Knowledge Graph Guided Heterogeneity-Informed Diffusion Model for Spatio-Temporal Generation

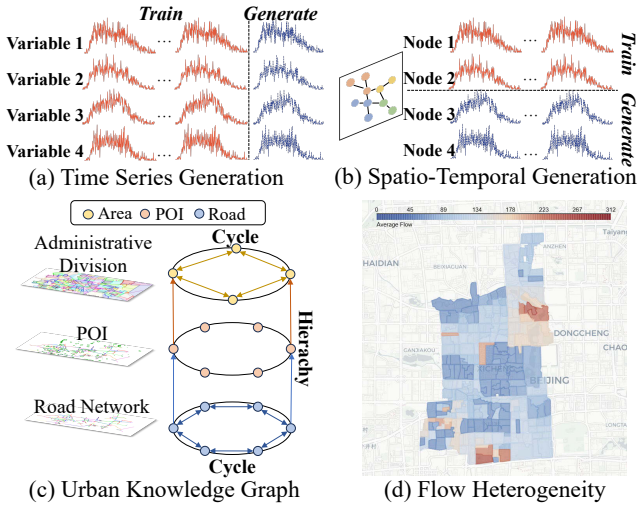
Zi'ang Wang[1], Lei Chen[1,2], Yuanchang Jin[1], Pan Deng[1]*, Shuangshuang Pang[1], Junting Liu[1], Yu Zhao[1]

[1]Beihang University, Beijing, China
[2]China Mobile Group Beijing Co., Ltd., Beijing, China
{wangziang, chenlei0123, 23373279, pandeng, ireland, liujunting, iyzhao}@buaa.edu.cn

Abstract

Spatio-temporal data generation aims to synthesize realistic urban data across graph nodes by learning spatial and temporal dependencies. This task plays a crucial role in urban planning by enabling the simulation of unobserved nodes. However, existing approaches face critical limitations that time series generation methods fail to generalize to unseen nodes, while spatio-temporal generative models are either restricted to the trajectory generation task or dependent on auxiliary data inputs. To bridge these gaps, we propose a Knowledge Graph Guided Heterogeneity-Informed Diffusion Model (KGDiff) in this paper through the following key innovations. First, we design a geometry-aware mixture of experts integrating Euclidean, hyperbolic, and hyperspherical representations to comprehensively encode urban structural knowledge. Next, we present a learnable meta spatio-temporal pattern module that normalizes node-specific heterogeneity before the generation process, and a conditional denoising process that progressively transforms random noise into realistic samples under structural guidance. Finally, extensive experiments across real-world urban datasets demonstrate that KGDiff achieves the state-of-art performance in generating realistic urban spatio-temporal data.

Code — https://github.com/fine8oy/KGDiff



(a) Time Series Generation (b) Spatio-Temporal Generation

(c) Urban Knowledge Graph (d) Flow Heterogeneity

Figure 1: (a) and (b) compare time series generation and spatio-temporal generation tasks. (c) illustrates mixed structures in an urban knowledge graph. (d) illustrates the daily average flow of each area in XC-Traffic.

Introduction

Generative modeling is increasingly vital in various areas such as natural language processing (Gong et al. 2023b,a, 2025) and computer vision (Chang et al. 2022; Huang et al. 2024; Fu et al. 2024). This trend has also drawn significant attention in the spatio-temporal community, where researchers (Wen et al. 2023; Liu et al. 2023a) are actively exploring generative models for spatio-temporal prediction tasks. However, such prediction tasks typically require substantial historical data, making them unsuitable for data-sparse scenarios. The inherent data-generation capacity of deep generative models to model spatio-temporal data is still underexplored. In this work, we focus on the spatio-temporal generation task, where the model is trained on a set of graph nodes and subsequently processes inferences on the newly introduced nodes. This task plays a crucial role in urban planning (Chai, Jiang, and Yu 2024; Yan and Li 2025) by enabling the simulation and evaluation of spatio-temporal patterns, especially for data-sparse and planning cases.

Recent advancements in time series generation (Jeha et al. 2022; Coletta et al. 2023; Zhicheng et al. 2024) aim to synthesize distributionally consistent data for each variable, as shown in Figure 1(a). This paradigm inherently requires generative models to input and fit samples from each individual variable during training. Notably, such approaches typically do not differentiate between training and testing sets, as their ultimate evaluation criterion is the distributional discrepancy between generated and real samples. This formulation presents a fundamental gap with our target spatio-temporal generation task as shown in Figure 1(b), where generalization to unobserved nodes is essential. Meanwhile, in the domain of spatio-temporal data, most existing generation tasks primarily focus on trajectory data (Jiang et al. 2023; Rao et al. 2025; Wang, Lin, and Li 2025) rather than spatio-temporal sequence data. The most relevant work

*Corresponding author.

to our spatio-temporal generation task is KSTDiff (Zhou et al. 2023). However, this approach requires auxiliary socio-economic indicators as additional inputs and is limited to generating flow data, failing to address other critical spatio-temporal data types such as road network speed (Guo et al. 2019; Zhao et al. 2023a).

The formation process of spatio-temporal data is inherently influenced by multiple urban semantics, particularly functional characteristics of each node and the topological information from neighboring nodes. This fundamental mechanism naturally motivates our formulation of the spatio-temporal generation task as a conditional generation problem. To accurately generate data, we identify and aim to tackle the following challenges: (1) **High-Fidelity Spatial Condition**. The generative process necessitates a spatially informed conditioning mechanism that comprehensively encodes topological properties of urban networks. Nonetheless, most approaches derive spatial embeddings directly from adjacency matrices in Euclidean space, and such representations fail to capture the intrinsic higher-order hierarchical structures and cyclic dependencies of real-world urban systems, as shown in Figure 1(c). This limitation critically undermines generation quality, as the output fidelity is fundamentally bounded by the expressiveness of its spatial conditions. (2) **Heterogeneity-Aware Generation**. As shown in Figure 1(d), urban spatio-temporal data exhibits remarkable heterogeneity across nodes, driven by location-specific functional patterns. For instance, residential areas typically manifest two traffic peaks, whereas specialized POIs like hospitals show three peak times including midday activity. These distinct patterns induce significant differences in both flow and road speed across the network, demanding a generation framework that explicitly models these spatio-temporal dependencies.

To address these challenges, in this paper, we propose a **K**nowledge **G**raph Guided Heterogeneity-Informed **Diff**usion Model for spatio-temporal generation named KGDiff. First, to model complex urban spatial information, we construct the urban knowledge graph based on areas, POIs, and roads data following UUKG (Ning et al. 2023), which is an open and unified urban knowledge graph construction approach. As for the spatial condition, we introduce a geometry mixture of experts, which aggregates structural information in the Euclidean, hyperbolic, and hypersphere spaces jointly. In order to solve the second challenge, we design a heterogeneity-informed spatio-temporal diffusion, aiming to train a denoising network to gradually generate the spatio-temporal data from random noise guided by the spatial condition. Moreover, the guidance is mapped to a meta spatio-temporal pattern to mitigate the impact of heterogeneity before input into the diffusion process. Our contributions are summarized as follows:

- **Problem**: Our work extends spatio-temporal generation to a broader range of urban data scenarios beyond flow generation, which is practical and challenging. We focus on addressing two key issues, including the need for spatial semantic conditioning and the challenge of heterogeneity-aware generation.
- **Methodology**: To solve the spatio-temporal generation problem, KGDiff jointly learns urban structural representations in diverse geometric spaces via a geometry mixture-of-experts module, and generates heterogeneous spatio-temporal data through meta pattern normalizing and conditional diffusion.
- **Evaluation**: We conduct extensive experiments on the spatio-temporal generation task on real-world datasets to validate the effectiveness of our proposed approach.

Related Works

Time Series Generation

Deep generative models have demonstrated strong capabilities in high-quality sample generation across various domains, including time series. Among them, Generative Adversarial Networks (GANs) were among the first to be applied to time series generation. TimeGAN (Yoon, Jarrett, and van der Schaar 2019) enhances the modeling of temporal dynamics by incorporating an embedding network and a supervised loss to preserve dependencies. COT-GAN (Xu et al. 2020) combines GAN and Causal Optimal Transfer principles to generate both low- and high-dimensional time series in a stable and efficient manner. Variational Autoencoders (VAEs) have also been explored, such as TimeVAE (Desai et al. 2021), which introduces an interpretable temporal latent structure and achieves competitive performance.

More recently, motivated by the success of diffusion models in modeling complex high-dimensional distributions, various works have adapted diffusion-based approaches for time series generation. DiffWave (Kong et al. 2021) is an early effort that generates realistic audio waveforms from noise using a diffusion process. DiffTime (Coletta et al. 2023) leverages recent developments in score-based diffusion models to capture temporal dynamics in time series. DiffusionTS (Yuan and Qiao 2024) further advances this line of work by integrating a Transformer architecture with disentangled temporal representations to improve generation quality.

Spatio-Temporal Generation

In the context of spatio-temporal generation, the majority of existing work has focused on trajectory generation. DiffTraj (Zhu et al. 2023) introduces an unconditional diffusion-based model for generating GPS trajectories. Diff-RNTraj (Wei et al. 2024) further incorporates road network constraints to improve the realism and feasibility of generated trajectories. Beyond trajectory data, spatio-temporal series data have also attracted attention within the community, such as LargeST (Liu et al. 2023b) and XC-Traffic (Zhao et al. 2023b). However, to the best of our knowledge, KSTDiff (Zhou et al. 2023) is the only method that explicitly addresses spatio-temporal series generation. Nevertheless, its reliance on auxiliary social information and its focus on regional flow data limit its general applicability to broader spatio-temporal scenarios. In contrast, we focus on the generation of spatio-temporal series at multiple granularities, including road-level traffic, sensor-level speeds, and

region-level flows, aiming for a more generalizable modeling framework.

Preliminaries

Diffusion Model

Diffusion models are deep generative models that learn a data distribution of variable $\boldsymbol{X}$ by a forward noising process and a reverse denoising process. The forward process transforms data $\boldsymbol{X}_0 \sim q(\boldsymbol{X}_0)$ into a sequence $(\boldsymbol{X}_1, \boldsymbol{X}_2, ..., \boldsymbol{X}_\tau)$ by adding Gaussian noise at each step as

$$q(\boldsymbol{X}_{1:\tau}|\boldsymbol{X}_0) = \prod_{i=1}^{\tau} \mathcal{N}(\sqrt{1-\beta_i}\boldsymbol{X}_{i-1}, \beta_i \boldsymbol{I}), \tag{1}$$

where $\beta_i \in (0, 1)$ controls the noise level. This leads to a closed-form expression as

$$q(\boldsymbol{X}_i|\boldsymbol{X}_0) = \mathcal{N}(\boldsymbol{X}_i; \sqrt{\bar{\alpha}_i}\boldsymbol{X}_0, (1-\bar{\alpha}_i)\boldsymbol{I}), \tag{2}$$

where $\alpha_i = 1 - \beta_i$ and $\bar{\alpha}_i = \prod_{i=1}^{\tau}(1-\beta_i)$, can also be reparameterized as

$$\boldsymbol{X}_i = \sqrt{\bar{\alpha}_i}\boldsymbol{X}_0 + \sqrt{1-\bar{\alpha}_i}\boldsymbol{\epsilon}, \tag{3}$$

where $\boldsymbol{\epsilon}$ is sampled from standard Gaussian noise $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. The reverse process starts from pure Gaussian noise $\boldsymbol{X}_\tau \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ and aims to recover $\boldsymbol{X}_0$ through a learnable denoising network $\boldsymbol{\epsilon}_\theta$ as

$$\begin{aligned} p_\theta(\boldsymbol{X}_{0:\tau}) &= p(\boldsymbol{X}_i)\prod_{i=1}^{\tau} p_\theta(\boldsymbol{X}_{i-1}|\boldsymbol{X}_i), \\ p_\theta(\boldsymbol{X}_{i-1}|\boldsymbol{X}_i) &= \mathcal{N}(\boldsymbol{X}_{i-1}; \mu_\theta(\boldsymbol{X}_i, i), \sigma_\theta(\boldsymbol{X}_i, i)\boldsymbol{I}), \end{aligned} \tag{4}$$

where μ_θ and σ_θ can be parameterized using the predicted noise as

$$\begin{aligned} \mu_\theta(\boldsymbol{X}_i, i) &= \frac{1}{\sqrt{\bar{\alpha}_i}}(\boldsymbol{X}_i - \frac{\beta_i}{\sqrt{1-\bar{\alpha}_i}}\boldsymbol{\epsilon}_\theta(\boldsymbol{X}_i, i)), \\ \sigma_\theta(\boldsymbol{X}_i, \boldsymbol{X}_0) &= \frac{1-\bar{\alpha}_{i-1}}{1-\bar{\alpha}_i}\beta_i. \end{aligned} \tag{5}$$

The denoising network $\boldsymbol{\epsilon}_\theta$ can be trained to minimize the prediction error as

$$\mathcal{L}(\theta) = \mathbb{E}_{\boldsymbol{X}_0 \sim q(\boldsymbol{X}_0), \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), i} \big|\big| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\boldsymbol{X}_i, i) \big|\big|. \tag{6}$$

Geometry Space

Geometry spaces are distinguished by the curvature value. Specifically, the curvature c is negative for hyperbolic space $\mathbb{H}$, positive for hypersphere space $\mathbb{S}$, and zero for Euclidean space $\mathbb{E}$. Poincaré ball provides closed-form expressions for many basic objects such as distance and angle. Here we only introduce the operation of hyperbolic space, and the operation of hypersphere space can be obtained by analogy.

For each point $\boldsymbol{x}$ in the hyperbolic space, we have a tangent space $\mathcal{T}_{\boldsymbol{x}}\mathbb{H}_c^n$. We can use the exponential map $\exp_{\boldsymbol{x}}^c : \mathcal{T}_{\boldsymbol{x}}\mathbb{H}_c^n \to \mathbb{H}_c^n$ and the logarithmic map $\log_{\boldsymbol{x}}^c : \mathbb{H}_c^n \to \mathcal{T}_{\boldsymbol{x}}\mathbb{H}_c^n$ to establish the connection between the hyperbolic space and the tangent which can be formulated as

$$\begin{aligned} \exp_{\boldsymbol{x}}^c(\boldsymbol{v}) &= \boldsymbol{x} \oplus_c \left(\tanh\left(\sqrt{c}\frac{\lambda_{\boldsymbol{x}}||\boldsymbol{v}||}{2}\right)\frac{\boldsymbol{v}}{\sqrt{c}||\boldsymbol{v}||}\right), \\ \log_{\boldsymbol{x}}^c(\boldsymbol{y}) &= \frac{2}{\sqrt{c}\lambda_{\boldsymbol{x}}}\tanh^{-1}\big(\sqrt{c}||-\boldsymbol{x} \oplus_c \boldsymbol{y}||\big)\frac{-\boldsymbol{x} \oplus_c \boldsymbol{y}}{||-\boldsymbol{x} \oplus_c \boldsymbol{y}||}, \end{aligned} \tag{7}$$

where $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{H}_c^n$, $v \in \mathcal{T}_{\boldsymbol{x}}\mathbb{H}_c^n$, $||\cdot||$ denotes the Euclidean norm and $\oplus_c$ represents Möbius addition as

$$\boldsymbol{x} \oplus_c \boldsymbol{y} = \frac{\big(1 + 2c\langle \boldsymbol{x}, \boldsymbol{y}\rangle + c||\boldsymbol{y}||^2\big)\boldsymbol{x} + \big(1 - c||\boldsymbol{x}||^2\big)\boldsymbol{y}}{1 + 2c\langle \boldsymbol{x}, \boldsymbol{y}\rangle + c^2||\boldsymbol{x}||^2||\boldsymbol{y}||^2}. \tag{8}$$

Similar to the scalar multiplication in Euclidean space, the multiplication in hyperbolic space can be defined by Möbius scalar multiplication between vectors $\boldsymbol{x} \in \mathbb{H}_c^n\{\boldsymbol{0}\}$:

$$\boldsymbol{r} \otimes_c \boldsymbol{x} = \frac{1}{\sqrt{c}}\tanh\big(\boldsymbol{r}\tanh^{-1}(\sqrt{c}||\boldsymbol{x}||)\big)\frac{\boldsymbol{x}}{||\boldsymbol{x}||}, \tag{9}$$

where $\boldsymbol{r} \in \mathbb{R}$.

Methodology

In this section, we present the proposed Knowledge Graph Guided Diffusion Heterogeneity-Informed Model (KGDiff) for spatio-temporal generation. Figure 2 provides an overview of the architecture.

Problem Formulation

Given a graph structure $\mathcal{G} = (\mathcal{E}_{ST}, \mathcal{R}_{ST})$, the node set $\mathcal{E}_{ST}$ contains variables in spatio-temporal data, and the edge set $\mathcal{R}_{ST}$ contains the physical connection of variables. We define spatio-temporal data as $\boldsymbol{X}_{1:T} = (x_1, ..., x_T) \in \mathbb{R}^{N \times T \times D}$, where N, T, and D are the number of nodes, time steps, and features, respectively. Assuming that a dataset $\mathbb{D}$ containing m samples can be formulated as $\mathbb{D} = \{\boldsymbol{X}_{1:T}^i\}_{i=1}^m$, the goal of spatio-temporal generation is to use a model f guided by a graph-based condition $\phi(\mathcal{G})$ to generate spatio-temporal data with the same distribution as $\mathbb{D}$, which can be formulated as

$$\hat{\boldsymbol{X}}_{1:T}^i = f(Z, \boldsymbol{\phi}), \tag{10}$$

where Z is the input sampled from the Gaussian distribution.

Urban Knowledge Graph Guidance

To better capture the environmental context of nodes in spatio-temporal graph structures, we construct a knowledge graph following the UUKG (Ning et al. 2023), which is a unified and publicly available urban knowledge graph construction benchmark. The urban knowledge graph is defined as a multi-relational graph:

$$\begin{aligned} &\mathcal{G}_{\text{urban}} = (\mathcal{E}, \mathcal{R}, \mathcal{F}), \\ &\mathcal{F} = \{(h, r, t)|h, t \in \mathcal{E}, r \in \mathcal{R}\}, \end{aligned} \tag{11}$$

where $\mathcal{E}$, $\mathcal{R}$ and $\mathcal{F}$ are the sets of entities, relations, and facts, respectively, and $\mathcal{E}_{ST} \subseteq \mathcal{E}$. The core set $\mathcal{F}$ describes triplets of head entity h, relation r and tail entity t. In this way, the urban knowledge graph provides a comprehensive representation of diverse urban environmental semantics, including multi-scale spatial adjacencies and key POIs that critically influence spatio-temporal data generation patterns.

To effectively harness the structured environmental knowledge encoded in our urban knowledge graph, we employ knowledge graph embedding approach to learn representations for each entity $\mathcal{E}$, especially the nodes $\mathcal{E}_{ST}$ in the graph $\mathcal{G}$. Specifically, we adopt the TuckER (Balazevic,

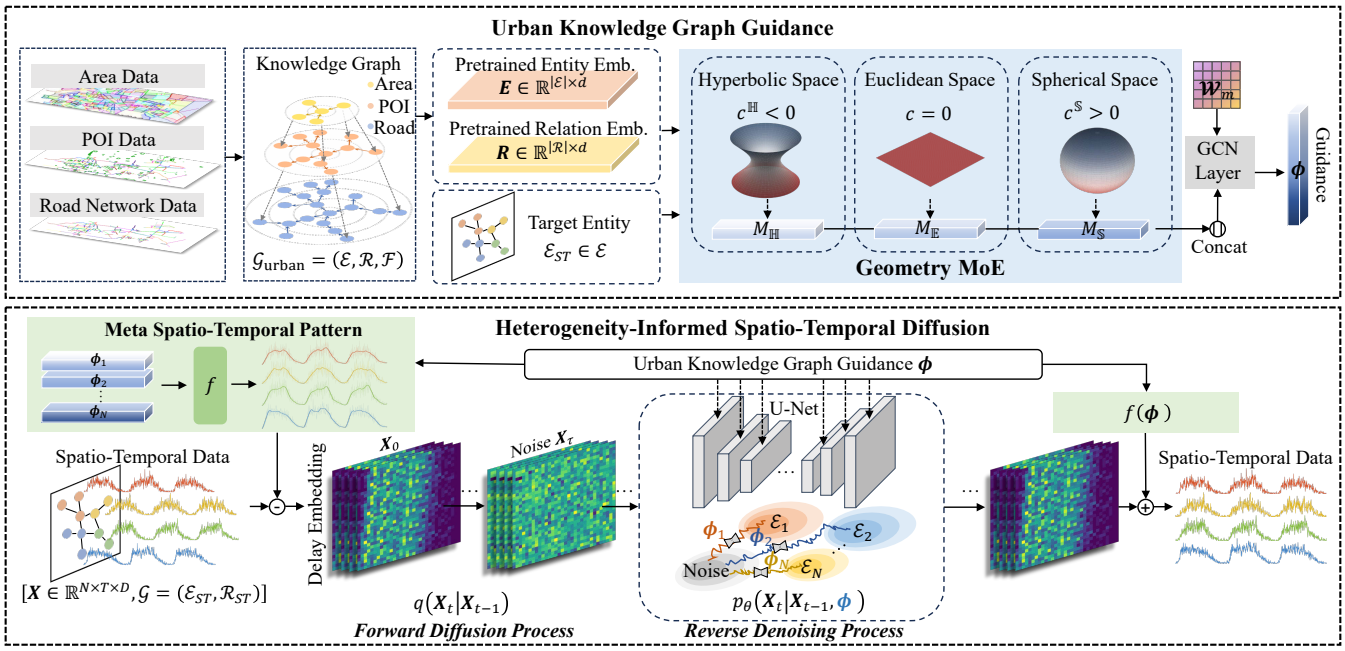


Figure 2: The architecture of the proposed KGDiff.

Allen, and Hospedales 2019) for pretraining entity and relation embeddings, selected for its demonstrated effectiveness in capturing complex relational patterns. The TuckER computes triplet plausibility scores through the formulation as

$$\psi(h, r, t) = \boldsymbol{\mathcal{W}} \times_1 \boldsymbol{e}_h \times_2 \boldsymbol{e}_r \times_3 \boldsymbol{e}_t, \tag{12}$$

where $\boldsymbol{\mathcal{W}} \in \mathbb{R}^{d\times d\times d}$ is a learnable tensor, $\times_n$ is the tensor product alone the n-th dimension, and $\boldsymbol{e}_h, \boldsymbol{e}_r, \boldsymbol{e}_t \in \mathbb{R}^d$ are the embeddings of head entity h, tail entity t, and relation r, respectively. While standard knowledge graph embedding models such as TuckER have demonstrated strong performance in link prediction tasks, their pretrained representations are not directly suitable for spatio-temporal generation. An additional task-specific fine-tune is necessary to adapt the pretrained embeddings for the generative task.

Geometry MoE

To bridge this gap, we introduce an embedding fine-tune module, Geometry Mixture-of-Experts. Given the pretrained entity embedding matrix $\boldsymbol{E} = \{\boldsymbol{e}_1, \boldsymbol{e}_2, ..., \boldsymbol{e}_{|\mathcal{E}|}\} \in \mathbb{R}^{|\mathcal{E}|\times d}$ and relation embedding matrix $\boldsymbol{R} = \{\boldsymbol{r}_1, \boldsymbol{r}_2, ..., \boldsymbol{r}_{|\mathcal{R}|}\} \in \mathbb{R}^{|\mathcal{R}|\times d}$, the module further refines node representations in the spatio-temporal graph. Graph neural networks update the embeddings of entities by aggregating the information from their neighbors based on $\boldsymbol{rt}$, where $\boldsymbol{t}$ denotes the tail entity embedding. However, this operation is usually obtained in the Euclidean space, which leads to insufficient neighbor information aggregation. In practice, the urban knowledge graph contains rich structural information such as high-order hierarchy and high-order cycle. Therefore, we integrate spatial information in diverse geometric spaces, including hyperbolic, Euclidean, and hypersphere spaces.

Specifically, given a central entity h_γ and its neighbor set $\mathcal{N}_\gamma = \{(r_\delta, t_\delta) \mid (h_\gamma, r_\delta, t_\delta) \in \mathcal{F}\}$, which denotes all neighbors of h_γ, the experts in the three geometric spaces can be defined as

$$\begin{aligned} M_\delta^{\mathbb{E}} &= \boldsymbol{r}_\delta \boldsymbol{t}_\delta, \\ M_\delta^{\mathbb{H}} &= \boldsymbol{r}_\delta \otimes_{c_1} \exp_{\boldsymbol{0}}^{c_1}(\boldsymbol{t}_\delta), \\ M_\delta^{\mathbb{S}} &= \boldsymbol{r}_\delta \otimes_{c_2} \exp_{\boldsymbol{0}}^{c_2}(\boldsymbol{t}_\delta), \end{aligned} \tag{13}$$

where $\boldsymbol{r}_\delta$ and $\boldsymbol{t}_\delta$ represent the embeddings of relation r_δ and tail entity t_δ in Euclidean space $\mathbb{E}$, $c_1 < 0$ and $c_2 > 0$ are two trainable curvatures for hyperbolic space $\mathbb{H}$ and hypersphere space $\mathbb{S}$. We combine $M_\delta^{\mathbb{E}}$, $M_\delta^{\mathbb{H}}$, and $M_\delta^{\mathbb{S}}$ from diverse geometry spaces and construct geometry mixture of experts as

$$M(r_\delta, t_\delta) = \boldsymbol{\mathcal{W}}_m \left[M_\delta^{\mathbb{E}} \,||\, \log_{\boldsymbol{0}}^{c_1}(M_\delta^{\mathbb{H}}) \,||\, \log_{\boldsymbol{0}}^{c_2}(M_\delta^{\mathbb{S}}) \right], \tag{14}$$

where $\boldsymbol{\mathcal{W}}_m \in \mathbb{R}^{d\times 3d}$ is a trainable transformation matrix, $||$ represents the concatenation of embeddings. $M(r_\delta, t_\delta)$ is still in Euclidean space to enable compatibility with the diffusion model, thereby allowing the spatial semantics to effectively condition the spatio-temporal generation. Then, the final embedding $\boldsymbol{\phi}_\gamma$ aggregated by a GCN layer can be defined as

$$\boldsymbol{\phi}_\gamma = \sigma\left(\frac{1}{|\mathcal{N}_\gamma|} \sum_{\delta \in \mathcal{N}_\gamma} M(r_\delta, t_\delta) \right), \tag{15}$$

where $\sigma(\cdot)$ is an activation function. We subsequently employ embeddings as the spatial-semantic condition to guide the diffusion process.

Meta Spatio-Temporal Pattern

The temporal pattern of each node is closely correlated with its spatial structural characteristic. Therefore, we first lever-

age the spatial guidance $\boldsymbol{\phi}$ to normalize heterogeneous temporal patterns before initiating the diffusion process. Specifically, we estimate a meta spatio-temporal pattern $f(\boldsymbol{\phi})$ for each node, which is trained together with the diffusion model. Thus, the diffusion process is learnable and customized based on the spatial information.

The meta spatio-temporal pattern is defined as a stable trend inherent to a given node, with distinct variations across different nodes. Spatial distribution disparities lead to significant differences in the mean value of each node. However, the denoising process of diffusion models initiates from Gaussian noise sampled from $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$. Consequently, heterogeneity substantially influences the diffusion model, necessitating the elimination of such heterogeneity prior to the diffusion process.

We use spatial guidance $\boldsymbol{\phi} \in \mathbb{R}^{N\times d}$ from urban knowledge graph to map the stable meta spatio-temporal pattern $f(\boldsymbol{\phi})$ for each node. If the node feature dimension $D > 1$ (e.g., traffic flow containing both inflow and outflow), a distinct meta pattern is mapped for each individual feature. The resulting tensors are then concatenated along the feature dimension, i.e., $f(\boldsymbol{\phi}) \in \mathbb{R}^{N\times T\times D}$. We then subtract the meta pattern from the original data, using the resulting heterogeneity-eliminated data as the starting point for the diffusion process. The forward diffusion process in Equation 2 is modified as

$$q(\boldsymbol{X}_i|\boldsymbol{X}_0, f(\boldsymbol{\phi})) = \mathcal{N}(\boldsymbol{X}_i; \sqrt{\alpha_i}\boldsymbol{X}_0 + (1-\sqrt{\alpha_i})f(\boldsymbol{\phi}), (1-\alpha_i)\boldsymbol{I}). \tag{16}$$

In the backward denoising process, the posterior is formulated as

$$q(\boldsymbol{X}_{i-1}|\boldsymbol{X}_i, \boldsymbol{X}_0, f(\boldsymbol{\phi})) = \mathcal{N}(\boldsymbol{X}_{i-1}; \tilde{\mu}(\boldsymbol{X}_i, \boldsymbol{X}_0, f(\boldsymbol{\phi})), \tilde{\beta}_i\boldsymbol{I}), \tag{17}$$

where

$$\begin{aligned}\tilde{\mu}(\boldsymbol{X}_i, \boldsymbol{X}_0, f(\boldsymbol{\phi})) &= \frac{\sqrt{\bar{\alpha}_{i-1}}\beta_i}{1-\bar{\alpha}_i}\boldsymbol{X}_0 + \frac{\sqrt{\alpha_i}(1-\bar{\alpha}_i)}{1-\bar{\alpha}_i}\boldsymbol{X}_i \\ &+\left(1+\frac{(\sqrt{\bar{\alpha}_i}-1)(\sqrt{\alpha_i}+\sqrt{\bar{\alpha}_{i-1}})}{1-\bar{\alpha}_i}\right)f(\boldsymbol{\phi}), \\ \tilde{\beta}_i &= \frac{1-\bar{\alpha}_{i-1}}{1-\bar{\alpha}_i}\beta_i.\end{aligned}$$

Heterogeneity-Informed Diffusion

After addressing spatial heterogeneity, we focus on modeling the temporal dimension for each individual sample during the diffusion process. To comprehensively capture the temporal characteristics of each individual sample, we employ delay embedding to transform every sample $\boldsymbol{X}_{1:T}$ into a two-dimensional matrix composed of sliding window segments. Let U, V be two parameters denoting the stride value and window size of the sliding window. We transform each sample $\boldsymbol{X}_{1:T} = (x_1, ..., x_T)$ to the matrix $\boldsymbol{X}$ as

$$\boldsymbol{X} = \begin{bmatrix} x_1 & x_{1+U} & \cdots & x_{T-V+1} \\ x_2 & x_{2+U} & \cdots & x_{T-V+2} \\ \vdots & \vdots & \ddots & \vdots \\ x_V & x_{V+U} & \cdots & x_T \end{bmatrix} \in \mathbb{R}^{N\times V\times W\times D}, \tag{18}$$

Datasets	Nodes	Time Step	Time Interval
XC-Traffic	235	17,520	30min
NYC-Bike	99	17,568	30min
Tongzhou	645	25,056	5min
PeMS-Bay	325	51,840	5min

Table 1: Detailed Datasets Attribute Information

where $W = [(T-V)/U]$. To ensure dimensional consistency between W and V for neural network input compatibility (i.e., $\boldsymbol{X} \in \mathbb{R}^{N\times V\times V\times D}$), zero padding is applied when necessary.

As for the denoising network, we adopt the EDM (Karras et al. 2022) as the backbone, which is widely used due to its balance between rapid sampling and high-quality generations. It takes noised data $\boldsymbol{X}_i \in \mathbb{R}^{N\times V\times V\times D}$, diffusion step i and urban knowledge graph guidance $\boldsymbol{\phi}$ as input, and outputs the predicted noise as

$$\hat{\boldsymbol{\epsilon}} = \boldsymbol{\epsilon}_\theta(\boldsymbol{X}_i, i, \boldsymbol{\phi}), \tag{19}$$

where $\hat{\boldsymbol{\epsilon}} \in \mathbb{R}^{N\times V\times V\times D}$ is of the same shape as $\boldsymbol{X}_0$. Specifically, we sequentially inject both the guidance $\boldsymbol{\phi}$ and diffusion step embedding into each UNet block, where they are incorporated into the $\boldsymbol{X}_i$ through separate addition operations.

Experiments

Experiment Settings

Datasets The experiments are conducted on 4 real-world datasets for spatio-temporal generation: XC-Traffic (Zhao et al. 2023b), NYC-Bike, Tongzhou, and PeMS-Bay (Li et al. 2018) datasets. XC-Traffic and NYC-Bike contain traffic flow in areas recorded at 30-minute intervals. Tongzhou and PeMS-Bay contain speed for road segments and highway sensors, respectively, recorded at 5-minute intervals. For Tongzhou dataset focusing on road network speeds, we construct a comprehensive knowledge graph incorporating area, POI, road, and corresponding categories entities. For area flow datasets XC-Traffic and NYC-Bike, we simplify the graph structure by removing road entities. Regarding sensor data of PeMS-BAY lacking physical adjacency, we only derive sensor connectivity through the distance.

Baselines We compare our generation results with several representative state-of-art generation methods, including (1) Unconditional generation models: TimeGAN (Yoon, Jarrett, and van der Schaar 2019), TimeVAE (Desai et al. 2021), DiffWave (Kong et al. 2021), DiffTime (Coletta et al. 2023), DiffusionTS (Yuan and Qiao 2024), ImagenTime (Naiman et al. 2024), and SDformer (Zhicheng et al. 2024); (2) Conditional generation models: DiffWave-con, DiffTime-con, KSTDiff (Zhou et al. 2023), DiffusionTS-con, and ImagenTime-con.

Implementation Details In our experiments, the train nodes and test nodes are partitioned based on their latitude values. We use each 24-hour period (48 steps for XC-Traffic

Methods	*XC-Traffic*				*NYC-Bike*				*Tongzhou*				*PeMS-Bay*			
	MAE	RMSE	MMD	JSD	MAE	RMSE	MMD	JSD	MAE	RMSE	MMD	JSD	MAE	RMSE	MMD	JSD
TimeGAN	59.58	76.02	1.82	0.63	5.22	7.55	1.35	0.68	14.05	14.18	2.60	0.83	26.99	27.58	5.74	0.83
TimeVAE	21.10	29.18	0.13	0.39	4.27	5.48	0.19	0.57	20.48	20.71	3.09	0.50	55.87	56.08	6.82	0.63
DiffWave	17.97	24.96	0.52	0.49	5.14	5.87	0.80	0.63	25.23	25.32	3.47	0.37	4.70	5.66	1.50	0.60
DiffTime	48.09	54.54	0.98	0.51	7.20	8.73	0.96	0.63	15.03	15.92	1.80	0.74	4.88	5.73	0.68	0.44
DiffusionTS	30.06	39.43	0.55	0.53	4.29	6.15	0.73	0.62	14.50	15.08	1.01	0.82	4.71	6.74	1.56	0.40
ImagenTime	62.24	78.96	1.98	0.45	8.38	10.60	1.40	0.31	24.64	24.67	4.17	0.29	10.30	10.85	1.47	0.26
SDformer	9.35	13.66	1.99	0.25	3.73	4.69	1.48	0.16	12.39	16.46	4.01	0.50	2.00	3.43	2.67	0.31
DiffWave-con	17.29	25.95	0.45	0.51	3.85	4.78	0.58	0.59	29.24	29.39	3.93	0.43	4.66	5.59	1.34	0.57
DiffTime-con	48.21	54.39	0.97	0.52	7.19	8.72	0.96	0.58	20.55	21.20	1.98	0.73	4.77	5.86	0.65	0.51
KSTDiff	53.99	97.09	1.88	0.55	5.97	9.61	0.70	0.31	11.49	14.93	2.72	0.28	4.80	8.20	1.57	0.28
DiffusionTS-con	32.73	43.02	0.35	0.61	4.58	6.43	0.76	0.61	11.84	12.21	0.82	0.81	5.64	6.63	1.20	0.56
ImagenTime-con	63.45	80.23	2.04	0.40	8.23	10.41	1.36	0.40	26.44	26.47	4.45	0.41	10.29	10.98	1.28	0.31
KGDiff	**7.66**	**11.03**	**0.04**	**0.24**	**2.40**	**3.11**	**0.09**	**0.15**	**5.52**	**5.80**	**0.24**	**0.20**	**1.53**	**2.33**	**0.15**	**0.13**

Table 2: Performance comparison with baselines on four datasets. Best results are presented in bold, and the second best results are underlined.

and NYC-Bike, while 288 steps for Tongzhou and PeMS-Bay) on each weekday as a sample. We set $U = 3, V = 16$ for XC-Traffic and NYC-Bike, and $U = 9, V = 32$ for Tongzhou and PeMS-Bay. We set diffusion steps $\tau = 200$, and the dimension of knowledge graph embeddings $d = 128$. All experiments are repeated three times, conducted on NVIDIA Tesla V100 GPUs.

Evaluation Metrics We generate the same number of samples as real data for the test nodes, and use the mean of all samples for each node to calculate Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). In addition, we use Maximum Mean Discrepancy (MMD) and Jensen-Shannon Divergence (JSD) to measure the distance between generated data distribution and real data distribution.

Main Results

The experimental results in Table 2 show that KGDiff consistently outperforms baseline methods, with an error reduction of 18.07% to 51.96% compared to the best baseline in terms of MAE. The significant improvements in metrics demonstrate that our method accurately generates spatio-temporal data under the guidance of spatial information.

First, the results reveal that SDformer achieves relatively strong performance among the unconditional generation methods. This demonstrates SDformer's capability to accurately capture the common patterns in the dataset when operating without conditional guidance, which is attributed to its next-token prediction design. However, such unconditional generation approaches fundamentally lack the capacity to model spatial dependencies, consequently failing to perform node-specific personalized generation. Our proposed KGDiff, in contrast, effectively models the inherent heterogeneity in spatio-temporal data and achieves accurate generation under the guidance of node embeddings.

Second, the performance of conditional generation models varies significantly across different dataset. KSTDiff demonstrates superior results on speed-related data compared to flow-related data, indicating that its reliance on additional data inputs (e.g., socio-economic indicators) in the flow estimator adversely affects its performance. Conversely, DiffWave shows weaker performance on road network data with larger node counts and more complex patterns, revealing the fundamental limitations of such conditional diffusion approaches in addressing spatio-temporal generation tasks. In contrast, our KGDiff consistently outperforms all baselines across datasets, demonstrating its robust adaptability to diverse real-world scenarios.

Methods	*XC-Traffic*				*Tongzhou*			
	MAE	RMSE	MMD	JSD	MAE	RMSE	MMD	JSD
w/o Geometry MoE	8.20	12.27	0.08	0.35	8.41	8.98	0.52	0.25
w/o $\mathbb{H}$	7.81	11.50	0.06	0.25	8.28	8.43	0.45	0.32
w/o $\mathbb{E}$	7.97	11.81	0.06	0.27	8.32	8.47	0.45	0.30
w/o $\mathbb{S}$	8.07	11.92	0.06	0.32	8.32	8.47	0.45	0.26
w/o Meta Pattern	8.30	12.14	0.12	0.29	8.62	8.77	0.49	0.27
w/o Guidance	8.43	12.19	0.13	0.29	13.05	15.86	2.33	0.37
KGDiff	**7.66**	**11.03**	**0.04**	**0.24**	**5.52**	**5.80**	**0.24**	**0.20**

Table 3: Ablation Study.

Ablation Studies

Discussion on each proposed module. We systematically remove specific modules from our KGDiff to evaluate the contribution of each carefully designed component and learning strategy. We design the following variants: (1)***w/o* Geometry MoE** represents that we remove the Geometry Mixture of Experts and directly use the entity embedding for guidance; (2) ***w/o*** $\mathbb{H}$**,** ***w/o*** $\mathbb{E}$ **and** ***w/o*** $\mathbb{S}$ represent that we remove the experts in hyperbolic, Euclidean, and hypersphere space, respectively; (3) ***w/o* Meta Pattern** represents that we remove the Meta Spatio-Temporal Pattern; (4) ***w/o* Guidance** represents that we remove the Urban Knowledge Graph Guidance, which corresponds to an unconditional generation setting.

The results shown in Table 3 validate the necessity of each proposed module. *w/o* $\mathbb{H}$ performs consistently worse than *w/o* $\mathbb{E}$ and *w/o* $\mathbb{S}$, reflecting the intrinsic hierarchical structures prevalent in fine-grained urban topology. The significant performance deterioration observed in both *w/o* Meta Pattern and *w/o* Guidance confirms their essential role in guiding accurate conditional generation.

Methods	MAE	RMSE	MMD	JSD
DiffWave	4.66	5.59	1.34	0.57
DiffWave+Geometry-MoE	**4.62**	**5.58**	**1.25**	**0.56**
DiffTime	4.77	5.86	0.65	0.51
DiffTime+Geometry-MoE	**4.75**	**5.72**	**0.60**	**0.49**
DiffusionTS	5.64	6.63	1.20	0.56
DiffusionTS+Geometry-MoE	**3.67**	**5.61**	**1.19**	**0.44**

Table 4: We integrate our Geometry MoE into three baseline methods of DiffWave, DiffTime, and DiffusionTS.

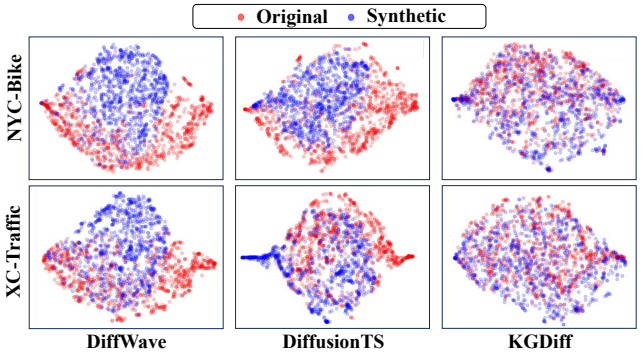


Figure 3: We plot the t-SNE of synthetic data (blue) generated with our method and SOTA methods in comparison with the real data (red).

Discussion on Geometry MoE. To evaluate the guidance capability of knowledge graph embeddings produced by our Geometry MoE, we integrate this module into three baseline diffusion models: DiffWave (Kong et al. 2021), DiffTime (Coletta et al. 2023), and DiffusionTS (Yuan and Qiao 2024). We evaluate these architectural variants for conditional spatio-temporal generation on the PeMS-Bay dataset, comparing their performance against the baseline approach using standard entity embeddings, as shown in Table 4. The incorporation of Geometry MoE produces statistically significant performance gains across all evaluated models. DiffWave and DiffTime achieve MMD reductions of 6.71% and 7.69%, respectively, while DiffusionTS achieves the most substantial MAE decrease of 34.93%. The results indicate that our spatial conditioning framework successfully transfers to various diffusion models, and the spatial geometry representations consistently improve spatial dependency modeling regardless of the base architecture.

Qualitative Results

Furthermore, we conduct comparative visualizations of generated versus real data distributions on both NYC-Bike and XC-Traffic datasets. We employ t-SNE (van der Maaten and Hinton 2008) to project the high-dimensional data into a 2D space as shown in Figure 3, followed by Kernel Density Estimation (KDE) for probability distribution visualization as show in Figure 4. Comparative analysis against DiffWave and DiffusionTS demonstrates that KGDiff generates spatio-temporal data with significantly closer distributional alignment to ground truth. While baseline methods fail to generate node-specific data under spatial guidance, our method shows superior spatial comprehension, producing distributions that better align with spatial-semantic characteristics.

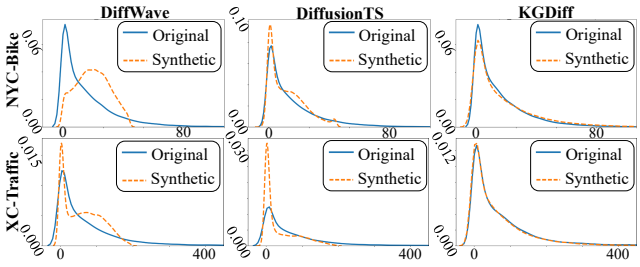


Figure 4: We compare the probability density functions between generated data and real data.

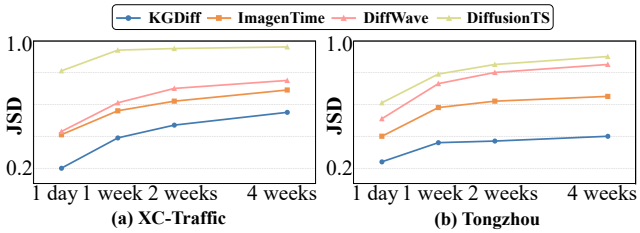


Figure 5: Performance comparison for long-term spatio-temporal generation.

Case Study

Real-world applications frequently encounter challenges in long-term spatio-temporal modeling. To systematically evaluate our method's capability in extended temporal horizons, we conduct experiments with progressively longer sequences on both XC-Traffic and Tongzhou datasets. Figure 5 presents the comparative JSD results among KGDiff, DiffWave, and DiffusionTS. The consistent superiority of our approach across all temporal scales demonstrates its robust capacity for modeling spatio-temporal dependencies at varying time horizons, while baseline methods exhibit significant performance degradation as sequence length increases.

Conclusion

In this work, we present KGDiff, a novel knowledge graph guided heterogeneity-informed diffusion framework for urban spatio-temporal generation tasks critical to urban planning. For the challenge of spatial-semantic conditioning, we design a Geometry Mixture-of-Experts that integrates multi-geometric representations to capture complex urban structural patterns. To address the problem of spatio-temporal heterogeneity, we introduce meta pattern extraction that normalizes node-specific characteristics by changing the denoising starting point to the stable temporal trend for each node. In addition, extensive experiments across four urban datasets demonstrate KGDiff's superior performance in quantitative metrics while qualitatively generating more spatially coherent patterns than existing methods.

Acknowledgments

This work is mainly supported by Beijing Municipal Natural Science Foundation (No.QY24128).

References

Balazevic, I.; Allen, C.; and Hospedales, T. 2019. TuckER: Tensor Factorization for Knowledge Graph Completion. In Inui, K.; Jiang, J.; Ng, V.; and Wan, X., eds., *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 5185–5194. Hong Kong, China: Association for Computational Linguistics.

Chai, H.; Jiang, T.; and Yu, L. 2024. 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*, KDD '24, 4828–4838. New York, NY, USA: Association for Computing Machinery.

Chang, H.; Zhang, H.; Jiang, L.; Liu, C.; and Freeman, W. T. 2022. Maskgit: Masked generative image transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11315–11325.

Coletta, A.; Gopalakrishnan, S.; Borrajo, D.; and Vyetrenko, S. 2023. On the Constrained Time-Series Generation Problem. In Oh, A.; Naumann, T.; Globerson, A.; Saenko, K.; Hardt, M.; and Levine, S., eds., *Advances in Neural Information Processing Systems*, volume 36, 61048–61059. Curran Associates, Inc.

Desai, A.; Freeman, C.; Wang, Z.; and Beaver, I. 2021. Timevae: A variational auto-encoder for multivariate time series generation. *arXiv preprint arXiv:2111.08095*.

Fu, T.-J.; Hu, W.; Du, X.; Wang, W. Y.; Yang, Y.; and Gan, Z. 2024. Guiding Instruction-based Image Editing via Multimodal Large Language Models. In *The Twelfth International Conference on Learning Representations*.

Gong, S.; Agarwal, S.; Zhang, Y.; Ye, J.; Zheng, L.; Li, M.; An, C.; Zhao, P.; Bi, W.; Han, J.; Peng, H.; and Kong, L. 2025. Scaling Diffusion Language Models via Adaptation from Autoregressive Models. In *The Thirteenth International Conference on Learning Representations*.

Gong, S.; Li, M.; Feng, J.; Wu, Z.; and Kong, L. 2023a. DiffuSeq: Sequence to Sequence Text Generation with Diffusion Models. In *The Eleventh International Conference on Learning Representations*.

Gong, S.; Li, M.; Feng, J.; Wu, Z.; and Kong, L. 2023b. DiffuSeq-v2: Bridging Discrete and Continuous Text Spaces for Accelerated Seq2Seq Diffusion Models. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Guo, S.; Lin, Y.; Feng, N.; Song, C.; and Wan, H. 2019. Attention Based Spatial-Temporal Graph Convolutional Networks for Traffic Flow Forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01): 922–929.

Huang, Y.; Xie, L.; Wang, X.; Yuan, Z.; Cun, X.; Ge, Y.; Zhou, J.; Dong, C.; Huang, R.; Zhang, R.; and Shan, Y. 2024. SmartEdit: Exploring Complex Instruction-based Image Editing with Multimodal Large Language Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 8362–8371.

Jeha, P.; Bohlke-Schneider, M.; Mercado, P.; Kapoor, S.; Nirwan, R. S.; Flunkert, V.; Gasthaus, J.; and Januschowski, T. 2022. PSA-GAN: Progressive Self Attention GANs for Synthetic Time Series. In *International Conference on Learning Representations*.

Jiang, W.; Zhao, W. X.; Wang, J.; and Jiang, J. 2023. Continuous Trajectory Generation Based on Two-Stage GAN. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(4): 4374–4382.

Karras, T.; Aittala, M.; Laine, S.; and Aila, T. 2022. Elucidating the design space of diffusion-based generative models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22. Red Hook, NY, USA: Curran Associates Inc.

Kong, Z.; Ping, W.; Huang, J.; Zhao, K.; and Catanzaro, B. 2021. DiffWave: A Versatile Diffusion Model for Audio Synthesis. In *International Conference on Learning Representations*.

Li, Y.; Yu, R.; Shahabi, C.; and Liu, Y. 2018. Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting. In *International Conference on Learning Representations*.

Liu, M.; Huang, H.; Feng, H.; Sun, L.; Du, B.; and Fu, Y. 2023a. PriSTI: A Conditional Diffusion Framework for Spatiotemporal Imputation. In *2023 IEEE 39th International Conference on Data Engineering (ICDE)*, 1927–1939.

Liu, X.; Xia, Y.; Liang, Y.; Hu, J.; Wang, Y.; BAI, L.; Huang, C.; Liu, Z.; Hooi, B.; and Zimmermann, R. 2023b. LargeST: A Benchmark Dataset for Large-Scale Traffic Forecasting. In Oh, A.; Naumann, T.; Globerson, A.; Saenko, K.; Hardt, M.; and Levine, S., eds., *Advances in Neural Information Processing Systems*, volume 36, 75354–75371. Curran Associates, Inc.

Naiman, I.; Berman, N.; Pemper, I.; Arbiv, I.; Fadlon, G.; and Azencot, O. 2024. Utilizing Image Transforms and Diffusion Models for Generative Modeling of Short and Long Time Series. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Ning, Y.; Liu, H.; Wang, H.; Zeng, Z.; and Xiong, H. 2023. UUKG: Unified Urban Knowledge Graph Dataset for Urban Spatiotemporal Prediction. In Oh, A.; Naumann, T.; Globerson, A.; Saenko, K.; Hardt, M.; and Levine, S., eds., *Advances in Neural Information Processing Systems*, volume 36, 62442–62456. Curran Associates, Inc.

Rao, X.; Shang, S.; Jiang, R.; Han, P.; and Chen, L. 2025. Seed: Bridging Sequence and Diffusion Models for Road Trajectory Generation. In *Proceedings of the ACM on Web Conference 2025*, WWW '25, 2007–2017. New York, NY, USA: Association for Computing Machinery.

van der Maaten, L.; and Hinton, G. 2008. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(86): 2579–2605.

Wang, J.; Lin, Y.; and Li, Y. 2025. GTG: Generalizable Trajectory Generation Model for Urban Mobility. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(1): 834–842.

Wei, T.; Lin, Y.; Guo, S.; Lin, Y.; Huang, Y.; Xiang, C.; Bai, Y.; and Wan, H. 2024. Diff-RNTraj: A Structure-Aware Diffusion Model for Road Network-Constrained Trajectory Generation. *IEEE Transactions on Knowledge and Data Engineering*, 36(12): 7940–7953.

Wen, H.; Lin, Y.; Xia, Y.; Wan, H.; Wen, Q.; Zimmermann, R.; and Liang, Y. 2023. DiffSTG: Probabilistic Spatio-Temporal Graph Forecasting with Denoising Diffusion Models. In *Proceedings of the 31st ACM International Conference on Advances in Geographic Information Systems*, SIGSPATIAL '23. New York, NY, USA: Association for Computing Machinery.

Xu, T.; Wenliang, L. K.; Munn, M.; and Acciaio, B. 2020. COT-GAN: Generating Sequential Data via Causal Optimal Transport. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.; and Lin, H., eds., *Advances in Neural Information Processing Systems*, volume 33, 8798–8809. Curran Associates, Inc.

Yan, H.; and Li, Y. 2025. Generative AI for Intelligent Transportation Systems: Road Transportation Perspective. *ACM Comput. Surv.* Just Accepted.

Yoon, J.; Jarrett, D.; and van der Schaar, M. 2019. Time-series Generative Adversarial Networks. In Wallach, H.; Larochelle, H.; Beygelzimer, A.; d'Alché-Buc, F.; Fox, E.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Yuan, X.; and Qiao, Y. 2024. Diffusion-TS: Interpretable Diffusion for General Time Series Generation. In *The Twelfth International Conference on Learning Representations*.

Zhao, Y.; Deng, P.; Liu, J.; Jia, X.; and Wang, M. 2023a. Causal Conditional Hidden Markov Model for Multimodal Traffic Prediction. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(4): 4929–4936.

Zhao, Y.; Deng, P.; Liu, J.; Jia, X.; and Zhang, J. 2023b. Generative Causal Interpretation Model for Spatio-Temporal Representation Learning. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '23, 3537–3548. New York, NY, USA: Association for Computing Machinery.

Zhicheng, C.; SHIBO, F.; Zhang, Z.; Xiao, X.; Gao, X.; and Zhao, P. 2024. SDformer: Similarity-driven Discrete Transformer For Time Series Generation. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Zhou, Z.; Ding, J.; Liu, Y.; Jin, D.; and Li, Y. 2023. Towards Generative Modeling of Urban Flow through Knowledge-enhanced Denoising Diffusion. In *Proceedings of the 31st ACM International Conference on Advances in Geographic Information Systems*, SIGSPATIAL '23. New York, NY, USA: Association for Computing Machinery.

Zhu, Y.; Ye, Y.; Zhang, S.; Zhao, X.; and Yu, J. 2023. DiffTraj: Generating GPS Trajectory with Diffusion Probabilistic Model. In Oh, A.; Naumann, T.; Globerson, A.; Saenko, K.; Hardt, M.; and Levine, S., eds., *Advances in Neural Information Processing Systems*, volume 36, 65168–65188. Curran Associates, Inc.