AirFormer: Predicting Nationwide Air Quality in China with Transformers

Yuxuan Liang[1], Yutong Xia[1], Songyu Ke[4,2], Yiwei Wang[1], Qingsong Wen[3]
Junbo Zhang[2], Yu Zheng[2], Roger Zimmermann[1]
[1]National University of Singapore, Singapore
[2]JD Intelligent Cities Research & JD iCity, JD Technology, Beijing, China
[3]DAMO Academy, Alibaba Group, Hangzhou, China
[4]Shanghai Jiao Tong University, Shanghai, China
{yuxliang, songyu-ke,msjunbozhang, msyuzheng}@outlook.com;
{xiayutong618,qingsongedu}@gmail.com; {y-wang,rogerz}@comp.nus.edu.sg

Abstract

Air pollution is a crucial issue affecting human health and livelihoods, as well as one of the barriers to economic growth. Forecasting air quality has become an increasingly important endeavor with significant social impacts, especially in emerging countries. In this paper, we present a novel Transformer termed AirFormer to predict nationwide air quality in China, with an unprecedented fine spatial granularity covering thousands of locations. AirFormer decouples the learning process into two stages: 1) a bottom-up deterministic stage that contains two new types of self-attention mechanisms to efficiently learn spatio-temporal representations; 2) a top-down stochastic stage with latent variables to capture the intrinsic uncertainty of air quality data. We evaluate AirFormer with 4-year data from 1,085 stations in Chinese Mainland. Compared to prior models, AirFormer reduces prediction errors by 5%∼8% on 72-hour future predictions. Our source code is available at https://github.com/yoshall/airformer.

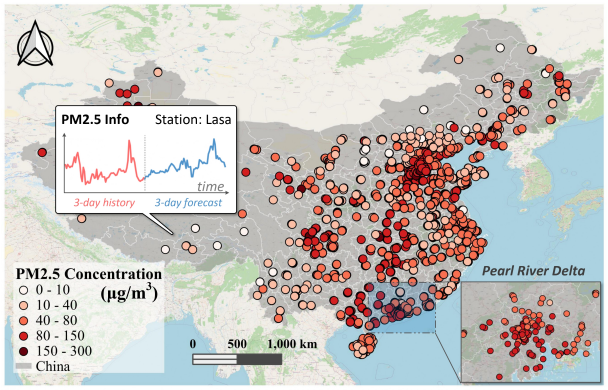


Figure 1: Distribution of 1,085 air quality stations in Chinese mainland. Each station provides hourly air quality readings.

Introduction

Air pollution refers to the release of air pollutants, such as gases, dust, fumes, and odors, into the atmosphere that are detrimental to human health and the environment. According to the World Health Organization (WHO), air pollution is one of the leading causes of death in the world today, accounting for seven million fatalities per year (Vallero 2014). Over 90% of people breathe air that contains more contaminants than the WHO's recommended levels, with those in emerging countries suffering the most, especially in China with 1.4 billion people (Wang and Hao 2012).

To inform citizens about real-time air quality, many Chinese cities have built a number of air quality monitoring stations. These stations report time series readings every hour, including the concentration of particulate matter ($PM_{2.5}$ and PM_{10}), NO_2, etc. For instance, a 50 μg/m^3 concentration of $PM_{2.5}$ refers to good air quality, indicating "a terrific day to be active outside" according to the definition of the U.S. Environmental Protection Agency (Lin et al. 2018). Beyond real-time monitoring, forecasting air quality also becomes an increasingly important endeavor with the economic, ecologic, and human toll that air pollution takes, which significantly aids human health protection and policy-making.

During the past decades, there has been a long-established line of research conducted for air quality prediction, ranging from classical dispersion models (Vardoulakis et al. 2003) to data-driven models (Zheng et al. 2015; Yi et al. 2018). Considering the computational expense, they mostly focused on predicting the air quality at a city scale with dozens of monitoring stations, e.g., there are 35 stations in Beijing. With recent advances in deep learning, researchers started to explore nationwide air quality prediction, i.e., *collectively* predict on nearly two hundred stations in China using Spatio-Temporal Graph Neural Networks (STGNNs) (Wang et al. 2020; Chen et al. 2021). STGNNs couple Graph Neural Networks (GNNs) with deep sequential models, e.g., Recurrent Neural Networks (RNNs), in which GNNs are used to capture the spatial correlations among stations (i.e., dispersion), and RNNs are utilized to learn the temporal dependencies.

Further, attention-based models, in particular Transformers, have become a strong alternative to capture spatial correlations in air quality data (Wang et al. 2021, 2022). They have two major merits over STGNNs. First, they jointly capture short and long-term interactions among different places at each layer, while STGNNs merely convolve the local surroundings. Second, the correlations of air quality between locations is highly dynamic, changing over time (Liang et al. 2018; Cheng et al. 2018). Using transformers can naturally address this issue (Wen et al. 2022; Zhou et al. 2022).

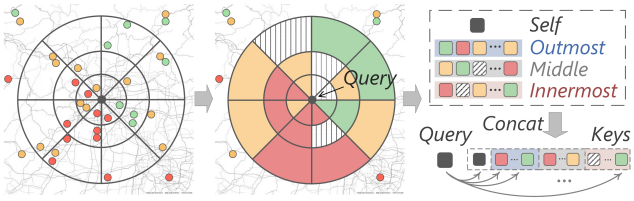
In this study, we broaden our scope to collectively predict air quality in Chinese mainland with an unprecedented fine spatial granularity using transformers, covering thousands of stations. As shown in Figure 1, our prediction targets encompass all provinces of Chinese mainland, with dense distribution in developed regions, e.g., Pearl River Delta. Such a fine coverage not only provides more useful information to the public with high social impacts, but also comprises more data samples that benefit model training (Zhao et al. 2015).

Despite these benefits, forecasting air quality with a fine spatial granularity poses a significant modeling difficulty for transformers – *efficiency*. Multi-head Self-Attention (MSA), which is the key operation of transformers for spatial modeling, takes quadratic computational complexity w.r.t the number of stations N. Such expense may become unaffordable with the growth of N, especially for our fine-grained data.

Meanwhile, the future air quality readings are intrinsically *uncertain* due to two factors – inaccurate or missing observations, and some unpredictable factors, e.g., vehicle exhaust, policy, and industrial emission. While earlier attempts have shown promising performance on air quality prediction via deterministic approaches, most of them still fall short of capturing such uncertainty within large-scale air quality data.

To tackle these challenges, we present a novel transformer architecture for nationwide air quality prediction in China, entitled **AirFormer**. Our method is motivated by the domain knowledge of air pollution, which enables us to build models with more interpretations. AirFormer decouples the solution to the two issues into dual stages – a *deterministic* stage and a *stochastic* stage. In the deterministic stage, we propose two new types of MSA to *efficiently* capture the spatial and temporal dependencies, respectively. In the stochastic stage, we explore the inclusion of *latent random variables* into the transformer. These latent variables are sampled from probability distributions that are learned from the deterministic hidden states, thus capturing the uncertainty of the input data. In summary, our contributions lie in four aspects:

- Considering that the spatial correlations among nearby locations are often stronger than those far away, we devise the *Dartboard Spatial MSA* (DS-MSA) to efficiently capture spatial relations. As its name suggests, each location attends to close surroundings at a fine granularity and faraway stations at a coarse granularity (see Fig. 2). DS-MSA only takes *linear complexity* w.r.t the number of stations.
- We devise *Causal Temporal MSA* (CT-MSA) for learning temporal dependencies. It ensures that the output of a step derives only from previous steps, i.e., causality. Locality is also introduced to improve efficiency, where the receptive field in each layer is gradually increased like convolutions.
- Leveraging recent advances in variational models, we enhance transformers with *latent variables* to capture the uncertainty of air quality data. To preserve the parallelism of transformers, the latent random variables are arranged hierarchically with implicit temporal dependencies.
- To the best of our knowledge, this is the first work for *collectively* forecasting air quality among thousands of locations. The empirical results show that AirFormer obtains 4.6%-8.2% lower prediction errors than existing models.



(a) Spatial partition (b) Spatial projection (c) Spatial attention

Figure 2: Sketch of DS-MSA. For a query location (the black point), we first divide its surroundings into regions bounded by three circles and four lines. Then we project other stations onto the regions to obtain regional representations. Finally, we use the station feature as the query and the regional representations as the keys and values for attention computation.

Preliminary

Problem Formulation

The readings of N air quality monitoring stations at a given time t can be denoted as $\mathbf{X}_t \in \mathbb{R}^{N \times D}$, where D is the number of measurements, including *air pollutants* (e.g., $PM_{2.5}$, NO_2) and *external factors* (e.g., weather, wind speed). Each entry x_{ij} indicates the value of the j-th measurement of the i-th station. Given the historical readings of all stations from the past T time steps, we aim to learn a function $\mathcal{F}(\cdot)$ that predicts D' kinds of measurements over the next τ steps: $\mathbf{X}_{1:T} \overset{\mathcal{F}(\cdot)}{\rightarrow} \mathbf{Y}_{1:\tau}$, where $\mathbf{X}_{1:T} \in \mathbb{R}^{T \times N \times D}$ is the history data and $\mathbf{Y}_{1:\tau} \in \mathbb{R}^{\tau \times N \times D'}$ is the future readings.

Multi-head Self-Attention (MSA)

MSA is the key operation of transformers to learn an alignment where each token in the sequence learns to gather messages from other tokens (Vaswani et al. 2017). Let $\mathbf{X} \in \mathbb{R}^{S \times C}$ be the input sequence with length S and feature dimension C. The operation of a single head is defined as:

$$\mathbf{X}_h = \text{Softmax}\left(\alpha \mathbf{Q}_h \mathbf{K}_h^\top\right) \mathbf{V}_h, \tag{1}$$

where $\mathbf{X}_h \in \mathbb{R}^{S \times C/N_h}$ is the output features; $\mathbf{Q}_h = \mathbf{X}\mathbf{W}_q$, $\mathbf{K}_h = \mathbf{X}\mathbf{W}_k$ and $\mathbf{V}_h = \mathbf{X}\mathbf{W}_v$ are queries, keys, and values, respectively; $\mathbf{W}_q, \mathbf{W}_k, \mathbf{W}_v \in \mathbb{R}^{C \times C/N_h}$ are the learnable parameters for linear projection, and N_h is the number of heads; α is a scaling factor. The computational complexity of Eq. (1) is quadratic w.r.t the sequence length S.

Variational Models with Latent Variables

Variational autoencoders (VAE) (Kingma and Welling 2013) have long been verified to be an effective modeling paradigm for recovering complex multimodal distributions over the latent space. VAE addresses the data distribution $p(\mathbf{x})$ using an unobserved latent variable $\mathbf{z}$ and is parameterized by θ as:

$$p_\theta(\mathbf{x}) = \int p_\theta(\mathbf{x}|\mathbf{z}) p_\theta(\mathbf{z}) d\mathbf{z}. \tag{2}$$

As the integral is usually intractable, VAE introduces an approximate posterior $q_\phi(\mathbf{z}|\mathbf{x})$ and implicitly optimizes the evidence lower bound (ELBO) of the marginal log-likelihood:

$$\log p_\theta(\mathbf{x}) \geq -KL\left(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z})\right) + \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\log p_\theta(\mathbf{x}|\mathbf{z})\right],$$

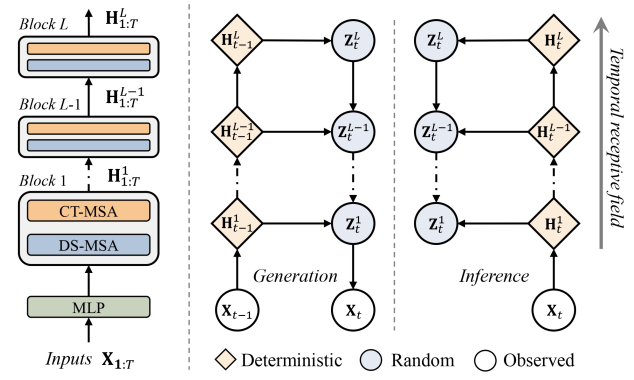
where KL denotes the KL divergence. The prior $p_\theta(\mathbf{z})$ and the posterior $q_\phi(\mathbf{z}|\mathbf{x})$ of the latent variables are usually taken to be Gaussian distribution with diagonal covariance, which inherently encodes the uncertainty of the input data.

Methodology

Figure 3 shows the framework of AirFormer for nationwide air quality prediction, which is decoupled into two stages:

- *Bottom-up deterministic stage*: We first transform the historical readings $\mathbf{X}_{1:T}$ into the feature space using a multi-layer perceptron (MLP). The transformed features are then fed to L AirFormer blocks to learn deterministic spatio-temporal representations. In each block, we factorize the space-time modeling along temporal and spatial domains, leading to the dual levels of MSA: DS-MSA for learning spatial interactions with linear complexity, and CT-MSA for capturing the temporal dependencies at each location. As shown in Figure 3(a), the output state at the l-th block is a 3D tensor, denoted as $\mathbf{H}^l_{1:T} \in \mathbb{R}^{T\times N\times C}$.
- *Top-down stochastic stage*: Once the deterministic representations are obtained, we produce latent variables $\mathbf{Z}$ at each level. To maintain the parallelism of transformers, we adhere to (Sønderby et al. 2016; Aksan and Hilliges 2018) not establishing explicit dependencies between various time steps. Instead, we implicitly build temporal dependencies by conditioning a latent variable $\mathbf{Z}^{l-1}_t$ on its higher-level variable $\mathbf{Z}^l_t$ as shown in Figure 3(b), where $\mathbf{Z}^i_t \in \mathbb{R}^{N\times C}$ and $i = \{1, \dots, L\}$. In this way, the lower-level latent variables focus more on the local information, while the upper-level ones have a larger receptive field due to their corresponding deterministic input. In our model, the generation task is to predict the next time step given all past steps using the prior $p_\theta(\mathbf{Z}^l_t|\mathbf{X}_{1:t-1})$, and the inference task is to approximate the posterior $q_\phi\left(\mathbf{Z}^l_t|\mathbf{X}_{1:t}\right)$. As AirFormer belongs to the VAE family, we train our model by jointly optimizing the prediction loss and the ELBO.

In the following parts, we start by introducing the modules for spatio-temporal representation learning in the deterministic stage, i.e., DS-MSA and CT-MSA. Next, we elaborate on the generation and inference model of the stochastic stage. Finally, we delineate the optimization of our model.



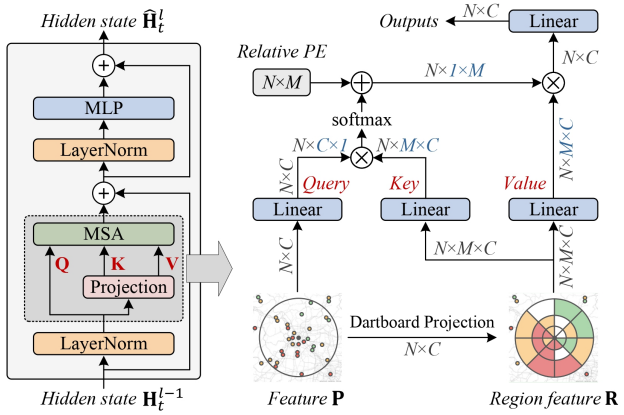
(a) Deterministic stage (b) Graphical view of the stochastic stage

Figure 3: AirFormer has two stages: (a) One to leverage two types of MSA to learn deterministic spatio-temporal representations. (b) The other to capture the uncertainty via latent random variables. The temporal receptive field of the states (i.e., $\mathbf{H}$ and $\mathbf{Z}$) is gradually increased from bottom to top.

Dartboard Spatial MSA (DS-MSA)

Motivation. Besides local emissions, the air quality of a place is impacted by its neighbors as air pollutants are dispersed among various locations. Given the hidden state of N stations at time t ($\mathbf{H}_t \in \mathbb{R}^{N\times C}$), the complexity of standard MSA for spatial modeling is $\mathcal{O}(N^2C)$, where C is the feature length. This quadratic cost makes it inefficient to accommodate with fine-grained data, e.g., 1,000+ stations.

Overview. We present a novel MSA termed Dartboard Spatial MSA (DS-MSA) for *efficiently* capturing the spatial dependencies among different locations at each time step. Notably, our DS-MSA possesses a large receptive field while only taking *linear* complexity w.r.t the number of stations, which is inspired by the spatial feature extraction in (Zheng et al. 2015). Figure 4(a) presents the pipeline of DS-MSA. At the l-th block, DS-MSA takes the hidden state $\mathbf{H}^{l-1}_t$ as inputs. The features are first normalized and used to generate the queries with a linear layer. For each query station, we project its surroundings into the dartboard distribution to obtain the keys and values. As a result, the number of keys (or values) is reduced to M, where M is the region number. We then perform MSA to learn spatial dependencies and finally utilize MLPs to produce the output $\hat{\mathbf{H}}^l_t \in \mathbb{R}^{N\times C}$.



(a) DS-MSA (b) Dartboard projection and MSA

Figure 4: Pipeline of DS-MSA. PE: position encoding.

Dartboard Projection & MSA. For a certain station i, we unify the formulation of the dartboard projection by introducing a *projection matrix* $\mathbf{A}_i \in \mathbb{R}^{M\times N}$ which denotes how nearby stations are mapped to M regions bound by the line fragments and circles, where each entry $a_{jk} \geq 0$ denotes the likelihood that the k-th station belongs to the region j. $\mathbf{A}_i$ has each row summing to one and each non-zero entry in the same row is evenly distributed (like average pooling).

Figure 2 shows an example in which we partition the surroundings of a query station in a dartboard fashion. The outmost circle has the largest semidiameter (e.g. 500km), while the innermost one has the smallest (e.g., 50km). The three circles focus on a common center (the query station) and are further divided by four lines heading in various directions. The stations falling outside the outmost circle are not considered for MSA. In this case, we have $M = 24 + 1 = 25$ ($+1$ for including the query station itself).

Assume the inputs of dartboard projection is $\mathbf{P} \in \mathbb{R}^{N \times C}$, we project the station features into regional representations that correspond to each station via the assignment matrix:

$$\mathbf{R}_i = \mathbf{A}_i \mathbf{P}, \quad \mathbf{R} = [\mathbf{R}_1, \mathbf{R}_2, \cdots, \mathbf{R}_N], \tag{3}$$

where $[\cdot, \cdot]$ is concatenation; $\mathbf{R} \in \mathbb{R}^{N \times M \times C}$ is the regional features after projection. Next, we generate the queries, keys and values using linear layers and perform MSA to capture the spatial relations between the query station and its corresponding regions (notations are detailed in Eq. 1):

$$\mathbf{X}_h = \text{Softmax}\left(\alpha \mathbf{Q}_h \mathbf{K}_h^\top + \mathbf{B}_h\right) \mathbf{V}_h, \tag{4}$$

where $\mathbf{B}_h \in \mathbb{R}^{N \times M}$ is learnable relative position encoding (Wu et al. 2021) to incorporate position information. We can also encode the impacts of external factors (e.g., wind direction and speed) in $\mathbf{B}_h$ to improve performance. In particular, the regions without any stations are *masked* during MSA. Figure 4(b) can help better understand this process.

Discussion. DS-MSA is designed by jointly considering the following factors: 1) *Spatial dependencies*: Considering the domain knowledge of air pollution dispersion that the spatial correlations between nearby locations are always stronger than those far away, each station attends to its surroundings in a dartboard fashion, i.e., close places at a fine granularity and the distant regions at a coarse granularity. 2) *Efficiency*: Since the number of regions M is small ($M \ll N$), the computational complexity $\mathcal{O}(MNC)$ grows linearly with the increase of the number of stations, which is more efficient than standard MSA. 3) *Lightweight*: By using the dartboard projection, DS-MSA does not introduce extra learnable parameters into standard MSA, thus being lightweight in practice.

Causal Temporal MSA (CT-MSA)

In addition to spatial dependencies, the air quality of a location depends on its history. Given the hidden state of a location across all past steps $\hat{\mathbf{H}}^l_{:,n} \in \mathbb{R}^{T \times C}$ (the output of DS-MSA), adopting MSA to learn temporal dependencies induces quadratic cost w.r.t the number of time steps T. Here, we propose Causal Temporal MSA (CT-MSA) as a strong and efficient alternative to standard MSA for temporal modeling. As depicted in Figure 5, its pipeline is mostly identical to standard MSA. Leveraging the domain knowledge of time series, CT-MSA has two major modifications:

Local window. Considering that nearby time steps usually have stronger correlations than faraway slots, we perform MSA within non-overlapping windows to capture local interactions among time steps, leading to $\mathcal{O}(TWC)$ cost ($\frac{T}{W}$ times less than standard MSA), where W is the window size. To preserve the large receptive field of standard MSA, we gradually increase the window size in different AirFormer blocks (from bottom to top), as shown in Figure 5.

Temporal causality. Since the air quality at the current step is not conditioned on its future, we follow WaveNet (Oord et al. 2016) to introduce causality into MSA (see Figure 5), which ensures the model cannot violate the temporal order of input data. Such causality can be easily implemented by masking the specific entries in the attention map. To allow position-aware MSA, we add learnable absolute position encoding (Vaswani et al. 2017) to the input of CT-MSA.

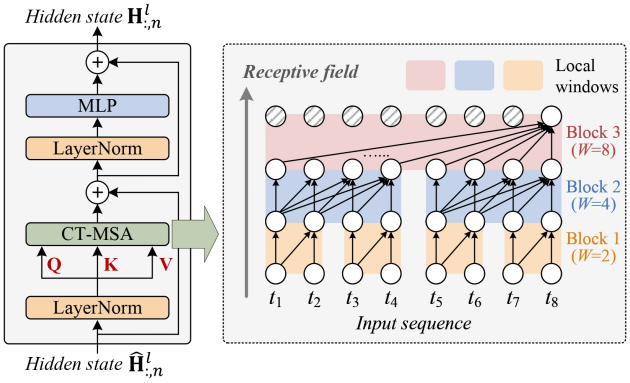


Figure 5: Left: CT-MSA. Right: An example of CT-MSA with local windows of size 2, 4, and 8 at each block.

Top-Down Stochastic Stage

After obtaining the deterministic representations, we build latent random variables to learn the *uncertainty* of air quality data, e.g., unpredictable factors and noisy observations.

Generation Model. As shown in Figure 3(b), the generation model aims to predict the next step given all *past* steps. As we have encoded the spatial dependencies among locations in deterministic states $\mathbf{H}_t$, we can factorize the prior distribution of a set of random variables $\mathcal{Z}_t = \{\mathbf{Z}^1_t, \ldots, \mathbf{Z}^L_t\}$ as:

$$\begin{aligned} p_\theta\left(\mathcal{Z}_t | \mathbf{X}_{1:t-1}\right) &= \prod_{n=1}^{N} p_\theta\left(\left\{\mathbf{z}^1_{t,n}, \ldots, \mathbf{z}^L_{t,n}\right\} | \mathbf{X}_{1:t-1}\right) \\ &= \prod_{n=1}^{N} p_\theta\left(\mathbf{z}^L_{t,n} | \mathbf{h}^L_{t-1,n}\right) \prod_{l=1}^{L-1} p_\theta\left(\mathbf{z}^l_{t,n} | \mathbf{z}^{l+1}_{t,n}, \mathbf{h}^l_{t-1,n}\right), \end{aligned} \tag{5}$$

where $\mathbf{z}^l_{t,n} \in \mathbb{R}^C$ and $\mathbf{h}^l_{t,n} \in \mathbb{R}^C$ is the n-th row of $\mathbf{Z}^l_t$ and $\mathbf{H}^l_t$, respectively. In Eq. (5), we follow VAE to set the prior distribution at each layer as a Gaussian distribution, i.e.,

$$p_\theta\left(\mathbf{z}^l_{t,n} | \mathbf{z}^{l+1}_{t,n}, \mathbf{h}^l_{t-1,n}\right) = \mathcal{N}\left(\mu^l_t, \sigma^l_t\right), \tag{6}$$

where the mean μ^l_t and the diagonal covariance σ^l_t are parameterized by a neural network $f^l(\mathbf{z}^{l+1}_{t,n}, \mathbf{h}^l_{t-1,n})$ shared by all locations. In this way, we implicitly build connections between hidden states at different time steps, as the upper-layer random variables (e.g., $\mathbf{z}^{l+1}_{t,n}$) contain more contextual information due to their larger receptive field.

Inference Model. In contrast, the inference model (see Figure 3(b)) is applied to approximate the posterior distribution of $\mathcal{Z}_t$ given both the *current* and *previous* steps:

$$\begin{aligned} q_\phi\left(\mathcal{Z}_t | \mathbf{X}_{1:t}\right) &= \prod_{n=1}^{N} q_\phi\left(\mathbf{z}^L_{t,n} | \mathbf{h}^L_{t,n}\right) \prod_{l=1}^{L-1} q_\phi\left(\mathbf{z}^l_{t,n} | \mathbf{z}^{l+1}_{t,n}, \mathbf{h}^l_{t,n}\right), \\ \text{where} \quad q_\phi\left(\mathbf{z}^l_{t,n} | \mathbf{z}^{l+1}_{t,n}, \mathbf{h}^l_{t,n}\right) &= \mathcal{N}\left(\hat{\mu}^l_t, \hat{\sigma}^l_t\right). \end{aligned} \tag{7}$$

Eq. (7) employs the same factorization as Eq. (5). The arguments of the Gaussian distribution at each layer is parameterized by a neural network $g^l(\mathbf{z}^{l+1}_{t,n}, \mathbf{h}^l_{t,n})$. Following a similar procedure as the generation model, the random variables generated by the posterior distribution can also effectively consider the spatio-temporal dependencies within air quality, thereby enhancing the predictive performance.

Prediction & Optimization

AirFormer makes predictions based on both the deterministic and stochastic hidden states at time T using an MLP:

$$\hat{\mathbf{Y}}_{1:\tau} = \text{MLP}([\mathbf{H}_T^1, \dots, \mathbf{H}_T^L, \mathbf{Z}_T^1, \dots, \mathbf{Z}_T^L]). \quad (8)$$

To train our model, we jointly optimize two loss functions:

$$\mathcal{L} = \mathcal{L}_{pred} + \mathcal{L}_{\text{ELBO}}. \quad (9)$$

$\mathcal{L}_{pred}$ denotes the L1 loss between the prediction $\hat{\mathbf{Y}}_{1:\tau}$ and the corresponding ground truth $\mathbf{Y}_{1:\tau}$; the second term $\mathcal{L}_{\text{ELBO}}$ indicates the sum of *negative* ELBO at all historical steps. Concretely, $\mathcal{L}_{\text{ELBO}}$ at each step is computed as:

$$\begin{aligned}\mathcal{L}_{\text{ELBO}}(\theta, \phi; t) &= \mathcal{L}_{rec}(\theta, \phi; t) + \mathcal{L}_{kl}(\theta, \phi; t) \\ &= -\mathbb{E}_{q_\phi(\mathcal{Z}_t|\mathbf{X}_t)}\left[\log p_\theta\left(\mathbf{X}_t|\mathcal{Z}_t\right)\right] \\ &\quad + KL\left(q_\phi\left(\mathcal{Z}_t|\mathbf{X}_{1:t}\right) \| p_\theta\left(\mathcal{Z}_t|\mathbf{X}_{1:t-1}\right)\right),\end{aligned} \quad (10)$$

where the first term is the reconstruction likelihood and the second term is the KL divergence between the prior and the posterior; $\mathcal{Z}_t = \{\mathbf{Z}_t^1, \dots, \mathbf{Z}_t^L\}$ is a set of latent variables at each layer. We further simplify the KL term ($\mathcal{L}_{kl}$) using the factorization in Eq. (5) and Eq. (7), resulting in

$$\mathcal{L}_{kl} = \sum_{n=1}^{N} KL\left(q_\phi\left(\mathbf{z}_{t,n}^L|\mathbf{h}_{t,n}^L\right) \| p_\theta\left(\mathbf{z}_{t,n}^L|\mathbf{h}_{t-1,n}^L\right)\right) + \sum_{n=1}^{N}\sum_{l=1}^{L-1} \mathbb{E}_{q_\phi}\left[KL\left(q_\phi\left(\mathbf{z}_{t,n}^l|\mathbf{z}_{t,n}^{l+1}, \mathbf{h}_{t,n}^l\right) \| p_\theta\left(\mathbf{z}_{t,n}^l|\mathbf{z}_{t,n}^{l+1}, \mathbf{h}_{t-1,n}^l\right)\right)\right]$$

Experiments

Dataset Description

Our system collected nationwide air quality data and meteorological data for a four-year period (Jan. 1, 2015 to Dec. 31, 2018) throughout Chinese mainland. The system details can be found in (Yi et al. 2018).

- *Air quality data*: We collected hourly air quality data from 1,976 stations covering 342 cities in China. Each air quality record contains the concentration of six types of pollutants, including $PM_{2.5}$, PM_{10}, NO_2, CO, O_3, and SO_2. Among them, the primary pollutant of air quality is $PM_{2.5}$, thus we employ its reading as the *prediction target*.
- *Meteorological data*: This dataset is collected every hour from 2,575 monitoring stations in China. Each record consists of weather conditions (e.g., sunny, rainy), temperature, humidity, wind speed, and wind direction.

We choose the air quality monitoring stations with a missing rate of $PM_{2.5}$ less than 20% to conduct our experiments, leading to a 1,085-station dataset. Figure 1 illustrates the station distribution. Our dataset has extensive coverage in both space and time scales compared to previous datasets (see Table 1). Following the prior studies (Liang et al. 2018; Wang et al. 2020), we predict the concentration of $PM_{2.5}$ over the next 24 steps given the past 24 steps, where each step stands for 3 hours. In other words, we perform a 72-hour future prediction based on the readings from the past 72 hours. The dataset is partitioned in chronological order with the first two years for training, the third year for validation, and the last year for testing. Z-score normalization is applied to the model inputs for fast convergence.

Dataset	Venue	#Node	Range	Scale	Joint?
(Zheng et al. 2015)	KDD	35	48m	City	✗
(Yi et al. 2018)	KDD	35	48m	City	✗
(Lin et al. 2018)	GIS	35	14m	City	✓
(Wang et al. 2020)	GIS	184	48m	Nation	✓
(Xu et al. 2021)	preprint	56	12m	Province	✓
(Chen et al. 2021)	preprint	209	28m	Nation	✓
Our dataset	AAAI	**1,085**	48m	Nation	✓

Table 1: Dataset comparison. m: month. Joint means collectively predicting the air quality at all locations.

Implementation Details

We implement our model by PyTorch 1.10 using a Quadro RTX A6000 GPU. The Adam optimizer (Kingma and Ba 2014) is utilized to train our model, and the batch size is 16. The learning rate starts from 5×10^{-4}, halved every three epochs. The number of AirFormer blocks is set to 4 with local window size $W = 3, 6, 12, 24$, respectively. For the hidden dimension C in DS-MSA and CT-MSA, we conduct a grid search over $\{8, 16, 32, 64\}$, and $C = 32$ obtains the best result. The number of heads in DS-MSA and CT-MSA is 2. In DS-MSA, we partition the space into regions by two circles with 50 and 200km semidiameter, respectively. f and g in the stochastic stage are parameterized by 3-layer MLPs.

Baselines for Comparison

We compare our AirFormer with the following baselines that belong to the following four categories:

- *Classical methods*: **HA** is history average which predicts air quality by the average value of historical readings in the corresponding periods. **VAR** is vector autoregression.
- *STGNN variants*: we also take some popular STGNNs as baselines, i.e., **DCRNN** (Li et al. 2017), **STGCN** (Yu, Yin, and Zhu 2018), **GWNET** (Wu et al. 2019), and **MTGNN** (Wu et al. 2020). They generalize well to our application.
- *Attention-based models*: **ASTGCN** (Guo et al. 2019) and **GMAN** (Zheng et al. 2020) are attention-based baselines for spatio-temporal forecasting. **STTN** (Xu et al. 2020) is a variant of transformers for traffic forecasting, which can be easily adapted to air quality prediction.
- *Air quality prediction*: we select three strong models for a comparison, including **DeepAir** (Yi et al. 2018), **$PM_{2.5}$-GNN** (Wang et al. 2020) and **GAGNN** (Chen et al. 2021). Some (Pan et al. 2018; Wang et al. 2021, 2022; Liang et al. 2022) are omitted due to out-of-GPU-memory (OOM).

For a fair comparison, we tune different hyperparameters for all baselines, finding the best setting for each.

Evaluation Metrics

We leverage Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) for evaluation, where a smaller metric means better performance. Moreover, we follow (Zheng et al. 2015; Yi et al. 2018) to discuss the errors on predicting sudden changes. The sudden changes are defined as the cases where $PM_{2.5}$ is larger than 75 $\mu g/m^3$ and changes more than a threshold (i.e., ± 20 $\mu g/m^3$) in the next three hours.

Model	#Param (K)	1-24h		25-48h		49-72h		Sudden change	
		MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
HA (Zhang, Zheng, and Qi 2017)	-	31.25	62.52	31.19	62.49	31.16	62.49	72.89	132.42
VAR (Toda 1991)	-	29.91	52.10	31.61	64.59	30.18	65.10	70.86	114.78
DCRNN (Li et al. 2017)	397	19.35	46.40	24.06	57.38	25.30	58.24	63.22	112.30
STGCN (Yu, Yin, and Zhu 2018)	453	19.06	42.69	24.09	56.50	25.10	58.96	61.35	111.89
GWNET (Wu et al. 2019)	825	17.79	39.49	23.32	53.17	25.00	57.01	59.33	105.75
MTGNN (Wu et al. 2020)	207	18.15	38.99	23.47	52.21	24.77	55.73	59.24	103.71
ASTGCN (Guo et al. 2019)	4,812	20.76	50.29	24.37	56.04	25.22	57.77	63.19	114.57
GMAN (Zheng et al. 2020)	269	19.60	45.70	23.79	54.25	24.89	56.33	61.83	109.57
STTN (Xu et al. 2020)	188	18.22	37.44	24.16	52.91	25.35	56.14	60.36	105.20
DeepAir (Yi et al. 2018)	183	17.47	39.12	23.40	53.48	24.95	56.92	60.26	109.95
$PM_{2.5}$-GNN (Wang et al. 2020)	101	20.20	48.96	25.04	59.56	26.31	60.46	63.64	119.00
GAGNN (Chen et al. 2021)	412	19.53	45.68	24.56	58.59	25.56	59.61	64.38	116.51
AirFormer (ours)	246	**16.03***	**32.36***	**21.65***	**44.67***	**23.64***	**50.22***	**54.92***	**90.15***

Table 2: 5-run results. The magnitude of #Param (the number of parameters) is Kilo. The bold/underlined font mean the best/the second best result. * denotes the improvement over the second best model is statistically significant at level 0.01 (Kim 2015).

Model Comparison

In this section, we perform a model comparison in terms of MAE and RMSE. We run each method five times and report the average metric of each model. As shown in Table 2, AirFormer *significantly* outperforms all competing baselines on both metrics according to the Student's T-test (Kim 2015) at level 0.01. In contrast to the second best method (i.e., DeepAir), AirFormer reduces the MAE by 8.2%, 7.5% and 5.3% on 24-, 48- and 72-hour future prediction, respectively. When predicting the sudden changes, AirFormer achieves at least 7.3% and 13.1% improvements on MAE and RMSE thanks to the robustness provided by stochastic latent spaces.

From Table 2, we can also observe that: 1) Deep-learning-based approaches surpass the classical methods (HA and VAR) by a large margin due to their greater learning capacity. 2) Although STGNN-based models, e.g., GWNET and MTGNN, are originally evaluated on traffic forecasting, they also generalize well on air quality prediction, showing comparable results to the second best method DeepAir. 3) Our AirFormer clearly outperforms STTN which is also a transformer model. This verifies that the domain knowledge of air pollution not only helps us design our model with more interpretations, but also enhances the predictive accuracy.

Ablation Study

Effects of DS-MSA. To study the effects of DS-MSA, we consider the following variants for comparison: a) **w/o DS-MSA**: We turn off DS-MSA in AirFormer, i.e., no spatial modeling in this model. b) **MSA**: We replace DS-MSA with standard MSA. c) **Local MSA**: DS-MSA is substituted by local MSA in which each query station attends to its neighbors within 500km. The results are shown in the upper part of Table 3. First, we find removing DS-MSA leads to a significant degradation on MAE, revealing the great importance of addressing spatial dependencies. Second, DS-MSA obtains lower prediction errors over all future horizons, while running **39%** faster than vanilla MSA and 21% faster than local MSA. This merit suggests that our DS-MSA has great potential to be a basic building block for capturing spatial dependencies within air quality data in practice.

We then discuss different settings of the dartboard. In Table 3, r_1-r_2-r_3 means dividing the space by 3 circles with r_1, r_2, and r_3 km semidiameter. Compared to the 1-circle partition with 50km semidiameter, the 50-200 dartboard achieves lower errors due to a larger receptive field. Despite the lowest MAE of 50-200-500, it results in more computational costs (33.8% slower than 50-200). To make a better trade-off, we choose 50-200 as our default setting.

To further investigate DS-MSA, we perform a case study on the 50-200 dartboard centered by Xizhimen (a hybrid district in Beijing). In Fig. 6, the attention weights are dispersed (almost <0.2) when there is no wind. If the wind blows from the east or southwest, the attention weights concentrate more on the corresponding directions. In light of this, DS-MSA is not only effective but can also be easily interpreted.

Variant	Time/epoch	1-24h	25-48h	49-72h
w/o DS-MSA	1,284	17.24	23.55	24.96
MSA	2,818	18.19	23.24	24.23
Local MSA	2,157	18.63	23.79	24.98
DS-MSA (50-200)	1,708	**16.03**	21.65	23.64
DS-MSA (50)	1,547	16.43	22.87	24.32
DS-MSA (50-200-500)	2,285	16.09	**21.30**	**22.85**

Table 3: Effects of DS-MSA over MAE.

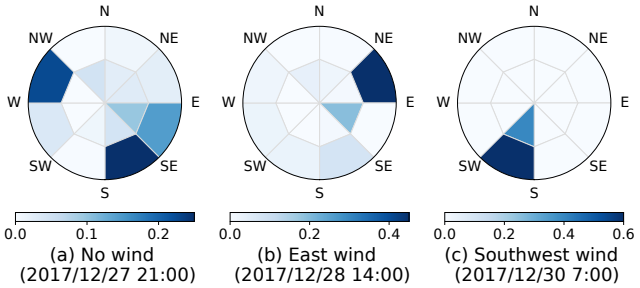


Figure 6: Visualization of DS-MSA at the first block. We omit the attention weight at the center station (query) itself.

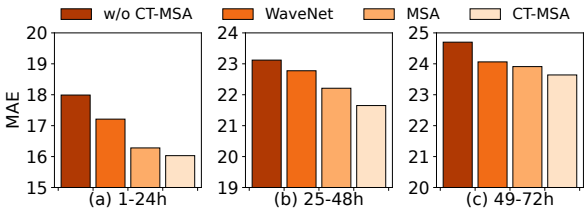


Figure 7: Effects of CT-MSA on MAE.

Varaint	Time/epo	1-72h	Sudden change
w/o Stochastic	1,559	20.97	57.52
AirFormer	1,708	**20.44**	**54.92**

Table 4: Effects of latent variables on MAE. w/o: without.

Effects of CT-MSA. To examine the efficacy of CT-MSA for capturing temporal dependencies, we compare our model with its variants integrated with various temporal modules: a) **w/o CT-MSA**: we remove CT-MSA from our AirFormer. b) **MSA**, **WaveNet**: we replace CT-MSA by standard MSA or WaveNet (Wu et al. 2019). The results are shown in Figure 7. Primarily, all the variants with temporal modules perform much better than w/o CT-MSA, verifying the necessity of temporal modeling. Moreover, both MSA-based methods surpass WaveNet, which reveals the superiority of MSA in air quality modeling. Remarkably, integrating causality and local windows into MSA consistently improves the performance over all future steps (see MSA vs. CT-MSA).

Effects of Latent Variables. As a crucial component of AirFormer, the stochastic stage empowers our model to capture the uncertainty within air quality data, thus boosting the performance. To investigate its effectiveness, we compare our AirFormer with its variant that turns off the stochastic stage. As shown in Table 4, integrating latent variables reduces the MAE on sudden changes by 4.5% while introducing little additional time (149 seconds) per training epoch.

Hyperparameter Study. Next, we investigate the effects of the number of AirFormer blocks L, the hidden dimension C, and the number of heads N_h. Figure 8(a) shows the results of AirFormer with 2 or 4 blocks vs. different C, from which we observe: 1) Stacking more AirFormer blocks consistently achieves lower MAE. We encounter the OOM issue when $L > 4$ and thus set $L = 4$ as our default setting. 2) Using a very small C (e.g., 16) leads to worse performance due to its limited capacity. 3) $C = 64$ obtain similar accuracy. In Figure 8(b), we find that the performance of AirFormer is not sensitive to the number of heads when $N_h > 1$.

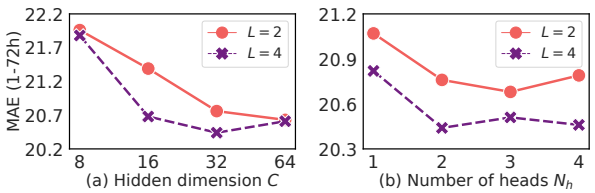


Figure 8: Effects of hyperparameters on MAE.

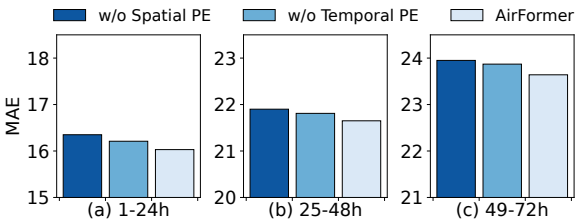


Figure 9: Effects of position encoding (PE) on MAE.

Effects of Position Encoding. Since MSA is permutation-invariant, we integrate position encoding (PE) into DS-MSA and CT-MSA to consider the order information. As depicted in Figure 9, removing either Spatial PE in DS-MSA or Temporal PE in CT-MSA will cause degenerated performance across all future horizons. Besides, the improvement of integrating Spatial PE is slightly higher than Temporal PE.

Related Works

Forecasting air quality is one of the core tasks in smart cities. **Physics-based models** represent the emission and dispersion of air pollutants as a dynamic system and simulate the process using numerical functions. They investigate the primary causes of air pollution, including chemicals, vehicles, and factories (Vardoulakis et al. 2003; Arystanbekova 2004; Daly and Zannetti 2007). However, it is non-trivial to acquire such a variety of data sources precisely.

Data-driven approaches have recently emerged as the most popular approach for air quality forecasting. This research line uses parameterized models, such as neural networks, to capture the spatio-temporal dependencies within air quality data. As opposed to physics-based models, they require far less complex domain knowledge and are usually more flexible. Typically, Zheng et al. (2015) developed a hybrid data-driven model that ensembles the prediction results from different views. Leveraging the large capacity of deep neural networks, DeepAir outperformed existing shallow models on both short and long-term predictions (Yi et al. 2018). Some follow-ups proposed either STGNNs (Wang et al. 2020) or attention-based models (Wang et al. 2021, 2022) to better address the spatio-temporal dependencies. These works, however, present some difficulties (e.g., inefficiency, degenerated performance) on a nationwide forecasting task. Besides, there are a stream of studies (Pan et al. 2019; Li et al. 2020; Liu et al. 2021; Pan et al. 2021; Shao et al. 2022) exploring new learning paradigms for spatio-temporal data.

Conclusion and Future Work

We have devised a transformer model for nationwide air quality prediction in China. To the best of our knowledge, this is the first work for collectively forecasting air quality among thousands of locations. Our model elaborately combines the spatio-temporal learning capabilities of transformers with the uncertainty measurement of stochastic latent spaces. Compared to prior methods, our model reduces the prediction errors by 4.6-8.2%. In the future, we will explore online learning and deploy our model to support public use.

Acknowledgments

This research was supported by the MSIT (Ministry of Science, ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2022-2020-0-01789) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation). It was also supported by the National Natural Science Foundation of China (62172034) and the Beijing Nova Program (Z201100006820053).

References

Aksan, E.; and Hilliges, O. 2018. STCN: Stochastic Temporal Convolutional Networks. In *International Conference on Learning Representations*.

Arystanbekova, N. K. 2004. Application of Gaussian plume models for air pollution simulation at instantaneous emissions. *Mathematics and Computers in Simulation*, 67(4-5): 451–458.

Chen, L.; Xu, J.; Wu, B.; Qian, Y.; Du, Z.; Li, Y.; and Zhang, Y. 2021. Group-aware graph neural network for nationwide city air quality forecasting. *arXiv preprint arXiv:2108.12238*.

Cheng, W.; Shen, Y.; Zhu, Y.; and Huang, L. 2018. A neural attention model for urban air quality inference: Learning the weights of monitoring stations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Daly, A.; and Zannetti, P. 2007. Air pollution modeling–An overview. *Ambient air pollution*, 15–28.

Guo, S.; Lin, Y.; Feng, N.; Song, C.; and Wan, H. 2019. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In *AAAI*, volume 33, 922–929.

Kim, T. K. 2015. T test as a parametric statistic. *Korean journal of anesthesiology*, 68(6): 540–546.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kingma, D. P.; and Welling, M. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Li, T.; Zhang, J.; Bao, K.; Liang, Y.; Li, Y.; and Zheng, Y. 2020. Autost: Efficient neural architecture search for spatio-temporal prediction. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 794–802.

Li, Y.; Yu, R.; Shahabi, C.; and Liu, Y. 2017. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926*.

Liang, Y.; Ke, S.; Zhang, J.; Yi, X.; and Zheng, Y. 2018. GeoMAN: Multi-Level Attention Networks for Geo-Sensory Time Series Prediction. In *IJCAI*.

Liang, Y.; Ouyang, K.; Wang, Y.; Pan, Z.; Yin, Y.; Chen, H.; Zhang, J.; Zheng, Y.; Rosenblum, D. S.; and Zimmermann, R. 2022. Mixed-Order Relation-Aware Recurrent Neural Networks for Spatio-Temporal Forecasting. *IEEE Transactions on Knowledge and Data Engineering*.

Lin, Y.; Mago, N.; Gao, Y.; Li, Y.; Chiang, Y.-Y.; Shahabi, C.; and Ambite, J. L. 2018. Exploiting spatiotemporal patterns for accurate air quality forecasting using deep learning. In *SIGSPATIAL*, 359–368.

Liu, X.; Liang, Y.; Zheng, Y.; Hooi, B.; and Zimmermann, R. 2021. Spatio-temporal graph contrastive learning. *arXiv preprint arXiv:2108.11873*.

Oord, A. v. d.; Dieleman, S.; Zen, H.; Simonyan, K.; Vinyals, O.; Graves, A.; Kalchbrenner, N.; Senior, A.; and Kavukcuoglu, K. 2016. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*.

Pan, Z.; Ke, S.; Yang, X.; Liang, Y.; Yu, Y.; Zhang, J.; and Zheng, Y. 2021. AutoSTG: Neural Architecture Search for Predictions of Spatio-Temporal Graph. In *Proceedings of the Web Conference 2021*, 1846–1855.

Pan, Z.; Liang, Y.; Wang, W.; Yu, Y.; Zheng, Y.; and Zhang, J. 2019. Urban traffic prediction from spatio-temporal data using deep meta learning. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1720–1730.

Pan, Z.; Liang, Y.; Zhang, J.; Yi, X.; Yu, Y.; and Zheng, Y. 2018. Hyperst-net: Hypernetworks for spatio-temporal forecasting. *arXiv preprint arXiv:1809.10889*.

Shao, Z.; Zhang, Z.; Wang, F.; and Xu, Y. 2022. Pre-training Enhanced Spatial-temporal Graph Neural Network for Multivariate Time Series Forecasting. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 1567–1577.

Sønderby, C. K.; Raiko, T.; Maaløe, L.; Sønderby, S. K.; and Winther, O. 2016. Ladder variational autoencoders. *Advances in neural information processing systems*, 29.

Toda, H. 1991. *Vector autoregression and causality*. Yale University.

Vallero, D. A. 2014. *Fundamentals of air pollution*. Academic press.

Vardoulakis, S.; Fisher, B. E.; Pericleous, K.; and Gonzalez-Flesca, N. 2003. Modelling air quality in street canyons: a review. *Atmospheric environment*, 37(2): 155–182.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in neural information processing systems*, 5998–6008.

Wang, C.; Zhu, Y.; Zang, T.; Liu, H.; and Yu, J. 2021. Modeling inter-station relationships with attentive temporal graph convolutional network for air quality prediction. In *WSDM*, 616–634.

Wang, S.; and Hao, J. 2012. Air quality management in China: Issues, challenges, and options. *Journal of Environmental Sciences*, 24(1): 2–13.

Wang, S.; Li, Y.; Zhang, J.; Meng, Q.; Meng, L.; and Gao, F. 2020. PM2.5-GNN: A Domain Knowledge Enhanced Graph Neural Network For PM2.5 Forecasting. In *SIGSPATIAL*, 163–166.

Wang, S.; Qiao, L.; Fang, W.; Jing, G.; Sheng, V. S.; and Zhang, Y. 2022. Air Pollution Prediction Via Graph Attention Network and Gated Recurrent Unit. *Computers, Materials and Continua*, 73(1): 673–687.

Wen, Q.; Zhou, T.; Zhang, C.; Chen, W.; Ma, Z.; Yan, J.; and Sun, L. 2022. Transformers in time series: A survey. *arXiv preprint arXiv:2202.07125*.

Wu, K.; Peng, H.; Chen, M.; Fu, J.; and Chao, H. 2021. Rethinking and improving relative position encoding for vision transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 10033–10041.

Wu, Z.; Pan, S.; Long, G.; Jiang, J.; Chang, X.; and Zhang, C. 2020. Connecting the dots: Multivariate time series forecasting with graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 753–763.

Wu, Z.; Pan, S.; Long, G.; Jiang, J.; and Zhang, C. 2019. Graph WaveNet for Deep Spatial-Temporal Graph Modeling. In *IJCAI*, 1907–1913.

Xu, J.; Chen, L.; Lv, M.; Zhan, C.; Chen, S.; and Chang, J. 2021. Highair: A hierarchical graph neural network-based air quality forecasting method. *arXiv preprint arXiv:2101.04264*.

Xu, M.; Dai, W.; Liu, C.; Gao, X.; Lin, W.; Qi, G.-J.; and Xiong, H. 2020. Spatial-temporal transformer networks for traffic flow forecasting. *arXiv preprint arXiv:2001.02908*.

Yi, X.; Zhang, J.; Wang, Z.; Li, T.; and Zheng, Y. 2018. Deep distributed fusion network for air quality prediction. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 965–973.

Yu, B.; Yin, H.; and Zhu, Z. 2018. Spatio-temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting. In *IJCAI*.

Zhang, J.; Zheng, Y.; and Qi, D. 2017. Deep Spatio-Temporal Residual Networks for Citywide Crowd Flows Prediction. In *AAAI*, 1655–1661.

Zhao, L.; Sun, Q.; Ye, J.; Chen, F.; Lu, C.-T.; and Ramakrishnan, N. 2015. Multi-task learning for spatio-temporal event forecasting. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, 1503–1512.

Zheng, C.; Fan, X.; Wang, C.; and Qi, J. 2020. Gman: A graph multi-attention network for traffic prediction. In *AAAI*, volume 34, 1234–1241.

Zheng, Y.; Yi, X.; Li, M.; Li, R.; Shan, Z.; Chang, E.; and Li, T. 2015. Forecasting fine-grained air quality based on big data. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, 2267–2276.

Zhou, T.; Ma, Z.; Wen, Q.; Wang, X.; Sun, L.; and Jin, R. 2022. FEDformer: Frequency Enhanced Decomposed Transformer for Long-term Series Forecasting. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, 27268–27286.