Multitask Air-Quality Prediction Based on LSTM-Autoencoder Model

Xinghan Xu and Minoru Yoneda

***Abstract*—With the development of the data-driven modeling techniques, using the neural network to simulate the transport process of atmospheric pollutants and constructing $PM_{2.5}$ time-series prediction model have become a hot topic. The existing data-driven approaches often ignore the dynamical relationships among multiple sites in urban areas, which results in nonideal prediction accuracy. In response to this problem, this article proposes a long short-term memory (LSTM) autoencoder multitask learning model to predict $PM_{2.5}$ time series in multiple locations city wide. The model could implicitly and automatically excavate the intrinsic relevance among the pollutants in different stations. And the meteorological information from the monitoring stations is fully utilized, which is beneficial for the performance of the proposed model. Specifically, multilayer LSTM networks can simulate the spatiotemporal characteristics of urban air pollution particles. And using the stacked autoencoder to encode the key evolution pattern of urban meteorological systems could provide important auxiliary information for $PM_{2.5}$ time-series prediction. In addition, multitask learning could automatically discover the dynamical relationship between multiple key pollution time series and solve the problem of insufficient use of multisite information in the modeling process of the traditional data-driven methods. The simulation results of $PM_{2.5}$ prediction in Beijing indicate the effectiveness of the proposed method.**

***Index Terms*—Autoencoder, long short-term memory (LSTM), multitask learning, $PM_{2.5}$.**

I. Introduction

With the development of industry and economy, the emission of various kinds of pollution gas and solid particle suspension increases year by year, which causes serious air pollution problems in many countries. China is the biggest developing country and the air pollution issue has become the concern among citizens [1]–[3]. In January 2017, haze in Beijing lasted for 26 days. Meanwhile, less than 1% of China's 500 largest cities reached the air-quality standards recommended by the World Health Organization. Therefore, air-quality prediction in city-wide area has been an important public concern.

As one of the most important indices to evaluate the air pollution, the prediction of $PM_{2.5}$ has been a key approach to study air pollution. Generally speaking, there are mainly two directions for $PM_{2.5}$ modeling: 1) knowledge-based models and 2) data-driven models [4], [5]. The knowledge-based models apply physical and chemical rules to model the transportation and transformation of air pollution particles. Many knowledge-based models have been proposed and some of them are currently being used for real-time air-quality forecast guidance. Di *et al.* [6] used the GEOS-Chem model to achieve the spatiotemporal prediction of air pollutants, including $PM_{2.5}$ and its major components and verified the validity of the model through simulation experiments. Saide *et al.* [7] proposed an air pollutant prediction system based on the WRF-Chem model and successfully used the system to predict the air quality in Santiago, Chile, in winter 2008. However, the successful applications of the knowledge-based models require a strong background in atmospheric science [8]. Furthermore, when the model is applied in different situations, the chemical and transport rules may change and the model is generally unavailable. Related studies indicate that the deterministic model performs worse than the data-driven model due to the highly complex and dynamic pollution processes of air quality as well as the uncertainties within models [9], [10].

With the arrival of the era of big data and artificial intelligence, the data-driven approach for air pollution modeling has been considered and applied in several $PM_{2.5}$ forecasting systems [11]. Cobourn [8] developed an enhanced $PM_{2.5}$ air-quality prediction model based on back-trajectory concentrations and nonlinear regression in Louisville, KY, USA. Qi *et al.* [1] proposed a deep-learning model that can be used simultaneously for interpretation, prediction, and feature analysis of the air-quality prediction models. The model fuses the interpretability in air-quality modeling into the deep-learning framework, and it is more interpretable without loss of prediction accuracy. Zheng *et al.* [12] proposed an air-quality prediction model that combines multiple sources of information in the city, taking into account the impact of road traffic and human production activities on air quality in the urban area.

Among these data-driven methods [1]–[3], [12]–[14], artificial neural networks have become the popular approach for air pollution modeling. Several applications have demonstrated the effectiveness of using the neural-network methods. Pérez *et al.* first used neural networks to predict the hourly $PM_{2.5}$ time series in Santiago, Chile, and prove that it is advanced than the persistence method and linear regression [15]. Ahani *et al.* proposed modeling the multistep-ahead

Manuscript received August 21, 2019; revised September 19, 2019; accepted September 30, 2019. This article was recommended by Associate Editor N. Zhang. *(Corresponding author: Xinghan Xu.)*

The authors are with the Department of Environmental Engineering, Kyoto University, Kyoto 6158540, Japan (e-mail: xu.xinghan.57u@st.kyoto-u.ac.jp).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2019.2945999

forecasting of fine particulate matter in urban areas. The results show that a recursive strategy with LASSO feature selection in the ARIMAX model outperforms the other methods [16]. Hsieh *et al.* [2] used the data from urban meteorological stations and heterogeneous data from multiple sensor sites in the city to infer future air quality and utilized the semisupervised learning methods to determine where to build the air-quality monitoring station according to human activities influence and meteorological dynamics in a city-wide area. Zhu *et al.* [3] proposed a method based on Granger causality analysis to determine which factor has the greatest causal effects on air quality in specific locations under urban big data and derived the air pollution influencing factors map of the Pearl River Delta region of China.

The deep-learning approach has been an effective method to model complex meteorological features and $PM_{2.5}$ time series. Considering the strong seasonal variation of air pollution, Bai *et al.* [17] proposed a seasonal stacked autoencoder model, which combines seasonal analysis and deep feature learning. The validity of the proposed method was proved by the data collected from three environmental monitoring stations in Beijing. Zhao *et al.* [18] proposed a model based on the long short-term memory (LSTM)-fully connected neural network, which can effectively use historical air-quality data to predict the concentration of $PM_{2.5}$ within 48 h. Qi *et al.* [19] and Wen *et al.* [20] fully exploited the advantages of the convolution neural network and LSTM neural network and proposed a spatiotemporal convolutional LSTM neural-network extended model and a graph convolutional network and LSTM network, respectively. The simulation results show that the proposed models can effectively predict the air quality in the future.

Multitask learning is an induction transfer method which aims to improve the generalization by using domain information contained in the training samples of related tasks [21]–[23]. It could improve generalization performance with the help of related tasks [10]. It is inspired by the human's learning process in which humans usually utilize the knowledge learned from the previous experience for a new task. Multitask learning could be divided into two categories: 1) symmetric multitask learning and 2) asymmetric multitask learning. The former one aims to promote the performance of all tasks simultaneously. However, asymmetric multitask learning seeks to promote the performance of a target task with the help of source tasks. In this sense, asymmetric multitask learning is related to transfer learning. For city-wide air-quality prediction, we intend to use multitask learning to implicitly learn the universal evolution pattern of multisite $PM_{2.5}$ time series.

To the best of our knowledge, most current deep-learning-based methods failed to simultaneously model the $PM_{2.5}$ time series and the meteorological data in a consolidated prognostic framework [24]–[26]. Motivated by the development of multitask learning in pattern recognition, we intend to model multilocation city-wide $PM_{2.5}$ time series. Considering the effectiveness of the deep-learning models, we propose the multitask LSTM model with a meteorological information encoder for the modeling of $PM_{2.5}$ time series. The multiple LSTM layer is carried out to extract the temporal and spatial characteristics of $PM_{2.5}$ time series. At the same time, we present the stacked autoencoder to encode the meteorological information of several locations. Furthermore, we use a cascade-parallel architecture for multitask learning and the prediction of $PM_{2.5}$ time series of different locations to discover the transportation pattern. The main contributions of this article are summarized as follows.

1) Considering the complex spatial and temporal dynamics of air pollutants, the temporal and spatial characteristics of particulate matter in multiple locations in the city are explored by spatial–temporal learning of multilayer LSTM networks.
2) Meteorological factors have a great influence on the evolution of $PM_{2.5}$. This article proposes encoding the key evolution modes of meteorological time series and provides important auxiliary information for $PM_{2.5}$ time-series prediction.
3) There is a strong correlation between the pattern of $PM_{2.5}$ time series among multiple locations, and this article utilizes multitask learning to automatically explore the patterns between key pollution monitoring stations and implicitly describes the relationship between various sites through the deep-learning model.
4) The modeling and simulation of $PM_{2.5}$ time series and meteorological observation information at multiple stations in Beijing show that the proposed method has achieved satisfactory performance due to the consideration of the relationship between multiple stations.

II. Proposed Model

In this part, we will mainly introduce the motivations and the problem formulation of this article. After that, the novel method proposed in this article is also introduced in detail.

A. Motivations and Problem Formulation

In the process of using the deep-learning method to predict the $PM_{2.5}$ time series in urban areas, the following aspects are mainly considered from the forecasting requirements.

1) Using Long Short-Term Memory Network to Learn Spatiotemporal Evolution Features: For existing $PM_{2.5}$ time-series prediction methods based on artificial neural networks, they mainly choose only one location for modeling, which would cause information loss. Therefore, we would adopt multilocation $PM_{2.5}$ time series. However, the traditional feedforward neural network could not capture the complex interactions among the multilocation time series. In this article, we adopt the LSTM neural network to model the complex evolution pattern behind the spatiotemporal time series.

The LSTM model was proposed in 1997 by Schmidhuber. The original purpose of the LSTM was to store information over extended time intervals [27]. In 2000, Gers and Schmidhuber proposed a modification version with peephole connections. In this method, the gates accept the input of the cell state. In 2005, Graves and Schmidhuber [29] presented the bidirectional LSTM that outperforms unidirectional ones. In 2014, Cho *et al.* [30] proposed a new structure of LSTM

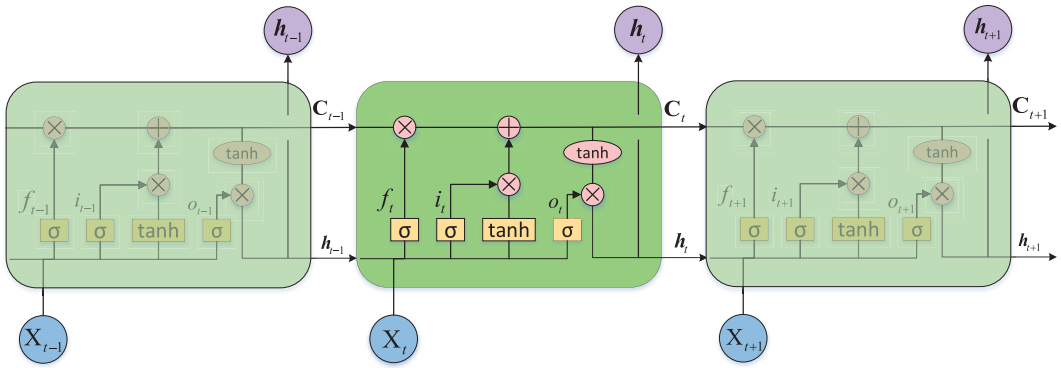

Fig. 1. LSTM unit.

called gated recurrent unit (GRU), which combines the forget gates and input gates into the update gates. Meanwhile, the GRU mixed the cell state and hidden state. Thus, the GRU is more simple and clear than the original LSTM model. In 2015, Yao *et al.* proposed a modification of LSTM by using the depth gate to connect memory cells of adjacent layers. It introduces linear dependence between lower and upper recurrent units [30]. Above the popular variants, there is no evidence to prove that one variant can improve the standard LSTM architecture significantly [31]. In this article, we focus on the application of the original LSTM for time-series prediction.

The LSTM network could be represented as Fig. 1. The procedure of LSTM could be viewed as a cell, where there are three gates to determine the states of the cell. In the picture, the forget gate $\boldsymbol{f}_i$ is a layer of the neural network, which could be written as

$$\boldsymbol{f}_t = \sigma\left(\mathbf{W}_f\left[\boldsymbol{h}_{t-1}, \mathbf{X}_t\right] + \boldsymbol{b}_f\right) \tag{1}$$

where σ is the sigmoid function. The forget gate determines what should be abandoned. $\boldsymbol{h}_{t-1}$ and $\mathbf{X}_t$ are the previous output of LSTM and present input. $\mathbf{W}_f$ and $\boldsymbol{b}_f$ are the weights and biases, respectively. The input gate $\boldsymbol{i}_t$ is another neural network, whose form is similar to forget gate $\boldsymbol{f}_i$ that

$$\boldsymbol{i}_t = \sigma\left(\mathbf{W}_i\left[\boldsymbol{h}_{t-1}, \mathbf{X}_t\right] + \boldsymbol{b}_i\right) \tag{2}$$

where the weights and biases are different.

The candidate value $\mathbf{C}_t$ can be represented as the combination of $\boldsymbol{h}_{t-1}$ and $\mathbf{X}_t$, where

$$\tilde{\mathbf{C}}_t = \tanh\left(\mathbf{W}_C\left[\boldsymbol{h}_{t-1}, \mathbf{X}_t\right] + \boldsymbol{b}_C\right). \tag{3}$$

Now, it is the time to update the state of cell $\mathbf{C}_t$. Multiply the historical state $\mathbf{C}_{t-1}$ with forget gate $\boldsymbol{f}_t$. Meanwhile, multiply the input gate with the candidate state $\tilde{\mathbf{C}}_t$, and add them together. It yields

$$\mathbf{C}_t = \boldsymbol{f}_t * \mathbf{C}_{t-1} + \boldsymbol{i}_t * \tilde{\mathbf{C}}_t. \tag{4}$$

Until now, we have updated the cell state. At last, the output $\boldsymbol{h}_t$ of the LSTM is based on the cell state, which combines the output gate and cell state

$$\begin{aligned} \boldsymbol{o}_t &= \sigma\left(\mathbf{W}_o\left[\boldsymbol{h}_{t-1}, \mathbf{X}_t\right] + \boldsymbol{b}_o\right) \\ \boldsymbol{h}_t &= \boldsymbol{o}_t * \tanh(\mathbf{C}_t). \end{aligned} \tag{5}$$

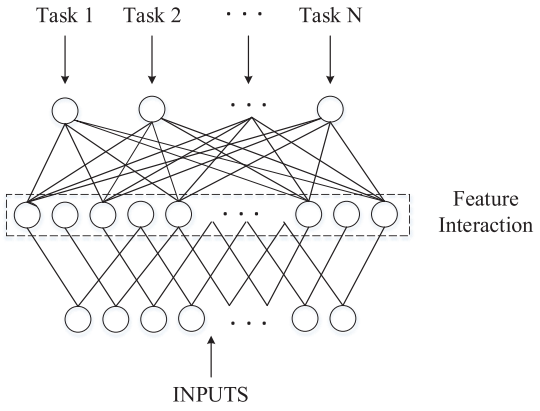

Fig. 2. Architecture of multitask learning.

The parameter could be trained by backpropagation through time (BPTT) algorithm, which is a variant of backpropagation. In the framework of Tensorflow, we use the Adam optimizer for parameter learning. The details of the Adam optimizer will be introduced in the simulation section.

2) Constructing Stacked Autoencoder for Meteorological Auxiliary Information Encoding: For $PM_{2.5}$ modeling, meteorological factors have a great impact [3], [32]. The traditional $PM_{2.5}$ modeling methods based on the mathematical–physical equation consider these factors by introducing lots of parameters. For artificial-intelligence modeling of the $PM_{2.5}$ time series, this research seldom considers the meteorological factors because of technique restriction. In this article, we impose utilizing the meteorological information and help to improve the prediction accuracy of the $PM_{2.5}$ time series. A stacked autoencoder with a sparse constraint is adopted to encode the evolution information.

In 2006, Hinton modified the autoencoder and made it better. An autoencoder is a kind of neural network, which aims to reconstruct a complete reconstruction from the input signal by learning an appropriate latent representation. The network explicitly defines a feature-learning function $\boldsymbol{h} = f_\theta(\mathbf{X})$, in which $f_\theta(\cdot)$ is called the encoder function. Then, the decoder function $\overline{\mathbf{X}} = g_\theta(\boldsymbol{h})$ learns to reconstruct the original signal. Feature vector $\boldsymbol{h}$ is the compressed representation of the original inputs. When the encoder and decoder learner f_θ and g_θ are stacked layer by layer, the learning paradigm

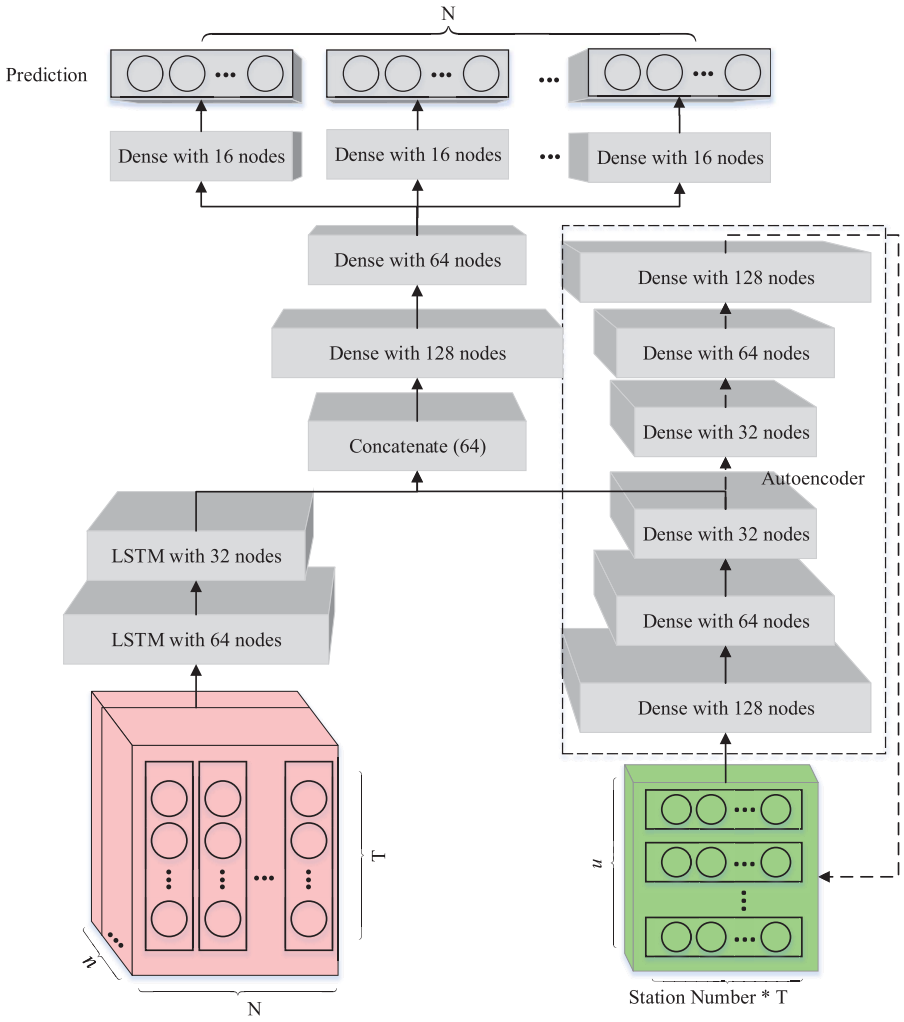

Fig. 3. Architecture of the proposed model.

is called stacked autoencoder neural networks. In this article, we aim to construct a vector representation for meteorological information and use it for modeling the $PM_{2.5}$ time series. The objective function of the meteorological autoencoder could be represented as follows:

$$\min \sum_{i=1}^{n} \text{Loss}\left(\mathbf{X}_{\text{Meteo},i} - g_\theta\left(f_\theta\left(\mathbf{X}_{\text{Meteo},i}\right)\right)\right) + R(\boldsymbol{\theta}) \quad (6)$$

where $\mathbf{X}_{\text{Meteo},i}$ is the meteorological information of the ith monitor location. Since the compressed representation $f_\theta(\mathbf{X}_{\text{Meteo}})$ has learned the key information for $PM_{2.5}$ time-series prediction hidden in inputs, it could be used for the auxiliary of the air pollutant modeling framework. In the stacked autoencoder, several constraints could be adopted for a particular purpose [33], [34]. $R(\boldsymbol{\theta})$ is the constraint term for the weights of the autoencoder. For example, some of the meteorological factors may not be relevant to the prediction of $PM_{2.5}$ time series or some of them contain little information for meteorological representation. Then, it could be ignored in the stacked autoencoder by using the sparse constraint. Sometimes, we expect the compressed representation to be smooth for the similar input factor. Then, the weight decaying constraint could be used for this purpose.

3) Using Multitask Learning for City-Wide Common Pattern Discovery: The deterministic modeling methods for air pollution, such as WRF-Chem, simulating the emission, transport, and mixing and chemical transformation of particles simultaneously with the meteorology. These modeling methods are interpretable for the modeling results. However, for the artificial intelligence modeling methods, the weights in the neural networks are not explainable for experts. In this article, we intend to discover the common evolution pattern of $PM_{2.5}$ time series of multiple monitor stations implicitly for the deep-learning model.

There are two kinds of multitask deep-learning methods, namely, hard parameter sharing and soft parameter sharing. The multitask learning based on the hard parameter sharing learns the common feature subspace in the basic layer of the neural network. In the higher layer of the neural network, it learns task-specific networks for particular tasks. In the basic layer, the parameter is totally the same. It would prevent overfitting problem and lead to better generalization. In the soft parameter sharing, the parameters of model for tasks may be

TABLE I
LOCATION OF AIR-QUALITY STATIONS IN BEIJING

Number	Location	Longitude	Latitude
1	Miyunshuiku	116.911E	40.499N
2	Nansanhuan	116.368E	39.856N
3	Nongzhanguan	116.461E	39.937N
4	Pingchang	116.23E	40.217N
5	Pinggu	117.1E	40.143N
6	Qianmen	116.395E	39.899N
7	Shunyi	116.655E	40.127N
8	Tiantan	116.407E	39.886N
9	Tongzhou	116.663E	39.886N
10	Wanliu	116.287E	39.987N
11	Wanshouxigong	116.352E	39.878N
12	Xizhimenbei	116.349E	39.954N
13	Yanqin	115.972E	40.453N
14	Yizhuang	116.506E	39.795N
15	Yongdingmennei	116.394E	39.876N
16	Yongledian	116.783E	39.712N
17	Yufa	116.3E	39.52N
18	Yungang	116.146E	39.824N

TABLE II
LOCATION OF METEOROLOGICAL MONITOR STATION IN BEIJING

Number	Location	Longitude	Latitude
1	Shunyi	116.6153E	40.1267N
2	Yanqing	115.9689E	40.4494N
3	Miyun	116.8642E	40.3775N
4	Huairou	116.6269E	40.3578N
5	Shangdianzi	117.1117E	40.6589N
6	Pinggu	117.1178E	40.1694N
7	Tongzhou	116.7567E	39.8475N
8	Chaoyang	116.5008E	39.9525N
9	Shijingshan	116.2053E	39.9425N
10	Fengtai	116.2453E	39.8703N
11	Daxing	116.3544E	39.7186N
12	Fangshan	116.1942E	39.7731N
13	Xiayunling	115.7406E	39.7286N

different and constrained by rules, which leads to a common feature subspace. For multistation air pollutant modeling, multitask learning can be used to constrain the underlying shared parameters of the deep-learning model. The guided model learns the shared representation of multiple task sites in the shared parameter layer and improves the parameter commonality of the model [35]. The parameter learning diagram of multitask learning is shown in Fig. 2. In order to minimize the loss function of multitask learning among the input and the output, the parameter optimizer, such as Adam, can automatically learn the interactions of the features in the optimization process. The objective function of multitask learning is as follows:

$$\min \sum_{i=1}^{N} \text{Loss}(\mathbf{X}, \mathbf{Y}_i, \boldsymbol{\theta}_i) \tag{7}$$

where $\mathbf{X}$ is the multitask input and $\mathbf{Y}_i$ is the respective target of multitask learning. $\boldsymbol{\theta}_i$ is the learning parameter corresponds to the ith task. N is the number of tasks. The constraints of multitask learning enable the optimizer to learn the shared representation of multiple tasks on the underlying parameters of the model to minimize the objective function.

B. Proposed Model

The architecture of our proposed model is shown in Fig. 3. The proposed method could be divided into two main parts in the low-level feature learning stage, namely, LSTM for spatial–temporal $PM_{2.5}$ time-series feature learning and autoencoder for meteorological factors encoding. Multilayer LSTM network is used for the spatiotemporal $PM_{2.5}$ time-series feature learning. The stacked autoencoder is used for the meteorological time-series encoding. Meteorological information, such as temperature, wind speed, and pressure, of multiple locations in the city could be fed into the model. Since the meteorological time series is densely sampled in space, we selected the meteorological monitoring station data that is spatially close to the $PM_{2.5}$ concentration monitoring station for auxiliary information modeling. The input dimension is high after phase space reconstruction. And concatenating the auxiliary information directly with the $PM_{2.5}$ feature representations may lead to difficulties. The stacked autoencoder could compress the useful information layer by layer and bring performance improvement. After LSTM feature learning and stacked autoencoder for $PM_{2.5}$ time series and meteorological time series, respectively, the encoding could be concatenated as a unified representation.

On the higher-level feature learning, we use two layers of dense network to learn comprehensive $PM_{2.5}$ evolution information and meteorological auxiliary. Based on the deep features, we use multiple subdense layers to model the $PM_{2.5}$ time series for multiple locations in the city-wide area and output the predicted values. The objective function of the entire model is

$$\min \sum_{i=1}^{N} \sum_{j=1}^{n} \left(y_{i,j} - f_{i,j}\left(\mathbf{X}_{\text{PM}_{2.5}}, \mathbf{X}_{\text{Meteo}}, \boldsymbol{\theta}\right)\right)^2 \tag{8}$$

where $y_{i,j}$ is the real value of the $PM_{2.5}$ time series. N is the number of the air-quality monitor stations. n is the number of time series. $\mathbf{X}_{\text{PM}_{2.5}}$ is the recorded value of all air-quality monitor stations. $\mathbf{X}_{\text{Meteo}}$ is the input of auxiliary meteorological information. $\boldsymbol{\theta}$ is all parameters of the proposed model.

III. SIMULATION ANALYSIS

In this section, we will introduce the dataset we use in this article and present the simulation analysis.

A. Dataset Description

Beijing's air-quality data includes the concentrations of several major air pollutants: $PM_{2.5}(\mu g/m^3)$, $PM_{10}(\mu g/m^3)$, $NO_2(\mu g/m^3)$, $CO(mg/m^3)$, $O_3(mg/m^3)$, and $SO_2(\mu g/m^3)$. There are 18 monitoring stations in Beijing. The name, latitude, and longitude are shown in Table I. The stations include urban area, suburban, and traffic pollution monitoring area.

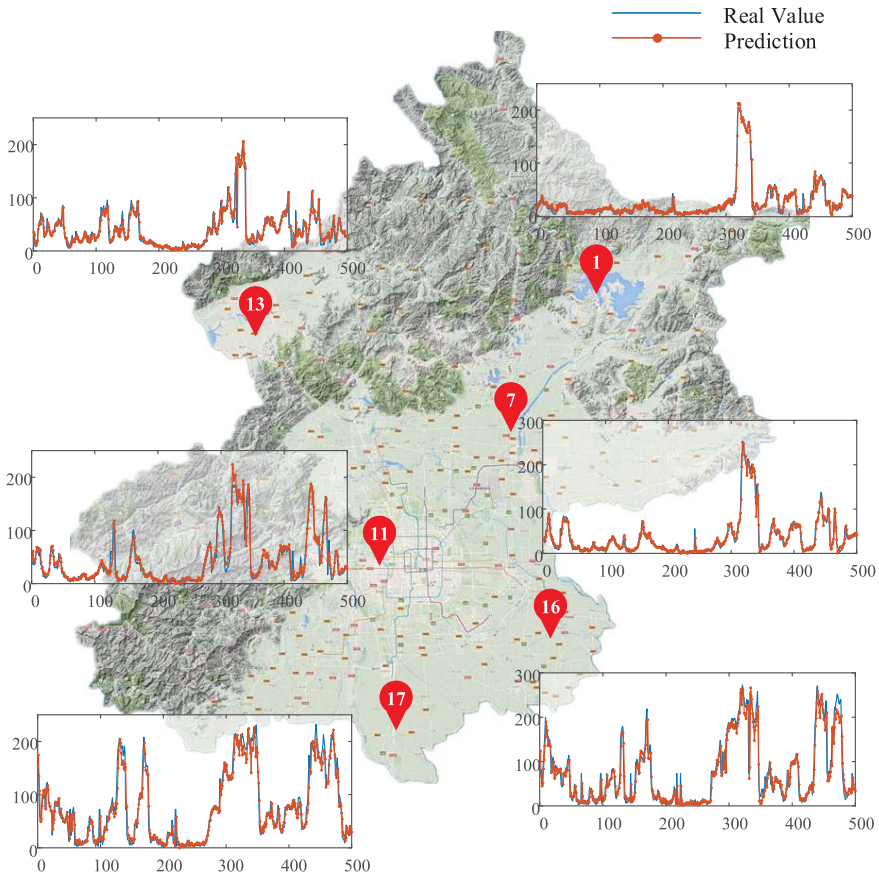

Fig. 4. Prediction of $PM_{2.5}$ time series in urban areas of Beijing.

The time series is sampled every hour from 4 P.M. on January 30, 2017 to 3 P.M. on January 31, 2018, for a total of 8784 samples.

In addition, the meteorological information is used for auxiliary to benefit the performance of the proposed model. In the Beijing dataset, the time series of 13 meteorological stations nearby the air-quality monitoring stations is adopted for the modeling. Temperature, pressure, humidity, wind direction, wind speed, and weather conditions are included in the meteorological data. The locations of the meteorological monitor stations are listed in Table II.

B. Experimental Settings

The architecture of the model is shown in Fig. 3. The optimization method is the most used ADAM optimizer. The model is run on the platform of Tensorflow 1.13 and Keras. In all experiments, the training data and the testing data account for 80% and 20% of the dataset, respectively. We use three evaluation indicators to compare the performance of the proposed model: 1) root-mean-square error (RMSE); 2) mean absolute error (MAE); and 3) symmetric mean absolute percentage error (SMAPE) according to the following equations:

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2} \tag{9}$$

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right| \tag{10}$$

$$\mathrm{SMAPE} = \frac{1}{2n}\sum_{i=1}^{n}\frac{\left|y_i - \hat{y}_i\right|}{\left(|y_i| + |\hat{y}_i|\right)} \times 100\% \tag{11}$$

where y is the predicted values in the model, and $\hat{y}$ is the target value.

In the simulation experiment of this article, the internal cell dimensions of the two-layer LSTM network are 64 and 32, respectively. The output of the first-layer LSTM network is the sequence with time step, and the output of the second-layer LSTM network is the last LSTM time-step vector without time information. In the stack autoencoder for encoding weather information, there are three layers of encoders and three layers of decoders. Among them, the nonlinear activation function of the encoder is a sigmoid function, and the nonlinear activation function of the decoder is the Relu function. The number of network nodes of the encoder and decoder are [128, 64, 32] and [32, 64, 128], respectively. In the entire autoencoder network, a weight decay regularization with a penalty coefficient of 0.0001 is employed.

After concatenating the vector representation and the weather information, the dimension of the representation is 64. This vector represents the dynamic representation of the evolution of the $PM_{2.5}$ time series with weather information coding. In the next layer of $PM_{2.5}$ spatiotemporal information and weather information interaction layer, the network nodes are

set as 128 and 64. On this basis, according to the idea of deep multitask learning, 18 multitask regression layers are connected to this layer to learn discriminative task-specific features of multitask learning. We use the Adam algorithm to optimize the parameters. The initial learning rate is set as 0.001. After that, the learning rate is multiplied by 0.1 for every 100 batches. A total of 500 batches are set, and the batch_size is 64. In LSTM, the input of each batch is the tensor of [N, 10, batch_size], and the input of the autoencoder is the matrix of [$N\times10$, batch_size].

In the LSTM comparative simulation experiment, the number of internal nodes of the network is [128, 64, 32]. The Adam optimizer is also used to optimize the model parameters. The selection of batch_size is the same as the method proposed in this article. The initial learning rate settings are the same as other similar methods to ensure that the other conditions are consistent. In the training process, by observing the loss function of the training set and the validation set, the early stopping is used to determine the batch number of the optimized iteration.

C. Experimental Analysis

In order to process the input data quickly and accurately, all inputs of the model are normalized to the range [−1, 1]. The proposed method is compared with several classical methods and the latest models regarding $PM_{2.5}$ time-series prediction.

In the feature sharing part of the model, it contains two parts: 1) extracting spatiotemporal distribution characteristics of the input data by LSTM and 2) extracting auxiliary information of meteorological factors by the autoencoder. The input of LSTM network is $PM_{2.5}$ time series with time varying and multilocation, and the temporal and spatial features of multiple locations are fused and extracted in the LSTM. At the same time, this part can also be seen as the hard parameter sharing of low-level multitask learning. Many different tasks share the same LSTM network. The parameter weights of feature interaction between different tasks can be learned by the optimization algorithm to achieve the goal of sharing multitask features. In the further layer of LSTM network unit, the features shared by the hard parameters of the underlying multitask learning are further studied in the higher dimension. After encoding by the two-layer LSTM network, the original spatiotemporal $PM_{2.5}$ time series is coded as a 32-D representation vector, which contains the spatiotemporal evolution information of all N stations. For 18 air pollution monitoring stations in Beijing, 18×10 = 180 time-series values can be compressed and characterized by 32-D vectors, which reflects the ability of the multitask learning to automatically discover the correlation between multiple tasks.

In the autoencoder part of the meteorological information, the weathered space–time information vector with input information of $11\times3\times10$ = 330 dimensions is compressed into a 32-D encoded representation through a compression autoencoding network of [128, 64, 32] structures. The dimension of the coding representation is much smaller than the original input dimension, which not only compresses the dimension of input information but also ensures that the compressed meteorological information contains enough meteorological evolution information.

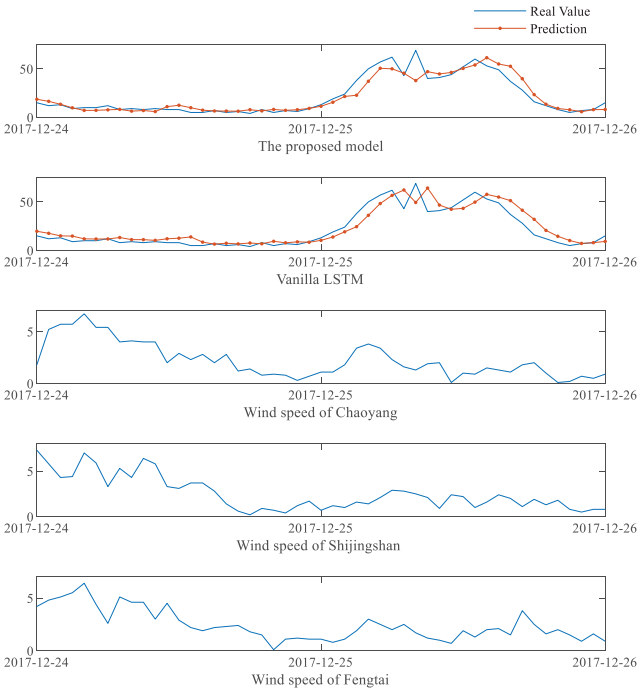

Fig. 5. Prediction comparison of our proposed model and vanilla LSTM (the normal part).

In the single-task regression part of the model, the fully connected network layer is used to learn the feature coding and auxiliary meteorological information coding of $PM_{2.5}$ space–time sequence. The learned features are connected through the task-specific layer of each task to learn the unique features of each task. In fact, all network parameters are shared in this network except the task-specific layer. This sharing of network parameters reduces the number of network nodes required for each task and the amount of computation, thereby, it is more suitable for joint multitask learning for a large number of monitoring sites.

The prediction results of the proposed model at several meteorological monitoring stations in Beijing are shown in Fig. 4. We selected the prediction results of several representative meteorological monitoring stations. These sites include rural areas, urban centers, industrial areas, and roadsides. It can be seen from the figure that the trends of $PM_{2.5}$ time series of multiple locations are the same, but the differences in the details between the locations are obvious. The parameter sharing part of the model learns the common trend of $PM_{2.5}$ time series of multiple meteorological monitoring stations. At the same time, the prediction results of $PM_{2.5}$ time series can be obtained by regression of 16 task-specific features. It is worth noting that there are 64 network nodes in the high-level feature layer of the model, which are connected to the prediction tasks of 18 sites. If the network does not learn the common representation of multitask, there will be only 3.5 neural-network nodes assigned to each task on average, and this is basically impossible. Therefore, this proves that the proposed model can take advantage of the goal of multitask learning and learn the

TABLE III
ONE-STEP-AHEAD PREDICTION RESULTS OF BEIJING $PM_{2.5}$ TIME SERIES

Station Name	Metrics	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	Avg
LSTM	RMSE	9.58	22.13	13.76	17.51	13.96	16.65	10.90	14.57	15.91	12.69	15.16	13.83	12.29	16.62	16.56	22.89	22.75	15.14	15.72
	MAE	5.56	12.66	7.84	9.86	8.29	9.93	6.33	8.06	8.79	7.09	8.61	7.85	7.24	8.46	9.14	13.37	12.23	7.73	8.84
	SMAPE	33.79	39.68	28.33	34.88	29.27	35.06	26.94	31.58	30.50	30.01	31.69	25.30	28.55	35.62	28.20	33.65	29.44	24.61	30.95
RNN	RMSE	9.69	22.41	13.67	17.66	14.13	16.73	10.93	14.78	15.89	12.81	15.12	13.96	12.00	16.87	16.72	22.93	22.69	15.43	15.80
	MAE	5.50	13.33	7.50	9.88	8.56	9.87	6.19	8.24	8.39	7.00	8.34	7.88	6.95	8.19	9.20	13.29	12.44	7.56	8.79
	SMAPE	33.19	41.25	27.25	34.64	30.70	35.23	26.02	31.90	28.21	29.41	30.19	25.23	26.58	32.53	30.00	33.25	31.33	23.47	30.58
ARIMA	RMSE	9.70	22.09	13.67	17.82	14.12	16.63	10.92	14.77	15.86	12.80	15.32	13.93	11.92	16.64	16.62	22.99	23.24	15.43	15.80
	MAE	5.46	13.42	7.50	9.83	8.51	9.79	6.15	8.30	8.32	7.01	7.91	7.95	6.84	8.25	9.12	13.35	13.32	7.53	8.81
	SMAPE	32.41	41.00	26.57	35.20	29.81	34.72	25.10	31.56	26.63	28.92	27.63	24.94	25.55	31.56	27.60	33.22	32.47	22.66	29.86
Proposed method	RMSE	8.91	20.16	14.03	17.08	11.80	14.41	9.32	15.12	11.40	13.50	13.50	14.00	11.67	14.21	15.30	21.56	21.72	13.82	14.52
	MAE	5.25	11.10	8.01	9.41	7.24	8.70	5.48	7.62	7.63	6.83	7.98	8.30	6.61	7.45	9.09	12.57	11.62	7.02	8.22
	SMAPE	32.87	33.51	30.98	33.95	26.45	33.75	25.76	31.21	26.55	29.91	30.55	27.17	26.13	30.72	29.41	31.05	25.83	22.84	29.37

TABLE IV
THREE-STEP-AHEAD PREDICTION RESULTS OF BEIJING $PM_{2.5}$ TIME SERIES

Station Name	Metrics	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	Avg
LSTM	RMSE	16.38	35.33	23.88	22.65	25.08	26.76	21.80	25.08	27.05	22.72	25.56	24.73	19.96	33.55	28.53	36.94	34.63	25.88	26.47
	MAE	9.78	24.04	14.70	14.18	16.38	16.88	13.57	15.64	17.68	13.49	16.37	15.44	13.20	19.19	17.79	24.97	22.79	15.54	16.76
	SMAPE	49.09	61.93	45.62	46.37	51.90	49.48	48.05	50.86	51.20	46.07	51.24	42.58	46.75	59.13	47.44	52.70	46.16	44.22	49.49
RNN	RMSE	16.51	35.07	23.92	22.98	25.04	26.84	21.99	25.18	27.16	22.82	25.61	24.98	19.72	32.45	28.53	36.87	35.10	25.94	26.48
	MAE	9.68	24.19	14.75	14.29	16.20	16.76	13.68	15.74	17.76	13.51	16.18	15.83	12.78	18.95	17.71	24.92	23.32	15.39	16.76
	SMAPE	47.99	63.22	46.83	46.35	51.02	48.87	48.77	51.41	51.85	46.51	50.43	43.61	44.05	59.97	47.43	52.82	49.66	42.20	49.61
ARIMA	RMSE	16.58	34.86	24.08	23.62	25.04	26.70	21.85	25.38	27.14	22.97	26.54	25.14	19.67	32.08	28.53	36.82	37.45	25.86	26.68
	MAE	9.31	24.45	14.90	14.04	16.02	16.76	13.27	15.90	17.31	13.44	15.01	16.11	12.63	19.14	18.02	25.21	26.24	15.24	16.83
	SMAPE	44.02	63.74	46.01	46.08	49.77	48.55	46.04	50.68	48.25	45.14	43.71	43.52	42.27	58.62	47.56	53.31	52.22	40.73	48.35
Proposed method	RMSE	16.17	33.33	22.05	22.75	21.17	23.32	20.42	23.28	26.88	20.80	22.97	23.17	21.02	26.85	25.56	33.58	41.44	22.79	24.87
	MAE	9.69	22.89	13.56	13.78	14.27	12.52	14.46	16.51	12.71	14.52	14.52	14.28	12.92	16.47	16.43	22.15	27.73	13.98	15.60
	SMAPE	48.57	61.15	45.62	44.90	46.84	47.73	49.14	46.88	50.01	46.24	48.49	40.59	46.21	55.82	47.53	49.16	45.56	40.60	47.84

common features of multiple tasks in the network. At the same time, it is also illustrated the validity of the proposed model in multitask prediction using auxiliary information.

The one-step-ahead prediction results and the three-step-ahead prediction results of $PM_{2.5}$ time series of the proposed method are shown in Tables III and IV, respectively. As can be seen from the tables, our proposed method achieves better performance with respect to various indicators (RMSE, MAE, and SMAPE), and the prediction error on each index is about 10% better than that of the compared methods. It is worth noting that although our proposed method does not achieve the best performance regarding some indicators, it takes the unified model method, considers the spatiotemporal dynamic relationship between $PM_{2.5}$ time series of multiple stations, and uses the meteorological coding auxiliary information to unify the prediction model of multiple sites into one frame. Within the framework, it has advantages in the efficient use of information and the computation of the model. The prediction method based on the LSTM model alone only considers the temporal and spatial characteristics of the $PM_{2.5}$ time series of multiple locations, but neglects the meteorological information which has great negative effects on $PM_{2.5}$. Therefore, the accuracy of the prediction model based on LSTM has not achieved the desired results. The $PM_{2.5}$ time-series forecasting model based on the ARIMA model cannot improve the forecasting effect of the model by means of multitask learning and meteorological auxiliary information. In terms of time comparison, our proposed model can deal with the $PM_{2.5}$ time-series prediction problem of multiple sites at the same time, while the comparison method can only deal with each task separately on each model, so the running time of the proposed method is about ten times shorter than that of the comparison method.

For more detailed comparisons, we compared the prediction results of our proposed model and the vanilla LSTM model. Specifically, we choose two ranges of representative prediction results from December 20, 2017 to December 24, 2017 and December 24, 2017 to December 26, 2017. Among the obtained results, the prediction results of Nongzhanguan air-quality monitor station is presented. Wind speed of three nearby meteorological monitor stations is presented along with the figure. The prediction results from December 24, 2017 to December 26, 2017 are shown in Fig. 5. In this part, the pattern of $PM_{2.5}$ time series is smooth and changes mildly. There is a little difference between the prediction results obtained by our proposed model and the vanilla LSTM. However, in Fig. 6, from December 20, 2017 to December 24, 2017, the

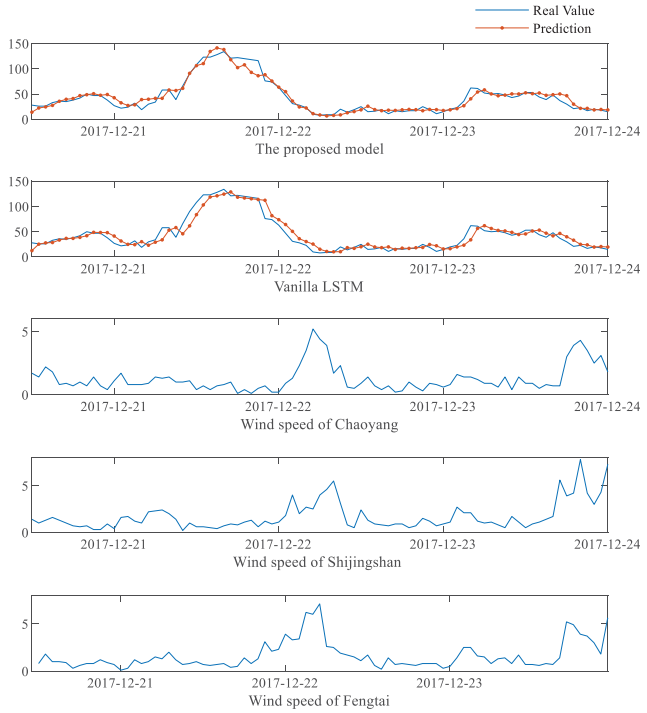

Fig. 6. Prediction comparison of our proposed model and vanilla LSTM (with drastic dynamic).

prediction results are quite different. The prediction result of our proposed model on December 21st performs better than the vanilla LSTM model. The prediction curve increases quickly so that it could forecast the air quality efficiently. In contrast, the vanilla LSTM failed to track the trend of the $PM_{2.5}$ time series.

IV. Conclusion

In this article, we proposed a novel multitask deep-learning model with autoencoded auxiliary information for air-quality time-series prediction. The proposed method could utilize the historical $PM_{2.5}$ time series and the meteorological time series of multilocations city wide. The multitask learning paradigm could capture the cross-task evolution pattern implicitly for time-series modeling. Simultaneously, we overcome the limitation of the data-driven $PM_{2.5}$ prediction methods (i.e., they neglect other provided information). The autoencoded meteorological auxiliary information could benefit from the performance of the $PM_{2.5}$ time-series prediction. The simulation results show that our proposed model could jointly predict the $PM_{2.5}$ time series at a city-wide level. Specifically, our proposed model could track the evolution pattern when the $PM_{2.5}$ time series encounter drastic changes. Meanwhile, the model could implicitly learn the common pattern of multiple $PM_{2.5}$ time series of multisite stations.

As for future work, we would like to model the $PM_{2.5}$ time series with more auxiliary information, such as the economy factor, emission of gas, and other deterministic processes that has never been considered by the traditional $PM_{2.5}$ prediction models, which deserves further and deeper explorations in the field of atmospheric modeling.

References

[1] Z. Qi, T. Wang, G. Song, W. Hu, X. Li, and Z. Zhang, "Deep air learning: Interpolation, prediction, and feature analysis of fine-grained air quality," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 12, pp. 2285–2297, Dec. 2018.

[2] H.-P. Hsieh, S.-D. Lin, and Y. Zheng, "Inferring air quality for station location recommendation based on urban big data," in *Proc. 21st ACM SIGKDD Int. Conf. Knowl. Disc. Data Min.*, Sydney, NSW, Australia, 2015, pp. 437–446.

[3] J. Y. Zhu, C. Sun, and V. O. K. Li, "An extended spatio-temporal Granger causality model for air quality estimation with heterogeneous urban big data," *IEEE Trans. Big Data*, vol. 3, no. 3, pp. 307–319, Sep. 2017.

[4] Y. Zhang, M. Bocquet, V. Mallet, C. Seigneur, and A. Baklanov, "Real-time air quality forecasting, part I: History, techniques, and current status," *Atmos. Environ.*, vol. 60, no. 32, pp. 632–655, 2012.

[5] G. O. Conti, B. Heibati, I. Kloog, M. Fiore, and M. Ferrante, "A review of AirQ models and their applications for forecasting the air pollution health outcomes," *Environ. Sci. Pollution Res.*, vol. 24, no. 7, pp. 6426–6445, 2017.

[6] Q. Di, P. Koutrakis, and J. Schwartz, "A hybrid prediction model for $PM_{2.5}$ mass and components using a chemical transport model and land use regression," *Atmos. Environ.*, vol. 131, pp. 390–399, Apr. 2016.

[7] P. E. Saide *et al.*, "Forecasting urban PM_10 and $PM_{2.5}$ pollution episodes in very stable nocturnal conditions and complex terrain using WRF-Chem Co tracer model," *Atmos. Environ.*, vol. 45, no. 16, pp. 2769–2780, 2011.

[8] W. G. Cobourn, "An enhanced $PM_{2.5}$ air quality forecast model based on nonlinear regression and back-trajectory concentrations," *Atmos. Environ.*, vol. 44, no. 25, pp. 3015–3023, 2010.

[9] B. Lv, W. G. Cobourn, and Y. Bai, "Development of nonlinear empirical models to forecast daily $PM_{2.5}$ and ozone levels in three large chinese cities," *Atmos. Environ.*, vol. 147, pp. 209–223, Dec. 2016.

[10] Y. Zhou, F.-J. Chang, L.-C. Chang, I.-F. Kao, Y.-S. Wang, and C.-C. Kang, "Multi-output support vector machine for regional multi-step-ahead $PM_{2.5}$ forecasting," *Sci. Total Environ.*, vol. 651, pp. 230–240, Feb. 2019.

[11] D. Radojević, D. Antanasijević, A. Perić-Grujić, M. Ristić, and V. Pocajt, "The significance of periodic parameters for ANN modeling of daily SO_2 and NOx concentrations: A case study of Belgrade, Serbia," *Atmos. Pollution Res.*, vol. 10, no. 2, pp. 621–628, 2019.

[12] Y. Zheng, F. Liu, and H.-P. Hsieh, "U-Air: When urban air quality inference meets big data," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Disc. Data Min.*, Chicago, IL, USA, 2013, pp. 1436–1444.

[13] J. Y. Zhu *et al.*, "PG-causality: Identifying spatiotemporal causal pathways for air pollutants with urban big data," *IEEE Trans. Big Data*, vol. 4, no. 4, pp. 571–585, Dec. 2018.

[14] Y. Zheng *et al.*, "Forecasting fine-grained air quality based on big data," in *Proc. 21st ACM SIGKDD Int. Conf. Knowl. Disc. Data Min.*, Sydney NSW, Australia, 2015, pp. 2267–2276.

[15] P. Pérez, A. Trier, and J. Reyes, "Prediction of $PM_{2.5}$ concentrations several hours in advance using neural networks in Santiago, Chile," *Atmos. Environ.*, vol. 34, no. 8, pp. 1189–1196, 2000.

[16] I. K. Ahani, M. Salari, and A. Shadman, "Statistical models for multi-step-ahead forecasting of fine particulate matter in urban areas," *Atmos. Pollution Res.*, vol. 10, no. 3, pp. 689–700, 2019.

[17] Y. Bai, Y. Li, B. Zeng, C. Li, and J. Zhang, "Hourly $PM_{2.5}$ concentration forecast using stacked autoencoder model with emphasis on seasonality," *J. Clean. Prod.*, vol. 224, pp. 739–750, Jul. 2019.

[18] J. Zhao, F. Deng, Y. Cai, and J. Chen, "Long short-term memory—fully connected (LSTM-FC) neural network for $PM_{2.5}$ concentration prediction," *Chemosphere*, vol. 220, pp. 486–492, Apr. 2019.

[19] Y. Qi, Q. Li, H. Karimian, and D. Liu, "A hybrid model for spatiotemporal forecasting of $PM_{2.5}$ based on graph convolutional neural network and long short-term memory," *Sci. Total Environ.*, vol. 664, pp. 1–10, May 2019.

[20] C. Wen *et al.* "A novel spatiotemporal convolutional long short-term neural network for air pollution prediction," *Sci. Total Environ.*, vol. 654, pp. 1091–1099, Mar. 2019.

[21] R. Caruana, "Multitask learning," *Mach. Learn.*, vol. 28, no. 1, pp. 41–75, 1997.

[22] X. Lu, X. Li, and L. Mou, "Semi-supervised multitask learning for scene recognition," *IEEE Trans. Cybern.*, vol. 45, no. 9, pp. 1967–1976, Sep. 2015.

[23] X. Gu, F.-L. Chung, H. Ishibuchi, and S. Wang, "Multitask coupled logistic regression and its fast implementation for large multitask datasets," *IEEE Trans. Cybern.*, vol. 45, no. 9, pp. 1953–1966, Sep. 2015.
[24] D. Lijie, Z. Changjiang, and M. Leiming, "Dynamic forecasting model of short-term PM2.5 concentration based on machine learning," *J. Comput. Appl.*, vol. 37, no. 11, pp. 3057–3063, 2017.
[25] Y. Hu, X. Sun, X. Nie, Y. Li, and L. Liu, "An enhanced LSTM for trend following of time series," *IEEE Access*, vol. 7, pp. 34020–34030, 2019.
[26] P.-W. Soh, J.-W. Chang, and J.-W. Huang, "Adaptive deep learning-based air quality prediction model using the most relevant spatial-temporal relations," *IEEE Access*, vol. 6, pp. 38186–38199, 2018.
[27] T. Zhang, L. Zang, Y. Wan, W. Wang, and Y. Zhang, "Ground-level $PM_{2.5}$ estimation over urban agglomerations in China with high spatiotemporal resolution based on Himawari-8," *Sci. Total Environ.*, vol. 676, pp. 535–544, Aug. 2019.
[28] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[29] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," *Neural Netw.*, vol. 18, nos. 5–6, pp. 602–610, 2005.
[30] K. Cho, B. van Merrienboer, C. Gulcehre, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machinetranslation," in *Proc. EMNLP*, 2014.
[31] K. Yao, T. Cohn, K. Vylomova, K. Duh, and C. Dyer, "Depth-gated recurrent neural networks," *arXiv:1508.03790*, 2015.
[32] J. He *et al.*, "Air pollution characteristics and their relation to meteorological conditions during 2014–2015 in major Chinese cities," *Environ. Pollution*, vol. 223, pp. 484–496, Apr. 2017.
[33] B. Du, W. Xiong, J. Wu, L. Zhang, L. Zhang, and D. Tao, "Stacked convolutional denoising auto-encoders for feature representation," *IEEE Trans. Cybern.*, vol. 47, no. 4, pp. 1017–1027, Apr. 2017.
[34] J. Han, D. Zhang, S. Wen, L. Guo, T. Liu, and X. Li, "Two-stage learning to predict human eye fixations via SDAEs," *IEEE Trans. Cybern.*, vol. 46, no. 2, pp. 487–498, Feb. 2016.
[35] C. Li, S. Huang, Y. Liu, and Z. Zhang, "Distributed jointly sparse multitask learning over networks," *IEEE Trans. Cybern.*, vol. 48, no. 1, pp. 151–164, Jan. 2018.

Xinghan Xu received the bachelor's degree from the Department of Atmospheric Science, Nanjing University, Nanjing, China, in 2010, and the master's degree from the Department of Environmental Engineering, Kyoto University, Kyoto, Japan, in 2015, where he is currently pursuing the Ph.D. degree in air quality modeling with the Graduate School of Engineering.

Minoru Yoneda received the bachelor's and Ph.D. degrees from the Graduate School of Engineering, Faculty of Engineering, Kyoto University, Kyoto, Japan.

He works on clarification of dynamism and exposure-dosage assessments of medium-boiling-point organic chemical substances and heavy metals and toxic organic chemical substances in general living environments to evaluate their risk for the public. He also aims to clarify the mechanisms of urban soil contamination with traces of harmful substances such as heavy metals, and to establish a method to decide optimum sampling points by using the probability theory. His current research interests include predictive assessment of health risks of environmental pollutants and environmental monitoring.