A., Memmesheimer, M., Feldmann, H., Renner, E., Wolke, R., Kerschbaumer, A., 2008. A model inter-comparison study focussing on episodes with elevated PM10 concentrations. Atmos. Environ. 42, 4567–4588. https://doi.org/10.1016/j.atmosenv.2008.01.068.
Stojov, V., Koteli, N., Lameski, P., Zdravevski, E., 2018. Application of Machine Learning and Time-Series Analysis for Air Pollution Prediction.
Suleiman, A., Tight, M.R., Quinn, A.D., 2018. Applying machine learning methods in managing urban concentrations of traffic-related particulate matter (PM 10 and PM 2.5). Atmos. Pollout. Res. 10 (1), 134–144. https://doi.org/10.1016/j.apr.2018.07.001.
Sun, W., Sun, J., 2017. Daily PM2.5 concentration prediction based on principal component analysis and LSSVM optimized by cuckoo search algorithm. J. Environ. Manag. 188, 144–152. https://doi.org/10.1016/j.jenvman.2016.12.011.
Wang, Z., Maeda, T., Hayashi, M., Hsiao, L.-F., Liu, K.-Y., 2001. A nested air quality prediction modeling system for urban and regional scales: application for high-ozone episode in taiwan. Water Air Soil Pollut. 130, 391–396. https://doi.org/10.1023/A:1013833217916.
Wang, T., Jiang, F., Deng, J., Shen, Y., Fu, Q., Wang, Q., Fu, Y., Xu, J., Zhang, D., 2012. Urban air quality and regional haze weather forecast for Yangtze River Delta region. Atmos. Environ. 58, 70–83. https://doi.org/10.1016/j.atmosenv.2012.01.014.
Wang, L., Geng, X., Ma, X., Liu, F., Yang, Q., 2018. Cross-city transfer learning for deep spatio-temporal prediction. ArXiv Prepr. ArXiv180200386.
Wen, C., Liu, S., Yao, X., Peng, L., Li, X., Hu, Y., Chi, T., 2019. A novel spatiotemporal convolutional long short-term neural network for air pollution prediction. Sci. Total Environ. 654, 1091–1099. https://doi.org/10.1016/j.scitotenv.2018.11.086.
Yang, Z., Wang, J., 2017. A new air quality monitoring and early warning system: air quality assessment and air pollutant concentration prediction. Environ. Res. 158, 105–117. https://doi.org/10.1016/j.envres.2017.06.002.
Ye, R., Dai, Q., 2018. A novel transfer learning framework for time series forecasting. Knowl. Based Syst. 156, 74–99. https://doi.org/10.1016/j.knosys.2018.05.021.
Yosinski, J., Clune, J., Bengio, Y., Lipson, H., 2014. How transferable are features in deep neural networks? In: In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q. (Eds.), Advances in Neural Information Processing Systems, vol. 27. Curran Associates, Inc., pp. 3320–3328.
Zhenghua, W., Zhihui, T., 2017. Prediction of air quality index based on improved neural network. In: 2017 International Conference on Computer Systems, Electronics and Control (ICCSEC). Presented at the 2017 International Conference on Computer Systems, Electronics and Control (ICCSEC), pp. 200–204. https://doi.org/10.1109/ICCSEC.2017.8446883.
Zhou, C., Wei, G., Zheng, H., Russo, A., Li, C., Du, H., Xiang, J., 2019. Effects of potential recirculation on air quality in coastal cities in the Yangtze River Delta. Sci. Total Environ. 651, 12–23. https://doi.org/10.1016/j.scitotenv.2018.08.423.
Rolling analysis of time series. In: Zivot, E., Wang, J. (Eds.), Modeling Financial Time Series with S-PLUS®. Springer New York, New York, NY, pp. 313–360. https://doi.org/10.1007/978-0-387-32348-0_9.