A spatiotemporal graph neural network with multi granularity for air quality prediction[1]*

Haibin Liao[1,†], Li Yuan[1,†], Mou Wu[2,*,†]

[1.] *School of Electronic and Electrical Engineering, Wuhan Textile University, 430200, Wuhan, China;*

[2.] *School of Computer Science and Technology, Hubei University of Science and Technology, 437100, Xianning, China*

Abstract

Air quality prediction is a complex system engineering. How to fully consider the impact of meteorological, spatial and temporal factors on air quality is the core problem. To address this central conundrum, in an elaborate encoder-decoder architecture, we propose a new air quality prediction method based on multi-granularity spatiotemporal graph network. At the encoder, firstly, we use multi granularity graph and the well-known HYSPLIT model to build spatial relationship and dynamic edge relationship between nodes, respectively, while meteorological, temporal and topographic characteristics are used to build node features and LSTM (Long Short Term Memory) is used to learn the time-series relationship of pollutant concentration. At the decoder, secondly, we use the attention mechanism LSTM for decoding and forecasting of pollutant concentration. The proposed model is capable of tracking different influences on prediction resulting from the changes of air quality. On a project-based dataset, we validate the effectiveness of the proposed model and examine its abilities of capturing both fine-grained and long-term influences in pollutant process. We also compare the proposed model with the state-of-the-art air quality forecasting methods on the dataset of Yangtze River Delta city group, the experimental results show the appealing performance of our model over competitive baselines.

Keywords

Air quality forecasting, spatiotemporal data, graph neural network, long short term memory

*STRL'24: Third International Workshop on Spatio-Temporal Reasoning and Learning, 5 August 2024, Jeju, South Korea

1* Corresponding author.

† These authors contributed equally.

liao_haibing@163.com (H.Liao); 2022048@wtu.edu.cn (L. Yuan); mou.wu@163.com (M. Wu)

1. INTRODUCTION

Air quality which is closely related to human public health, has been a common research hotspot focused by scholars all over the world. At present, many air quality monitoring stations (stations for short) have been built in major cities to monitor the concentration of air pollutants (PM2.5, PM10, O3, etc.) and meteorological parameters (temperature, pressure, wind speed, wind direction, humidity, etc.). However, these stations only can monitor real-time air quality, and fail to provide air quality prediction (AQP) and auxiliary support for urban intelligent decision-making or activity planning. How to construct an AQP model using a large amount of historical monitoring data has become a hot research topic in the field of environment engineering. Unfortunately, AQP is an extremely complex system engineering. On the one hand, air quality is related to pollutant emission, which is a type of time sequence and has periodicity; On the other hand, there exist physical and chemical changes of pollutants in the air, such as diffusion and deposition, which are greatly affected by meteorological and geographical locations; Finally, air quality also has certain probability, such as unexpected pollution leakage events will lead to a sharp decline in air quality.

The commonly used types of AQP include mechanism model (MM) and machine learning (ML) methods. The MM method [1-6], also known as numerical model, uses atmospheric physical and chemical reactions to model the emission and diffusion process of air pollutants, and then carries out AQP. For example, Gaussian diffusion models of AERMOD and ADMS, Lagrange models of CALPUFF and HYSPLIT can be applied to small-scale and medium AQP [1]; and the third-generation air quality models [2] such as CMAQ, CAMX, WRF-CHEM, NAQPMS, and so on can be applied to predict large-scale air quality. However, most of MM methods require many empirical parameters and assumptions, which are prone to be reliable for a specific environment but not for all urban environments [6]. For example, AERMOD is an empirical model which is mainly applicable to small-scale air diffusion simulation and pollutant forecasting. And the third-generation air quality model needs comprehensive and accurate source list and meteorological field data as input to predict, and its application is limited.

With the development of deep learning, AQP methods based on ML have attracted more attention [6, 7, 15, 16]. The AQP method based on ML takes advantage of large historical observation data for training and testing, finding out the change law of pollutant concentration, and then predicts the air quality, which include linear statistical models [8], fitting optimization techniques [9], and deep learning methods [10]. Since deep learning has a powerful function to automatically extract nonlinear features, recent literatures about AQP often rely on deep learning models. Zhang et al [12] proposed a deep learning AQP method combining CNN (Convolutional Neural Network) and LSTM (Long Short Term Memory), which achieved good results and made scholars see the dawn of the application of deep learning in AQP. Subsequently, Du et al. [13] used one-dimensional CNN to capture local time trend and bidirectional LSTM to extract long-term time series features, and then to construct a hybrid neural network for AQP. Liang et al. [14] proposed a GeoMAN based on LSTM and encoder-decoder architecture for AQP, and took advantage of the attention mechanism to capture the spatial impact relationship among different stations. The above methods captured temporal correlation well by LSTM, but their capture of spatial relationship was obviously insufficient. Although CNN can be used to establish spatial relationships, it is a static spatial relationship, and the distance among stations is fixed. Due to the influence of weather and terrain, the spatial relationship among stations is not a simple static distance relationship, but a dynamic. In addition, the stations in the city are unevenly distributed and sparse, so interpolation is required in the construction of CNN, resulting in many virtual stations, which will affect the forecasting results.

In contrast, graph-based models naturally sidestep the above issue since they shape the concentration values into graph nodes and keep their original distributions in graph structures. Because graph can construct non-Euclidean entity distribution, it can capture spatial relationships well. Thus, to compensate for the lack of spatial relationship learning in the above methods, the methods based on GNN (Graph Neural Network) are applied to AQP. Qi et al. [17] used GNN to learn the spatial relationship among stations, and LSTM to learn the time correlation of stations, so as to build a comprehensive forecasting model GC-LSTM. Lin et al. [18] used diffusion convolution operation to replace matrix multiplication in GRU (Gate Recurrent Unit) for sequence modeling, and combined with graph convolution operation to build GC-DCRNN for AQP. Xu et al. [19] proposed ST-MFGCN for AQP. Its main innovation is to obtain the spatiotemporal variation law

of vehicle emissions by building a graph structure traffic network, and then predict traffic pollution emissions. Xu et al [20] proposed HighAir, i.e., a hierarchical graph neural network-based AQP method, which adopted an encoder-decoder architecture and considered complex air quality influencing factors, e.g., weather and land usage.

The above methods based on GNN use graph structure to effectively construct the spatial relationship among stations, but they fail to fully construct the edges in the graph structure. For example, literature [20] simply used distance and wind direction similarity to construct edge weights. In order to better build the edge relationship, Wang et al [21] identified a set of critical domain knowledge for PM2.5 forecasting and developed a novel graph based model PM2.5-GNN，they used domain knowledge（wind speed and direction，distance， advection coefficient）to construct edge weights. Although they make use of domain knowledge, they simply list some impact factors, which is not enough. The pollution impact relationship among stations is comprehensively determined by meteorological conditions and landform. It is a complex process and needs to be analyzed by using professional models.

In this paper, we use LSTM to learn the cycle and season of pollutant concentration, and use GNN to learn the spatial relationship of pollutants among stations, and thus building a multi granularity spatiotemporal graph neural network model, called MGST−GNN. In order to better capture the mutation of pollutant concentration and its impact on air diffusion and deposition, the mechanism model HYSLPLIT is used to dynamically construct the adjacency matrix and edge relation of the spatiotemporal graph.

The main contributions of this paper include:

•We innovatively propose a dynamic spatiotemporal graph model combining mechanism model and graph neural network. The adjacency matrix and edge weight vector of dynamic graph are constructed based on the simulation results of diffusion, transport and deposition of polluted air mass by mechanism model, so that the architecture learns the spatial influence relationship among multi granularity stations.

•We propose to add time characteristic attributes of quarter, month, week, hour and holiday to each node in the encoder, use LSTM based on attention mechanism for temporal learning in the decoder to enhance MGST-GNN.

2. PRELIMINARY

2.1. Related concepts and definition

Definition 1 Region and Stations: we set $R=\{r_a,1\le a\le N\}$ as a set of N regions , $L=\{l_a,1\le a\le N\}$ as a location set of N regions , $S_a=\{s_{a,i},1\le i\le|S_a|\}$ as a set of stations in region r_a , $\mathcal{L}_a=\{l_{a,i},1\le i\le|S_a|\}$ as a location set of stations in region r_a . Where, $l_{a,i}$ is composed of longitude and latitude of a station, and l_a is the mean value of $l_{a,i}$ in $\mathcal{L}_a$.

Definition 2 Time Feature (TF): TF includes five features: quarter, month, week, hour and holiday. We represent the quarter by using one-hot encoding adopted for 4-bit binary representation, as shown in Table 1.

Table.1

Quarter feature representing method

	Spring	summer	fall	winter
Spring season	1	0	0	0
summertime	0	1	0	0
autumn	0	0	1	0
wintertime	0	0	0	1

Similarly, the month is represented with 12-bit binary by one-hot encoding; week is represented with 7-bit binary by one-hot encoding; Hour is represented with 24-bit binary by one-hot coding; Holiday is represented with 2-bit binary by one-hot coding (0 means non-holidays, 1 means holiday).

Definition 3 Geomorphic Feature (GF): GF contains the topography and land usage information of a station, which consider the altitude and five land usage categories: residential area, park, mountain, water (river or pool), and industry. Where, the altitude is divided into four categories: very high (more than 1300m), high (1000-1300m), medium (500-1000m) and low (less than 500m), and the corresponding categories

are represented by the numbers 1, 2, 3 and 4; The land usage type is determined by the number of major land use types within 10km around the station. For example, the altitude of $s_{a,i}$ is 800 meters, and there is one residential area, two park, one mountain, three pool, and two industrial facilities within the perception radius 10 kilometers of $s_{a,i}$. Thus, the GF vector $gf_{a,i}$ can be represented as [3, 1, 2, 1, 3, 2].

Definition 4 Weather Data (WD): The WD of region r_a and stations $s_{a,i}$ at time slot t is represented as a vector wd_a^t and $wd_{a,i}^t$, including temperature, humidity, rainfall, wind speed, wind direction, and air pressure.

Definition 5 Pollutant Concentration Data: The pollutant concentration data includes the concentrations of six major pollutants such as PM2.5, PM10, SO2, CO2, CO and O3. Among them, PM2.5 and O3 are the most concerned at present. Therefore, the later experiments focus on the concentration forecasting of these two pollutants.

Definition 6 HYSPLIT: HYSPLIT [5, 22] is a complete system for computing simple air parcel trajectories, as well as complex transport, dispersion, chemical transformation, and deposition simulations. A common application is a back trajectory analysis to determine the origin of air masses and establish source-receptor relationships. HYSPLIT has also been used in a variety of simulations describing the atmospheric transport, dispersion, and deposition of pollutants and hazardous materials.

In this paper, HYSPLIT is used to establish the source-receptor relationship among nodes by trajectory analysis. When HYSPLIT is used for trajectory analysis, it only needs to input the meteorological data of the simulation area and the coordinate information of the initial point of the simulation. The meteorological data can be downloaded from the official website of Air Resources Laboratory (ARL) [2] for free. Figure 1 describes an example of trajectory analysis using HYSPLIT. The example takes Beijing Center (116 ° 20 ', 39 ° 56') as the starting point and 10:00 on May 6, 2022 as the starting time to predict the air mass trajectory in the next 48 hours. It can be seen from the figure that the location and time of each track can be obtained by HYSPLIT.

2 https://nomads.ncep.noaa.gov/pub/data/nccf/com/hysplit

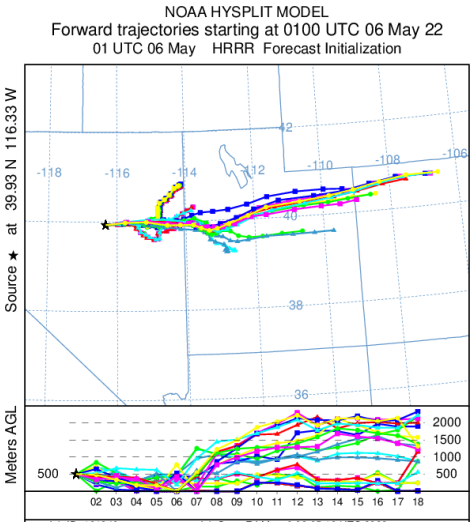


Figure.1: trajectory analysis chart (Different colored lines in the figure represent trajectories of different heights.)

2.2. Problem statement

AQP task: Given region locations L, station locations $\mathcal{L}$, geomorphic feature $gf_{a,i}$, τ_{in} hours of pollutant concentration data con, $\tau_{in}+\tau_{out}$ hours of weather data wd, and the AQP task aims to forecast the pollutant concentrations of stations for the next τ_{out} hours, where τ_{in} denotes the length of historical time window and τ_{out} denotes forecasting horizon.

3. METHODOLOGY

3.1. AQP model framework

Existing AQP methods based on GNN [13, 18] mostly simulate the spatial relations among stations by constructing a flat static graph. In order to overcome the shortcomings of flat static graphs, we build an encoder-decoder architecture based on dynamic multi-granularity spatiotemporal graph by referring to literature [21], as shown in figure 2. In the encoding stage, multi-granularity graph network is used to learn the spatial relationship among stations and LSTM

network is used to learn its temporal relationship. In the decoding stage, auxiliary data and attention mechanism are used to enhance LSTM forecasting learning and decode the future pollutant concentration value. This framework fully considers three key factors affecting air quality, namely, meteorology, space and time. Among them, the multi-granularity spatiotemporal graph neural network (as shown in figure 3) is the focus.

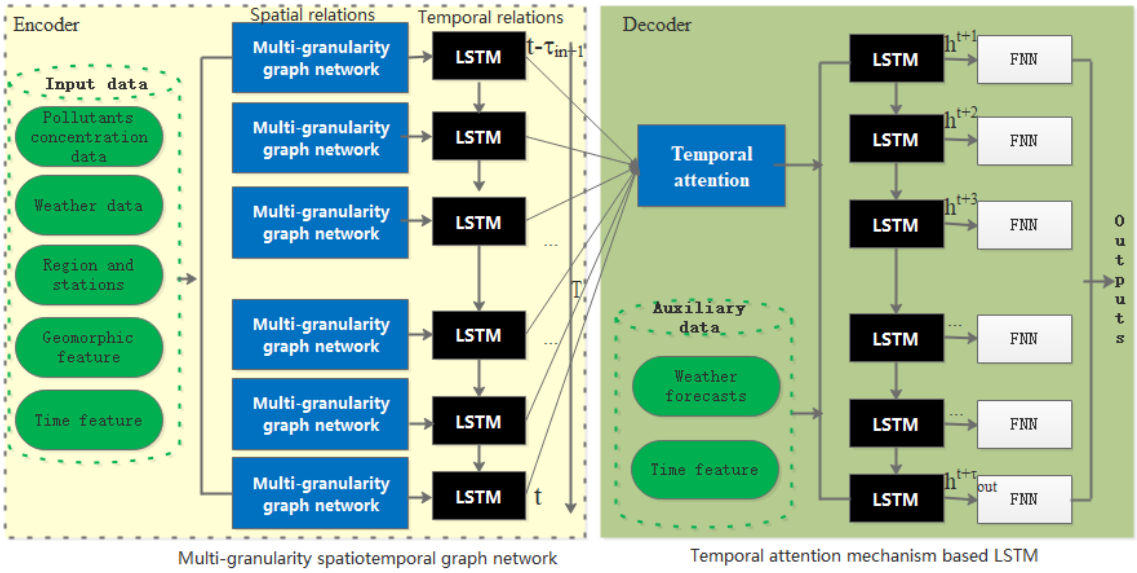


Figure.2: Framework diagram of AQP model

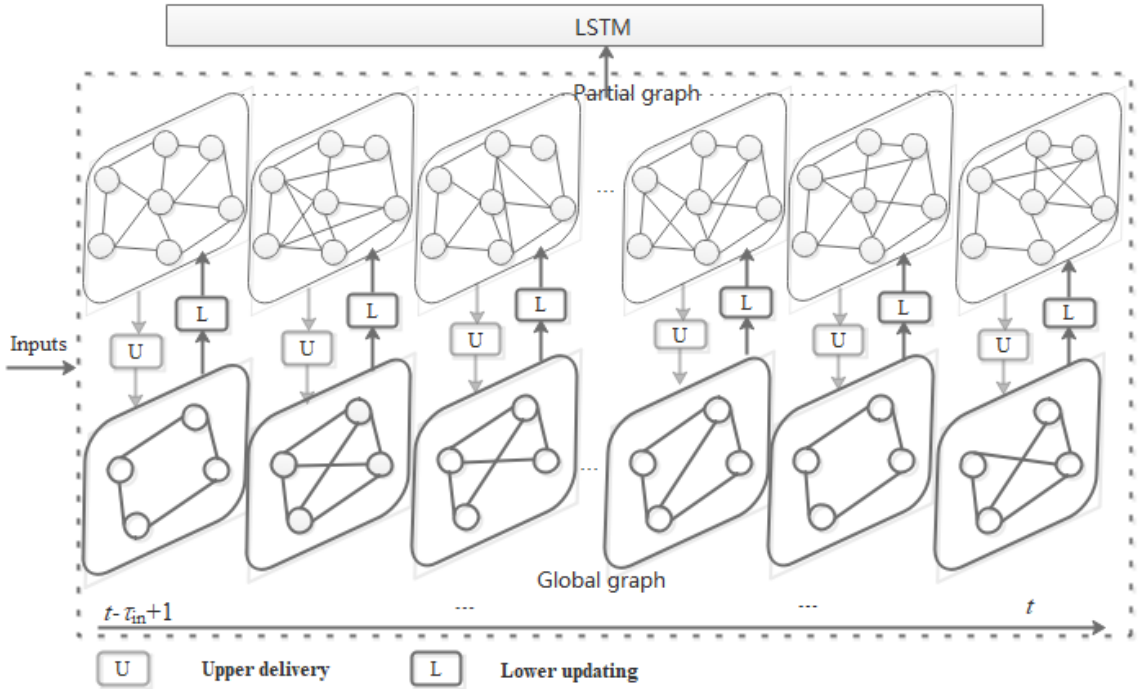


Figure.3: Multi-granularity spatiotemporal graph neural network

3.2. Spatiotemporal graph construction

We propose a multi-granularity spatiotemporal graph network (MGST_GNN) by referring to reference [20]. Assume that V is the set of nodes, E is the set of connected edges among nodes, and u is the global attribute of a station-level partial subgraph. MGST_GNN can be divided into global and partial granularity graph structures. Global graph and partial subgraph interact through message passing mechanism. Partial subgraph's information is upper delivery to the global graph; the global graph's information lower updating to the partial subgraph. In each time period, MGST_GNN will calculate the attribute of each global node and use it to update the corresponding partial node attribute. For each region of the partial subgraph, we calculate the overall air quality information representation of the regions at each time slot, forming a sequence sent to the LSTM to get the representations of current and historical pollutant concentration.

Taking the AQP of all stations in a city as an example, all stations in the city are clustered according to their longitude and latitude information to obtain N categories, that is, N regions are clustered. Therefore, each region can be used as a node in the global graph, and each station in the region can be used as a node in the partial subgraph. A graph composed of region-level nodes is called a global graph, while a graph composed of station-level nodes is called a partial subgraph. From this, a multi-granularity graph network can be constructed.

1. Attribute definitions of graph node

To the node in the global graph, its attribute is the pollutant concentration value of the node region. To the node in partial subgraph, its attribute is composed of the pollutant concentration value of the node station, GF and TF.

2. Edge and attribute definitions of graphs

Wind speed, wind direction, rainfall and other meteorological data have a decisive influence on the horizontal transmission of pollutants. To make use of this domain knowledge, we adopt the air quality model HYSPLIT to learn the relationship among nodes and build dynamic connection edges. Taking the edge calculation of global graph as an example, the city is divided into grids according to the clustered regions, so that each region falls into a unique grid. For each region (node v_a), HYSPLIT is used to calculate all trajectories starting from v_a and stepping in hours within the next time *t*. Track the time when each trajectories crosses the grid where the node is located, and record the number of pass trajectories and crossing time of the grid from other nodes except v_a, and dynamically construct the global connection edge and its attribute vector. Repeat these steps until each station is analyzed as a starting point. For example, taking v_1 as the starting point, using HYSPLIT to calculate that there are 5 trajectories passing through v_2 in the following 48 hours, and the time interval is 1 hour, 3 hours, 4 hours, 8 hours and 48 hours in the future, then the edge relationship between node v_1 and v_2 is formed, and its attribute vector is: e_{12} =[1,0,1,1,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1]. From e_{12}, we can get the specific impact time of node v_1 to v_2. It is a dynamic, specific and accurate edge attribute, which is calculated by the mechanism model according to the meteorological conditions and geographical location. Station-level partial subgraph edge and attribute calculation are similar.

3.3. Modeling of spatial dependencies

MGST_GNN can model spatial dependencies with different granularities by message passing mechanism, which mainly includes message aggregation and description updating. Message aggregation is shown as follows:

$$M_a = \{(x_s, x_a, e_{a,s})\}_{s \in \mathrm{N}(a)} \quad (1)$$

$$m_a \leftarrow \psi(M_a) \quad (2)$$

where, M_a denotes the set of all the messages passed to node v_a; x_a is the attribute of node v_a; x_s is the attribute of a neighbor node; $e_{a,s}$ is the edge attribute; $\mathrm{N}(a)$ denotes the neighbor node set of node v_a (Dynamically determined by HYSPLIT

trajectory); m_a is the aggregation vector of node v_a; $\psi(\cdot)$ denotes aggregate function.

Description updating is shown below:

$$\begin{cases} x'_a \leftarrow \phi_1\left(m_a, x_a\right), \\ in\ global\ graph \\ x'_{a,i} \leftarrow \phi_2\left(m_{a,i}, x_{a,i}, u_a\right), \\ in\ partial\ graph \end{cases} \tag{3}$$

where, $\varphi_1 and \varphi_2$ denote update function; $m_{a,i}$ is the aggregation vector of node $v_{a,i}$ in the $a\text{-}th$ partial subgraph; $x_{a,i}$ is the attribute of node $v_{a,i}$ in the $a\text{-}th$ partial subgraph; u_a is the global attribute of $a\text{-}th$ partial subgraph; the $\varphi_1 and \varphi_2$ can be implemented using different FNNs.

The specific message aggregation method is from the partial subgraph to the global graph. The information transmitted from the partial subgraph includes the current and historical pollutant concentration values, and the average method is adopted to aggregate the information transmitted from the partial subgraph:

$$con_a^t = mean(con_{a,i}^t), \quad 1 \le i \le \left|S_a\right| \tag{4}$$

where, con_a^t denotes the aggregate value at time t of node v_a in the global graph; $con_{a,i}^t$ denotes the pollutant concentration value at time t of node $v_{a,i}$; $\left|S_a\right|$ is the number of nodes of the $a\text{-}th$ partial subgraph. To node in the global graph, message aggregation is used to calculate the global representation of pollutant concentration in each time slot, thus forming a series of sequences: $\left\{con_a^{t-\tau_{in}+1}, con_a^{t-\tau_{in}+2}, \cdots, con_a^t \mid 1 \le a \le N\right\}$ These sequences are then fed into the global LSTM to learn the current and historical representations.

The specific description updating method is to update the partial subgraph using the global graph. The information transmitted from the global graph includes the historical pollutant concentration values of all regions. At each time slot, the output of the global graph is fed into the FNN to obtain a downward update vector, which is used to update the global attribute u of the partial subgraph. The u include meteorological parameters and downward update vector, which are used in the partial subgraph by message passing. Therefore, the nodes in the station-level partial subgraph can make use of the historical information of neighbor nodes in the region-level global graph.

3.4. Modeling of temporal dependencies

In order to capture the temporal dependence, we adopt an encoder-decoder architecture, as shown in figure 1. In each time slot, node attributes in the partial subgraph form a sequence $X_{a,i} = \left\{x_{a,i}^{t-\tau_{in}+1}, x_{a,i}^{t-\tau_{in}+2}, \cdots, x_{a,i}^t \mid 1 \le a \le N, 1 \le i \le \left|S_a\right|\right\}$. That is, each time slot in the historical has a corresponding node attribute. The LSTM in the encoder uses $X_{a,i}$ as the input and the final state of the LSTM as the input of the decoder. The input of the LSTM in the decoder includes not only the output of the encoder, but also the node attribute in the partial subgraph. The output of LSTM in the decoder is used as the input of FNN, and FNN outputs the predicted pollutant concentration value in the future τ_{out} time slot.

To make better use of temporal characteristics, temporal attention mechanism is introduced in the decoding stage to learn the dynamic temporal correlation between future time and historical time. Give the hidden state $h'_{t'-1}$ and cell state $c'_{t'-1}$ of the LSTM in the decoder at time $t'-1$, then at time t', the attention weight of the hidden state h_t output by the encoder is calculated as follows:

$$\rho_{t'}^t = v^T \tanh\left(W\left[h'_{t'-1}; c'_{t'-1}\right] + Uh_t + b\right) \tag{5}$$

$$\lambda_{t'}^t = \frac{\exp\left(\rho_{t'}^t\right)}{\sum_{t=1}^{T} \exp\left(\rho_{t'}^t\right)} \tag{6}$$

where, $\lambda_{t'}^t$ is the attention weight; v, b, W, U are the parameters to be learned. Through the above formula (5) and (6), the attention weight of all historical

hidden states in the encoder can be calculated, and then, the hidden state h_t is weighted and summed to obtain the time context vector c：

$$c_{t'} = \sum_{t=1}^{T} \lambda_{t'}^{t} h_t \quad (7)$$

The output result $\hat{o}_{t'-1}$ at time $t'-1$ of decoder, the meteorological data $wd_{t'}$ at time t', the time feature $tf_{t'}$ and the time context vector $c_{t'}$ are connected as the input for the LSTM of the decoder at time t', and it is used to update the hidden state $h'_{t'}$:

$$h'_{t'} = \mathrm{LSTM}\left(\mathrm{h}'_{t'-1}, \left[\hat{o}_{t'-1}; wd_{t'}; tf_{t'}; c_{t'}\right]\right) \quad (8)$$

4. EXPERIMENTS

4.1. Experimental datasets

Jinan database (JN)：JN is a city-level data set collected by us in projects. Jinan is located in the middle of Shandong Province, China. There are 130 air monitoring stations in Jinan. Each station outputs the concentration values of pollutants (PM2.5, PM10, SO2, CO2, Co, O3) and meteorological parameters (rainfall, surface pressure, temperature, humidity, wind speed and wind direction) every hour. We collected the historical monitoring data of 130 stations in Jinan from January 1st, 2019, to January 1st, 2022, as the training and test set.

We divide the 130 stations into 13 regions, and the global graph consists of 13 regions. Each region is used as a global node in the global graph. The stations in each region form a partial subgraph, and the stations in the region are the nodes of the partial subgraph.

Yangtze River Delta city group database (YRD): The city group contains ten cities: Shanghai, Hangzhou, Suzhou, Ningbo, Shaoxing, Jiaxing, Wuxi, Zhoushan, Nantong, and Huzhou. We used air pollution prediction system[3] to collect historical pollutant concentration values and meteorological parameters of corresponding stations, and the time span was from January 1th, 2019, to December 31th, 2022. Therefore, we collected 3 years of historical monitoring data as the training and test set. Each city in the Yangtze River Delta city group is a global graph node, and each station in the city is a node of the corresponding partial subgraph.

Geographic features and weather forecast data are obtained by: Geographic features are collected from the map engine of AMAP[4].The perception radius is set to 1000 m.

Weather forecast data are collected from the Air Resources Laboratory （ ARL ）[5] .This website can download the meteorological forecast data for the following 26 days at most, with an accuracy of 0.25× 0.25 degrees, which can be updated four times a day.

4.2. Experimental settings

We split the dataset into training data, validation data, and test data by the radio of 0.7:0.1:0.2. We choose Adam [23] as the optimizer in the training phase. During the training phase, the batch size is set to 128 and the epoch size is set to 500, and use RMSprop [19] for 50 epochs with learning rate as 5−4. The hidden size of GNNs is set to 32, and the hidden state size of LSTMs is set to 64.

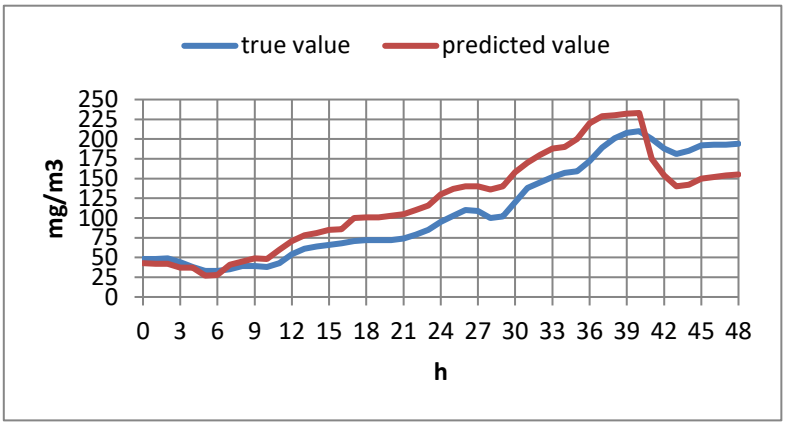
We implement our method by PyTorch [24], constructing GNNs with PyTorch geometric library [25] and implement HYSPLIT trajectory analysis and edge weight construction with PySplit [22]. The code is released on GitHub. A server with one CPU (Intel®Xeon®Platinum), and one GPU (NVIDIA Tesla T4) accomplishes all computing tasks.

We introduce two metrics: mean absolute error (MAE) and symmetric mean absolute percentage error (SMAPE) to evaluate the performances of methods. In the experiments, we utilize previous 48-hour observations and select the results of 1 hour, 6 hours, 12 hours, 18 hours, 24 hours, 36 hours and 48 hours ahead forecasting to report. All experiments are repeated 5 times to avoid contingency. Figure 4 depicts the predicted PM 2.5 and O3 concentrations and their corresponding real values of a station in Jinan city from 00:00 on January 2nd, 2021 to 23:00 on January 3rd, 2021 for consecutive 48 hours. From figure 4, we can see that it is easier to predict O3 concentration than PM2.5, because O3 has more periodic regularity and stability than PM2.5. Therefore, the subsequent experiments were carried out with the forecasting of PM2.5 concentration.

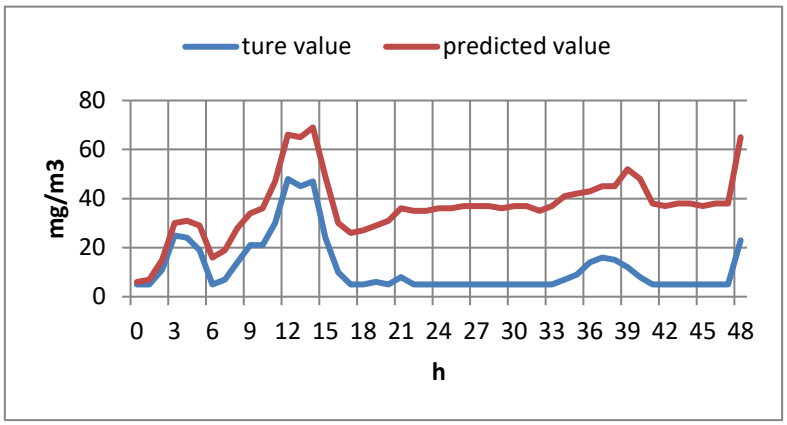
3 http://airprediction.urban-computing.com

4 https://lbs.amap.com/api/webservice/guide/api/search/

5 https://nomads.ncep.noaa.gov/pub/data/



（a）PM2.5



（b）O3

Figure.4: 48 - hour predicted value versus true value

4.3. Multi-source factor evaluation

To verify the effectiveness of multiple factors, we compare MGST_GNN with three variants, each of which removes one kind of factors. Specifically, MGST_GNN w/o wdf removes meteorological forecast data; MGST_GNN w/o tf removes time feature and MGST_GNN w/o gf removes geomorphic feature.

The performances of MGST_GNN and its variants are given in Table 2, as can be seen from the table:

MGST_GNN outperforms the other three variants, indicating that all factors can improve the performance of AQP. The rank of the effectiveness of factors is WD > TF> GF. The result shows that air quality is mostly impacted by weather conditions.

4.4. Model Component Evaluation

To explore the effectiveness of different components, we compare MGST_GNN with three variants:

MGST_GNN w/o multi granularity, which removes the global graph and the corresponding interactions, i.e., the region representation is removed from the global attributes of the partial subgraphs;

MGST_GNN w/o HYSPLIT, which removes dynamic edge weight vector by HYSPLIT and use vector of wind direction and distance instead;

MGST_GNN w/o taLSTM, which removes temporal attention mechanism based LSTM at the decoder, and directly use LSTM instead.

The performances of MGST_GNN and its variants are given in Table 3, as can be seen from the table:

MGST_GNN outperforms MGST_GNN w/o multi granularity in all metrics. The result indicates that the air quality of adjacent regions is beneficial, which can

be used to model the diffusion processes of air pollutants from adjacent regions.

MGST_GNN outperforms MGST_GNN w/o HYSPLIT. The result indicates that compared with taking wind direction and distance as the edge weight vector, using HYSPLIT to dynamically adjust the weights of edges is a more effective strategy, which can model the effect patterns of wind direction on air pollutant diffusion with domain knowledge.

The attention mechanism based LSTM at the decoder is helpful for longer-term forecasting, but for short-term forecasting.

Table.2

the results of factor evaluation based on JN database

method	Metric	1h	6h	12h	18h	24h	36h	48h
MGST_GNN	MAE	**6.15**	**15.24**	**20.56**	**25.62**	**34.83**	**43.98**	52.06
	SMAPE	**0.07**	**0.10**	**0.14**	**0.19**	**0.26**	**0.33**	**0.46**
w/o wdf	MAE	7.06	17.17	23.68	29.33	38.56	46.70	53.78
	SMAPE	0.07	0.11	0.16	0.21	0.28	0.35	0.47
w/o tf	MAE	6.76	17.16	23.56	28.45	34.96	45.55	**52.01**
	SMAPE	0.07	0.11	0.16	0.20	0.26	0.34	0.46
w/o gf	MAE	6.71	16.78	22.56	28.64	34.95	46.56	52.11
	SMAPE	0.07	0.11	0.15	0.20	0.26	0.35	0.46

Table.3

the results of model component evaluation based on JN database

method	Metric	1h	6h	12h	18h	24h	36h	48h
MGST_GNN	MAE	**6.15**	**15.24**	**20.56**	**25.62**	34.83	43.98	**52.06**
	SMAPE	**0.07**	**0.10**	**0.14**	**0.19**	**0.26**	**0.33**	**0.46**
w/o multi granularity	MAE	7.12	17.10	23.59	28.02	34.95	48.02	53.58
	SMAPE	0.08	0.12	0.16	0.21	0.26	0.36	0.47
w/o HYSPLIT	MAE	7.16	18.06	23.59	28.16	**34.76**	48.35	53.02
	SMAPE	0.08	0.13	0.16	0.21	0.26	0.36	0.47
w/o taLSTM	MAE	6.15	15.85	21.56	28.64	34.96	**43.16**	52.25
	SMAPE	0.07	0.10	0.15	0.21	0.26	0.33	0.46

4.5. Comparison with other prediction methods

To verify the advanced nature of our method, we compare MGST_GNN with the methods HighAir[20], PM2.5-GNN[21], GC-DCRNN[18], GC-LSTM[17], ST-UNet[26], and STA-LSTM[7].

The performances of MGST_GNN and other prediction methods are given in Table 4 and 5, as can be seen from the table:

MGST_GNN outperforms GC-DCRNN, GC-LSTM and ST_UNet, especially in long-term forecasting. It indicates that a multi-granularity structure can model spatial dependencies more effectively than a flat structure. The reason is that the multi granularity graph not only considers the local impact of neighboring stations on the prediction station, but also the global impact of different regions on the prediction station, which makes the mining of spatial relationships more sufficient.

MGST_GNN outperforms HighAir. This is because in the aspect of spatial relationship learning, MGST_GNN uses the professional HYSPLIT model to build edge weight vector. It comprehensively uses meteorological and topographical conditions to calculate the influence relationship among nodes and the specific influence time, making the construction of spatial relationship more accurate and delicate. HighAir simply uses wind direction and distance to construct rough edge weight vector. In terms of temporal relation learning, MGST_GNN adds time feature, and uses the attention mechanism based LSTM in decoder. HighAir simply uses LSTM to build temporal relationships.

MGST_GNN outperforms PM2.5-GNN. This is because MGST_GNN uses a multi-granularity spatiotemporal graph network and combines meteorological data, topographic features, time features and professional models to construct node and edge attributes. However, PM2.5-GNN is only a single-granularity spatiotemporal graph network. In the construction of edge attributes, although they make use of domain knowledge, they simply list part of the influencing factors, which is still insufficient. When building node attributes, they only use meteorological data.

Table.4

Model comparison results based on JN database

method	Metric	1h	6h	12h	18h	24h	36h	48h
MGST_GNN	MAE	9.23	**19.37**	**23.87**	30.81	**39.52**	**47.69**	**56.13**
	SMAPE	**0.09**	**0.13**	**0.15**	**0.22**	**0.32**	**0.35**	**0.49**
HighAir	MAE	**9.18**	19.89	24.57	**30.59**	41.77	50.36	57.38
	SMAPE	0.09	0.13	0.16	0.22	0.33	0.36	0.50
PM2.5-GNN	MAE	9.86	20.12	25.54	32.57	10.96	51.09	58.34
	SMAPE	0.09	0.14	0.16	0.23	0.32	0.37	0.50
GC-DCRNN	MAE	9.56	20.56	26.88	33.47	46.56	54.24	61.82
	SMAPE	0.09	0.14	0.17	0.24	0.35	0.39	0.52
GC-LSTM	MAE	10.05	21.35	26.53	34.68	45.59	54.98	60.76
	SMAPE	0.10	0.15	0.17	0.24	0.35	0.39	0.51
ST-UNet	MAE	9.86	21.69	25.94	30.87	43.54	54.87	59.25
	SMAPE	0.09	0.15	0.16	0.22	0.34	0.39	0.51

STA-LSTM	MAE	10.14	22.08	26.76	34.96	45.67	53.78	62.08
	SMAPE	0.10	0.16	0.17	0.25	0.35	0.38	0.52

Table.5

Model comparison results based on YRD database

method	Metric	1h	6h	12h	18h	24h	36h	48h
MGST_GNN	MAE	7.16	**17.29**	**21.25**	27.92	**36.97**	**45.28**	**54.73**
	SMAPE	**0.08**	**0.12**	**0.14**	**0.20**	**0.27**	**0.34**	**0.48**
HighAir	MAE	**7.12**	17.83	22.65	**27.49**	38.85	48.36	55.68
	SMAPE	0.08	0.12	0.15	0.20	0.29	0.36	0.49
PM2.5-GNN	MAE	7.15	18.09	23.14	29.57	37.96	48.99	56.12
	SMAPE	0.08	0.13	0.15	0.21	0.28	0.36	0.50
GC-DCRNN	MAE	7.56	18.56	24.88	30.67	43.26	51.84	60.02
	SMAPE	0.08	0.13	0.17	0.22	0.33	0.45	0.53
GC-LSTM	MAE	8.13	18.97	24.53	31.52	42.51	51.98	58.96
	SMAPE	0.09	0.13	0.17	0.22	0.33	0.45	0.52
ST-UNet	MAE	7.22	18.69	23.36	27.86	40.16	51.97	57.85
	SMAPE	0.08	0.13	0.16	0.20	0.32	0.45	0.51
STA-LSTM	MAE	8.15	19.21	24.62	31.96	42.87	50.68	60.37
	SMAPE	0.09	0.14	0.17	0.23	0.33	0.37	0.53

5. CONCLUSION

In this paper, the influence of meteorological, spatial and temporal factors on AQP is fully considered, and an encoder-decoder architecture based on multi-granularity spatiotemporal graph network is proposed to predict pollutant concentration over a long period of time. Compared with the existing models, the striking characteristic of this paper is that the meteorological, spatial terrain and time factors are considered comprehensively through the professional air quality model, while other models take the influencing factors as the splitting parameter input.

That is, the model in this paper integrates the advantages of mechanism model and machine learning, and namely it is a comprehensive model. The experimental results show that the proposed model is of progressiveness and has good applicability.

ACKNOWLEDGMENT

The authors gratefully acknowledge the NOAA Air Resources Laboratory (ARL) for the provision of the HYSPLIT transport and dispersion model and/or READY website (https://www.ready.noaa.gov) used in this publication.

REFERENCES

[1] N. Jittra, N. Pinthong, and S. Thepanondh, "Performance evaluation of AERMOD and CALPUFF air dispersion models in industrial complex area," *Air, Soil and Water Research*, vol. 8, pp. ASWR–S32 781, 2015.

[2] Y. Zhang, "Air quality modelling: Current status, major challenges and future prospects," Air Quality and Climate Change, vol. 51, no. 3, pp.41–47, 2017.

[3] N. K. Arystanbekova, "Application of gaussian plume models for air pollution simulation at instantaneous emissions," Mathematics and Computers in Simulation, vol. 67, no. 4-5, pp. 451–458, 2004.

[4] A. Stein, R. R. Draxler, G. D. Rolph, B. J. Stunder, M. Cohen, and F. Ngan, "NOAA's HYSPLIT atmospheric transport and dispersion modeling system," Bulletin of the American Meteorological Society, vol. 96, no. 12, pp. 2059–2077, 2015.

[5] K. Wang, Y. Zhang, A. Nenes, and C. Fountoukis, "Implementation of dust emission and chemistry into the community multiscale air quality modeling system and initial application to an asian dust storm episode," Atmospheric Chemistry and Physics, vol. 12, no. 21, pp. 10 209–10 237,2012.

[6] X. Yi, J. Zhang, Z. Wang, T. Li, and Y. Zheng, "Deep distributed fusion network for air quality prediction," in Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining, 2018, pp. 965–973.

[7] X. Zou, J. Zhao, D. Zhao, B. Sun, Y. He, and S. Fuentes, "Air quality prediction based on a spatiotemporal attention mechanism," Mobile Information Systems, vol. 2021, 2021.

[8] S. Moisan, R. Herrera, and A. Clements, "A dynamic multiple equation approach for forecasting PM2.5 pollution in Santiago, Chile," International Journal of Forecasting, vol. 34, no. 4, pp. 566–581, 2018.

[9] M. Niu, K. Gan, S. Sun, and F. Li, "Application of decomposition ensemble learning paradigm with phase space reconstruction for day ahead PM2.5 concentration forecasting," Journal of environmental management, vol. 196, pp.110–118, 2017.

[10] J. Ma, Z. Li, J. C. Cheng, Y. Ding, C. Lin, and Z. Xu, "Air quality prediction at new stations using spatially transferred bi-directional long short-term memory network," Science of The Total Environment, vol.705, p. 135771, 2020.

[11] Y. Wang, G. Song, L. Du, and Z. Lu, "Real-time estimation of the urban air quality with mobile sensor system," ACM Transactions on Knowledge Discovery from Data (TKDD), vol. 13, no. 5, pp. 11–19, 2019.

[12] J. Zhang, Y. Zheng, D. Qi, R. Li, and X. Yi, "DNN-based prediction model for spatio-temporal data," in Proceedings of the 24th ACM SIGSPATIAL international conference on advances in geographic information systems, 2016, pp. 1–4.

[13] S. Du, T. Li, Y. Yang, and S.-J. Horng, "Deep air quality forecasting using hybrid deep learning framework," IEEE Transactions on Knowledge and Data Engineering, vol. 33, no. 6, pp. 2412–2424, 2019.

[14] Y. Liang, S. Ke, J. Zhang, X. Yi, and Y. Zheng, "GeoMAN: Multi-level attention networks for geo-sensory time series prediction." in IJCAI, vol. 2018, 2018, pp. 3428–3434.

[15] M. Xu, Y. Yang, M. Han, T. Qiu, and H. Lin, "Spatio-temporal interpolated echo state network for meteorological series prediction," IEEE transactions on neural networks and learning systems, vol. 30, no. 6, pp. 1621–1634, 2018.

[16] L. Chen, Y. Ding, D. Lyu, X. Liu, and H. Long, "Deep multi-task learning based urban air quality index modelling," Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, vol. 3, no. 1, pp. 1–17, 2019.

[17] Y. Qi, Q. Li, H. Karimian, and D. Liu, "A hybrid model for spatiotemporal forecasting of PM2.5 based on graph convolutional neural network and long short-term memory," Science of the Total Environment, vol. 664, pp. 1–10, 2019.

[18] Y. Lin, N. Mago, Y. Gao, Y. Li, Y.-Y. Chiang, C. Shahabi, and J. L. Ambite, "Exploiting spatiotemporal patterns for accurate air quality forecasting using deep learning," in Proceedings of the 26th ACM SIGSPATIAL international conference on advances in geographic information systems, 2018, pp. 359–368.

[19] Z. Xu, Y. Kang, Y. Cao, and Z. Li, "Spatiotemporal graph convolution multifusion network for urban vehicle emission prediction," IEEE Transactions on Neural Networks and Learning Systems, vol. 32, no. 8, pp. 3342–3354, 2020.

[20] J. Xu, L. Chen, M. Lv, C. Zhan, S. Chen, and J. Chang, “HighAir: A hierarchical graph neural network-based air quality forecasting method,” arXiv preprint arXiv:2101.04264, 2021.

[21] S. Wang, Y. Li, J. Zhang, Q. Meng, L. Meng, and F. Gao, “PM2.5-GNN: A domain knowledge enhanced graph neural network for PM2.5 forecasting,” in Proceedings of the 28th International Conference on Advances in Geographic Information Systems, 2020, pp. 163–166.

[22] M. S. Warner, “Introduction to PySPLIT: A Python toolkit for NOAA ARL’s HYSPLIT model,” Computing in Science & Engineering, vol. 20, no. 5, pp. 47–62, 2018.

[23] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” arXiv preprint arXiv:1412.6980, 2014.

[24] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga et al., “PyTorch: An imperative style, high-performance deep learning library,” Advances in neural information processing systems, vol. 32, 2019.

[25] M. Fey and J. E. Lenssen, “Fast graph representation learning with PyTorch geometric,” arXiv preprint arXiv:1903.02428, 2019.

[26] B. Yu, H. Yin, and Z. Zhu, “ST-UNet: A spatio-temporal Unetwork for graph-structured time series modeling,” arXiv preprint arXiv:1903.05631, 2019.

APPENDIX

A.1 DATA AUGMENTATION AND EXPERIMENTATION

Aiming at the difficulty of collecting training samples for air quality prediction, we use the method of data augmentation to reconstruct samples. Due to the distribution difference between real weather data and forecast weather data, Gaussian noise is introduced to the meteorological data in the sample to enhance the data and improve the generalization ability of the model. In addition, due to the influence of monitoring instruments, environment and other factors, the pollutant concentration values collected may be biased. Therefore, the pollutant concentration values of samples are disturbed up and down by 1 metric to enhance the number of samples and improve the model generalization ability. Therefore, Gaussian noise is introduced into meteorological data and random perturbation is for pollutant concentration data to enhance the samples by three times. The specific augmentation methods are shown in Table 5. Finally, 26136 samples were obtained from JN and YRD respectively.

In order to verify the data augmentation effect, we conduct a comparison experiment between data augmentation and non-data augmentation, as shown in Table 6, where w/o DA indicates that data augmentation technology is not used. As seen from Table 6: using our data augmentation method, the effect is effectively improved.

A.2 ABLATION STUDY AND EXPERIMENTATION

In this paper, we are the first to use the professional model HYSPLIT to build the graph dynamically. Therefore, this appendix section will demonstrate the effectiveness of using HYSPLIT to construction graph dynamically through ablation experiments. Specifically, the ablation experiment was conducted based on the current advanced spatiotemporal graph neural network HighAir and PM2.5-GNN. In HighAir, it uses the distance among nodes to statically construct the edge of the graph, and uses the wind direction information between nodes to calculate the edge attribute vector. Therefore, we use HYSPLIT instead of the graph construction method in HighAir. In PM2.5-GNN, it uses the distance between nodes and the altitude of the position to statically construct the edge of the graph, and uses the parameter of domain knowledge between nodes to calculate the edge attribute vector. Therefore, we use HYSPLIT instead of the graph construction method in PM2.5-GNN. The experimental results are shown in Table 7, where HighAir_HYSPLIT and PM2.5-GNN_HYSPLIT respectively indicate that HYSPLIT is used to replace the original graph construction. As can be seen from the table, HYSPLIT builds dynamic graphs better than static graphs of HighAir and PM2.5-GNN.

In order to further verify the effectiveness of using HYSPLIT to dynamically construct graph, especially its advantages for air quality prediction in complex scenarios such as abrupt change in pollutant concentration. We develop a Dataset-mini, where we focus on heating season (November to February). Dataset-mini is more challenging for two reasons. Firstly, during winters, heating emissions can dramatically increase the frequency of air pollution occurrence. Secondly, the direction of prevailing wind is north or northwest, which contributes to pollutant's long-distance transport from North China to South China. The results in Table 9 show that using HYSPLIT to dynamically construct graph can significantly improve the accuracy of the model's air quality prediction on sudden changes in pollution and regional impacts caused by strong winds. This method eliminates the construction process of specially designed auxiliary network to learn the edges of graph and provides a new way for the construction of graph neural network, which can be easily extended to other spatiotemporal forecasting tasks. For example, in water quality prediction, a professional hydrodynamic model (MIKE) can be used to dynamically construct the graph structure, so as to better learn the influence of water quality in different regions on the prediction points.

Table.5

List of sample augmentation

Sample type \ Augmentation type	Meteorological data plus noise	Concentration value disturbance
The original sample	-	-
Sample 1	√	-
Sample 2	-	√
Sample 3	√	√

Table.6

the MAE results of data augmentation evaluation based on JN database

method	Pollutant category	1h	6h	12h	18h	24h	36h	48h
MGST_GNN	pm2.5	**6.15**	**15.24**	**20.56**	**25.62**	**34.83**	**43.98**	**52.06**
	O3	**5.06**	**13.36**	**17.14**	**22.16**	**29.90**	**37.93**	**44.61**
w/o DA	pm2.5	6.25	16.97	22.18	28.23	36.96	46.31	54.69
	O3	5.11	14.84	18.86	23.72	30.78	39.19	48.07

Table.7

ablation experiment result based on YRD database

method	Metric	1h	6h	12h	18h	24h	36h	48h
HighAir	MAE	7.12	17.83	22.65	27.49	38.85	48.36	55.68
	SMAPE	0.08	0.12	0.15	0.20	0.29	0.36	0.49
HighAir_HYSPLIT	MAE	7.12	17.26	21.18	27.50	37.65	46.58	54.75
	SMAPE	0.08	0.12	0.14	0.20	0.28	0.35	0.48
PM2.5-GNN	MAE	7.15	18.09	23.14	29.57	37.96	49.21	56.12
	SMAPE	0.08	0.13	0.15	0.21	0.28	0.37	0.50
PM2.5-GNN_HYSPLIT	MAE	7.13	18.01	22.79	29.05	36.97	47.95	55.52
	SMAPE	0.08	0.13	0.15	0.21	0.27	0.35	0.49

Table.8

ablation experiment result based on Data-mini

method	Metric	1h	6h	12h	18h	24h	36h	48h
HighAir	MAE	8.02	20.14	27.55	35.27	44.18	56.46	65.47
	SMAPE	0.08	0.14	0.20	0.27	0.33	0.50	0.59
HighAir_HYSPLIT	MAE	7.86	18.73	25.17	32.94	40.49	50.97	56.17
	SMAPE	0.08	0.13	0.18	0.25	0.30	0.37	0.51
PM2.5-GNN	MAE	8.11	21.30	27.96	34.66	43.88	57.69	67.08
	SMAPE	0.09	0.14	0.20	0.27	0.33	0.51	0.60
PM2.5-GNN_HYSPLIT	MAE	7.95	19.06	26.14	32.99	40.85	50.89	55.78
	SMAPE	0.08	0.13	0.19	0.25	0.30	0.37	0.51