Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 9627776, 11 pages
https://doi.org/10.1155/2021/9627776

Research Article

Air Quality Prediction Model Based on Spatiotemporal Data Analysis and Metalearning

Kejia Zhang,[1] Xu Zhang,[1] Hongtao Song,[1] Haiwei Pan,[1] and Bangju Wang[2]

[1]*College of Computer Science and Technology, Harbin Engineering University, Harbin /150001, China*
[2]*College of Science, Huazhong Agricultural University, Wuhan /430070, China*

Correspondence should be addressed to Hongtao Song; songhongtao@hrbeu.edu.cn

Received 26 June 2021; Accepted 16 August 2021; Published 28 August 2021

Academic Editor: Xiao Zhang

With the continuous improvement of people's quality of life, air quality issues have become one of the topics of daily concern. How to achieve accurate predictions of air quality in a variety of complex situations is the key to the rapid response of local governments. This paper studies two problems: (1) how to predict the air quality of any monitoring station based on the existing weather and environmental data while considering the spatiotemporal correlation among monitoring stations and (2) how to maintain the accuracy and stability of the forecast even when the available data is severely insufficient. A prediction model combining Long Short-Term Memory networks (LSTM) and Graph Attention (GAT) mechanism is proposed to solve the first problems. A metalearning algorithm for the prediction model is proposed to solve the second problem. LSTM is used to characterize the temporal correlation of historical data and GAT is used to characterize the spatial correlation among all the monitoring stations in the target city. In the case of insufficient training data, the proposed metalearning algorithm can be used to transfer knowledge from other cities with abundant training data. Through testing on public data sets, the proposed model has obvious advantages in accuracy compared with baseline models. Combining with the metalearning algorithm, it gives a much better performance in the case of insufficient training data.

1. Introduction

Because of the increasingly serious air pollution all over the world, air quality has become one of the most socially concerned issues. In many countries, air quality has become a key indicator to measure the happiness index of residents. In order to achieve real-time monitoring of air quality, almost all countries have arranged a large number of air quality monitoring stations in major cities. Besides, more and more mobile portable monitoring devices are participating in air quality monitoring [1, 2]. Although many monitoring methods have been applied, it is still extremely challenging to make accurate predictions of air quality. Especially in the case of insufficient monitoring data or poor data quality, it is more difficult to maintain the accuracy and stability of the prediction model.

Air quality is affected by a variety of complex factors [3–6], including meteorological factors, industrial factors, fuel factors, traffic factors, and other human activity factors. The development of related monitoring equipment leads the collection of air quality data more and more comprehensive [7]. With the application of a series of spatiotemporal prediction models [8], air quality prediction has made considerable progress. Time correlation refers to the impact of historical monitoring data on future data, and spatial correlation refers to the mutual influence among adjacent monitoring stations. Most of the existing research works [3, 4] focus on establishing prediction models based on time correlation, while there are obvious shortcomings in the study of spatial correlation. The reason for this phenomenon is because the diffusion of air pollutants is affected by various factors such as geographical location, wind direction, wind speed, air pressure, and air humidity. The impact of each factor on the relevance of different regions is difficult to accurately model. In this paper, we propose a spatiotemporal model for air quality prediction. The proposed model combines Long and Short-Term Memory networks (LSTM) and Graph ATtention (GAT) mechanism [9], where LSTM is used to

capture the correlation in the time domain and GAT is used to model the spatial correlation among different regions.

In recent years, many deep learning models [3, 4, 10–13] have achieved good results in air quality prediction. However, the accuracy of these predicting models highly depends on the sufficiency of training data. In reality, a lack of sufficient training data is the most common situation. As we all know, air quality monitoring in developing countries mainly depends on monitoring stations arranged by the government. There are few or no such stations in small cities and towns. Insufficient training data makes it difficult for the existing prediction models to achieve accurate results in these small cities and towns. Even in large cities, government-monitoring stations are very sparse. Although there are some unofficial monitoring devices that can provide data as a supplement, the data collected by these simple monitoring devices are often of poor quality with large amounts of various dirty data and missing values. Therefore, making accurate predictions based on insufficient training data is a realistic and challenging problem. Transfer learning (metalearning) [14] is currently the most effective method to solve this problem. Some transfer learning models [15–17] have been proposed to predict air quality with insufficient data. However, these methods require a strong similarity between the source domain and the target domain. Different cities and towns (especially large cities and small towns) have huge differences in pollution levels, climate, pollutant diffusion conditions, and density of monitoring sites. This makes it difficult for the existing transfer learning technology to successfully transfer the knowledge acquired in large cities to the air quality prediction of small and medium cities. To meet these challenges, based on the proposed prediction model, we give a metalearning algorithm for knowledge transfer among cities with huge differences.

The main contributions of this paper are as follows:

(i) Proposing a spatiotemporal model by combining LSTM and GAT for accurate air quality prediction

(ii) Designing a metalearning algorithm for the proposed model, which can transfer knowledge among different cities and make an accurate prediction in case of insufficient training data

(iii) Verifying the advantages of the proposed model and meta-learning algorithm in the aspect of prediction accuracy through a large number of experiments

The rest of this paper is organized as Section 2 introduces some related research works in the area of air quality prediction, transfer learning, and metalearning; Section 3 gives the definition of the problems; the proposed prediction model and metalearning algorithm are introduced in Section 4; after showing the experimental results to prove the effectiveness of the proposed model and metalearning algorithm in Section 5, Section 6 summarizes the whole paper.

2. Related Works

This section will briefly present the related research works in the area of air quality prediction, transfer learning, and metalearning.

2.1. Air Quality Prediction. The machine learning models for air quality prediction can be divided into two categories: basic learning models and deep learning models. Basic learning models include linear regression, supporting vector regression, random forest, and LightGBM. Land Use Regression (LUR) [5, 6] makes air quality predictions through a linear regression model that takes into account multiple factors like regional population level, traffic condition, and land use condition. LUR does not consider the complicated spatiotemporal correlation of air pollution data, so the accuracy of prediction is poor. Later, the basic time series model autoregressive integrated moving average model (ARIMA) [18] appeared, which was used for time series forecasting with strong periodicity. However, it does not perform well for complex weather conditions. Random forest [19], LightGBM [20], deep learning methods have become widely used methods in air pollution prediction. Later, in order to further improve the accuracy of prediction, Zheng et al. [21] proposed U-Air, which uses a spatial classifier based on an artificial neural network (ANN) and a temporal classifier based on the linear-chain conditional random field (CRF) to capture temporal and spatial characteristics. Convolutional neural networks (CNN) are used to process data from Euclidean structures. For example, they are very effective in the field of image recognition, and it is impractical to use CNN directly to capture the spatial relationships between monitoring stations for sparse graph structures consisting of monitoring stations. The ConvLSTM model proposed in [22] combines CNN and LSTM to characterize the spatiotemporal relationship between monitoring stations, and it is still applicable to the spatial relationship in Euclidean space. The emergence of Graph Convolutional Networks (GCN) [23, 24] has made up for the deficiencies of CNN and is widely used in traffic data. GCN has realized the full use of the traffic network. GAT [9] are proposed on the basis of GCN, using an attention mechanism, and are good at capturing dynamic relationships between nodes. The ST-GAT model proposed by Zhang et al. [25] can dynamically capture the dynamic dependencies in the traffic network, making the traffic speed prediction results more advanced than existing models.

2.2. Transfer Learning and Meta-Learning. To improve the practicality of air quality prediction models, the obstacles caused by insufficient data must be resolved. Transfer learning can be divided into three categories according to the difference in source domains and target domains and tasks, namely, inductive transfer learning, transitive transfer learning, and unsupervised transfer learning [14]. In recent years, transfer learning combined with deep neural networks (DNN) has been widely used. The VGG model proposed in the image field [26], with the help of this model, can achieve fast and accurate model training under a small number of sets. Unlike image data, air quality data are more complex in spatial and temporal distribution. Hu et al. [27] proposed a DNN-based sharing model that fused multi-source wind speed data together to solve the problem of insufficient wind farm data. However, this model does not

provide a solution to the knowledge transfer of spatially related data.

Metalearning [28–30] can quickly initialize the model by learning knowledge in multiple different learning tasks in order to widely adapt to a variety of situations. Literature [29] firstly proposes the concept of metalearning, also known as learning. The goal is to train a metalearning model on multiple learning tasks, so as to use a small number of training samples to solve new learning tasks. A model-independent metalearning algorithm MAML is proposed in [28]. MAML deals with the situation of insufficient training data by transferring data and models among multiple learning tasks. Each update step consists of multitask pretraining, model migration, target task training, and model parameter synchronization. Unlike previous metalearning methods, MAML uses gradients to update model and does not introduce additional parameters. Literature [27] proposes a MAML-based spatiotemporal prediction model, which is used for urban traffic prediction and water quality prediction by transferring knowledge among multiple cities.

2.3. Summary of Related Works. Through the introduction of the above related works, it can be seen that the existing air quality prediction methods rarely consider the spatial relationship between multiple monitoring stations. A few spatiotemporal prediction models lack the ability to dynamically model spatial correlation based on weather and other related factors. The only methods that can dynamically model spatial correlation do not consider how to deal with insufficient training data. Some existing methods in the area of transfer learning and metalearning can solve the insufficient-training-data situation to a certain extent by transferring the knowledge from other source domains, but these methods lack the ability to adapt to the air quality spatiotemporal prediction models and cannot be directly applied to the scenarios targeted in this article. For this reason, this paper proposes a spatiotemporal model for air quality prediction and a metalearning algorithm for this model. The prediction model can dynamically and accurately model the temporal and spatial correlation in air quality prediction. The metalearning algorithm is used to establish a more accurate prediction model in the case of insufficient training data. As far as we know, it is the first time that metalearning has been used for air quality prediction.

3. Problem Formulation

This paper will solve two problems: prediction problem and transfer learning problem. The prediction problem is how to build a prediction model for the target pollutant in the city with sufficient training data. The transfer learning problem is how to build a prediction model in the target city with insufficient training data, given the source cities with sufficient data. The symbols used in this paper are given in Table 1.

3.1. Prediction Problem. Suppose that there is a set of urban monitoring stations $\boldsymbol{S} = \{s_1, s_2, \cdots, s_n\}$ in the target city. We use a fixed time interval while counting historical data and making predictions. The prediction problem is building a model to predict the concentration of a certain pollutant sampled by a specified monitoring station in the future. The target air pollutant can be one of PM2.5, PM10, SO_2, NO_2, O_3, CO, AQI (can be regarded as a comprehensive pollutant). Suppose that the current time is t. The input of the prediction model contains (1) a specified monitoring station, (2) the historical monitoring data of the target pollutant sampled from time $t-k$ to time t, (3) the historical weather information from time $t-k$ to time t, and (4) the weather forecast information from time $t+1$ to time $t+l$. The output of the prediction model is the predicted value of the target pollutant sampled by the specified station from time $t+1$ to time $t+l$. In practical applications, we usually set $k+1=2l$.

For a monitoring station $s \in \boldsymbol{S}$, define the historical data vector of s as $\boldsymbol{x}_s = (x_s^{t-k}; x_s^{t-k+1}; \cdots; x_s^t)$, where x_s^i $(t-k \le i \le t)$ is the concentration of the target pollutant sampled by station s at time i. The historical data vectors of all monitoring stations can be represented by a matrix $\boldsymbol{X} = (\boldsymbol{x}_{s_1}, \boldsymbol{x}_{s_2}, \cdots, \boldsymbol{x}_{s_n})$.

The weather information used by the prediction model includes temperature, humidity, pressure, wind direction, and wind speed. The historical weather dataset of the target city is expressed as $\boldsymbol{W}_h = (\boldsymbol{w}^{t-k}, \boldsymbol{w}^{t-k+1}, \cdots, \boldsymbol{w}^t)$, in which $\boldsymbol{w}^i$ $(t-k \le i \le t)$ represents the vector of the above weather indexes sampled at time i. The weather forecast dataset of the target city is expressed as $\boldsymbol{W}_f = (\boldsymbol{w}^{t+1}, \boldsymbol{w}^{t+2}, \cdots, \boldsymbol{w}^{t+l})$, in which $\boldsymbol{w}^i (t+1 \le i \le t+l)$ represents the forecast value of the above weather indexes at time i. Usually, $\boldsymbol{W}_h$ and $\boldsymbol{W}_f$ are given by the meteorological department.

Given the target monitoring station $s \in \boldsymbol{S}$, our goal is to predict the concentration of the target pollutant sampled by s from time $t+1$ to time $t+l$, which can be expressed as a vector $\boldsymbol{y}_s = (y_s^{t+1}; y_s^{t+2}; \cdots, y_s^{t+l})$. Suppose that f_θ with parameter θ is the model we build, so we have

$$\hat{\boldsymbol{y}}_s = f_\theta(\boldsymbol{X}, \boldsymbol{W}_h, \boldsymbol{W}_f, s), \tag{1}$$

where $\hat{\boldsymbol{y}}_s$ is the predicion of $\boldsymbol{y}_s$.

Let $\boldsymbol{D}_{tc}$ be the training dataset of the target city. $\boldsymbol{D}_{tc}$ contains the historical monitoring data of all monitoring stations, the historical weather data, and the historical weather forecast data collected from the target city over a period of time. The prediction problem can be formally defined as how to build an accurate prediction model f_θ based on $\boldsymbol{D}_{tc}$.

3.2. Transfer Learning Problem. In addition to constructing the prediction model, another important issue to be solved in this paper is how to make accurate predictions when there is little training data. In this case, we will transfer knowledge from the source cities with sufficient training data to the target city with insufficient data. Suppose that we have m source cities with sufficient training data. Let $\boldsymbol{D}_{sc}^1$, $\boldsymbol{D}_{sc}^2$, ..., $\boldsymbol{D}_{sc}^m$ be the training datasets collected from the source cities, respectively. Let $\boldsymbol{D}_{tc}$ be the insufficient training dataset collected from the target city. The transfer learning problem

Table 1: Symbols and explanations.

Symbol	Meaning
$\boldsymbol{S} = \{s_1, s_2, \cdots, s_n\}$	Set of monitoring stations in the target city
k	Length of the historical data
t	Current time
l	Length of the forecast window
x_s^i	Concentration of target pollutant monitored by station s at time i (past)
$\boldsymbol{x}_s = \left(x_s^{t-k}; x_s^{t-k+1}; \cdots, x_s^t\right)$	Historcical data vector of station s
$\boldsymbol{X} = \left(\boldsymbol{x}_{s_1}, \boldsymbol{x}_{s_2}, \cdots, \boldsymbol{x}_{s_n}\right)$	Historcical data vectors of all stations
$\boldsymbol{w}^i (t-k \le i \le t)$	Historical weather of the target city at time i
$\boldsymbol{W}_h = \left(\boldsymbol{w}^{t-k}, \boldsymbol{w}^{t-k+1}, \cdots, \boldsymbol{w}^t\right)$	Historical weather dataset
$\boldsymbol{w}^i (t+1 \le i \le t+l)$	Weather forecast of the target city at time i
$\boldsymbol{W}_f = \left(\boldsymbol{w}^{t+1}, \boldsymbol{w}^{t+2}, \cdots, \boldsymbol{w}^{t+l}\right)$	Weather forecast dataset
$\boldsymbol{T} = \left\{\boldsymbol{D}_{\boldsymbol{sc}}^{\boldsymbol{1}}, \boldsymbol{D}_{\boldsymbol{sc}}^{\boldsymbol{2}}, \cdots, \boldsymbol{D}_{\boldsymbol{sc}}^{\boldsymbol{m}}\right\}$	Set of training dataset from source cities
$\boldsymbol{D}_{tc}$	Training dataset from the target city
$y_s^i (t+1 \le i \le t+l)$	Concentration of target pollutant monitored by station s at time i (future)
$\boldsymbol{y}_s = \left(y_s^{t+1}; y_s^{t+2}; \cdots, y_s^{t+l}\right)$	Concentration of target pollutant at station s in the future
f_θ	Prediction model with parameter θ
$\mathscr{L}_{\boldsymbol{B}}(f_\theta)$	Loss of model f_θ on training batch $\boldsymbol{B}$
$\widehat{\boldsymbol{y}}_s = \left(\widehat{y}_s^{t+1}; \widehat{y}_s^{t+2}; \cdots, \widehat{y}_s^{t+l}\right)$	Predicted value of $\boldsymbol{y}_s$.
r	Influence radius of monitoring stations
G_r	Directed graph built on all the monitoring stations
$N_r(s)$	$N_r(s) = \{s\} \cup \left\{s' \mid \left\langle s', s\right\rangle \in G_r\right\}$
$\boldsymbol{h}_s^i$	Output vector of the LSTM unit on station s at time i
$\boldsymbol{c}_s^i$	Cell state vector of the LSTM unit on station s at time i
$\boldsymbol{z}_s^i$	Output vector of GAT on station s at time i
$\alpha_{s',s}$	GAT's similarity score between node s' and s
$\widehat{\alpha}_{s',s}$	Weight of the edge $\left\langle s', s\right\rangle$ in G_r calculated by GAT

is defined as how to build an accurate prediction model f_θ for the target city based on $\boldsymbol{D}_{\boldsymbol{tc}} \cup \boldsymbol{D}_{\boldsymbol{sc}}^{\boldsymbol{1}} \cup \boldsymbol{D}_{\boldsymbol{sc}}^{\boldsymbol{2}} \cup \cdots \cup \boldsymbol{D}_{\boldsymbol{sc}}^{\boldsymbol{m}}$.

4. Methodology

4.1. Monitoring Station Graph. In order to measure the mutual influence among different monitoring stations, we initially model all monitoring stations as a directed graph G_r. Monitoring stations are represented by the nodes (vertices) in G_r. Given two nodes s_i and s_j, there are directed edges $\langle s_i, s_j\rangle$ and $\langle s_j, s_i\rangle$ if the Euclidean distance between s_i and s_j is less than or equal to r. r is the influence radius of monitoring stations, i.e., the maximum range affected by the pollutant in the diffusion process. As shown in Figure 1, by setting $r = 20$km, we get the graph among 34 monitoring stations located in Beijing, China. The weights of the edges in G_r will be calculated by GAT mechanism and change over time. Hereinafter, we use the term "node" to refer to monitoring station and define set $N_r(s)$ as

$$N_r(s) = \{s\} \cup \left\{s' \mid \left\langle s', s\right\rangle \in G_r\right\}. \tag{2}$$

4.2. Air Quality Prediction Model. To solve the prediction problem, we propose a spatiotemporal prediction model (referred as GAT-LSTM) as shown in Figure 2. The model is built by a recurrent neural network incorporating graph attention mechanism, which means that it has encoder-decoder structure. The encoder is used to embed historical data, and the decoder is used to generate the predicted value in the future. It uses LSTM to model time correlation of a

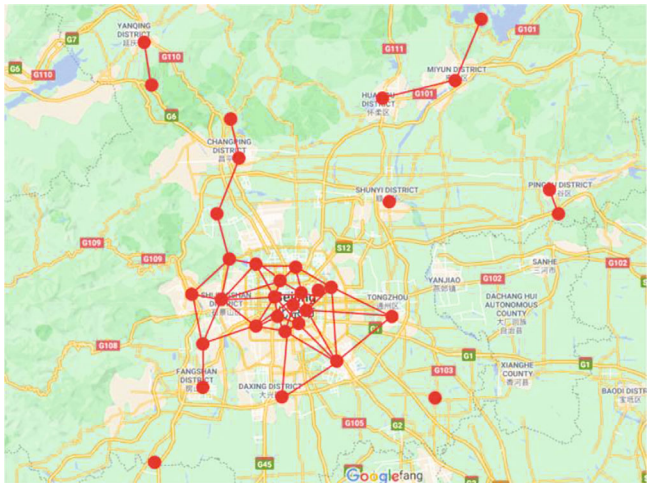

FIGURE 1: The Graph of 34 monitoring stations in Beijing ($r = 20$km).

node's own data and uses GAT to capture spatial correlation among nodes.

Suppose that the current time is t. Take node s as an example. In the encoding phase, we use $k+1$ LSTM units to receive s's historical data from time $t-k$ to time t, and form them into a recurrent neural network. For time $i\,(t-k \le i \le t)$, the input of the corresponding LSTM unit is $(x_s^i; \boldsymbol{w}^i)$, where x_s^i is the concentration of target pollutant monitored by s at time i, and $\boldsymbol{w}^i$ is the historical weather data at time i. Let $\boldsymbol{h}_s^i$ and $\boldsymbol{c}_s^i$ be the output vector (blue lines in Figure 2) and the cell state vector (gray lines in Figure 2) of the LSTM unit at time i, respectively. Unlike the traditional approach passing $\boldsymbol{h}_s^i$ and $\boldsymbol{c}_s^i$ directly to the next LSTM unit, we pass $\boldsymbol{h}_s^i$ to GAT to find spatial correlation among different nodes. Let $\boldsymbol{z}_s^i$ be the output vector of GAT for node s at time i. In the end, $\boldsymbol{z}_s^i$ and $\boldsymbol{c}_s^i$ are passed to the next LSTM unit. The structure of LSTM unit is as shown in Figure 3.

In the decoding phase, l LSTM units are used to generate $(\hat{y}_s^{t+1}; \hat{y}_s^{t+2}; \cdots, \hat{y}_s^{t+l})$, i.e., the predicted concentration of the target pollutant sampled by node s. For time $j\,(t+1 \le j \le t+l)$, the input of the corresponding LSTM unit is $(y_s^{j-1}; \boldsymbol{w}^j)$, where y_s^{j-1} is the prediction of the previous moment and $\boldsymbol{w}^j$ is the weather forecast data at time j. As with the coding phase, we pass the LSTM's output vector $\boldsymbol{h}_s^j$ to GAT to generate vector $\boldsymbol{z}_s^j$. In addition to being passed to the next LSTM unit, $\boldsymbol{z}_s^j$ is also passed to a Feedforward Neural Network (FNN) to generate the output y_s^j.

Base on the monitoring station graph G_r, we use a GAT to model the spatial relationship among different nodes. In GAT, each node uses the attention mechanism [31] to collect information from neighbor nodes (weighting and summing the feature vectors of neighbor nodes) and uses the collected information to update its own feature vector. Unlike GCN, the weight of an edge in GAT is calculated based on the similarity of the feature vectors of the two corresponding nodes and changes dynamically with the change of the node's data. GAT is very sensitive to the changes of the spatial correlation among nodes caused by weather factors such as wind speed and wind direction.

The GAT mechanism can be demonstrated by Figure 4. At any time, the input of GAT is the output vectors of all nodes' LSTM units, i.e., $\{\boldsymbol{h}_{s_1}, \boldsymbol{h}_{s_2}, \cdots, \boldsymbol{h}_{s_n}\}$. The output of GAT is $\{\boldsymbol{z}_{s_1}, \boldsymbol{z}_{s_2}, \cdots, \boldsymbol{z}_{s_n}\}$. Each $\boldsymbol{z}_s$ is passed to the next LSTM unit on the corresponding node as a hidden state vector. To get $\boldsymbol{z}_s$, for each $s' \in N_r(s)$, GAT firstly calculates the similarity score between node s' and s by

$$\alpha_{s',s} = \boldsymbol{v}^{\mathrm{T}} \tanh\left(U_1 \boldsymbol{h}_{s'} + U_2 \boldsymbol{h}_s\right), \tag{3}$$

where vector $\boldsymbol{v}$ and matrix U_1 and U_2 are the parameters that need to be learned. Then, $\widehat{\alpha}_{s',s}$ is calculated by normalizing all the $\alpha_{s',s}$ through the softmax layer:

$$\widehat{\alpha}_{s',s} = \mathrm{softmax}\left(\alpha_{s',s}\right) = \frac{\exp\left(\alpha_{s',s}\right)}{\sum_{u \in N_r(s)} \exp\left(\alpha_{u,s}\right)}. \tag{4}$$

$\widehat{\alpha}_{s',s}$ can be seen as the weight of the edge $\langle s', s\rangle$ in G_r. Finally, $\boldsymbol{z}_s$ is calculated by a weighted summation of all its neighbors' $\boldsymbol{h}$, i.e.,

$$\boldsymbol{z}_s = \sum_{s' \in N_r(s)} \widehat{\alpha}_{s',s} \boldsymbol{h}_{s'}. \tag{5}$$

4.3. Metalearning Algorithm. To solve the transfer learning problem, we propose a metalearning algorithm named MetaGAT-LSTM (given by Algorithm 1) for training the GAT-LSTM model in the target city with insufficient training data. The algorithm will build an accurate prediction model by transferring knowledge from source cities with sufficient data. It uses a modified version of Model-Agnostic Meta-Learning (MAML) [28] as the parameter learning method.

Let $\boldsymbol{T} = \{\boldsymbol{D}_{sc}^1, \boldsymbol{D}_{sc}^2, \cdots, \boldsymbol{D}_{sc}^m\}$ be the set of datasets from source cities. Define the distribution over $\boldsymbol{T}$ as $\mathscr{P}(\boldsymbol{T})$, in which the probability of choosing dataset $\boldsymbol{D} \in \boldsymbol{T}$ is

$$\Pr\left(\boldsymbol{D}\right) = \frac{|\boldsymbol{D}|}{\Sigma_{\boldsymbol{D}' \in \boldsymbol{T}} |\boldsymbol{D}'|}. \tag{6}$$

Let f_θ with parameter θ be the prediction model at the beginning of each training iteration. At first, with respect to $\mathscr{P}(\boldsymbol{T})$, we sample k datasets $\boldsymbol{D}_1, \boldsymbol{D}_2, \cdots, \boldsymbol{D}_k$ from $\boldsymbol{T}$ with replacement (Line 3 in Algorithm 1). Then, get the next training batches $\boldsymbol{B}_1, \boldsymbol{B}_2, \cdots, \boldsymbol{B}_k$ from $\boldsymbol{D}_1, \boldsymbol{D}_2, \cdots, \boldsymbol{D}_k$, respectively (Line 4 in Algorithm 1) and get the next training batch $\boldsymbol{B}_0$ from $\boldsymbol{D}_{tc}$ (Line 5 in Algorithm 1). The model's parameter is updated in two steps. In the first step (Line 6 ~ 10 in Algorithm 1), for each $\boldsymbol{B}_i\,(1 \le i \le k)$, we get the first-step adapted parameter θ_i by

$$\theta_i = \theta - \beta \nabla \mathscr{L}_{\boldsymbol{B}_i}(f_\theta). \tag{7}$$

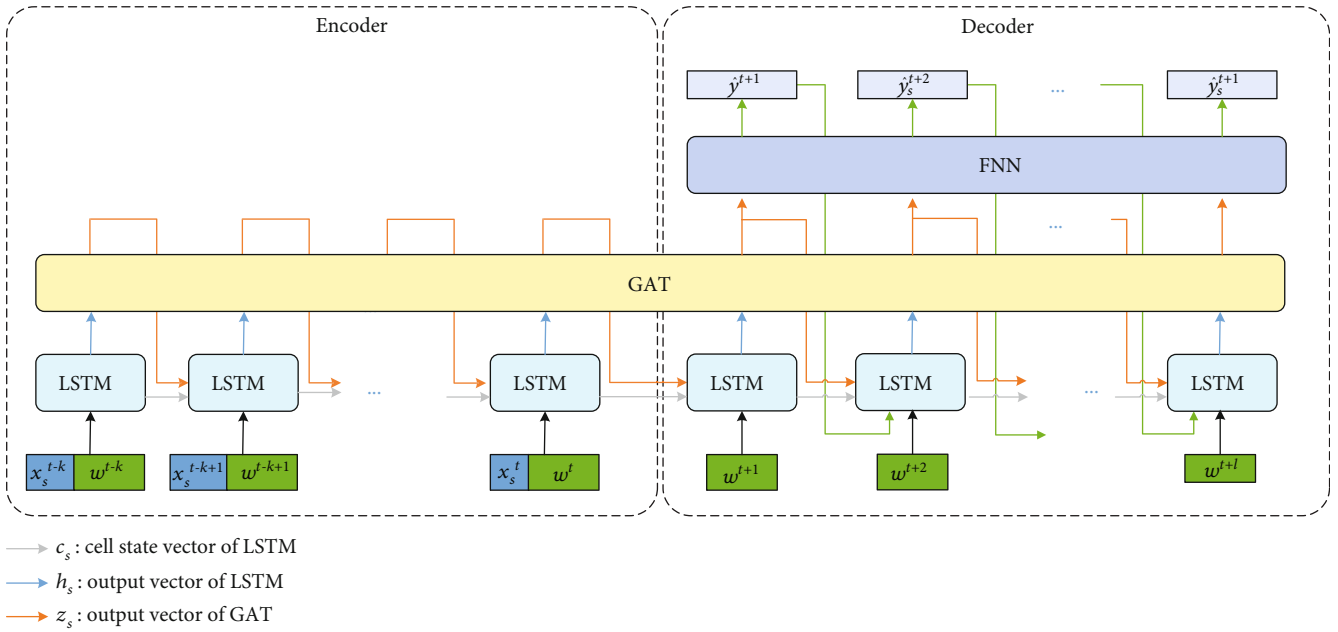

Figure 2: The air quality prediction model GAT-LSTM.

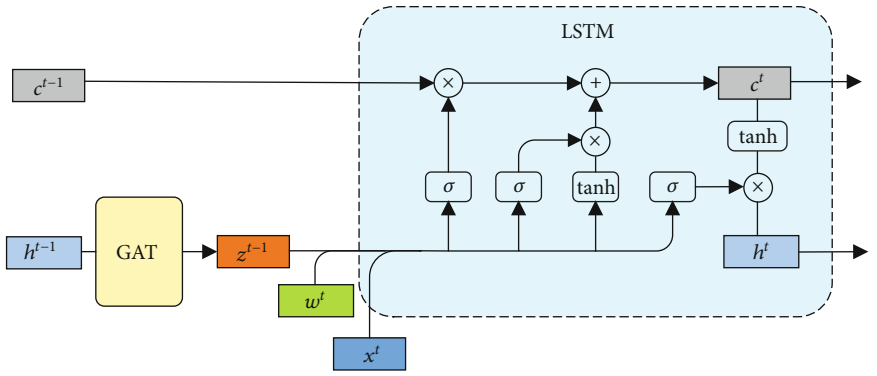

Figure 3: The LSTM unit in GAT-LSTM.

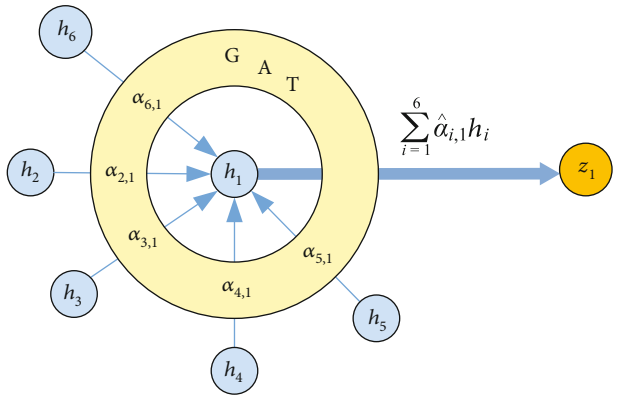

Figure 4: The GAT mechanism for node 1 (suppose that node 2 ~ 6 are 1's neighbors in G_r).

where $\mathscr{L}_{\boldsymbol{B}_i}(f_\theta)$ is the loss of the original model f_θ on training batch $\boldsymbol{B}_i$, $\nabla\mathscr{L}_{\boldsymbol{B}_i}(f_\theta)$ is gradient of $\mathscr{L}_{\boldsymbol{B}_i}(f_\theta)$ and β is the learning rate in the first step. In the second step (Line 11 in Algorithm 1), we get the second-step adapted parameter by

$$\theta = \theta - \gamma \sum_{i=1}^{k} \nabla\mathscr{L}_{\boldsymbol{B}_0}\left(f_{\theta_i}\right), \tag{8}$$

where $\mathscr{L}_{\boldsymbol{B}_0}(f_{\theta_i})$ is the loss of first-step adapted model f_{θ_i} on training batch $\boldsymbol{B}_0$ and γ is the learning rate in the second step.

5. Experimental Results and Analysis

5.1. Dataset. We use real datasets collected from four cities in China (Beijing, Tianjin, Shenzhen, Guangzhou) to verify the effectiveness and efficiency of the proposed model and meta-learning algorithm. These cities are very different in

Input: $\boldsymbol{T} = \{\boldsymbol{D}_{sc}^1, \boldsymbol{D}_{sc}^2, \cdots, \boldsymbol{D}_{sc}^m\}$: The set of the training datasets from source cities;
$\boldsymbol{D}_{tc}$: Datasets from target city;
$\mathscr{P}(\boldsymbol{T})$: Distribution over $\boldsymbol{T}$;
β, γ: Learning rates
Output: f_θ: The GAT-LSTM model for the target city
1. Randomly initialize θ
2. **While** not done **do**:
3. Sample k datasets $\boldsymbol{D}_1, \boldsymbol{D}_2, \cdots, \boldsymbol{D}_k$ from $\boldsymbol{T}$ with replacement w.r.t. $\mathscr{P}(\boldsymbol{T})$
4. Get next training batches $\boldsymbol{B}_1, \boldsymbol{B}_2, \cdots, \boldsymbol{B}_k$ from $\boldsymbol{D}_1, \boldsymbol{D}_2, \cdots, \boldsymbol{D}_k$, respectively
5. Get next training batch $\boldsymbol{B}_0$ from $\boldsymbol{D}_{tc}$
6. **For** $\boldsymbol{B}_i$ in $\{\boldsymbol{B}_1, \boldsymbol{B}_2, \cdots, \boldsymbol{B}_k\}$ **do**:
7. Calculate $\nabla \mathscr{L}_{\boldsymbol{B}_i}(f_\theta)$ with respect to $\boldsymbol{B}_i$
8. Calculate first-step adapted parameter $\theta_i \longleftarrow \theta - \beta \nabla \mathscr{L}_{\boldsymbol{B}_i}(f_\theta)$
9. Calculate $\nabla \mathscr{L}_{\boldsymbol{B}_0}(f_{\theta_i})$ with respect to $\boldsymbol{B}_0$
10. **End for**
11. Calculate second-step adapted parameter $\theta \longleftarrow \theta - \gamma \sum_{i=1}^{k} \nabla \mathscr{L}_{\boldsymbol{B}_0}(f_{\theta_i})$
12. **End while**

Algorithm 1: MetaGAT-LSTM.

geographic coordinates, city size, population density, etc., resulting in very different air quality distributions. For example, the air pollution situation in Beijing and Tianjin is much more serious than that in Shenzhen and Guangzhou. Each dataset contains the air quality data (from all monitoring stations), weather data, and weather forecast data collected from a city within one year. The period of data sampling is one hour. Taking the dataset of Beijing as an example, the air quality data contains the concentration of six major pollutants (PM2.5, PM10, SO2, NO2, O3, CO) and AQI sampled by 36 monitoring stations within one year. The weather data contains basic weather, temperature, humidity, air pressure, wind speed, and wind direction collected within one year. Weather forecast data contains the forecast value of the above weather indexes published by Beijing Meteorological Bureau. There are missing and dirty values in these datasets. In order to exploit the data as much as possible, we fill in missing values with the mean in a period of time and delete the tuples with too many consecutive missing data; Table 2 shows the details of these datasets.

5.2. Experiment Setting. There are two groups of experiments. In the first group, we compare GAT-LSTM with the most effective method to date and benchmark models. For each dataset, we divide it into train set and test set, then train these models on the same train set and evaluate their effectiveness on the same test set. The models used for comparison include the following:

(i) *ARIMA: Auto Regressive Integrated Moving Average.* ARIMA is the most common statistical model used for time series forecasting

(ii) *LSTM [32].* LSTM can learn the time dependence in time series. Compared with RNN, they can deal with longer time series and obtain better results

(iii) *ST-DNN [33].* The spatiotemporal models combing of Convolution Neural Networks (CNN) and LSTM for air quality prediction. ST-DNN is the most effective method to date

In the second group, we set one city as target city and the other cities as source cities. Then, delete most of the data from the target city and only keep a small part for training. The proposed metalearning algorithm is used to build a prediction model by transfer learning knowledge from source cities with sufficient data. We compare the proposed metalearning algorithm MetaGAT-LSTM with the following transfer learning methods:

(i) *Fine-Tuning.* First, use the data of a single city to pretrain the GAT-LSTM model and, then, fine-tune the model on the target city, which is called the single-source domain fine-tuning (Single-FT). Secondly, use the data from multiple source cities to pretrain the GAT-LSTM model then fine-tune it on the target city, which is called the multisource domain fine-tuning (Multi-FT).

(ii) *MAML [28].* Use data from all cities to jointly train the model for the target. MAML is implemented based on the metalearning method

The target pollutant is AQI (can be seen as a single pollutant). We use Root Means Square Error (RMSE), Mean Absolute Error (MAE), and ACCuracy (ACC) to evaluate models, which are defined as

$$\mathrm{RMSE} = \sqrt{\frac{1}{|\mathscr{T}|} \sum_{(\boldsymbol{X},\boldsymbol{y}) \in \mathscr{T}} \|f(\boldsymbol{X}) - \boldsymbol{y}\|_2^2}, \tag{9}$$

$$\mathrm{MAE} = \frac{1}{|\mathscr{T}|} \sum_{(\boldsymbol{X},\boldsymbol{y}) \in \mathscr{T}} \|f(\boldsymbol{X}) - \boldsymbol{y}\|_1, \tag{10}$$

Table 2: Data details.

Attribute		Number of different items			
		Beijing	Tianjin	Shenzhen	Guangzhou
Time span		2014/5/1-2015/4/30	2014/5/1-2015/4/30	2014/5/1-2015/4/30	2014/5/1-2015/4/30
AQIs		278023	189604	88139	281436
Station ID		36	27	11	42
Major pollutants	PM_{10} NO_2 CO O_3 SO_2	278023	189604	88139	281436
Historical weather	District ID	17	20	7	5
	Basic weather Temperature Pressure Humidity Wind speed Wind direction	116867	106614	30305	55632
Weather forecast	District ID	17	20	6	5
	Basic weather Wind strength Wind direction	390702	361624	106380	51870

$$\mathrm{ACC} = 1 - \frac{1}{|\mathscr{T}|} \sum_{(\boldsymbol{X},\boldsymbol{y})\in\mathscr{T}} \frac{\|f(\boldsymbol{X}) - \boldsymbol{y}\|_1}{\|\boldsymbol{y}\|_1}. \tag{11}$$

Here, $\mathscr{T}$ is the test set. $\boldsymbol{y}$ is sample's label (true monitoring data in the future) and $f(\boldsymbol{X})$ is the predicted value of $\boldsymbol{y}$. $\|\bullet\|_1$ and $\|\bullet\|_2$ are L1 and L2 norm, respectively.

In GAT-LSTM, the dimension of the GAT's output vector, the LSTM's output vector, and cell state vector are all set to 128. While training, we use dropout [34] and batch normalization [35] to strengthen the training effect. The batch size is set to 64. The number of epochs is set to 3.

5.3. Experiment Results. At first, we need to find an appropriate influence radius r for building the directed graph G_r in GAT-LSTM, so we compare the performance of GAT-LSTM with different r. Table 3 and Figure 5 give the comparison results on the dataset from Beijing. They show that when $r < 20$km, the accuracy of GAT-LSTM increases as r increases. The reason for this phenomenon is that when r is within a reasonable range, a larger r allows the model to consider more spatial correlation, thereby providing a more accurate prediction. The accuracy reaches its peak at $r = 20$ km. When $r > 20$km, the accuracy decreases slightly as r increases. This phenomenon is because too large r will cause the model to incorrectly estimate the correlation among some remote monitoring sites based on the data similarity. Thus, we set $r = 20$ km in the following experiments.

In the first group of experiments, we use the dataset from Beijing to evaluate all the prediction models. We divide the dataset into training set (70%), validation set (20%), and test set (10%). Each model takes data from the past 48 hours as input ($k + 1 = 48$), then outputs prediction values for the next l hours. Table 4 shows the experiment results with different l. The best results are marked in bold. It can be seen that the traditional linear model ARIMA does not perform well under the influence of multiple complex factors. LSTM's performance is acceptable for short-term prediction and drops quickly with the increase of l. Spatial correlation plays an important role in air quality prediction. By using CNN to extract spatial correlation among monitoring stations, the ST-DNN performs much better than ARIMA and LSTM. However, the spatial correlation built by ST-DNN cannot change dynamically with the change of weather, which reduces its predictive effects. By using GAT to dynamically model spatial correlation, GAT-LSTM gives the best performance in all cases. The performance of all models declines with the increase of l, but the decline rate of GAT-LSTM is lower than the other three, which shows that it is suitable for long-term prediction.

Table 3: The performance of GAT-LSTM with different influence radius r on Beijing dataset ($l = 6$).

r (km)	RMSE	MAE	ACC
5	39.1	24.8	0.767
10	33.3	21.9	0.778
20	28.5	20.1	0.799
25	30.3	20.5	0.797
40	29.7	20.2	0.796
60	28.8	20.5	0.798
100	29.3	20.2	0.796

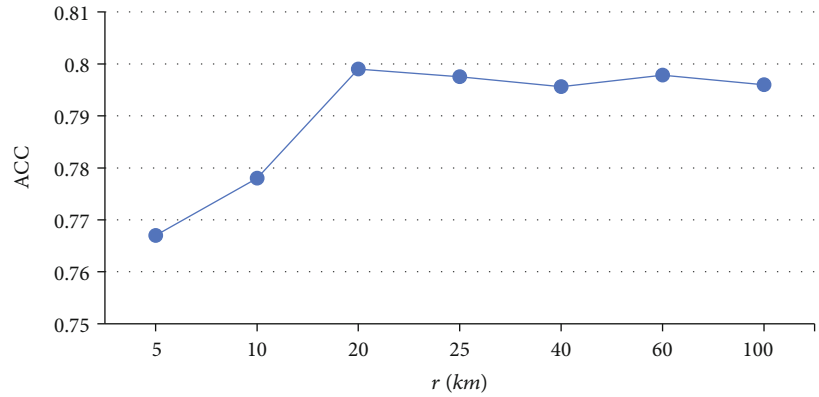

Figure 5: The relationship between influence radius r and prediction accuracy ($l = 6$).

Table 4: Comparison of mean prediction results by different methods among 36 monitoring stations in Beijing (RMSE, MAE, ACC).

Methods	+6 h			+12 h			+24 h			+48 h		
	RMSE	MAE	ACC	RMSE	MAE	ACC	RMSE	MAE	ACC	RMSE	MAE	ACC
ARIMA	53.3	40.7	0.622	70.7	59.2	0.432	94.1	68.2	0.321	104.3	78.2	0.201
LSTM	48.2	34.5	0.723	57.3	47.3	0.582	71.2	55.1	0.473	85.7	66.3	0.367
ST-DNN	35.6	23.8	0.761	52.2	37.1	0.67	63.2	49.3	0.546	70.4	57.7	0.474
GAT-LSTM	28.5	20.1	0.799	47.9	34.8	0.698	55.7	45.2	0.621	65.8	48.9	0.501

Table 5: RMSE of transfer learning methods over different sizes of training dataset in the target city. (taking Beijing as the target city and other cities as the source cities).

Methods		24 h	72 h	240 h	720 h	2400 h
Single-FT	Tianjin	62.7	62.1	53.5	50.5	45.6
	Shenzhen	71.3	70.7	67.6	61.0	55.2
	Guangzhou	70.9	69.0	66.2	59.8	54.5
Multi-FT		63.1	61.2	55.3	52.5	43.1
MAML		59.4	57.8	49.4	45.1	36.2
MetaGAT-LSTM		55.8	53.7	45.3	40.2	31.4

Table 6: RMSE of transfer learning methods over different sizes of training dataset in the target city. (taking Shenzhen as the target city and other cities as the source cities).

Methods		24 h	72 h	240 h	720 h	2400 h
Single-FT	Tianjin	77.3	76.2	72.0	66.3	58.1
	Beijing	79.2	77.2	73.1	68.9	59.2
	Guangzhou	65.6	60.3	51.9	49.3	43.7
Multi-FT		66.3	65.1	58.9	56.7	45.4
MAML		61.5	60.5	51.6	48.2	40.3
MetaGAT-LSTM		58.2	57.5	52.3	47.7	35.6

In the second group of experiments, we execute two experiments by taking Beijing and Shenzhen as the target cities, respectively. We delete most of the data from the target city and only keep a small part for training. With different sizes of the training dataset (in target city), the results of comparing MetaGAT-LSTM with other transfer learning methods are given by Tables 5 and 6. It can be seen all the methods performs better with larger training dataset. Table 5 shows that Single-FT from the Tianjin is better than that from the other two cities. Table 6 shows that Single-FT from Guangzhou is better than that from the other two cities. The climate and geographical location cause similarity

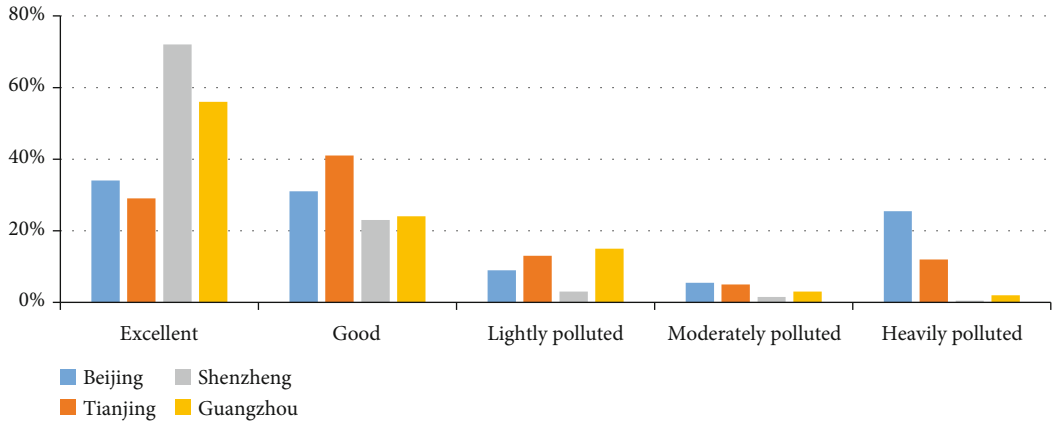

Figure 6: AQI distribution map of four cities.

of air conditions in Tianjin and Beijing, as well as the similarity of air conditions in Guangzhou and Shenzhen. This can be proven by Figure 6, in which the AQI distribution of the four cities from 2014/5/1 to 2015/4/30 is given. The more similar the two cities' datasets are, the better Single-FT performs. Multi-FT enriches the training samples by using the data from all source cities. It is better than Single-FT in some cases. However, because of simply mixing all datasets, it may cause negative migration and give an even worse performance compared with Single-FT in some cases. Both MAML and MetaGAT-LSTM are better than the fine-tuning methods. MetaGAT-LSTM outperforms MAML in all cases by more rationally integrating data from all cities for joint training.

6. Conclusions

In this paper, we propose a spatiotemporal model GAT-LSTM by combining LSTM and GAT for air quality prediction, then design a metalearning algorithm for GAT-LSTM for transfer learning. By more accurately modeling the temporal and spatial correlation of pollutants at all monitoring stations, GAT-LSTM gives a better performance compared with the up-to-date air quality prediction models. In the case of insufficient training data from the target city, the proposed metalearning algorithm for GAT-LSTM can effectively transfer knowledge from source cities with sufficient data and jointly training an accurate prediction model. A number of comparative experiments show the effectiveness of the proposed prediction model and metalearning algorithm. In the future, we may consider more factors related to air quality to improve prediction's accuracy. On the other hand, it will be reasonable to apply the proposed model and metalearning algorithm to other fields.

Data Availability

The source code implemented in this article can be obtained from a GitHub repository (https://github.com/123scarecrow/paperCode), which also includes data analysis code, data pre-processing code, and training data generation code. The data used in the experiments comes from the website: http://research.microsoft.com/apps/pubs/?id=246398.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This research was supported by the National Key R&D Program of China under Grant No. 2020YFB1710200, the National Natural Science Foundation of China under Grant No. 62072135 and 61672181, and the Fundamental Research Funds for the Central Universities under Grant No. 3072020CF0602 and 201-510318070.

References

[1] Z. Cai and X. Zheng, "A private and efficient mechanism for data uploading in smart cyber-physical systems," *IEEE Transactions on Network Science and Engineering (TNSE)*, vol. 7, no. 2, pp. 766–775, 2020.

[2] X. Fang, J. Luo, G. Luo, W. Wu, Z. Cai, and Y. Pan, "Big data transmission in industrial IoT systems with small capacitor supplying energy," *IEEE Transactions on Industrial Informatics (TII)*, vol. 15, no. 4, pp. 2360–2371, 2019.

[3] W. Cheng, Y. Shen, Y. Zhu, and L. A. Huang, "A neural attention model for urban air quality inference: learning the weights of monitoring stations," in *32nd AAAI Conference on artificial intelligence (AAAI 2018*, pp. 2151–2158, New Orleans, USA, 2018.

[4] H. P. Hsieh, S. D. Lin, and Y. Zheng, "Inferring air quality for station location recommendation based on urban big data," in *21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (ACM SIGKDD 2015)*, pp. 437–446, Sydney, Australia, 2015.

[5] L. D. Mercer, A. A. Szpiro, L. Sheppard et al., "Comparing universal kriging and land-use regression for predicting concentrations of gaseous oxides of nitrogen (NOx) for the Multi-Ethnic Study of Atherosclerosis and Air Pollution (MESA Air)," *Atmospheric Environment*, vol. 45, no. 26, pp. 4412–4420, 2011.

[6] A. Shamsoddini, M. R. Aboodi, and J. Karami, "Tehran air pollutants prediction based on random forest feature selection method," *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, vol. XLII-4/W4, pp. 483–488, 2017.

[7] Z. Cai, Z. Xiong, H. Xu, P. Wang, W. Li, and Y. Pan, "Generative adversarial networks: a survey towards private and secure applications," *Journal of the ACM*, vol. 22, no. 10, p. 111, 2020.

[8] J. Zhang, Y. Zheng, and D. Qi, "Deep spatio-temporal residual networks for citywide crowd flows prediction," in *31st AAAI Conference on artificial intelligence (AAAI 2017)*, pp. 1655–1661, San Francisco, USA, 2017.

[9] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," in *International Conference on Learning Representations (ICLR 2018)*, Vancouver, Canada, 2018.

[10] Y. Zhang, Q. Lv, D. Gao et al., "Multi-group encoder-decoder networks to fuse heterogeneous data for next-day air quality prediction," in *Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI 2019)*, pp. 4341–4347, Macao, China, 2019.

[11] Z. Cai and Z. He, "Trading private range counting over big IoT data," in *The 39th IEEE International Conference on Distributed Computing Systems (ICDCS 2019)*, vol. 2019no. 1, pp. 144–153, Dallas, USA, 2019-July.

[12] Z. Xiong, Z. Cai, D. Takabi, and W. Li, "Privacy threat and defense for federated learning with non-i.i.d. data in AIoT," *IEEE Transactions on Industrial Informatics*1.

[13] J. Pang, Y. Huang, Z. Xie, Q. Han, and Z. Cai, "Realizing the heterogeneity: a self-organized federated learning framework for IoT," *IEEE Internet of Things*, vol. 8, no. 5, pp. 3088–3098, 2021.

[14] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2009.

[15] M. Ribeiro, K. Grolinger, H. F. ElYamany, W. A. Higashino, and M. A. M. Capretz, "Transfer learning with seasonal and trend adjustment for cross-building energy forecasting," *Energy and Buildings*, vol. 165, pp. 352–363, 2018.

[16] H. I. Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P. A. Muller, "Transfer learning for time series classification," in *IEEE International Conference on Big Data (Big Data 2018)*, pp. 1367–1376, Seattle, USA, 2018.

[17] P. Xiong, Y. Zhu, Z. Sun et al., "Application of transfer learning in continuous time series for anomaly detection in commercial aircraft flight data," in *In IEEE International Conference on Smart Cloud (Smart Cloud)*, pp. 13–18, 2018.

[18] M. Lippi, M. Bertini, P. Frasconi et al., "Short-term traffic flow forecasting: an experimental comparison of time-series analysis and supervised learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 871–882, 2013.

[19] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[20] Z. Luo, J. Huang, K. Hu, X. Li, and P. Zhang, "Accu air: winning solution to air quality prediction for KDD Cup 2018," in *25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (ACM SIGKDD 2019)*, pp. 1842–1850, Anchorage, USA, 2019.

[21] Y. Zheng, F. Liu, and H. P. Hsieh, "U-air: when urban air quality inference meets big data," in *19th International Conference on Knowledge Discovery and Data Mining (ACM SIGKDD 2013)*, pp. 1436–1444, Chicago, USA, 2013.

[22] C. J. Huang and P. H. Kuo, "A deep cnn-lstm model for particulate matter (PM2.5) forecasting in smart cities," *Sensors*, vol. 18, no. 7, p. 2220, 2018.

[23] J. Bruna, W. Zaremba, A. Szlam, and Y. Le Cun, "Spectral networks and locally connected networks on graphs," in *International Conference on Learning Representations (ICLR 2014)*, Banff, Canada, 2014.

[24] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," *Advances in Neural Information Processing Systems*, vol. 29, pp. 3844–3852, 2016.

[25] C. Zhang, J. Q. James, and Y. Liu, "Spatial-temporal graph attention networks: a deep learning approach for traffic forecasting," *IEEE Access*, vol. 7, pp. 166246–166256, 2019.

[26] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations (ICLR 2015)*, San Diego, USA, 2015.

[27] Q. Hu, R. Zhang, and Y. Zhou, "Transfer learning for short-term wind speed prediction with deep neural networks," *Renewable Energy*, vol. 85, pp. 83–95, 2016.

[28] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proceedings of the 34th International Conference on Machine Learning (ICML 2017)*, vol. 70, pp. 1126–1135, 2017.

[29] J. Schmidhuber, *Evolutionary principles in self-referential learning, or on learning how to learn: the meta-meta-hook*, Technische Universität München, New York, NY, 1987.

[30] H. Yao, Y. Liu, Y. Wei, X. Tang, and Z. Li, "Learning from multiple cities: a meta-learning approach for spatial-temporal prediction," in *The World Wide Web Conference*, pp. 2181–2191, 2019.

[31] A. Vaswani, N. Shazeer, N. Parmar et al., "Attention is all you need," *In Advances in neural information processing systems*, pp. 5998–6008, 2017.

[32] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[33] 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.

[34] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[35] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 34th International Conference on Machine Learning (ICML 2015)*, vol. 37, pp. 448–456, Lille, France, 2015.