Causality-Aware Spatiotemporal Graph Neural Networks for Spatiotemporal Time Series Imputation

Baoyu Jing
University of Illinois
Urbana-Champaign, IL, USA
baoyuj2@illinois.edu

Dawei Zhou
Virginia Polytechnic Institute and State University
Blacksburg, VA, USA
zhoud@vt.edu

Kan Ren
Microsoft Research Asia
Shanghai, China
kan.ren@microsoft.com

Carl Yang
Emory University
Atlanta, GA, USA
j.carlyang@emory.edu

Abstract

Spatiotemporal time series are usually collected via monitoring sensors placed at different locations, which usually contain missing values due to various failures, such as mechanical damages and Internet outages. Imputing the missing values is crucial for analyzing time series. When recovering a specific data point, most existing methods consider all the information relevant to that point regardless of the cause-and-effect relationship. During data collection, it is inevitable that some unknown confounders are included, e.g., background noise in time series and non-causal shortcut edges in the constructed sensor network. These confounders could open backdoor paths and establish non-causal correlations between the input and output. Over-exploiting these non-causal correlations could cause overfitting. In this paper, we first revisit spatiotemporal time series imputation from a causal perspective and show how to block the confounders via the frontdoor adjustment. Based on the results of frontdoor adjustment, we introduce a novel Causality-Aware Spatiotemporal Graph Neural Network (Casper), which contains a novel Prompt Based Decoder (PBD) and a Spatiotemporal Causal Attention (SCA). PBD could reduce the impact of confounders and SCA could discover the sparse causal relationships among embeddings. Theoretical analysis reveals that SCA discovers causal relationships based on the values of gradients. We evaluate Casper on three real-world datasets, and the experimental results show that Casper could outperform the baselines and could effectively discover the causal relationships.

CCS Concepts

• **Information systems** → **Data mining**; **Sensor networks**.

Keywords

Spatiotemporal Time Series Imputation, Spatiotemporal Graph Neural Network, Causal Attention

This work is licensed under a Creative Commons Attribution International 4.0 License.

CIKM'24, October 21–25, 2024, Boise, ID, USA
© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-0436-9/24/10
https://doi.org/10.1145/3627673.3679642

ACM Reference Format:
Baoyu Jing, Dawei Zhou, Kan Ren, and Carl Yang. 2024. Causality-Aware Spatiotemporal Graph Neural Networks for Spatiotemporal Time Series Imputation. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM '24), October 21–25, 2024, Boise, ID, USA.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3627673.3679642

1 Introduction

Spatiotemporal data mining [3] is the cornerstone of analyzing and understanding the patterns of spacetime and human activities, such as environmental monitoring [43, 50, 57, 94], e-business [28, 31, 32, 35, 41, 77, 80, 95] and social science [12, 16, 17, 76, 79, 86, 88, 90, 96]. Time series [30, 42, 70, 91] is one of the most common data types, which is usually collected by monitoring sensors. For example, traffic flow time series [43], e.g., speed, is recorded by the radar sensors on roads. Air pollution time series [94], e.g., concentrations of PM2.5, is collected from air quality monitoring sites across cities.

In the real world, it is not uncommon that the collected spatiotemporal time series is incomplete with missing data due to various failures, e.g., sensors have mechanical damage. The missing data usually significantly impacts the process and conclusion of data analysis. Therefore, how to reconstruct the missing data from the observed data, i.e., imputation, is a fundamental problem of spatiotemporal time series analysis. In recent years, deep learning methods become the mainstream for time series imputation. Most existing deep time series imputation methods [7, 45, 46] use Recurrent Neural Network (RNN) to capture the temporal dynamics of time series and autoregressively recover the missing data by the predicted values. Recent deep learning methods [83] propose to use non-autoregressive structures, e.g., Transformer [68], to avoid the progressive error propagation incurred via the autoregression in RNN by concurrently considering the entire input context. However, these methods only consider the temporal patterns yet overlook the spatial relationships among sensors, e.g., geographical distances. To further account for spatial relationships, graph neural networks [39, 69, 74, 75, 87, 92] are extended to the spatiotemporal setting [24, 29, 50, 70]. Although these methods have achieved impressive performance in recovering the missing values, they tend to include all the available information related to the missing point as references without distinguishing whether there is a causal relationship between them.

When collecting datasets, it is inevitable to include some unknown confounders [54]. For example, the background noise might be recorded, and non-causal shortcut edges might be built for two sensors. Let's take the air monitoring sensor network as a concrete example to understand the non-causal edges. A common practice to build the network is adding an edge for two sensors if their distance is below a threshold [50]. Although simple and usually effective, the distance-based network does not necessarily imply the real causality between sensors. In the real world, air flow between two locations could be influenced by other factors, e.g., wind direction and terrain. An example is shown in Figure 6 in Section 4.4. Simply exploiting the shortcut edges without discovering the causality could make the model overfit the training data and be vulnerable to noise during inference.

To reduce the negative effects brought by confounders, we first review the process of spatiotemporal time series imputation from a causal perspective [54] to show the causal relationships among the input, output, embeddings, and confounders. The results show that confounders could establish undesired non-causal shortcut backdoor paths between the input and output. Then, we show how to eliminate the backdoor paths via the frontdoor adjustment [54]. Based on the results of the frontdoor adjustment, we introduce a novel Causality-Aware Spatiotemporal Graph Neural Network (Casper), which is equipped with a novel Prompt Based Decoder (PBD) and a Spatiotemporal Causal Attention (SCA). The proposed PBD effectively reduces the impact of unknown confounders by injecting the global context information of datasets into the embeddings. Inspired by [26], which uses learnable prompts to capture the contextual information of downstream tasks when tuning visual models, PBD leverages prompts to learn the contextual information of datasets automatically rather than employing external models to approximate the context. To further enforce sparse causality between embeddings, we introduce SCA, which determines the cause-and-effect relationship via a causal gate. It can be theoretically proven that the proposed causal gate (1) enforces the sparsity since it converges to 0 or 1; (2) is a gradient-based explanation similar to [61], which determines the causality based on the values of gradients. We extensively evaluate Casper on three real-world datasets. The experimental results show that Casper could significantly outperform baselines and could effectively discover causality.

The major contributions of the paper are summarized as follows:

- We review the spatiotemporal time series imputation task from a causal perspective, where we show the problems of the undesired confounders. Then, we show how to eliminate confounders via the frontdoor adjustment.
- We propose a novel Causality-Aware Spatiotemporal Graph Neural Network (Casper) based on the frontdoor adjustment. Casper is equipped with a novel Spatiotemporal Causal Attention (SCA) and a Prompt Based Decoder (PBD). PBD effectively blocks the backdoor paths and SCA explicitly reveals the causality between embeddings.
- We provide theoretical analysis to deeply understand how Casper determines causal and non-causal relationships.
- We evaluate Casper on three real-world datasets. The experimental results show that Casper could outperform the baselines and effectively discover causal relationships.

2 Preliminary

In this section, we briefly introduce the definitions of spatiotemporal time series and spatiotemporal time series imputation. We also review the definitions of Granger causality and attention function.

Definition 1 (Incomplete Spatiotemporal Time Series)**.** We denote an incomplete spatiotemporal time series with missing values as $\mathbf{G} = (\mathbf{X}, \mathbf{A}, \mathbf{M})$, where $\mathbf{X} \in \mathbb{R}^{N \times T}$ is the multivariate time series collected from N sensors with totally T steps, $\mathbf{A} \in \mathbb{R}^{N \times N}$ is the adjacency matrix of the sensor network, $\mathbf{M} \in \{0, 1\}^{N \times T}$ is the binary mask and 0/1 denotes the absence/presence of a data point.

Definition 2 (Spatiotemporal Time Series Imputation)**.** Given an incomplete spatiotemporal time series $\mathbf{G} = (\mathbf{X}, \mathbf{A}, \mathbf{M})$, we denote $\mathbf{Y} \in \mathbf{R}^{N \times T}$ as the complete time series of $\mathbf{X}$, such that $\mathbf{X} = \mathbf{M} \odot \mathbf{Y}$, where $\odot$ is the Hadamard product. The task is to build a function $\hat{\mathbf{Y}} = f(\mathbf{G})$ to minimize the reconstruction error, e.g., Mean Absolute Error (MAE), between $\hat{\mathbf{Y}}$ and $\mathbf{Y}$.

Definition 3 (Granger Causality [9, 19])**.** Let $\mathbf{X} \in \mathbb{R}^{N \times T}$ be the values of past T steps of N time series, and $\hat{y}_{i,T+1} = f_i(\mathbf{X}) \in \mathbb{R}$ be the *time series forecasting function* predicting the future value of the i-th time series at the $T+1$ step, where $i \in \{1, \ldots, N\}$. The i'-th time series is said to Granger cause the i-th time series if there exists a point $x'_{i',t'} \neq x_{i',t'}$, $t' \in \{1, \ldots, T\}$, such that $f_i(\mathbf{X}') \neq f_i(\mathbf{X})$, where $\mathbf{X}'$ is obtained by replacing $x_{i',t'}$ in $\mathbf{X}$ with $x'_{i',t'}$.

Generally, if $x_{i',t'}$ impacts the prediction of the future value of the i-th time series, then the i'-th time series Granger causes the i-th time series. In the case that f_i is a linear model:

$$\hat{y}_{i,T+1} = f_i(\mathbf{X}) = \sum_{i'=1,t'=1}^{N,T} \alpha_{i',t'} x_{i',t'}. \tag{1}$$

if the coefficient $\alpha_{i',t'} \neq 0$, then the i'-th time series Granger causes the i-th time series.

Definition 4 (Attention Function)**.** Let $\mathbf{q}$ and $\{\mathbf{k}_i\}_{i=1}^N$ be the d-dimensional query and key embeddings, let s and f_v be the scoring and message functions, then the attention function is defined as:

$$\mathbf{h} = \sum_{i=1}^{N} \alpha_i \mathbf{v}_i, \quad \alpha_i = \frac{\exp(s(\mathbf{q}, \mathbf{k}_i))}{\sum_{i'=1}^{N} \exp(s(\mathbf{q}, \mathbf{k}_{i'}))}, \quad \mathbf{v}_i = f_v(\mathbf{q}, \mathbf{k}_i), \tag{2}$$

where $\mathbf{h} \in \mathbb{R}^d$ is the output, and $\mathbf{v}_i$ is the message from $\mathbf{k}_i$ to $\mathbf{q}$.

3 Methodology

In this section, we first provide a causal view of the spatiotemporal imputation task, and show how to eliminate the impact of unknown confounds by frontdoor adjustment, based on which, we introduce a novel Causality-Aware Spatiotemporal Graph Neural Network (Casper). Finally, we provide further analysis of Casper.

3.1 Causal View of Spatiotemporal Imputation

Given an incomplete spatiotemporal time series $\mathbf{G} = (\mathbf{X}, \mathbf{A}, \mathbf{M})$, a standard deep imputation model $f = f^D \circ f^E$, where f^E, f^D are the encoder and decoder, works as follows: (1) f^E extracts embeddings $\mathcal{H} = \{\mathbf{h}_{i,t}\}_{i=1,t=1}^{N,T}$ from $\mathbf{G}$, (2) f^D generates predictions $\{\hat{y}_{i,t}\}_{i=1,t=1}^{N,T}$ based on $\mathcal{H}$ to recover $\mathcal{Y} = \{y_{i,t}\}_{i=1,t=1}^{N,T}$. The model f is trained by

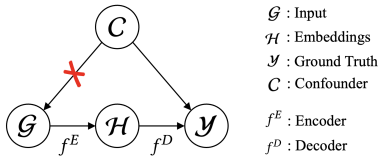


Figure 1: The Structure Causal Model (SCM) for spatiotemporal imputation. The frontdoor adjustment removes the edge between confounder $\mathcal{C}$ and input $\mathcal{G}$.

a reconstruction error e.g., MAE or RMSE. Since minimizing MAE (or RMSE) is equivalent to maximizing the log-likelihood of Laplace (or Gaussian) distribution [22], and thus we can view the objective of spatiotemporal imputation as maximizing $P(\mathcal{Y}|\mathcal{G})$. Most existing studies focus on maximizing $P(\mathcal{Y}|\mathcal{G})$ yet few discuss the cause-and-effect relationship between $\mathcal{G}$ and $\mathcal{Y}$. In this paper, we study their causality based on the Structure Causal Model (SCM) [54].

Structure Causal Model. During data collection, it is inevitable that some unknown confounders $\mathcal{C}$ are included in datasets, which influence both $\mathcal{G}$ and $\mathcal{Y}$. For example, sensors might record random background noise, and the constructed sensor network might contain shortcut edges. The undesired information might bridge the input $\mathcal{G}_M$ and the output $\mathcal{Y}$ with spurious correlations, which could lead to overfitting and make the model error-prone.

The causal relationship between $\mathcal{G}, \mathcal{H}, \mathcal{Y}$ and $\mathcal{C}$ can be modeled by the Structure Causal Model (SCM) [54] shown in Figure 1. First, it is evident that $\mathcal{C}$ and $\mathcal{H}$ are not d-separable [52], since $\mathcal{C}$ can reach $\mathcal{H}$ via the path $\mathcal{C} \to \mathcal{G} \to \mathcal{H}$. This means that $\mathcal{H}$ and $\mathcal{C}$ are not independent and thus $\mathcal{H}$ contains information of $\mathcal{C}$. Second, besides $\mathcal{G} \to \mathcal{H} \to \mathcal{Y}$, $\mathcal{C}$ introduces backdoor paths between $\mathcal{G}$ and $\mathcal{Y}$, as well as $\mathcal{H}$ and $\mathcal{Y}$: $\mathcal{G} \leftarrow \mathcal{C} \to \mathcal{Y}, \mathcal{H} \leftarrow \mathcal{G} \leftarrow \mathcal{C} \to \mathcal{Y}$. The model f might take advantage of the backdoor paths to make decisions instead of struggling to discover the real cause-and-effect relationships [65, 81]. Our goal is to eliminate the backdoor paths.

Frontdoor Adjustment. In statistics [52], a simple way to exclude the variable $\mathcal{C}$ in the SCM in Figure 1 is to marginalize it out. However, marginalization requires $\mathcal{C}$ to be observable and measured by the marginal distribution $P(\mathcal{C})$, but in spatiotemporal imputation, $\mathcal{C}$ is usually unknown and difficult to measure. Rather than directly marginalizing $\mathcal{C}$ out, we resort to the frontdoor adjustment [54], which uses Pearl's do-calculus [54] to block the backdoor paths. We follow the three steps of the frontdoor adjustment as follows.

(1) *Remove the backdoor path from $\mathcal{G}$ to $\mathcal{H}$.* Given $\mathcal{G}$, there is no backdoor path from $\mathcal{G}$ to $\mathcal{H}$. Note that $\mathcal{G}$ cannot reach $\mathcal{H}$ via $\mathcal{G} \leftarrow \mathcal{C} \to \mathcal{Y} \leftarrow \mathcal{H}$ according to the d-separation theory [52]. Therefore, we have:

$$P(\mathcal{H}|do(\mathcal{G})) = P(\mathcal{H}|\mathcal{G}). \tag{3}$$

(2) *Remove the backdoor path from $\mathcal{H}$ to $\mathcal{Y}$.* There is a backdoor path between $\mathcal{H}$ and $\mathcal{Y}$: $\mathcal{H} \leftarrow \mathcal{G} \leftarrow \mathcal{C} \to \mathcal{Y}$. This backdoor path can be blocked by marginalizing out $\mathcal{G}$:

$$P(\mathcal{Y}|do(\mathcal{H})) = \sum_{\mathbf{G}} P(\mathcal{Y}|\mathcal{H}, \mathbf{G}_M)P(\mathbf{G}). \tag{4}$$

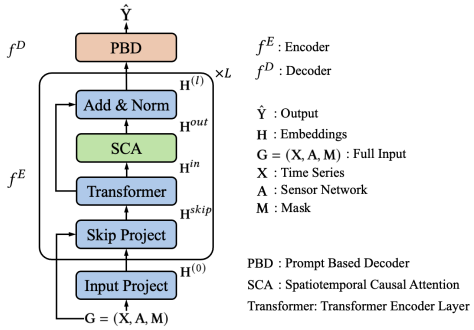


Figure 2: Overview of Casper.

(3) *Combine the results of the above two steps:*

$$\begin{aligned} P(\mathcal{Y}|do(\mathcal{G})) &= \sum_{i,t} P(\mathbf{h}_{i,j}|do(\mathcal{G}))P(\mathcal{Y}|do(\mathbf{h}_{i,t})) \\ &= \sum_{i,t} P(\mathbf{h}_{i,t}|\mathcal{G}) \sum_{\mathbf{G}} P(\mathcal{Y}|\mathbf{h}_{i,t}, \mathbf{G})P(\mathbf{G}). \end{aligned} \tag{5}$$

In general, $P(\mathbf{h}_{i,t}|\mathcal{G})$ can be viewed as the encoder f^E, and the rest part, including $\sum_{\mathbf{G}} P(\mathcal{Y}|\mathbf{h}_{i,t}, \mathcal{G})P(\mathbf{G})$ and the sum over all data points $\sum_{i,t}$, can be viewed as the decoder f^D. In the next subsection, we show how to implement Equation (5).

3.2 Architecture of Casper

In this subsection, based on Equation (5), we propose a novel Causality-Aware Spatiotemporal Graph Neural Network (Casper). We first present an overview of Casper, consisting of a Prompt Based Decoder (PBD) and an encoder with Spatiotemporal Causal Attention (SCA). Next, we elaborate PBD and SCA in detail.

Overview. Figure 2 shows an overview of Casper. The encoder f^E is comprised of an *input project* and L layers of the combination of *skip project*, *transformer*, *SCA* and *add & norm*. Let $\mathbf{m} \in \mathbb{R}^d$ be the embedding for the missing points. The *input project* module encodes the raw input $\mathbf{G} = (\mathbf{X}, \mathbf{A}, \mathbf{M})$ into $\mathbf{H}^{(0)}$ via:

$$\mathbf{H}^{(0)} = \text{MLP}(\mathbf{X}) \odot \mathbf{M} + \mathbf{m} \odot (1 - \mathbf{M}), \tag{6}$$

where MLP stands for Multi-Layer Perceptron and $\odot$ denotes the Hadamard product. The *skip project* module prevents gradient vanishing and improves the performance by injecting the $\mathbf{G}$ into the embeddings from the previous layer $\mathbf{H}^{(l-1)}$:

$$\mathbf{H}^{skip} = \mathbf{H}^{(l-1)} + \text{MLP}(\mathbf{X}) \odot \mathbf{M} + \mathbf{m} \odot (1 - \mathbf{M}). \tag{7}$$

The *transformer* encoder layer [68] learns temporal information for each time series within $\mathbf{H}^{skip}$:

$$\mathbf{H}^{in} = \text{Transformer}(\mathbf{H}^{skip}). \tag{8}$$

SCA discovers spatiotemporal causal relationships among embeddings based on $\mathbf{A}$, and encodes causal information into embeddings:

$$\mathbf{H}^{out} = \text{SCA}(\mathbf{H}^{in}, \mathbf{A}). \tag{9}$$

The final embeddings of the l-th layer are given by:

$$\mathbf{H}^{(l)} = \text{LayerNorm}(\mathbf{H}^{in} + \mathbf{H}^{out}). \tag{10}$$

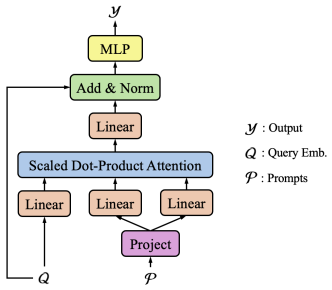


Figure 3: Prompt Based Decoder (PBD).

Given the embeddings $\mathbf{H} = \mathbf{H}^{(L)}$ obtained by the encoder f^E, the *PBD* module in f^D generates the predictions $\hat{\mathbf{Y}} = \{\hat{y}_{i,t}\}_{i=1,t=1}^{N,T}$:

$$\hat{\mathbf{Y}} = \text{PBD}(\mathbf{H}). \tag{11}$$

Prompt Based Decoder. Suppose we are given the input $\mathbf{G}$ and the target is to recover $\mathcal{Y} = y_{i,t}$. In Equation (5), the decoder f^D is comprised of (1) a sum over all possible $\mathbf{G}'$ in the dataset $\sum_{\mathbf{G}'} P(y_{i,t}|\mathbf{h}_{i',t'}, \mathbf{G}')P(\mathbf{G}')$ and (2) a sum over all data points $\sum_{i',t'}$ in $\mathbf{G}$. For (2), since the encoders nowadays, e.g., Transformer [68], are so powerful that could encode sufficient context information $\mathbf{G}$ in $\mathbf{h}_{i,t}$, and thus the decoder f^D could only take $\mathbf{h}_{i,t}$ as input [50], instead of all possible $\mathbf{h}_{i',t'}$. Therefore, we could drop $\sum_{i',t'}$ and only implement $\sum_{\mathbf{G}'} P(y_{i,t}|\mathbf{h}_{i,t}, \mathbf{G}')P(\mathbf{G}')$.

Now, the challenge is how to implement the sum over $\mathbf{G}'$. Simple solutions include randomly sampling a set from the training data, or clustering $\mathbf{G}'$ into K clusters and using the cluster centers as an approximation. However, random sampling could be unstable in practice, and clustering requires extra pre-trained models to extract embeddings of $\mathbf{G}'$ in advance. Inspired by [26], which uses prompts to capture the context information of the downstream task, we introduce a Prompt Based Decoder (PBD) to automatically capture the global context information of the dataset during model training.

An illustration of the proposed PBD is shown in Figure 3. $\mathcal{Q} = \mathbf{h}_{i,t}$ is the query, $\mathcal{P} = \{\mathbf{p}_n\}_{n=1}^{N_P}$ is the set of learnable prompts, which are randomly initialized embedding vectors. In Figure 3, the *Project* is a linear function followed with a LayerNorm [4]. Details of the scaled dot-product attention can be found in [68], and it can be easily extended into the multi-head version as in [68].

Spatiotemporal Causal Attention. Attention functions (Definition 4) have become indispensable in deep learning models [27, 68], which could effectively capture the context information for the target embedding. Although attention scores could show the correlation between embeddings, correlation does not necessarily imply causality and thus sometimes could induce undesired non-causal information into embeddings [65]. Based on the frontdoor adjustment, PBD could eliminate the impact of unknown confounders $\mathcal{C}$ and ensure the causality between the input $\mathcal{G}$ and the output $\mathcal{Y}$ by summing over $\mathbf{G}$, i and t. However, it guides the attention functions to discover the causal relationships at a high level, and thus the learned causal relationships, i.e., attention scores, might still be dense and a little bit difficult to interpret (see Figuers 5c and 5g). To directly guide the model to discover the sparse causal relationships,

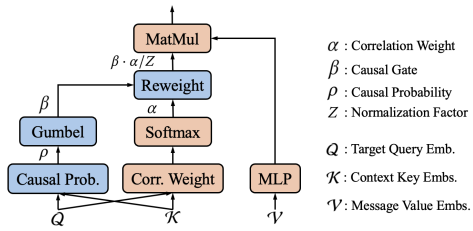


Figure 4: Spatiotemporal Causal Attention (SCA). Corr. Weight and Causal Prob. correspond to Equation (14)(16)**.**

we first define the causality for embeddings (Definition 5) based on the Granger causality [19] (Definition 3), and then introduce a novel Spatiotemporal Causal Attention (SCA) module to discover the sparse causality between embeddings.

Definition 5 (Unconstrained Granger Causality for Embeddings)**.** Denote the target embedding as $\mathcal{Q} = \mathbf{h}_{i,t}^{in}$ and the set of context embeddings as $\mathcal{K} = \{\mathbf{h}_{i',t'}^{in}\}_{i'=1,t'=1}^{N,T}$. Let $\mathbf{h}_{i,t}^{out} = f_{i,t}(\mathcal{Q}; \mathcal{K})$ be an embedding updating function, e.g., attention function. If there is a $\mathbf{h}_{i',t'}^{in} \in \mathcal{K}$ s.t. changing the value of $\mathbf{h}_{i',t'}^{in}$ will change the value of $\mathbf{h}_{i,t}^{out}$, then $\mathbf{h}_{i',t'}^{in}$ Granger causes $\mathbf{h}_{i,t}^{out}$.

We do not strictly enforce the time $t' \le t$ for the imputation task as for the forecasting task (Definition 3), since (1) a missing value could appear at the beginning of the input time series segment, and there are no prior points available; (2) most imputation methods in the literature consider both past $t' \le t$ and future $t' > t$ reference points $\mathbf{h}_{i',t'}$ for the data point to be imputed $\mathbf{h}_{i,t}$; (3) given the learned weight $w_{i',t'}$ between $\mathbf{h}_{i',t'}$ and $\mathbf{h}_{i,t}$, it is easy to distinguish whether it is from the past or future by comparing t' and t, and thus we can easily obtain the time-constrained causal graph if necessary.

Let $f_{i,t}$ be an attention function as shown in Definition 4:

$$\mathbf{h}_{i,t}^{out} = \sum_{i'=1,t'=1}^{N,T} \alpha_{i',t'} \mathbf{v}_{i',t'}, \quad \mathbf{v}_{i',t'} = f_v(\mathbf{h}_{i,t}^{in}, \mathbf{h}_{i't'}^{in}), \tag{12}$$

where $\alpha_{i',t'}$ is the attention weight, and f_v is the message function. According to Definition 5, if $\alpha_{i',t'} \neq 0$, then $\mathbf{h}_{i',t'}^{in}$ Granger causes $\mathbf{h}_{i,t}^{out}$; otherwise $\mathbf{h}_{i',t'}^{in}$ does not Granger cause $\mathbf{h}_{i,t}^{out}$. In practice, without directly manipulating, $\alpha_{i',t'} > 0$ holds for many noisy messages $\mathbf{v}_{i',t'}$, as shown in Figure 5c-5d in our experiments. To further enforce the weights of the noisy points to be zero and discover Granger causality, we propose a novel SCA. As shown in Figure 4, SCA is comprised of two components: (1) a spatiotemporal graph attention function (orange), which learns the correlation between embeddings; (2) a causal gate (blue), which discovers the causal and non-causal relationships.

Let $\mathcal{Q} = \mathbf{h}_{i,t}^{in}$ be the target query embedding, and $\mathcal{N}(i)$ be the neighbors of the i-th sensor, i.e., $\mathbf{A}[i, i'] \neq 0, \forall i' \in \mathcal{N}(i)$. Denote the context keys as $\mathcal{K} = \{\mathbf{h}_{i',t'}^{in}\}_{i',t'}$ and the message values from $\mathcal{N}(i)$ to the point (i, t) as $\mathcal{V} = \{\mathbf{v}_{i',t'}\}_{i',t'}$, where $i' \in \mathcal{N}(i)$, $t' \in \{1, \cdots, T\}$ and $\mathbf{v}_{i',t'} = \text{MLP}([\mathbf{h}_{i,t}^{in}; \mathbf{h}_{i',t'}^{in}])$. We define SCA as:

$$\mathbf{h}_{i,t}^{out} = \frac{1}{Z} \sum_{i' \in \mathcal{N}(i)} \sum_{t'=1}^{T} \beta_{i',t'} \alpha_{i',t'} \mathbf{v}_{i',t'}, \tag{13}$$

where $Z \in \mathbb{R}$ is a normalization factor, $\alpha_{i',t'}$ is the correlation weight, $\beta_{i',t'} \sim \text{Bernoulli}(\rho_{i',t'})$ is the causal gate, and $\rho_{i',t'}$ is the probability that $\mathbf{h}^{in}_{i',t'}$ Granger causes $\mathbf{h}^{out}_{i,t}$. According to Definition 5, if $\beta_{i',t'} \cdot \alpha_{i',t'} > 0$, then $\mathbf{h}^{in}_{i',t'}$ Granger causes $\mathbf{h}^{out}_{i,t}$; otherwise, there is no Granger causality between $\mathbf{h}^{in}_{i',t'}$ and $\mathbf{h}^{out}_{i,t}$. The function of calculating correlation weight $\alpha_{i',t'}$ is given by:

$$\alpha_{i't'} = \frac{\exp(s(\mathbf{h}_{i,t}; \mathbf{h}_{i',t'}))}{\sum_{j \in \mathcal{N}(i)} \sum_{r=1}^{T} \exp(s(\mathbf{h}_{i,t}; \mathbf{h}_{j,r}))}, \tag{14}$$

$$s(\mathbf{h}^{in}_{i,t}; \mathbf{h}^{in}_{i',t'}) = (\mathbf{W}_Q \mathbf{h}^{in}_{i,t})^T (\mathbf{W}_K \mathbf{h}^{in}_{i',t'})/\sqrt{d}, \tag{15}$$

where $\mathbf{W}_Q, \mathbf{W}_K \in \mathbb{R}^{d \times d}$ are learnable weights, and d is the size of the hidden dimension. We build a neural network to learn the probability $\rho_{i',t'}$ of the causal gate $\beta_{i',t'}$:

$$\rho_{i',t'} = \sigma(\mathbf{W}_c[\mathbf{W}_{Q_c} \mathbf{h}^{in}_{i,t}; \mathbf{W}_{K_c} \mathbf{h}^{in}_{i',t'}]), \tag{16}$$

where $\mathbf{W}_c \in \mathbb{R}^{1 \times 2d}, \mathbf{W}_{Q_c}, \mathbf{W}_{K_c} \in \mathbb{R}^{d \times d}$ are learnable weights, and σ is the Sigmoid activation function.

There are two practical issues of directly using ρ in the above equation. First, the sampling operation $\beta_{i',t'} \sim \text{Bernoulli}(\rho_{i',t'})$ is in-differentiable. To address this issue, we use the differentiable reparameterization technique Gumbel-Softmax [25] to obtain $\beta_{i',t'}$:

$$\beta_{i',t'} = \frac{\exp((\log \rho_{i',t'} + g)/\tau)}{\exp((\log \rho_{i',t'} + g)/\tau) + \exp((\log(1 - \rho_{i',t'}) + g)/\tau)}, \tag{17}$$

where $g = -\log(-\log(u))$, $u \sim \text{Uniform}(0, 1)$, and τ is the temperature parameter.

Second, if $\rho_{i',t'}$ is not close to 0 or 1, the model's decision could be ambiguous during inference. For example, if $\rho_{i',t'} = 0.2$, then for the same input data, for 20% time, the model shows $\mathbf{h}^{in}_{i',t'}$ Granger causes $\mathbf{h}^{out}_{i,t}$, and for the other 80% time, the model shows $\mathbf{h}^{in}_{i',t'}$ does not Granger cause $\mathbf{h}^{out}_{i,t}$. To avoid such an ambiguous situation, we enforce $\rho_{i',t'} \to 0/1$ by placing the l_1 regularization over $\rho_{i',t'}$.

It can be theoretically proven that $\rho_{i',t'}$ will converge to 0 or 1 (see Section 3.3). Additionally, in practice, the correlation weight α can be easily extended to the multi-head version as in [68].

Loss Function. Casper is trained by the masked MAE. For a given spatiotemporal time series segment with N nodes and T length, the loss is defined as:

$$\mathcal{L} = \sum_{i=1,t=1}^{N,T} m_{i,t} \cdot |y_{i,t} - \hat{y}_{i,t}| + \lambda ||\Phi||_1, \tag{18}$$

where $m_{i,t}$ is the mask, $y_{i,t}$ is the ground-truth value, $\hat{y}_{i,t}$ is the predicted value, Φ is the set of all ρ in SCA, λ is a tunable coefficient, and $||\cdot||_1$ denotes the l_1 norm.

3.3 Framework Analysis

In this subsection, we provide further analysis of the proposed Casper, including theoretical analysis and complexity analysis.

Theoretical Analysis. We theoretically prove that $\rho_{i',t'}$ in Equation (16) will converge to 0 or 1 in Theorem 1, and thus $\rho_{i',t'}$ indicates the Granger causality. If $\rho_{i',t'} = 0$, then $\beta_{i',t'} \cdot \alpha_{i',t'} = 0$, showing that $\mathbf{h}^{in}_{i',t'}$ does not Granger cause $\mathbf{h}^{out}_{i,t}$. Moreover, from the proof of Theorem 1, it can be observed that $\rho_{i',t'}$ is actually a gradient-based explanation (see Remark 1), which determines causal and non-causal relationships based on the gradients. Compared with the classic gradient-based explanation methods [61], which needs extra steps to calculate gradients after the model is trained, the proposed ρ has two advantages: (1) ρ does not require extra steps for calculating derivatives, and the value of ρ could directly provide the explanation; (2) the parameters associated with ρ are jointly trained with the model, and thus it can guide the model to focus on the most important relationships during training.

Theorem 1 (Convergence of ρ)**.** *ρ could converge to 0 or 1 by updating its parameters based on $\mathcal{L}$ in Equation (18).*

Proof Sketch. For simplicity, let's only consider the loss for a single point (i, t), where $m_{i,t} = 1$ and $y_{i,t} - \hat{y}_{i,t} > 0$:

$$\mathcal{L}_{i,t} = y_{i,t} - \hat{y}_{i,t} + \lambda ||\Phi||_1 \tag{19}$$

Since Equation (16) is essentially a linear function with a Sigmoid activation, we can rewrite it as:

$$\rho_{i',t'} = \sigma(\mathbf{w}^T \mathbf{h}^{in}), \quad \mathbf{h}^{in} = [\mathbf{h}^{in}_{i,t}; \mathbf{h}^{in}_{i',t'}], \quad \mathbf{w} \in \mathbb{R}^{2d} \tag{20}$$

Let $w_j = \mathbf{w}[j]$ and $h^{in}_j = \mathbf{h}^{in}[j]$, then the gradient at w_j is:

$$\begin{aligned} \frac{\partial \mathcal{L}_{i,t}}{\partial w_j} &= \frac{\partial y_{i,t} - \hat{y}_{i,t}}{\partial \mathbf{h}^{out}_{i,t}} \cdot \frac{\partial \mathbf{h}^{out}_{i,t}}{\partial \beta_{i',t'}} \cdot \frac{\partial \beta_{i',t'}}{\partial \rho_{i',t'}} \cdot \frac{\partial \rho_{i',t'}}{\partial w_j} + \lambda \frac{\partial \rho_{i',t'}}{w_j} \\ &= (\lambda - g) \rho_{i',t'} (1 - \rho_{i',t'}) h^{in}_j \end{aligned} \tag{21}$$

$$g = \frac{\partial \hat{y}_{i,t}}{\partial \mathbf{h}^{out}_{i,t}} \cdot \frac{\partial \mathbf{h}^{out}_{i,t}}{\partial \beta_{i',t'}} \cdot \frac{\partial \beta_{i',t'}}{\partial \rho_{i',t'}} = \frac{\partial \hat{y}_{i,t}}{\partial \mathbf{h}^{out}_{i,t}} \cdot \frac{\alpha_{i',t'}}{Z} \mathbf{v}_{i',t'} \cdot 1 \tag{22}$$

where $\mathbf{h}^{out}_{i,t}, \beta_{i',t'}$ are from Equation (17). In gradient descent, the updating function of w_j is:

$$w^{(k+1)}_j = w^{(k)}_j - \eta(\lambda - g) \rho^{(k)}_{i',t'} (1 - \rho^{(k)}_{i',t'}) h^{in}_j \tag{23}$$

where k is the iteration index and $\eta > 0$ is the learning rate. As we only consider the parameters $\mathbf{w}$ of ρ, for simplicity, let's fix all other parameters in the model. Therefore, $\partial \hat{y}_{i,t} / \partial \mathbf{h}^{out}_{i,t}, \mathbf{v}_{i',t'}, \alpha_{i',t'}$ are fixed. The normalization factor $Z \leq 1$, and $Z = 1$ if and only if $\beta_{i',t'} = 1$ for $\forall i', t'$. $\rho_{i',t'}(1 - \rho_{i',t'}) > 0$ since $\rho_{i',t'} \in (0, 1)$. Now, suppose $g > \lambda$, if $h^{in}_j > 0$, then $w^{(k+1)}_j > w^{(k)}_j$. When $k \to \infty$, $w^{(k)}_j \to +\infty$. Otherwise, if $h^{in}_j < 0$, then $w^{(k+1)}_j < w^{(k)}_j$. When $k \to \infty$, $w^{(k)}_j \to -\infty$. Therefore, when $k \to \infty$, $\mathbf{w}^T \mathbf{h} \to +\infty$. As a result, $\rho^{(k)}_{i',t'} = \sigma(\mathbf{w}^{(k)^T} \mathbf{h}^{in}) \to 1$, as $k \to \infty$. Similarly, if $g < \lambda$, then $\rho^{(k)}_{i',t'}$ will converge to 0. □

Remark 1 (ρ is a gradient based explanation)**.** From the proof of Theorem 1, it can be noted that, if $g > \lambda$, then $\rho_{i',t'} \to 1$; if $g < \lambda$, then $\rho_{i',t'} \to 0$. This phenomenon reflects that $\rho_{i',t'}$ serves as a binary indicator showing whether the gradient at $\rho_{i',t'}$ is greater or less than the threshold λ.

Complexity Analysis. The complexities of the input project, the skip project, and the transformer layer are $O(NT)$, $O(NT)$, and $O(NT^2)$. The complexities of SCA and PBD are $O(ET)$ and $O(NN_PT)$, where E is the number of edges in the sensor network, and N_P is the number of prompts. The overall complexity is $O(\max(E, NT, NN_P)T)$. In SCA, if we calculate attention weights and causal gates for each pair of the data points without using $\mathbf{A}$, then the complexity will be extremely high: $O(N^2T^2) \gg O(ET)$.

4 Experiments

4.1 Experimental Setup

In this subsection, we briefly explain the datasets, evaluation metrics, and baselines used for the experiments.

Datasets. Three public real-world benchmark datasets are used to evaluate Casper. **AQI** [94] is an hourly record of air pollutants from 437 air quality monitoring stations in China from May 2014 to April 2015. We also use the popular **AQI-36** [7] which is a reduced version of the full AQI containing records from 36 sensor stations scattered around Beijing. **METR-LA** [43] contains traffic speed time series collected from 207 sensors on highways in Los Angeles for 4 months. **PEMS-BAY** [43] is a traffic speed time series collected from 325 sensors on highways in San Francisco Bay Area for 6 months. The time series records in METR-LA and PEMS-BAY are collected every 5 minutes. For AQI, METR-LA, and PEMS-BAY, the temporal window is set as $T = 24$, and for AQI-36, the temporal window is 36. To be consistent with prior works [50], the adjacency matrices of sensor networks are built by applying a thresholded Gaussian kernel [43, 64] over the geographical distances between sensor stations.

For AQI and AQI-36, we use the evaluation masks in [50, 82] which simulates the real missing data distribution in the datasets. We refer to this setting as the general missing. For METR-LA and PEMS-BAY, we consider both points missing and block missing settings as in [50]. In point missing, 25% data points are masked out. In block missing, 5% spatial blocks and 0.15% temporal blocks ranging from 1 hour to 4 hours are masked out.

Evaluation Metrics. We use the standard Mean Absolute Error (MAE) and Mean Squared Error (MSE) between the ground truth values and the imputed values as the evaluation metrics.

Baselines. We consider three groups of methods as our baselines. (1) Traditional statistical methods: the mean value of the sequence (MEAN); neighbor mean (KNN); matrix factorization (MF); multiple imputations using chained equations (MICE [72]); vector auto-regression (VAR); (2) Early deep learning models: rGAIN [10]: an adversarial method similar to [45, 51]; BRITS [7]: a bidirectional RNN imputation method; (3) Recent deep learning models: ST-Transformer [50]: a spatiotemporal extension of the original Transformer [68]; GRIN [10]: a graph enhanced recurrent neural network; SPIN [50]: a spatiotemporal graph attention based imputation model; PoGeVon [70]: a recent spatiotemporal imputation method which is based on the position-aware spatiotemporal graph variational auto-encoder. Whenever possible, the results of baselines are copied from the corresponding paper.

Implementation Details. Most of the training configurations follow prior works [50]. We set the size of embeddings as 32. The numbers of layers for the encoder for AQI-36 and other datasets are 2 and 4 respectively. We use the Adam optimizer [38] with a learning rate of 0.0008 and a cosine scheduler to train the model. The maximum number of epochs is 300 and the patience of early stopping is 40 epochs. Batch size is fixed as 8. During training, $p \in [0.2, 0.5, 0.8]$ data points are randomly masked out for each batch, and the loss is calculated based on these masked points.

4.2 Overall Performance

In this subsection, we show the overall performance (MSE and MAE) of Casper for the imputation task to demonstrate the overall competence of Casper for imputation.

The overall performance of different methods is presented in Table 1. The upper, middle, and lower groups of baselines are the traditional statistical methods, the RNN methods and the recent methods (Transformer based and graph based methods). Generally speaking, the RNN methods perform better than the statistical methods, and the recent methods further outperform the RNN methods. When imputing a data point, these methods exploit all the available information in the context, without identifying the causal relationships between the data point and the context. However, it is inevitable that some confounders are included in the data, such as the non-causal shortcut edges. Over-reliance on the confounders could lead to overfitting and make the model susceptible to noise. The proposed Casper could effectively remove the impact of confounders. As shown in Table 1, Casper achieves the lowest overall MAE and MSE scores and also has lower standard deviations, demonstrating the effectiveness of enforcing the model to discover causality during training.

4.3 Ablation Study

In this subsection, we study the impact of different components in Casper, and the results are shown in Table 2.

Effectiveness of PBD,SCA and **A**. In the upper part of Table 2, we investigate the impact of the Prompt-Based Decoder (PBD), the Spatiotemporal Causal Attention (SCA) and the sensor network **A**. First, a significant performance drop on MAE and MSE can be observed when we remove PBD (replace PBD with an MLP) and/or SCA (remove the causal gate β) from the full model Casper, indicating the effectiveness of the frontdoor adjustment and the causal gate for improving the overall performance. It is also worth noting that, on MSE, the standard deviations of the ablated versions of Casper (w/o PBD and/or SCA) are significantly higher than the full model Casper, demonstrating that enforcing the model to focus on the causality could improve its robustness. Second, although **A** contains non-causal relationships among sensors, it still has critical contributions to the overall performance, demonstrating the necessity of considering **A** when imputing missing values.

Effectiveness of the Prompts. In the middle part of Table 2, we demonstrate the effectiveness of using the learnable prompts to capture the global contextual information of the datasets by replacing the prompts with other approximations, including K-means cluster centers and randomly sampled data. To obtain the other two approximations, we first pre-train an imputation model, i.e., Casper without SCA and PBD, and then extract the embedding of each training sample **G** by applying average/max pooling over the embeddings of all the points in **G**. For the cluster center approximation, we apply K-means over the embeddings to obtain 1,000 cluster centers. For the sampling approximation, we randomly sample 1,000 embeddings for each training sample. Compared with the prompts, the cluster centers and randomly sampled embeddings not only perform worse but also require extra effort to obtain embeddings, which demonstrates the superiority of prompts.

Table 1: Performance (MAE, MSE) of different methods.

	General Missing				Point Missing				Block Missing			
	AQI-36		AQI		METR-LA		PEMS-BAY		METR-LA		PEMS-BAY	
	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
Mean	53.48±0.00	4578.08±00.00	39.60±0.00	3231.04±00.00	7.56±0.00	142.22±0.00	5.42±0.00	86.59±0.00	7.48±0.00	139.54±0.00	5.46±0.00	87.56±0.00
KNN	30.21±0.00	2892.31±00.00	34.10±0.00	3471.14±00.00	7.88±0.00	129.29±0.00	4.30±0.00	49.80±0.00	7.79±0.00	124.61±0.00	4.30±0.00	49.90±0.00
MF	30.54±0.26	2763.06±63.35	26.74±0.24	2021.44±27.98	5.56±0.03	113.46±1.08	3.29±0.01	51.39±0.64	5.46±0.02	109.61±0.78	3.28±0.01	50.14±0.13
MICE	30.37±0.09	2594.06±07.17	26.98±0.10	1930.92±10.08	4.42±0.07	55.07±1.46	3.09±0.02	31.43±0.41	4.22±0.05	51.07±1.25	2.94±0.02	28.28±0.37
VAR	15.64±0.08	833.46±13.85	22.95±0.30	1402.84±52.63	2.69±0.00	21.10±0.02	1.30±0.00	6.52±0.01	3.11±0.08	28.00±0.76	2.09±0.10	16.06±0.73
rGAIN	15.37±0.26	641.92±33.89	21.78±0.50	1274.93±60.28	2.83±0.01	20.03±0.09	1.88±0.02	10.37±0.20	2.90±0.01	21.67±0.15	2.18±0.01	13.96±0.20
BRITS	14.50±0.35	662.36±65.16	20.21±0.22	1157.89±25.66	2.34±0.00	16.46±0.05	1.47±0.00	7.94±0.03	2.34±0.01	17.00±0.14	1.70±0.01	10.50±0.07
ST-Transformer	11.98±0.53	557.22±46.52	18.11±0.25	1135.46±89.27	2.16±0.00	13.66±0.03	0.74±0.00	1.96±0.03	3.54±0.00	52.22±0.99	1.70±0.02	20.37±0.43
GRIN	12.08±0.47	523.14±57.17	14.73±0.15	775.91±28.49	1.91±0.00	10.41±0.03	0.67±0.00	1.55±0.01	2.03±0.00	13.26±0.05	1.14±0.01	6.60±0.10
SPIN	11.77±0.54	455.53±12.27	13.92±0.15	773.60±26.64	1.90±0.01	18.47±0.31	0.70±0.01	1.91±0.01	1.98±0.01	18.47±0.31	1.06±0.02	7.42±0.16
PoGeVon	10.92±0.24	493.94±51.89	14.18±0.04	740.57±8.01	1.96±0.01	11.08±0.05	0.67±0.01	**1.51±0.03**	1.95±0.01	13.08±0.08	1.54±0.02	17.18±0.48
Casper	**10.09±0.13**	**396.16±12.94**	**13.30±0.06**	**658.07±4.88**	**1.84±0.00**	**9.99±0.01**	**0.65±0.00**	1.63±0.01	**1.92±0.01**	**11.98±0.23**	**1.00±0.00**	**5.37±0.04**

Table 2: Ablation study on the AQI-36 dataset.

	MAE	MSE
Casper	**10.09±0.13**	**396.16±12.94**
w/o PBD (i.e., PBD→MLP)	10.36±0.11	445.03±29.03
w/o SCA (i.e., w/o β)	10.45±0.17	426.64±34.41
w/o PBD, SCA	10.84±0.20	472.24±42.77
w/o PBD, SCA, $\mathbf{A}$	14.86±0.21	767.96±25.32
Prompts → K-means Centers (max)	10.18±0.14	427.20±28.84
Prompts → K-means Centers (avg)	10.23±0.15	421.43±19.40
Prompts → Sampling (max)	10.25±0.18	421.78±33.26
Prompts → Sampling (avg)	10.56±0.19	474.27±50.55
unconstrained→constrained causality	10.83±0.19	477.85±45.62
w/o skip project	10.35±0.21	438.02±28.38

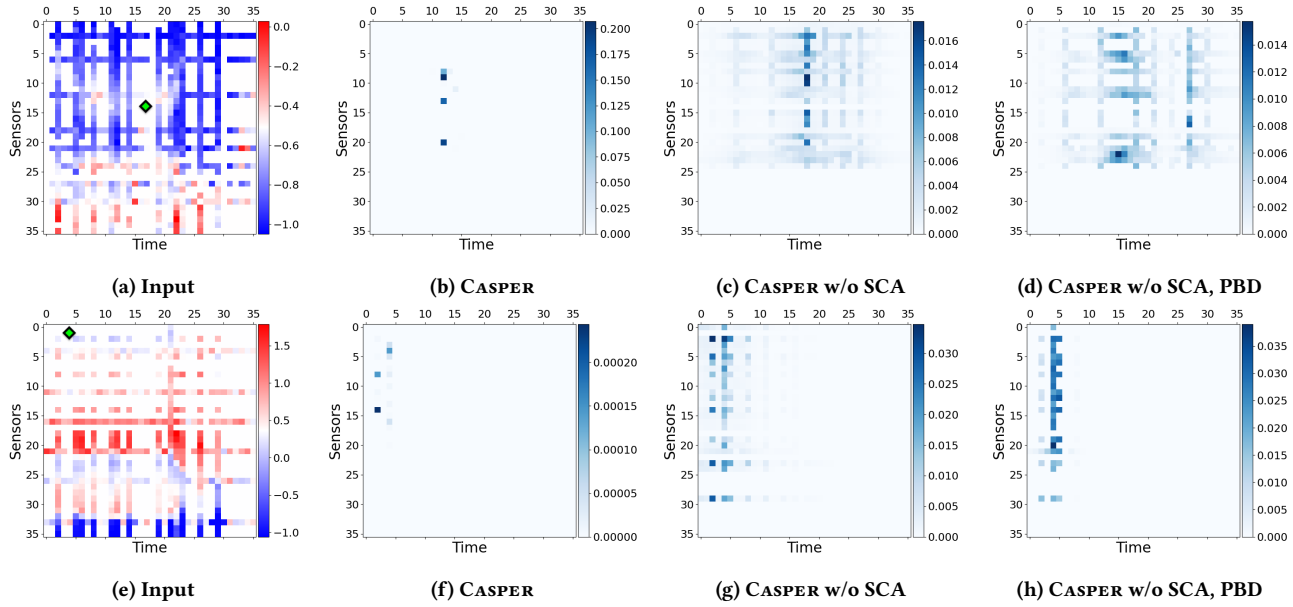
Effectiveness of Other Components. In the lower part of Table 2, we study the impact of time constraints and the skip project layer in the model. First, if we enforce the time constraint, i.e., $t' \leq t$ in Definition 5, for the imputation task, then the performance will significantly drop. These results demonstrate that it is vital to take into consideration both the past and future points for the imputation task. Second, removing the skip project layer will have negative impacts on the overall performance, showing the power of the skip project, which aligns with the observation of the residue connections in the literature [20, 33, 78].

4.4 Visualization

In this subsection, we provide visualization to further analyze Casper's ability of causality discovery.

Attention Maps. In Figure 5, we visualize two input time series from the test set of AQI-36, and their corresponding attention maps of the last encoder layer from different models, i.e., Casper, Casper w/o SCA (no causal gate) and Casper w/o SCA, PBD (no causal gate, PBD→MLP). Figure 5a and 5e are the inputs and the diamonds are the target query points. Other figures show the attention scores between the query point and all other points in the context. Figure 5b and 5f are the causal aware attention scores $\beta \cdot \alpha / Z$ in Equation (13). Figure 5c-5d and Figure 5g-5h are the attention scores α. First, by comparing the last two figures in each row, we can observe that the attention maps learned by Casper w/o SCA are sparser than Casper w/o SCA, PBD. This observation demonstrates that the frontdoor adjustment can effectively remove the noise and confounders by forcing the model to focus on only a small set of important points. Second, by comparing Casper and Casper w/o SCA, we can observe that SCA further improves the sparsity of attention by focusing on only a few points. Since l_1 norm is placed over the causal probabilities ρ during training, therefore, points with non-zero attention weights are critical for the query point, which cannot be removed. According to the Granger causality (Definition 5), these non-zero points are the causes for the query.

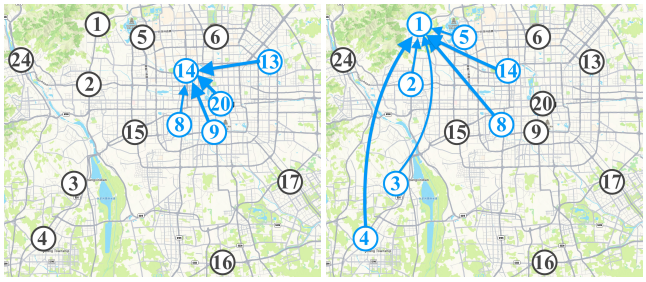
Discovered Causal Relationships. We draw the most salient causal relationships in Figure 5b and 5f on the map of Beijing in Figure 6a and 6b. Each number corresponds to a sensor and the arrow width corresponds to the attention weights. It is evident that for the query sensors 14 and 1 in the two figures, not every nearby sensor has a causal relationship with them. In Figure 6a, although the sensors 6, 5, 15 are very close to the query 14, Casper discovers that there is no causal relationship among them. We conjecture that this is because the wind blew generally westward and northward in Beijing for Figure 5b and 5f [89]. In Figure 6b, the neighboring sensors 24, 15, 6, 20, 9, 13 are not regarded as the causes for the query sensor 1. According to the spatial relationships, the information of the sensors 15, 6, 20, 9, 13 might be included in the causal sensors such as 2, 5 14, and 8, since the causal sensors are in between with the query and these non-causal sensors. However, there is no other sensor between 1 and 24. By taking a closer look at the upper-left corner of the map, we find that sensors 1 and 24 are separated by the Fragrant Hills. Therefore, the air quality at sensor 1 might not be directly influenced by sensor 24 for the period of the input data. The two examples in Figure 6 show that the proposed Casper could effectively discover the causal relationships among sensors, which provides better insights for further data analysis. Additionally, the two examples also show that the simple distance-based sensor network construction is biased, which contains many non-causal correlations since it ignores other factors in the real world, such as wind direction and terrain.



(a) Input (b) Casper (c) Casper w/o SCA (d) Casper w/o SCA, PBD

(e) Input (f) Casper (g) Casper w/o SCA (h) Casper w/o SCA, PBD

Figure 5: Input matrices and the associated attention maps of Casper and its ablated versions. Diamonds are the query points.



(a) Causal relations in Figure 5b. (b) Causal relations in Figure 5f.

Figure 6: Discovered causal relationships.

4.5 Causal Graph Discovery on the Quasi-Realistic Dataset

In the real world, the ground truth causal graphs are usually unavailable, and thus it is difficult to quantitatively evaluate the ability of causal graph discovery. Therefore, we follow [9] and evaluate our Casper on a quasi-realistic data called DREAM-3 [55], which is a gene expression data regulation dataset.

DREAM-3 contains $N = 100$ gene expression levels and the length of the expressions is $T = 21$. The goal is to discover the causal relationship among the 100 gene expression levels. For DREAM-3, we train Casper via the imputation task, and use similar training configurations as AQI-36. Following [9], we use AUC between the ground-truth graph and the discovered graph as the evaluation metric. Specifically, for Casper we obtain a causal weight matrix $\mathbf{C}_{i,t} \in \mathbb{R}^{N\times T}$ for each data point $x_{i,t}$, where $\mathbf{C}_{i,t}[i',t'] = \beta_{i',t'}$ in Equation 17. By concatenating all the $\mathbf{C}_{i,t}$ together, where $i \in \{1,\cdots,N\}$ and $t \in \{1,\cdots,T\}$, we have a tensor $C \in \mathbb{R}^{N\times T\times N\times T}$, where the first two dimensions correspond to the data point $x_{i,t}$, and the last two dimensions correspond to the causal weight $\mathbf{C}_{i,t}$ for $x_{i,t}$. We obtain the final causal weight matrix by max pooling over the two dimensions corresponding to the time, and normalizing by T^2: $\mathbf{A} = \text{max-pool}(C, \text{dim=2}, 4)/T^2 \in \mathbb{R}^{N\times N}$.

Table 3: Causal Graph Discovery on DREAM-3.

Models	PCMCI[60]	NGC[66]	eSRU[36]	LCCM[11]	NGM[5]	CUTS[9]	Casper
AUC	0.5517	0.5579	0.5587	0.5046	0.5477	0.5915	**0.6325**

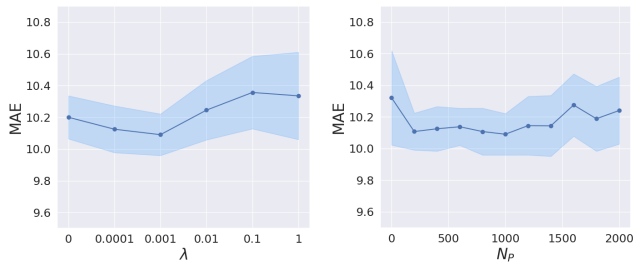
The results of Casper and several SOTA time series causal discovery baselines are presented in Table 3, where the results of baselines are copied from [9]. Table 3 shows that Casper achieves the highest AUC score. This experiment quantitatively demonstrates that Casper has a strong ability of discovering causal relationships.

4.6 More Results

In this subsection, we provide more experimental results of Casper, including convergence of ρ, and sensitivity analysis.

Convergence of ρ. We set 0.1 and 0.9 as the thresholds to round ρ. Specifically, if $\rho \le 0.1$ then we regard ρ has converged to 0; similarly, if $\rho \ge 0.9$, then we regard ρ has converged to 1. The statistical results of the AQI-36 dataset show that 98% ρ converges to 0 or 1, which corroborates Theorem 1.

Sensitivity Analysis. We present the results of sensitivity experiments of λ and the number of prompts N_P in Figure 7a-7b. For λ, the lowest MAE can be obtained when λ is around 0.001. For N_P, the best results can be obtained when $N_P \in [200, 1400]$.



(a) The weight of l_1 norm λ. **(b) The number of prompts N_P.**

Figure 7: Sensitivity experiments.

5 Related Work

In this section, we briefly review the most relevant works to ours, including spatiotemporal time series methods as well as causal inference methods.

5.1 Spatiotemporal Time Series Imputation

Spatiotemporal time series imputation is one of the fundamental tasks for time series analysis [14, 71, 93]. Traditional machine learning approaches are based on statistical analysis, such as linear autoregression [13, 82] and matrix factorization [84]. At present, deep learning methods have become the mainstream. Most existing deep learning methods are based on Recurrent Neural Networks (RNN). GRU-D [8] is one of the first RNN-based imputation models. BRITS [7] leverage bi-directional RNN to impute missing data. GAIN [45] and E2GAN [46] further apply Generative Adversarial Network (GAN) [18] to enhance the performance. mTAND [63] adds attention mechanism to RNN. These methods suffer from error propagation and accumulation brought by the auto-regression nature of RNN. To address this issue, non-autoregressive methods are proposed such as NAOMI [44], NRTSI [62] and the recent Transformer [68] based methods [83]. There are also some other types of methods, such as Ordinary Differential Equations (ODEs) methods [58, 58], state space models [2] and diffusion models [67] The above methods mainly focus on the temporal patterns of time series, yet largely ignore the spatial relationships, e.g., distance, among time series. To capture spatial relationships, graph neural networks [39, 69] are extended to the spatiotemporal setting. LG-ODE [24] combines graph neural networks with ODE methods [58]. RETIME [34] introduces a retrieval-based time series model, which leverages retrieved time series as an augmentation for the target time series. NET^3 [29] introduces a tensor graph neural network to model the high-order relationships among time series. GRIN [10] introduces a bidirectional message passing RNN with a spatial decoder. SPIN [50] presents a sparse spatiotemporal graph neural network for spatiotemporal time series imputation. Recently, PoGeVon [70] proposes a position-aware graph neural network based variational auto-encoder to impute both time series and edges. However, these methods try to exploit all the available related information for the target missing point, without distinguishing the causal and non-causal relationships, which might have the overfitting problem and make the model vulnerable to noise. Our proposed Casper could distinguish causal and non-causal relationships.

5.2 Causal Inference

Causality theory [54] provides theoretical guidance to design causality-aware models. It has been widely explored in the computer vision domain to discover causal relationships [6], generate counterfactual samples [1, 40, 85] and reduce bias [23, 56, 81]. In the graph mining domain, CGI [15] studies how to select trustworthy neighbors during inference; CLEAR [48] explores how to generate counterfactual explanations for graph-level prediction models based on Independent Component Analysis (ICA) [37]; HyperSCI [49] explores the Individual Treatment Effect (ITE) on hyper-graphs. NEAT [47] investigates the impact of MRSA infection via the Neyman-Rubin potential outcome framework [59]. CAL [65] introduces a causal attention learning framework for graph neural networks based on the backdoor adjustment [54]. There are two differences between Casper and the above graph methods: (1) our setting is the *dynamic* spatiotemporal setting but their setting is *static* graph; (2) Casper is based on the frontdoor adjustment and Granger causality, which is fundamentally different from their theoretical basis of causality. In the time series domain, the Granger causality [19] is widely used for analyzing the causality between time series in the forecasting setting. GrID-Net [73] leverages the Granger causality to infer regulatory locus–gene links. cLSTM [66] and economy-SRU [36] integrates the Granger causality with LSTM [21] and SRU [53]. However, these methods require the input time series to be fully observed. CUTS [9] is a recently proposed two-stage model, which first imputes missing data and then discovers causality between time series. There are three major differences between Casper and CUTS. (1) CUTS does not consider confounders, while Casper removes confounders via the frontdoor adjustment. (2) CUTS has to re-train the causal model for each input segment, but Casper does not have such a requirement. (3) CUTS is a two-stage model, while Casper is a one-stage end-to-end model.

6 Conclusion

In this paper, we review the spatiotemporal time series imputation task via the Structure Causal Model (SCM), which shows the causal relationships among the input, output, embeddings, and confounders. The confounders could open shortcut backdoor paths between the input and output, which could mislead the model to learn the non-causal relationships. We use the frontdoor adjustment to block the backdoor paths. Based on the results of the frontdoor adjustment, we propose a novel Causality-Aware Spatiotemporal Graph Neural Network (Casper), which is comprised of a Prompt Based Decoder (PBD) and an encoder equipped with Spatiotemporal Causal Attention (SCA). The proposed PBD could reduce the impact of the confounders at a high level. For SCA, we first extend the definition of Granger causality for time series to embeddings. Then we introduce the architecture of SCA based on the definition, which could discover the sparse causal relationships among embeddings. Theoretical analysis shows that SCA decides causal and non-casual relationships based on the values of gradients. Experimental results on three real-world benchmark datasets show that Casper could outperform the baseline methods for the imputation task. Further analysis shows that Casper could effectively discover the sparse causal relationships.

References

[1] Ehsan Abbasnejad, Damien Teney, Amin Parvaneh, Javen Shi, and Anton van den Hengel. 2020. Counterfactual vision and language learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 10044–10054.

[2] Juan Lopez Alcaraz and Nils Strodthoff. 2022. Diffusion-based Time Series Imputation and Forecasting with Structured State Space Models. *Transactions on Machine Learning Research* (2022).

[3] Gowtham Atluri, Anuj Karpatne, and Vipin Kumar. 2018. Spatio-temporal data mining: A survey of problems and methods. *ACM Computing Surveys (CSUR)* 51, 4 (2018), 1–41.

[4] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450* (2016).

[5] Alexis Bellot, Kim Branson, and Mihaela van der Schaar. 2021. Neural graphical modelling in continuous-time: consistency guarantees and algorithms. In *International Conference on Learning Representations*.

[6] Yoshua Bengio, Tristan Deleu, Nasim Rahaman, Nan Rosemary Ke, Sebastien Lachapelle, Olexa Bilaniuk, Anirudh Goyal, and Christopher Pal. 2019. A Meta-Transfer Objective for Learning to Disentangle Causal Mechanisms. In *International Conference on Learning Representations*.

[7] Wei Cao, Dong Wang, Jian Li, Hao Zhou, Lei Li, and Yitan Li. 2018. Brits: Bidirectional recurrent imputation for time series. *Advances in neural information processing systems* 31 (2018).

[8] Zhengping Che, Sanjay Purushotham, Kyunghyun Cho, David Sontag, and Yan Liu. 2018. Recurrent neural networks for multivariate time series with missing values. *Scientific reports* 8, 1 (2018), 6085.

[9] Yuxiao Cheng, Runzhao Yang, Tingxiong Xiao, Zongren Li, Jinli Suo, Kunlun He, and Qionghai Dai. 2022. CUTS: Neural Causal Discovery from Irregular Time-Series Data. In *The Eleventh International Conference on Learning Representations*.

[10] Andrea Cini, Ivan Marisca, and Cesare Alippi. 2021. Filling the G_ap_s: Multivariate Time Series Imputation by Graph Neural Networks. In *International Conference on Learning Representations*.

[11] Edward De Brouwer, Adam Arany, Jaak Simm, and Yves Moreau. 2020. Latent convergent cross mapping. In *International Conference on Learning Representations*.

[12] Boxin Du, Si Zhang, Yuchen Yan, and Hanghang Tong. 2021. New frontiers of multi-network mining: Recent developments and future trend. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 4038–4039.

[13] James Durbin and Siem Jan Koopman. 2012. *Time series analysis by state space methods*. Vol. 38. OUP Oxford.

[14] Philippe Esling and Carlos Agon. 2012. Time-series data mining. *ACM Computing Surveys (CSUR)* 45, 1 (2012), 1–34.

[15] Fuli Feng, Weiran Huang, Xiangnan He, Xin Xin, Qifan Wang, and Tat-Seng Chua. 2021. Should graph convolution trust neighbors? a simple causal inference method. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1208–1218.

[16] Shengyu Feng, Baoyu Jing, Yada Zhu, and Hanghang Tong. 2022. Adversarial graph contrastive learning with information regularization. In *Proceedings of the ACM Web Conference 2022*. 1362–1371.

[17] Shengyu Feng, Baoyu Jing, Yada Zhu, and Hanghang Tong. 2024. Ariel: Adversarial graph contrastive learning. *ACM Transactions on Knowledge Discovery from Data* 18, 4 (2024), 1–22.

[18] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. *Advances in neural information processing systems* 27 (2014).

[19] Clive WJ Granger. 1969. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica: journal of the Econometric Society* (1969), 424–438.

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[21] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[22] Timothy O Hodson. 2022. Root-mean-square error (RMSE) or mean absolute error (MAE): When to use them or not. *Geoscientific Model Development* 15, 14 (2022), 5481–5487.

[23] Xinting Hu, Kaihua Tang, Chunyan Miao, Xian-Sheng Hua, and Hanwang Zhang. 2021. Distilling causal effect of data in class-incremental learning. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*. 3957–3966.

[24] Zijie Huang, Yizhou Sun, and Wei Wang. 2020. Learning continuous system dynamics from irregularly-sampled partial observations. *Advances in Neural Information Processing Systems* 33 (2020), 16177–16187.

[25] Eric Jang, Shixiang Gu, and Ben Poole. 2016. Categorical Reparameterization with Gumbel-Softmax. In *International Conference on Learning Representations*.

[26] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. 2022. Visual prompt tuning. In *European Conference on Computer Vision*. Springer, 709–727.

[27] Baoyu Jing, Shengyu Feng, Yuejia Xiang, Xi Chen, Yu Chen, and Hanghang Tong. 2022. X-GOAL: Multiplex heterogeneous graph prototypical contrastive learning. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 894–904.

[28] Baoyu Jing, Chanyoung Park, and Hanghang Tong. 2021. Hdmi: High-order deep multiplex infomax. In *Proceedings of the Web Conference 2021*. 2414–2424.

[29] Baoyu Jing, Hanghang Tong, and Yada Zhu. 2021. Network of tensor time series. In *Proceedings of the Web Conference 2021*. 2425–2437.

[30] Baoyu Jing, Yansen Wang, Guoxin Sui, Jing Hong, Jingrui He, Yuqing Yang, Dongsheng Li, and Kan Ren. 2024. Automated Contrastive Learning Strategy Search for Time Series. *arXiv preprint arXiv:2403.12641* (2024).

[31] Baoyu Jing, Yuchen Yan, Kaize Ding, Chanyoung Park, Yada Zhu, Huan Liu, and Hanghang Tong. 2024. Sterling: Synergistic representation learning on bipartite graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 12976–12984.

[32] Baoyu Jing, Yuchen Yan, Yada Zhu, and Hanghang Tong. [n. d.]. Coin: Co-cluster infomax for bipartite graphs. In *NeurIPS 2022 Workshop: New Frontiers in Graph Learning*.

[33] Baoyu Jing, Zeyu You, Tao Yang, Wei Fan, and Hanghang Tong. 2021. Multiplex Graph Neural Network for Extractive Text Summarization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. 133–139.

[34] Baoyu Jing, Si Zhang, Yada Zhu, Bin Peng, Kaiyu Guan, Andrew Margenot, and Hanghang Tong. 2022. Retrieval based time series forecasting. *arXiv preprint arXiv:2209.13525* (2022).

[35] Wang-Cheng Kang and Julian McAuley. 2018. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*. IEEE, 197–206.

[36] Saurabh Khanna and Vincent YF Tan. 2019. Economy Statistical Recurrent Units For Inferring Nonlinear Granger Causality. In *International Conference on Learning Representations*.

[37] Ilyes Khemakhem, Diederik Kingma, Ricardo Monti, and Aapo Hyvarinen. 2020. Variational autoencoders and nonlinear ica: A unifying framework. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 2207–2217.

[38] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[39] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).

[40] Murat Kocaoglu, Christopher Snyder, Alexandros G Dimakis, and Sriram Vishwanath. 2018. CausalGAN: Learning Causal Implicit Generative Models with Adversarial Training. In *International Conference on Learning Representations*.

[41] Bolian Li, Baoyu Jing, and Hanghang Tong. 2022. Graph communal contrastive learning. In *Proceedings of the ACM web conference 2022*. 1203–1213.

[42] Jianbo Li, Lecheng Zheng, Yada Zhu, and Jingrui He. 2021. Outlier impact characterization for time series data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 11595–11603.

[43] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. 2018. Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting. In *International Conference on Learning Representations*.

[44] Yukai Liu, Rose Yu, Stephan Zheng, Eric Zhan, and Yisong Yue. 2019. Naomi: Non-autoregressive multiresolution sequence imputation. *Advances in neural information processing systems* 32 (2019).

[45] Yonghong Luo, Xiangrui Cai, Ying Zhang, Jun Xu, et al. 2018. Multivariate time series imputation with generative adversarial networks. *Advances in neural information processing systems* 31 (2018).

[46] Yonghong Luo, Ying Zhang, Xiangrui Cai, and Xiaojie Yuan. 2019. E2gan: End-to-end generative adversarial network for multivariate time series imputation. In *Proceedings of the 28th international joint conference on artificial intelligence*. AAAI Press Palo Alto, CA, USA, 3094–3100.

[47] Jing Ma, Chen Chen, Anil Vullikanti, Ritwick Mishra, Gregory Madden, Daniel Borrajo, and Jundong Li. 2023. A Look into Causal Effects under Entangled Treatment in Graphs: Investigating the Impact of Contact on MRSA Infection. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 4584–4594.

[48] Jing Ma, Ruocheng Guo, Saumitra Mishra, Aidong Zhang, and Jundong Li. 2022. Clear: Generative counterfactual explanations on graphs. *Advances in Neural Information Processing Systems* 35 (2022), 25895–25907.

[49] Jing Ma, Mengting Wan, Longqi Yang, Jundong Li, Brent Hecht, and Jaime Teevan. 2022. Learning causal effects on hypergraphs. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 1202–1212.

[50] Ivan Marisca, Andrea Cini, and Cesare Alippi. 2022. Learning to reconstruct missing data from spatiotemporal graphs with sparse observations. *Advances in Neural Information Processing Systems* 35 (2022), 32069–32082.

[51] Xiaoye Miao, Yangyang Wu, Jun Wang, Yunjun Gao, Xudong Mao, and Jianwei Yin. 2021. Generative semi-supervised learning for multivariate time series imputation. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35.

8983–8991.
[52] Kevin P Murphy. 2012. *Machine learning: a probabilistic perspective*. MIT press.
[53] Junier B Oliva, Barnabás Póczos, and Jeff Schneider. 2017. The statistical recurrent unit. In *International Conference on Machine Learning*. PMLR, 2671–2680.
[54] Judea Pearl and Dana Mackenzie. 2018. *The book of why: the new science of cause and effect*. Basic books.
[55] Robert J Prill, Daniel Marbach, Julio Saez-Rodriguez, Peter K Sorger, Leonidas G Alexopoulos, Xiaowei Xue, Neil D Clarke, Gregoire Altan-Bonnet, and Gustavo Stolovitzky. 2010. Towards a rigorous assessment of systems biology models: the DREAM3 challenges. *PloS one* 5, 2 (2010), e9202.
[56] Jiaxin Qi, Yulei Niu, Jianqiang Huang, and Hanwang Zhang. 2020. Two causal principles for improving visual dialog. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 10860–10869.
[57] Shane Roach, Connie Ni, Alexei Kopylov, Tsai-Ching Lu, Jiejun Xu, Si Zhang, Boxin Du, Dawei Zhou, Jun Wu, Lihui Liu, et al. 2020. CANON: Complex Analytics of Network of Networks for Modeling Adversarial Activities. In *2020 IEEE International Conference on Big Data (Big Data)*. IEEE, 1634–1643.
[58] Yulia Rubanova, Ricky TQ Chen, and David K Duvenaud. 2019. Latent ordinary differential equations for irregularly-sampled time series. *Advances in neural information processing systems* 32 (2019).
[59] Donald B Rubin. 2005. Causal inference using potential outcomes: Design, modeling, decisions. *J. Amer. Statist. Assoc.* 100, 469 (2005), 322–331.
[60] Jakob Runge, Peer Nowack, Marlene Kretschmer, Seth Flaxman, and Dino Sejdinovic. 2019. Detecting and quantifying causal associations in large nonlinear time series datasets. *Science advances* 5, 11 (2019), eaau4996.
[61] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*. 618–626.
[62] Siyuan Shan, Yang Li, and Junier B Oliva. 2023. Nrtsi: Non-recurrent time series imputation. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 1–5.
[63] Satya Narayan Shukla and Benjamin Marlin. 2020. Multi-Time Attention Networks for Irregularly Sampled Time Series. In *International Conference on Learning Representations*.
[64] David I Shuman, Sunil K Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. 2013. 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 (2013), 83–98.
[65] Yongduo Sui, Xiang Wang, Jiancan Wu, Min Lin, Xiangnan He, and Tat-Seng Chua. 2022. Causal attention for interpretable and generalizable graph classification. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 1696–1705.
[66] Alex Tank, Ian Covert, Nicholas Foti, Ali Shojaie, and Emily B Fox. 2021. Neural granger causality. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 8 (2021), 4267–4279.
[67] Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. 2021. Csdi: Conditional score-based diffusion models for probabilistic time series imputation. *Advances in Neural Information Processing Systems* 34 (2021), 24804–24816.
[68] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).
[69] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903* (2017).
[70] Dingsu Wang, Yuchen Yan, Ruizhong Qiu, Yada Zhu, Kaiyu Guan, Andrew Margenot, and Hanghang Tong. 2023. Networked time series imputation via position-aware graph enhanced variational autoencoders. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2256–2268.
[71] Jun Wang, Wenjie Du, Wei Cao, Keli Zhang, Wenjia Wang, Yuxuan Liang, and Qingsong Wen. 2024. Deep learning for multivariate time series imputation: A survey. *arXiv preprint arXiv:2402.04059* (2024).
[72] Ian R White, Patrick Royston, and Angela M Wood. 2011. Multiple imputation using chained equations: issues and guidance for practice. *Statistics in medicine* 30, 4 (2011), 377–399.
[73] Alexander P Wu, Rohit Singh, and Bonnie Berger. 2021. Granger causal inference on DAGs identifies genomic loci regulating transcription. In *International Conference on Learning Representations*.
[74] Haobo Xu, Yuchen Yan, Dingsu Wang, Zhe Xu, Zhichen Zeng, Tarek F Abdelzaher, Jiawei Han, and Hanghang Tong. 2024. SLOG: An Inductive Spectral Graph Neural Network Beyond Polynomial Filter. In *Forty-first International Conference on Machine Learning*.
[75] Yuchen Yan, Yuzhong Chen, Huiyuan Chen, Minghua Xu, Mahashweta Das, Hao Yang, and Hanghang Tong. 2024. From trainable negative depth to edge heterophily in graphs. *Advances in Neural Information Processing Systems* 36 (2024).
[76] Yuchen Yan, Yongyi Hu, Qinghai Zhou, Lihui Liu, Zhichen Zeng, Yuzhong Chen, Menghai Pan, Huiyuan Chen, Mahashweta Das, and Hanghang Tong. 2024. Pacer: Network embedding from positional to structural. In *Proceedings of the ACM on Web Conference 2024*. 2485–2496.
[77] Yuchen Yan, Baoyu Jing, Lihui Liu, Ruijie Wang, Jinning Li, Tarek Abdelzaher, and Hanghang Tong. 2024. Reconciling competing sampling strategies of network embedding. *Advances in Neural Information Processing Systems* 36 (2024).
[78] Yuchen Yan, Lihui Liu, Yikun Ban, Baoyu Jing, and Hanghang Tong. 2021. Dynamic knowledge graph alignment. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35. 4564–4572.
[79] Yuchen Yan, Si Zhang, and Hanghang Tong. 2021. Bright: A bridging algorithm for network alignment. In *Proceedings of the web conference 2021*. 3907–3917.
[80] Yuchen Yan, Qinghai Zhou, Jinning Li, Tarek Abdelzaher, and Hanghang Tong. 2022. Dissecting cross-layer dependency inference on multi-layered inter-dependent networks. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 2341–2351.
[81] Xu Yang, Hanwang Zhang, Guojun Qi, and Jianfei Cai. 2021. Causal attention for vision-language tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 9847–9857.
[82] Xiuwen Yi, Yu Zheng, Junbo Zhang, and Tianrui Li. 2016. ST-MVL: filling missing values in geo-sensory time series data. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence*.
[83] A Yarkın Yıldız, Emirhan Koç, and Aykut Koç. 2022. Multivariate time series imputation with transformers. *IEEE Signal Processing Letters* 29 (2022), 2517–2521.
[84] Hsiang-Fu Yu, Nikhil Rao, and Inderjit S Dhillon. 2016. Temporal regularized matrix factorization for high-dimensional time series prediction. *Advances in neural information processing systems* 29 (2016).
[85] Zhongqi Yue, Tan Wang, Qianru Sun, Xian-Sheng Hua, and Hanwang Zhang. 2021. Counterfactual zero-shot and open-set visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 15404–15414.
[86] Zhichen Zeng, Boxin Du, Si Zhang, Yinglong Xia, Zhining Liu, and Hanghang Tong. 2024. Hierarchical multi-marginal optimal transport for network alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 16660–16668.
[87] Zhichen Zeng, Ruizhong Qiu, Zhe Xu, Zhining Liu, Yuchen Yan, Tianxin Wei, Lei Ying, Jingrui He, and Hanghang Tong. 2024. Graph Mixup on Approximate Gromov–Wasserstein Geodesics. In *Forty-first International Conference on Machine Learning*.
[88] Zhichen Zeng, Si Zhang, Yinglong Xia, and Hanghang Tong. 2023. Parrot: Position-aware regularized optimal transport for network alignment. In *Proceedings of the ACM Web Conference 2023*. 372–382.
[89] J.P Zhang, T Zhu, QH Zhang, CC Li, HL Shu, Y Ying, ZP Dai, X Wang, XY Liu, AM Liang, et al. 2012. The impact of circulation patterns on regional transport pathways and air quality over Beijing and its surroundings. *Atmospheric Chemistry and Physics* 12, 11 (2012), 5031–5053.
[90] Xiangyu Zhao and Jiliang Tang. 2017. Modeling temporal-spatial correlations for crime prediction. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. 497–506.
[91] Lecheng Zheng, Zhengzhang Chen, Jingrui He, and Haifeng Chen. 2024. MULAN: Multi-modal Causal Structure Learning and Root Cause Analysis for Microservice Systems. In *Proceedings of the ACM on Web Conference 2024*. 4107–4116.
[92] Lecheng Zheng, Dongqi Fu, Ross Maciejewski, and Jingrui He. 2021. Deeper-GXX: deepening arbitrary GNNs. *arXiv preprint arXiv:2110.13798* (2021).
[93] Lecheng Zheng, Baoyu Jing, Zihao Li, Hanghang Tong, and Jingrui He. 2024. Heterogeneous Contrastive Learning for Foundation Models and Beyond. *arXiv preprint arXiv:2404.00225* (2024).
[94] Yu Zheng, Xiuwen Yi, Ming Li, Ruiyuan Li, Zhangqing Shan, Eric Chang, and Tianrui Li. 2015. Forecasting fine-grained air quality based on big data. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*. 2267–2276.
[95] Dawei Zhou, Lecheng Zheng, Jiawei Han, and Jingrui He. 2020. A data-driven graph generative model for temporal interaction networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 401–411.
[96] Qinghai Zhou. 2023. *Closed-loop network anomaly detection*. Ph. D. Dissertation. University of Illinois at Urbana-Champaign.