

Article

# Urban PM<sub>2.5</sub> Concentration Prediction via Attention-Based CNN–LSTM

Songzhou Li <sup>1</sup>, Gang Xie <sup>1,2,3,\*</sup>, Jinchang Ren <sup>1,4,\*</sup>, Lei Guo <sup>5</sup>, Yunyun Yang <sup>1</sup> and Xinying Xu <sup>1</sup>

<sup>1</sup> College of Electrical and Power Engineering, Taiyuan University of Technology, Taiyuan 030024, China; lisongzhou0283@link.tyut.edu.cn (S.L.); yangyunyun@tyut.edu.cn (Y.Y.); xuxinying@tyut.edu.cn (X.X.)

<sup>2</sup> Shanxi Key Laboratory of Advanced Control and Intelligent Information System, Taiyuan 030024, China

<sup>3</sup> School of Electronic Information Engineering, Taiyuan University of Science and Technology, Taiyuan 030024, China

<sup>4</sup> Department of Electronic and Electrical Engineering, University of Strathclyde, Glasgow G11XW, UK

<sup>5</sup> College of Information and Computer, Taiyuan University of Technology, Taiyuan 030024, China; guolei0036@link.tyut.edu.cn

\* Correspondence: xiegang@tyut.edu.cn (G.X.); jinchang.ren@strath.ac.uk (J.R.)

Received: 15 January 2020; Accepted: 7 March 2020; Published: 12 March 2020



**Abstract:** Urban particulate matter forecasting is regarded as an essential issue for early warning and control management of air pollution, especially fine particulate matter (PM<sub>2.5</sub>). However, existing methods for PM<sub>2.5</sub> concentration prediction neglect the effects of featured states at different times in the past on future PM<sub>2.5</sub> concentration, and most fail to effectively simulate the temporal and spatial dependencies of PM<sub>2.5</sub> concentration at the same time. With this consideration, we propose a deep learning-based method, AC-LSTM, which comprises a one-dimensional convolutional neural network (CNN), long short-term memory (LSTM) network, and attention-based network, for urban PM<sub>2.5</sub> concentration prediction. Instead of only using air pollutant concentrations, we also add meteorological data and the PM<sub>2.5</sub> concentrations of adjacent air quality monitoring stations as the input to our AC-LSTM. Hence, the spatiotemporal correlation and interdependence of multivariate air quality-related time-series data are learned by the CNN–LSTM network in AC-LSTM. The attention mechanism is applied to capture the importance degrees of the effects of featured states at different times in the past on future PM<sub>2.5</sub> concentration. The attention-based layer can automatically weigh the past feature states to improve prediction accuracy. In addition, we predict the PM<sub>2.5</sub> concentrations over the next 24 h by using air quality data in Taiyuan city, China, and compare it with six baseline methods. To compare the overall performance of each method, the mean absolute error (MAE), root-mean-square error (RMSE), and coefficient of determination ( $R^2$ ) are applied to the experiments in this paper. The experimental results indicate that our method is capable of dealing with PM<sub>2.5</sub> concentration prediction with the highest performance.

**Keywords:** PM<sub>2.5</sub> concentration prediction; deep learning; AC-LSTM network; attention mechanism

## 1. Introduction

Air pollution is a serious environmental problem that is attracting increasing attention worldwide [1]. With the rapid development of the Chinese economy and the acceleration of industrialization, urban air pollution is getting worse. As one of the main pollutants in the air, fine particulate matter (PM<sub>2.5</sub>) contains a large amount of toxic and harmful substances due to its small particle size. It not only stays in the atmosphere for a long time, but also has a long transport distance, resulting in a decrease in air visibility, seriously affecting our living environment and physical health. In response to it, the Chinese government established air quality monitoring stations in most cities,

to detect PM<sub>2.5</sub> and other air pollutant concentrations in real time [2]. However, it is inevitable for the government to bear a significant financial burden because of expensive equipment [3,4]. In addition to monitoring, there is a rising demand for the prediction of future air quality. Obviously, the prediction of real-time and future PM<sub>2.5</sub> concentration is essential for air pollution control and the prevention of health issues caused by air pollution.

With the development of machine learning in recent years, artificial neural network (ANN), support vector regression (SVR), and other methods were successfully applied to the prediction of air pollutant concentration. Zheng et al. [5] used the spatial features of roads, factories, and parks in the prediction area to predict the concentration of PM<sub>10</sub> and NO<sub>2</sub>. Li et al. [6] used SVR to predict the PM<sub>2.5</sub> concentration of a target station using observation data from the surrounding monitoring stations. Although all these aforementioned methods made use of the spatial features that affect the concentrations of pollutants, the temporal correlation of air pollutants and the time-delay characteristics of PM<sub>2.5</sub> were not considered.

Due to the dynamic nature of relevant atmospheric environments, the recurrent neural network (RNN) is especially suitable to simulate the temporal evolution of air pollutant distributions because RNNs can handle arbitrary sequences of inputs, thereby guaranteeing the capacity to learn temporal sequences [7]. Ong et al. [8] used meteorological data to predict PM<sub>2.5</sub> concentration using an RNN. Feng et al. [9] combined random forest (RF) and an RNN to analyze and forecast the next 24-h PM<sub>2.5</sub> concentration of air pollutants in Hangzhou, China. When there is a long time lag in the traditional RNN, however, it may suffer from problems such as gradient disappearance and gradient explosion [10]. These RNN-based methods do not take full advantage of spatial features either. Additionally, the states of the feature formation at different times will also have different effects on future PM concentrations. The existing studies did not consider the effects of feature states of the past different times on air pollutants, but only extracted the temporal correlation features of historical data.

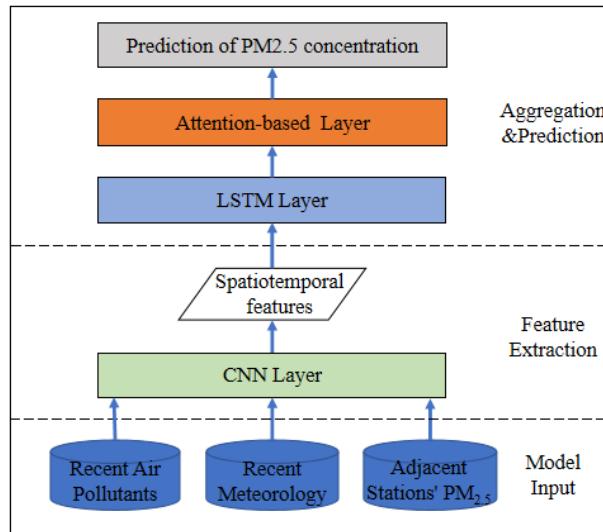
To tackle the aforementioned problems, we propose an attention-based convolutional neural network (CNN)-long short-term memory (LSTM) model, AC-LSTM, for predicting the PM<sub>2.5</sub> concentrations over the next 24 h. The proposed AC-LSTM model comprises a one-dimensional convolutional neural network (CNN), long short-term memory (LSTM) network [10], and attention-based network. As a representative network of RNN, the LSTM network overcomes the defect of gradient disappearance and gradient explosion of the traditional RNN due to its special cell structure [10]. It can capture the spatiotemporal correlation and interdependence of air quality-related time-series data at the same time. The joint one-dimensional CNN aims to extract spatiotemporal features from air quality data and local spatial correlation features of PM<sub>2.5</sub> concentrations among air monitoring stations. The attention mechanism is an effective mechanism to obtain superior results, as demonstrated in image recognition [11], machine translation [12] and sentence summarization [13]. Therefore, the attention mechanism [12] was applied in the AC-LSTM model, used to capture the importance degrees of effects of past feature states at different times on PM<sub>2.5</sub> concentration in this paper.

The major contributions of this paper are as follows: (1) by analyzing the spatiotemporal correlation of air quality data, we propose a novel deep learning method that can capture the spatiotemporal dependency of air pollutant concentration, to predict PM<sub>2.5</sub> concentrations in the next 24 h; (2) according to the importance degrees of effects of past feature states on PM<sub>2.5</sub> concentration, the attention-based layer weighs the past featured states in our predictive model to improve prediction accuracy; (3) comparing the performances of six popular machine learning methods in the air pollution prediction problem, we validate the practicality and feasibility of the proposed model in PM<sub>2.5</sub> concentration prediction.

## 2. Overview of the AC-LSTM Framework

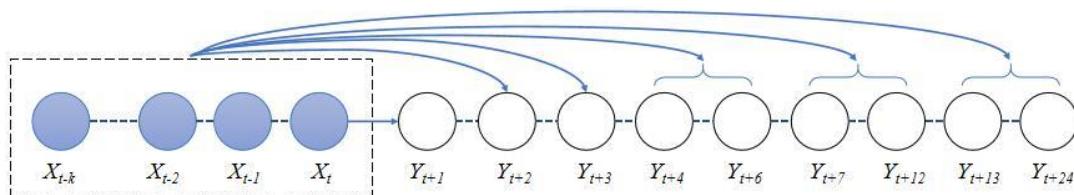
As shown in Figure 1, the framework of our approach consists of three major parts: model input, feature extraction, and aggregation and prediction. Since PM<sub>2.5</sub> concentration is extremely affected by spatiotemporal features, recent air pollutant concentration, meteorological data, and

the PM<sub>2.5</sub> concentrations of all adjacent stations are stacked to construct an input tensor for the one-dimensional CNN layer. In this way, the spatiotemporal features are extracted by the CNN layer. Then, the spatiotemporal correlation is learned by the LSTM layer. Because of the different effects of past states of different times on the PM<sub>2.5</sub> concentration, the attention-based layer can weigh the feature states at past different hours. Finally, the aggregation and prediction of the proposed model is achieved.



**Figure 1.** Framework of the proposed approach.

How the model predicts the PM<sub>2.5</sub> concentrations of the next 24 h is described in Figure 2. As shown in Figure 2,  $X_t$  represents the input data of the model at time  $t$  (e.g., air quality data, meteorological data in Figure 1),  $Y_{t+1}$  represents the predicted value of the PM<sub>2.5</sub> concentration at time  $t + 1$ , and  $k$  represents the time lag. We group the air quality data within a particular time lag to formulate different inputs (shown in the broken rectangle) for multiscale predictors, which are used to train separate models corresponding to different time intervals. The time lag of the model input indicates how many hours the input data are in the past. Each blue arrow shown in Figure 2 represents a different predictor. Afterward, a separate model is trained for each hour over the next 3 h. With respect to the next 7–24 h, it is divided into three time intervals, i.e., 4–6, 7–12, and 13–24 h, where separate models are trained to predict the mean PM<sub>2.5</sub> concentration during each time interval.



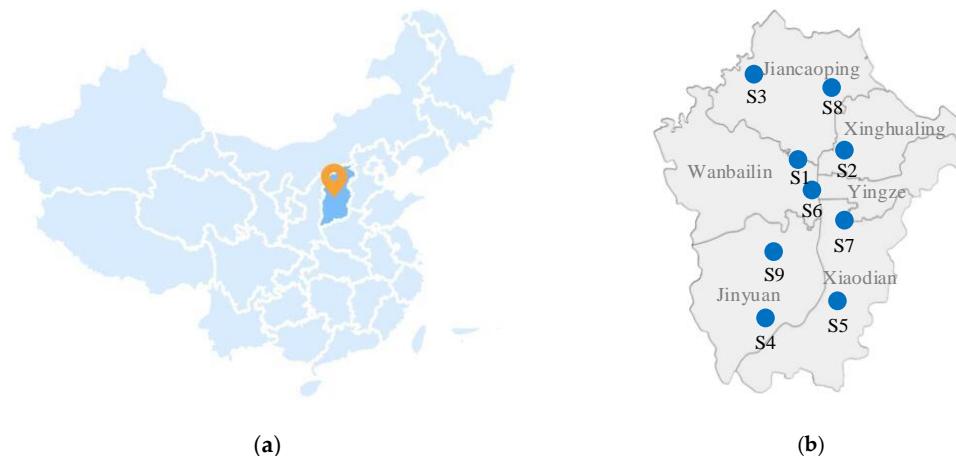
**Figure 2.** Illustration of the multiscale predictors.

### 3. Data and Method

#### 3.1. Data Description

The spatiotemporal variation of atmospheric particulate matter is affected by various factors such as pollution emission sources and meteorological conditions [14,15]. The PM<sub>2.5</sub> concentration is not only related to the atmospheric state and PM<sub>2.5</sub> concentration at the previous time, but also the PM<sub>2.5</sub> concentration in the adjacent areas [16,17]. The air quality data used for the AC-LSTM model input consists of readings of pollutant concentrations from air quality monitoring stations and meteorological data.

In this paper, the air quality data from nine air quality monitoring stations in Taiyuan City, China, were obtained from the National Environmental Protection Bureau and the Shanxi Provincial Environmental Protection Department. The location map of Taiyuan is shown in Figure 3a, and the yellow coordinate represents Taiyuan. The experimental data were collected from 1 January 2014 to 25 December 2016 at an hourly rate, and the spatial locations of the air quality monitoring stations are illustrated in Figure 3b. The experimental data contain the concentrations of PM<sub>10</sub>, PM<sub>2.5</sub>, SO<sub>2</sub>, NO<sub>2</sub>, O<sub>3</sub>, and CO. The detailed characteristics of Taiyuan air quality monitoring stations are shown in Table 1. The meteorological data include indicators such as air pressure, temperature, wind speed, humidity, and visibility.



**Figure 3.** (a) The location map of Taiyuan City; (b) distribution of air quality monitoring stations in Taiyuan City.

**Table 1.** Characteristics of Taiyuan air quality monitoring stations. N—north; E—east.

Stations	Code	Monitoring Environment	Coordinates
JianCaoPing	S1	Urban: residential area	N 37.887, E 112.522
JianHe	S2	Urban: residential area	N 37.910, E 112.573
ShangLan	S3	Rural area	N 38.010, E 112.434
JinYuan	S4	Suburban: residential area	N 37.712, E 112.469
XiaoDian	S5	Urban: residential area	N 37.739, E 112.558
TaoYuan	S6	Urban: residential area	N 37.869, E 112.536
WuCheng	S7	Urban: commercial area	N 37.819, E 112.570
NanZhai	S8	Suburban: industrial park	N 37.985, E 112.549
JinSheng	S9	Suburban: industrial area	N 37.780, E 112.488

### 3.2. Data Preprocessing

The collected data were preprocessed so as to improve the data quality and carry out data mining. Air quality monitoring equipment and meteorological monitoring equipment will cause leakage in data collection due to machine failure, regular inspection and maintenance, unstable transmission, bad weather, and other uncontrollable factors. The existence of such missing values will have some impact on data mining. The missing values are normally required to be removed or filled to ensure the performance of modeling [18]. On the one hand, when there are several missing values in a data record, we directly remove them; on the other hand, a linear interpolation [19] is implemented to fill empty values when there is only one missing value in a data record. After that, features in the data that are described by text, such as weather (sunny, cloudy, foggy, snowy, rainy, etc.) and wind direction (north, west, east, south, northwest, northeast, etc.), are quantified.

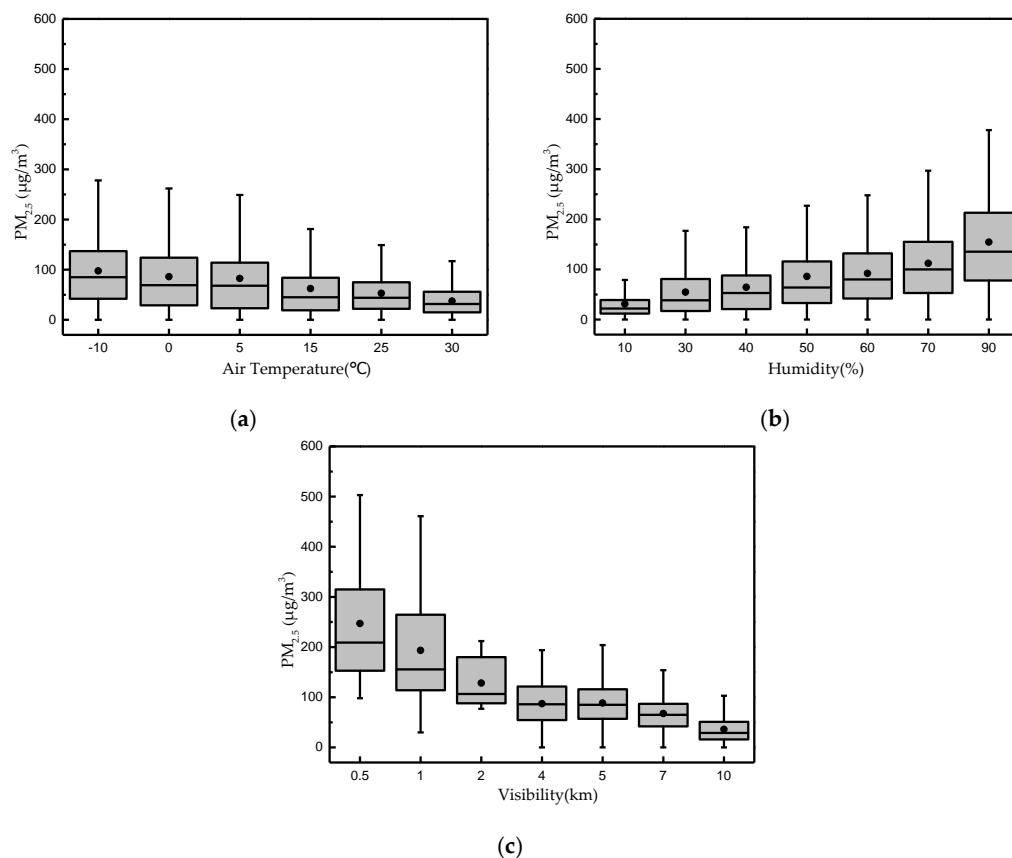
Furthermore, to accelerate the convergence of the model and reduce the impact of outliers, the features in the data records are normalized as follows:

$$f^* = \frac{f - f_{\min}}{f_{\max} - f_{\min}}, \quad (1)$$

where  $f_{\min}$  represents the minimum value, and  $f_{\max}$  represents the maximum value.

### 3.3. Correlation between Meteorological Features and PM<sub>2.5</sub>

PM<sub>2.5</sub> concentrations are correlated with meteorological features, as shown in the Figure 4. In Figure 4, the black dots in the box represent the average PM<sub>2.5</sub> concentration, and the horizontal line in the middle of the box represents the median. When the air temperature near the ground is high, the atmospheric convection is strengthened, and the concentration of pollutants can be reduced. When the air temperature near the ground is low, the atmosphere tends to form an inversion layer, which is not conducive to the diffusion of pollutants. It can be seen from Figure 4a that the concentration of PM<sub>2.5</sub> decreases with the increase in temperature, indicating that the increase in temperature is conducive to the diffusion and dilution of PM<sub>2.5</sub>.



**Figure 4.** Box charts of meteorological features and fine particulate matter (PM<sub>2.5</sub>) concentration: (a) air temperature; (b) humidity; (c) visibility.

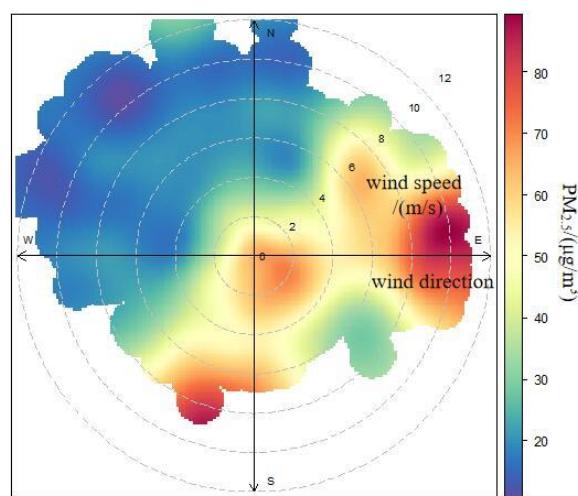
A higher relative humidity results in a weaker diffusion ability of PM<sub>2.5</sub>. Such an environment will cause an increase in the hygroscopicity of pollutants and accelerate the chemical transformation of pollutants, thus aggravating the degree of air pollution. As can be seen from Figure 4b, there is a correlation between PM<sub>2.5</sub> and relative humidity, that is, with the increase in relative humidity, PM<sub>2.5</sub> concentration will also increase.

The main reason for the loss of visibility is air pollution, with particulate matter having the greatest impact on visibility. Low visibility means heavy air pollution, while high visibility means light air

pollution. Figure 4c shows that a higher visibility denotes a lower concentration of PM<sub>2.5</sub>. There is a significant correlation between PM<sub>2.5</sub> concentration and visibility.

Weather is an important factor affecting air quality. Rain and snow wash particulate matter and other pollutants from the air, effectively purifying the air. However, fog, sandstorms, and haze will increase the pollution level and reduce air quality. In addition, air pressure affects the flow of air, which affects the migration of PM<sub>2.5</sub>.

Wind direction determines the direction of migration and horizontal diffusion of atmospheric pollutants. The relationship between wind direction, wind speed, and PM<sub>2.5</sub> is shown in Figure 5. Taiyuan is located in north-central China, with northerly and northwesterly winds prevailing. In the north and west of Taiyuan, there are a large number of factories, such as steel mills and power plants. When northerly wind prevails in Taiyuan, a large number of pollutants are transported from north to south, aggravating the urban pollution. According to Figure 5, when the wind direction is westerly or northerly, PM<sub>2.5</sub> concentration is relatively high. A higher wind level facilitates PM<sub>2.5</sub> migration, which leads to more pollution downwind.



**Figure 5.** Relationship between wind direction, wind speed, and PM<sub>2.5</sub>.

### 3.4. Spatiotemporal Correlation Analysis

Because of meteorological conditions, especially wind speed and wind direction, air pollutants are affected by the environment in the surrounding area. Similarly, particulate matter stays in the air for a long time and is more susceptible to the surrounding area. To analyze the spatial correlation of PM<sub>2.5</sub> concentrations, the Pearson correlation coefficient [20] is calculated among all monitoring stations, and the results are shown in Table 2. From the table, the correlation coefficients among most stations were greater than 0.7, except for the correlation coefficient at S3. This indicates that PM<sub>2.5</sub> concentrations are highly correlated among most stations. The reason for the small correlation coefficient between S3 and other stations is that it is far away from other stations and it is located in a rural area. Thus, the spatial correlation of PM<sub>2.5</sub> concentrations can be used to optimize the input of the model for improving the prediction performance. As shown in Figure 1, PM<sub>2.5</sub> concentrations of the adjacent stations are added to the model input. In the experiment, we used PM<sub>2.5</sub> concentrations of all stations as input because the number of air quality monitoring stations is too small in Taiyuan city.

The PM<sub>2.5</sub> concentration is highly correlated in the temporal domain, which is similarly affected by other features in the past. The autocorrelation functions [21] below were used to measure the temporal correlations among the PM<sub>2.5</sub> concentration time series at each station. The detailed formula of the autocorrelation function is as follows:

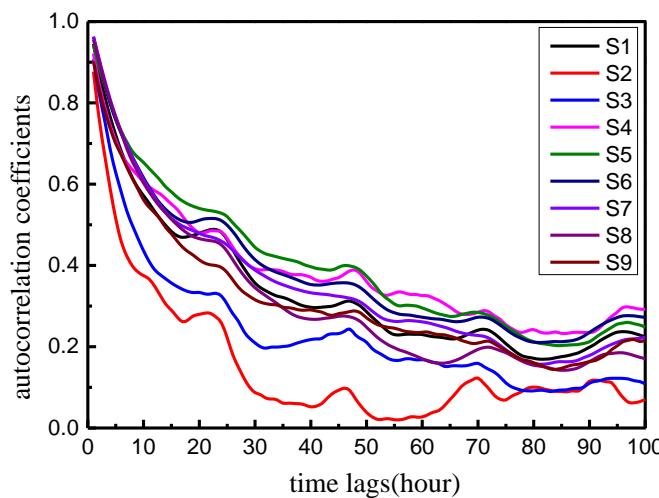
$$\rho_k = \frac{\text{Cov}(y(t), y(t+k))}{\sigma_{y(t)} \sigma_{y(t+k)}}, \quad (2)$$

where  $\rho_k$  represents the autocorrelation coefficient when the time lag is  $k$ ,  $y(t)$  represents the PM<sub>2.5</sub> concentration vector,  $y(t + k)$  represents the PM<sub>2.5</sub> concentration vector after  $k$  hours,  $Cov(y(t), y(t + k))$  is the covariance of  $y(t)$  and  $y(t + k)$ , and  $\sigma_{y(t)}$  and  $\sigma_{y(t+k)}$  represent the standard deviations of  $y(t)$  and  $y(t + k)$ , respectively.

**Table 2.** Correlation coefficients of PM<sub>2.5</sub> among all monitoring stations.

Station	S1	S2	S3	S4	S5	S6	S7	S8	S9
S1	1	0.95	0.2	0.69	0.8	0.95	0.86	0.95	0.72
S2	0.95	1	0.24	0.76	0.86	0.82	0.82	0.95	0.77
S3	0.2	0.24	1	0.29	0.57	0.08	0.53	0.37	0.4
S4	0.69	0.76	0.29	1	0.87	0.51	0.76	0.8	0.96
S5	0.8	0.86	0.57	0.87	1	0.62	0.93	0.93	0.9
S6	0.95	0.82	0.08	0.51	0.62	1	0.78	0.84	0.57
S7	0.86	0.82	0.53	0.76	0.93	0.78	1	0.94	0.85
S8	0.95	0.95	0.37	0.8	0.93	0.84	0.94	1	0.83
S9	0.72	0.77	0.4	0.96	0.9	0.57	0.85	0.83	1

The autocorrelation coefficients of all stations when the value of the time lag  $k$  is different are shown in Figure 6. As seen, in general, the autocorrelation coefficients of all stations declined. When the time lag was smaller, the autocorrelation coefficient was larger. This indicates that a PM<sub>2.5</sub> concentration closer to the current time has a stronger correlation with the PM<sub>2.5</sub> concentration at the current time. The PM<sub>2.5</sub> concentrations within a lag of 15 h are strongly correlated to each other in a period of one day. It is worth noting that, when the time lag was 24, 48, and 72 h, the autocorrelation coefficients of each station in Figure 6 showed a temporary rise. This is very likely due to the periodic living pattern across different days in the same geographical environment and season. As PM<sub>2.5</sub> tends to stay in the air for a long time, its concentrations in the past few hours will affect observed data in the future. In addition to PM<sub>2.5</sub> concentration, the past weather conditions, such as wind and rain/snow, also affect PM<sub>2.5</sub> concentration. According to the above analysis, the current state of observed PM<sub>2.5</sub> concentration is closely related to that in the past states. These findings can help us choose a suitable time lag for multiscale predictors.



**Figure 6.** Comparison of the autocorrelation coefficients at different time lags for different stations.

In general, PM<sub>2.5</sub> concentration is strongly influenced by the spatiotemporal correlations among monitoring stations and the past states of the prediction area. The AC-LSTM model proposed in this

paper can capture the spatiotemporal relations of the variation in PM<sub>2.5</sub> concentrations. The attention mechanism is introduced in the proposed method to weigh the past states, which helps to measure the importance of past states at different times, as modeled using the LSTM, for PM<sub>2.5</sub> concentrations.

### 3.5. Method

#### 3.5.1. Convolutional Neural Network

The CNN has excellent performance in image processing [22], and it can be effectively applied to time series analysis [2]. CNN's local perception and weight sharing features can reduce the number of parameters for processing multivariate time series, thereby improving the learning efficiency [2]. Spatiotemporal features can be easily extracted by the one-dimensional (1D) CNN (1D-CNN) from the model input. Let the given model input be  $X = [x_1, x_2, \dots, x_t]$ , consisting of meteorological data, pollutant concentrations, and PM<sub>2.5</sub> concentrations at adjacent stations in the past. Firstly, the model input  $X$  is input to the 1D-CNN layer; hence, we have

$$l_t = \tanh(x_t * k_t + b_l), \quad (3)$$

where  $x_t$  represents the input vector,  $k_t$  is the convolution kernel,  $b_l$  represents bias vector, and  $l_t$  is the output vector of the 1D-CNN layer. The output of the 1D-CNN layer is a spatiotemporal feature matrix  $L = [l_1, l_2, \dots, l_t]$ .

#### 3.5.2. LSTM Network

As a special kind of RNN, the LSTM network [10] is capable of learning long-term dependencies. It has the advantage of connecting previous information to the present task [23,24]. Because of its special memory cell architecture, the LSTM network overcomes the defects of the traditional RNN, especially the problems of gradient disappearance and gradient explosion. The architecture of an LSTM memory cell is shown in Figure 7, where each cell has three "gate" structures, namely, the input gate, the forget gate, and the output gate. A chain of repeating cells forms the LSTM layer. The calculation process of the spatiotemporal feature matrix  $L = [l_1, l_2, \dots, l_t]$  in the LSTM layer is given in Equations (4)–(9).

$$f_t = \sigma(W_f \cdot [h_{t-1}, l_t] + b_f), \quad (4)$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, l_t] + b_i), \quad (5)$$

$$\tilde{c}_t = \tanh(W_c \cdot [h_{t-1}, l_t] + b_c), \quad (6)$$

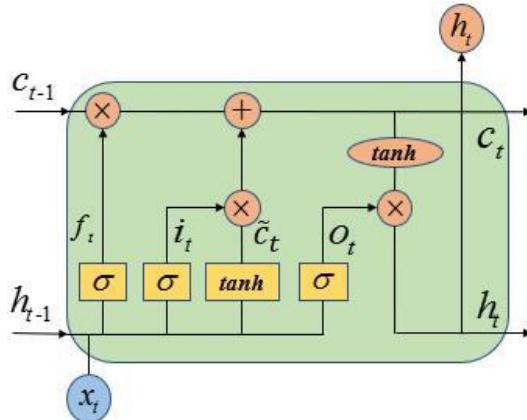
$$c_t = f_t \circ c_{t-1} + i_t \circ \tilde{c}_t, \quad (7)$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, l_t] + b_o), \quad (8)$$

$$h_t = o_t \circ \tanh(c_t), \quad (9)$$

where  $W_f$ ,  $W_i$ , and  $W_c$  denote the weight vector of the input gate, output gate, and forget gate, respectively, whereas  $b_f$ ,  $b_i$ ,  $b_c$ , and  $b_o$  are the bias vectors for the three gates, and  $\sigma$  denotes the sigmoid activation function.

Actually, Equation (4) represents the forget gate and it decides what information should be thrown away from the cell state, where  $f_t$  denotes the output of the forget gate. Equations (5) and (6) represent the input gate, which decides what new information should be stored in the cell state, where  $i_t$  and  $\tilde{c}_t$  denote the output of the input gate, and  $c_t$  denotes the activation vector of the current cell. Equations (8) and (9) represent the output gate, where  $o_t$  denotes the output of the output gate.  $h_{t-1}$  is the hidden state of the last cell, and  $h_t$  is the state of the current cell. The feature state matrix  $H = [h_1, h_2, \dots, h_t]$  is the output of the LSTM layer.



**Figure 7.** Architecture of long short-term memory (LSTM) memory cell.

### 3.5.3. Attention Layer

The attention mechanism [12] allows the model to capture the most important parts of the PM<sub>2.5</sub> concentration when different features of past states are considered. In order to take advantage of the information of the past states, an attention-based layer is added to the LSTM layer in the proposed AC-LSTM model. It ranks the importance degrees of different feature states in the past as follows, where  $H = [h_1, h_2, \dots, h_t]$  is the feature state matrix in the attention layer:

$$u_t = \tanh(W_h h_t + b_h), \quad (10)$$

$$\alpha_t = \frac{\exp(u_t^T v)}{\sum_t \exp(u_t^T v)}, \quad (11)$$

$$s = \sum_t \alpha_t h_t, \quad (12)$$

where  $u_t$  and  $v$  denote the projection vectors,  $\alpha_t$  is the normalized attention weight of  $h_t$ , and  $s$  is the weighted output vector of the attention layer.

According to the importance of each vector in the feature state matrix  $H$ , Equations (10) and (11) can calculate the normalized weight of each vector. Equation (12) gives the weighted vector  $s$ . This achieves the importance of measuring feature states at different times. Eventually, the weighted vector  $s$  passes through a layer of a fully connected network to obtain the PM<sub>2.5</sub> concentration of the prediction task.

## 4. Results and Discussion

The collected dataset is divided into two parts: the data of the first 28 months are used to train the model, and the data of the last 8 months are used to test the performance of the developed models when benchmarking with others. The mean absolute error (MAE), root-mean-square error (RMSE), and coefficient of determination ( $R^2$ ) are used as evaluation metrics to evaluate the performance of the different models in this paper.

### 4.1. Experimental Set-Up

This section describes the hardware and software environment of the experiment and the configuration of hyperparameters [2]. The code for all the prediction methods in this paper was written in Python. Our model and other deep learning comparison models were implemented through Keras, an open source deep learning library based on Tensorflow. All experiments were conducted on a Server with two NVIDIA GTX 1080Ti graphics processing units (GPUs) and an Intel Xeon central processing unit (CPU) E5.

There are several hyperparameters in the AC-LSTM prediction model, including the time lag, the number of LSTM layers, the number of nodes in each LSTM layer, and the learning rate. They need to be preset before the model structure is built. Under the condition that all other parameters remain unchanged, we determined the optimal hyperparameter for that selected through our experiments. In the end, we built our model structure using four LSTM layers, and the number of nodes in each LSTM layer was set to 800. The learning rate was 0.0001 in all experiments. The above setting seemed to outperform all others in our experiments.

The time lag is one of the most important hyperparameters. It determines the number of past hours used in the model input and is necessary for multiscale prediction tasks. To this end, we evaluated the performance of the model with different time lags in order to find the optimal time lag in the model. At different time lags, we predicted the PM<sub>2.5</sub> concentrations of all stations in the training set in the next hour. The calculated MAE and RMSE are compared in Table 3. When the time lag was 10, the RMSE of the model was the lowest. While the time lag was 14, the MAE was the lowest. According to the analysis in Section 3.4 and the previous studies on RNN [10], if the time lag is too small, the temporal correlation between time-series data cannot be fully learned and the prediction accuracy will decrease. However, a large time lag may lead to a longer time for training and unnecessary noise. As a result, the time lag in our model was set to 12 for the one-hour prediction task. Of course, for prediction tasks of different time scales, we can also find the optimal lag through experiments in a similar way.

**Table 3.** Effect of different time lags. MAE—mean absolute error; RMSE—root-mean-square error.

Time Lag	2	4	6	8	10	12	14	16
MAE	8.21	7.89	7.9	7.82	7.75	7.68	7.61	8.04
RMSE	13.81	13.2	13.22	13.08	13.01	13.06	13.09	13.42

#### 4.2. Effects of Different Features

The input of our model was composed of three types of features: pollutant concentration ( $F_p$ ), meteorological data ( $F_m$ ), and PM<sub>2.5</sub> concentrations of adjacent monitoring stations ( $F_a$ ). To evaluate the effectiveness of different features in the proposed AC-LSTM model, we conducted experiments with different combinations of features and computed the errors on the multiscale prediction tasks. Because the number of monitoring stations in Taiyuan is too small, we used PM<sub>2.5</sub> concentrations from all stations rather than from adjacent stations. The effects of various features in AC-LSTM are shown in Tables 4 and 5. As can be seen, by gradually adding features, the prediction accuracy of the model could be generally improved. Except for the lowest MAE in the next 1 h and 13–24 h prediction tasks, the model with three types of features as input had the best overall performance. This shows that the past feature states and the PM<sub>2.5</sub> concentrations of adjacent monitoring stations can help predict the PM<sub>2.5</sub> concentration.

**Table 4.** The mean absolute error (MAE) of various features in AC-LSTM.

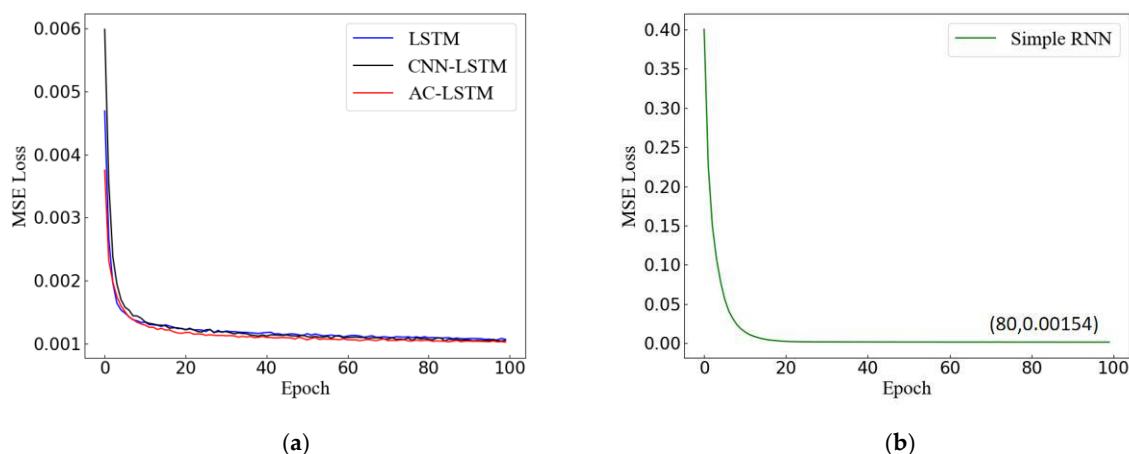
Features	1 h	2 h	3 h	4 h–6 h	7 h–12 h	13 h–24 h
$F_p$	7.61	8.12	8.21	8.53	8.57	9.04
$F_p + F_m$	7.98	7.99	7.99	8.51	8.6	8.89
$F_p + F_m + F_a$	7.68	7.97	7.98	8.38	8.51	8.98

**Table 5.** The root-mean-square error (RMSE) of various features in AC-LSTM.

Features	1 h	2 h	3 h	4 h–6 h	7 h–12 h	13 h–24 h
$F_p$	13.11	13.74	13.89	14.58	14.59	15.12
$F_p + F_m$	13.09	13.62	13.73	14.56	14.54	15.01
$F_p + F_m + F_a$	13.06	13.23	13.64	14.41	14.45	14.83

#### 4.3. Model Convergence

After setting appropriate model parameters, it was necessary to verify whether AC-LSTM converges during training. Therefore, the training loss of AC-LSTM model in the one-hour PM<sub>2.5</sub> prediction task was calculated, as shown in Figure 8, and the results were compared with three other deep learning methods (simple RNN, LSTM, and CNN-LSTM). The parameters of all models in Figure 8 were the same, and the mean square error (MSE) after data normalization was used as the loss function for training. It can be seen from Figure 8 that all models converged at epoch = 20. In Figure 8a, after 20 epochs in the one-hour prediction task, the MSE losses of the three models were close, but the MSE loss of the AC-LSTM model was slightly smaller than those of the other two models, LSTM and CNN-LSTM. At epoch = 1, the MSE loss of the simple RNN model in Figure 8b was nearly 100 times greater than that of the three models in Figure 8a. In addition, at epoch = 80, the loss value of the simple RNN model was 0.00154, while none of the other three models in Figure 8a had values greater than 0.0015. Obviously, the three models of LSTM, CNN-LSTM, and AC-LSTM in Figure 8a had better convergence results because of the special memory cell architecture.



**Figure 8.** The loss convergence of deep learning methods in one-hour PM<sub>2.5</sub> prediction: (a) loss convergence of LSTM, convolution neural network (CNN)-LSTM, and AC-LSTM models; (b) loss convergence of simple recurrent neural network (RNN) model.

#### 4.4. Model Comparison

To verify the feasibility and efficacy of the proposed model in this paper, we compared our proposed AC-LSTM model with six state-of-the-art models, including support vector regression (SVR) [6], random forest regression (RFR) [9], multilayer perceptron (MLP) [25], simple RNN [9,26], LSTM [27], and CNN-LSTM [28]. After training all the models with the same training and testing datasets, the PM<sub>2.5</sub> concentrations of all stations at different time scales were predicted for performance evaluation. We selected appropriate time lag and hyperparameters for different scale prediction tasks in our AC-LSTM model in the same way. Furthermore, each experiment was repeated five times, and the averaged results were used for comparison, as shown in Tables 6 and 7.

The prediction results from our approach and six others in terms of MAE and RMSE are compared in Tables 6 and 7, where several interesting observations can be highlighted. Firstly, the performance of all models gradually deteriorated as the time to predict became longer. For this purpose, the detailed comparison results of each model for different scale prediction tasks are shown in Figures A1–A4 in Appendix A. From Figures A1–A4, it is more obvious that the prediction accuracy of the three models (SVR, RFR, and MLP) worsened as the time to predict became longer. The lack of sufficient and directly relevant input data makes it difficult to predict PM<sub>2.5</sub> concentrations for longer future periods.

**Table 6.** The performances of the different models in terms of mean absolute error (MAE). SVR—support vector regression; RFR—random forest regression; MLP—multilayer perceptron.

Models	1 h	2 h	3 h	4 h–6 h	7 h–12 h	13 h–24 h
SVR	7.72	12.37	15.6	22.4	26.79	30.2
RFR	7.9	12.59	16.02	21.74	25.77	28.86
MLP	7.82	12.27	15.71	23.02	27.2	30.43
Simple RNN	8.91	8.9	8.88	9.39	9.75	9.94
LSTM	8.37	8.38	8.7	8.49	8.98	9.03
CNN-LSTM	7.79	7.97	8.05	8.38	8.79	8.92
AC-LSTM	7.68	7.97	7.98	8.38	8.51	8.98

**Table 7.** The performances of the different models in terms of root-mean-square error (RMSE).

Models	1 h	2 h	3 h	4 h–6 h	7 h–12 h	13 h–24 h
SVR	13.46	20.92	26.14	35.48	41.59	49.09
RFR	13.57	20.99	26.25	33.06	38.47	43.46
MLP	13.7	20.73	26.15	36.01	42.68	48.07
Simple RNN	14.1	14.24	14.62	15.19	15.38	15.15
LSTM	13.91	13.97	14.32	14.58	14.99	15.11
CNN-LSTM	13.25	13.73	13.84	14.43	14.53	15.02
AC-LSTM	13.06	13.23	13.64	14.41	14.45	14.83

Secondly, the performance of the four deep learning methods, i.e., simple RNN, LSTM, CNN-LSTM, and AC-LSTM, was much better than that of the three traditional shallow learning methods, SVR, RFR, and MLP, particularly in predicting over an hour. As can be seen from Tables 4 and 5, the MAE and RMSE of the four deep learning methods were relatively low. The predicted values of the four models on the multiscale prediction task were closer to the observed values in Figures A1–A4.

Thirdly, as can be seen from Figure A1 and the tables, the prediction accuracy of the three non-deep learning models on the one-hour prediction task was comparable to that of the four deep learning models. However, according to the goodness-of-fit plots for all models in Figure A5 the predicted value distributions of the three models were relatively dispersed, and their  $R^2$  values were lower than those of the four deep learning models. The predicted value distributions of the four deep learning models were close to a 45-degree line ( $y = x$ ). This, on one hand, shows the limitation of conventional approaches; on the other hand, it fully demonstrates the superior performance of the deep learning models in modeling long-term dependency for effective prediction of the PM<sub>2.5</sub> concentration in the future. The reason for this is that these three traditional shallow models cannot process time-series data and fail to learn the temporal correlation of air pollutants. By contrast, simple RNN is able to predict PM<sub>2.5</sub> concentrations over the next 24 h. Compared to simple RNN, the three models of LSTM, CNN-LSTM, and AC-LSTM bring further improved result from overcoming the defects of the conventional RNN.

Furthermore, according to the tables, the MAE and RMSE of AC-LSTM models were the lowest compared to other benchmarking models, except for the MAE for the 13–24 h prediction task. The predicted values of AC-LSTM models on the multiscale prediction task were closer to the observed values in Figures A1–A4. Moreover, the  $R^2$  of the AC-LSTM model in the one-hour PM<sub>2.5</sub> prediction task in Figure A5 was highest. After adding the attention mechanism, AC-LSTM could outperform the LSTM and CNN-LSTM in multiscale prediction tasks. The results show that the proposed AC-LSTM model can effectively learn the spatiotemporal correlation of air pollutants, and it is suitable for predicting urban PM<sub>2.5</sub> concentration in the future.

However, our study has several limitations. Emissions have a significant impact on air quality. Since emission data are difficult to obtain, the data collected in this paper do not include emissions from factories and vehicles in the area. This does affect the prediction accuracy of our model. Moreover, when a sudden pollution accident occurs, the PM<sub>2.5</sub> concentration changes suddenly. Whether the proposed model can predict it well still needs to be demonstrated.

## 5. Conclusions and Future Work

In this paper, we propose an attention-based CNN–LSTM model to predict urban PM<sub>2.5</sub> concentrations over the next 24 h. By taking the pollutant concentration in air quality data, meteorological data, and PM<sub>2.5</sub> concentrations in adjacent monitoring stations as the input, the model can learn the spatiotemporal correlation and long-term dependence of PM<sub>2.5</sub> concentrations. At the same time, the attention mechanism can capture the importance degrees of different feature states based on past time and further improve the prediction accuracy of the model. The experimental results show that the AC-LSTM model improved performance in the multiscale prediction tasks. Several main conclusions of this paper can be highlighted as follows:

1. Through the analysis of air quality data, PM<sub>2.5</sub> concentration has a strong spatiotemporal correlation. Due to the air flow, PM<sub>2.5</sub> concentration in the predicted area can be easily affected by the PM<sub>2.5</sub> concentrations of the adjacent monitoring stations. As PM<sub>2.5</sub> stays in the air for a long time, the past feature states also affect future PM<sub>2.5</sub> concentration. This motivated the design of a spatiotemporal model for effective prediction of PM<sub>2.5</sub> concentrations;
2. The experimental results indicate that, in addition to using only the pollutant concentrations of the air monitoring stations, adding the meteorological data and the PM<sub>2.5</sub> concentrations of the adjacent monitoring stations can improve the prediction accuracy of the model, especially for prediction tasks on time scales over one hour;
3. The proposed AC-LSTM model can be applied to multiscale predictors at different time gaps. When compared with the traditional machine learning methods, such as SVR, MLP, and RFR, its prediction accuracy was improved significantly, especially in predicting the PM<sub>2.5</sub> concentrations over the gap of one hour. In comparison with deep learning methods, such as simple RNN, LSTM, and CNN–LSTM, AC-LSTM produced improved prediction with lower MAE and RMSE measures due to the introduced attention mechanism in the LSTM model.

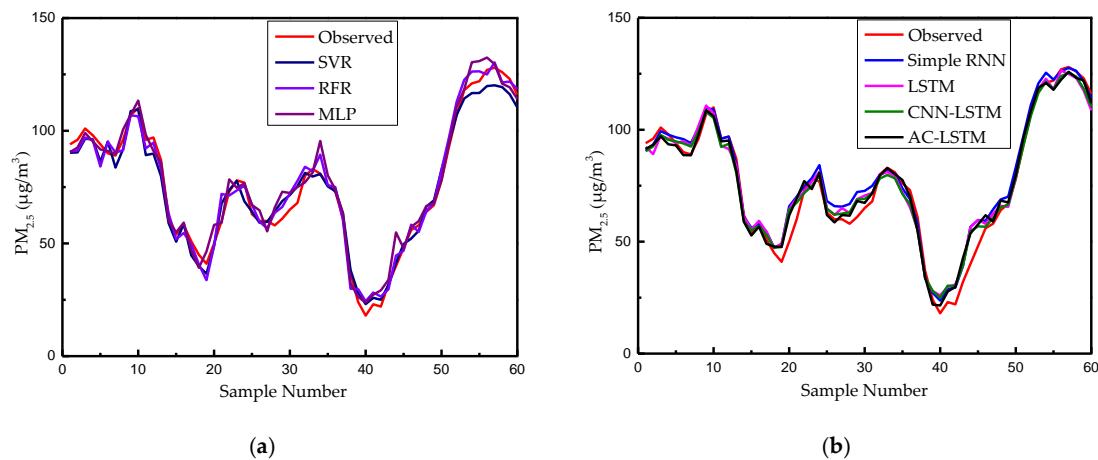
Although the proposed model can support the multiscale prediction of PM<sub>2.5</sub> concentrations in the temporal domain, in the future, we will also explore its expansion for multiscale prediction in the spatial domain. In addition, the model will also be extended for predicting other pollutants. Last but not least, sensing data, especially satellite data, will also be utilized for large-scale prediction of the PM<sub>2.5</sub> concentrations and other pollutants for early warning of air pollution and the protection of people's health.

**Author Contributions:** Methodology, S.L., Y.Y., and X.X.; software, S.L. and L.G.; supervision, G.X. and J.R.; writing—original draft, S.L.; writing—review and editing, G.X. and J.R. All authors read and agreed to the published version of the manuscript.

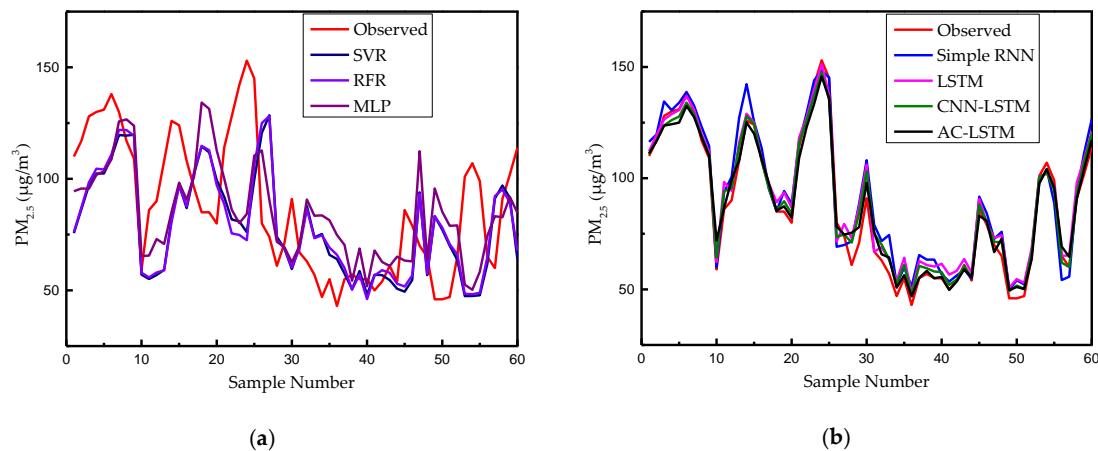
**Funding:** This work was supported in part by the Shanxi International Cooperation Project (No. 201803D421039), the Key Research and Development Plan of Shanxi Province (No. 201703D111027), the Key Research and Development Plan of Shanxi Province (No. 201703D111023), and The Hundreds Talent Program of Shanxi Province, China.

**Conflicts of Interest:** The authors declare no conflict of interest.

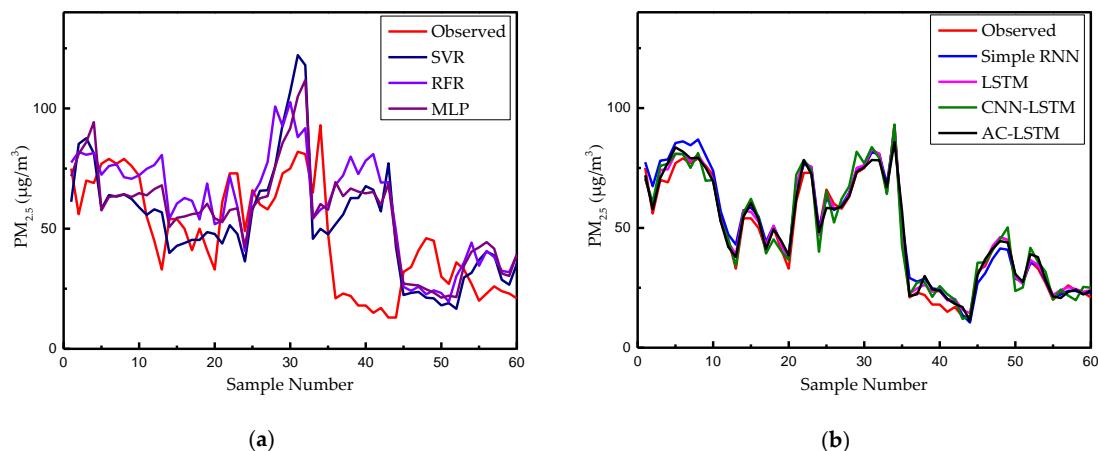
## Appendix A



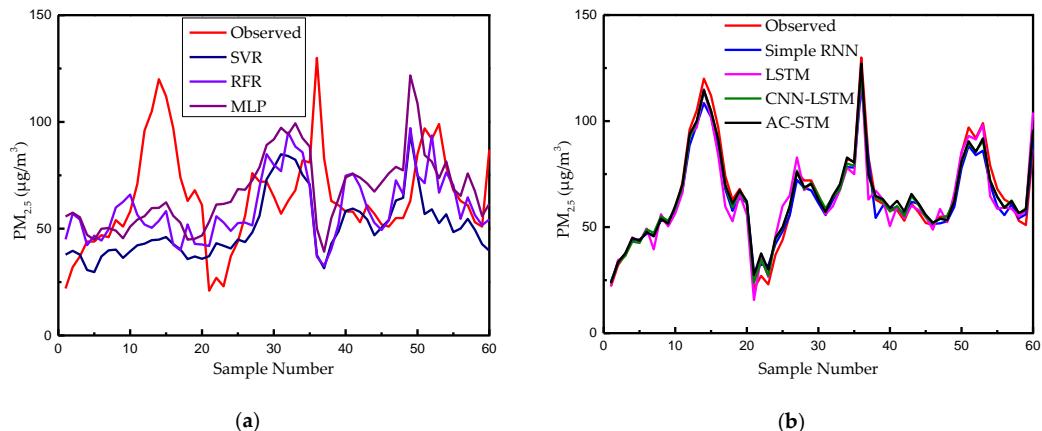
**Figure A1.** The results of all models in the one-hour PM<sub>2.5</sub> prediction: (a) SVR, RFR, and MLP; (b) simple RNN, LSTM, CNN-LSTM, and AC-LSTM.



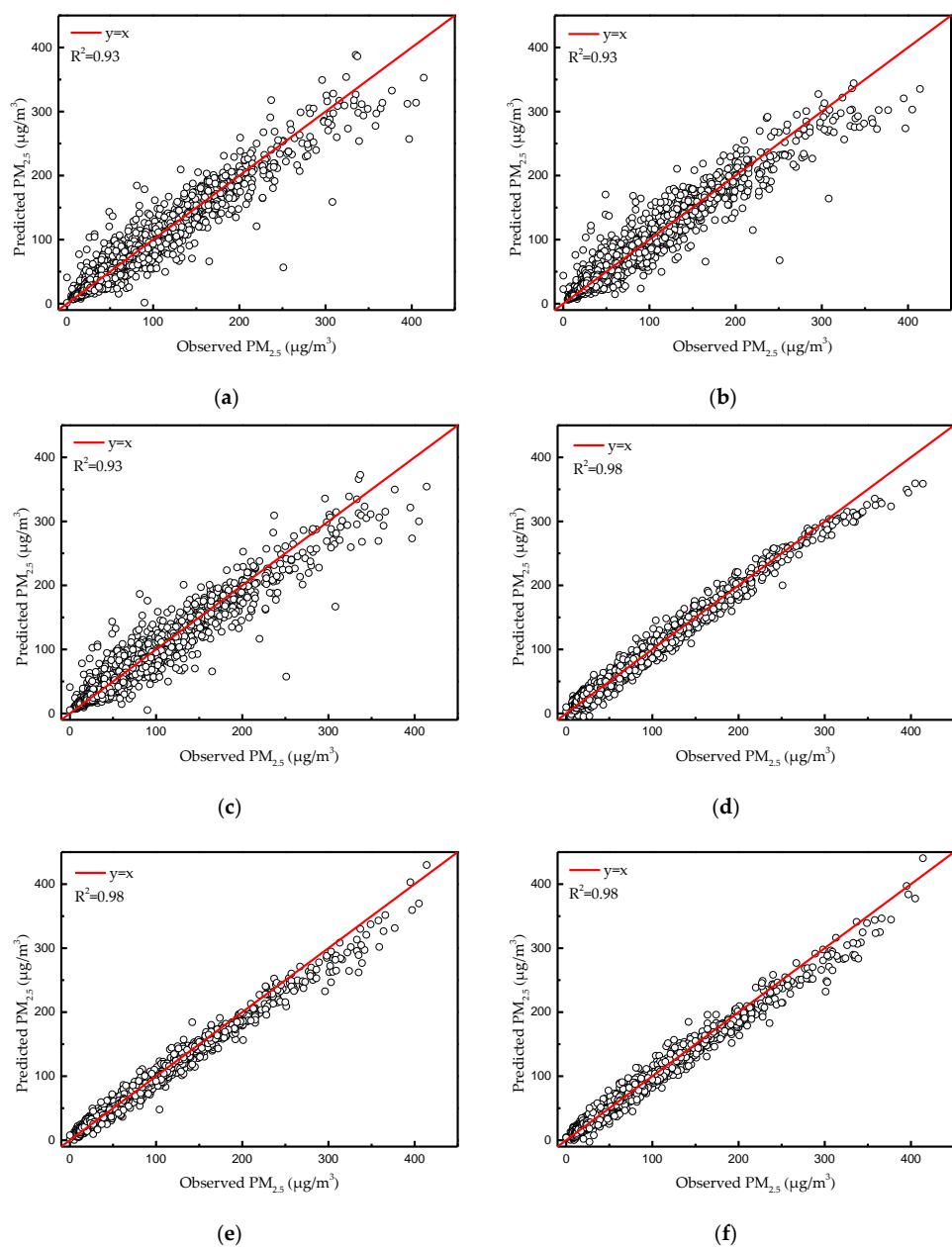
**Figure A2.** The results of all models in the 5-h PM<sub>2.5</sub> prediction: (a) SVR, RFR, and MLP; (b) simple RNN, LSTM, CNN-LSTM, and AC-LSTM.



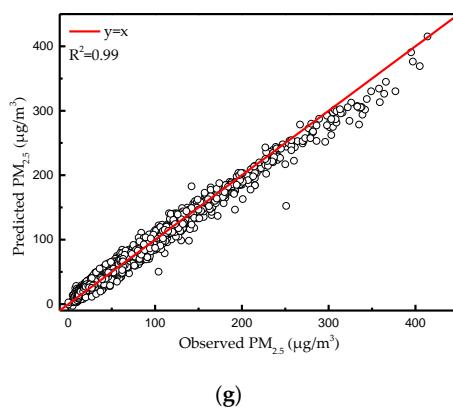
**Figure A3.** The results of all models in the 10-h PM<sub>2.5</sub> prediction: (a) SVR, RFR, and MLP; (b) simple RNN, LSTM, CNN-LSTM, and AC-LSTM.



**Figure A4.** The results of all models in the 19-h PM<sub>2.5</sub> prediction: (a) SVR, RFR, and MLP; (b) simple RNN, LSTM, CNN-LSTM, and AC-LSTM.



**Figure A5. Cont.**



**Figure A5.** The goodness-of-fit plots for all models in the one-hour PM<sub>2.5</sub> prediction: (a) SVR; (b) RFR; (c) MLP; (d) simple RNN; (e) LSTM; (f) CNN–LSTM; (g) AC-LSTM.

## References

- Kurt, A.; Oktay, A.B. Forecasting air pollutant indicator levels with geographic models 3 days in advance using neural networks. *Expert Syst. Appl.* **2010**, *37*, 7986–7992.
- Du, S.; Li, T.; Yang, Y.; Horng, S.J. Deep Air Quality Forecasting Using Hybrid Deep Learning Framework. *arXiv* **2018**, arXiv:1812.04783.
- Li, J.; Li, H.; Ma, Y.; Wang, Y.; Abokifa, A.; Lu, C.; Biswas, P. Spatiotemporal distribution of indoor particulate matter concentration with a low-cost sensor network. *Build. Environ.* **2018**, *127*, 138–147.
- Song, C.; Wu, L.; Xie, Y.; He, J.; Chen, X.; Wang, T.; Lin, Y.; Jin, T.; Wang, A.; Liu, Y.; et al. Air pollution in China: Status and spatiotemporal variations. *Environ. Pollut.* **2017**, *227*, 334–347.
- Zheng, Y.; Liu, F.; Hsieh, H.P. U-air: When urban air quality inference meets big data. In Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Chicago, IL, USA, 11–14 August 2013; pp. 1436–1444.
- He, H.; Li, M.; Wang, W.; Wang, Z.; Xue, Y. Prediction of PM<sub>2.5</sub> concentration based on the similarity in air quality monitoring network. *Build. Environ.* **2018**, *137*, 11–17.
- Ma, X.; Tao, Z.; Wang, Y.; Yu, H.; Wang, Y. Long short-term memory neural network for traffic speed prediction using remote microwave sensor data. *Transp. Res. Part. C Emerg. Technol.* **2015**, *54*, 187–197.
- Ong, B.T.; Sugiura, K.; Zettsu, K. Dynamically pre-trained deep recurrent neural networks using environmental monitoring data for predicting PM<sub>2.5</sub>. *Neural Comput. Appl.* **2016**, *27*, 1553–1566. [PubMed]
- Feng, R.; Zheng, H.; Gao, H.; Zhang, A.R.; Huang, C.; Zhang, J.X.; Luo, K.; Fan, J.R. Recurrent Neural Network and random forest for analysis and accurate forecast of atmospheric pollutants: A case study in Hangzhou, China. *J. Clean. Prod.* **2019**, *231*, 1005–1015.
- Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [PubMed]
- Recurrent models of visual attention. Available online: <https://arxiv.org/abs/1406.6247> (accessed on 24 June 2014).
- Bahdanau, D.; Cho, K.; Bengio, Y. Neural machine translation by jointly learning to align and translate. *arXiv* **2014**, arXiv:1409.0473.
- Rush, A.M.; Chopra, S.; Weston, J. A neural attention model for abstractive sentence summarization. *arXiv* **2015**, arXiv:1509.00685.
- Chaloulakou, A.; Kassomenos, P.; Spyrellis, N.; Demokritou, P.; Koutrakis, P. Measurements of PM10 and PM<sub>2.5</sub> particle concentrations in Athens, Greece. *Atmos. Environ.* **2003**, *37*, 649–660. [CrossRef]
- Hussein, T.; Karppinen, A.; Kukkonen, J.; Härkönen, J.; Aalto, P.P.; Hämeri, K.; Kerminen, V.M.; Kulmala, M. Meteorological dependence of size-fractionated number concentrations of urban aerosol particles. *Atmos. Environ.* **2006**, *40*, 1427–1440. [CrossRef]
- Chen, J.; Lu, J.; Avise, J.C.; DaMassa, J.A.; Kleeman, M.J.; Kaduwela, A.P. Seasonal modeling of PM<sub>2.5</sub>, in California's San Joaquin Valley. *Atmos. Environ.* **2014**, *92*, 182–190. [CrossRef]
- Zhang, C.; Ni, Z.; Ni, L. Multifractal detrended cross-correlation analysis between PM<sub>2.5</sub> and meteorological factors. *Phys. A Stat. Mech. Appl.* **2015**, *438*, 114–123. [CrossRef]

18. Ma, J.; Cheng, J.C.P. Estimation of the building energy use intensity in the urban scale by integrating GIS and big data technology. *Appl. Energy* **2016**, *183*, 182–192. [[CrossRef](#)]
19. Junninen, H.; Niska, H.; Tuppurainen, K.; Ruuskanen, J.; Kolehmainen, M. Methods for imputation of missing values in air quality data sets. *Atmos. Environ.* **2004**, *38*, 2895–2907. [[CrossRef](#)]
20. Pearson, K. Notes on Regression and Inheritance in the Case of Two Parents. *Proc. R. Soc. Lond.* **1895**, *58*, 240–242.
21. Box, G.E.P.; Jenkins, G.M.; Reinsel, G.C.; Ljung, G.M. *Time Series Analysis: Forecasting and Control*; John Wiley & Sons: Hoboken, NJ, USA, 2015.
22. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *60*, 1097–1105. [[CrossRef](#)]
23. Ma, J.; Ding, Y.; Cheng, J.C.P.; Jiang, F.; Xu, Z. Soft detection of 5-day BOD with sparse matrix in city harbor water using deep learning techniques. *Water Res.* **2019**, *170*, 115350. [[CrossRef](#)]
24. Peng, L.; Liu, S.; Liu, R.; Wang, L. Effective long short-term memory with differential evolution algorithm for electricity price prediction. *Energy* **2018**, *162*, 1301–1314. [[CrossRef](#)]
25. Jusoh, N.; Ibrahim, W.J.W. Evaluating Fuzzy Time Series and Artificial Neural Network for Air Pollution Index Forecasting. In *Proceedings of the Second International Conference on the Future of ASEAN (ICoFA) 2017—Volume 2*; Saian, R., Abbas, M.A., Eds.; Springer: Singapore, 2018; pp. 113–121.
26. Prakash, A.; Kumar, U.; Kumar, K.; Jain, V.K. A Wavelet-based Neural Network Model to Predict Ambient Air Pollutants' Concentration. *Environ. Model. Assess.* **2011**, *16*, 503–517. [[CrossRef](#)]
27. Li, X.; Peng, L.; Yao, X.; Cui, S.; Hu, Y.; You, C.; Chi, T. Long short-term memory neural network for air pollutant concentration predictions: Method development and evaluation. *Environ. Pollut.* **2017**, *231*, 997–1004. [[CrossRef](#)] [[PubMed](#)]
28. Huang, C.J.; Kuo, P.H. A deep cnn-lstm model for particulate matter (PM<sub>2.5</sub>) forecasting in smart cities. *Sensors* **2018**, *18*, 2220. [[CrossRef](#)] [[PubMed](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).