Eulerian Neural Network Informed by Chemical Transport for Air Quality Forecasting

Xukai Zhang[1], Shuliang Wang[1], Guangyin Jin[2], Ziqiang Yuan[1], Hanning Yuan[1]*, Sijie Ruan[1]*
[1]Beijing Institute of Technology, [2]Sapienza University of Rome
{xkzhang,slwang2011,ziqiangy,yhn6,sjruan}@bit.edu.cn
jinguangyin96@gmail.com

Abstract

Air pollution remains one of the most critical environmental challenges globally, posing severe threats to public health, ecological sustainability, and climate governance. While existing physics-based and data-driven models have made progress in air quality forecasting, they often struggle to jointly capture the complex spatiotemporal dynamics and ensure spatial continuity of pollutant distributions. In this study, we introduce CTENet, a novel chemical transport deep learning model that embeds the Advection-Diffusion-Reaction equation into a Physics-Informed Neural Network (PINN) framework using an Eulerian representation to model the spatiotemporal evolution of pollutants. Extensive experiments on two real-world datasets demonstrate that CTENet consistently outperforms state-of-the-art (SOTA) baselines, achieving a remarkable RMSE improvement of 45.8% on the USA dataset and 21.0% on the China dataset.

1 Introduction

Air quality refers to the concentration of pollutantssuch as $PM_{2.5}$, PM_{10}, ozone (O_3), sulfur dioxide (SO_2), nitrogen oxides (NO_x), and carbon monoxide (CO), and their impact on human health, ecosystems, and the environment. Exposure to these pollution is closely associated with respiratory and cardiovascular diseases, as well as increased risks of premature mortality[1, 2, 3]. Accurate air quality prediction is crucial for mitigating health risks, guiding public health, shaping policies, and enhancing environmental monitoring in smart, sustainable cities.

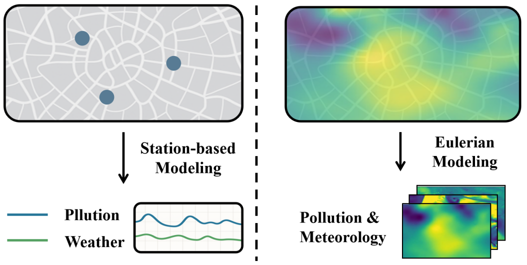
In recent years, extensive research has explored various modeling approaches for air quality prediction. Traditional physics-based models solve partial differential equations (PDEs) to simulate pollutant transport driven by atmospheric dynamics [4, 5]. While these models offer strong interpretability and theoretical grounding, they are often computationally expensive and sensitive to input accuracy. In contrast, simple data-driven models learn patterns directly from historical pollutant and meteorological observations [6, 7], but they may suffer from limited generalization in unseen or changing environments. Recent deep learning methods have further advanced this direction by modeling air quality as a multivariate time series at discrete stations, or by constructing graph representations of station networks to capture spatiotemporal dependencies [8, 9]. To improve predictive performance, these models typically incorporate auxiliary meteorological variablessuch as temperature and humiditycollected from ground-based monitoring sites. In addition, some multimodal approaches attempt to incorporate complementary information from heterogeneous sources to improve task understanding, robustness, and generalization [10, 11]. Beyond these data-driven models, physics-guided deep learning aims to combine the strengths of both approaches by incorporating physical knowledge into data-driven architectures[12, 13]. This hybrid paradigm seeks to enhance model accuracy and generalizability while ensuring physically plausible predictions.

*Sijie Ruan and Hanning Yuan are the corresponding authors of this paper.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

Despite considerable progress, several core challenges in air quality forecasting persist:

Structural mismatch between continuous Eulerian dynamics and discrete pollutant representations: Whether or not they incorporate physical knowledge, most deep learning methods for air quality prediction adopt station-based modeling paradigms, where data are represented as multivariate time series or discrete graphs[14, 15, 16, 17, 18], as illustrated in Figure 1a. While these approaches effectively capture temporal dependencies and localized station interactions, they neglect the spatial continuity modeling of pollutant distributions. This limits their capacity to model the smooth and dynamic spatiotemporal evolution of air pollution over larger regions, shown in Figure 1b. Additionally, the physical processes of diffusion and advection are inherently continuous and occur over continuous Eulerian space, rather than being naturally represented on discrete station graphs. Modeling these processes on graphs leads to a structural mismatch.



(a) Discrete pollutant representations

(b) continuous Eulerian representations

Figure 1: Two approaches to air pollution representation.

Incomplete representation of physicochemical drivers: Another critical limitation lies in the insufficient modeling of atmospheric chemical transformation processes and their meteorological drivers. While physical transport phenomena like advection and diffusion have received some attention in hybrid models[13, 19], the incorporation of complex atmospheric chemistrysuch as photochemical reactions and gas-to-particle conversion[20] that generate secondary pollutants (e.g., ozone and $PM_{2.5}$)remains limited , which are strongly modulated by meteorological factors such as solar radiation , boundary layer height, vertical temperature gradients. These variables are crucial for accurately simulating chemical reactions and phase transitions in secondary pollutant formation.

Unlike prior physics-guided models that rely on station-based or graph-based data, this paper proposes a novel approach performing Eulerian representation of pollutant evolution, enabling effective modeling of large-scale spatiotemporal diffusion, advection, and chemical transformations. It also integrates continuous meteorological data, including satellite-derived inversion, to enrich the modeling of underlying physicochemical dynamics.

Specifically, we propose the Chemical Transport Eulerian Network (CTENet), a deep learning architecture guided by chemical transport modeling (CTM) principles. CTENet embeds the Advection-Diffusion-Reaction (ADR) equation within a custom-designed neural network that operates on an Eulerian spatial domain, enabling direct simulation of pollutant transport and transformation over continuous space and time. The framework consists of several key modules: (1) the Eulerian Pollution Encoder, which encodes pollutant concentration fields within the continuous spatial domain; (2) the Meteorological Encoder, responsible for integrating high-dimensional meteorological data, including wind vector fields; (3) the Eulerian ADR Decoder, which decodes pollutant evolution through a physics-constrained differential equation network reflecting the ADR processes. In addition, we employ a modular and replaceable Spatiotemporal Sequence Predictor that functions as a Wind Predictor, Meteorology Predictor, and Pollutant Predictor within the overall framework.

In summary, the main contributions of this paper are as follows:

- We introduce a novel neural network, CTENet, which integrates a spatiotemporal sequence prediction network with a custom differential equation module, effectively capturing spatiotemporal variations in air quality. To the best of our knowledge, this is the first PINN approach for air quality prediction that adopts a spatial continuity perspective under the Eulerian framework.

- We integrate CTM with deep learning under an Eulerian framework to improve the modeling of spatiotemporal continuity in air quality data, explicitly simulating diffusion, advection, and reaction over continuous spatial domains for enhanced interpretability.

- We evaluate CTENet on real-world air quality datasets from China and the USA, achieving about 45.8% and 21.0% lower RMSE respectively compared to leading baselines.

2 Related Work

Air Quality Prediction. Air quality prediction is fundamental for smart city development and urban planning, attracting extensive research efforts. Existing methods generally fall into two categories: physics-based models and data-driven models. The former category characterizes air pollution dynamicssuch as diffusion, advection, and chemical transformationthrough differential equations solved by numerical methods like finite difference or finite element methods [4, 5]. While these models offer interpretability grounded in physical laws, they often require substantial computational resources and depend on high-quality input data. Data-driven models leverage historical pollutant data and contextual information, including meteorological variables, to learn spatiotemporal dependencies. Early machine learning methods such as random forests and support vector machines showed initial success [21] but were limited in capturing complex spatiotemporal patterns. With advances in deep learning, architectures combining convolutional and recurrent networks (e.g., CNN-LSTM) [22] and graph-based networks [15, 16] have significantly improved prediction accuracy by modeling spatial and temporal dependencies more effectively. Attention-based models [17] and recent methods like Airformer [18] further enhance large-scale spatial correlation modeling, but they do not explicitly model the continuous spatial distribution of pollutants. In addition, some studies, such as STFNN [23] and AirRadar [24], emphasize modeling the spatial continuity of atmospheric processes, exhibiting strong capability in reconstructing and reasoning about pollutant fields, although they focus on inferring pollutant distributions rather than forecasting.

Physics Guided Deep Learning. Recent advancements have explored the integration of physical principles into deep learning architectures [37, 38]. A prevalent approach involves incorporating physical knowledge into the loss function, wherein deviations from known physical laws are penalized to promote physical consistency and enhance model generalizability [39, 40]. However, the effectiveness of this method critically depends on the accuracy of the underlying physical knowledge. Inaccuracies can introduce inductive biases, thereby hindering the models representational capacity. Alternatively, hybrid frameworks have been proposed that embed scientific knowledge into specific components of the model architecture, enabling a more flexible and potentially more robust integration of physics into the learning process [41, 42]. In recent air quality forecasting studies, physics-guided frameworks have been increasingly adopted to better capture the underlying dynamics of pollutant transport and transformation. For example, some studies have incorporated advection-diffusion mechanisms and fluid dynamics constraints into learning models [12]. Other works introduced state-space concepts inspired by physical systems to represent temporal evolution [43]. Meanwhile, several approaches [13, 19] have modeled air quality dynamics as graph-based systems constrained by advectiondiffusion processes, enhancing interpretability and generalization. However, the physical processes are limited to the graph edges, without modeling spatial continuity.

3 Problem Formulation

Given historical pollutant concentration data $P_{1:T} \in \mathbb{R}^{T \times C_P \times N}$ from N observation stations located at spatial coordinates $\mathcal{S} = \{(h_n, w_n)\}_{n=1}^{N}$, and the corresponding continuous meteorological data $M_{1:T} \in \mathbb{R}^{T \times C_M \times H \times W}$, where C_M denotes the number of channels, including those for east-west and north-south wind components as well as other meteorological variables.

We aim to predict the pollutant concentrations at all station locations over the future time period from $T+1$ to $T+\tau$, denoted by $\hat{P}_{T+1:T+\tau}$. Formally, the task is to learn the function $\mathcal{F}$:

$$\hat{P}_{T+1:T+\tau} = \mathcal{F}(P_{1:T}, M_{1:T}), \quad \hat{P}_{T+1:T+\tau} \in \mathbb{R}^{\tau \times C_P \times N} \tag{1}$$

4 Methodology

4.1 Model Overview

In this paper, we propose the Chemical Transport Eulerian Network (CTENet), as illustrated in Figure 2. The framework comprises three key components: the Eulerian Pollution Encoder, the Meteorological Encoder, and the Eulerian ADR Decoder. Additionally, a replaceable Spatiotemporal Sequence Predictor functions as the Wind, Meteorology, and Pollutant Predictor in the framework.

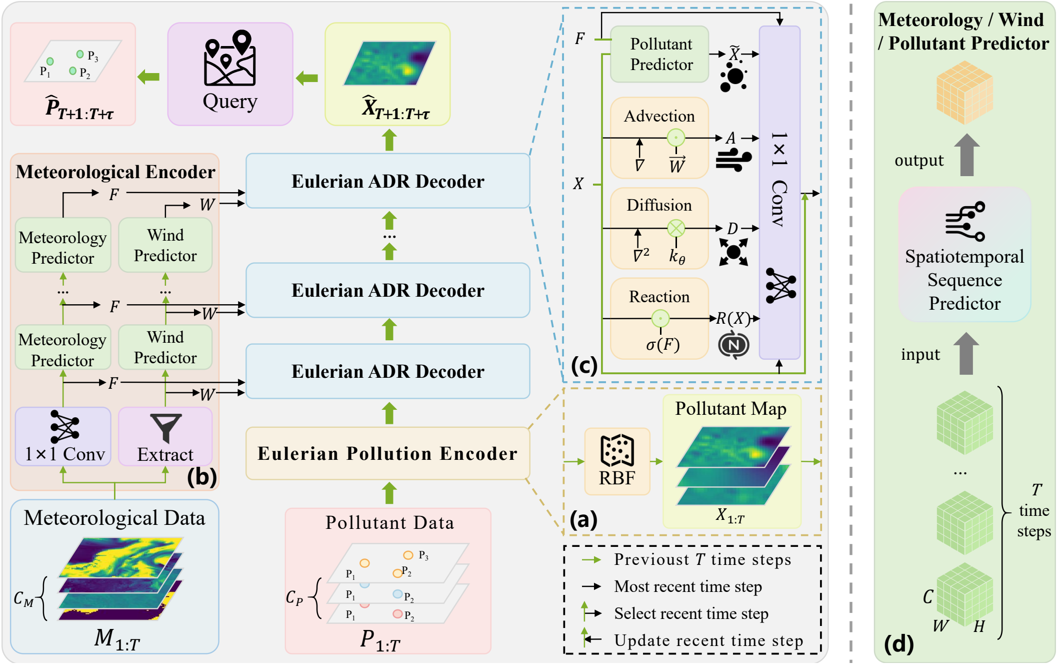


Figure 2: Structure of CTENet. Pollutant data is processed by the (a) Eulerian Pollution Encoder, while meteorological data are fed into the (b) Meteorological Encoder. The outputs are jointly passed to the multi-layer (c) Eulerian ADR Decoder, followed by a query step to generate the final output. (d) Spatiotemporal Sequence Predictor represents the general building block used in the three forecasting modules.

4.2 Eulerian Pollution Encoder

At each time step, pollutant concentrations from monitoring stations are assigned to their spatial locations, averaging values when multiple stations fall within the same pixel. However, these monitoring stations are sparsely distributed and cannot fully cover the entire spatial domain, as shown in Figure 3. For locations without monitoring stations, we perform Radial Basis Function (RBF) interpolation to generate a continuous Eulerian representation of pollutant distribution X, as shown in Figure 4. This continuous representation facilitates the modeling of spatial pollutant dynamics and serves as a proxy for learning air quality prediction over the entire domain.

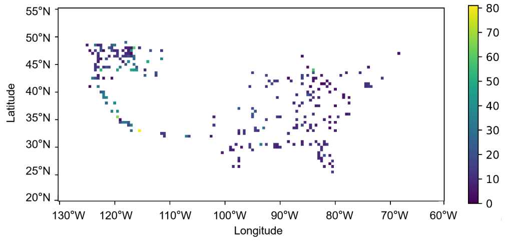


Figure 3: A sample of ground truth for PM$_{2.5}$ concentrations across the continental United States

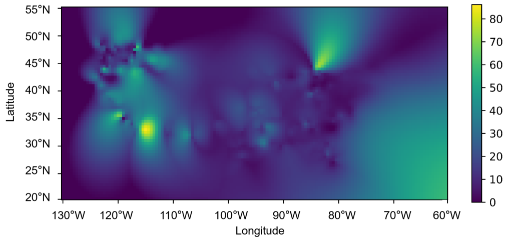


Figure 4: A sample of RBF interpolation result for PM$_{2.5}$ concentrations across the continental United States

Compared to other interpolation methods, RBF is particularly well-suited for modeling air pollutant data due to its flexibility, smoothness, and ability to handle irregularly distributed monitoring stations. It provides a kernel-based and distance-based interpolation framework that adapts well to the complex, non-linear spatial variability often observed in environmental data, making it ideal for constructing continuous inputs for deep learning models.

The fundamental idea of RBF interpolation is to infer a continuous field from discrete observations by expressing the spatial relationships among data points through radial basis functions. Specifically, the interpolated value at a spatial location $\mathbf{x}$ is computed as:

$$rbf(\mathbf{x}) = \sum_{i=1}^{n_t} \lambda_i^{(t)} \, \phi\left(\left\| \mathbf{x} - \mathbf{x}_i^{(t)} \right\| \right), \quad t \in \{1, \ldots, T\} \tag{2}$$

Here, $\mathbf{x}_i$ represents the position of the i-th known data point, and $\|\mathbf{x} - \mathbf{x}_i^{(t)}\|$ denotes the Euclidean distance between $\mathbf{x}$ and $\mathbf{x}_i^{(t)}$. The function $\phi(\cdot)$ is the chosen radial basis function, and $\lambda_i^{(t)}$ is the interpolation weight for each data point. To determine the weights $\lambda^{(t)} = [\lambda_1^{(t)}, \ldots, \lambda_n^{(t)}]^\top$, we solve the following linear system based on the known observations $y_i^{(t)}$ at the locations $\mathbf{x}_i^{(t)}$:

$$\Phi^{(t)} \lambda^{(t)} = y^{(t)}, \quad t \in \{1, \ldots, T\} \tag{3}$$

where $\Phi_{ij}^{(t)} = \phi\left(\|\mathbf{x}_i^{(t)} - \mathbf{x}_j^{(t)}\| \right)$ is the kernel matrix constructed from pairwise distances between known data points, and $y^{(t)} = [y_1^{(t)}, \ldots, y_n^{(t)}]^\top$ is the vector of known pollutant values.

In this paper, we adopt the **Multiquadric** function as the radial basis function, defined as:

$$\phi(r) = \sqrt{r^2 + c^2} \tag{4}$$

where $r = \|\mathbf{x} - \mathbf{x}_i^{(t)}\|$ and $c > 0$ is a smoothing parameter that controls the flatness of the basis function. The Multiquadric kernel is particularly effective in capturing smooth spatial patterns in complex environmental fields.

4.3 Meteorological Encoder

Meteorological data M is processed by extracting wind features for ADR modeling and applying channel-wise transformation to allow interactions among heterogeneous meteorological variables, yielding a compact yet informative feature representation. Below, we detail these two processing methods.

First, we extract the wind-related channels from M, denoted as $\vec{W} \in \mathbb{R}^{T \times 2 \times H \times W}$, where the two channels represent the eastwest and northsouth components of horizontal wind. These wind features are essential for capturing directional wind patterns that influence pollutant dispersion.

In parallel, we apply a 1×1 convolution over the full meteorological tensor M, acting on the channel dimension, to reduce the number of channels while enhancing the representational capacity:

$$F = Conv_{1\times 1}(M), \quad F \in \mathbb{R}^{T \times C_F \times H \times W} \tag{5}$$

where C_F denotes the number of output channels in the transformed meteorological representation. Since future weather conditions have a significant impact on air quality, meteorological forecasting is integrated into the model. The *Meteorological Encoder* includes two specialized components: the *Wind Predictor*, which forecasts wind vectors $\vec{W}$, and the *Meteorology Predictor*, which predicts general meteorological features F. Each predictor can adopt any **Spatiotemporal Sequence Predictor**, such as ConvLSTM [44] or TAU [45]. The inputs to these predictors are four-dimensional tensors of shape (T, C, H, W) as shown in Figure 2 (d), where C is the number of channels specific to each predictor depending on the input variable types. Each predictor generates a one-step-ahead forecast, producing an output with a time dimension of 1. The predicted outputs from both components are subsequently incorporated into the *Eulerian ADR Decoder* to enhance predictive performance and ensure better alignment between future meteorological conditions and pollutant dispersion patterns.

4.4 Eulerian ADR Decoder

Eulerian ADR Decoder integrates crucial domain knowledge from atmospheric sciences, particularly Chemical Transport Models (CTM)[46], to simulate pollutant dynamics grounded in physical and chemical principles. Central to CTM is the AdvectionDiffusionReaction (ADR) equation[47], which governs the spatiotemporal evolution of pollutants in the atmosphere.

The main idea of ADR equation is to describe how pollutant concentrations change over time and space due to four fundamental mechanisms: advection, diffusion, chemical reaction, and external source term such as sources and sinks. The continuous form of ADR equation can be expressed as:

$$\frac{\partial X}{\partial t} + \underbrace{\vec{W} \cdot \nabla X}_{Advection} = \underbrace{k_\theta \cdot \nabla^2 X}_{Diffusion} + \underbrace{R(X)}_{Reaction} + \underbrace{S}_{Source} \tag{6}$$

where k_θ is the diffusion coefficient. The mathematical analysis of ADR equation can be found in Appendix H,and the boundary conditions in I .

To enable numerical implementation, we discretize the continuous ADR equation 6 using the implicit forward-time central-space (FTCS) finite difference method [48] to approximate the spatial and temporal derivatives.

Spatial discretization: We apply the central difference method [49], which allows us to approximate spatial gradients and diffusion terms in a numerically stable manner:

$$\nabla X \approx \left(\frac{X[i+1,j] - X[i-1,j]}{2\Delta x}, \frac{X[i,j+1] - X[i,j-1]}{2\Delta y} \right) \tag{7}$$

$$\nabla^2 X[i,j] \approx \frac{X[i+1,j] + X[i-1,j] + X[i,j+1] + X[i,j-1] - 4X[i,j]}{4\Delta x \Delta y} \tag{8}$$

Here, $X[i,j]$ refers to the pollutant concentration at the grid point (i,j), and Δx, Δy are the spatial resolutions in the longitude and latitude directions, respectively.

Temporal discretization: We adopt the explicit Euler method [49] for temporal discretization. The explicit Euler update for the pollutant concentration X is given by:

$$\hat{X}_{T+1} = X_T + \Delta t \left(-\vec{W}_T \cdot \nabla X_T + k_\theta \cdot \nabla^2 X_T + R(X_T) + S_T \right) \tag{9}$$

where Δt is the time step size, X_T is the concentration at time step T, and X_{T+1} is the updated concentration at the next time step.

For different terms, we adopt explicit and implicit modeling strategies based on real-world considerations. The following introduces each term in equation 9:

Advection term is represented as $A_{T+1} = -\vec{W}_T \cdot \nabla X_T$, modeling the transport of pollutants due to the wind. The advection term accounts for the movement of pollutants along the direction of the wind, simulating the large-scale, directional transport across the spatial domain. Explicitly modeling advection allows the system to capture the influence of wind fields on pollutant movement, making it especially crucial for accurate spatiotemporal predictions in areas where wind plays a significant role in pollutant dispersion. This is particularly important in scenarios involving strong wind fields or long-range pollutant transport, where pollutants can travel vast distances within short periods.

Diffusion term is represented as $D_{T+1} = k_\theta \cdot \nabla^2 X_T$, where the diffusion coefficient k_θ is learnable here to capture dynamic diffusion behavior during training. This term models the random spread of pollutants due to molecular motion or turbulence, accounting for the gradual diffusion of pollutants across areas with varying concentration levels. The diffusion term is essential for capturing local mixing effects and pollutant dispersion in regions with weak wind or when pollutants encounter obstacles that hinder direct transport. By incorporating a learnable diffusion coefficient, the model can adapt to different environmental conditions and capture the dynamic nature of pollutant dispersion.

Reaction term models nonlinear chemical transformations among pollutants, such as oxidation reactions, photochemical reactions, and secondary particulate matter formation. These reactions are highly sensitive to atmospheric conditions, including temperature, solar radiation, and humidity, among others. Instead of attempting to model thousands of complex chemical reaction equations, which are difficult to construct accurately, we opt for a more flexible approach. Meteorological features $F_{1:T}$ are mapped through a sigmoid activation, yielding modulation coefficients constrained to the range $(0,1)$, which can be interpreted as a form of *environment-guided soft attention* applied along the channel dimension. Specifically, the chemical reaction term is computed by element-wise multiplying the pollutant concentrations with the corresponding modulation coefficients, as shown below:

$$R(X_{T+1}) = \sigma(F_T) \odot X_T \tag{10}$$

Source term is represented as S_{T+1},which is not explicitly parameterized in our model, but implicitly captured by the deep neural network. This allows the model to flexibly learn complex spatiotemporal emission patternssuch as anthropogenic and biogenic sources or depositionbased on observational data and meteorological inputs, without relying on fixed or analytical formulations of typically inaccessible or sparsely available factors.

Additionally, an intermediate representation $\tilde{X}_{T+1}$ is generated by applying the *Pollutant Predictor* to the historical sequence $X_{1:T}$. The *Pollutant Predictor* shares a similar architecture with the *Wind Predictor* and the *Meteorology Predictor*.

$$\tilde{X}_{T+1} = PollutantPredictor\left(X_{1:T} \parallel F_{1:T}\right), \quad \tilde{X}_{T+1} \in \mathbb{R}^{1\times C_P W\times H\times W} \tag{11}$$

The vertical bar $\|$ denotes concatenation along the channel dimension.

In contrast to the additive formulation in equation 9, we employ a modified approach where the components are concatenated along the channel dimension, allowing the model to flexibly learn complex, nonlinear interactions between features while preserving their individual information before fusion. In addition, we concatenate the predicted future state $\tilde{X}_{T+1}$, which serves as a proxy for the true future state, with the meteorological features F_T, providing the model with a richer context that goes beyond the traditional ADR modeling capabilities. Finally, a 1×1 convolution is applied to generate the final prediction $\hat{X}_{T+1}$:

$$\hat{X}_{T+1} = Conv_{1\times 1}\Big(X_T \parallel \Delta t\left(A_{T+1} \| D_{T+1} \| R(X_{T+1})\right) \parallel \tilde{X}_{T+1} \parallel F_T\Big) \tag{12}$$

4.5 Query and Optimization

Upon obtaining the predicted Eulerian representation $\hat{X}_{T+1} \in \mathbb{R}^{1\times C_P\times H\times W}$, we query the values at the station locations to obtain the final output as:

$$\hat{P}_{T+1} = Query(\hat{X}_{T+1}, \mathcal{S}), \quad \hat{P}_{T+1} \in \mathbb{R}^{1\times C_P\times N} \tag{13}$$

CTENet performs recursive multi-step forecasting of pollutant concentrations over a future horizon. At each iteration, it takes the past T time steps of input to predict the pollutant concentration $\hat{X}$, and uses a sliding window mechanism to advance the input sequence by one step—replacing the oldest input with the latest prediction. By applying this process iteratively, CTENet generates the multi-step prediction sequence $\hat{P}_{T+1:T+\tau} \in \mathbb{R}^{\tau\times C_P\times N}$.

For optimization, we minimize the discrepancy between the predicted and ground-truth pollutant concentrations at the station locations over the entire forecast horizon. Specifically, the model, parameterized by Θ, is trained by minimizing RMSE loss:

$$\mathcal{L}(\Theta) = \sqrt{\frac{1}{\tau N}\sum_{t=T+1}^{T+\tau}\sum_{n=1}^{N}\left\| P_t^{(n)} - \hat{P}_t^{(n)} \right\|_2^2} \tag{14}$$

where $P_t^{(n)}$ and $\hat{P}_t^{(n)}$ denote the ground truth and predicted pollutant concentration vectors at station n and time t.

5 Experiments

We compare our proposed CTENet with the representative and SOTA models to evaluate their effectiveness on two real-world pollutant datasets. Following the common practice in previous related studies, this study focuses on PM$_{2.5}$ concentration as the target variable, with meteorological data serving as auxiliary variables. Unless otherwise specified, TAU [45] is used as the default *Spatiotemporal Sequence Predictor* type in all experiments. Further experiments details are provided in the Appendix C and D.

Dataset: The pollutant monitoring station datasets are collected from China[18] and the United States[50], covering the entire year of 2018. Simultaneously, we obtain meteorological data from the National Centers for Environmental Prediction (NCEP) [51] for the same time and locations.

Baselines: We choose the baseline methods from three categories: (1) Statistical models, including Historical Average (HA) and Vector Auto-Regression (VAR)[52]. (2) Graph-based models, including STGCN [53], DCRNN [54], GTS [55], AirFormer [18], AirPhyNet [13], and PM$_{2.5}$-GNN [16].

(3) Spatiotemporal sequence prediction models, including TAU. M To ensure fair comparisobn with site-based models, we assign the pixel value at each grid cell to all stations located within that cell. More details can be found in Appendix C.

Implement Details: Each dataset is divided into training, validation, and test sets in a $7 : 1.5 : 1.5$ ratio. We use data from the past 72 hours to predict pollutant concentrations for the next 24, 48, and 72 hours. Our implementation code is released for public use[2].

5.1 Main Results

Table 1 presents a performance comparison between CTENet and several baseline methods across two datasets. We can draw the following observations: (1) CTENet outperforms the other baseline models in terms of overall performance, regardless of whether ConvLSTM or TAU is used as the prediction module. Specifically, compared to the best-performing baseline, our best CTENet variant achieves an average MAE improvement of 33.1% and an RMSE improvement of 45.8% in the USA, while in China, it achieves an average MAE improvement of 15.8% and an RMSE improvement of 21.0%. This substantial improvement underscores the effectiveness and superiority of the CTENet framework. Notably, when using TAU as the predictor, CTENet achieves optimal results in most cases; (2) Graph-based deep learning-based methods clearly outperform traditional statistical methods in prediction accuracy, demonstrating the powerful capability of deep learning in learning complex spatiotemporal relationships and modeling nonlinear patterns; (3) Spatio-temporal deep learning models originally developed for traffic forecasting, such as GTS and STGCN, exhibit suboptimal performance in our experiments. This may be attributed to their lack of specialized architectural adaptations for processing high-dimensional meteorological inputs. (4) Spatiotemporal sequence prediction models, such as TAU, show great potential in fine-grained air quality prediction tasks, performing on par with SOTA graph-based models.

Table 1: $PM_{2.5}$ prediction performance: the best-performing result is highlighted in bold, while the second-best is underlined for easy comparison.

Methods	USA Data						China Data					
	24h		48h		72h		24h		48h		72h	
	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
HA	5.30	11.57	5.66	12.54	5.99	13.23	21.64	38.03	22.76	39.12	23.58	40.03
VAR	6.32	14.41	5.78	12.74	5.76	12.94	24.74	39.85	25.43	41.85	26.66	44.14
STGCN	4.29	9.03	4.51	9.03	4.63	9.08	31.43	43.72	31.91	44.06	32.69	44.75
DCRNN	5.40	14.50	5.42	12.81	5.38	13.48	28.14	49.81	27.45	47.36	27.39	47.63
GTS	5.57	14.65	5.60	14.32	5.61	14.18	23.46	41.70	23.50	42.53	23.85	44.41
AirFormer	4.05	10.44	4.40	10.74	4.60	10.89	19.09	36.08	20.89	38.42	21.85	39.61
AirPhyNet	4.47	11.36	4.79	11.40	4.94	11.48	18.75	36.35	19.97	37.16	20.74	37.64
$PM_{2.5}$-GNN	4.38	9.77	4.63	9.66	4.76	9.63	17.71	33.25	19.12	34.16	19.73	34.53
TAU	4.71	12.51	4.94	13.56	5.22	13.90	15.85	26.80	15.43	27.35	15.60	26.85
CTENet w/ ConvLSTM	4.12	8.46	4.31	8.66	4.43	8.84	13.79	23.14	14.44	23.79	**15.28**	**24.47**
CTENet w/ TAU	**2.66**	**4.86**	**2.99**	**4.86**	**3.10**	**5.00**	**10.90**	**16.99**	**13.28**	**22.60**	15.92	26.74
% Best Improvement	34.43	46.02	32.06	46.18	32.68	44.86	31.24	36.60	13.97	17.36	2.04	8.86

Overall, these experimental results validate the exceptional performance of CTENet in air quality prediction tasks. By combining a CTM-based pollutant model with an Eulerian spatiotemporal modeling framework, CTENet effectively leverages spatial continuity and temporal dynamics, enabling more accurate and physically consistent air quality predictions.

5.2 Ablation Study

To evaluate the individual contributions of each component in the model for 24-hour predictions, we conduct a systematic ablation study by progressively removing components. Figure 5 presents the ablation results for the ADR terms, while ablation experiments related to the interpolation method are provided in Appendix F.

Effect of Advection and Difussion term: Full CTENet outperforms models with the removal of the diffusion or advection modules on both the China and USA datasets, highlighting the benefits of incorporating physical knowledge. The removal of the diffusion term leads to a relatively minor degradation in performance, particularly in the China dataset, possibly because the multilayer CNNs in the predictor are able to partially compensate for its absence by implicitly modeling local spatial dependencies. The theoretical proof is provided in Appendix B. In contrast, removing the Advection

[2] `https://github.com/santafirefox0/CTENet`

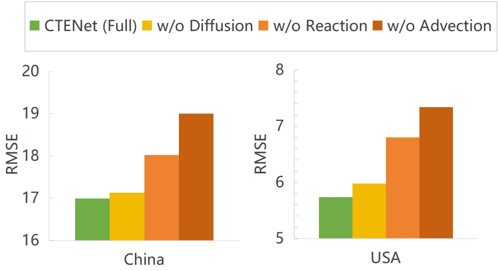


Figure 5: Ablation study of ADR terms.

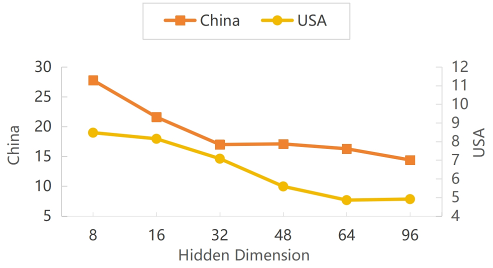


Figure 6: Hidden dimension analysis on RMSE.

module causes a more significant performance decline, emphasizing the difficulty standard deep networks face in learning pollutant transport due to wind, and the necessity of integrating domain-specific physical mechanisms into the model architecture.

Effect of Reaction term: Removing the reaction term results in a noticeable performance decline, even though historical meteorological features remain part of the input. This indicates that the benefit of reaction term lies not in simply incorporating additional features, but in its role as a dynamic weighting mechanism. By adaptively modulating the influence of meteorological inputs, the reaction term enhances the model's ability to represent complex pollutant dynamics, highlighting the value of structured chemical integration in CTENet.

5.3 Hidden Dimension Analysis

In this experiment, we evaluate the RMSE performance of CTENet under different hidden dimension settings for two datasets, as illustrated in Figure 6. The results show that increasing the hidden dimension generally leads to lower RMSE on both datasets, indicating improved model performance. However, the improvement is not indefinite: for the China dataset, the RMSE plateaus beyond a hidden dimension of 32, while the USA dataset achieves its lowest RMSE at 64. This suggests that increasing the hidden dimension yields diminishing returns, as model complexity eventually exceeds the intrinsic structure of the data, potentially leading to overfitting. Moreover, the optimal hidden dimension varies across datasets, likely due to differences in data complexity or noise characteristics.

5.4 Case Study

To demonstrate the interpretability and physical consistency of our model, we present a case study based on predicted $PM_{2.5}$ concentrations over central China. Figure 7 displays the spatial distribution of $PM_{2.5}$ at four consecutive time steps. In the highlighted red box, northwest winds are observed at the first three time steps. As a result, the pollutant mass shifts toward the southeast in the subsequent frame, indicating that the transport of $PM_{2.5}$ is consistent with the wind direction. This spatial displacement of pollution illustrates how CTENet captures the dynamic advection of pollutants, consistent with atmospheric physical principles, and reflects the model's capacity to learn meaningful spatiotemporal interactions.

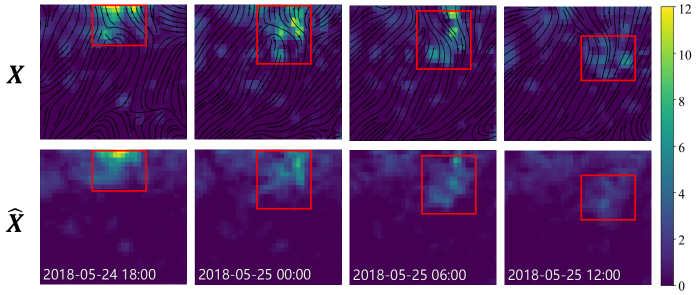


Figure 7: Visualization of $PM_{2.5}$ concentrations as heatmaps. The first row shows reference distributions X overlaid with wind streamlines, and the second row shows model predictions $\hat{X}$.

6 Conclusion and Future Work

In this work, we propose the CTENet network for air quality prediction, which integrates physics-guided models with deep learning techniques to effectively model the spatiotemporal evolution of pollutants. Comprehensive experiments on several benchmark datasets have validated the effectiveness of the proposed method, achieving a remarkable RMSE improvement of 45.8% on the USA dataset and 21.0% on the China dataset. Our current formulation lacks uncertainty quantification, limiting its applicability in risk-sensitive scenarios such as probabilistic forecasting or decision-making under uncertainty. Future work will explore integrating uncertainty-aware mechanisms to improve reliability and interpretability.

Acknowledgments and Disclosure of Funding

This research is supported by National Natural Science Foundation of China (No. 62306033, 42371480).

References

[1] Mohammad Naim Azimi and Mohammad Mafizur Rahman. Unveiling the health consequences of air pollution in the worlds most polluted nations. *Scientific Reports*, 14(1):9856, 2024.

[2] Ioannis Manisalidis, Elisavet Stavropoulou, Agathangelos Stavropoulos, and Eugenia Bezirtzoglou. Environmental and health impacts of air pollution: a review. *Frontiers in public health*, 8:14, 2020.

[3] Sanjay Rajagopalan and Philip J Landrigan. Pollution and the heart. *New England Journal of Medicine*, 385(20):1881–1892, 2021.

[4] Sotiris Vardoulakis, Bernard EA Fisher, Koulis Pericleous, and Norbert Gonzalez-Flesca. Modelling air quality in street canyons: a review. *Atmospheric environment*, 37(2):155–182, 2003.

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

[6] Aaron Daly and Paolo Zannetti. Air pollution modeling–an overview. *Ambient air pollution*, pages 15–28, 2007.

[7] Jindong Han, Weijia Zhang, Hao Liu, and Hui Xiong. Machine learning for urban air quality analytics: A survey. *arXiv preprint arXiv:2310.09620*, 2023.

[8] V Athira, P Geetha, Rab Vinayakumar, and KP Soman. Deepairnet: Applying recurrent networks for air quality prediction. *Procedia computer science*, 132:1394–1403, 2018.

[9] Dewen Seng, Qiyan Zhang, Xuefeng Zhang, Guangsen Chen, and Xiyuan Chen. Spatiotemporal prediction of air quality based on lstm neural network. *Alexandria Engineering Journal*, 60(2), 2021.

[10] Shuliang Wang, Ziyu Wang, Sijie Ruan, Haoyu Han, Keqin Xiong, Hanning Yuan, Ziqiang Yuan, Guoqing Li, Jie Bao, and Yu Zheng. Delvmap: Completing residential roads in maps based on couriers trajectories and satellite imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–14, 2024.

[11] Shuliang Wang, Xinyu Pan, Sijie Ruan, Haoyu Han, Ziyu Wang, Hanning Yuan, Jiabao Zhu, and Qi Li. Diffcrime: A multimodal conditional diffusion model for crime risk map inference. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3212–3221, 2024.

[12] Lianfa Li, Roxana Khalili, Frederick Lurmann, Nathan Pavlovic, Jun Wu, Yan Xu, Yisi Liu, Karl O'Sharkey, Beate Ritz, Luke Oman, et al. Physics-informed deep learning to reduce the bias in joint prediction of nitrogen oxides. *arXiv preprint arXiv:2308.07441*, 2023.

[13] Kethmi Hirushini Hettige, Jiahao Ji, Shili Xiang, Cheng Long, Gao Cong, and Jingyuan Wang. Airphynet: Harnessing physics-guided neural networks for air quality prediction. *arXiv preprint arXiv:2402.03784*, 2024.

[14] Haomin Yu, Jilin Hu, Xinyuan Zhou, Chenjuan Guo, Bin Yang, and Qingyong Li. Cgf: A category guidance based pm _{2.5} sequence forecasting training framework. *IEEE Transactions on Knowledge and Data Engineering*, 35(10):10125–10139, 2023.

[15] Jindong Han, Hao Liu, Hengshu Zhu, Hui Xiong, and Dejing Dou. Joint air quality and weather prediction based on multi-adversarial spatiotemporal networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 4081–4089, 2021.

[16] Shuo Wang, Yanran Li, Jiang Zhang, Qingye Meng, Lingwei Meng, and Fei 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*, pages 163–166, 2020.

[17] Chunyang Wang, Yanmin Zhu, Tianzi Zang, Haobing Liu, and Jiadi Yu. Modeling inter-station relationships with attentive temporal graph convolutional network for air quality prediction. In *Proceedings of the 14th ACM international conference on web search and data mining*, pages 616–634, 2021.

[18] Yuxuan Liang, Yutong Xia, Songyu Ke, Yiwei Wang, Qingsong Wen, Junbo Zhang, Yu Zheng, and Roger Zimmermann. Airformer: Predicting nationwide air quality in china with transformers. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 14329–14337, 2023.

[19] Jindong Tian, Yuxuan Liang, Ronghui Xu, Peng Chen, Chenjuan Guo, Aoying Zhou, Lujia Pan, Zhongwen Rao, and Bin Yang. Air quality prediction with physics-guided dual neural odes in open systems. In *The Thirteenth International Conference on Learning Representations*, 2025.

[20] Arvydas Juozaitis, Saulius Trakumas, Rasa Girgždien, Aloyzas Girgždys, Dalia Šopauskien, and Vidmantas Ulevičius. Investigations of gas-to-particle conversion in the atmosphere. *Atmospheric research*, 41(3-4):183–201, 1996.

[21] Ana Suárez Sánchez, Paulino José García Nieto, Francisco Javier Iglesias-Rodríguez, and José Antonio Vilán Vilán. Nonlinear air quality modeling using support vector machines in gijón urban area (northern spain) at local scale. *International Journal of Nonlinear Sciences and Numerical Simulation*, 14(5):291–305, 2013.

[22] Xiuwen Yi, Junbo Zhang, Zhaoyuan Wang, Tianrui Li, and Yu Zheng. Deep distributed fusion network for air quality prediction. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 965–973, 2018.

[23] Yutong Feng, Qiongyan Wang, Yutong Xia, Junlin Huang, Siru Zhong, and Yuxuan Liang. Spatio-temporal field neural networks for air quality inference. *arXiv preprint arXiv:2403.02354*, 2024.

[24] Qiongyan Wang, Yutong Xia, Siru ZHong, Weichuang Li, Yuankai Wu, Shifen Cheng, Junbo Zhang, Yu Zheng, and Yuxuan Liang. Airradar: Inferring nationwide air quality in china with deep neural networks. *arXiv preprint arXiv:2501.13141*, 2025.

[25] Jiahao Ji, Jingyuan Wang, Zhe Jiang, Jiawei Jiang, and Hu Zhang. Stden: Towards physics-guided neural networks for traffic flow prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 4048–4056, 2022.

[26] Jes Fenger. Urban air quality. *Atmospheric environment*, 33(29):4877–4900, 1999.

[27] Andy P Jones. Indoor air quality and health. *Atmospheric environment*, 33(28):4535–4564, 1999.

[28] Hajime Akimoto. Global air quality and pollution. *Science*, 302(5651):1716–1719, 2003.

[29] Gaganjot Kaur Kang, Jerry Zeyu Gao, Sen Chiao, Shengqiang Lu, and Gang Xie. Air quality prediction: Big data and machine learning approaches. *Int. J. Environ. Sci. Dev*, 9(1):8–16, 2018.

[30] Jingyang Wang, Jiazheng Li, Xiaoxiao Wang, Jue Wang, and Min Huang. Air quality prediction using ct-lstm. *Neural Computing and Applications*, 33:4779–4792, 2021.

[31] Kunwar P Singh, Shikha Gupta, Atulesh Kumar, and Sheo Prasad Shukla. Linear and nonlinear modeling approaches for urban air quality prediction. *Science of the Total Environment*, 426:244–255, 2012.

[32] Zhongshan Yang and Jian Wang. A new air quality monitoring and early warning system: Air quality assessment and air pollutant concentration prediction. *Environmental research*, 158:105–117, 2017.

[33] Junshan Wang and Guojie Song. A deep spatial-temporal ensemble model for air quality prediction. *Neurocomputing*, 314:198–206, 2018.

[34] Yuan Huang, Yuxing Xiang, Ruixiao Zhao, and Zhe Cheng. Air quality prediction using improved pso-bp neural network. *Ieee Access*, 8:99346–99353, 2020.

[35] Jun Ma, Jack CP Cheng, Changqing Lin, Yi Tan, and Jingcheng Zhang. Improving air quality prediction accuracy at larger temporal resolutions using deep learning and transfer learning techniques. *Atmospheric Environment*, 214:116885, 2019.

[36] Xiang Li, Ling Peng, Yuan Hu, Jing Shao, and Tianhe Chi. Deep learning architecture for air quality predictions. *Environmental Science and Pollution Research*, 23:22408–22417, 2016.

[37] Zaharaddeen Karami Lawal, Hayati Yassin, Daphne Teck Ching Lai, and Azam Che Idris. Physics-informed neural network (pinn) evolution and beyond: A systematic literature review and bibliometric analysis. *Big Data and Cognitive Computing*, 6(4):140, 2022.

[38] Rui Wang and Rose Yu. Physics-guided deep learning for dynamical systems: A survey. *arXiv preprint arXiv:2107.01272*, 2021.

[39] Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707, 2019.

[40] Arka Daw, Anuj Karpatne, William D Watkins, Jordan S Read, and Vipin Kumar. Physics-guided neural networks (pgnn): An application in lake temperature modeling. In *Knowledge guided machine learning*, pages 353–372. Chapman and Hall/CRC, 2022.

[41] Yohai Bar-Sinai, Stephan Hoyer, Jason Hickey, and Michael P Brenner. Learning data-driven discretizations for partial differential equations. *Proceedings of the National Academy of Sciences*, 116(31):15344–15349, 2019.

[42] Zichao Long, Yiping Lu, and Bin Dong. Pde-net 2.0: Learning pdes from data with a numeric-symbolic hybrid deep network. *Journal of Computational Physics*, 399:108925, 2019.

[43] Ahmad Mohammadshirazi, Aida Nadafian, Amin Karimi Monsefi, Mohammad H Rafiei, and Rajiv Ramnath. Novel physics-based machine-learning models for indoor air quality approximations. *arXiv preprint arXiv:2308.01438*, 2023.

[44] Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. *Advances in neural information processing systems*, 28, 2015.

[45] Cheng Tan, Zhangyang Gao, Lirong Wu, Yongjie Xu, Jun Xia, Siyuan Li, and Stan Z Li. Temporal attention unit: Towards efficient spatiotemporal predictive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18770–18782, 2023.

[46] Melissa A Venecek, Xin Yu, and Michael J Kleeman. Predicted ultrafine particulate matter source contribution across the continental united states during summertime air pollution events. *Atmospheric Chemistry and Physics*, 19(14):9399–9412, 2019.

[47] Kaveh Zamani and Fabián A Bombardelli. Analytical solutions of nonlinear and variable-parameter transport equations for verification of numerical solvers. *Environmental Fluid Mechanics*, 14:711–742, 2014.

[48] Surattana Sungnul, Kanokwarun Para, Elvin J Moore, Sekson Sirisubtawee, and Sutthisak Phongthanapanich. A finite difference method for solution of integer-order and caputo fractional-time advection-diffusion-reaction equations: Convergence analysis and application to air pollution. *Engineering Letters*, 33(2), 2025.

[49] John Charles Butcher. *Numerical methods for ordinary differential equations*. John Wiley & Sons, 2016.

[50] United States Environmental Protection Agency. Air quality system (aqs) data download. `https://aqs.epa.gov/aqsweb/airdata/download_files.html`, 2025. Accessed: 2025-04-16.

[51] NOAA Environmental Modeling Center. Global forecast system (gfs) 1.0 degree. `https://www.ncei.noaa.gov/products/weather-climate-models/global-forecast`, 2004. [indicated subset used].

[52] Christopher A Sims. Macroeconomics and reality. *Econometrica: journal of the Econometric Society*, pages 1–48, 1980.

[53] Bing Yu, Haoteng Yin, and Zhanxing Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint arXiv:1709.04875*, 2017.

[54] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926*, 2017.

[55] Chao Shang, Jie Chen, and Jinbo Bi. Discrete graph structure learning for forecasting multiple time series. *arXiv preprint arXiv:2101.06861*, 2021.

[56] Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via deeponet based on the universal approximation theorem of operators. *Nature machine intelligence*, 3(3):218–229, 2021.

[57] ZiDong Wang, Zeyu Lu, Di Huang, Tong He, Xihui Liu, Wanli Ouyang, and Lei Bai. Predbench: Benchmarking spatio-temporal prediction across diverse disciplines. In *European Conference on Computer Vision*, pages 286–304. Springer, 2024.

[58] Stephan Rasp, Stephan Hoyer, Alexander Merose, Ian Langmore, Peter Battaglia, Tyler Russell, Alvaro Sanchez-Gonzalez, Vivian Yang, Rob Carver, Shreya Agrawal, et al. Weatherbench 2: A benchmark for the next generation of data-driven global weather models. *Journal of Advances in Modeling Earth Systems*, 16(6):e2023MS004019, 2024.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: See Abstract and Section 1 Introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See Section 6.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: See Appendix B , etc.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The relevant source code has been released and the link can be found at the end of the abstract in the paper. This provides a clear path for reproducibility and verification of the experimental results, ensuring that others can replicate the findings from the paper.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The relevant source code has been released and the link can be found at the end of the abstract in the paper.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so No is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The training and test details can be found in Section 5, Appendix C Baselines Details, the Appendix D Experimental Setup, and the code.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: See Appendix J Statistical Significance Analysis.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: See Appendix D Experimental Setup and Appendix G.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research presented in this paper adheres to the NeurIPS Code of Ethics. Ethical considerations were fully reviewed and followed throughout the research process, ensuring respect for privacy, fairness, and transparency. Additionally, the study avoids harmful biases and complies with applicable laws and regulations.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: See Section 6.

    Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: This paper does not involve the release of models or datasets with high misuse risk.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All data used in this study are obtained from publicly available datasets. Their sources are cited in Appendix D, where the corresponding usage conditions. We have ensured that all licenses have been properly respected.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.

- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

A Notation

Table 2: Summary of notations used in the paper

Symbol	Description
N	Number of observation stations
T, τ	Input time step and prediction time step, respectively
H, W	Total number of grid points in the latitude and longitude dimensions
P_t	Historical pollutant concentration data at time t
X_t	Spatially interpolated historical pollutant concentration data at time t
M_t	Continuous historical meteorological data at time t
C_P	Number of pollutant types and the number of pollutant channels
C_M	Number of meteorological channels
C_F	Number of meteorological feature channels.
$\vec{W}_t$	Continuous wind vector at time t
F_t	Continuous meteorological features at time t
S_t	Source Term at time t
$R(\cdot)$	Function modeling the chemical transformation of pollutants
k_θ	Learnable diffusion coefficient
$\odot$	Element-wise product
$\sigma(\cdot)$	Sigmoid function: $\sigma(x) = \frac{1}{1+e^{-x}}$, used to squash values between 0 and 1.
$\|\|$	Channel-wise concatenation

B Proof of Convolution Kernel Fitting Diffusion

The diffusion term $k_\theta \cdot \nabla^2 X$ is formulated as the product of the diffusion coefficient k_θ and the Laplacian operator of the pollutant concentration X, whose discrete form is defined in Equation (8). In this section, we demonstrate how a 3×3 convolution operation in a convolutional neural network can approximate the discretized Laplacian operator. The convolution operation can be viewed as a weighted summation of spatial data using a sliding window, which can be expressed as:

$$Y[i,j] = \sum_{m=-1}^{1} \sum_{n=-1}^{1} K[m,n] \cdot X[i+m, j+n] \tag{15}$$

where K is the convolution kernel, $X[i,j]$ is the input data, and $Y[i,j]$ is the result of the convolution operation. The model ultimately learns an appropriate convolution kernel K, which, when close to the following form, causes the convolution operation to approximate the Laplacian operator:

$$K = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \tag{16}$$

This completes the proof. A related empirical demonstration can be found in [56].

C Baselines Details

We compare our CTENet with SOTA models for long-term graph sequence prediction and spatiotemporal sequence prediction models, some of which are specifically optimized for air quality prediction.

- **HA** uses the average of the historical observations as the predictions. This simple baseline can provide insights into the effectiveness of more complex models.

- **VAR** [52] is a classical statistical model that was originally proposed for macroeconomic analysis. It utilizes past values of the multivariate time series to make predictions. It captures temporal dependencies without considering the spatial aspect.
- **STGCN** [53] integrates both spatial and temporal dependencies, leveraging graph-based convolutions to predict air quality over time.
- **DCRNN** [54] improves on STGCN by incorporating recurrent layers and diffusion-based graph convolutions to capture dynamic relationships in both spatial and temporal domains.
- **GTS** [55] proposes a method for simultaneously learning the graph structure and forecasting multivariate time series using GNNs, where the graph is unknown, by optimizing a probabilistic graph model.
- **Airformer** [18] is a novel Transformer-based model designed to predict nationwide air quality in China with fine spatial granularity, incorporating a two-stage learning process to efficiently capture spatio-temporal representations and intrinsic uncertainty.
- **AirPhyNet** [13] is a novel approach that integrates physical principles of air particle movement (diffusion and advection) into a neural network architecture on graph-based data structures, improving air quality prediction by capturing spatio-temporal relationships and providing interpretable, physically meaningful predictions.
- **PM$_{2.5}$-GNN** [16] is a novel graph-based model designed to capture both fine-grained and long-term dependencies in PM$_{2.5}$ forecasting, incorporating critical domain knowledge and validated on real-world data, with an online deployment for free forecasting service.
- **TAU** [45] decomposes temporal attention into intra-frame statical attention and inter-frame dynamical attention to efficiently capture both spatial dependencies within a single frame and temporal dependencies across frames, enabling parallelization of the temporal module in spatiotemporal predictive learning.After referring to [57, 58], we have also decided to adopt TAU as one of the predictors in our model.

The key hyperparameters in other baselines are also tuned for the best performance.

D Experimental Setup

Datasets Details: We conduct extensive experiments on two real-world pollutant monitoring station datasets collected from China and the United States, covering the entire year of 2018, to evaluate the performance of our proposed CTENet. The Chinese dataset[18] includes 480 primary monitoring stations. The United States dataset[50] consists of 365 monitoring stations across the contiguous United States. Both datasets provide hourly air quality. Simultaneously, we obtain meteorological data from the National Centers for Environmental Prediction (NCEP) [51] for the same time and locations, covering 354 channels with information on temperature, humidity, wind speed, potential height, vertical velocity, sunlight, cloud cover, and other variables at different isobaric levels, with a spatial resolution of 0.5ř in both latitude and longitude.

Implementation Details: To align the meteorological data temporally, we treat each 6-hour period as one time step, averaging the pollutant concentrations within that period. During model training, we employ techniques such as regularization, gradient clipping, and progressive learning to improve training efficiency.

E Loss and Evaluation Metrics

Let $\mathbf{x} = (x_1, \dots, x_m)$ represent the ground truth, and $\hat{\mathbf{x}} = (\hat{x}_1, \dots, \hat{x}_m)$ represent the predicted pollutant concentrations.

The loss function used in this paper are defined as follows:

Mean Squared Error (MSE) Loss

$$MSELoss(\mathbf{x}, \hat{\mathbf{x}}) = \frac{1}{m} \sum_{i=1}^{m} (x_i - \hat{x}_i)^2 \tag{17}$$

The evaluation metrics used in this paper are defined as follows:

Mean Absolute Error (MAE)

$$\mathrm{MAE}(\mathbf{x}, \hat{\mathbf{x}}) = \frac{1}{m} \sum_{i=1}^{m} |x_i - \hat{x}_i| \tag{18}$$

Root Mean Square Error (RMSE)

$$\mathrm{RMSE}(\mathbf{x}, \hat{\mathbf{x}}) = \sqrt{\frac{1}{m} \sum_{i=1}^{m} (x_i - \hat{x}_i)^2} \tag{19}$$

F Ablation Study of Interpolation

As shown in 8, To investigate the effectiveness of the interpolation method used in our model, we conduct an ablation study comparing three variants: (1) our default RBF interpolation, (2) nearest-neighbor interpolation, and (3) a variant without any interpolation. As shown in Figure 8, the RBF interpolation achieves the best performance among the three settings, demonstrating its ability to provide smoother and more physically consistent spatial inputs, which benefits downstream prediction. Nearest-neighbor interpolation, while simple, introduces artifacts due to abrupt transitions between points. Removing interpolation altogether results in degraded performance, primarily because the model fails to effectively capture spatial continuity and interactions when operating on sparsely and irregularly distributed observational data.

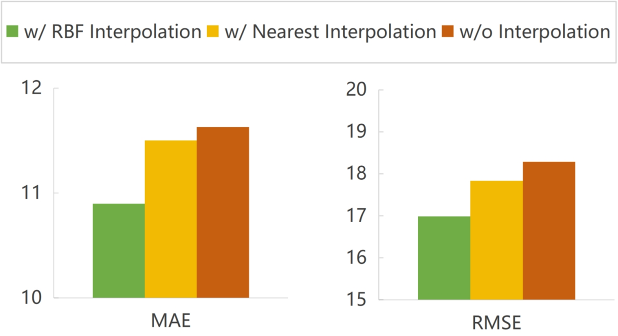


Figure 8: Ablation Study of Interpolation.

G Computational Complexity Analysis

The main computational cost of the proposed algorithm arises from the Eulerian ADR Decoder, including the computation of the three ADR terms and the three Predictor modules. The ADR terms consist of advection, diffusion, and reaction, each with a computational complexity of $\mathcal{O}(B \cdot H \cdot W)$. Each of the three Predictors (e.g., TAU3) contributes a computational complexity of $\mathcal{O}(B \cdot T \cdot D^2 \cdot H \cdot W)$, where B is the batch size, T is the length of the input sequence, and D is the hidden dimension. Summing the complexities of all components, the total computational complexity is:

$$\mathcal{O}(B \cdot H \cdot W) + \mathcal{O}(B \cdot T \cdot D^2 \cdot H \cdot W) = \mathcal{O}(B \cdot T \cdot D^2 \cdot H \cdot W) \tag{20}$$

For reference, the computational complexity of graph-based methods is approximately $\mathcal{O}(B \cdot T \cdot D^2 \cdot N)$, where N is the number of spatial nodes in the graph,which is typically smaller than $H \cdot W$, as shown in Table 3. On a single RTX 4090 GPU, the complete training process can be completed within 3 to 6 hours, depending on the data volume and forecast horizon. More importantly, during inference, generating a 72-hour forecast takes less than 0.15 seconds, which is fast enough to meet the requirements of most real-time applications.

Table 3: Comparison of Theoretical Computational Complexity between CTENet and graph-based Methods.

Methods	**CTENet**	**Graph-based Methods**
Theoretical Complexity	$\mathcal{O}(B \cdot T \cdot D^2 \cdot H \cdot W)$	$\mathcal{O}(B \cdot T \cdot D^2 \cdot N)$
China Dataset	$\mathcal{O}(B \cdot T \cdot D^2 \cdot 80 \cdot 130)$	$\mathcal{O}(B \cdot T \cdot D^2 \cdot 480)$
USA Dataset	$\mathcal{O}(B \cdot T \cdot D^2 \cdot 70 \cdot 140)$	$\mathcal{O}(B \cdot T \cdot D^2 \cdot 365)$

H Mathematical Analysis of the ADR Equation

The ADR equation 6 satisfies the existence, uniqueness, and stability conditions, making it a well-posed problem for modeling physical processes. A formal proof of these properties is provided below.

Existence and Uniqueness: The ADR equation is a second-order linear parabolic PDE in both space and time. The well-known theory of existence for parabolic equations asserts that, for smooth (analytic) coefficients and appropriate initial and boundary conditions, there always exists at least one solution in a finite domain.

To prove the uniqueness of the solution for the ADR equation, we follow the standard method of analysis for parabolic partial differential equations. By using the energy method, we assume two solutions $X_1(t)$ and $X_2(t)$ satisfy the ADR equation. Defining the difference $w(t) = X_1(t) - X_2(t)$, we derive the following equation for w:

$$\frac{dw}{dt} + \vec{v} \cdot \nabla w = k_\theta \cdot \nabla^2 w + R'(X_1) \cdot w \tag{21}$$

Applying the energy method by multiplying both sides by w and integrating over the spatial domain, we obtain:

$$\frac{d}{dt} \int_\Omega w^2 \, dx = -k_\theta \int_\Omega |\nabla w|^2 \, dx \tag{22}$$

This shows that $\int_\Omega w^2 \, dx$ is decreasing over time, implying that $w(t)$ tends to zero, which leads to the conclusion $X_1(t) = X_2(t)$. Hence, the ADR equation has a unique solution.

Stability: For stability, we consider small perturbations in the initial condition. Let X_0 and X_1 be two initial conditions such that $|X_0 - X_1| < \epsilon$. The difference between the solutions $X_1(t)$ and $X_2(t)$ is given by:

$$w(t) = X_1(t) - X_2(t) \tag{23}$$

Using the energy method again, we find that the difference in the solutions remains bounded and decreases over time:

$$\int_\Omega w^2 \, dx \leq \int_\Omega (X_0 - X_1)^2 \, dx \tag{24}$$

Thus, small changes in the initial condition lead to small changes in the solution, demonstrating the stability of the ADR equation. This confirms that the ADR equation is a well-posed problem under the given initial and boundary conditions.

I Boundary conditions

In the absence of external pollutant data beyond the study area, we apply Neumann boundary conditions [48] to maintain the stability and convergence of the numerical method. To minimize boundary effects, the computational domain is set at least 300 km away from the nearest monitoring station.

Neumann conditions enforce a zero-gradient (no-flux) condition at the boundaries of the domain:

$$\frac{\partial X}{\partial x} = 0, \quad \frac{\partial X}{\partial y} = 0 \quad \text{on the boundary} \tag{25}$$

J Statistical Significance Analysis

In this appendix, we derive an approximate probability that method A performs better than method B, based solely on their respective summary statistics: mean absolute error (MAE) and root mean square error (RMSE). Let these values be denoted as $\text{MAE}_A, \text{RMSE}_A$ for method A and $\text{MAE}_B, \text{RMSE}_B$ for method B.

Table 1 can be interpreted as paired tests between our method and each baseline model.

Let y_i denote the ground truth value and $\hat{y}_i^{(A)}, \hat{y}_i^{(B)}$ denote the predictions of methods A and B on sample i. We define the absolute prediction error as:

$$d_i^{(A)} = \left| \hat{y}_i^{(A)} - y_i \right|, \quad d_i^{(B)} = \left| \hat{y}_i^{(B)} - y_i \right|. \tag{26}$$

We aim to estimate the probability that $d_i^{(A)} < d_i^{(B)}$, i.e., method A outperforms B on a randomly drawn sample. We assume the errors $d_i^{(A)}$ and $d_i^{(B)}$ are independent and approximately follow Gaussian distributions:

$$d_i^{(A)} \sim \mathcal{N}(\mu_A, \sigma_A^2), \quad d_i^{(B)} \sim \mathcal{N}(\mu_B, \sigma_B^2), \tag{27}$$

with parameters estimated from available statistics:

$$\mu_A \approx \text{MAE}_A, \quad \mu_B \approx \text{MAE}_B, \tag{28}$$

$$\sigma_A^2 \approx \text{RMSE}_A^2 - \text{MAE}_A^2, \quad \sigma_B^2 \approx \text{RMSE}_B^2 - \text{MAE}_B^2. \tag{29}$$

Define the difference in absolute errors:

$$\Delta d_i = d_i^{(B)} - d_i^{(A)}, \quad \Delta d_i \sim \mathcal{N}(\mu_B - \mu_A, \sigma_A^2 + \sigma_B^2). \tag{30}$$

Standardizing this with a normal variable:

$$Z = \frac{\Delta d_i - (\mu_B - \mu_A)}{\sqrt{\sigma_A^2 + \sigma_B^2}} \sim \mathcal{N}(0, 1), \tag{31}$$

The probability that method A outperforms method B is then given by:

$$\begin{aligned} p = P(d_i^{(A)} < d_i^{(B)}) &= \Phi\left(\frac{\mu_B - \mu_A}{\sqrt{\sigma_A^2 + \sigma_B^2}} \right) \\ &= \Phi\left(\frac{\text{MAE}_B - \text{MAE}_A}{\sqrt{(\text{RMSE}_A^2 - \text{MAE}_A^2) + (\text{RMSE}_B^2 - \text{MAE}_B^2)}} \right) \end{aligned} \tag{32}$$

where $\Phi(\cdot)$ denotes the standard normal cumulative distribution function. As a result, the likelihood that method A performs better than method B on n samples is:

$$P(A \text{ better than } B) \approx 1 - \Phi\left(\frac{n/2 - np}{\sqrt{np(1-p)}} \right). \tag{33}$$

For all evaluations, each experiment includes at least 174 sequences, with each sequence containing observations from no fewer than 365 stations, as detailed in Appendix D. Each station provides a temporal sequence of observations, which is treated as an independent sample for statistical evaluation. Thus, the total number of samples is $n = 174 \times 365$. Based on our analysis, our algorithm outperforms the best SOTA baseline with statistical significance at a confidence level of at least 99.5%.