applied sciences

MDPI

Article

An Improved Hybrid Transfer Learning-Based Deep Learning Model for PM2.5 Concentration Prediction

Jianjun Ni [1,2,*], Yan Chen [1], Yu Gu [1], Xiaolong Fang [1] and Pengfei Shi [1,2,*]

[1] College of Internet of Things Engineering, Hohai University, Changzhou 213022, China; stcy401_doc@hhu.edu.cn (Y.C.); gyguyu@hhu.edu.cn (Y.G.); fangxiaolong@hhu.edu.cn (X.F.)
[2] Jiangsu Key Laboratory of Power Transmission & Distribution Equipment Technology, Hohai University, Changzhou 213022, China
* Correspondence: njjhhuc@gmail.com (J.N.); shipf@hhu.edu.cn (P.S.); Tel.: +86-519-8519-1711 (J.N.)

Abstract: With the improvement of the living standards of the residents, it is a very important and challenging task to continuously improve the accuracy of PM2.5 (particulate matter less than 2.5 μm in diameter) prediction. Deep learning-based networks, such as LSTM and CNN, have achieved good performance in recent years. However, these methods require sufficient data to train the model. The performance of these methods is limited for the sites where the data is lacking, such as the newly constructed monitoring sites. To deal with this problem, an improved deep learning model based on the hybrid transfer learning strategy is proposed for predicting PM2.5 concentration in this paper. In the proposed model, the maximum mean discrepancy (MMD) is used to select which station in the source domain is most suitable for migration to the target domain. An improved dual-stage two-phase (DSTP) model is used to extract the spatial–temporal features of the source domain and the target domain. Then the domain adversarial neural network (DANN) is used to find the domain invariant features between the source and target domains by domain adaptation. Thus, the model trained by source domain site data can be used to assist the prediction of the target site without degradation of the prediction performance due to domain drift. At last, some experiments are conducted. The experimental results show that the proposed model can effectively improve the accuracy of the PM2.5 prediction at the sites lacking data, and the proposed model outperforms most of the latest models.

Keywords: PM2.5 prediction; transfer learning; domain adversarial neural network; dual-stage two-phase model

Citation: Ni, J.; Chen, Y.; Gu, Y.; Fang, X.; Shi, P. An Improved Hybrid Transfer Learning-Based Deep Learning Model for PM2.5 Concentration Prediction. *Appl. Sci.* **2022**, *12*, 3597. https://doi.org/10.3390/app12073597

Academic Editor: Hyung-Sup Jung

Received: 2 March 2022
Accepted: 29 March 2022
Published: 1 April 2022

1. Introduction

With the rapid development of industrialization and urbanization in recent decades, the PM2.5 emissions of developing countries have increased substantially. Serious PM2.5 pollution has caused many adverse effects on economic activities. For example, as of 2015, for every 5 μg/m^3 increase in PM2.5 concentration, all other things being equal, GDP per capita will decrease by about 2500 China Yuan [1]. Therefore, how to accurately predict PM2.5 concentration becomes more and more important. The commonly used prediction methods can be divided into two categories: statistics and machine learning algorithms. Statistical methods predict air quality by applying statistics-based models, such as the autoregressive integrated moving average (ARIMA) model [2–4], multiple linear regression (MLR) model [5–7], and generalized additive model (GAM) [8–10]. However, the earlier linear models described above assume that the relationship between variables and target labels is linear, which is not suitable for nonlinear and unstable air quality prediction problems.

In order to overcome this limitation, researchers began to adopt nonlinear machine learning methods. For example, Yang et al. [11] used support vector regression (SVR) to predict the PM2.5 concentration in Beijing and verified that the accuracy of the proposed

model was better than that of other methods. Li et al. [12] proposed a stacked automatic encoder (SAE) model for air quality prediction and demonstrated that the model exhibited a better performance than linear models such as ARIMA. Feng et al. [13] used the set of back-propagation neural networks (BP) to predict daily biomass combustion pollutant emissions. Zhang et al. [14] used the genetic algorithm (GA) combined with the artificial neural network (ANN) to predict local indoor air quality with two ventilation models. Although the nonlinear machine learning methods have achieved satisfactory performance in predicting air pollution, they are unable to learn from the long-term effects of air pollution, because their models are shallow networks with few model parameters. The generalization ability of these models to complex prediction problems is limited.

To solve the problem of models with fewer parameters, people have started to use deep neural networks recently, which have been used widely in image processing, natural language understanding, and so forth [15–18]. For example, Seng et al. [19] proposed a multi-output multi-index supervised learning comprehensive prediction model (MMSL) based on long-term and short-term memory (LSTM) to predict the overall air quality in Beijing. Yan et al. [20] used the CNN-LSTM model based on spatial–temporal clustering to predict the air quality of Beijing in multi-sites. Experiments show that CNN-LSTM and LSTM generally have a better performance than the BP neural network. Feng et al. [21] proposed a method based on WRF/RNN to predict the air pollutants in Hangzhou over the next 24 h. Qin et al. [22] proposed a dual-stage attention-based recurrent neural network (DA-RNN), where the attention mechanism is used in the input stage of the encoder and decoder, so that the most relevant input features can be selected adaptively. Liu et al. [23] proposed a dual-stage two-phase attention-based recurrent neural network (DSTP-RNN), where a DSTP-based structure was used to enhance the spatial correlation of an exogenous series, and a two-stage attention mechanism was used to generate stable response weights. However, this method only uses the data of one site without considering the influence of the data of other sites on the model. To solve the above problems, in our previous work [24], an improved attention-based dual-stage two-phase fully connected (DSTP-FC) model was proposed to improve the accuracy of PM2.5 concentration prediction, where an exogenous series correlation method is used to calculate the relationship between the target series and the exogenous series, and the PM2.5 concentrations are predicted by a modified DSTP model.

Although advanced deep learning methods can get good results in air quality prediction, these deep learning-based methods all need enough historical datasets to train the models. For datasets with very little data, these methods do not provide very good prediction results. To solve the above data shortage problem, Ma et al. [25] proposed a transfer learning-based bidirectional long short term memory (TL-BiLSTM) network to predict the air quality of new stations lacking data. This method transfers the knowledge learned from the existing air quality monitoring stations to the new monitoring stations to improve the prediction accuracy of the new stations. Fong et al. [26] proposed a transfer learning model combining LSTM and RNN to predict the concentration of air pollutants. Their method inputs the data of all source domain sites into the model for pretraining, then adds the number of network layers to input the data of the target domain to train and predict the air quality of the target domain. Fang et al. [27] proposed a hybrid deep migration learning strategy based on long and short-term memory (LSTM) and domain adversarial neural networks (DANN), where the temporal features of the source and target buildings are extracted by LSTM, and DANN is used to find the domain invariant features between the source and target buildings through domain adaptation.

The above-mentioned methods have achieved satisfactory performance in the case of new site data shortages, but there are still some problems that should be further studied. For example, the temporal feature extractors of these models are all based on LSTM, which treat all input features equally and fail to pay attention to the important features. The TL-BiLSTM model is a single-site migration, and for the case where the source domain has multiple sites, it is not known which site in the source domain is selected for migration. The

LSTM-RNN model inputs the data of all source domain sites into the model for pre-training, which is unsuitable when the number of source sites is large, because this method will input a lot of redundant data, resulting in the over-fitting and calculation problems [28].

To deal with these problems above, an improved hybrid transfer learning-based deep learning model is proposed in this paper for PM2.5 concentration prediction. When the amount of data in the target domain is small, the model cannot be well trained only by using the data in the target domain. If the transfer learning-based method is used, the model trained on the source domain data is not applicable to the target domain data, when the source and target domain data have different distributions. Thus, the motivation of this study is to use the domain adaptive migration learning method to find the domain invariant characteristics between the source domain and the target domain, and to use the data of the source domain and the target domain to predict the PM2.5 concentration in the target domain with fewer data.

The main contributions of this paper are summarized as follows: (1) An improved hybrid transfer learning model with a dual-stage two-phase model (DSTP) and a domain adversarial neural network (DANN) is proposed; (2) The maximum mean discrepancy (MMD) is introduced into the air quality prediction based on transfer learning, which is used to select which station in the source domain is most suitable for migration to the target domain; (3) An improved dual-stage two-phase (DSTP) model is used to extract the spatial–temporal features of the source domain and the target domain. Various experiments on several cities in China are conducted, and the results verify the efficiency and the generalization ability of the proposed method.

This paper is organized as follows: Section 2 describes the proposed method and presents the structure of the proposed deep learning-based model; Section 3 presents the experiments and results; Section 4 discusses the performance of different feature extractors, the generalization ability and the robustness of the proposed method, and the setting of hyperparameters; Section 5 provides the conclusion and possible future research directions.

2. Proposed Model

In this paper, a hybrid transfer learning model is proposed. The input of the model includes historical air quality and meteorological data of source and target domains. Firstly, the source domain site selection method based on MMD is used to find the source domain site closest to the target domain. Then the data of the two sites are input into the improved DSTP model together. A feature extractor based on the DSTP model is used to extract the spatial–temporal features of training data from source and target site data. The obtained spatial–temporal features are input into the domain classification model and the regression prediction model, respectively. In this paper, a domain adversarial neural network (DANN) is used to find the domain invariant features between the source domain and the target domain through adversarial domain adaptation of DSTP feature extractor and domain classifier. Finally, the regression prediction model based on the fully connected layer is used to predict the values of the source and target sites. The test data of the target site are input into the pre-trained DSTP-DANN model for PM2.5 concentration prediction. The framework of the proposed model is shown in Figure 1, which will be described in detail below.

Remark 1. *The method presented in this paper is different from those that fine-tune by freezing the first few layers of the model. This method uses the domain adaptation of an adversarial neural network to conduct transfer learning. DANN combines domain adaptation and feature learning in a training process, so that the features of domain invariance can be predicted. Then, the proposed transfer learning-based model trained by source domain site data can be used to assist in predicting target site data without degradation of the prediction performance due to domain drift.*

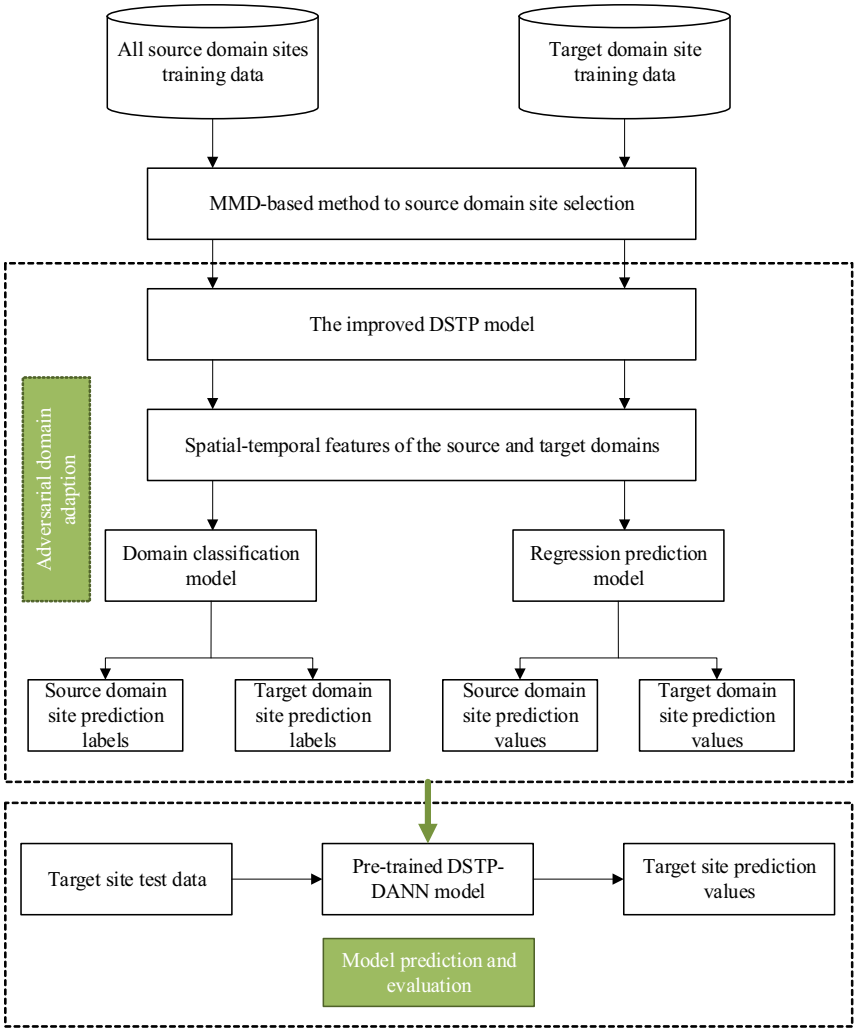

Figure 1. Framework of the proposed model.

2.1. Site Selection for Source Domain Based on MMD

Because there are many source domain sites, it is necessary to measure the distribution distance between source domain sites and target domain sites, and select the source domain sites closest to the target domain sites. Recent studies have proved that the maximum mean discrepancy (MMD) in the regenerative kernel Hilbert space is an effective method for estimating the distance between two distributions [29]. Based on two distributed samples, the average difference between two samples corresponding to f can be obtained by subtracting the function mean of different samples, and MMD is the maximum value of the average difference. For the convenience of calculation, the square form of MMD is generally adopted. The process of using MMD to estimate the difference between two domains is as follows.

The source domain site data in a given source domain is denoted as:

$$D_s = (x_1, x_2, \ldots, x_n), \tag{1}$$

where x represents the source domain site data and n represents the source domain site data number. The target site data in the target domain is denoted as:

$$D_t = (z_1, z_2, \ldots, z_m), \tag{2}$$

where z represents the target domain site data and m represents the target domain site data number. The nonlinear mapping function in the Hilbert space of the regenerative kernel is denoted as ϕ. Then the squared form of MMD is defined as follows:

$$\mathrm{MMD}_H^2 = \| \frac{1}{n} \sum_{i=1}^{n} \phi(x_i) - \frac{1}{m} \sum_{i=1}^{m} \phi(z_i) \|^2 . \quad (3)$$

The difference in distribution between two domains is the distance between the two data distributions. The smaller the MMD value, the closer the two domains are. Currently, MMD has been widely used in transfer learning algorithms [30–32]. The proposed method is used to select the source domain site that is most suitable for migration to the target domain site by calculating the similarity between the source domain and the target domain based on MMD.

2.2. Spatial-Temporal Features Extraction Based on DSTP

The center site is the site to be predicted, and the best matching site of the center site is determined by the exogenous series correlation method. The main reason to use this DSTP model is that a stable attention weight can be obtained by the DSTP model, which uses a dual-stage attention mechanism in the encoder stage. Thus, temporal and spatial features can be extracted simultaneously [24].

Given all sites' data, each site contains n exogenous series and a target series (series to be predicted). Within the window size S_W of the central site collection, the k-th exogenous series is represented by:

$$x^k = (x_1^k, x_2^k, \ldots, x_{S_W}^k)^{\mathrm{T}} \in \mathbb{R}^{S_W}. \quad (4)$$

All exogenous series within window size S_W are represented by:

$$X = (x_1, x_2, \ldots, x_{S_W})^{\mathrm{T}} \in \mathbb{R}^{n \times S_W}. \quad (5)$$

The target series is represented by:

$$Y = (y_1, y_2, \ldots, y_{S_W})^{\mathrm{T}} \in \mathbb{R}^{S_W}. \quad (6)$$

In this study, the encoder adopts a two-stage attention mechanism, which aims to study the spatial correlation between the exogenous series of the central site collection, its matching sites' exogenous series and target series. Specifically, the spatial correlation between the exogenous series of the central site collection and the exogenous series of the matching sites is studied in the first stage of attention. In the second stage of attention, the weighted features are studied again, that is, the spatial correlation among the exogenous series of the central site collection, its target series and matching sites' target series. Thus, the two-stage spatial mechanism ensures that the learned spatial correlations are stable. The decoder is a temporal attention mechanism designed to learn the temporal correlation among the encoder hidden state, the target series of the central site collection, and the target series of the matching site.

2.2.1. First Stage of Attention

The data from the central site and its matching sites are input into the model together, which can be used to study the exogenous series relationships between them and can improve the accuracy of predicting PM2.5 concentrations. The exogenous series correlation method is used to find matching sites. Given the k-th feature x^k of the central site collection at time t, the k-th feature $x_k^{(best)}$ of the exogenous series of the best matching site can be obtained by the exogenous series correlation method [24]. The spatial correlation between

the exogenous attributes of the learning central site collection and the matching site in the input attention mechanism is:

$$f_t^k = v_f^{\mathrm{T}} tanh\Big(W_f\Big[h_{t-1}^f : s_{t-1}^f\Big] + U_f x^k + M_f x_k^{(best)}\Big), \tag{7}$$

where $[* : *]$ is a concatenation operation, and $v_f \in \mathbb{R}^{S_W}$, $W_f \in \mathbb{R}^{S_W \times 2m}$, $U_f, M_f \in \mathbb{R}^{S_W \times S_W}$ are the parameters to learn; $h_{t-1}^f \in \mathbb{R}^m$ and $s_{t-1}^f \in \mathbb{R}^m$ are the hidden state and unit state of the encoder LSTM unit at the previous time. After f_t^k is calculated, the Softmax function is used to normalize to get the attention weight α_t^k. α_t^k is determined by h_{t-1}^f, s_{t-1}^f, the k-th feature x^k of the current input, and the k-th feature $x_k^{(best)}$ of the $best$-th matching site, which measures the importance of the k-th feature at time t. $\widetilde{x_t}$ is the combination of all features at moment t, which is defined as follows:

$$\widetilde{x_t} = \left(\alpha_t^1 x_t^1, \alpha_t^2 x_t^2, \ldots, \alpha_t^n x_t^n\right)^{\mathrm{T}}. \tag{8}$$

Then, the hidden states h_{t-1}^f and $\widetilde{x_t}$ are input into the LSTM layer to update the hidden state of the current moment, and $\widetilde{x_t}$ is input into the attention of the second stage.

2.2.2. Second Stage of Attention

This module aims to learn the spatial correlation between the exogenous series and the target series of the central site collection and the target series of the matching sites. The specific method is to combine the target series of the central site collection with the exogenous series of the corresponding time and add the target series of the best matching site. The attention weights for the input attention mechanism are as follows:

$$s_t^k = v_s^{\mathrm{T}} tanh(W_s\left[h_{t-1}^s : s_{t-1}^s\right] + U_s\left[\widetilde{x}^k : y^k\right] + M_s y_k^{(best)}), \tag{9}$$

where $v_s \in \mathbb{R}^{S_W}$, $W_s \in \mathbb{R}^{S_W \times 2q}$, $U_s \in \mathbb{R}^{S_W \times S_W}$, $M_s \in \mathbb{R}^{S_W}$ are the parameters to be learned. $h_{t-1}^s \in \mathbb{R}^q$ and $s_{t-1}^s \in \mathbb{R}^q$ are the hidden state and unit state of the encoder LSTM unit at the previous time; and q is the hidden size in the second attention module.

After s_t^k is calculated, it is normalized by Softmax function to get β_t^k. The corresponding target variable y^k is connected to the k-th attribute $\widetilde{x}^k$ to form a new vector z^k, namely:

$$z^k = \left[\widetilde{x}^k : y^k\right] \in \mathbb{R}^{(n+1) \times S_W}. \tag{10}$$

Note that the weight β_t^k measures the importance of z^k at the moment t, and any attribute value at any time has its corresponding weight:

$$\widetilde{z_t} = \left(\beta_t^1 z_t^1, \beta_t^2 z_t^2, \ldots, \beta_t^{n+1} z_t^{n+1}\right)^{\mathrm{T}}. \tag{11}$$

Then, h_{t-1}^s and $\widetilde{z_t}$ are input into the LSTM layer to update the hidden state h_t^s at the current moment, and h_t^s is input into the temporal attention stage.

2.2.3. Decoder with Temporal Attention

The decoder with temporal attention can adaptively select the encoder hidden state most relevant to the target series by weighting the encoder hidden state. The encoder with spatial attention outputs the hidden state, and the decoder learns the temporal relations of the hidden state through the attention mechanism within a window size S_W. Based on the hidden state $h_{t-1}^d \in \mathbb{R}^p$ and unit state $s_{t-1}^d \in \mathbb{R}^p$ of the decoder LSTM unit at the previous time, the attention weight of each encoder hidden state in the attention module at the moment t can be calculated. The attention weights for the temporal attention mechanism are as follows:

$$d_t^i = v_d^{\mathrm{T}} tanh\left(W_d\left[h_{t-1}^d : s_{t-1}^d\right] + U_d h_i^s + b_d\right), \tag{12}$$

where $v_d, b_d \in \mathbb{R}^p, W_d \in \mathbb{R}^{q\times 2p}, U_d \in \mathbb{R}^{p\times p}$ are parameters to learn; p is the hidden size of the third attention module, and $h_i^s \in H^s$ is the i-th encoder hidden state of the second attention module. After d_t^i is calculated, it is normalized by Softmax function to get γ_t^i. The context vector c_t is defined as follows:

$$c_t = \sum_{j=1}^{S_W} \gamma_t^j h_j^s. \tag{13}$$

The temporal relationship between all the hidden state of the central site collection and the target series of matching sites is again learned by concatenating the target series of matching sites:

$$\widetilde{y}_{t-1} = \widetilde{W}^{\mathrm{T}}[y_{t-1} : c_{t-1}] + \widetilde{b} + \widetilde{H}^{\mathrm{T}} y_{t-1}^{(best)}, \tag{14}$$

where $\widetilde{W}^{\mathrm{T}} \in \mathbb{R}^{q+1}$ and $\widetilde{b}$, $\widetilde{H}^{\mathrm{T}} \in \mathbb{R}$ are the parameters that map the connection to the size of the hidden state of the decoder. Then, $\widetilde{y}_{t-1}$ and h_{t-1}^d are input into the LSTM layer to update the hidden state h_t^d at the current moment. The final multi-step prediction formula is as follows:

$$y_{S_W+1}, \ldots, y_{S_W+\tau} = v_y^{\mathrm{T}}\left(W_y\left[h_t^d : c_t\right] + b_y\right) + b_y', \tag{15}$$

where $W_y \in \mathbb{R}^{p\times(p+q)}$ and $b_y \in \mathbb{R}^p$ are parameters that map concatenation to the size of the decoder hidden state; $\left[h_t^d : c_t\right] \in \mathbb{R}^{p+q}$ represents the concatenation of the decoder hidden state and the context vector; $v_y \in \mathbb{R}^{\tau\times p}$ is the weight and $b_y' \in \mathbb{R}^\tau$ is the deviation, where τ is the time steps to predict in the future. The linear function produces the final prediction result.

2.3. DSTP-DANN Based on Transfer Learning

Figure 2 shows the proposed DSTP-DANN structure based on transfer learning (defined as TL-DSTP-DANN). The TL-DSTP-DANN structure consists of three main components: feature extractor, regression predictor, and domain classifier. The feature extractor is based on the improved DSTP model (see Section 2.2), and the regression predictor and domain classifier are both fully connected layers.

The training optimization loss of the TL-DSTP-DANN model includes regression loss and domain classification loss. The regression loss for PM2.5 prediction is defined as the mean squared error:

$$L_y^i(\theta_f, \theta_y) = \frac{1}{n}\sum_{i=1}^{n}(y_i - y_i')^2, \tag{16}$$

where n is the batch size of training data; y_i and y_i' represent the actual and predicted values of PM2.5, respectively. The loss for domain label classification is defined as the dichotomous cross-entropy:

$$L_d^i(\theta_f, \theta_d) = \frac{1}{n}\sum_{i=1}^{n} l_i \log\frac{1}{l_i'} + \frac{1}{n}\sum_{i=1}^{n}(1-l_i)\log\frac{1}{1-l_i'}, \tag{17}$$

where l_i and l_i' represent the actual domain label and the prediction domain label, respectively. In this study, we set the source domain label to 0 and the target domain label to 1.

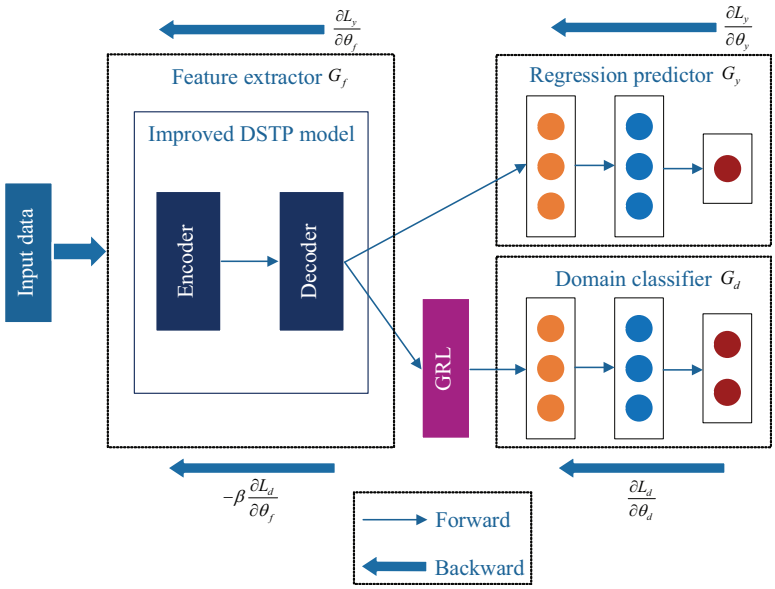

Figure 2. The proposed TL-DSTP-DANN structure. (1) Feature extractor: The feature extractor G_f is used to extract the temporal and spatial features of the input time-series data. (2) Regression predictor: The purpose of the regression predictor G_y is to find the mapping between the extracted spatial–temporal features and PM2.5 concentrations based on the source and the target domain data. (3) Domain classifier: The domain classifier G_d distinguishes whether the extracted features come from the source or target domain.

In the training process, to obtain domain invariant features, the distribution of two features is as similar as possible. The parameter θ_f of feature mapping is found to maximize the loss of the domain classifier, and at the same time, the parameter θ_d of the domain classifier is found to minimize the loss of the domain classifier. The minimum and maximum change between losses cannot be directly realized by gradient update in the back-propagation process of neural networks. The difference between these two losses is achieved by inserting a gradient reversal layer (GRL) between the feature extractor and the domain classifier.

In this paper, DANN is used to search for domain-invariant features between source domain and target domain through the domain adaptation of DSTP feature extractor and domain classifier. The main reason for using the DANN is that it can combine domain adaptation and feature learning in a training process, so that the parameters learned can be directly applied to the target domain without reducing its prediction accuracy due to the domain deviation [27].

The idea of this paper is very similar to the generative adversarial networks (GANs). The generating model *G*: Equivalent to a feature extractor, the goal is to make the domain classifier not correctly identify the domain labels (the two feature distributions should be as similar as possible). The discriminant model *D*: Determine whether a label is the label of the target domain, and the target is to distinguish whether the extracted features come from the source domain or the target domain. GANs is implemented by the competition between *G* and *D*.

During the training process, the two models *G* and *D* can be enhanced simultaneously by competing with each other. Because of the existence of discriminant model *D*, *G* can learn the similar features of the two distributions well without a lot of prior knowledge and prior distribution, and finally make the data generated by the model achieve the effect of faking the truth (that is, *D* cannot distinguish whether the features extracted by *G* come from the source domain or the target domain, so *G* and *D* reach a certain Nash equilibrium [33]).

In the proposed TL-DSTP-DANN model, GRL acts as a constant transform during forward-propagation, gaining gradients at the latter level and changing its sign during

backward propagation. In particular, GRL can be regarded as a pseudo function R_β, and the following equations are its forward and backward propagation processes:

$$R_\beta(x) = x \tag{18}$$

$$\frac{dR_\beta}{dx} = -\beta Q, \tag{19}$$

where Q is a unit matrix; β is a positive hyperparameter, which realizes the trade-off between regression loss and domain classification loss, and the setting of β refers to [34]. Because the difference between the regression loss and the domain classification loss is relatively large, the model loss is the sum of the regression loss and the domain classification loss. The GRL layer is followed by the domain classifier, and a hyperparameter is set in the GRL layer to achieve a balance between two loss functions.

In this paper, the source domain site data is denoted as $D_s = \left(\left[x_1, y_1^s\right], \left[x_2, y_2^s\right], \ldots, \left[x_n, y_n^s\right]\right)$, where x_i and y_i^s represent the source domain site's exogenous data and target data, respectively, n represents the source domain data number. The target domain site data are denoted as $D_t = \left(\left[z_1, y_1^t\right], \left[z_2, y_2^t\right], \ldots, \left[z_m, y_m^t\right]\right)$, where z_i and y_i^t represent the target domain site's exogenous data and target data, respectively, m represents the target domain data number. The expression of the final objective "pseudo-function" is:

$$L(\theta_f, \theta_y, \theta_d) = A + B + C + D \tag{20}$$

$$A = \frac{1}{n}\sum_{i=1}^{n} L_y\left(G_y\left(G_f\left(x_i; \theta_f\right); \theta_y\right), y_i^s\right) \tag{21}$$

$$B = \frac{1}{n}\sum_{i=1}^{n} L_d\left(G_d\left(R_\beta\left(G_f\left(x_i; \theta_f\right)\right); \theta_d\right), l_i\right) \tag{22}$$

$$C = \frac{1}{m}\sum_{i=1}^{m} L_y\left(G_y\left(G_f\left(x_i; \theta_f\right); \theta_y\right), y_i^t\right) \tag{23}$$

$$D = \frac{1}{m}\sum_{i=1}^{m} L_d\left(G_d\left(R_\beta\left(G_f\left(z_i; \theta_f\right)\right); \theta_d\right), l_i\right), \tag{24}$$

where θ_f, θ_y, θ_d denote the network connection weights of the feature extractor, regression predictor and domain classifier, respectively; G_f, G_y, G_d represent the feature extractor, regression predictor and domain classifier, respectively. The gradient descent method is used to update the learning weights in the TL-DSTP-DANN model, which is expressed as follows:

$$\theta_f \leftarrow \theta_f - \mu\left(\frac{\partial L_y^i}{\partial \theta_f} - \beta\frac{\partial L_d^i}{\partial \theta_f}\right) \tag{25}$$

$$\theta_y \leftarrow \theta_y - \mu\frac{\partial L_y^i}{\partial \theta_y} \tag{26}$$

$$\theta_d \leftarrow \theta_d - \mu\frac{\partial L_d^i}{\partial \theta_d}, \tag{27}$$

where μ represents the learning rate. The pseudo-code of the proposed TL-DSTP-DANN training process is shown in Algorithm 1.

Algorithm 1 TL-DSTP-DANN model training process.

Input: source domain site data D_s, target domain site data D_t
Output: Parameters of the model θ_f,θ_y,θ_d

for $i = 1 \rightarrow n$ **do**
  Forward:
    Calculate the regression loss $L_y^i(\theta_f, \theta_y)$ by Equation (16)
    Calculate the loss of domain label classification $L_d^i(\theta_f, \theta_d)$ by Equation (17)
    Calculate the loss of "pseudo-function" $L\left(\theta_f, \theta_y, \theta_d\right)$ by Equation (20)
  Backward:
    Calculate gradient $\frac{\partial L\left(\theta_f, \theta_y, \theta_d\right)}{\partial \theta}$
  Update:
    Update the network weight parameter θ by Equations (25)–(27)
end for
return $\theta_f, \theta_y, \theta_d$

3. Experiments

3.1. Experiment Setting and Data Source

The dataset used in this paper was collected by Microsoft Research's Urban Air project [35]. We select datasets related to Beijing and Tianjin to evaluate the proposed TL-DSTP-DANN neural network. The distribution of monitoring sites in Beijing and Tianjin is shown in Figure 3, where the red points are all the sites in Beijing, and the black point is the Tianjin site to be predicted. For reasons of data collection, the historical data of the Tianjin site are only one month, and the historical data for the Beijing sites are from 1 May 2014 to 30 April 2015 (1 year, 8759 data). Because the data of Tianjin site are very few, the training effect is not good if they are directly used. This paper investigates how to use data migration learning from all sites in Beijing to predict PM2.5 concentrations at the Tianjin site. The dataset collects air quality records from 36 sites in Beijing and 1 site in Tianjin. Each air quality record contains six pollutants: PM2.5, PM10, SO_2, NO_2, CO and O_3. Each weather record contains seven items: time, weather, temperature, pressure, humidity, wind speed, and wind direction. Table 1 shows the details of the datasets used in this paper. There are very complex relationships among these factors [28]. It is the main reason why the deep learning-based method is used in this study.

Table 1. Dataset summary.

Attribute	Source Site	Target Site
District	Beijing of China	Tianjin of China
Time Range	1 May 2014–30 April 2015	30 March 2015–30 April 2015
Number of Sample	8759	745
Number of Site	36	1
Standard Deviation	81.23	49.78
Mean Value	84.73	76.66

The first 75% of the Tianjin site is selected as the training data, and the remaining 25% as the test data. For data from the Beijing stations, because PM2.5 data from individual stations are highly correlated, consecutive missing values greater than one row are filled in using the IDW interpolation [36] method based on the PM2.5 concentrations at adjacent stations. If the consecutive missing values are less than two rows, the linear interpolation method is used. For the data from the Tianjin site, linear interpolation is used directly.

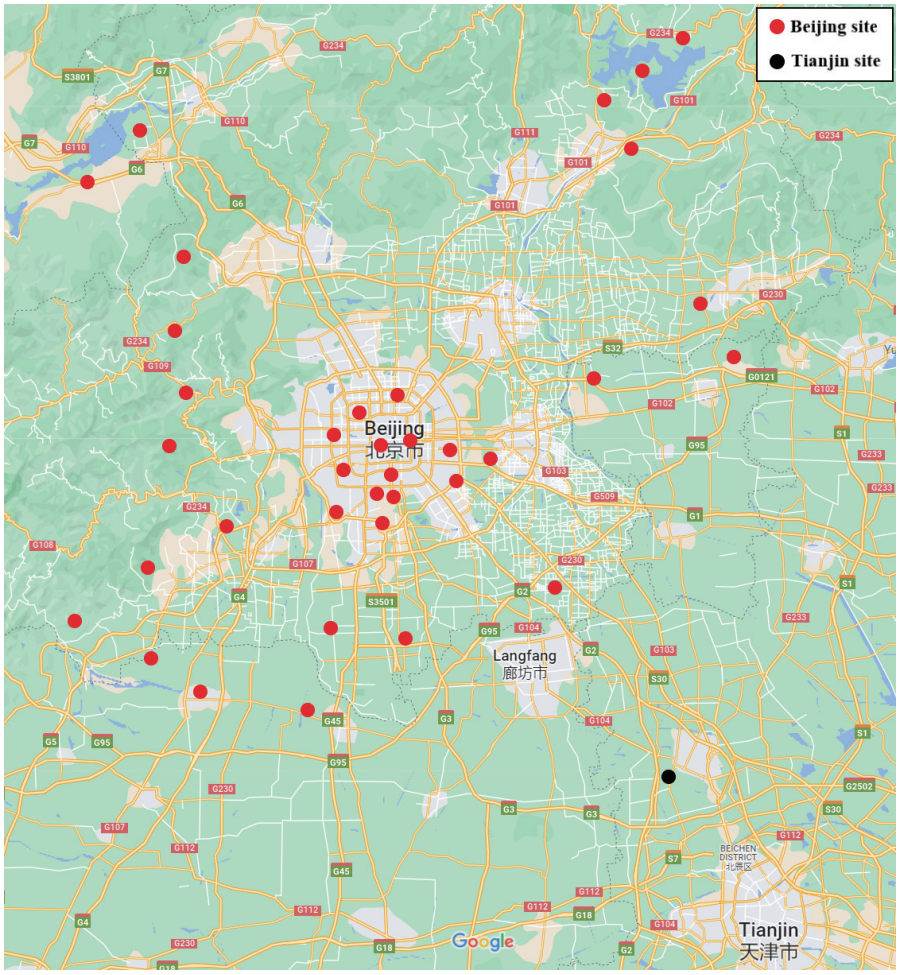

Figure 3. Distribution of monitoring sites in Beijing and Tianjin, the red points are all the sites in Beijing, and the black point is the Tianjin site to be predicted.

For proving the effectiveness of the proposed method for stations with fewer data, the TL-DSTP-DANN network is used to model the data and predict the PM2.5 concentration for the 3rd hour in the future. Appropriate hyperparameters are set to the model to produce the best performance. The prediction time step is set to 8. In order to use all source domain data and target domain training data, the batch sizes are set to 275 and 18 for the source and target domain data, respectively. A back-propagation algorithm is used to train all models, with regression losses for PM2.5 prediction as MSE loss functions and losses for domain label classification defined as dichotomous cross-entropy loss functions. During training, small-batch stochastic gradient descent is used combined with the Adam optimizer, setting the upper limit of the training period to 120, and the learning rate to 0.001. Each attention module uses a layer of LSTM network, where the hidden state of LSTM network is set to the same, that is, $m = p = q = 128$. In order to improve the prediction accuracy, we use the minimum-maximum normalization method given in the formula for normalization:

$$x = \frac{x - min}{max - min} \tag{28}$$

For evaluating the effectiveness of the method, three metrics are used in the experiment, including the root mean square error (RMSE), the mean absolute error (MAE), and the mean absolute percentage error (MAPE). These three metrics are often used to evaluate the performance of the deep learning-based prediction methods [25,36], which are defined as follows:

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - y_i')^2} \tag{29}$$

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - y_i'| \tag{30}$$

$$\mathrm{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - y_i'|}{y_i}, \tag{31}$$

where y_i is the true value and y_i' is the predicted value. The smaller the value of these three indicators, the higher the prediction accuracy and the better the performance of the model.

3.2. Model Comparison

In this paper, we use some state-of-the-art models to test the superiority of the proposed model (TL-DSTP-DANN). In these comparison experiments, the hyperparameters of the compared models are set as following principle: For the models that provided the hyperparameters, we use the original hyperparameters directly. Otherwise, we adjust the hyperparameters to achieve the best performance. The compared models are introduced as follows.

TL-LSTM: LSTM is used to learn from the long-term dependence of PM2.5, and transfer learning is applied to transfer the features of the source domain to the target domain.

TL-BiLSTM [25]: the TL-BiLSTM model is proposed to predict the air quality of new stations lacking data. This approach uses data from existing sites to pre-train the Stacked BiLSTM model. Then, freeze the first few hidden layers of the basic model, and fine-tune the remaining hidden layers using the data of the newly-built sites.

TL-DSTP [24]: This method uses an improved DSTP model to transfer source domain site data to assist in predicting PM2.5 concentrations at target domain sites.

The results of the proposed model compared to the baselines are shown in Table 2. As can be seen from the results in Table 2, the combined results of TL-BiLSTM outperform TL-LSTM, indicating that the accuracy of transfer learning experiments using the Stacked BiLSTM model is better than that of LSTM. Meanwhile, the combined results of the TL-DSTP model outperform the TL-BiLSTM, indicating that transfer learning using the improved DSTP model is better than the BiLSTM. Compared with the model TL-DSTP, the proposed TL-DSTP-DANN model reduces 13.09%, 8.90%, and 13.04% in MAE, RMSE, and MAPE, respectively. The results show that the proposed model have good prediction performance for the PM2.5 concentrations at target domain sites with less data.

Table 2. Results of comparison between the proposed method and baselines.

Methods	MAE	RMSE	MAPE
TL-LSTM	21.80	26.26	0.43
TL-BiLSTM	17.40	20.94	0.33
TL-DSTP	13.52	17.53	0.23
TL-DSTP-DANN	11.75	15.97	0.20

To verify the performance of the proposed method, the true and predicted values of PM2.5 concentrations at the Tianjin site predicted by the four models from 23 April 2015 to 30 April 2015 are shown in Figure 4. As can be seen from the figure, compared with the TL-DSTP and TL-DSTP-DANN models, the predicted values of TL-LSTM and TL-BiLSTM models are quite different from the true values. At the same time, the error between the predicted value and the true value of the TL-DSTP-DANN model is smaller than that of TL-DSTP.

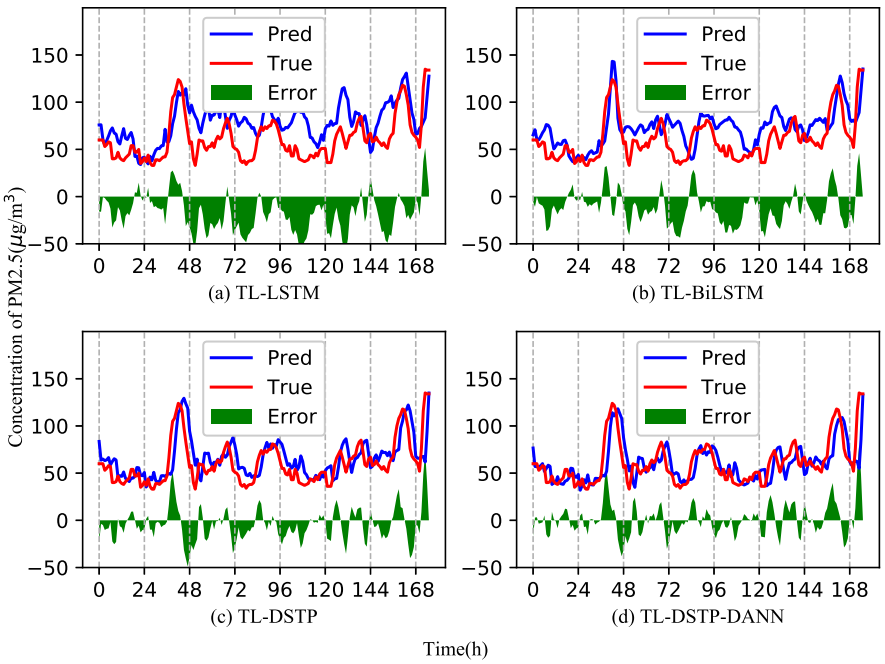

Figure 4. Performance comparison of four models for predicting PM2.5 concentration at Tianjin site.

4. Discussion

4.1. Comparison of Different Feature Extractors

The TL-DSTP-DANN structure proposed in this study uses the DSTP model as the feature extractor. CNN, LSTM, and BiLSTM are then used as feature extractors in combination with the DANN module, and the regression predictor and domain classifier are left unchanged to verify that the DSTP model is the most accurate as a feature extractor. In the TL-CNN-DANN model, the convolution layer and the pooling layer are used to extract features of the input time series. The LSTM model is used as the feature extractor in the TL-LSTM-DANN model, and the BiLSTM model is used as the feature extractor in the TL-BiLSTM-DANN model. The experimental accuracy of four different feature extractors combined with the DANN module is shown in Table 3.

Table 3. Accuracy comparison of different feature extractors combined with DANN for PM2.5 prediction.

Methods	MAE	RMSE	MAPE
TL-CNN-DANN	19.68	23.05	0.39
TL-LSTM-DANN	14.70	18.51	0.27
TL-BiLSTM-DANN	13.36	16.55	0.25
TL-DSTP-DANN	11.75	15.97	0.20

The results show that the TL-CNN-DANN model is the worst because CNN is more suitable for extracting spatial features. The TL-LSTM-DANN model performs better than TL-CNN-DANN because the LSTM model can better extract temporal features of long time series than the CNN module. The TL-BiLSTM-DANN model performs better than the TL-LSTM-DANN because the BiLSTM considers the information contained in subsequent time series to adjust modeling and computation. The TL-DSTP-DANN model has the best performance compared to other DANN-based structures, mainly because DSTP can extract spatial–temporal features. The models using different feature extractors to predict PM2.5 concentrations at the Tianjin site are shown in Figure 5, from which it can be seen that the TL-DSTP-DANN model has the smallest error between the predicted and true values.

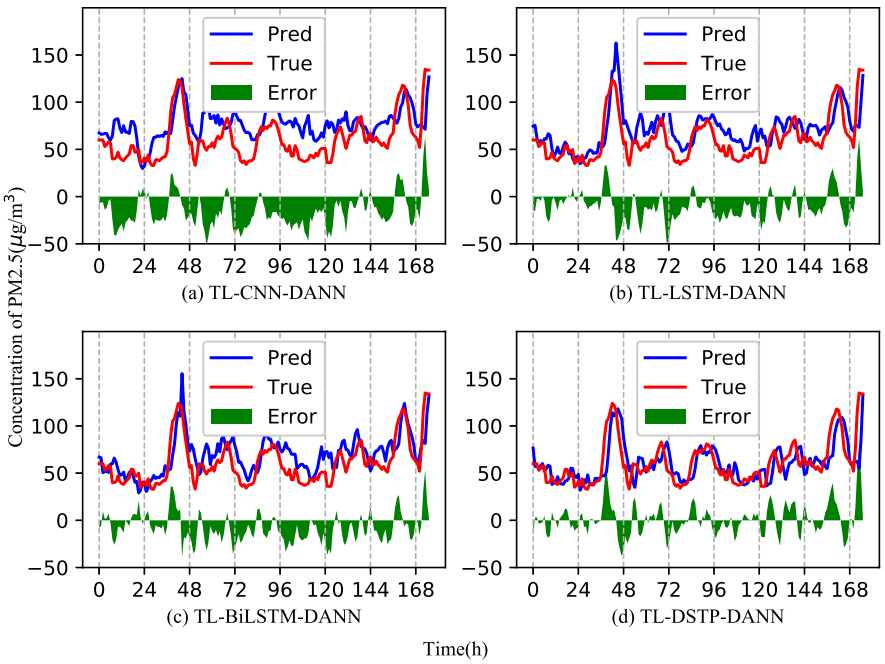

Figure 5. Prediction of PM2.5 concentrations at Tianjin site using models with different feature extractors.

4.2. Test of Generalization Ability for Different Regional Sites

The previous experiment was to transfer data from Beijing to Tianjin. Because Tianjin and Beijing are very close, the effect of transfer learning is not universal. Cities farther away from Beijing are now selected as the target domains to verify the generality of the proposed method. In this study, the PM2.5 concentration in Guangzhou was predicted by transfer learning from the Beijing sites data. The details of the dataset in Guangzhou are as follows: the time range and the number of the samples are the same as those of Tianjin site. The standard deviation and mean value of the dataset in Guangzhou are 22.48 and 37.02, respectively. To facilitate comparison, we intercepted the Guangzhou site at the same point in time as the previous target domain. Seventy-five percent of the target domain data are used for training, and the remaining 25% of the samples are used for testing. The interpolation of missing values is the same as that in Section 3.1. The experimental results of using different models to predict PM2.5 concentrations in Guangzhou are shown in Table 4.

Table 4. Performance of different models in predicting PM2.5 concentrations in Guangzhou.

Methods	MAE	RMSE	MAPE
TL-LSTM	18.68	24.72	0.38
TL-BiLSTM	16.13	20.99	0.30
TL-DSTP	13.45	19.89	0.21
TL-DSTP-DANN	12.20	17.84	0.20

The results in Table 4 show that the indicators of TL-DSTP-DANN are the best, indicating that the proposed method has good generalization ability. The true and predicted values of PM2.5 concentrations at the Guangzhou site predicted using different models are shown in Figure 6. It can be seen from the figure that compared with the TL-LSTM and TL-DSTP-DANN models, the predicted and true values of the TL-BiLSTM and TL-DSTP models in 24–72 h are quite different. However, the predicted results of the TL-LSTM model in 120–168 h are not ideal, while TL-DSTP-DANN is relatively ideal for predicting PM2.5 concentration in Guangzhou. Therefore, the comprehensive performance of the TL-DSTP-DANN model is the best.

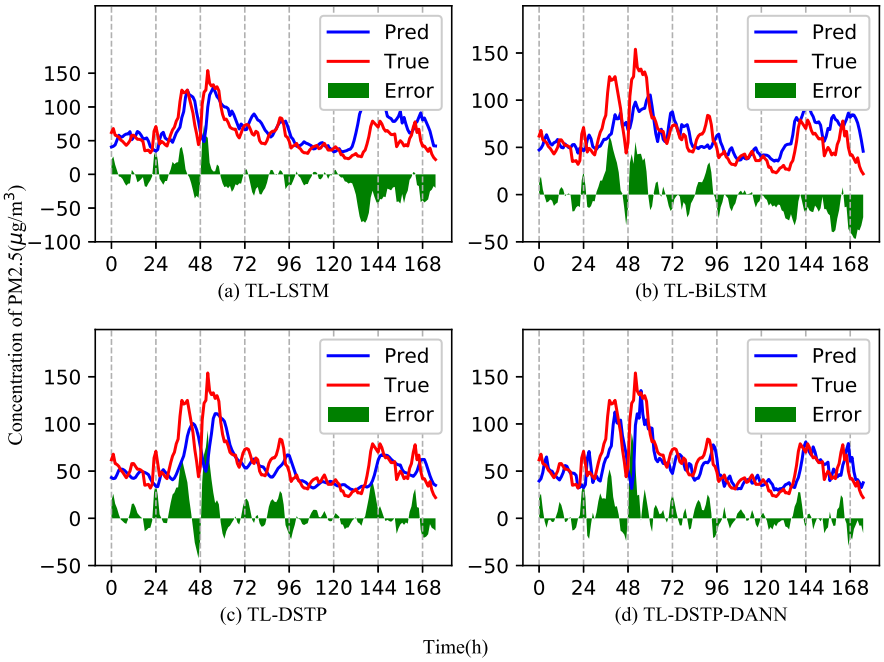

Figure 6. Prediction of real and predicted PM2.5 concentrations at Guangzhou station using different models.

4.3. Cross Validation Experiment

Cross-validation or Monte Carlo simulation method can be used to evaluate the robustness of the model [37] . In this study, a five-fold cross-validation experiment is conducted to further test the robustness of the proposed method (see [38] for details). Since the data are too few, the early stopping strategy is adopted to prevent overfitting, and the maximum batch of early stopping is 30. In this cross validation experiment, the target dataset is divided into six subsets. Then, the first subset is used to predict the second subset, the first two subsets are used as the training set to predict the third subset, and so on in a similar fashion. The average value of the results of all subsets is used as the final evaluation. The experimental results of the cross validation are as follows: MAE = 13.67, RMSE = 17.98, and MAPE = 0.22. The results show that the proposed method can achieve good prediction results, when the training data are too few, which mean that the proposed model has good robustness.

4.4. Setting of Hyperparameters

The main hyperparameters in the proposed model are the batch size of source domain data and target domain data, time step and hidden state size of LSTM. Most of the hyperparameters can refer to our previous work [24] and the related literature [27]. Here, just the time step is discussed, which is the hyperparameter closely related to the proposed model.

Reasonable setting of the time step has a great influence on experimental accuracy and speed. The larger the value of time step is, the more the characteristics of the sample it contains. However, the influence of past data on the current PM2.5 concentration will become weaker and weaker with the increase of the time step. If the time step is too large, it will lead to the reduction of the experimental accuracy. At the same time, the time of experimental training will increase with the increase of the time step. To get an appropriate setting of the time step, some experiments are conducted. The experimental results corresponding to different time steps are shown in Figure 7. It can be seen from the figure that the comprehensive performance of the proposed model is the best when the time step is 8. Thus, the time step is set as 8 in this study.

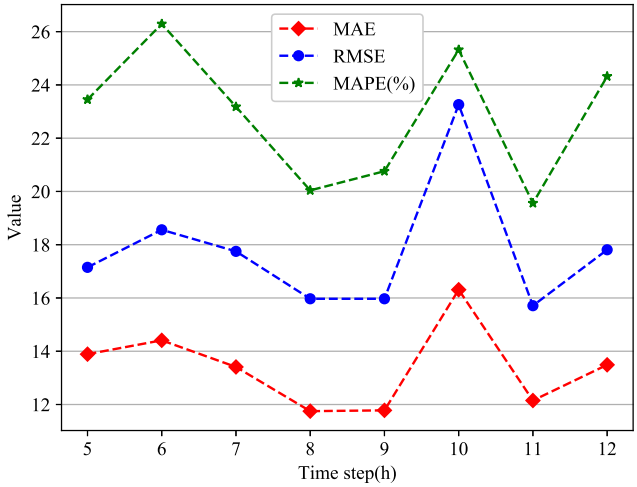

Figure 7. The experimental results corresponding to different time steps.

5. Conclusions and Future Work

In this paper, a dual-stage two-phase model and an adversarial domain adaptation hybrid transfer learning strategy are proposed to predict PM2.5 concentration, especially for new sites with relatively little historical data. Firstly, the maximum mean discrepancy (MMD) is introduced into the proposed model to select the most suitable source domain site. Then, inputting data from the source domain and the target domain together into an improved DSTP model, the DSTP model extracts the spatial–temporal characteristics of both. DANN finds domain invariant features between source domain and target domain by fusing extracted spatial–temporal features. Finally, the PM2.5 concentration in the target domain is predicted by a regression predictor. To evaluate the performance of the proposed model, we use air quality data from the Beijing sites to assist in predicting PM2.5 concentrations at the Tianjin and Guangzhou sites. The main experimental results are as follows: (1) Compared with other transfer learning prediction models (including TL-LSTM, TL-BiLSTM, TL-DSTP), the proposed TL-DSTP-DANN model decreases by more than 8.5% in MAE, RMSE and MAPE; (2) Transfer learning can obviously improve the performance of PM2.5 prediction in newly built monitoring stations with insufficient data; (3) The comprehensive experimental results of the improved DSTP model combined with DANN are better than those of CNN, LSTM, and BiLSTM. Compared with the BiLSTM, the MAE of the improved DSTP model decreases by 12.05%, and the MAPE decreases by 20%.

In our future work: (1) The current dataset contains only historical air pollutant concentrations and meteorological data, lacking relatively important geographical data, but geographical factors have an impact on PM2.5 concentrations. In the future, the proposed method could provide higher prediction accuracy if datasets containing geographical information are available; (2) An algorithm needs to be investigated to determine which of the multiple source domain data are most suitable for transfer learning to the target domain; (3) Whether the method proposed in this paper can be used to predict other air pollutants, such as O_3 and SO_2.

Author Contributions: Funding acquisition, J.N.; Project administration, J.N. and P.S.; Writing—original draft, Y.C.; Writing—review and editing, X.F. and Y.G. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (61873086), and the Science and Technology Support Program of Changzhou (CE20215022).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Publicly available datasets were analyzed in this study. This data can be found here: https://www.microsoft.com/en-us/research/project/urban-air, accessed on 1 December 2021.

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

Nomenclature

MMD	maximum mean discrepancy
DSTP	dual-stage two-phase
TL	transfer learning
DANN	domain adversarial neural network
CNN	convolutional neural network
LSTM	long and short-term memory
BiLSTM	bidirectional long short term memory
GANs	generative adversarial networks
GRL	gradient reversal layer
G_f	feature extractor
G_y	regression predictor
G_d	domain classifier
θ_f	weights of the feature extractor
θ_y	weights of the regression predictor
θ_d	weights of the domain classifier
D_s	source domain
D_t	target domain
RMSE	root mean square error
MAE	mean absolute error
MAPE	mean absolute percentage error

References

1. Hao, Y.; Peng, H.; Temulun, T.; Liu, L.Q.; Mao, J.; Lu, Z.N.; Chen, H. How harmful is air pollution to economic development? New evidence from PM2.5 concentrations of Chinese cities. *J. Clean. Prod.* **2018**, *172*, 743–757. [CrossRef]
2. Aladağ, E. Forecasting of particulate matter with a hybrid ARIMA model based on wavelet transformation and seasonal adjustment. *Urban Clim.* **2021**, *39*, 100930. [CrossRef]
3. Zafra, C.; Ángel, Y.; Torres, E. ARIMA analysis of the effect of land surface coverage on PM10 concentrations in a high-altitude megacity. *Atmos. Pollut. Res.* **2017**, *8*, 660–668. [CrossRef]
4. Zhang, L.; Lin, J.; Qiu, R.; Hu, X.; Zhang, H.; Chen, Q.; Wang, J. Trend analysis and forecast of PM2.5 in Fuzhou, China using the ARIMA model. *Ecol. Indic.* **2018**, *95*, 702–710. [CrossRef]
5. Shams, S.R.; Jahani, A.; Kalantary, S.; Moeinaddini, M.; Khorasani, N. The evaluation on artificial neural networks (ANN) and multiple linear regressions (MLR) models for predicting SO_2 concentration. *Urban Clim.* **2021**, *37*, 100837. [CrossRef]
6. Cakir, S.; Sita, M. Evaluating the performance of ANN in predicting the concentrations of ambient air pollutants in Nicosia. *Atmos. Pollut. Res.* **2020**, *11*, 2327–2334. [CrossRef]
7. Chen, L.; Zhu, J.; Liao, H.; Yang, Y.; Yue, X. Meteorological influences on PM2.5 and O_3 trends and associated health burden since China's clean air actions. *Sci. Total. Environ.* **2020**, *744*, 140837. [CrossRef]
8. Ma, Y.; Ma, B.; Jiao, H.; Zhang, Y.; Xin, J.; Yu, Z. An analysis of the effects of weather and air pollution on tropospheric ozone using a generalized additive model in Western China: Lanzhou, Gansu. *Atmos. Environ.* **2020**, *224*, 117342. [CrossRef]
9. Ravindra, K.; Rattan, P.; Mor, S.; Aggarwal, A.N. Generalized additive models: Building evidence of air pollution, climate change and human health. *Environ. Int.* **2019**, *132*, 104987. [CrossRef]
10. Feng, Y.; Yang, Q.; Tong, X.; Chen, L. Evaluating land ecological security and examining its relationships with driving factors using GIS and generalized additive model. *Sci. Total. Environ.* **2018**, *633*, 1469–1479. [CrossRef]
11. Yang, W.; Deng, M.; Xu, F.; Wang, H. Prediction of hourly PM2.5 using a space-time support vector regression model. *Atmos. Environ.* **2018**, *181*, 12–19. [CrossRef]
12. Li, X.; Peng, L.; Hu, Y.; Shao, J.; Chi, T. Deep learning architecture for air quality predictions. *Environ. Sci. Pollut. Res.* **2016**, *23*, 22408–22417. [CrossRef] [PubMed]
13. Feng, X.; Fu, T.M.; Cao, H.; Tian, H.; Fan, Q.; Chen, X. Neural network predictions of pollutant emissions from open burning of crop residues: Application to air quality forecasts in southern China. *Atmos. Environ.* **2019**, *204*, 22–31. [CrossRef]
14. Zhang, T.; Li, X.; Zhao, Q.; Rao, Y. Control of a novel synthetical index for the local indoor air quality by the artificial neural network and genetic algorithm. *Sustain. Cities Soc.* **2019**, *51*, 101714. [CrossRef]

15. Mutabazi, E.; Ni, J.; Tang, G.; Cao, W. A Review on Medical Textual Question Answering Systems Based on Deep Learning Approaches. *Appl. Sci.* **2021**, *11*, 5456. [CrossRef]
16. Ni, J.; Chen, Y.; Chen, Y.; Zhu, J.; Ali, D.; Cao, W. A Survey on Theories and Applications for Self-Driving Cars Based on Deep Learning Methods. *Appl. Sci.* **2020**, *10*, 2749. [CrossRef]
17. Tng, S.S.; Le, N.Q.K.; Yeh, H.Y.; Chua, M.C.H. Improved Prediction Model of Protein Lysine Crotonylation Sites Using Bidirectional Recurrent Neural Networks. *J. Proteome Res.* **2022**, *21*, 265–273. [CrossRef]
18. Le, N.Q.K.; Nguyen, B.P. Prediction of FMN Binding Sites in Electron Transport Chains Based on 2-D CNN and PSSM Profiles. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2021**, *18*, 2189–2197. [CrossRef]
19. Seng, D.; Zhang, Q.; Zhang, X.; Chen, G.; Chen, X. Spatiotemporal prediction of air quality based on LSTM neural network. *Alex. Eng. J.* **2021**, *60*, 2021–2032. [CrossRef]
20. Yan, R.and Liao, J.; Yang, J.; Sun, W.; Nong, M.; Li, F. Multi-hour and multi-site air quality index forecasting in Beijing using CNN, LSTM, CNN-LSTM, and spatiotemporal clustering. *Expert Syst. Appl.* **2021**, *169*, 114513. [CrossRef]
21. Feng, R.; Zheng, H.J.; Gao, H.; Zhang, A.R.; Huang, C.; Zhang, J.X.; Fan, J.R. Recurrent Neural Network and random forest for analysis and accurate forecast of atmospheric pollutants: a case study in Hangzhou, China. *J. Clean. Prod.* **2019**, *231*, 1005–1015. [CrossRef]
22. Qin, Y.; Song, D.; Cheng, H.; Cheng, W.; Jiang, G.; Cottrell, G.W. A dual-stage attention-based recurrent neural network for time series prediction. In Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17, Melbourne, Australia, 19–25 August 2017; pp. 2627–2633.
23. Liu, Y.; Gong, C.; Yang, L.; Chen, Y. DSTP-RNN: A dual-stage two-phase attention-based recurrent neural network for long-term and multivariate time series prediction. *Expert Syst. Appl.* **2020**, *143*, 113082. [CrossRef]
24. Shi, P.; Fang, X.; Ni, J.; Zhu, J. An Improved Attention-Based Integrated Deep Neural Network for PM2.5 Concentration Prediction. *Appl. Sci.* **2021**, *11*, 4001. [CrossRef]
25. Ma, J.; Li, Z.; Cheng, J.C.; Ding, Y.; Lin, C.; Xu, Z. Air quality prediction at new stations using spatially transferred bi-directional long short-term memory network. *Sci. Total. Environ.* **2020**, *705*, 135771. [CrossRef] [PubMed]
26. Fong, I.H.; Li, T.; Fong, S.; Wong, R.K.; Tallon-Ballesteros, A.J. Predicting concentration levels of air pollutants by transfer learning and recurrent neural network. *Knowl.-Based Syst.* **2020**, *192*, 105622. [CrossRef]
27. Fang, X.; Gong, G.; Li, G.; Chun, L.; Li, W.; Peng, P. A hybrid deep transfer learning strategy for short term cross-building energy prediction. *Energy* **2021**, *215*, 119208. [CrossRef]
28. Zhao, J.; Deng, F.; Cai, Y.; Chen, J. Long short-term memory-Fully connected (LSTM-FC) neural network for PM2.5 concentration prediction. *Chemosphere* **2019**, *220*, 486–492. [CrossRef]
29. Li, J.; Wu, W.; Xue, D.; Gao, P. Multi-source deep transfer neural network algorithm. *Sensors* **2019**, *19*, 3992. [CrossRef]
30. Yang, B.; Lei, Y.; Jia, F.; Li, N.; Du, Z. A polynomial kernel induced distance metric to improve deep transfer learning for fault diagnosis of machines. *IEEE Trans. Ind. Electron.* **2019**, *67*, 9747–9757. [CrossRef]
31. Tzeng, E.; Hoffman, J.; Darrell, T.; Saenko, K. Simultaneous deep transfer across domains and tasks. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 11–18 December 2015; pp. 4068–4076.
32. Han, T.; Liu, C.; Yang, W.; Jiang, D. Deep transfer network with joint distribution adaptation: A new intelligent fault diagnosis framework for industry application. *ISA Trans.* **2020**, *97*, 269–281. [CrossRef]
33. Ayturan, Y.A.; Ayturan, Z.C.; Altun, H.O. Air Pollution Modelling with Deep Learning: A Review. *Int. J. Environ. Pollut. Environ. Model.* **2018**, *1*, 58–62.
34. Ganin, Y.; Lempitsky, V. Unsupervised domain adaptation by backpropagation. In Proceedings of the 32nd International Conference on Machine Learning, Lile, France, 6–11 July 2015; Volume 2, pp. 1180–1189.
35. Zheng, Y.; Yi, X.; Li, M.; Li, R.; Shan, Z.; Chang, E.; Li, T. Forecasting fine-grained air quality based on big data. In Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining, Sydney, Australia, 10–13 August 2015; pp. 2267–2276.
36. Ma, J.; Cheng, J.C.; Lin, C.; Tan, Y.; Zhang, J. Improving air quality prediction accuracy at larger temporal resolutions using deep learning and transfer learning techniques. *Atmos. Environ.* **2019**, *214*, 116885. [CrossRef]
37. Bokde, N.D.; Yaseen, Z.M.; Andersen, G.B. ForecastTB—An R Package as a Test-Bench for Time Series Forecasting—Application of Wind Speed and Solar Radiation Modeling. *Energies* **2020**, *13*, 2578. [CrossRef]
38. Yang, M.; Fan, H.; Zhao, K. PM2.5 Prediction with a Novel Multi-Step-Ahead Forecasting Model Based on Dynamic Wind Field Distance. *Int. J. Environ. Res. Public Health* **2019**, *16*, 4482. [CrossRef] [PubMed]