IMPUTATION AS INPAINTING: DIFFUSION MODELS FOR SPATIOTEMPORAL DATA IMPUTATION

Taeyoung Yun, Haewon Jung & Jiwoo Son
KAIST
{99yty, zora07, sonleave25}@kaist.ac.kr

ABSTRACT

Spatiotemporal data mining plays a crucial role in real-world scenarios such as air quality monitoring and intelligent traffic management. However, real-world spatiotemporal data collected in such scenarios is often incomplete due to sensor failures or transmission loss. Spatiotemporal imputation aims to fill in the missing values based on the observed values and their underlying spatiotemporal dependence. Previous dominant imputation models relied on autoregressive methods, which suffered from error accumulation. To overcome this limitation, emerging generative models like diffusion probabilistic models (DPM) can be employed for imputing missing values. These models are conditioned on observations to avoid relying solely on inaccurate historical imputation methods. However, applying diffusion models to spatiotemporal imputation presents challenges, particularly in extracting and utilizing conditional information from observed data. In this paper, we propose a novel framework for utilizing diffusion models for spatiotemporal imputation, by formulating imputation as inpainting problem. We first train an unconditional diffusion model for predicting the whole spatiotemporal data. To condition the generative process, we follow the scheme proposed by RePaint (Lugmayr et al., 2022), only alter reverse diffusion iterations by sampling the unobserved regions using the observed data information. To model spatial dependencies, we utilize a GNN-based backbone for DPM. We compare our model with state-of-the-art baselines in various missing patterns of two real-world spatiotemporal benchmark datasets.

1 INTRODUCTION

Spatiotemporal data is a type of data with intrinsic spatial and temporal patterns, which is widely applied in the real world for tasks such as air quality monitoring (Cao et al., 2018; Yi et al., 2016), traffic status forecasting (Li et al., 2018; Wu et al., 2019), weather prediction (Zhou et al., 2021) and so on. However, due to the sensor failures and transmission loss (Yi et al., 2016), the incompleteness in spatiotemporal data is a common problem, characterized by the randomness of missing value's positions and the diversity of missing patterns, which results in incorrect analysis of spatiotemporal patterns and further interference on downstream tasks. In recent years, extensive research (Cao et al., 2018; Cini et al., 2022; Fortuin et al., 2020) has dived into spatiotemporal imputation, with the goal of exploiting spatiotemporal dependencies from available observed data to impute missing values.

In the early studies, spatiotemporal imputation was commonly performed using statistical (Ansley & Kohn, 1984) and traditional machine learning techniques (Nelwamondo et al., 2007; Beretta & Santaniello, 2016) which involved imputing data either along the temporal or spatial dimension. But these methods impute missing values based on strong assumptions, such as the temporal smoothness and the similarity between time series and ignore the complexity of spatiotemporal correlations. With the advancements in deep learning, researchers have explored the utilization of recurrent neural networks (RNNs) for imputing missing values, as evidenced by studies such as (Cao et al., 2018; Cini et al., 2022). RNNs possess the ability to recursively update their hidden state, allowing for the recursive imputation of missing values. Although these approaches effectively capture spatiotemporal relationships with existing observations, they are prone to error accumulation, resulting in performance degradation.

In recent times, diffusion probabilistic models (DPM) (Sohl-Dickstein et al., 2015; Song et al., 2020; Ho et al., 2020) have gained attention as powerful generative models with remarkable performance across multiple tasks. Researchers have started employing these models for imputing missing values in multivariate time series. These methods initiate the imputation process by generating random samples from a Gaussian noise distribution and subsequently convert this noise into estimates for the missing values (Tashiro et al., 2021). By leveraging the flexibility of diffusion models in terms of neural network architecture, they can effectively overcome the error accumulation issue encountered in RNN-based methods. i.e., utilize attention mechanisms. Nevertheless, the application of diffusion models in spatiotemporal imputation encounters challenges in effectively construction of conditional information that captures spatiotemporal correlations. This motivates us to seek for an alternative approach to solve data imputation.

Contribution. This paper presents a novel imputation framework that addresses the imputation problem by treating it as an inpainting problem. We first train an unconditional diffusion model that predicts the whole spatiotemporal data. To condition the generative process, we consider missing values as masks in image inpainting and apply scheme proposed by RePaint (Lugmayr et al., 2022) for imputation. To model spatial dependecies, we leverages a GNN-based backbone for DPM. To this end, our proposed approach enhances generalization capabilities across different patterns of missing values.

2 Preliminaries

2.1 SpatioTemporal Data Imputation

We can formulate spatiotemporal data as sequences of weighted directed graphs, where we observe a graph $\mathcal{G}_t$ with N_t nodes at each time step t. A graph is a couple $\mathcal{G}_t = (\mathbf{X}_t, \mathbf{W}_t)$, where $\mathbf{X}_t \in \mathbb{R}^{N_t \times d}$ is the node-attribute matrix whose i-th row contains the d-dimensional node-attribute vector $\mathbf{x}_t^i \in \mathbb{R}^d$ associated with the i-th node; entry $w_t^{i,j}$ of the adjacency matrix $\mathbf{W}_t \in \mathbb{R}^{N_t \times N_t}$ denotes the scalar weight of the edge (if any) connecting the i-the node and j-th node. While this problem setting can be easily extended to more general classes of graphs with dynamic edge setting, we focus on problems where the topology of the graph is fixed and does not change over time (e.g traffic network), i.e., at each time step $\mathbf{W}_t = \mathbf{W}$ and $N_t = N$.

To model the presence of missing values, we consider, at each step t, a binary mask $\mathbf{M}_t \in \{0,1\}^{N_t \times d}$ where each row $\mathbf{m}_t^i$ indicates which of the corresponding node attributes of $\mathbf{x}_t^i$ are available in $\mathbf{X}_t$. It follows that, $m_t^{i,j} = 0$ implies $x_t^{i,j}$ is missing; conversely, if $m_t^{i,j} = 1$, then $x_t^{i,j}$ stores the actual sensor reading. We denote by $\widetilde{\mathbf{X}}_t$ the the complete node-attribute matrix without any missing data.

The objective of spatio-temporal data imputation is to impute missing values in a sequence of input data. More formally, given a sequence $\mathcal{G}_{[t:t+T]}$ of length T, we tries to minimize the missing data reconstruction error defined as:

$$\mathcal{L}\left(\hat{\mathbf{X}}_{[t:t+T]}, \widetilde{\mathbf{X}}_{[t:t+T]}, \mathbf{M}_{[t:t+T]}\right) = \sum_{h=t}^{t+T} \sum_{i=1}^{N_t} \frac{\langle \mathbf{m}_h^i, \ell(\hat{\mathbf{x}}_h^i, \tilde{\mathbf{x}}_h^i) \rangle}{\langle \mathbf{m}_h^i, \mathbf{m}_h^i \rangle} \tag{1}$$

where $\hat{\mathbf{x}}_h^i$ is the reconstructed $\tilde{\mathbf{x}}_h^i$, $\ell(\cdot,\cdot)$ is an element-wise error function (e.g. absolute or squared error) and $\langle \cdot,\cdot \rangle$ indicates standard dot product.

2.2 Denoising Diffusion Probabilistic Models

Diffusion models have gained significant popularity in recent years, owing to their ability to generate data similar to the training data. In the 2020s, several seminal papers were published, showcasing the capabilities of diffusion models. These models have outperformed GANs (Goodfellow et al., 2014) in image synthesis, making them a preferred choice for many applications. Recently, diffusion models have been implemented in OpenAI's image generation model, DALL-E 2 (Ramesh et al., 2022).

Fundamentally, diffusion models are generative models that operate by progressively adding Gaussian noise to the training data (**forward process**) and then learning to recover the original data by reversing this noising process (**reverse process**). The forward process is defined by the following Markov chain. To prevent confusion of notation on time series, we use $k = 0, \cdots, K$ for denoising steps:

$$q(\mathbf{x}_{1:K}|\mathbf{x}_0) := \prod_{k=1}^{K} q(\mathbf{x}_k|\mathbf{x}_{k-1}) \text{ where } q(\mathbf{x}_k|\mathbf{x}_{k-1}) := \mathcal{N}(\sqrt{1-\beta_k}\mathbf{x}_{k-1}, \beta_k I) \tag{2}$$

and β_t is a small positive constant that represents a noise level.

On the other hand, the reverse process denoises $\mathbf{x}_K$ to recover $\mathbf{x}_0$, and is defined by the following Markov chain:

$$p_\theta(\mathbf{x}_{0:K}) := p(\mathbf{x}_K) \prod_{k=1}^{K} p_\theta(\mathbf{x}_{k-1}|\mathbf{x}_k), \mathbf{x}_K \sim \mathcal{N}(0, I) \tag{3}$$

$$p_\theta(\mathbf{x}_{k-1}|\mathbf{x}_k) := \mathcal{N}(\mathbf{x}_{k-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_k, k), \boldsymbol{\sigma}_\theta(\mathbf{x}_k, k)I) \tag{4}$$

is then optimized by maximizing the evidence lower bound defined as $\mathbb{E}_q[\frac{p_\theta(\mathbf{x}_{0:K})}{q(\mathbf{x}_{1:K}|\mathbf{x}_0)}]$. After training, sampling from the diffusion model consists of sampling $\mathbf{x}_K \sim p(\mathbf{x}_K)$ and running the reverse diffusion chain to go from $k = K$ to $k = 0$.

The diffusion model can generate data by passing randomly sampled noise through the learned denoising process. After training, diffusion models can produce high-quality synthetic data, making them valuable for various applications.

3 METHODOLOGY

In this section, we introduce a new approach for imputing missing values in spatiotemporal data using diffusion models. Before delving into the details of our proposed methodology, we'll provide an overview of how existing diffusion models have previously tackled spatiotemporal imputation problems.

CSDI (Tashiro et al., 2021) utilizes score-based diffusion model conditioned on observed data to impute missing values. Specifically, they formulate conditional diffusion model as follows:

$$p_\theta(\mathbf{X}^{\text{ta}}_{t,0:K} \mid \mathbf{X}^{\text{co}}_{t,0}) = p(\mathbf{X}^{\text{ta}}_{t,K}) \prod_{k=1}^{K} p_\theta(\mathbf{X}^{\text{ta}}_{t,k-1} \mid \mathbf{X}^{\text{ta}}_{t,k}, \mathbf{X}^{\text{co}}_{t,0}), \qquad \mathbf{X}^{\text{ta}}_{t,K} \sim \mathcal{N}(0, I) \tag{5}$$

$$p_\theta(\mathbf{X}^{\text{ta}}_{t,k-1} \mid \mathbf{X}^{\text{ta}}_{t,k}, \mathbf{X}^{\text{co}}_{t,0}) = \mathcal{N}(\mathbf{X}^{\text{ta}}_{t,k-1}; \boldsymbol{\mu}_\theta(\mathbf{X}^{\text{ta}}_{t,k}, k \mid \mathbf{X}^{\text{co}}_{t,0}), \boldsymbol{\sigma}_\theta(\mathbf{X}^{\text{ta}}_{t,k}, k \mid \mathbf{X}^{\text{co}}_{t,0})\mathbf{I}) \tag{6}$$

where $\mathbf{X}^{\text{ta}}_{t,k}$ is generated missing values and $\mathbf{X}^{\text{co}}_0$ is an observed data at time step t, respectively.

However, this modeling approach has certain limitations. Firstly, it is challenging to extract meaningful information from observed data, especially when dealing with time correlation dependencies. This is because the global conditional representation encapsulates all details concerning these dependencies. Moreover, accommodating various missing patterns becomes complex due to the variable positions of $\mathbf{X}^{\text{co}}$ and $\mathbf{X}^{\text{ta}}$.

To this end, we introduce a new framework to impute missing values in spatiotemporal data, by formulating the problem as inpainting problem. Consider imputation as inpainting problem has following advantages. First of all, we do not need to train an conditional diffusion model and do not care about how to extract appropriate conditional information from an observed data. Instead, we can use a powerful unconditional diffusion model trained on various time series data, which produces accurate predictions on any missing patterns.

Our methodology for imputing missing values in spatiotemporal data using diffusion models can be divided into two steps, as shown in Figure 1:

3.1 TRAINING AN GNN-BASED UNCONDITIONAL DIFFUSION MODEL

The first stage of our process involves training an unconditional diffusion model. Unlike the conventional approach, where the model is conditioned on observed data, we allow our model to learn freely from the inherent distribution and dynamics of the data. This results in a more robust model that can capture intricate patterns and relationships within the dataset.

Additionally, to utilize spatial dependencies, we adopt Graph Neural Network as backbone of our diffusion model instead of standard UNet architecture. Given input graph $\mathcal{G}_t = (\mathbf{X}_t, \mathbf{W})$, our diffusion model can be formulated as follows:

$$p_\theta(\mathbf{X}_{t,0:K}) = p(\mathbf{X}_{t,K}) \prod_{k=1}^{K} p_\theta(\mathbf{X}_{t,k-1} \mid \mathbf{X}_{t,k}) \tag{7}$$

$$p_\theta(\mathbf{X}_{t,k-1} \mid \mathbf{X}_{t,k}) = \mathcal{N}(\mathbf{X}_{t,k-1}; \boldsymbol{\mu}_\theta(\mathbf{X}_{t,k}, \mathbf{W}, t), \sigma_k^2 I) \tag{8}$$

Following (Ho et al., 2020), we also define $\boldsymbol{\mu}_\theta(\mathbf{X}_{t,k}, \mathbf{W}, t)$ as follows:

$$\boldsymbol{\mu}_\theta(\mathbf{X}_{t,k}, \mathbf{W}, t) = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{X}_{t,k} - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} \boldsymbol{\epsilon}_\theta(\mathbf{X}_{t,k}, \mathbf{W}, t) \right) \tag{9}$$

where we parametrize $\boldsymbol{\epsilon}_\theta(\mathbf{X}_{t,k}, \mathbf{W}, t)$ as L-layer GNN. Specifically, we utilize graph attention networks (GAT) as our backbone network. Finally, our training objective can be written as follows:

$$\mathcal{L}(\theta) = \mathbb{E}_{\boldsymbol{\epsilon}, t, \mathbf{X_{t,0}}} \left[\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{X}_{t,k}, \mathbf{W}, t) \|^2 \right] \tag{10}$$

3.2 Imputation as Inpainting

Upon the successful training of the unconditional diffusion model, we employ the inpainting approach for missing value imputation. Among various inpainting approaches, We follow the scheme of Repaint (Lugmayr et al., 2022), due to its simplicity and effectiveness. For every reverse step, we alter the known region $\mathbf{M}_t \odot \mathbf{X}_{t,k}$ as long as we keep the correct properties of the corresponding distribution as follows:

$$\mathbf{X}_{t,k-1}^{\text{known}} \sim \mathcal{N}(\sqrt{\bar{\alpha}_t}\mathbf{X}_{t,0}, (1-\hat{\alpha}_t)\mathbf{I}) \tag{11}$$

$$\mathbf{X}_{t,k-1}^{\text{unknown}} \sim \mathcal{N}(\boldsymbol{\mu}_\theta(\mathbf{X}_{t,k}, \mathbf{W}, k), \sigma_k^2 I) \tag{12}$$

$$\mathbf{X}_{t,k-1} = \mathbf{M}_t \odot \mathbf{X}_{t,k-1}^{\text{known}} + (1-\mathbf{M}_t) \odot \mathbf{X}_{t,k-1}^{\text{unknown}} \tag{13}$$

In other words, we sample $\mathbf{X}_{t,k-1}^{\text{known}}$ using the known pixels of given image and only sample $\mathbf{X}_{t,k-1}^{\text{known}}$ from the pretrained diffusion model. These are then combined to the new sample $\mathbf{X}_{t-1}$ using the mask operation.

After conditioning on the known region, we apply resampling technique as proposed in RePaint to improve the reverse process itself for inpainting. In resampling, we diffuse the output $\mathbf{X}_{t,k-1}$ back to $\mathbf{X}_{t,k}$ by sampling from the forward process, $\mathbf{X}_{t,k} \sim \mathcal{N}(\sqrt{1-\beta_t}\mathbf{X}_{t,k-1}, \beta_t I)$. Although this operation scales back the output and adds noise, it eventually leads to a new $\mathbf{X}_{t,k}^{\text{unknown}}$ which is more harmonized with $\mathbf{X}_{t,k}^{\text{known}}$.

4 Experiments

4.1 Experiment Setting

Datasets. Our experimental evaluation is conducted on two real-world datasets: METR-LA and PEMS-BAY. The METR-LA (Li et al., 2018) consists of traffic speed data collected from 207 sensors along the highway in Los Angeles County (Jagadish et al., 2014) over a duration of 4 months. Similarly, PEMS-BAY (Li et al., 2018) includes traffic speed data collected from 325 sensors on

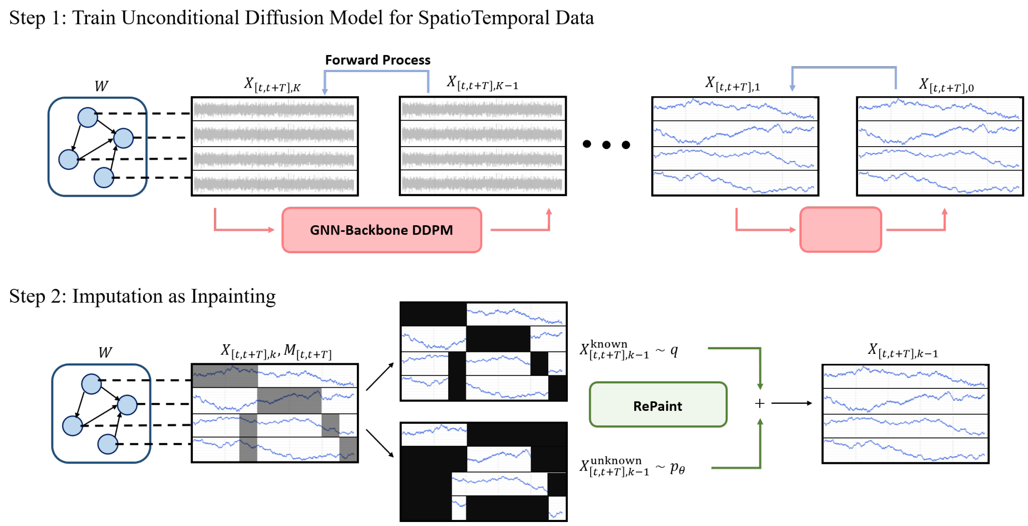

Figure 1: Overview of our proposed method. **(Step 1)**: Train unconditional diffusion model to generate time series data. **(Step 2)**: For each denoising step, we sample the known region from observed data and the unknown region from the DDPM output.

highways in the San Francisco Bay Area over a period of 6 months. Both traffic datasets are sampled every 5 minutes. For the incorporation of geographic information, we construct the adjacency matrix based on the geographic distances between monitoring stations or sensors, following previous works (Li et al., 2018). Specifically, we utilize a thresholded Gaussian kernel (Shuman et al., 2013) to generate the adjacency matrix for all two datasets.

Imputation Scenario. We use the artificially injected missing strategy provided by (Cini et al., 2022) for evaluation, which includes two missing patterns: (1) **Block Missing:** based on randomly masking 5% of the observed data, mask observations ranging from 1 to 4 hours for each sensor with 0.15% probability; (2) **Point missing:** randomly mask 25% of observations.

Baselines. In order to compare the previous work and our proposed method, we consider a range of classic models and state-of-the-art techniques for spatiotemporal imputation. The baselines encompass statistical methods such as MEAN and KNN, a low-rank matrix factorization method known as MF, deep autoregressive methods including BRITS (Cao et al., 2018) and GRIN (Cini et al., 2022), as well as deep generative methods like rGAIN (Miao et al., 2021) and CSDI (Tashiro et al., 2021).

4.2 Experiment Results

In our evaluation, we employ the mean absolute error (MAE) and mean squared error (MSE) as metrics to quantify the efficacy of spatiotemporal imputation.

Our proposed model outperforms traditional methods and achieves comparable performance to state-of-the-art (SOTA) models, despite being an unconditioned generative model. Despite high expectations from the powerful performance of diffusion, it did not function as anticipated. The likely cause of this inadequate performance may be attributed to the unconditional generative model's difficulty in capturing the intricate distribution of time series data. The numerical results derived from these experiments are presented in Table 1.

4.3 Ablation Studies

Furthermore, we initiate ablation studies to scrutinize the influence of the selected backbone on the parametrization of diffusion models. These findings are showcased in Table 2. From our analyses, it appears that the deployment of a GNN architecture yields greater effectiveness in contrast to the elementary usage of a UNet architecture for the purpose of spatiotemporal data imputation.

This discernment provides an implication that the GNN architecture skillfully capitalizes on spatial dependencies, thereby bringing about an enhancement in the quality of imputation. The exploitation of such dependencies allows for a more accurate representation of the data's inherent structures, contributing to the model's ability to handle complex patterns and associations in spatiotemporal data. This further emphasizes the importance of suitable architecture choice for such intricate data processing tasks.

Table 1: Performance of all methods on benchmark datasets. Experiment is conducted with 3 different random seeds, mean and one standard deviation is reported.

Category	Method	METR-LA				PEMS-BAY			
		Block Missing		Point Missing		Block Missing		Point Missing	
		MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
Heuristic	Mean	7.48±0.00	139.54±0.00	7.56±0.00	142.22±0.00	5.46±0.00	87.56±0.00	5.42±0.00	86.59±0.00
	KNN	7.79±0.00	124.61±0.00	7.88±0.00	129.29±0.00	4.30±0.00	49.90±0.00	4.30±0.00	49.80±0.00
	MF	5.46±0.02	109.61±0.78	5.56±0.03	113.46±1.08	3.28±0.01	50.14±0.13	3.29±0.01	51.39±0.64
Learning based	rGAIN	2.90±0.01	21.67±0.15	2.83±0.01	20.03±0.09	2.18±0.01	13.96±0.20	1.88±0.02	10.37±0.20
	BRITS	2.34±0.01	17.00±0.14	2.34±0.00	16.46±0.05	1.70±0.01	10.50±0.07	1.47±0.00	7.94±0.03
	GRIN	2.03±0.00	13.26±0.05	1.91±0.00	10.41±0.03	1.14±0.01	6.60±0.10	0.67±0.00	1.55±0.01
Diffusion based	CSDI	**1.98**±0.00	**12.62**±0.60	**1.79**±0.00	**8.96**±0.08	**0.86**±0.00	**4.39**±0.02	**0.57**±0.00	**1.12**±0.03
	Ours	3.67±0.00	40.63±0.00	2.53±0.00	14.86±0.00	2.02±0.00	11.25±0.00	1.42±0.00	8.19±0.00

Table 2: Ablation studies on the choice of backbone for our diffusion models.

Method	METR-LA				PEMS-BAY			
	Block Missing		Point Missing		Block Missing		Point Missing	
	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
Ours (UNet)	4.58±0.01	56.89±0.02	3.16±0.00	28.53±0.00	2.08±0.00	13.11±0.01	1.86±0.00	12.10±0.01
Ours (GNN)	3.67±0.00	40.63±0.00	2.53±0.00	14.86±0.00	2.02±0.00	11.25±0.00	1.42±0.00	8.19±0.00

5 Related Works

5.1 Spatio-Temporal Data Imputation

In recent years, there has been significant research on spatiotemporal data imputation using deep learning methods. Most of these approaches focus on multivariate time series and utilize recurrent neural networks (RNNs) as the core for modeling temporal relationships (Cao et al., 2018; Cini et al., 2022; Yoon et al., 2018; Che et al., 2018). The RNN-based approach for imputation was initially introduced by GRU-D (Che et al., 2018) and has since been widely adopted in deep autoregressive imputation methods. Among the RNN-based methods, BRITS (Cao et al., 2018) imputes missing values based on the hidden state using a bidirectional RNN and considers the correlation between features. GRIN (Cini et al., 2022) extends BRITS by incorporating graph neural networks to leverage the inductive bias of historical spatial patterns for imputation.

In the CSDI framework (Tashiro et al., 2021), which is based on a diffusion model, deep generative models are utilized for imputing missing data in multivariate time series. CSDI leverages score-based diffusion models conditioned on observed data and incorporate a two-dimensional attention mechanism to capture temporal and feature correlations. One challenge faced by CSDI is that during training, it takes the concatenation of observed values and noisy information as input. This can increase the difficulty of the attention mechanism's learning process. Different from existing diffusion model-based imputation methods, our proposed method treats the imputation problem as an inpainting problem, which is commonly used in image generation tasks. By treating the imputation problem as an inpainting problem, we aim to leverage techniques commonly used in image generation to handle missing data in multivariate time series.

5.2 Image Inpainting with Diffusion Models

In the domain of image processing, inpainting is a technique employed to substitute missing or corrupted sections of images. The objective of this process is to accomplish a plausible and visually coherent substitution, utilizing information from the surrounding areas of the image. Various methodologies can be utilized for this purpose, encompassing deep learning approaches (Yu et al., 2019; Xiong et al., 2019; Liu et al., 2018) as well as diffusion-based processes (Lugmayr et al., 2022; Zhang et al., 2023).

The study by Lugmayr et al. Lugmayr et al. (2022) introduces RePaint, a novel approach to free-form inpainting utilizing DDPM, which exhibits effectiveness even for extreme masks. This method leverages a pre-trained unconditional DDPM as the generative prior and modifies the reverse diffusion iterations by sampling the unmasked regions using the provided image data.

In parallel, the research conducted by Zhang et al. Zhang et al. (2023) presents a new algorithm known as COPAINT, designed for coherent image inpainting using DDIM. Adopting a Bayesian framework, COPAINT simultaneously modifies both the revealed and hidden regions, while approximating the posterior distribution in a manner that enables the error to gradually diminish through the denoising steps. The experimental findings indicate that COPAINT surpasses existing diffusion-based methods in terms of both objective and subjective metrics.

6 Conclusion and Future Work

We present a novel framework to impute missing values in spatiotemporal data using diffusion models by reformulating imputation problem as inpainting problem. While our approach is easy to implement and show comparative performance, it does not reach performance of using an conditional diffusion model. One of the main weaknesses of our model is a lack of consideration on temporal dependencies. There are diffusion models which consider temporal correlations such as DiffWave (Kong et al., 2021), which is also a backbone architecture of the (Tashiro et al., 2021). It might be powerful if we integrate GNN and Diffwave to build a diffusion model that can capture both spatial and temporal dependencies.

References

Craig F Ansley and Robert Kohn. On the estimation of arima models with missing values. In *Time series analysis of irregularly observed data*, pp. 9–37. Springer, 1984.

Lorenzo Beretta and Alessandro Santaniello. Nearest neighbor imputation algorithms: a critical evaluation. *BMC medical informatics and decision making*, 16(3):197–208, 2016.

Wei Cao, Dong Wang, Jian Li, Hao Zhou, Lei Li, and Yitan Li. Brits: Bidirectional recurrent imputation for time series. *NeurIPS*, 31, 2018.

Zhengping Che, Sanjay Purushotham, Kyunghyun Cho, David Sontag, and Yan Liu. Recurrent neural networks for multivariate time series with missing values. *Scientific reports*, 8(1):1–12, 2018.

Andrea Cini, Ivan Marisca, and Cesare Alippi. Filling the g_ap_s: Multivariate time series imputation by graph neural networks. In *ICLR*, 2022.

Vincent Fortuin, Dmitry Baranchuk, Gunnar Rätsch, and Stephan Mandt. Gp-vae: Deep probabilistic time series imputation. In *AISTATS*, pp. 1651–1661. PMLR, 2020.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in Neural Information Processing Systems*, 27, 2014.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 33: 6840–6851, 2020.

Hosagrahar V Jagadish, Johannes Gehrke, Alexandros Labrinidis, Yannis Papakonstantinou, Jignesh M Patel, Raghu Ramakrishnan, and Cyrus Shahabi. Big data and its technical challenges. *Communications of the ACM*, 57(7):86–94, 2014.

Hyosoon Jang, Sangwoo Mo, and Sungsoo Ahn. Diffusion probabilistic models for graph-structured prediction. *arXiv preprint arXiv:2302.10506*, 2023.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. Diffwave: A versatile diffusion model for audio synthesis. In *ICLR*, 2021.

Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In *ICLR*, 2018.

Guilin Liu, Fitsum A. Reda, Kevin J. Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11461–11471, 2022.

Xiaoye Miao, Yangyang Wu, Jun Wang, Yunjun Gao, Xudong Mao, and Jianwei Yin. Generative semi-supervised learning for multivariate time series imputation. In *AAAI*, volume 35, pp. 8983–8991, 2021.

Fulufhelo V Nelwamondo, Shakir Mohamed, and Tshilidzi Marwala. Missing data: A comparison of neural network and expectation maximization techniques. *Current Science*, pp. 1514–1521, 2007.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

David I Shuman, Sunil K Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE signal processing magazine*, 30(3):83–98, 2013.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, pp. 2256–2265. PMLR, 2015.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. Csdi: Conditional score-based diffusion models for probabilistic time series imputation. *NeurIPS*, 34:24804–24816, 2021.

Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Chengqi Zhang. Graph wavenet for deep spatial-temporal graph modeling. In *IJCAI*, 2019.

Wei Xiong, Jiahui Yu, Zhe Lin, Jimei Yang, Xin Lu, Connelly Barnes, and Jiebo Luo. Foreground-aware image inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

Xiuwen Yi, Yu Zheng, Junbo Zhang, and Tianrui Li. St-mvl: filling missing values in geo-sensory time series data. In *IJCAI*, 2016.

Jinsung Yoon, William R Zame, and Mihaela van der Schaar. Estimating missing data in temporal data streams using multi-directional recurrent neural networks. *IEEE Transactions on Biomedical Engineering*, 66(5):1477–1490, 2018.

Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S. Huang. Free-form image inpainting with gated convolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

Guanhua Zhang, Jiabao Ji, Yang Zhang, Mo Yu, Tommi Jaakkola, and Shiyu Chang. Towards coherent image inpainting using denoising diffusion implicit models, 2023.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *AAAI*, volume 35, pp. 11106–11115, 2021.

A IMPLEMENTATION DETAILS

In this subsection, we describe the implementation details of our method. For UNet-Based diffusion model, we utilize implementation from (Ho et al., 2020) and its Pytorch version[1]. For GNN-based diffusion model, we modify implementation from (Jang et al., 2023) on fully-supervised setting since we train an unconditional diffusion models. For GNN architecture, we use GCN (Kipf & Welling, 2016) architecture. To impute missing values, we utilize codebase from (Lugmayr et al., 2022).

For both models, we train diffusion models for 100,000 epochs with batch size of 32, using one RTX 3090 NVIDIA GPU. We follow default setting of original implementations for other hyperparamters such as learning rate, weight decay, and hidden dimension sizes. We check that after 50,000 steps, loss is converged. During imputation stage, we set the diffusion step $K = 50$, which is the same with the setting of (Tashiro et al., 2021).

[1]https://github.com/lucidrains/denoising-diffusion-pytorch