Expert Systems With Applications 211 (2023) 118422

Contents lists available at ScienceDirect

Expert Systems With Applications

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

An air quality prediction model based on improved *Vanilla* LSTM with multichannel input and multiroute output

Wei Fang [a,*], Runsu Zhu [a], Jerry Chun-Wei Lin [b]

[a] Jiangsu Provincial Engineering Laboratory of Pattern Recognition and Computational Intelligence, Jiangnan University, Wuxi, Jiangsu, China
[b] Department of Computer Science, Electrical Engineering and Mathematical Sciences, Western Norway University of Applied Sciences, Bergen, Norway

ARTICLE INFO

Keywords:
Long short-term memory (LSTM)
Air quality prediction
Deep learning
Dynamic time warping

ABSTRACT

Long short-term memory (LSTM), especially *vanilla* LSTM (VLSTM), has been widely used in air quality prediction field. However, VLSTM has many more parameters, thereby making training slow and prediction performance unstable. The VLSTM network input data have not been selected for better efficiency. In this paper, we propose an air quality prediction model based on the improved VLSTM with multichannel input and multiroute output (IVLSTM-MCMR). The proposed model includes the IVLSTM and MCMR modules. The proposed IVLSTM module is developed by improving the VLSTM inner structure of VLSTM in order to reduce the number of parameters that help to accelerate the convergence. A new historical information usage approach is further proposed to obtain a stable training process. For the MCMR module, a multichannel data input model (MC) with an improved linear similarity dynamic time warping is introduced to choose the valid data as the input of IVLSTM. A multiroute output model (MR) is designed to integrate the results from MC, in which the results of different target stations with different features are output by different routes. We evaluate the proposed model with the collected data from Beijing, China, and the experimental results show that our model achieves improvements regarding the predication performance.

1. Introduction

Clean air is a basic requirement to keep humans fit (Chiu & Yang, 2015; D'Amato, Vitale, Lanza, Molino, & D'Amato, 2016). Air quality has therefore drawn much more attention, especially the concentration of $PM2.5$ (Chen et al., 2017; Conti, Heibati, Kloog, Fiore, & Ferrante, 2017). From the earliest establishment of air pollution diffusion models through empirical knowledge (Pfender, Graw, Bradley, Carney, & Maxwell, 2006; Vardoulakis, Fisher, Pericleous, & Gonzalez-Flesca, 2003; Wayland, White, Dickerson, & Dye, 2002), to linear regression (McCollister & Wilson, 1975) (LR), random forests (Yu, Yang, Yang, Han, & Move, 2016), and other traditional machine learning methods (Wang, Zhang, Guo, & Lu, 2017), researchers have tried a variety of methods of air quality prediction. With the rapid development of machine learning, deep learning networks, especially long short-term memory (LSTM), have been widely used for air quality prediction (Du, Li, Yang, & Horng, 2021; Freeman, Taylor, Gharabaghi, & Thé, 2018; Kim et al., 2010; Kim, Kim, Sung, & Yoo, 2009; Qi et al., 2018; Yan et al., 2021).

LSTM was first proposed in Hochreiter and Schmidhuber (1997); then, LSTM included only the cell, the input gate, and the output gate. LSTM did not include the forget gate and had no peephole connections. In this structure, full gate recurrence was used, but an exact gradient was not used for network training. The proposal of the forget gate (Gers, Schmidhuber, & Cummins, 2000) enabled LSTM to reset its own cell states. To learn precise timings, cell states need to be the input to the gates (Gers & Schmidhuber, 2000). In this way, the historical information generated during training process can be used to make precise timings easier to learn. Full backpropagation through time (BPTT) training is proposed to check LSTM gradients using finite differences (Graves & Schmidhuber, 2005). Compared to the standard approaches (Hochreiter & Schmidhuber, 1997), *Vanilla* LSTM (VLSTM), as shown in Fig. 1, incorporates the improvements mentioned above and shows outstanding prediction ability (Greff, Srivastava, Koutník, Steunebrink, & Schmidhuber, 2017) compared to the standard approaches (Hochreiter & Schmidhuber, 1997).

Although VLSTM has been widely used (Kanjo, Younis, & Ang, 2019; Liu et al., 2020; Liu, Wang and Xu, 2020) in many domains and applications, it produces more parameters than the original LSTM structure proposed in Hochreiter and Schmidhuber (1997), thereby making it more difficult to converge during the training (Allen-Zhu, Li, & Song, 2019; Ergen & Kozat, 2018; Greff et al., 2017). On the one hand, during the network training process, the updating cost of the

* Corresponding author.
E-mail addresses: fangwei@jiangnan.edu.cn (W. Fang), 6181910040@stu.jiangnan.edu.cn (R. Zhu), jerrylin@ieee.org (J.C.-W. Lin).

https://doi.org/10.1016/j.eswa.2022.118422
Received 24 October 2021; Received in revised form 20 July 2022; Accepted 3 August 2022
Available online 27 August 2022

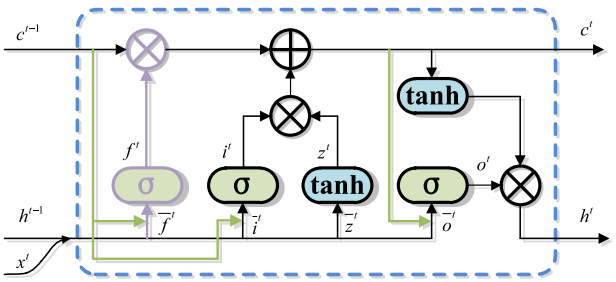


Fig. 1. The inner structure of VLSTM.

parameters becomes more expensive with the increasing of the number of parameters. On the other hand, higher dimension of parameter space makes the training process more complex. The complexity makes the network more possible to be trapped in the local optimum, which results in the training process much longer or achieves an overfitting network. Additionally, the way historical information is used by VLSTM makes the recurrent training process more sensitive to the input data; consequently, a few abnormal data may cause much degradation to the convergence of the whole deep learning network. The input data is air quality data, which is collected from the terminal sensors and has high possibility to be ill-conditioned data. When the data from terminal sensors is missing, the collected value can only be zero which is not a true value. Suppose that when the collected values are high in a period, a missing data will cause a steeply drop and a steeply arise within 2 timestamps, which may cause the degradation to the convergence of the whole deep learning network. The accuracy of air quality prediction by VLSTM networks is also affected by the input data and the output method to some extent, so the validity of input data and efficiency of output integration are important as well. An established model, such as VLSTM (Wang & Song, 2018), makes some progress on data input selection and output integration.

Therefore, to address these issues, we propose an improved VLSTM with multichannel input and multiroute output model (IVLSTM-MCMR) for air quality prediction. Two modules are then developed and investigated in IVLSTM-MCMR: the IVLSTM module and MCMR module. The IVLSTM module aims to reduce the number of parameters, and it also makes better use of historical information during model training thus the training process could be more stable. MCMR module consists of a multichannel input model (MC) and a multiroute output model (MR), which is used to address valid data selection problems. In MC, an improved data selection method based on the linear similarity dynamic time warping (LS-DTW) is introduced. MR is proposed to find a better way to aggregate the output results from five different channels generated by MC.

The structure of this paper is organized as follows. Related work is given in Section 2. The proposed VLSTM-MCMR is introduced in Section 3. The performance of VLSTM-MCMR is evaluated and compared with the performance of state-of-the-art methods in Section 4. Finally, the conclusion is drawn in Section 5.

2. Related work

For air quality prediction, several methods and models have been proposed in the literature and can generally be divided into three categories (Zhang, Bocquet, Mallet, Seigneur, & Baklanov, 2012). The first type comprises models based on the empirical assumptions (Pfender et al., 2006; Vardoulakis et al., 2003; Wayland et al., 2002). The second type comprises models that use parametric or nonparametric statistical methods (Donnelly, Misstear, & Broderick, 2015; Guizilini & Ramos, 2015). The last type uses deep learning networks to predict air quality (Han, Lam, Li, & Zhang, 2020; Navares & Aznarte, 2020; Soh, Chen, Huang, & Chu, 2017; Wang & Song, 2018; Wen et al., 2019; Yi et al., 2020).

Empirical approaches include the persistence method (McKeen et al., 2009), climatology method (Cheng et al., 2007), and empirical method (Pfender et al., 2006; Vardoulakis et al., 2003; Wayland et al., 2002), which are too complicated to be eliminated, and the results are not good. Approaches such as regression trees (Burrows et al., 1995), Gaussian process regression (Nguyen-Tuong, Peters, & Seeger, 2009), and least squares (Ip, Vong, Yang, & Wong, 2010) are parametric or nonparametric statistical methods, that compared with empirical approaches, have better prediction results. However, these statistical methods still cannot perform well in complex situations. Artificial neural networks (ANNs), which are also parametric methods, can improve prediction performance by their good performance on nonlinear problems (McKendry, 2002; Zhao, Zhang, Wang, Bai, & Liu, 2010), but it cannot make full use of input time series.

Recurrent neural network (RNNs) (Kim et al., 2010, 2009) and LSTM (Fan et al., 2017; Li et al., 2017) are the most commonly used deep learning methods in the field of air quality prediction. LSTM outperforms RNNs (Fan et al., 2017). Since LSTM was proposed in 1997 (Hochreiter & Schmidhuber, 1997), researchers have also made many improvements to the structure of LSTM. As is known that, air quality prediction is a kind of time series problem. In Ang, Ng, and Chua (2020) and Torres, Hadjout, Sebaa, Martínez-Álvarez, and Troncoso (2021), a set of deep learning methods for time series analysis and forecasting have been reviewed, including GRU, ResNet, SENet, convLSTM, LSTM-FCN, LSTM-RNN, CNN-LSTM, DRNN, TCN, etc. VLSTM has a better effect on precise timing prediction because of VLSTM's peephole connections (Gers & Schmidhuber, 2000; Gers, Schraudolph, & Schmidhuber, 2002). Li et al. first applied VLSTM to air quality prediction (Li et al., 2017). Wang et al. used put weather data as VLSTM inputs to improve the accuracy of the final prediction to a certain extent (Wang & Song, 2018). Because of its deep learning network module, the spatial–temporal ensemble (STE) model in Wang and Song (2018) outperforms the ensemble model proposed by Zheng et al. (2015). Wu et al. used VLSTM to predict the wavelet value to achieve the final goal of predicting the air quality index by using basic wavelet decomposition technology (Wu & Lin, 2019). Soh, Chang, and Huang (2018), Soh et al. (2017) and Wen et al. (2019) combined a CNN and VLSTM for air quality prediction; the authors aim to introduce more auxiliary data to improve the output performance of VLSTM. Most of the work on air quality prediction based on LSTM focuses on increasing the type and quantity of auxiliary data to improve the prediction effect, while optimizing the internal LSTM structure has not been considered. Thus, the widely used internal LSTM structure is still the VLSTM structure. In addition, it is useful to consider separating these data into multiple independent channels to avoid negative effects between data of different aspects. Then, a multiple-choice output route of results integration is helpful.

3. Methodology

In this section, we present the proposed IVLSTM-MCMR model for air quality prediction in detail. In IVLSTM-MCMR, an IVLSTM structure is first proposed as a deep learning network module with a reduced number of parameters and better usage of historical information. The MCMR module in IVLSTM-MCMR is a more efficient data input and output method, which is designed for choosing valid data in deep learning networks and aggregating the final results as another component.

3.1. Overview

The overall structure of the IVLSTM-MCMR model is shown in Fig. 2. The model can be divided into the IVLSTM module and MCMR module. In the IVLSTM module, new internal structures are proposed in order to reduce the number of VLSTM parameters and find a new way

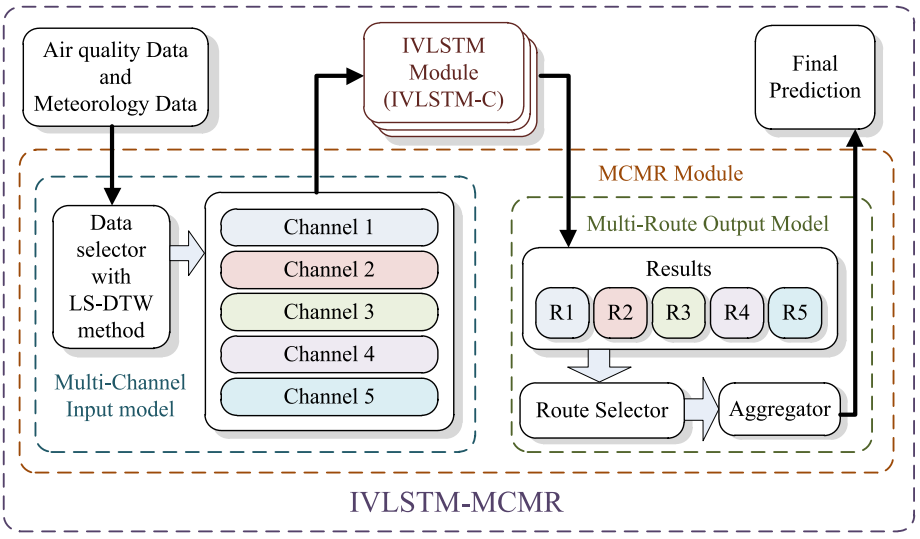


Fig. 2. The overall structure of IVLSTM-MCMR model for air quality prediction.

to make use of historical information generated during training process. Fewer parameters can make the deep learning network converge more easily, and the historical information used in the new method can make the training process more stable. The MCMR module, it consists of the MC model and MR model. Data series of multiple channels are chosen by the LS-DTW algorithm in the MC model. In this way, data series with high validity are input into the IVLSTM module separately in case the data negatively affect each other. Multiple input channels generate multiple outputs, which need to be aggregated. Then, MR is designed to find an appropriate route to aggregate results that are separate from the IVLSTM module, depending on the inherent features of the target stations.

3.2. IVLSTM module

In general, VLSTM has more parameters, thereby possibly decelerating training and reducing accuracy. In addition, the way VLSTM uses historical information generated by training is inappropriate and makes training unstable. To address these two problems, three variants of VLSTM have been studied in this paper; these variants are shown in Fig. 3.

As shown in Fig. 3(a), the forget gate and the input gate develop a complementary relationship in the module, which is called IVLSTM-A. When calculating the input gate and forget gate values, the computational effort of IVLSTM-A is half that of VLSTM. If the number of parameters is reduced, the model more easily converges during training. The model with fewer parameters trained under the same number of iterations should have a higher prediction accuracy. The forward propagation formulas are shown as Eqs. (1) to (6).

$$i^t = \sigma(W_{ix^t} x^t + W_{ih^{t-1}} h^{t-1} + W_{ic^{t-1}} c^{t-1} + b_i), \tag{1}$$

$$f^t = 1 - i^t, \tag{2}$$

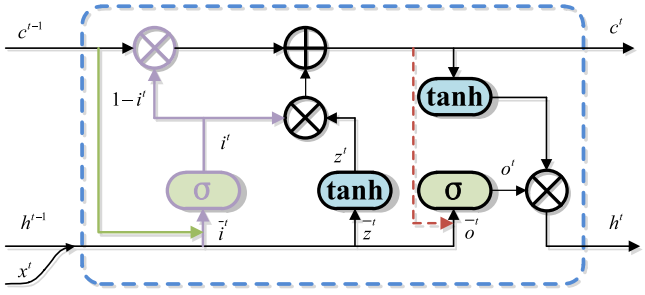
$$z^t = \tanh(W_{zx^t} x^t + W_{zh^{t-1}} h^{t-1} + b_z), \tag{3}$$

$$c^t = i^t \odot z^t + f^t \odot c^{t-1}, \tag{4}$$

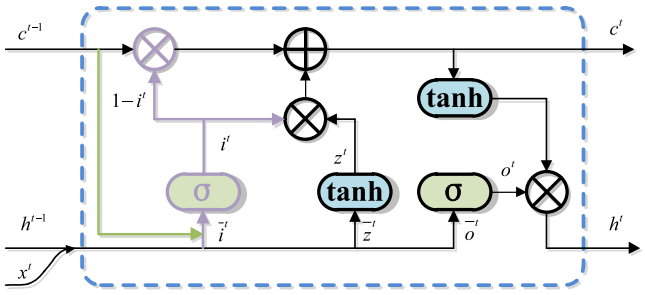
$$o^t = \sigma(W_{ox^t} x^t + W_{oh^{t-1}} h^{t-1} + W_{oc^t} c^t + b_o), \tag{5}$$

$$h^t = \tanh(c^t) \odot o^t. \tag{6}$$

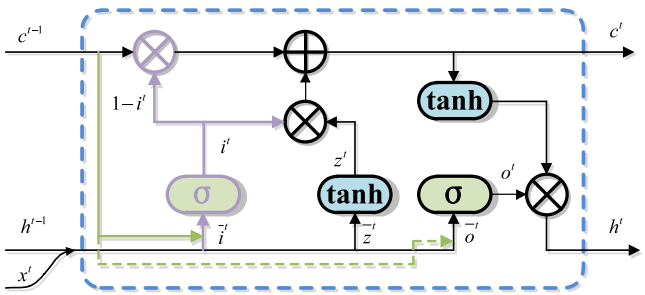
where x represents the input vector of the new data; c represents the input vector of the cell state; h represents the input vector of the loop output; t represents the time node where the current VLSTM unit is located; f, i, z, and o respectively represent the forget gate, input gate, acceleration gate, and output gate, respectively; W_{mn} represents the



(a) The Inner Structure of IVLSTM-A



(b) The Inner Structure of IVLSTM-B



(c) The Inner Structure of IVLSTM-C

Fig. 3. Three variants of VLSTM.

weight vector corresponding to the input vector n in gate m; and b_m represents the bias vector of gate m.

The historical information in the VLSTM training is stored mainly in the VLSTM's cell state. During the training, c^{t-1} is used as a part of the input vector, which uses the historical information input into the VLSTM unit at time t. Additionally, c^t, as the updated cell state, contains more information at time t. Therefore, in IVLSTM-A, the c^t vector is added to the output gate. In this way, $\tanh(c^t)$ and o^t, which are the two point multiplication factors of h^t, are both affected by the change in c^t, as shown in Eq. (6). The utility of c^t makes the final output h^t too sensitive to the change in c^t, thereby increasing the instability of the training process. Therefore, based on IVLSTM-A and IVLSTM-B, as shown in Fig. 3(b), replacing the forward propagation, that is, replacing Eq. (5) with Eq. (7), is proposed for reducing the sensitivity of h^t to c^t

$$o^t = \sigma(W_{ox^t} x^t + W_{oh^{t-1}} h^{t-1} + b_o) \tag{7}$$

To further improve the accuracy of network prediction, as shown in Fig. 3(c), IVLSTM-C is designed based on IVLSTM-B. The added path in IVLSTM-C introduces c^{t-1} information into the output gate. On the one hand, this change can enhance the use of historical information. In addition, this change reduces the impact of abnormal cell states during the propagation, thereby making the convergence of the deep learning network training process more stable. The VLSTM structure has an important effect in time series prediction. The correct and effective use of historical information is the key to VLSTM's high-precision prediction. IVLSTM-C uses the information of c^{t-1} three times, and at the same time, node t is used to enhance the effect of historical information on the output value at time point t. The input of c^{t-1} to the output gate also neutralizes the abnormal fluctuation of c^t in another part of the output information $\tanh(c^t)$. When the similarity between c^{t-1} and c^t is high, IVLSTM-C is similar to IVLSTM-A since the cell state vectors are similar. There can be a certain enhancement to the same features of historical information. Two situations are then arise when the similarity between c^{t-1} and c^t is extremely low. One is that when the similarity between c_t and c_{t+1} is extremely high, thus indicating that the change between c^{t-1} and c^t is steady, the large change in cell state is not due to abnormal values but features that need to be changed. Under these circumstances, the features are enhanced at time $t+1$. The other situation is that c^{t-1} is similar to c^{t+1} but is very different from c^t, thus indicating that the new cell state generated at time t is abnormal. In time slices t and $t+1$, the abnormal features are neutralized by c^{t-1} and c^{t+1}. This process makes a stable iteration performed on the weights, and a more stable output is produced. The final forward propagation formula needs only to replace Eq. (7) with Eq. (8).

$$o^t = \sigma(W_{ox^t} x^t + W_{oh^{t-1}} h^{t-1} + W_{oc^{t-1}} c^{t-1} + b_o) \tag{8}$$

3.3. MCMR model

A complete deep learning network model requires not only an appropriate inner structure but also effective data input and output methods. As a real-world practical problem, air quality prediction by using the IVLSTM module requires a data selection model to choose valid input data and an aggregation model to output final prediction results. In this section, the MC input model is proposed to select valid data corresponding to the target station and form different data input channels. The usage of the MR output model to aggregate the results generated by IVLSTM channels is also introduced here. Through five input channels, effective data are selected from the overall information in different aspects to achieve a more comprehensive use of global data. Through three output routes, paths are selected according to the features of the target station, and the corresponding channel output results are integrated to obtain the final prediction.

3.3.1. MC input model

In the MC model, five channels are generated to train IVLSTM separately to prevent data interference. The data of the predicted target station are divided into Channels 1 and 2, while the data of location-related stations of the target station are set as Channel 3. Channels 4 and 5 contain the data of the time-related stations of the predicted target. The data of the predicted target station are directly input to the channel without any processing. The data of location-related stations of the target station are chosen by the KNN algorithm (Wang & Song, 2018). The remaining two channels are time-related station data channels, which require targeted selection. The focus of this aspect is how to select the station more related to the time series data of the target station.

DTW (Soh et al., 2017) is an algorithm that compares the similarity of two time series through dynamic programming. This algorithm can well reflect the similarity under a nonlinear relationship by folding time and can be used as a similarity selection for time-related stations of the target station. However, for a real-world problem, the inputs of DTW have multiple dimensions. To better determine similarity, DTW should pay different attention to different dimensions. In this paper, LS-DTW is proposed as a new algorithm to make the similarity of two real-world time series data more reasonable and will be utilized for the selecting of time-related stations. The purpose is to strengthen the influence of similarity by using different factors of target station time series data on DTW selection. Air quality factors or weather factors with a stronger similarity to the target prediction sequence should be assigned greater weights based on DTW selection. The LS-DTW algorithm contains two components. The first component calculates the linear similarity index through linear misalignment regression between the time series of the nonpredicted factors and the predicted factor of the target station, while the second component is DTW process between the data of the target station and the data of other stations. Suppose we have two time series inputs, A and B, and the output by the DTW process if the similarity index of A and B.

$$X_1 = (a_{i+1} \quad a_{i+2} \quad \cdots \quad a_L)^T \tag{9}$$

$$M_1 = \begin{pmatrix} 1 & a_i & a_{i-1} & \cdots & a_1 \\ 1 & a_{i+1} & a_i & \cdots & a_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & a_{L-1} & a_{L-2} & \cdots & a_{L-i-1} \end{pmatrix} \tag{10}$$

$$e_1 = f_{LR}(X_1, M_1) \tag{11}$$

$$X_2 = (a_{j+t+1} \quad a_{j+t+2} \quad \cdots \quad a_L)^T \tag{12}$$

$$M_2 = \begin{pmatrix} 1 & a_{j+t} & \cdots & a_{j+1} & b_{j+t+1} & \cdots & b_{t+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & a_{L-1} & \cdots & a_{L-t} & b_L & \cdots & b_{L-j+1} \end{pmatrix} \tag{13}$$

$$e_2 = f_{LR}(X_2, M_2) \tag{14}$$

As shown in Eqs. (9) to (14), a_n denotes the nth value of time series A and b_m denotes the mth value of time series B, X denotes the target vector of LR, M denotes the misalignment metric, e denotes the error of the LR process, i and j denote counters for the whole cycle, L denotes the length of series A and B, and t denotes the value of counter i when e_1 obtains its minimum value. The minimum error of e_2 is calculated, and $\frac{1}{e_2+1}$ is used as the similarity coefficient. The smaller e_2 is, the closer the similarity coefficient is to 1. The closer the similarity coefficient is to 1, the more similar the target series and compared series are. For the second part, ten-dimensional data of target Station T and other Stations O are put into the DTW process to obtain a ten-dimensional vector of DTW values. Finally, the similarity index calculated by the first part is used as a reference to make DTW pay different attention to different dimensions.

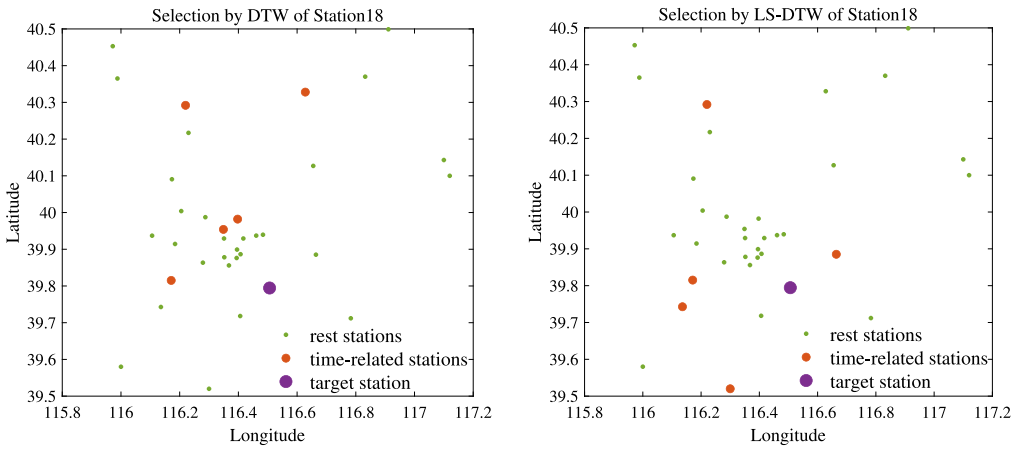


Fig. 4. Comparison of DTW and LS-DTW selection.

The whole process of LS-DTW is shown in Alg. 1. The attention weights of different dimensions of the target station are stated in Lines 1–3. Full-dimensional DTW distances between other stations and the target station are stated in Lines 4–8. Then, the weighted values of all stations are calculated in Lines 9–14. The top Nth values of similarity between stations are found in Lines 15 and 16. We find that the stations chosen by LS-DTW and DTW actually have large differences. As shown in Fig. 4, only two of the top five stations with valid data and chosen by using DTW and LS-DTW, only two of them are the same stations; the other three stations vary greatly by geographic location. This comparison result indicates that LS-DTW finds a new way to define stations with higher data validity.

Algorithm 1 LS-DTW

Input: The time series of historical air quality and meteorology of all stations, X; the time series of PM2.5 of target station, Y; the number of the most time-related stations N.
Output: The set of the most N^{th} time-related stations S.
1: **for** each factor i of target station t **do**
2: $W_i = LinearSimilarity\left(Y, X_{ti}\right)$;
3: **end for**
4: **for** each station r **do**
5: **for** each factor i of station r **do**
6: D_{ir} =DynamicTimeWarping$\left(X_{ti}, X_{ri}\right)$;
7: **end for**
8: **end for**
9: **for** each other station r **do**
10: $Weighted\,Dist_R = W \cdot D_r$;
11: **if** r equals t **then**
12: $Weighted\,Dist_r = 0$;
13: **end if**
14: **end for**
15: $Stationid$ =Sort($Weighted\,Dist$);
16: $S \leftarrow$ the top N^{th} of $Stationid$;

3.3.2. MR output model

The MC model generates five channels, and each channel corresponds to a dependent IVLSTM module; these channels also output five prediction results separately. Combining these results mechanically is not the best solution. Therefore, the MR output model is proposed here to find a way to organically aggregate these results, as shown in Fig. 5. The output method is then used to find the optimal solution from the final prediction results that achieved from the MC model.

In the MR model, based on the determination of five data input channels, these results are selectively used to form different aggregation routes. In the real world, monitoring stations are divided into three

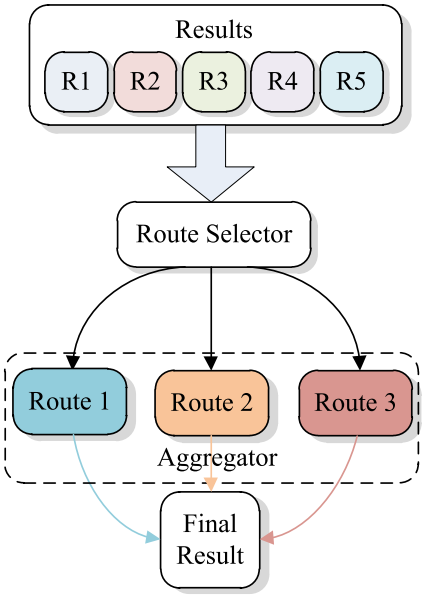


Fig. 5. The MR structure.

main categories. One comprises the stations that are the source of air pollution. The main feature of these stations is the continuous generation of air pollution factors near themselves, such as stations located in the main traffic road or near the factory. The other comprises stations that are the main recipients of air pollutants. These stations are characterized by the fact that they do not produce air pollutants but are affected by the diffusion of pollutants from their surroundings. The last category contains complex mixed stations with both features of the above two categories. On the one hand, these stations are air pollution sources. However, these stations are also affected by the spread of air pollutants from the surrounding area.

According to the respective features of these three kinds of stations, three kinds of routes corresponding to the above mentioned different situation stations in turn are designed in this paper. Concerning the first kind of station, since its air pollutants radiate to the outside centred on itself and are not easily affected by the pollutants from surrounding stations, we use only the route with input channels containing the information about this station itself to make a final prediction of the station's its air quality. Concerning the second kind of station, since it does not produce air pollutants and is affected mainly by the

surrounding environment, we use only the route with input channels containing information related to the station to make a final prediction of its air quality. Concerning the third kind of station, we use the route with five input channels together for the final prediction to ensure the final concentration of air pollutants around the station. Due to the synergy of the two factors mentioned in relation to the first and second kinds of stations, more accurate prediction results can thus be obtained.

Algorithm 2 Generation of IVLSTM-MCMR model

Input: The time series of historical air quality and meteorology of all stations, X; the time series of target station, T;.
Output: The complete model $IVLSTM\text{-}MCMR$.
1: **if** $Channel_i$ has no data **then**
2: $Channel_{1,2}$ =Divide(T);
3: $Channel_3$ =KNN(X, T);
4: S =LS-DTW$(X, T_{PM2.5})$;
5: $Channel_{4,5}$ =Divide(X_S);
6: **end if**
7: **for** each $Channel_i$ in $ChannelSet$ **do**
8: $Channel_i$ is used to train $IVLSTM$ and get $Result_i$;
9: **end for**
10: **for** each result $Result_i$ in the set of all results $ResultSet$ **do**
11: **if** $Result_i$ correspond to features of target station **then**
12: $Result_i$ is merged into set E;
13: **end if**
14: **end for**
15: $MRModel$ =SVR($E, groundtruth$);
16: $IVLSTM\text{-}MCMR$ is MC plus $IVLSTM$ plus MR.

Finally, the overall process of IVLSTM-MCMR is shown in Alg. 2. In Lines 1–6, all of the data are input into the MC part of IVLSTM-MCMR first and divided into five different channels by using the KNN and LS-DTW methods. In Lines 7–9, data from five channels are sent to the IVLSTM module to train the IVLSTM networks. In Lines 10–15, the results generated in different IVLSTMs are put into the MR part and integrated through different routes. Finally, in Line 16, IVLSTM-MCMR consists of the three parts generated in Lines 1–15.

4. Experimental results and analysis

4.1. Datasets and parameter settings

Air pollution data and meteorology data from 35 monitoring stations in Beijing from 1 May 2014 to 30 April 2018 are used as datasets, in which the timestamp is set as an hour to update the data periodically. Air pollution data includes CO, NO_2, SO_2, O_3, $PM10$, and $PM2.5$. Meteorological data include temperature, humidity, wind direction, and wind speed, which are updated eight times a day. The datasets used in the experiments are obtained from Wang and Song (2018) and Zheng et al. (2015). Among all of the factors, the concentration of $PM2.5$ is an essential component of the AQI (Conti et al., 2017) and is used as a predictive target. The input data are aligned by performing the mean interpolation on the meteorological data. A random noise is also added to the supplemented data in order to avoid manual regular intervention in the training of the deep learning model. All the data are then normalized and scaled to [0,1] in the preprocessing step to ensure that all the factors in the datasets can be fully used.

The final IVLSTM module investigated in this paper is a three hidden layer network. The first layer of the IVLSTM module is the LSTM network, which connects two fully connected layers for data regression. The LSTM layer node parameters are set to 20, and the fully connected layers are 70 and 30. We use a 10:1 ratio to partition the data into non-overlapped training set and testing set by using the SGD update method with 40 batch sizes, and the training epoch is set as 100. The initial learning rate is set as 1 and is decreased by 0.99 per generation. All the weight matrices are initialized randomly.

Table 1
Results of LSTM inner structures of the time interval of 6 h prediction.

Structure	MAE	Std. of MAE	RMSE	Std. of RMSE	Acc.	Std. of Acc.
GRU	23.22	0.3880	32.15	0.5496	0.5899	0.0067
$VLSTM$	23.04	0.5135	31.04	0.6593	0.5924	0.0087
$IVLSTM-A$	22.92	0.2377	31.08	0.3161	0.5961	0.0039
$IVLSTM-B$	22.87	0.2620	31.00	0.3257	0.5973	0.0046
$IVLSTM-C$	**22.82**	0.3161	**30.88**	0.4050	**0.5978**	0.0053

4.2. Performance metrics

We use the mean absolute error (MAE), root mean square deviation (RMSE), and accuracy to evaluate the effectiveness and robustness of the designed model, and these metrices are defined as follows from Eqs. (15) to (17).

$$MAE = \frac{\sum_i |\hat{y}_i - y_i|}{n} \tag{15}$$

$$RMSE = \sqrt{\frac{\sum_i (\hat{y}_i - y_i)^2}{n}} \tag{16}$$

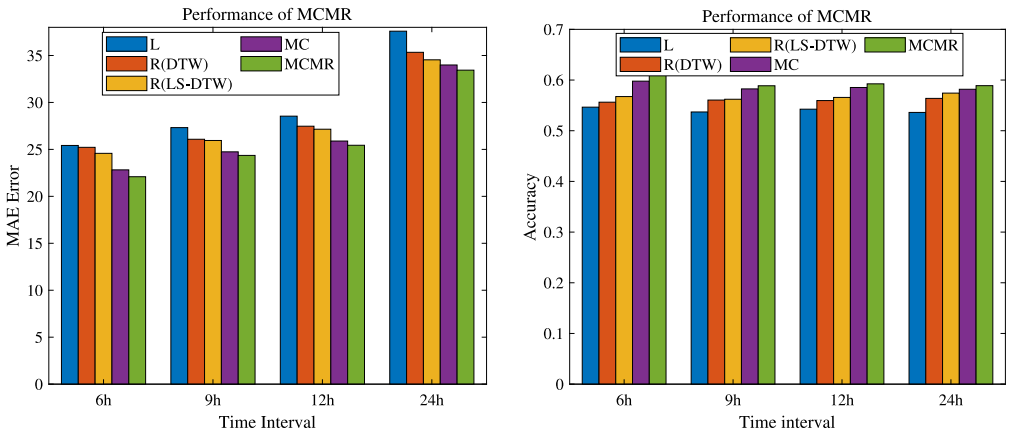
$$Accuracy = 1 - \frac{\sum_i |\hat{y}_i - y_i|}{\sum_i y_i} \tag{17}$$

4.3. Experimental results

A set of experiments is carried out to evaluate the proposed IVLSTM-MCMR. The performance difference of the three variants of IVLSTM is first studied. Second, an ablation experiment is carried out on the MC data input and MR data output method. The validity and necessity of the MC model are verified, and the necessity of multiple-route integration of the final prediction result is verified. Third, the performance of IVLSTM-MCMR is evaluated by comparing the experimental results with those of the established models. Finally, the overall results and prediction comparison charts are given.

4.3.1. Performance evaluation of three variants of IVLSTM

In this section, the GRU (Cho et al., 2014) structure, VLSTM structure, IVLSTM-A, IVLSTM-B, and IVLSTM-C are used to predict air quality. To ensure the efficiency of this group of experiments, all stations in this section use the same input channel, and no output route selection step is performed. The experimental results are listed in Table 1. Table 1 shows that the prediction effect of VLSTM is better than that of GRU. This finding proves that for the air quality dataset, a structure with excessive reduction of parameters has a disadvantage in prediction accuracy since much information is lost due to the simple structure of GRU. The aforementioned finding also demonstrates the rationality of the further improvement based on the VLSTM structure used in this paper as well. The results of IVLSTM are better than those of the VLSTM structure, thus indicating improvements by the proposed method. The result of IVLSTM-A shows that the network performance is improved due to reducing the number of parameters by saving the parameters of the forget gate. The results of IVLSTM-B and IVLSTM-C indicate that the improvement in the sensitivity of the output result to the current cell state c^t of the LSTM is successful. The prediction results of IVLSTM-C are better than those of IVLSTM-B, thus indicating that using historical information of cell state $ct-1$ reduces the impact of abnormal cell states during propagation, thereby making the convergence of the deep learning network training more stable. The $std.$ of each index shows that the final IVLSTM-C is much more robust than the original VLSTM structure, and GRU structure. As IVLSTM-C is the most effective structure, we use IVLSTM-C as the final structure in IVLSTM and IVLSTM-MCMR.



(a) MAE of Different Improved Models with MCMR (b) Accuracy of Different Improved Models with MCMR

Fig. 6. Ablation experiment of MCMR module.

4.3.2. Performance evaluation of MCMR

In this section, the effectiveness of the MCMR module is studied, and experimental results are shown in Fig. 6. In Fig. 6, L represents the model using only the data of the target station itself without the data of related stations. R(DTW) represents the model with data from time-related stations by using DTW as the time-related station selection method. R(LS-DTW) represents the model with the same type data as R(DTW) but uses LS-DTW as the time-related station selection method. MC means that the model uses all results of five channels to integrate the final result without route selection. MCMR is the final model adding the route selection model to MC. We find that Model L, which uses only its own information, has the worst prediction effect. The effect of Model R(LS-DTW) is better than that of R(DTW), thus verifying the effectiveness of the Model LS-DTW selection algorithm. The comparison between MC and other models except MCMR shows the effectiveness and necessity of the multichannel input model. Compared with the prediction results of Models L, R(DTW), R(LS-DTW), and MC, the prediction result of MCMR has been greatly improved, thus showing that the route selection method is effective.

4.3.3. Comparison results with established models

In this section, we compare the proposed IVLSTM-MCMR with the traditional algorithms of LR, RT, and ANN. Additionally, we compare IVLSTM-MCMR with state-of-the art FFA (Zheng et al., 2015) and STE (Wang & Song, 2018) models. GRU is also taken in the comparison. The prediction results and accuracy are obtained under the prediction time intervals of 6 h, 9 h, 12 h and 24 h. As shown in Fig. 7, the traditional methods of LR and RT are far from the designed IVLSTM-MCMR in terms of prediction accuracy and prediction error. The more representative models FFA and STE both outperform the traditional methods in air quality prediction. Compared with the STE model, IVLSTM-MCMR improved the prediction results in terms of the MAE at time intervals of 6 h, 9 h, 12 h, and 24 h by 7.2%, 8.4%, 3.5%, and 12.8%, respectively. In terms of accuracy, except for the time interval of 12 h, when the prediction results of IVLSTM-MCMR are almost the same as those of STE, the prediction results of IVLSTM-MCMR are the best, and the effect is particularly improved in the time interval of 24 h. As GRU is considered, it has similar performance with STE in terms of MEA and accuracy.

4.3.4. Overall results

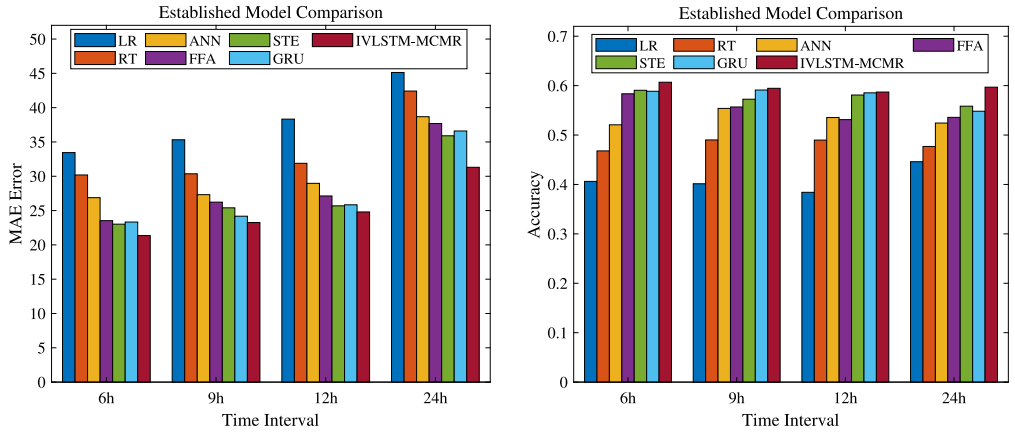
Table 2 shows the MAE, RMSE and accuracy of the final results separately in time intervals of 6 h, 12 h, and 24 h. Table 2 shows that compared with the other models, IVLSTM-MCMR achieves almost the best results on all the tests. The largest and smallest improvements by IVLSTM-MCMR are with time intervals of 24 h and 12 h, respectively. From the perspective of time stage, 24 h is the length of a day, and the regularity is stronger. Compared with 24 h, human social activities are basically in the opposite state at the time interval of 12 h, so it is more difficult to grasp the regularity and improve the prediction effect. Fig. 8 shows the comparison charts. The blue line represents the ground truth from real-world monitor stations, and the red line represents the prediction results from IVLSTM-MCMR. Fig. 8(a) shows the predictive effect with high-scale fluctuation while Fig. 8(b) displays the predictive effect with low-scale fluctuation. In both situations, the proposed IVLSTM-MCMR can respond quickly to changes in time series, thus showing the good performance of IVLSTM-MCMR in terms of precision and efficiency. In this experiment, three different time intervals have been studied. It should be noted that the prediction results at different time intervals correspond to different models, and the training sets used by these models in the training process are the training data collected at different time intervals.

Table 2
Results of all methods.

Model	MAE			RMSE			Accuracy		
	6 h	12 h	24 h	6 h	12 h	24 h	6 h	12 h	24 h
LR	33.45	38.33	45.13	42.02	49.31	57.23	0.4062	0.3841	0.4460
RT	30.19	31.89	42.42	41.09	44.46	58.38	0.4679	0.4898	0.4769
ANN	26.88	28.97	38.68	35.64	39.20	51.08	0.5207	0.5354	0.5243
FFA	23.53	27.13	37.69	32.57	38.00	48.44	0.5835	0.5312	0.5358
L	25.29	28.31	36.94	28.68	32.70	35.78	0.5437	0.5461	0.5437
R(DTW)	24.18	30.33	42.30	33.93	43.95	54.93	0.5934	0.5785	0.5439
R(LS-DTW)	23.44	26.06	32.82	28.16	31.15	33.75	0.5863	0.5836	0.5896
MC	22.82	25.87	32.54	26.73	**30.27**	33.17	0.5973	0.5857	0.5932
STE	23.02	25.69	35.90	31.14	35.65	49.92	0.5906	**0.5878**	0.5584
GRU	23.33	25.84	36.59	31.93	36.55	52.13	0.5886	0.5855	0.5482
IVLSTM-MCMR	**21.36**	**24.80**	**31.31**	**26.31**	30.58	**33.04**	**0.6069**	0.5870	**0.5969**

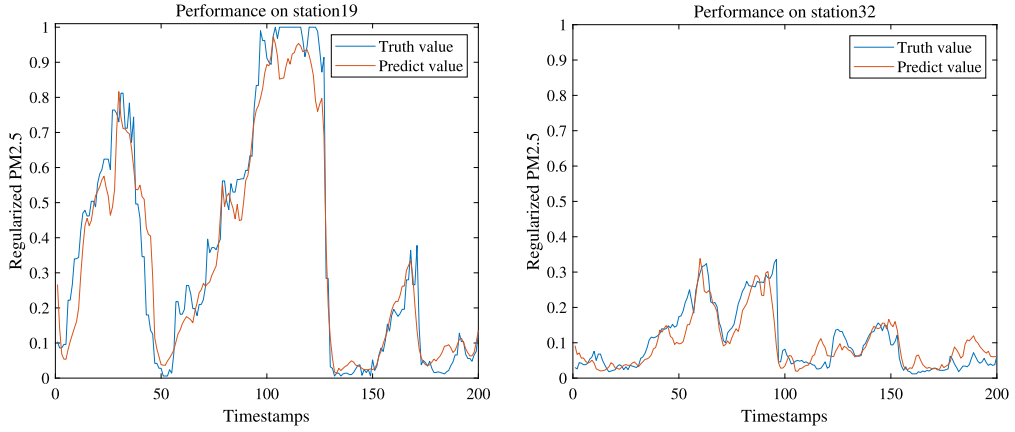
4.4. Statistical comparison

Following the similar idea in Gu, Qiao, and Lin (2018), we carry out the statistical comparison to further evaluate the effectiveness of the proposed algorithm. Since the LR, RT, and FFA are the exact algorithms without standard deviations, they are not considered in



(a) MAE of established models with IVLSTM-MCMR (b) Accuracy of established models with IVLSTM-MCMR

Fig. 7. MCMR results compared with established models.



(a) Station 19 is a station with high-scale fluctuation (b) Station 32 is a station with low-scale fluctuation

Fig. 8. Final prediction of Station 19 and Station 32.

Table 3
Total rank results of four algorithms on three prediction time intervals.

Algorithms	Accuracy			Total rank
	6 h	12 h	24 h	
ANN	4	4	4	12
STE	2	3	2	7
GRU	3	2	3	8
IVLSTM-MCMR	1	1	1	3

statistical comparison. For L, R(DTW), R(LS-DTW), and MC, they are variants of IVLSTM-MCMR with different kinds of input data and the performance of them have been studied in Section 4.3.2, they are also not in the comparison here. Therefore, we take ANN, STE, GRU, and IVLSTM-MCMR into statistical comparison by analysing the average best accuracy values shown in Table 2 based on a multiple comparison procedure ANOVA with 0.05 as the level of significance (Sthle & Wold, 1989; Sun, Fang, Wu, Palade, & Xu, 2012). Table 3 shows the resulting rank of four algorithms on each prediction time interval and the total rank. From the rank results, it can be seen that IVLSTM-MCMR has the best performance among the compared algorithms and STE runs the second.

5. Conclusion

In this paper, we introduced an air quality prediction model based on the improved *Vanilla* VLSTM with multiChannel input and multi-Route output, which is called IVLSTM-MCMR. In the proposed model, the number of parameters in the IVLSTM module is reduced to accelerate training. Additionally, the historical information is utilized in a better way in IVLSTM so that the developed model would have more stable training. Three different inner structures of IVLSTM are designed and studied by analysing the forward propagation formulations of VLSTM. The MCMR module is designed to select the input data from multiple channels and integrate the output data from multiple routes more efficiently. The LS-DTW algorithm is assigned in the MCMR module to select the spatial–temporal similar monitoring station more reasonably. Experimental results demonstrate the effectiveness of the proposed IVLSTM-MCMR model for air quality prediction when compared with both traditional algorithms and state-of-the-art algorithms. Future research will design more effective inner structures for VLSTM and consider more air pollution sources in the multichannel input module to improve prediction accuracy.

CRediT authorship contribution statement

Wei Fang: Supervision. **Runsu Zhu:** Methodology, Writing. **Jerry Chun-Wei Lin:** Resources, Language.

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.

Data availability

Data will be made available on request.

Acknowledgements

This work was supported in part by the National Natural Science foundation of China under Grant 62073155, 62002137, 62106088, in part by "Blue Project" in Jiangsu Universities, China, in part by Guangdong Provincial Key Laboratory under Grant 2020B121201001.

References

Allen-Zhu, Z., Li, Y., & Song, Z. (2019). A convergence theory for deep learning via over-parameterization. In *International Conference on Machine Learning* (pp. 242–252). PMLR.

Ang, J.-S., Ng, K.-W., & Chua, F.-F. (2020). Modeling time series data with deep learning: A review, analysis, evaluation and future trend. In *2020 8th international conference on information technology and multimedia (ICIMU)* (pp. 32–37). http://dx.doi.org/10.1109/ICIMU49871.2020.9243546.

Burrows, W. R., Benjamin, M., Beauchamp, S., Lord, E. R., McCollor, D., & Thomson, B. (1995). CART decision-tree statistical analysis and prediction of summer season maximum surface ozone for the vancouver, montreal, and atlantic regions of Canada. *Journal of Applied Meteorology*, *34*(8), 1848–1862.

Chen, Y., Wang, L., Li, F., Du, B., Choo, K.-K. R., Hassan, H., et al. (2017). Air quality data clustering using EPLS method. *Information Fusion*, *36*, 225–232.

Cheng, C. S., Campbell, M., Li, Q., Li, G., Auld, H., Day, N., et al. (2007). A synoptic climatological approach to assess climatic impact on air quality in south-central Canada. Part II: future estimates. *Water, Air, and Soil Pollution*, *182*(1–4), 117–130.

Chiu, H. F., & Yang, C. Y. (2015). Air pollution and daily clinic visits for migraine in a subtropical city: Taipei, Taiwan. *Journal of Toxicology and Environmental Health, Part A*, *78*(9), 549–558.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., et al. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.

Conti, G. O., Heibati, B., Kloog, I., Fiore, M., & Ferrante, M. (2017). A review of AirQ Models and their applications for forecasting the air pollution health outcomes. *Environmental Science and Pollution Research*, *24*(7), 6426–6445.

D'Amato, G., Vitale, C., Lanza, M., Molino, A., & D'Amato, M. (2016). Climate change, air pollution, and allergic respiratory diseases: an update. *Current Opinion in Allergy and Clinical Immunology*, *16*(5), 434–440.

Donnelly, A., Misstear, B., & Broderick, B. (2015). Real time air quality forecasting using integrated parametric and non-parametric regression techniques. *Atmospheric Enviroment*, *103*, 53–65.

Du, S., Li, T., Yang, Y., & Horng, S.-J. (2021). Deep air quality forecasting using hybrid deep learning framework. *IEEE Transactions on Knowledge and Data Engineering*, *33*(6), 2412–2424. http://dx.doi.org/10.1109/TKDE.2019.2954510.

Ergen, T., & Kozat, S. S. (2018). Online training of LSTM networks in distributed systems for variable length data sequences. *IEEE Transactions on Neural Networks and Learning Systems*, *29*(10), 5159–5165. http://dx.doi.org/10.1109/TNNLS.2017.2770179.

Fan, J., Li, Q., Hou, J., Feng, X., Karimian, H., & Lin, S. (2017). A spatiotemporal prediction framework for air pollution based on deep RNN. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *4*, 15.

Freeman, B. S., Taylor, G., Gharabaghi, B., & Thé, J. (2018). Forecasting air quality time series using deep learning. *Journal of the Air & Waste Management Association*, *68*(8), 866–886.

Gers, F. A., & Schmidhuber, J. (2000). Recurrent nets that time and count. In *Proceedings of the international joint conference on neural networks, Vol. 3* (pp. 189–194).

Gers, F. A., Schmidhuber, J., & Cummins, F. (2000). Learning to forget: Continual prediction with LSTM. *Neural Computation*, *12*(10), 2451–2471.

Gers, F. A., Schraudolph, N. N., & Schmidhuber, J. (2002). Learning precise timing with LSTM recurrent networks. *Journal of Machine Learning Research*, *3*(Aug), 115–143.

Graves, A., & Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, *18*(5–6), 602–610.

Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., & Schmidhuber, J. (2017). LSTM: A search space odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, *28*(10), 2222–2232. http://dx.doi.org/10.1109/TNNLS.2016.2582924.

Gu, K., Qiao, J., & Lin, W. (2018). Recurrent air quality predictor based on meteorology- and pollution-related factors. *IEEE Transactions on Industrial Informatics*, *14*(9), 3946–3955.

Guizilini, V. C., & Ramos, F. T. (2015). A nonparametric online model for air quality prediction. In *Twenty-ninth AAAI conference on artificial intelligence*.

Han, Y., Lam, J. C., Li, V. O., & Zhang, Q. (2020). A domain-specific Bayesian deep-learning approach for air pollution forecast. *IEEE Transactions on Big Data*, 1. http://dx.doi.org/10.1109/TBDATA.2020.3005368.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780.

Ip, W. F., Vong, C. M., Yang, J., & Wong, P. K. (2010). Least squares support vector prediction for daily atmospheric pollutant level. In *IEEE/ACIS international conference on computer and information science* (pp. 23–28).

Kanjo, E., Younis, E. M., & Ang, C. S. (2019). Deep learning analysis of mobile physiological, environmental and location sensor data for emotion detection. *Information Fusion*, *49*, 46–56.

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 Journal of Chemical Engineering*, *27*(6), 1675–1680.

Kim, M., Kim, Y., Sung, S., & Yoo, C. (2009). Data-driven prediction model of indoor air quality by the preprocessed recurrent neural networks. In *International conference on control, automation, and systems* (pp. 1688–1692).

Li, X., Peng, L., Yao, X., Cui, S., Hu, Y., You, C., et al. (2017). Long short-term memory neural network for air pollutant concentration predictions: Method development and evaluation. *Environmental Pollution*, *231*, 997–1004.

Liu, J., Li, T., Xie, P., Du, S., Teng, F., & Yang, X. (2020). Urban big data fusion based on deep learning: An overview. *Information Fusion*, *53*, 123–133.

Liu, J., Wang, Z., & Xu, M. (2020). DeepMTT: A deep learning maneuvering target-tracking algorithm based on bidirectional LSTM network. *Information Fusion*, *53*, 289–304.

McCollister, G. M., & Wilson, K. R. (1975). Linear stochastic models for forecasting daily maxima and hourly concentrations of air pollutants. *Atmospheric Enviroment*, *9*(4), 417–423.

McKeen, S., Grell, G., Peckham, S., Wilczak, J., Djalalova, I., Hsie, E. Y., et al. (2009). An evaluation of real-time air quality forecasts and their urban emissions over eastern texas during the summer of 2006 Second Texas Air Quality Study field study. *Journal of Geophysical Research: Atmospheres*, *114*(D7).

McKendry, I. G. (2002). Evaluation of artificial neural networks for fine particulate pollution (PM10 and PM2. 5) forecasting. *Journal of the Air & Waste Management Association*, *52*(9), 1096–1101.

Navares, R., & Aznarte, J. L. (2020). Predicting air quality with deep learning LSTM: Towards comprehensive models. *Ecological Informatics*, *55*, Article 101019.

Nguyen-Tuong, D., Peters, J. R., & Seeger, M. (2009). Local gaussian process regression for real time online model learning. In *Advances in neural information processing systems* (pp. 1193–1200).

Pfender, W., Graw, R., Bradley, W., Carney, M., & Maxwell, L. (2006). Use of a complex air pollution model to estimate dispersal and deposition of grass stem rust urediniospores at landscape scale. *Agricultural and Forest Meteorology*, *139*(1–2), 138–153.

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 Transactions on Knowledge and Data Engineering*, *30*(12), 2285–2297. http://dx.doi.org/10.1109/TKDE.2018.2823740.

Soh, P. W., Chang, J. W., & Huang, J. W. (2018). Adaptive deep learning-based air quality prediction model using the most relevant spatial-temporal relations. *IEEE Access*, *6*, 38186–38199.

Soh, P. W., Chen, K. H., Huang, J. W., & Chu, H. J. (2017). Spatial-Temporal pattern analysis and prediction of air quality in Taiwan. In *10th international conference on ubi-media computing and workshops (Ubi-Media)* (pp. 1–6).

Sthle, L., & Wold, S. (1989). Analysis of variance (ANOVA). *Chemometrics and Intelligent Laboratory Systems*, *6*(4), 259–272. http://dx.doi.org/10.1016/0169-7439(89)80095-4, URL: https://www.sciencedirect.com/science/article/pii/0169743989800954.

Sun, J., Fang, W., Wu, X., Palade, V., & Xu, W. (2012). Quantum-behaved particle swarm optimization: analysis of individual particle behavior and parameter selection. *Evolutionary Computation*, *20*(3), 349–393.

Torres, J. F., Hadjout, D., Sebaa, A., Martínez-Álvarez, F., & Troncoso, A. (2021). Deep learning for time series forecasting: a survey. *Big Data*, *9*(1), 3–21.

Vardoulakis, S., Fisher, B. E., Pericleous, K., & Gonzalez-Flesca, N. (2003). Modelling air quality in street canyons: a review. *Atmospheric Enviroment*, *37*(2), 155–182.

Wang, J., & Song, G. (2018). A deep spatial-temporal ensemble model for air quality prediction. *Neurocomputing*, *314*, 198–206.

Wang, J., Zhang, X., Guo, Z., & Lu, H. (2017). Developing an early-warning system for air quality prediction and assessment of cities in China. *Expert Systems with Applications*, *84*, 102–116.

Wayland, R., White, J., Dickerson, P., & Dye, T. (2002). Communicating real-time and forecasted air quality to the public. *Environmental Management*, 28–36.

Wen, C., Liu, S., Yao, X., Peng, L., Li, X., Hu, Y., et al. (2019). A novel spatiotemporal convolutional long short-term neural network for air pollution prediction. *Science of the Total Environment*, *654*, 1091–1099.

Wu, Q., & Lin, H. (2019). A novel optimal-hybrid model for daily air quality index prediction considering air pollutant factors. *Science of the Total Environment*, *683*, 808–821.

Yan, R., Liao, J., Yang, J., Sun, W., Nong, M., & Li, F. (2021). Multi-hour and multi-site air quality index forecasting in Beijing using CNN, LSTM, CNN-LSTM, and spatiotemporal clustering. *Expert Systems with Applications*, *169*, Article 114513.

Yi, X., Duan, Z., Li, R., Zhang, J., Li, T., & Zheng, Y. (2020). Predicting fine-grained air quality based on deep neural networks. *IEEE Transactions on Big Data*, 1. http://dx.doi.org/10.1109/TBDATA.2020.3047078.

Yu, R., Yang, Y., Yang, L., Han, G., & Move, O. (2016). RAQ–a random forest approach for predicting air quality in urban sensing systems. *Sensors*, *16*(1), 86.

Zhang, Y., Bocquet, M., Mallet, V., Seigneur, C., & Baklanov, A. (2012). Real-time air quality forecasting, part I: History, techniques, and current status. *Atmospheric Enviroment*, *60*, 632–655.

Zhao, H., Zhang, J., Wang, K., Bai, Z., & Liu, A. (2010). A GA-ANN model for air quality predicting. In *2010 international computer symposium* (pp. 693–699).

Zheng, Y., Yi, X., Li, M., Li, R., Shan, Z., Chang, E., et al. (2015). Forecasting fine-grained air quality based on big data. In *Proceedings of the ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 2267–2276).