Collaborative Deterministic–Probabilistic Forecasting for Real-World Spatiotemporal Systems

Anonymous Author(s)
Affiliation
Address
email

Abstract

Probabilistic forecasting is crucial for real-world spatiotemporal systems, such as climate, energy, and urban environments, where quantifying uncertainty is essential for informed, risk-aware decision-making. While diffusion models have shown promise in capturing complex data distributions, their application to spatiotemporal forecasting remains limited due to complex spatiotemporal dynamics and high computational demands. In this work, we propose **CoST**, a novel framework that **Co**llaborates deterministic and diffusion models for **S**patio**T**emporal forecasting. CoST formulates a mean-residual decomposition strategy: it leverages a powerful deterministic model to capture the conditional mean and a lightweight diffusion model to learn residual uncertainties This collaborative formulation simplifies learning objectives, enhances forecasting accuracy, enables uncertainty quantification, and significantly improves computational efficiency. To address spatial heterogeneity, we further design a scale-aware diffusion mechanism to guide the diffusion process. Extensive experiments across ten real-world datasets from climate, energy, communication, and urban systems show that CoST achieves 25% performance gains over state-of-the-art baselines, while significantly reducing computational cost. Code and datasets are available at: `https://anonymous.4open.science/r/CoST_8069`.

1 Introduction

Real-world spatiotemporal systems underpin many critical domains, such as climate science, energy systems, communication networks, and urban environments. Accurate forecasting of the dynamics is essential for planning, resource allocation, and risk management [58, 5, 59, 51]. Existing approaches fall into two categories: deterministic and probabilistic forecasting. Deterministic methods estimate the conditional mean by minimizing MAE or MSE losses to capture spatiotemporal patterns [64, 37, 63]. In contrast, probabilistic methods aim to learn the full predictive distribution of observed data [46, 31, 62], enabling uncertainty quantification to support forecasting. This is particularly important in many domains, for example, in climate modeling and renewable energy, where assessing prediction reliability is essential for risk-aware decisions such as disaster preparedness and energy grid management [42, 54].

In this paper, we highlight the critical role of probabilistic forecasting in capturing uncertainty and improving the reliability of spatiotemporal predictions. However, it is non-trivial due to three challenges. First, these systems exhibit complex evolving dynamics, characterized by periodic trends, seasonal variations, and stochastic fluctuations [7, 62]. Second, these systems involve intricate spatiotemporal interactions and nonlinear dependencies [24, 63]. Third, real-world applications require both computationally efficient and scalable models [41, 51]. Recently, diffusion models have been widely adopted for probabilistic forecasting [57, 62, 46, 51]. Compared with existing

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

approaches such as Generative Adversarial Networks (GANs) [20, 15] and Variational Autoencoders (VAEs) [28, 30], diffusion models offer superior capability in capturing complex data distributions while ensuring stable training [22, 53, 23]. These advantages make diffusion models a promising alternative. However, originally developed for image generation, they face inherent limitations in capturing temporal correlations in sequential data, as evidenced in video generation [66, 45, 9, 17] and time series forecasting [62, 47, 46, 50].

To address this issue, recent efforts have explored incorporating temporal correlations as conditional inputs to guide the diffusion process [46, 50, 57], or injecting temporal priors into the noised data to explicitly model temporal correlations across time steps [31, 51, 62]. While these approaches improve temporal modeling, they remain constrained by the inherent limitations of the diffusion framework [46, 31, 47]. In contrast, we introduce a new perspective: rather than relying solely on diffusion models to capture the full data distribution, we propose a collaborative approach that combines a deterministic model and a diffusion model, leveraging their complementary strengths for probabilistic forecasting. Our design offers two key advantages. First, by leveraging powerful deterministic models to predict the conditional mean, it effectively captures the primary spatiotemporal patterns and benefits from advancements in established architectures. Second, instead of requiring the diffusion model to learn the full data distribution from scratch, we employ it to model the residuals, focusing its capacity on capturing uncertainty beyond the mean. This collaborative framework simplifies the learning objectives for each component and enhances both predictive accuracy and probabilistic expressiveness.

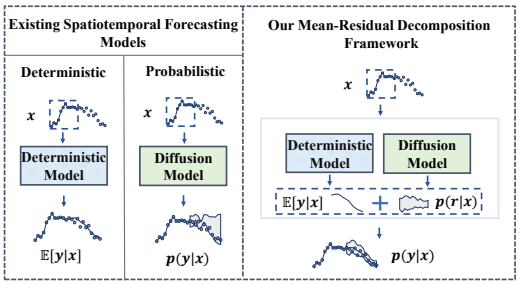

Figure 1: Comparison of existing models with our mean-residual decomposition framework.

Building on this insight, we propose **CoST**, a novel framework that **Co**llaborates deterministic and diffusion models for **S**patio**T**emporal forecasting. As illustrated in Figure 1, we first leverage an advanced deterministic spatiotemporal forecasting model to estimate the conditional mean $\mathbb{E}[y|x]$, effectively capturing the regular patterns. Based on this, we model the residual distribution $p(r|x) = p((y - \mathbb{E}[y|x])|x)$ using a diffusion model, which complements the deterministic forecasting with uncertainty quantification. Since the diffusion model focuses solely on residuals, it allows us to adopt a lightweight denoising network and mitigate the computational overhead associated with multi-step diffusion processes. To address spatial heterogeneity, we quantify differences across spatial units and introduce a scale-aware diffusion mechanism. More importantly, we propose a comprehensive evaluation protocol for spatiotemporal probabilistic forecasting by incorporating metrics such as QICE and IS, rather than relying solely on traditional measures like CRPS, MAE, and RMSE. In summary, our main contributions are as follows:

- We highlight the importance of probabilistic forecasting for complex spatiotemporal systems and introduce a novel perspective that integrates deterministic and probabilistic modeling in a collaborative framework.
- We propose **CoST**, a mean-residual decomposition approach that employs a deterministic model to estimate the conditional mean and a diffusion model to capture the residual distribution. We further design a scale-aware diffusion mechanism to address spatial heterogeneity.
- Extensive experiments on ten real-world datasets spanning climate science, energy systems, communication networks, and urban environments show that CoST consistently outperforms state-of-the-art baselines on both deterministic and probabilistic metrics, achieving an average improvement of 25% while offering notable gains in computational efficiency.

2 Related Work

Spatiotemporal deterministic forecasting. Deterministic forecasting of spatiotemporal systems focuses on point estimation. These models are typically trained with loss functions like MSE or MAE to learn the conditional mean $\mathbb{E}[y|x]$, capturing regular patterns. Common deep learning architectures include MLP-based [49, 44, 67], CNN-based [29, 34, 64], and RNN-based [2, 33, 56, 55] models,

valued for their efficiency. GNN-based methods [1, 3, 18, 25] capture spatial dependencies in graph-based data, while Transformer-based models [10, 12, 37, 61, 5] are effective at modeling complex temporal dynamics.

Spatiotemporal probabilistic forecasting. The core of probabilistic forecasting lies in modeling uncertainty, aiming to capture the full data distribution [60, 53]. This is particularly suited for modeling the stochastic nature of spatiotemporal systems. While early methods focused on Bayesian approaches, recent advances have explored generative models such as GANs [26, 48, 65], VAEs [11, 13, 68], and diffusion models [53, 8, 32]. Diffusion models, in particular, have gained traction for their ability to model complex distributions with stable training, yielding strong performance in spatiotemporal forecasting [46, 47, 50, 51].

Diffusion-based spatiotemporal probabilistic forecasting. Most diffusion-based forecasting methods formulate the task as conditional generation without explicitly modeling temporal dynamics, which hinders the generation of temporally coherent sequences [53, 16, 57, 47]. Moreover, the progressive corruption of time series during diffusion often distorts key patterns like long-term trends and periodicity, making temporal recovery difficult [62, 35]. To address this, methods such as TimeGrad [46] and TimeDiff [50] incorporate temporal embeddings as conditional inputs to enhance temporal awareness. Other approaches like NPDiff [51], TMDM [31], and Diffusion-TS [62] inject temporal priors into the diffusion process to better preserve temporal dynamics. More recently, DYffusion [47] redefines the denoising process to explicitly model temporal transitions at each diffusion step. Unlike prior methods, we avoid using diffusion to model temporal dynamics. Instead, we decouple forecasting into deterministic mean prediction and residual uncertainty estimation. The diffusion model focuses solely on the residuals, simplifying learning and allowing for a smaller denoising network, which greatly reduces the computational cost of the iterative diffusion process.

3 Preliminaries

We provide a summary of notations used in this paper in Appendix A.1 for clarity.

Spatiotemporal systems. Spatiotemporal systems underpin many domains such as climate science, energy, communication networks, and urban environments. The data recording spatiotemporal dynamics are typically represented as a tensor $\mathbf{x} \in \mathbb{R}^{T \times V \times C}$, where T, V, and C denote the temporal, spatial, and feature dimensions, respectively. Depending on the spatial structure, the data can be organized as grid-structured ($V = H \times W$) or graph-structured (where V represents the set of nodes). Given a historical context $\mathbf{x}^{co} = \mathbf{x}^{t-M+1:t}$ of length M, the goal is to predict future targets $\mathbf{x}^{ta} = \mathbf{x}^{t+1:t+P}$ over a horizon P using a model $\mathcal{F}$.

Conditional diffusion models. The diffusion-based forecasting includes a forward process and a reverse process. In the forward process, noise is added incrementally to the target data $\mathbf{x}_0^{ta}$, gradually transforming the data distribution into a standard Gaussian distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$. At any diffusion step, the corrupted target data can be computed using the one-step forward equation:

$$\mathbf{x}_n^{ta} = \sqrt{\bar{\alpha}_n}\mathbf{x}_0^{ta} + \sqrt{1-\bar{\alpha}_n}\epsilon, \quad \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{1}$$

where $\bar{\alpha}_n = \prod_{i=1}^{n} \alpha_i$ and $\alpha_n = 1 - \beta_n$. In the reverse process, prediction begins by first sampling $\mathbf{x}_N^{ta}$ from the standard Gaussian distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$, followed by a denoising procedure through the following Markov process:

$$\begin{aligned}
&p_\theta(\mathbf{x}_{0:N}^{ta}) := p(\mathbf{x}_N^{ta}) \prod_{n=1}^{N} p_\theta(\mathbf{x}_{n-1}^{ta} | \mathbf{x}_n^{ta}, \mathbf{x}_0^{co}), \\
&p_\theta(\mathbf{x}_{n-1}^{ta} | \mathbf{x}_n^{ta}) := \mathcal{N}(\mathbf{x}_{n-1}^{ta}; \mu_\theta(\mathbf{x}_n^{ta}, n | \mathbf{x}_0^{co}), \Sigma_\theta(\mathbf{x}_n^{ta}, n)), \\
&\mu_\theta(\mathbf{x}_n^{ta}, n | \mathbf{x}_0^{co}) = \frac{1}{\sqrt{\bar{\alpha}_n}} \left(\mathbf{x}_n^{ta} - \frac{\beta_n}{\sqrt{1-\bar{\alpha}_n}} \epsilon_\theta(\mathbf{x}_n^{ta}, n | \mathbf{x}_0^{co}) \right)
\end{aligned} \tag{2}$$

where the variance $\Sigma_\theta(\mathbf{x}_n^{ta}, n) = \frac{1-\bar{\alpha}_{n-1}}{1-\bar{\alpha}_n}\beta_n$, and $\epsilon_\theta(\mathbf{x}_n^{ta}, n | \mathbf{x}_0^{co})$ is predicted by the denoising network trained by the loss function below:

$$\mathcal{L}(\theta) = \mathbb{E}_{n, \mathbf{x}_0, \epsilon} \left[\left\| \epsilon - \epsilon_\theta(\mathbf{x}_n^{ta}, n | \mathbf{x}_0^{co}) \right\|_2^2 \right]. \tag{3}$$

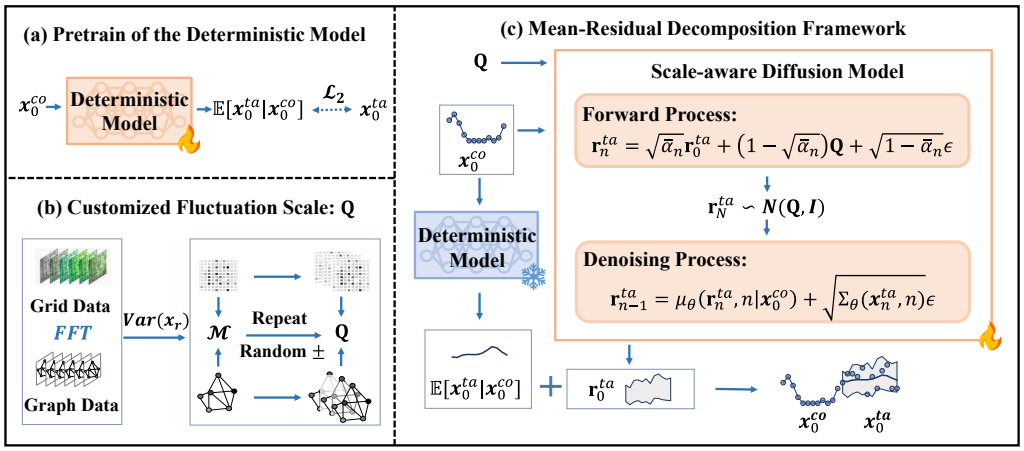

Figure 2: Overview of CoST: (a) Pretraining of the deterministic model; (b) Computation of the customized fluctuation scale; (c) Overall framework of the mean-residual decomposition.

Evaluations of probabilistic forecasting. We argue that probabilistic forecasting should be assessed from two key perspectives: *Data Distribution*—the predicted distribution should match the empirical distribution, and *Prediction Usability*—prediction intervals should achieve high coverage while remaining sharp. While metrics like CRPS, MAE and RMSE are widely used, they fail to assess: **(i)** the accuracy of quantile-wise coverage; **(ii)** whether the interval width reflects true uncertainty. To address this, we introduce Quantile Interval Coverage Error (QICE) [21] and Interval Score (IS) [19] as complementary metrics.

(i) QICE measures the mean absolute deviation between the empirical and expected proportions of ground-truth values falling into each of equal-sized quantile intervals. QICE evaluates how well the predicted distribution aligns with the expected coverage across quantiles, which is defined as follows:

$$\text{QICE} := \frac{1}{M_{\text{QIs}}} \sum_{m=1}^{M_{\text{QIs}}} \left| r_m - \frac{1}{M_{\text{QIs}}} \right|, \quad r_m = \frac{1}{N} \sum_{n=1}^{N} \mathbb{1}_{y_n \geq \hat{y}_n^{\text{low}_m}} \cdot \mathbb{1}_{y_n \leq \hat{y}_n^{\text{high}_m}}, \tag{4}$$

where $\hat{y}_n^{\text{low}_m}$ and $\hat{y}_n^{\text{high}_m}$ denote the bounds of the m-th quantile interval for y_n. Ideally, each QI should contain $1/M_{\text{QIs}}$ of the observations, yielding a QICE of 0. Lower QICE indicates better alignment between predicted and true distributions.

(ii) IS evaluates prediction interval (PI) quality by jointly accounting for sharpness and empirical coverage, and is defined as:

$$\text{IS} := \frac{1}{N} \sum_{n=1}^{N} \left[(u_n^{\alpha_{CI}} - l_n^{\alpha_{CI}}) + \frac{2}{\alpha_{CI}} (l_n^{\alpha_{CI}} - y_n) \mathbb{1}_{y_n < l_n^{\alpha_{CI}}} + \frac{2}{\alpha_{CI}} (y_n - u_n^{\alpha_{CI}}) \mathbb{1}_{y_n > u_n^{\alpha_{CI}}} \right], \tag{5}$$

where $u_n^{\alpha_{CI}}$ and $l_n^{\alpha_{CI}}$ are the upper and lower bounds of the central prediction interval for the n-th data point, derived from the corresponding predictive quantiles. A narrower interval improves the score, while missed coverage incurs a penalty scaled by α_{CI}. Lower IS indicates better performance.

4 Methodology

In this section, we propose CoST, a unified framework that combines the strengths of deterministic and diffusion models. Specifically, we first train a deterministic model to predict the conditional mean, capturing the regular spatiotemporal patterns. Then, guided by a customized fluctuation scale, we employ a scale-aware diffusion model to learn the residual distribution, enabling fine-grained uncertainty modeling. An overview of the CoST architecture is shown in Figure 2.

4.1 Theoretical Analysis of Mean-Residual Decomposition

Current diffusion-based probabilistic forecasting approaches typically employ a single diffusion model to capture the full distribution of data, incorporating both the regular spatiotemporal patterns

and the random fluctuations. However, jointly modeling these components remains challenging [62]. Inspired by [38] and the Reynolds decomposition in fluid dynamics [43], we propose to decompose the spatiotemporal data $\mathbf{x}^{ta}$ as follows:

$$\mathbf{x}^{ta} = \underbrace{\mathbb{E}[\mathbf{x}^{ta}|\mathbf{x}^{co}]}_{:=\boldsymbol{\mu}(Deterministic)} + \underbrace{(\mathbf{x}^{ta} - \mathbb{E}[\mathbf{x}^{ta}|\mathbf{x}^{co}])}_{:=\mathbf{r}(Diffusion)}, \tag{6}$$

where $\boldsymbol{\mu}$ is the conditional mean representing the regular patterns, and $\mathbf{r}$ is the residual representing the random variations. If the deterministic model approximates the conditional mean accurately, the expected residual becomes negligible, i.e. $\mathbb{E}[\mathbf{r}|\mathbf{x}^{co}] \approx 0$, and we can obtain that $\mathrm{var}(\mathbf{r}|\mathbf{x}^{co}) = \mathrm{var}(\mathbf{x}^{ta}|\mathbf{x}^{co})$. Based on the law of total variance [4], we can express the variance of the target data and residuals as:

$$\mathrm{var}(\mathbf{r}) = \mathbb{E}[\mathrm{var}(\mathbf{r}|\mathbf{x}^{co})] + \underbrace{\mathrm{var}(\mathbb{E}[\mathbf{r}|\mathbf{x}^{co}])}_{=0}, \quad \mathrm{var}(\mathbf{x}^{ta}) = \mathbb{E}[\mathrm{var}(\mathbf{x}^{ta}|\mathbf{x}^{co})] + \underbrace{\mathrm{var}(\mathbb{E}[\mathbf{x}^{ta}|\mathbf{x}^{co}])}_{\geq 0}. \tag{7}$$

Due to $\mathrm{var}(\mathbf{r}|\mathbf{x}^{co}) = \mathrm{var}(\mathbf{x}^{ta}|\mathbf{x}^{co})$, we have $\mathrm{var}(\mathbf{r}) \leq \mathrm{var}(\mathbf{x}^{ta})$. Moreover, the highly dynamic nature of the spatiotemporal system results in a larger $\mathrm{var}(\mathbb{E}[\mathbf{x}^{ta}|\mathbf{x}^{co}])$, which consequently makes $\mathrm{var}(\mathbf{r})$ smaller compared to $\mathrm{var}(\mathbf{x}^{ta})$. Our core idea is that if a deterministic model can accurately predict the conditional mean, that is, $\boldsymbol{\mu} \approx \mathbb{E}_\theta[\mathbf{x}^{ta}|\mathbf{x}]$, then the diffusion model can be dedicated solely to learning the simpler residual distribution. This design avoids the challenge diffusion models face in modeling complex spatiotemporal dynamics, while fully exploiting their strength in uncertainty estimation. By collaborating high-performing deterministic architectures and diffusion models, our method effectively captures regular dynamics and models uncertainty via residual learning.

4.2 Mean Prediction via Deterministic Model

To capture the conditional mean $\mathbb{E}_\theta[\mathbf{x}^{ta}|\mathbf{x}^{co}]$, our framework leverages existing high-performance deterministic architectures, which are designed to capture complex spatiotemporal dynamics efficiently. In our main experiments, we use the STID [49] model as the backbone for mean prediction, and also validate our framework with ConvLSTM [52], STNorm [14], and iTransformer [36] to ensure its generality (See Section 5.1). In the first stage of training, we pretrain the deterministic model for 50 epochs using historical conditional inputs $\mathbf{x}^{co}$ to output the mean estimate $\mathbb{E}_\theta[\mathbf{x}^{ta}|\mathbf{x}^{co}]$. The model is trained with the standard $\mathcal{L}_2$ loss:

$$\mathcal{L}_2 = \left\| \mathbb{E}_\theta[\mathbf{x}^{ta}|\mathbf{x}^{co}] - \mathbf{x}^{ta} \right\|_2^2. \tag{8}$$

4.3 Residual Learning via Diffusion Model

The residual distribution of spatiotemporal data is not independently and identically distributed (i.i.d.) nor does it follow a fixed distribution, such as $\mathcal{N}(0, \sigma)$. Instead, it often exhibits complex spatiotemporal dependence and heterogeneity. We use the diffusion model to focus on learning the distribution of residual $\mathbf{r}^{ta} = \mathbf{x}^{ta} - \mathbb{E}_\theta[\mathbf{x}^{ta}|\mathbf{x}^{co}]$. Accordingly, the target data $\mathbf{x}^{ta}$ for diffusion models in Eqs. (1), (2), and (3) is replaced by $\mathbf{r}^{ta}$. We incorporate timestamp information as a condition in the denoising process and concatenate the context data $\mathbf{x}_0^{co}$ with noised residual $\mathbf{r}_n^{ta}$ as input to capture real-time fluctuations. Notably, no noise is added to $\mathbf{x}_0^{co}$ during diffusion training or inference. To model the spatial patterns of the residuals, we propose a scale-aware diffusion process to further distinguish the heterogeneity for different spatial units. In this section, we detail the calculation of $\mathbf{Q}$ and how it is integrated into the scale-aware diffusion process.

(i) Customized fluctuation scale. Specifically, we apply the Fast Fourier Transform (FFT) to spatiotemporal sequences in the training set to quantify fluctuation levels in different spatial units and use the custom scale $\mathbf{Q}$ as input to account for spatial heterogeneity in residual. Specifically, we first employ FFT to extract the fluctuation components for each spatial unit within the training set. The detailed steps are as follows:

$$\mathbf{A}_{\mathrm{k}} = \left|\mathrm{FFT}(\mathbf{x})_{\mathrm{k}}\right|, \quad \phi_{\mathrm{k}} = \phi\left(\mathrm{FFT}(\mathbf{x})_{\mathrm{k}}\right), \quad \mathbf{A}_{\max} = \max_{\mathrm{k}\in\left\{1,\cdots,\lfloor\frac{L}{2}\rfloor+1\right\}} \mathbf{A}_{\mathrm{k}},$$
$$\mathcal{K} = \left\{\mathrm{k} \in \left\{1, \cdots, \left\lfloor\frac{L}{2}\right\rfloor + 1\right\} : \mathbf{A}_{\mathrm{k}} < 0.1 \times \mathbf{A}_{\max}\right\}, \tag{9}$$
$$\mathbf{x}_{\mathbf{r}}[i] = \sum_{\mathrm{k}\in\mathcal{K}} \mathbf{A}_{\mathrm{k}}\Big[\cos\left(2\pi\mathbf{f}_{\mathrm{k}} i + \phi_{\mathrm{k}}\right) + \cos\left(2\pi\bar{\mathbf{f}}_{\mathrm{k}} i + \bar{\phi}_{\mathrm{k}}\right)\Big],$$

where $\mathbf{A}_{\mathrm{k}}, \phi_{\mathrm{k}}$ reprent the amplitude and phase of the $\mathrm{k}-$th frequency component. L is the temporal length of the training set. $\mathbf{A}_{\max}$ is the maximum amplitude among the components, obtained using the $\max$ operator. $\mathcal{K}$ represents the set of indices for the selected residual components. $\mathbf{f}_{\mathrm{k}}$ is the frequency of the k-th component. $\bar{\mathbf{f}}_{\mathrm{k}}, \bar{\phi}_{\mathrm{k}}$ represent the conjugate components. $\mathbf{x}_{\mathbf{r}}$ ref to the extracted residual component of the training set. We then compute the variance σ_v^2 of the residual sequence for each location v and expand it to match the shape as $\mathbf{r}_0^{ta} \in \mathbb{R}^{B\times V\times P}$, where B represents the batch size. And we can get the variance tensor $\mathcal{M}$:

$$\mathcal{M}_{b,v,p} = \sigma_v^2, \forall b \in \{1, \cdots, B\}, \forall v \in \{1, \cdots, V\}, \forall p \in \{1, \cdots, P\}. \tag{10}$$

The residual fluctuations are bidirectional, encompassing both positive and negative variations, so we generate a random sign tensor $\mathbf{S} \in \mathbb{R}^{B\times V\times P}$ for $\mathcal{M}$, where each element $S_{b,v,p}$ of $\mathbf{S}$ is sampled from a Bernoulli distribution with $p = 0.5$. The customized fluctuation scale $\mathbf{Q}$ is computed as:

$$\mathbf{Q}_{b,v,p} = S_{b,v,p} \times \mathcal{M}_{b,v,p}, \forall b \in \{1, \cdots, B\}, \forall v \in \{1, \cdots, V\}, \forall p \in \{1, \cdots, P\}. \tag{11}$$

Then $\mathbf{Q}$ is used as the input of the denoising network.

(ii) Scale-aware diffusion process. The vanilla diffusion models assume a shared prior distribution $\mathcal{N}(0, I)$ across all spatial locations, failing to capture spatial heterogeneity. To further model such differences, we adopt the technique proposed by [21] to make the residual learning location-specific conditioned on $\mathbf{Q}$. Specifically, we redefine the noise distribution at the endpoint of the diffusion process as follows:

$$p(\mathbf{r}_N^{ta}) = \mathcal{N}(\mathbf{Q}, I), \tag{12}$$

Accordingly, the Eq (1) in the forward process is rewritten as:

$$\mathbf{r}_n^{ta} = \sqrt{\bar{\alpha}_n}\mathbf{r}_0^{ta} + (1 - \sqrt{\bar{\alpha}_n})\mathbf{Q} + \sqrt{1 - \bar{\alpha}_n}\epsilon, \quad \epsilon \sim \mathcal{N}(0, I). \tag{13}$$

And in the denoising process, we sample $\mathbf{r}_N^{ta}$ from $\mathcal{N}(\mathbf{Q}, I)$, and denoise it use Eq (2), the computation of $\mu_\theta(\mathbf{r}_n^{ta}, n|\mathbf{x}_0^{co})$ in Eq (2) is modified as:

$$\mu_\theta(\mathbf{r}_n^{ta}, n|\mathbf{x}_0^{co}) = \frac{1}{\sqrt{\bar{\alpha}_n}}\left(\mathbf{r}_n^{ta} - \frac{\beta_n}{\sqrt{1 - \bar{\alpha}_n}}\epsilon_\theta(\mathbf{r}_n^{ta}, n|\mathbf{x}_0^{co})\right) + (1 - \frac{1}{\sqrt{\bar{\alpha}_n}})\mathbf{Q}. \tag{14}$$

This modification allows the diffusion process to be conditioned on location-specific priors $\mathbf{Q}$, enhancing its ability to model spatial heterogeneity in uncertainty.

4.4 Training and Inference

Our training follows a two-stage training procedure: we first pretrain a deterministic model to predict the conditional mean, then train a diffusion model to capture the residual distribution. The full procedure is outlined in Algorithm 1. The inference consists of two paths: the pretrained deterministic model predicts the conditional mean, and the diffusion model estimates the residuals. Their outputs are combined to form the final prediction, as detailed in Algorithm 2.

5 Experiments

Datasets. We evaluate our method on ten datasets spanning four domains, including climate (SST-CESM2 and SST-ERA5), energy (SolarPower), communication (MobileNJ and MobileSH), and urban systems (CrowdBJ, CrowdBM, TaxiBJ, BikeDC and Los-Speed), each featuring distinct spatiotemporal characteristics. Detailed information on the datasets can be found in Appendix C.1.

Baselines. We compare against six representative state-of-the-art baselines commonly adopted in spatiotemporal modeling, including: **D3VAE [30]**, **DiffSTG [57]**, **TimeGrad [46]**, **CSDI [53]**, **DYffusion [47]**, and **NPDiff [51]**. Detailed descriptions of each baseline are provided in Appendix C.2.

Table 1: Short-term forecasting results in terms of CRPS, QICE, and IS. **Bold** indicates the best performance, while underlining denotes the second-best. DYffusion is limited to grid-format data, and '-' denotes results that are not applicable.

Model	Climate			MobileSH			TaxiBJ			SolarPower			CrowdBJ		
	CRPS	QICE	IS	CRPS	QICE	IS	CRPS	QICE	IS	CRPS	QICE	IS	CRPS	QICE	IS
D3VAE	0.053	0.071	15.8	0.856	0.105	1.729	0.433	0.160	985.7	0.475	0.083	731.1	0.668	0.099	53.6
DiffSTG	0.026	0.068	7.42	0.303	0.078	0.526	0.299	0.074	416.5	0.213	0.068	240.6	0.436	0.089	32.1
TimeGrad	0.042	0.147	16.0	0.489	0.143	0.759	0.170	0.102	213.2	1.000	0.128	781.7	0.385	0.113	48.6
CSDI	0.027	0.019	5.18	0.200	0.052	0.295	0.122	0.048	121.8	0.267	0.050	221.6	0.306	0.028	16.4
NPDiff	0.022	0.031	4.24	0.201	0.106	0.627	0.222	0.112	474.1	0.209	0.020	**175.3**	0.287	0.120	34.5
DYffusion	**0.020**	0.123	12.4	0.230	0.096	0.573	**0.084**	0.054	99.5	-	-	-	-	-	-
CoST	0.021	**0.009**	**4.04**	**0.147**	**0.014**	**0.215**	0.100	**0.023**	**95.3**	**0.208**	**0.019**	192.1	**0.215**	**0.014**	**11.5**

Table 2: Short-term forecasting results in terms of MAE and RMSE.

Model	Climate		MobileSH		TaxiBJ		SolarPower		CrowdBJ	
	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
D3VAE	1.75	2.31	0.186	0.373	49.3	84.8	60.1	122.8	5.16	10.1
DiffSTG	0.90	1.13	0.066	0.103	41.8	69.4	31.1	63.8	3.68	6.63
TimeGrad	1.31	1.48	0.047	0.053	29.1	34.1	39.3	94.8	4.37	5.43
CSDI	0.94	1.20	0.044	0.075	18.2	31.6	38.8	69.6	2.71	5.51
NPDiff	0.79	1.07	0.037	0.057	26.7	52.2	32.1	53.6	2.05	3.27
DYffusion	0.86	1.07	0.050	0.072	**12.3**	**18.0**	-	-	-	-
CoST	**0.74**	**0.96**	**0.033**	**0.051**	15.1	25.6	**29.7**	**51.9**	**1.92**	**3.04**

Metrics. To evaluate the performance, we employ two deterministic metrics, MAE and RMSE, along with three probabilistic metrics: CRPS, QICE and IS. For QICE, we set the number of QIs, denoted as M_{QIs}, to 10. We choose 10 bins for QICE to align with the original proposal [21]. which offers a balanced trade-off between granularity and stability. For IS, we choose a confidence level of 90% (i.e., $\alpha_{CI} = 0.1$) following common practice in spatiotemporal forecasting tasks [53, 46].

Experimental configuration. We define the short-term forecasting task as predicting the next 12 time steps based on the previous 12 observations, following [51, 57]. Since the temporal granularity varies across datasets, the actual time duration corresponding to these 12 steps differs accordingly. In addition to the standard 12-step setting commonly used in spatiotemporal forecasting, we evaluate long-term forecasting by predicting 64 future steps based on the preceding 64 observations, following [63, 27, 40]. Detailed training and model configurations are provided in Appendix C.3.

5.1 Spatiotemporal Probabilistic Forecasting

Short-term forecasting. Table 1 presents the results of probabilistic metrics for selected datasets. Due to space constraints, the remaining results are in Appendix Table 6. As shown in Table 1, CoST consistently outperforms baseline methods across all evaluated datasets. Compared to the best-performing baseline methods on each dataset, CoST demonstrates an average improvement of 17.4% in CRPS and 46.6% in QICE metrics, indicating its superior ability to accurately capture the true distribution characteristics. Moreover, CoST achieves a 16.5% improvement in the IS metric, suggesting that its prediction intervals not only maintain compactness but also exhibit higher coverage, thereby better reflecting the uncertainty of data. Although certain individual metrics may not reach the optimal level on specific datasets, CoST consistently maintains performance comparable to the best methods. Beyond probabilistic metrics, we also report deterministic evaluation results (MAE and RMSE) in Table 2 and Appendix Table 7. The results show that our method achieves an average reduction of 7% in MAE and 6.1% in RMSE datasets. This suggests that the integration with a strong conditional mean estimator enables CoST to better capture regular patterns compared to other probabilistic baseline models.

Long-term forecasting. As shown in Appendix Table 8, CoST achieves substantial improvements in long-term forecasting under probabilistic metrics, with an improvement of 15.0% and 70.4% in terms of CRPS and QICE. Despite adopting a simple MLP architecture, CoST achieves higher overall accuracy than CSDI, a Transformer-based model tailored for capturing long-range dependencies. Furthermore, it provides significantly better training efficiency and inference speed, as detailed in Section 5.4. In addition, CoST performs well on deterministic metrics (Appendix Table 9), achieving an average reduction of 9.0% in MAE and 11.0% in RMSE compared to the best-performing baseline.

Framework generalization. To demonstrate the generality of CoST, we instantiate it with four representative spatiotemporal forecasting models: STID [49], STNorm [14], ConvLSTM [52], and

Table 3: Performance of different deterministic backbone models within the CoST framework. 'Diffusion (w/o m)' denotes the results obtained using a single diffusion model.

Model	Climate					MobileNJ					BikeDC				
	MAE	RMSE	CRPS	QICE	IS	MAE	RMSE	CRPS	QICE	IS	MAE	RMSE	CRPS	QICE	IS
Diffussion (w/o m)	1.070	1.361	0.030	0.030	6.58	0.195	0.6711	0.159	0.036	1.364	2.387	10.79	1.090	0.059	12.6
+iTransformer	0.818	1.088	0.023	0.018	4.83	0.122	0.207	0.123	0.021	0.815	0.526	2.23	0.454	0.035	3.82
Reduction	23.6%	20.1%	23.3%	40.0%	26.6%	37.4%	69.2%	22.6%	41.7%	40.2%	78.0%	79.3%	58.3%	40.7%	69.7%
+ ConvLSTM	0.889	1.151	0.027	0.024	5.54	0.137	0.231	0.120	0.025	0.913	0.454	2.01	0.443	0.037	6.07
Reduction	16.9%	15.4%	10.0%	20.0%	15.8%	29.7%	65.6%	24.5%	30.6%	33.1%	81.0%	81.4%	59.4%	37.3%	51.8%
+STNorm	0.819	1.066	0.023	0.007	4.52	0.144	0.276	0.123	0.016	0.825	0.600	2.71	0.500	0.029	3.74
Reduction	23.5%	21.7%	23.3%	76.7%	31.3%	26.2%	58.9%	22.6%	55.6%	39.5%	74.9%	74.9%	54.1%	50.8%	70.3%

iTransformer [36]. These models cover a diverse set of deep learning architectures, including CNNs, RNNs, MLPs, and Transformers. As shown in Table 3, CoST consistently enhances the performance of these backbones by effectively integrating deterministic and probabilistic modeling. Compared to using a single diffusion model, CoST yields more accurate predictions and better-calibrated uncertainty estimates, validating the framework's broad applicability and effectiveness.

Case study of SST forecasting. To assess our model's ability to quantify uncertainty under complex climate dynamics, we evaluate its performance in a key region for ENSO-related Sea Surface Temperature (SST) forecasting. As shown in Figure 3, our model produces high-fidelity SST forecasts that closely match ground truth across both warm pool and cold tongue regions. In addition to accurate mean predictions, it provides well-calibrated uncertainty estimates, revealing elevated variance in the central equatorial Pacific, especially near 0° latitude and 140°–130°W, where sharp thermocline gradients and nonlinear feedbacks make forecasting particularly challenging. These high-uncertainty areas align with known regions of model divergence in climate science [39, 6, 7], demonstrating that our method delivers both accurate predictions and geophysically consistent uncertainty estimates.

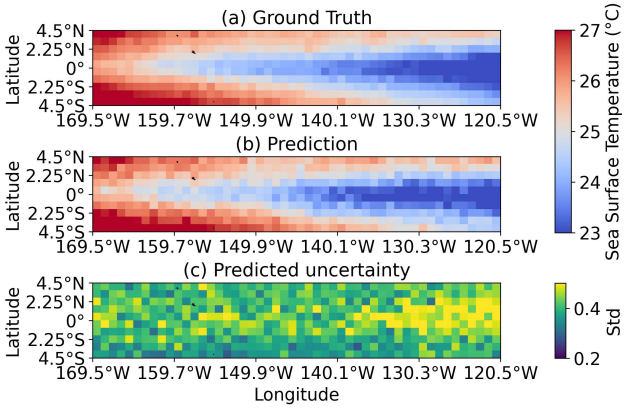

Figure 3: (a) and (b) show the ground-truth and predicted value of SST, and (c) displays the spatial distribution of forecasting uncertainty.

5.2 Ablation Study

We perform an ablation study to assess the contribution of each proposed module. Specifically, we construct three model variants by progressively removing key components: **(w/o s):** removes the scale-aware diffusion process; **(w/o q)** excludes the customized fluctuation scale as a prior; **(w/o m)** removes the conditional mean predictor, relying solely on the diffusion model. We conduct experiments on two datasets and visualize the results in terms of CRPS and IS metrics, as illustrated in Appendix Figure 8. Results show that the deterministic predictor notably improves performance by capturing regular spatiotemporal patterns, while also reducing the diffusion model's complexity. Adding the customized fluctuation scale further enhances accuracy, indicating its utility in providing valuable fluctuation information across different spatial units. And the scale-aware diffusion process enables the diffusion model to better utilize this condition.

5.3 Qualitative Analysis

Analysis of distribution alignment. As shown in Figure 4, the ground truth exhibits clear spatiotemporal multi-modality. In Figure 4(a), three peaks likely correspond to different time points or varying states at the same time. CoST accurately captures all three peaks, while CSDI only fits two, showing CoST's superior multi-modal modeling. In Figure 4(b), both models capture two peaks, but CoST aligns better with the peak spacing in the true distribution, reflecting stronger temporal

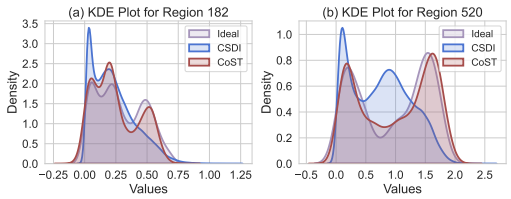

Figure 4: KDE plots of the MobileSH dataset for different regions: (a) Region 182, (b) Region 520.

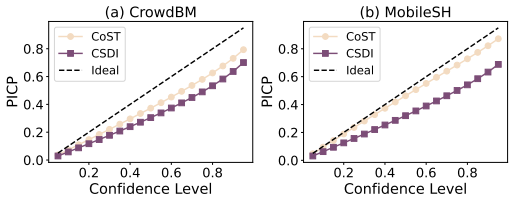

Figure 5: PICP comparison between our model and CSDI on CrowdBM and MobileSH.

sensitivity. These strengths arise from CoST's hybrid design: the diffusion component models residual uncertainty to capture multi-modal traits, while the deterministic backbone learns regular trends. See Appendix C.5.1 for more analysis and results.

Analysis of prediction quality. To intuitively demonstrate the effectiveness of our predictions, we visualize results on the CrowdBJ dataset in Figure 6, comparing our model with the best baseline, CSDI. As shown in Figures 6 (a, c, f), our model, aided by a deterministic backbone, better captures regular spatiotemporal patterns. Meanwhile, the diffusion module enhances uncertainty modeling by focusing on residuals, as reflected in Figures 6 (b, d, e). Beyond sample-level comparison, we evaluate prediction interval calibration via dynamic quantile error curves on CrowdBM and MobileSH (Figure 5). For each confidence level α, we compute the corresponding quantile interval and its Prediction Interval Coverage Probability (PICP). Closer alignment with the diagonal (black dashed line) indicates better calibration. Our model consistently outperforms CSDI in this regard.

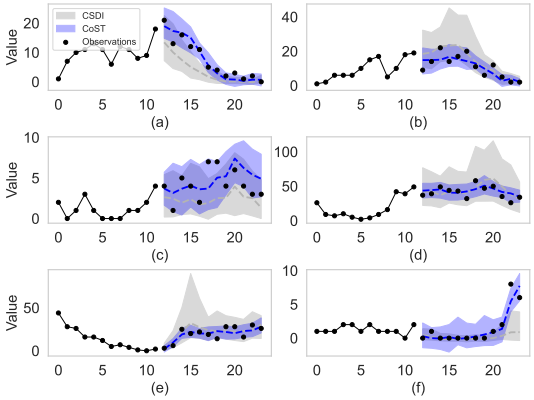

Figure 6: Visualizations of predictive uncertainty for both CSDI and CoST on the CrowdBJ dataset. The shaded regions represent the 90% confidence interval. The dashed lines denote the median of the predicted values for each model.

5.4 Computational Cost

We benchmark training and inference time (including 50 sampling iterations and pretraining for our mean predictor) on the MobileSH dataset. As shown in Appendix Table 10, our method achieves markedly higher efficiency than existing probabilistic models in both training and inference. This efficiency is particularly advantageous for real-world applications such as mobile traffic prediction. Notably, CSDI leverages the Transformer's expressive power but incurs substantial computational cost, limiting its applicability in time-sensitive settings.

6 Conclusion

In this work, we highlight the importance of probabilistic forecasting for complex spatiotemporal systems and propose CoST, a collaborative framework that integrates deterministic and diffusion models. By decomposing data into a conditional mean and residual component, CoST bridges deterministic and probabilistic modeling, enabling accurate capture of both regular patterns and uncertainties. Extensive experiments on seven real-world datasets show that CoST consistently outperforms state-of-the-art methods with an average improvement of 25%. Our approach offers an effective solution for combining precise pattern learning with uncertainty modeling in spatiotemporal forecasting.

Limitations and future work. CoST relies on a strong deterministic backbone, which may limit its applicability in domains lacking mature models. Moreover, it has not yet been validated on complex physical systems governed by PDEs or coupled dynamics. Future work will explore physics-informed extensions, adaptive decomposition, and more generalizable architectures.

References

[1] Lei Bai, Lina Yao, Salil Kanhere, Xianzhi Wang, Quan Sheng, et al. Stg2seq: Spatial-temporal graph to sequence model for multi-step passenger demand forecasting. *arXiv preprint arXiv:1905.10069*, 2019.

[2] Lei Bai, Lina Yao, Salil S Kanhere, Zheng Yang, Jing Chu, and Xianzhi Wang. Passenger demand forecasting with multi-task convolutional recurrent neural networks. In *Advances in Knowledge Discovery and Data Mining: 23rd Pacific-Asia Conference, PAKDD 2019, Macau, China, April 14-17, 2019, Proceedings, Part II 23*, pages 29–42. Springer, 2019.

[3] Lei Bai, Lina Yao, Can Li, Xianzhi Wang, and Can Wang. Adaptive graph convolutional recurrent network for traffic forecasting. *Advances in neural information processing systems*, 33:17804–17815, 2020.

[4] Dimitri Bertsekas and John N Tsitsiklis. *Introduction to probability*, volume 1. Athena Scientific, 2008.

[5] Oussama Boussif, Ghait Boukachab, Dan Assouline, Stefano Massaroli, Tianle Yuan, Loubna Benabbou, and Yoshua Bengio. Improving* day-ahead* solar irradiance time series forecasting by leveraging spatio-temporal context. *Advances in Neural Information Processing Systems*, 36:2342–2367, 2023.

[6] Mark A Cane. The evolution of el niño, past and future. *Earth and Planetary Science Letters*, 230(3-4):227–240, 2005.

[7] Dan Cao, Jiahua Zhang, Lan Xun, Shanshan Yang, Jingwen Wang, and Fengmei Yao. Spatiotemporal variations of global terrestrial vegetation climate potential productivity under climate change. *Science of The Total Environment*, 770:145320, 2021.

[8] Haoye Chai, Tao Jiang, and Li Yu. Diffusion model-based mobile traffic generation with open data for network planning and optimization. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4828–4838, 2024.

[9] Pascal Chang, Jingwei Tang, Markus Gross, and Vinicius C Azevedo. How i warped your noise: a temporally-correlated noise prior for diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024.

[10] Changlu Chen, Yanbin Liu, Ling Chen, and Chengqi Zhang. Bidirectional spatial-temporal adaptive transformer for urban traffic flow forecasting. *IEEE Transactions on Neural Networks and Learning Systems*, 34(10):6913–6925, 2022.

[11] Jiayuan Chen, Shuo Zhang, Xiaofei Chen, Qiao Jiang, Hejiao Huang, and Chonglin Gu. Learning traffic as videos: a spatio-temporal vae approach for traffic data imputation. In *International Conference on Artificial Neural Networks*, pages 615–627. Springer, 2021.

[12] Weihuang Chen, Fangfang Wang, and Hongbin Sun. S2tnet: Spatio-temporal transformer networks for trajectory prediction in autonomous driving. In *Asian conference on machine learning*, pages 454–469. PMLR, 2021.

[13] Miguel Ángel De Miguel, José María Armingol, and Fernando Garcia. Vehicles trajectory prediction using recurrent vae network. *IEEE Access*, 10:32742–32749, 2022.

[14] Jinliang Deng, Xiusi Chen, Renhe Jiang, Xuan Song, and Ivor W Tsang. St-norm: Spatial and temporal normalization for multi-variate time series forecasting. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pages 269–278, 2021.

[15] Nan Gao, Hao Xue, Wei Shao, Sichen Zhao, Kyle Kai Qin, Arian Prabowo, Mohammad Saiedur Rahaman, and Flora D Salim. Generative adversarial networks for spatio-temporal data: A survey. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 13(2):1–25, 2022.

[16] Zhihan Gao, Xingjian Shi, Boran Han, Hao Wang, Xiaoyong Jin, Danielle Maddix, Yi Zhu, Mu Li, and Yuyang Bernie Wang. Prediff: Precipitation nowcasting with latent diffusion models. *Advances in Neural Information Processing Systems*, 36:78621–78656, 2023.

[17] Songwei Ge, Seungjun Nah, Guilin Liu, Tyler Poon, Andrew Tao, Bryan Catanzaro, David Jacobs, Jia-Bin Huang, Ming-Yu Liu, and Yogesh Balaji. Preserve your own correlation: A noise prior for video diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22930–22941, 2023.

[18] Xu Geng, Yaguang Li, Leye Wang, Lingyu Zhang, Qiang Yang, Jieping Ye, and Yan Liu. Spatiotemporal multi-graph convolution network for ride-hailing demand forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 3656–3663, 2019.

[19] Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, 102(477):359–378, 2007.

[20] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[21] Xizewen Han, Huangjie Zheng, and Mingyuan Zhou. Card: Classification and regression diffusion models. *Advances in Neural Information Processing Systems*, 35:18100–18115, 2022.

[22] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[23] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in Neural Information Processing Systems*, 35:8633–8646, 2022.

[24] Zhanhong Jiang, Chao Liu, Adedotun Akintayo, Gregor P Henze, and Soumik Sarkar. Energy prediction using spatiotemporal pattern networks. *Applied Energy*, 206:1022–1039, 2017.

[25] Guangyin Jin, Yuxuan Liang, Yuchen Fang, Zezhi Shao, Jincai Huang, Junbo Zhang, and Yu Zheng. Spatio-temporal graph neural networks for predictive learning in urban computing: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 2023.

[26] Junchen Jin, Dingding Rong, Tong Zhang, Qingyuan Ji, Haifeng Guo, Yisheng Lv, Xiaoliang Ma, and Fei-Yue Wang. A gan-based short-term link traffic prediction approach for urban road networks under a parallel learning framework. *IEEE Transactions on Intelligent Transportation Systems*, 23(9):16185–16196, 2022.

[27] Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, et al. Time-llm: Time series forecasting by reprogramming large language models. *arXiv preprint arXiv:2310.01728*, 2023.

[28] Diederik P Kingma. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

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

[30] Yan Li, Xinjiang Lu, Yaqing Wang, and Dejing Dou. Generative time series forecasting with diffusion, denoise, and disentanglement. *Advances in Neural Information Processing Systems*, 35:23009–23022, 2022.

[31] Yuxin Li, Wenchao Chen, Xinyue Hu, Bo Chen, Mingyuan Zhou, et al. Transformer-modulated diffusion models for probabilistic multivariate time series forecasting. In *The Twelfth International Conference on Learning Representations*, 2024.

[32] Lequan Lin, Zhengkun Li, Ruikun Li, Xuliang Li, and Junbin Gao. Diffusion models for time-series applications: a survey. *Frontiers of Information Technology & Electronic Engineering*, 25(1):19–41, 2024.

[33] Zhihui Lin, Maomao Li, Zhuobin Zheng, Yangyang Cheng, and Chun Yuan. Self-attention convlstm for spatiotemporal prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 11531–11538, 2020.

[34] Lingbo Liu, Ruimao Zhang, Jiefeng Peng, Guanbin Li, Bowen Du, and Liang Lin. Attentive crowd flow machines. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 1553–1561, 2018.

[35] Shuai Liu, Xiucheng Li, Gao Cong, Yile Chen, and Yue Jiang. Multivariate time-series imputation with disentangled temporal representations. In *The Eleventh international conference on learning representations*, 2023.

[36] Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. In *The Twelfth International Conference on Learning Representations*.

[37] Ziqing Ma, Wenwei Wang, Tian Zhou, Chao Chen, Bingqing Peng, Liang Sun, and Rong Jin. Fusionsf: Fuse heterogeneous modalities in a vector quantized framework for robust solar power forecasting. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 5532–5543, 2024.

[38] Morteza Mardani, Noah Brenowitz, Yair Cohen, Jaideep Pathak, Chieh-Yu Chen, Cheng-Chin Liu, Arash Vahdat, Mohammad Amin Nabian, Tao Ge, Akshay Subramaniam, et al. Residual corrective diffusion modeling for km-scale atmospheric downscaling, 2024. *URL https://arxiv.org/abs/2309.15214*, 2023.

[39] Michael J McPhaden, Stephen E Zebiak, and Michael H Glantz. Enso as an integrating concept in earth science. *science*, 314(5806):1740–1745, 2006.

[40] Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. *arXiv preprint arXiv:2211.14730*, 2022.

[41] Tim Palmer. Climate forecasting: Build high-resolution global climate models. *Nature*, 515(7527):338–339, 2014.

[42] TN Palmer. Towards the probabilistic earth-system simulator: A vision for the future of climate and weather prediction. *Quarterly Journal of the Royal Meteorological Society*, 138(665):841–861, 2012.

[43] Stephen B Pope. Turbulent flows. *Measurement Science and Technology*, 12(11):2020–2021, 2001.

[44] Yanjun Qin, Haiyong Luo, Fang Zhao, Yuchen Fang, Xiaoming Tao, and Chenxing Wang. Spatio-temporal hierarchical mlp network for traffic forecasting. *Information Sciences*, 632:543–554, 2023.

[45] Haonan Qiu, Menghan Xia, Yong Zhang, Yingqing He, Xintao Wang, Ying Shan, and Ziwei Liu. Freenoise: Tuning-free longer video diffusion via noise rescheduling. *arXiv preprint arXiv:2310.15169*, 2023.

[46] Kashif Rasul, Calvin Seward, Ingmar Schuster, and Roland Vollgraf. Autoregressive denoising diffusion models for multivariate probabilistic time series forecasting. In *International Conference on Machine Learning*, pages 8857–8868. PMLR, 2021.

[47] Salva Rühling Cachay, Bo Zhao, Hailey Joren, and Rose Yu. Dyffusion: A dynamics-informed diffusion model for spatiotemporal forecasting. *Advances in neural information processing systems*, 36:45259–45287, 2023.

[48] Divya Saxena and Jiannong Cao. D-gan: Deep generative adversarial nets for spatio-temporal prediction. *arXiv preprint arXiv:1907.08556*, 2019.

[49] Zezhi Shao, Zhao Zhang, Fei Wang, Wei Wei, and Yongjun Xu. Spatial-temporal identity: A simple yet effective baseline for multivariate time series forecasting. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 4454–4458, 2022.

[50] Lifeng Shen and James Kwok. Non-autoregressive conditional diffusion models for time series prediction. In *International Conference on Machine Learning*, pages 31016–31029. PMLR, 2023.

[51] Zhi Sheng, Yuan Yuan, Jingtao Ding, and Yong Li. Unveiling the power of noise priors: Enhancing diffusion models for mobile traffic prediction. *arXiv preprint arXiv:2501.13794*, 2025.

[52] 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.

[53] Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. Csdi: Conditional score-based diffusion models for probabilistic time series imputation. *Advances in Neural Information Processing Systems*, 34:24804–24816, 2021.

[54] Lucas R Vargas Zeppetello, Adrian E Raftery, and David S Battisti. Probabilistic projections of increased heat stress driven by climate change. *Communications Earth & Environment*, 3(1):183, 2022.

[55] Yunbo Wang, Zhifeng Gao, Mingsheng Long, Jianmin Wang, and S Yu Philip. Predrnn++: Towards a resolution of the deep-in-time dilemma in spatiotemporal predictive learning. In *International conference on machine learning*, pages 5123–5132. PMLR, 2018.

[56] Yunbo Wang, Mingsheng Long, Jianmin Wang, Zhifeng Gao, and Philip S Yu. Predrnn: Recurrent neural networks for predictive learning using spatiotemporal lstms. *Advances in neural information processing systems*, 30, 2017.

[57] Haomin Wen, Youfang Lin, Yutong Xia, Huaiyu Wan, Qingsong Wen, Roger Zimmermann, and Yuxuan Liang. Diffstg: Probabilistic spatio-temporal graph forecasting with denoising diffusion models. In *Proceedings of the 31st ACM International Conference on Advances in Geographic Information Systems*, pages 1–12, 2023.

[58] Peng Xie, Tianrui Li, Jia Liu, Shengdong Du, Xin Yang, and Junbo Zhang. Urban flow prediction from spatiotemporal data using machine learning: A survey. *Information Fusion*, 59:1–12, 2020.

[59] Lei Xu, Nengcheng Chen, Zeqiang Chen, Chong Zhang, and Hongchu Yu. Spatiotemporal forecasting in earth system science: Methods, uncertainties, predictability and future directions. *Earth-Science Reviews*, 222:103828, 2021.

[60] Yiyuan Yang, Ming Jin, Haomin Wen, Chaoli Zhang, Yuxuan Liang, Lintao Ma, Yi Wang, Chenghao Liu, Bin Yang, Zenglin Xu, et al. A survey on diffusion models for time series and spatio-temporal data. *arXiv preprint arXiv:2404.18886*, 2024.

[61] Cunjun Yu, Xiao Ma, Jiawei Ren, Haiyu Zhao, and Shuai Yi. Spatio-temporal graph transformer networks for pedestrian trajectory prediction. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII 16*, pages 507–523. Springer, 2020.

[62] Xinyu Yuan and Yan Qiao. Diffusion-ts: Interpretable diffusion for general time series generation. *arXiv preprint arXiv:2403.01742*, 2024.

[63] Yuan Yuan, Jingtao Ding, Jie Feng, Depeng Jin, and Yong Li. Unist: A prompt-empowered universal model for urban spatio-temporal prediction. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4095–4106, 2024.

[64] Junbo Zhang, Yu Zheng, and Dekang Qi. Deep spatio-temporal residual networks for citywide crowd flows prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31, 2017.

[65] Liang Zhang, Jianqing Wu, Jun Shen, Ming Chen, Rui Wang, Xinliang Zhou, Cankun Xu, Quankai Yao, and Qiang Wu. Satp-gan: Self-attention based generative adversarial network for traffic flow prediction. *Transportmetrica B: Transport Dynamics*, 9(1):552–568, 2021.

[66] Zhongwei Zhang, Fuchen Long, Yingwei Pan, Zhaofan Qiu, Ting Yao, Yang Cao, and Tao Mei. Trip: Temporal residual learning with image noise prior for image-to-video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8671–8681, 2024.

[67] Zijian Zhang, Ze Huang, Zhiwei Hu, Xiangyu Zhao, Wanyu Wang, Zitao Liu, Junbo Zhang, S Joe Qin, and Hongwei Zhao. Mlpst: Mlp is all you need for spatio-temporal prediction. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 3381–3390, 2023.

[68] Fan Zhou, Qing Yang, Ting Zhong, Dajiang Chen, and Ning Zhang. Variational graph neural networks for road traffic prediction in intelligent transportation systems. *IEEE Transactions on Industrial Informatics*, 17(4):2802–2812, 2020.

A Background

A.1 Glossary

We summarize all notations and symbols used throughout the paper in Table 4.

Table 4: Glossary of notations and symbols used in this paper.

Symbol	Used for
$\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$	Graph structure where $\mathcal{V}$ is the node set, $\mathcal{E}$ is the edge set, and $\mathbf{A}$ is the adjacency matrix.
$\mathbf{x} \in \mathbb{R}^{T \times V \times C}$	Spatiotemporal data.
T	The length of spatiotemporal series.
V	The number of spatial units.
C	The number of feature dimensions.
B	Batch size.
P	Prediction horizon.
M	Historical horizon.
N	The number of diffusion steps.
H	Height of the grid-based data
W	Width of the grid-based data
$\mathbf{Q}$	Customized fluctuation scale.
$\mathcal{M}$	The variance tensor.
$\mathbf{S}$	The random sign tensor.
$\{\cdot\}^{\mathrm{co}}$	Historical (conditional) term.
$\{\cdot\}^{\mathrm{ta}}$	Predicted (target) term.
$\{\cdot\}_{\mathrm{n}}$	Noisy data at n-th diffusion step.
μ	Mean.
$\mathbf{r}$	Residual.
ϵ	Gaussian noise.
$\mathcal{K}$	K The set of indices for the selected FFT components
$\{\beta_n\}_{n=1}^{N}$	The noise schedule in the diffusion process.
$\alpha_n, \bar{\alpha}_n$	$\alpha_n = 1 - \beta_n$, $\bar{\alpha}_n = \prod_{i=1}^{n} \alpha_i$.
$\epsilon_\theta(\cdot)$	The denoising network with parameter θ.
α_{CI}	Significance level for the prediction interval.
$\mathbb{1}_{(\cdot)}$	Indicator function, which takes the value 1 when a certain condition is true, and 0 when the condition is false.

A.2 Spatiotemporal Data

Spatiotemporal data typically come in two forms: (i) **Grid-based data**, where the spatial dimension V can be expressed in a two-dimensional form as $H \times W$, with H and W denoting height and width, respectively. (ii) **Graph-based data**, where V denotes the number of nodes in a spatial graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$, defined by its set of nodes $\mathcal{V}$, the set of edges $\mathcal{E}$ and the adjacency matrix $\mathcal{A}$. Its elements a_{ij} show if there's an edge between node i and j in $\mathcal{V}$, $a_{ij} = 1$ when there's an edge and $a_{ij} = 0$ otherwise.

B Methodology

B.1 Algorithm

The training and inference procedures of CoST are summarized in Algorithm 1 and Algorithm 2, respectively.

C Experiments

C.1 Datasets

In our experiments, we evaluate the proposed method on ten real-world datasets across four domains: **climate**, **energy**, **communication systems**, and **urban systems**. For climate forecasting, we train our models on the simulated SST-CESM2 dataset and evaluate them on the observational SST-ERA5 dataset, using the first 30 years for validation and the remaining years for testing. The remaining datasets are partitioned into training, validation, and test sets with a 6:2:2 ratio, and all datasets are standardized during training.Table 5 provides a summary of the datasets. The details are as follows:

Algorithm 1 Training

Stage 1: Pretraining of Deterministic Model $\mathbb{E}_\theta$
repeat
  Estimate the conditional mean $\mathbb{E}_\theta[\mathbf{x}_0^{ta}|\mathbf{x}_0^{co}]$.
  Update $\mathbb{E}_\theta$ using the following loss function:

$$\mathcal{L}_2 = \left\|\mathbb{E}_\theta[\mathbf{x}_0^{ta}|\mathbf{x}_0^{co}] - \mathbf{x}_0^{ta}\right\|_2^2$$

until The model has converged.
Stage 2: Training of Diffusion Model ϵ_θ
repeat
  Initialize $n \sim \text{Uniform}(1, \ldots, N)$ and $\epsilon \sim \mathcal{N}(0, I)$.
  Calculate the target $\mathbf{r}_0^{ta} = \mathbf{x}_0^{ta} - \mathbb{E}_\theta[\mathbf{x}_0^{ta}|\mathbf{x}_0^{co}]$.
  Calculate noisy targets $\mathbf{r}_n^{ta}$ using Eq. (13).
  Update ϵ_θ using the following loss function:

$$\mathcal{L}(\theta) = \left\|\epsilon - \epsilon_\theta(\mathbf{r}_n^{ta}, n|\mathbf{x}_0^{co})\right\|_2^2$$

until The model has converged.

Algorithm 2 Inference

Input: Context data $\mathbf{x}_0^{co}$, customized fluctuation scale $\mathbf{Q}$, trained diffusion model ϵ_θ, trained deterministic model $\mathbb{E}_\theta$
Output: Target data $\mathbf{x}_0^{ta}$
Estimate the conditional mean $\mathbb{E}_\theta[\mathbf{x}_0^{ta}|\mathbf{x}_0^{co}]$
Sample $\mathbf{r}_N^{ta}$ from $\epsilon \sim \mathcal{N}(\mathbf{Q}, I)$
for $n = N$ to 1 **do**
  Estimate the noise $\epsilon_\theta(\mathbf{r}_n^{ta}, n|\mathbf{x}_0^{co})$
  Calculate the $\mu_\theta(\mathbf{r}_n^{ta}, n|\mathbf{x}_0^{co})$ using Eq. (14)
  Sample $\mathbf{r}_{n-1}^{ta}$ using Eq. (2)
end for
Return: $\mathbf{x}_0^{ta} = \mathbb{E}_\theta[\mathbf{x}_0^{ta}|\mathbf{x}_0^{co}] + \mathbf{r}_0^{ta}$

- **Climate.** We utilize two datasets for sea surface temperature (SST) prediction in the Niño 3.4 region (5°S–5°N, 170°W–120°W), which is widely used for monitoring El Niño events: (i) SST-CESM2, simulated SST data from the CESM2-FV2 model of the CMIP6 project, covering the period from 1850 to 2014, with a spatial resolution of $1^\circ \times 1^\circ$. (ii) SST-ERA5: reanalysis data from ERA5, containing SST and 10-meter wind speed (U10/V10) variables from 1940 to 2025, with an original spatial resolution of approximately $0.25^\circ \times 0.25^\circ$. All data are regridded to a $1^\circ \times 1^\circ$ resolution for consistency. The CESM2 data are used for training, while the first 30 years of ERA5 are used for validation and the remaining years for testing.
- **Energy.** This dataset contains real-time meteorological measurements and photovoltaic (PV) power output collected from a PV power station in China, spanning from March 1st to December 31st, 2024. The features include: total active power output of the PV grid-connection point (P), ambient temperature, back panel temperature, dew point, relative humidity, atmospheric pressure, global horizontal irradiance (GHI), diffuse and direct radiation, wind direction and wind speed. Our forecasting task focuses on GHI, which is the key variable for solar power prediction. Due to data privacy restrictions, the raw dataset cannot be publicly released.
- **Communication Systems.** Mobile communication traffic datasets are collected from two major cities in Shanghai and Nanjing, capturing the spatiotemporal dynamics of network usage patterns.
- **Urban Systems.** We adopt five widely used public datasets representing various urban sensing signals: (i) CrowdBJ and CrowdBM, crowd flow data from Beijing and Baltimore, respectively. (ii) TaxiBJ, taxi trajectory-based traffic flow data from Beijing. (iii) BikeDC, bike-sharing demand data from Washington D.C. (iv) Los-Speed, traffic speed data from the Los Angeles road network. These datasets have been extensively used in spatiotemporal forecasting research and provide diverse signals for evaluating model generality across cities and domains.

Table 5: The basic information of grid-based spatio-temporal data.

Dataset	Location	Type	Temporal Period	Spatial partition	Interval
SST-CESM2	Global (Niño 3.4)	Simulated SST	1850-2014	$1° \times 1°$	Monthly
SST-ERA5	Global (Niño 3.4)	Reanalysis SST / U10 / V10	1940-2025	$0.25° \times 0.25°$	Monthly
SolarPower	China (a PV station)	GHI / Weather / PV power	2024/03/01 - 2024/12/31	Station-level	15 min
TaxiBJ	Beijing	Taxi flow	2014/03/01 - 2014/06/30	32×32	Half an hour
TaxiBJ	Beijing	Taxi flow	2014/03/01 - 2014/06/30	32×32	Half an hour
TaxiBJ	Beijing	Taxi flow	2014/03/01 - 2014/06/30	32×32	Half an hour
BikeDC	Washington, D.C.	Bike flow	2010/09/20 - 2010/10/20	20×20	Half an hour
MobileSH	Shanghai	Mobile traffic	2014/08/01 - 2014/08/21	32×28	One hour
MobileNJ	Nanjing	Mobile traffic	2021/02/02 - 2021/02/22	20×28	One hour
CrowdBJ	Beijing	Crowd flow	2018/01/01 - 2018/01/31	1010	One hour
CrowdBM	Baltimore	Crowd flow	2019/01/01 - 2019/05/31	403	One hour
Los-Speed	Los Angeles	Traffic speed	2012/03/01 - 2012/03/07	207	5 minutes

C.2 Baselines

We provide a brief description of the baselines used in our experiments:

- **D3VAE [30]:** Aims at short-period and noisy time series forecasting. It combines generative modeling with a bidirectional variational auto-encoder, integrating diffusion, denoising, and disentanglement.
- **DiffSTG [57]:** First applies diffusion models to spatiotemporal graph forecasting. By combining STGNNs and diffusion models, it reduces prediction errors and improves uncertainty modeling.
- **TimeGrad [46]:** An autoregressive model based on diffusion models. It conducts probabilistic forecasting for multivariate time series and performs well on real-world datasets.
- **CSDI [53]:**Utilizes score-based diffusion models for time series imputation. It can leverage the correlations of observed values and also shows remarkable results on prediction tasks.
- **DYffusion [47]:** A training method for diffusion models in probabilistic spatiotemporal forecasting. It combines data temporal dynamics with diffusion steps and performs well in complex dynamics forecasting.
- **NPDiff [51]:** A general noise prior framework for mobile traffic prediction. It uses the data dynamics to calculate noise prior for the denoising process and achieve effective performance.

C.3 Experimental Configuration

In our experiment, for our model, we set the training maximum epoch for both the deterministic model and the diffusion model to 50, with early stopping based on patience of 5 for both models. For the diffusion model, we set the validation set sampling number to 3, and the average metric computed over these samples is used as the criterion for early stopping. For the baseline models, we set the maximum training epoch to 100 and the early stopping patience also to 5. We set the number of samples to 50 for computing the experimental results presented in the paper. For the denoising network architecture, we adopt a lightweight variant of the MLP-based STID [49]. Specifically, we set the number of encoder layers to 8 and the embedding dimension to 128. The diffusion model employs a maximum of 50 diffusion steps, using a linear noise schedule with $\beta_1 = 0.0001$ and $\beta_N = 0.5$. During training, we set the initial learning rate to 0.001, and after 20 epochs, we adjust it to 4e-4. We use the Adam optimizer with a weight decay of 1e-6. All experiments are conducted with fixed random seeds. Models with lower GPU memory demands are run on NVIDIA TITAN Xp (12GB GDDR5X) and NVIDIA GeForce RTX 4090 (24GB GDDR6X) GPUs under a Linux environment. For the DYffusion [47] baseline, which requires substantially more resources, training is performed on NVIDIA A100 (80GB HBM2e) and A800 (40GB HBM2e).

C.4 Geographic Extent of the ENSO Region

To provide geographic context for the SST case study presented in Section 3, Figure 7 illustrates the global location and spatial extent of the selected region. The red box highlights the area from 4.5°S to 4.5°N and 169.5°W to 120.5°W in the central-to-eastern equatorial Pacific, a region known for strong ocean-atmosphere coupling and ENSO-related variability.

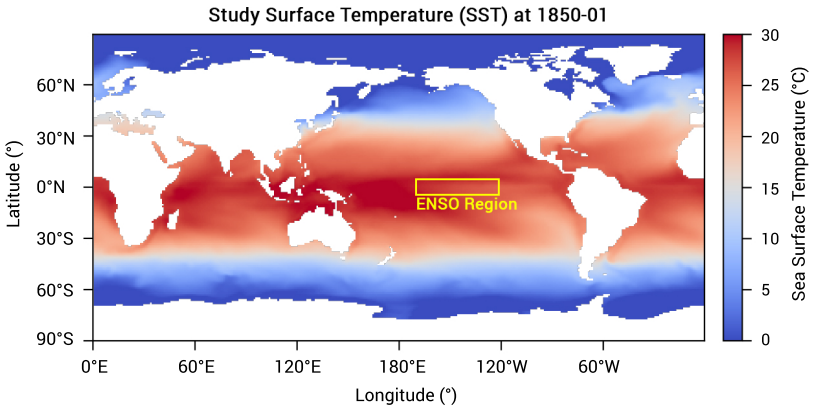

Figure 7: Global map indicating the spatial extent of ENSO region (highlighted in yellow). The region spans from 4.5°S to 4.5°N and 169.5°W to 120.5°W in the equatorial Pacific.

C.5 Additional Experimental Results

Table 6: Short-term forecasting results in terms of CRPS, QICE, and IS. **Bold** indicates the best performance, while underlining denotes the second-best. DYffusion is limited to grid-format data, and '-' denotes results that are not applicable.

Model	BikeDC			MobileNJ			CrowdBM			Los-Speed		
	CRPS	QICE	IS	CRPS	QICE	IS	CRPS	QICE	IS	CRPS	QICE	IS
D3VAE	0.785	0.157	8.77	0.565	0.096	6.03	0.593	0.110	136.4	0.119	0.089	90.5
DiffSTG	0.692	0.157	8.08	0.291	0.071	3.11	0.453	0.047	68.5	0.078	0.045	50.9
TimeGrad	0.469	0.130	5.65	0.432	0.162	5.87	**0.240**	0.085	46.9	**0.031**	0.098	**20.8**
CSDI	0.529	0.057	4.79	0.111	0.039	0.80	0.390	0.054	61.1	0.059	0.026	30.8
NPDiff	0.442	0.066	7.11	0.128	0.133	2.22	0.331	0.119	91.2	0.057	0.023	30.5
DYffusion	0.573	0.079	6.46	0.196	0.080	1.80	-	-	-	-	-	-
CoST	**0.419**	**0.028**	**3.45**	**0.089**	**0.032**	**0.66**	0.256	**0.027**	**37.8**	0.056	**0.023**	31.9

Table 7: Short-term forecasting results in terms of MAE and RMSE. **Bold** indicates the best performance, while underlining denotes the second-best. DYffusion is limited to grid-format data, and '-' denotes results that are not applicable.

Model	BikeDC		MobileNJ		CrowdBM		Los-Speed	
	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
D3VAE	0.871	3.59	0.580	1.135	11.0	24.7	8.28	11.9
DiffSTG	0.770	4.02	0.317	0.649	8.88	21.3	5.38	9.75
TimeGrad	0.843	**1.07**	0.340	0.357	10.1	12.4	**2.33**	**3.00**
CSDI	0.592	3.10	0.129	0.237	7.31	19.3	4.53	8.07
NPDiff	**0.435**	1.90	0.123	0.175	5.42	13.7	4.07	7.64
DYffusion	0.480	1.37	0.222	0.357	-	-	-	-
CoST	0.492	1.76	**0.102**	**0.172**	**5.04**	**12.1**	4.05	7.30

Table 8: Long-term forecasting results in terms of CRPS, QICE, and IS. **Bold** indicates the best performance, while underlining denotes the second-best. DYffusion is limited to grid-format data, and '-' denotes results that are not applicable.

Model	MobileSH			Climate			CrowdBJ			CrowdBM			Los-Speed		
	CRPS	QICE	IS	CRPS	QICE	IS	CRPS	QICE	IS	CRPS	QICE	IS	CRPS	QICE	IS
D3VAE	0.798	0.129	1.830	0.075	0.083	24.0	0.710	0.109	63.9	0.674	0.108	152.3	0.138	0.101	113.2
DiffSTG	0.374	0.107	0.923	0.027	0.077	7.90	0.370	0.094	31.3	0.400	0.073	67.1	0.124	0.080	104.6
TimeGrad	0.245	0.075	0.408	0.041	0.101	14.2	0.371	0.073	32.4	0.237	0.049	33.9	0.192	0.081	98.8
CSDI	0.158	0.045	**0.216**	0.036	0.073	6.80	0.229	0.038	12.0	0.235	0.052	33.7	0.134	0.090	**59.2**
NPDiff	0.204	0.102	0.611	0.109	0.115	41.3	0.288	0.114	33.6	0.331	0.111	90.8	1.366	0.126	950.4
DYffusion	0.308	0.086	0.550	0.030	0.147	15.2	-	-	-	-	-	-	-	-	-
CoST	**0.158**	**0.016**	0.218	**0.024**	**0.011**	**4.87**	**0.217**	**0.011**	**11.5**	**0.235**	**0.009**	**31.2**	**0.089**	**0.040**	64.6

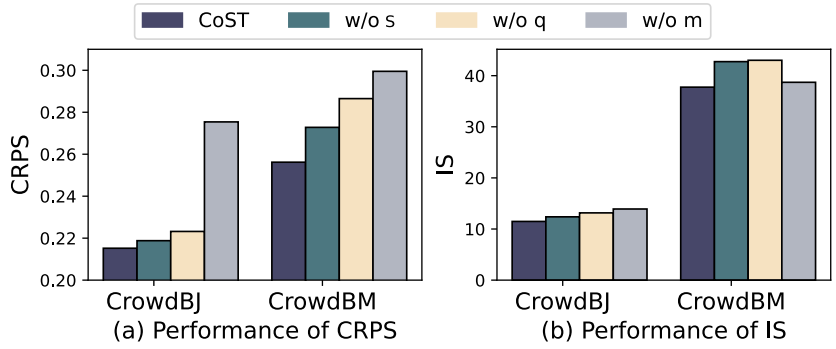

Figure 8: Ablation study on the CrowdBJ and CrowdBM comparing variants in terms of (a) CRPS and (b) IS.

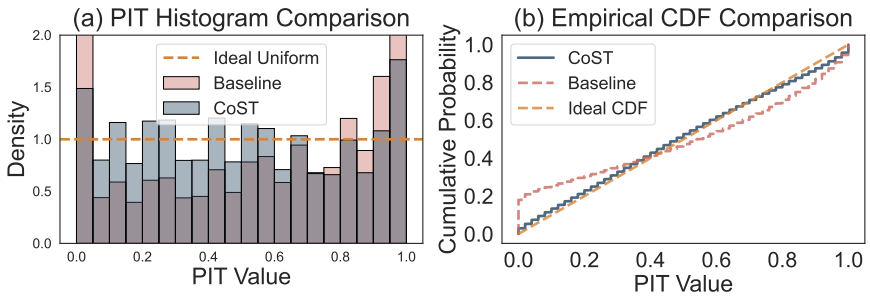

Figure 9: PIT analysis on the MobileSH dataset: (a) PIT histogram and (b) PIT empirical CDF.

Table 10: Comparison of training and inference time on the MobileSH dataset.

Model	Train Time	Inference Time
D3VAE	3min 27s	2min 15s
DiffSTG	24min 16s	18min 38s
TimeGrad	5min	2min
CSDI	48min 40s	38min 49s
DyDiffusion	33h	3h
CoST	2min	50s

Table 9: Long-term forecasting results in terms of MAE and RMSE. **Bold** indicates the best performance, while underlining denotes the second-best. DYffusion is limited to grid-format data, and '-' denotes results that are not applicable.

Model	MobileSH		SST		CrowdBJ		CrowdBM		Los-Speed	
	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
D3VAE	0.207	0.392	2.39	3.13	5.63	11.4	12.4	28.2	9.43	13.3
DiffSTG	0.078	0.125	0.94	1.19	3.04	6.37	7.59	18.8	7.77	14.2
TimeGrad	0.058	0.072	1.30	1.64	3.48	4.83	5.25	**7.40**	18.2	22.3
CSDI	**0.035**	0.057	1.31	1.63	1.99	3.64	**4.64**	12.4	11.3	15.0
NPDiff	0.037	0.057	1.91	2.82	2.06	3.28	5.44	13.8	46.0	58.3
DYffusion	0.047	0.066	**0.85**	**1.06**	-	-	-	-	-	-
CoST	0.035	**0.053**	0.86	1.13	**1.92**	**3.05**	4.74	11.2	**5.94**	**10.8**

C.5.1 Analysis of Distribution Alignment.

Additionally, we present the PIT (Probability Integral Transform) histogram in Figure 9 (a) and the PIT empirical cumulative distribution function (CDF) in Figure 9 (b) to visually reflect the

alignment of the full distribution. Ideally, the true values' quantiles in the predictive distribution should follow a uniform distribution, corresponding to the dashed line in Figure 9 (a). In the case of perfect calibration, the PIT CDF should closely resemble the yellow diagonal line. Clearly, our model outperforms CSDI.

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.

   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 Section 4.1

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: We release all the code and data, as well as instructions for how to replicate the results. See abstract and Section C.3.

   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: We have submitted code and data anonymously as supplementary materials.

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: We provide sufficient information on experimental setting. See Section C.3.

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: We report the the statistical significance of the experiments suitably and correctly. See Section C.3.

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: We provide sufficient information on the computer resources. See Section C.3.

   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: We make sure that the presented research conforms with the NeurIPS Code of Ethics.

   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: We provide thorough discussion about broader impacts of this work. 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: The paper poses no such risks.

    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 assets used in the paper are properly credited. The license and terms of use are explicitly mentioned and 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: [Yes]

    Justification: All new assets introduced in the paper are well documented and we provide the documentation alongside the 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]

We did not use large language models (LLMs) as an important, original, or non-standard component of the core methods in this research

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.