Neural Networks 202 (2026) 108986

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Full Length Article

GeoMAE: Masking representation learning for spatio-temporal graph forecasting with missing values

Songyu Ke [a,b,c,1], Chenyu Wu [d], Yuxuan Liang [e], Huiling Qin [f], Junbo Zhang [b,c,d,g,*], Yu Zheng [d,h,g]

[a] College of Computer and Data Science, Fuzhou University, Fuzhou, Fujian, China
[b] JD Intelligent Cities Research, Beijing, China
[c] JD iCity, JD Technology, Beijing, China
[d] Southwest Jiaotong University, Chengdu, Sichuan, China
[e] Hong Kong University of Science and Technology (Guangzhou), Guangzhou Guangdong, China
[f] Beijing Normal University, Zhuhai Guangdong, China
[g] Beijing Key Laboratory of Traffic Data Mining and Embodied Intelligence, Beijing, China
[h] Xidian University, Xi'an Shaanxi, China

ARTICLE INFO

Keywords:
Representation learning
Self-supervised learning
Learning with missing data
Urban computing
Spatio-temporal data mining

ABSTRACT

The ubiquity of missing data in urban intelligence systems, attributable to adverse environmental conditions and equipment failures, poses a significant challenge to the efficacy of downstream applications, notably in the realms of traffic forecasting and energy consumption prediction. Therefore, it is imperative to develop a robust spatio-temporal learning methodology capable of extracting meaningful insights from incomplete datasets. Despite the existence of methodologies for spatio-temporal graph forecasting in the presence of missing values, unresolved issues persist.

Primarily, the majority of extant research is predicated on time-series analysis, thereby neglecting the dynamic spatial correlations inherent in sensor networks. Additionally, the complexity of missing data patterns compounds the intricacy of the problem. Furthermore, the variability in maintenance conditions results in a significant fluctuation in the ratio and pattern of missing values, thereby challenging the generalizability of predictive models.

In response to these challenges, this study introduces GeoMAE, an enhanced spatio-temporal representation learning model with a self-supervised auxiliary loss. The model comprises three principal components: an input preprocessing module, an attention-based spatio-temporal forecasting network (STAFN), and an auxiliary learning task, which inspired by Masking AutoEncoders to enhance the robustness of spatio-temporal representation learning.

Empirical evaluations on real-world datasets demonstrate that GeoMAE significantly outperforms existing benchmarks, achieving up to 13.20% relative improvement in MAE over the best baseline models (e.g., 22.42 vs. 25.01 MAE at 25% missing rate).

1. Introduction

Spatio-temporal representation learning has emerged as a pivotal research area, underpinning various intelligent applications in smart cities which are essential across multiple domains. For instance, precise weather forecasting can significantly mitigate the detrimental impacts of natural disasters through early prevention; advanced traffic prediction systems help optimize traffic flow and substantially reduce congestion; environmental monitoring enables rapid identification of pollution hotspots within urban environments.

However, the prevalence of missing values in spatio-temporal data, attributable to sensor failures, network interruptions, human errors, and other issues during data acquisition, transmission, and storage, significantly complicates the representation learning process. These missing values profoundly impact the complex temporal and spatial dependencies inherent in spatio-temporal data, thereby impeding the ability

* Corresponding author.
E-mail addresses: songyuke@fzu.edu.cn (S. Ke), chenywu@swjtu.edu.cn (C. Wu), yuxliang@outlook.com (Y. Liang), qinhuiling@bnu.edu.cn (H. Qin), songyuke@fzu.edu.cn, msjunbozhang@outlook.com (J. Zhang), msyuzheng@outlook.com (Y. Zheng).

[1] This research was partially done when the first author was an intern at JD Intelligent Cities Research & JD iCity under the supervision of the fifth author.

https://doi.org/10.1016/j.neunet.2026.108986
Received 1 December 2025; Received in revised form 16 March 2026; Accepted 11 April 2026
Available online 18 April 2026

to discern underlying patterns. Moreover, imputation of missing values, while often necessary for deep learning models, incurs additional computational costs and may introduce biases that adversely affect the models' generalization capabilities. Consequently, effectively addressing missing values and learning representations from incomplete spatio-temporal data has become a critical research direction in this domain.

Current representation learning approaches for incomplete data can be broadly categorized into two principal paradigms. The first approach treats missing value imputation as a distinct task, employing independent methods or models to accurately impute missing values before applying dedicated representation learning models for downstream tasks such as prediction or classification. Traditional machine learning methods, including linear regression, K-Nearest Neighbors, random forests, clustering, and principal component analysis (Haworth & Cheng, 2012; Henrickson et al., 2015; Ku et al., 2016; Stekhoven & Bühlmann, 2012), alongside advanced deep learning techniques such as Variational Autoencoders (VAE) (Fortuin et al., 2020; Nazábal et al., 2020), Generative Adversarial Networks (GAN) (Li et al., 2019; Yoon et al., 2018), and Denoising Diffusion Probabilistic Models (DDPM) (Tashiro et al., 2021), have been deployed for data imputation. This paradigm offers flexibility in selecting appropriate imputation techniques based on data characteristics and missing patterns. However, it necessitates training both imputation and downstream task models, increasing computational complexity, while errors from the imputation model may propagate to downstream tasks, potentially degrading performance.

The alternative paradigm involves directly modeling incomplete data for downstream tasks, streamlining the processing pipeline and mitigating the impact of imputation errors. For instance, decision tree-based models typically treat missing values as special categories requiring no additional handling. Models based on Neural Ordinary Differential Equations (Neural ODEs) view irregularly spaced observations as multiple measurements of a continuous process, attempting to fit the data using Neural ODEs (Chang et al., 2023; Chen et al., 2018; Habiba & Pearlmutter, 2020). Nevertheless, significant temporal irregularities in spatio-temporal data collection can adversely affect Neural ODEs, and their substantial computational demands challenge application to large-scale datasets. The TriD-MAE model employs multi-period Temporal Convolutional Networks (TCNs) to capture temporal correlations in incomplete data, coupled with a masked autoencoding auxiliary task to enhance representation robustness for multivariate time series prediction (Zhang et al., 2023). The GinAR model introduces an imputation-aware attention mechanism to learn spatio-temporal dependencies from incomplete data for subsequent forecasting tasks (Yu et al., 2024).

Despite these advances, constructing effective representation learning models for incomplete spatio-temporal data still faces significant challenges:

1. **Complex and Dynamic Spatio-Temporal Dependencies**
   Spatio-temporal data exhibits intricate and dynamically evolving dependencies. Taking air quality prediction as an example, readings from a monitoring station correlate not only with its historical data but also with concurrent measurements from nearby stations. These spatial correlations are highly dynamic, influenced by geographic location, temporal factors, and external conditions such as wind direction and speed. Additionally, strong inter-variable correlations exist, for instance, sulfur dioxide and nitrogen oxides can transform into PM2.5 particles under specific conditions. Effectively modeling these temporal, spatial, and variable correlations amidst missing values remains a fundamental challenge for representation learning models.
2. **Diverse Missing Patterns**
   As illustrated in Fig. 1, spatio-temporal data manifests various missing patterns, broadly classified into random missing, row missing, column missing, and block missing. Random missing, often caused by sporadic device errors or packet loss, typically affects individual time points and locations. Row missing occurs when all sensor readings are absent for a specific period, often due to system-wide failures. Column missing refers to continuous absence of one or more variables from a specific location, usually resulting from sensor maintenance or environmental damage. Block missing denotes simultaneous column missing across multiple sensors over extended periods, causing substantial information loss. Real-world datasets often present a mixture of these patterns, posing considerable challenges for representation learning.
3. **Fluctuating Missing Conditions**
   In practical systems, data missingness closely relates to maintenance quality, which is directly constrained by operational budgets. During well-funded periods, systems are properly maintained, sensors operate reliably, and missing rates remain low. Conversely, budget constraints lead to inadequate maintenance, increased equipment failures, and consequently, degraded data quality with higher missing rates and more complex patterns. As illustrated in Fig. 2, which shows missing proportions for Beijing's air quality data across years, the missing rate in 2016 is notably lower than in other years, demonstrating this budget-driven phenomenon. **Current research often overlooks these realistic fluctuations, typically assuming fixed missing patterns and rates, potentially compromising model generalization in real-world applications.**

To address these challenges and enhance spatio-temporal representation learning with incomplete data, we propose GeoMAE, a novel model for spatio-temporal graph data prediction with missing values. Our approach comprises three key components:

1. **Input Preprocessing Module:** To mitigate the impact of the missing pattern and ratio drifting, we employ random imputation strategies and design a new masking matrix, which can enhance model adaptability to different levels of data incompleteness.
2. **Attention-based Spatio-Temporal Representation Network:** To address the challenge of complex and dynamic spatio-temporal dependencies, we propose a new Spatio-Temporal Attention Forecasting Network (STAFN), which leverages spatio-temporal and spatial attention modules to efficiently capture complex dependencies and learn effective representations from incomplete data.
3. **Multi-task Optimization with Attribute Masking:** To address the two challenges (2 and 3) of real-world spatio-temporal modeling with missing data, we design a self-supervised auxiliary loss to lead the model to capture the invariant latent representation of the physical world. It can minimize distances between representation vectors of the same spatio-temporal point under different masking patterns, reducing the impact of missing data noise.

The main contributions of this work are fourfold:

1. We present one of the first comprehensive investigations into the impact of varying missing data conditions (including rates and patterns) on representation learning performance, with targeted architectural designs to mitigate these effects.
2. We propose a novel input processing module that effectively alleviates distributional shifts caused by varying missing rates, significantly improving model generalization capability.
3. We design a self-supervised multi-task learning framework that constructs augmented samples through simulated missing scenarios, enhancing representation robustness against data incompleteness.
4. Extensive experiments on real-world datasets demonstrate the superiority of our approach compared to state-of-the-art methods.

2. Related works

2.1. Missing value imputation

Spatio-temporal data often contains a significant amount of missing values due to various errors occurring during data acquisition, processing, transmission, and storage. A range of methods have been developed for missing value imputation. The simplest approach is mean

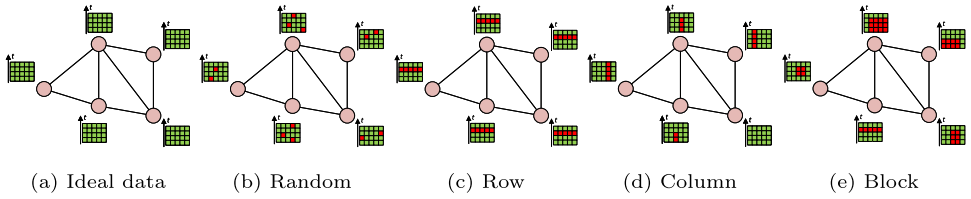


Fig. 1. Illustrations of data samples with different missing patterns, where the x-axis represents different features, the y-axis is the time axis, the red block indicates a missing point, and green blocks are valid data.

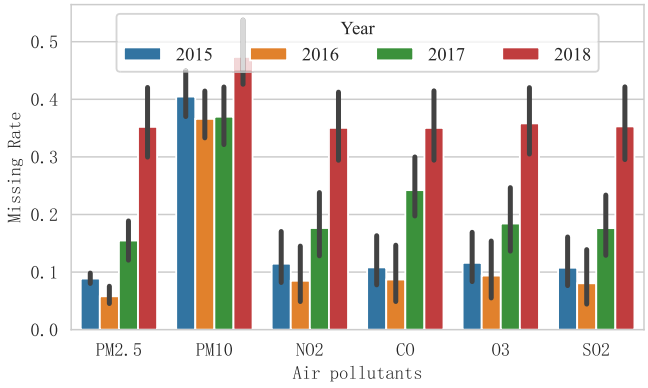


Fig. 2. The changes on missing proportions of features (best viewed in the colored version). Most pollutant data show a minimum missingness ratio in 2016, and the ratio increases significantly in 2018, which exerts negative effects on spatio-temporal modeling.

imputation, which fills missing entries with the average value of the available data. This method is straightforward and preserves the mean of the dataset. For data that is Missing Completely at Random (MCAR), mean imputation can maintain the original data distribution reasonably well. Alternatively, using the most recent non-missing value can be a suitable choice for time-ordered data. Additionally, various statistical methods are commonly employed, including least squares, naive Bayes, and principal component analysis (Henrickson et al., 2015; Tan et al., 2013; Tang et al., 2018).

With advances in machine learning, algorithms such as K-Nearest Neighbors (KNN), Random Forests, and Generative Adversarial Networks (GANs) have been widely applied. Missforest (Stekhoven & Bühlmann, 2012), an R package, provides a general-purpose imputation method that first uses mean or mode for initial imputation, then iteratively trains random forest models on each feature with missing values to predict and update the imputations until convergence. Zhong et al. (2004) developed a genetic algorithm integrated with an artificial neural network for missing value estimation (Zhong et al., 2004). Zhao et al. (2015) proposed a Bayesian Canonical Polyadic Decomposition (CPD) framework for incomplete tensor decomposition and recovery (Zhao et al., 2015). GAIN (Yoon et al., 2018) adapted GANs for missing data imputation by generating imputations that follow the distribution of the observed data.

However, spatio-temporal data possesses unique temporal and spatial attributes, and often contains a high proportion of Not Missing at Random (NMAR) values, making the general methods mentioned above less suitable. For example, mean imputation ignores the temporal proximity and periodicity inherent in spatio-temporal data (Zhang et al., 2017). Extreme weather events can cause sensor failures leading to NMAR missingness, and ignoring dynamic external factors like meteorological conditions can significantly reduce imputation precision.

Therefore, researchers have designed models specifically tailored for spatio-temporal data. (Haworth & Cheng, 2012) utilized a KNN-based method to recover missing traffic data by leveraging information from neighboring road segments (Haworth & Cheng, 2012). Ku et al. (2016) introduced a clustering-based imputation method that uses Stacked Denoising Autoencoders to obtain high-dimensional representations of traffic data, followed by clustering to identify similar samples for imputation (Ku et al., 2016). Yi et al. (2016) proposed the ST-MVL model, which captures spatio-temporal characteristics from four aspects: spatial proximity, temporal proximity, spatial similarity, and temporal similarity, combining them using a linear model to generate final imputations (Yi et al., 2016). GANs have also been adapted for spatio-temporal data generation and imputation. Gao et al. (2022) surveyed various GAN architectures suitable for time series and spatio-temporal data (Gao et al., 2022). To mitigate GANs' training instability, Qin et al. (2021) introduced the ST-SCL model, which incorporates a regression loss alongside the adversarial loss, simplifying training and improving performance on traffic data imputation tasks (Qin et al., 2021).

In recent studies, ImputeFormer employs low-rankness-induced transformers to model global spatiotemporal correlations for generalizable imputation (Nie et al., 2024), while SSD-TS integrates linear state space models into a diffusion-based framework to enhance the quality of missing value recovery (Gao et al., 2025).

2.2. *Directly modeling incomplete data*

The aforementioned research primarily treats missing value imputation as a separate machine learning task. Subsequent prediction, classification, or recommendation tasks require fitting new models on the imputed data. While this two-step approach is modular and widely applicable, it may introduce bias or noise, and imputation errors can propagate to downstream tasks. Therefore, some studies have explored directly modeling incomplete data for end-to-end task performance.

(Che et al., 2018) proposed GRU-D, which directly models incomplete time series data by learning representations that account for missing patterns (Che et al., 2018). Hsu et al. (2020) developed Fuzzy-CNN for object detection in images with missing pixels (Hsu et al., 2020). Habiba and Pearlmutter (2020) enhanced GRU-D by incorporating Neural ODEs to improve the modeling of incomplete data (Habiba & Pearlmutter, 2020). Chang et al. (2023) proposed TN-ODE to handle irregular time intervals between sequential elements, facilitating the modeling of incomplete time series (Chang et al., 2023). Nasiri and Ebadzadeh (2022) employed a Multi-Functional Recurrent Fuzzy Neural Network (MFRFNN) to directly model complex, chaotic time series containing missing values (Nasiri & Ebadzadeh, 2022). Zhang et al. (2023) introduced TriD-MAE, which uses multi-period Temporal Convolutional Networks (TCNs) to capture correlations in incomplete time series data for direct future value prediction (Zhang et al., 2023). However, TriD-MAE lacks specific structures for capturing spatial correlations, such as graph neural networks, which limits its performance on spatio-temporal graph data. In contrast, GeoMAE integrates spatial modeling through its STAFN architecture. Moreover, Yu et al. (2024) designed GinAR/GinAR+, which utilizes an imputation-aware attention mechanism to learn spatio-temporal dependencies from incomplete data for subsequent forecasting tasks (Yu et al., 2025a, 2024). And Merlin employs multi-view representation learning to tackle unfixed missing

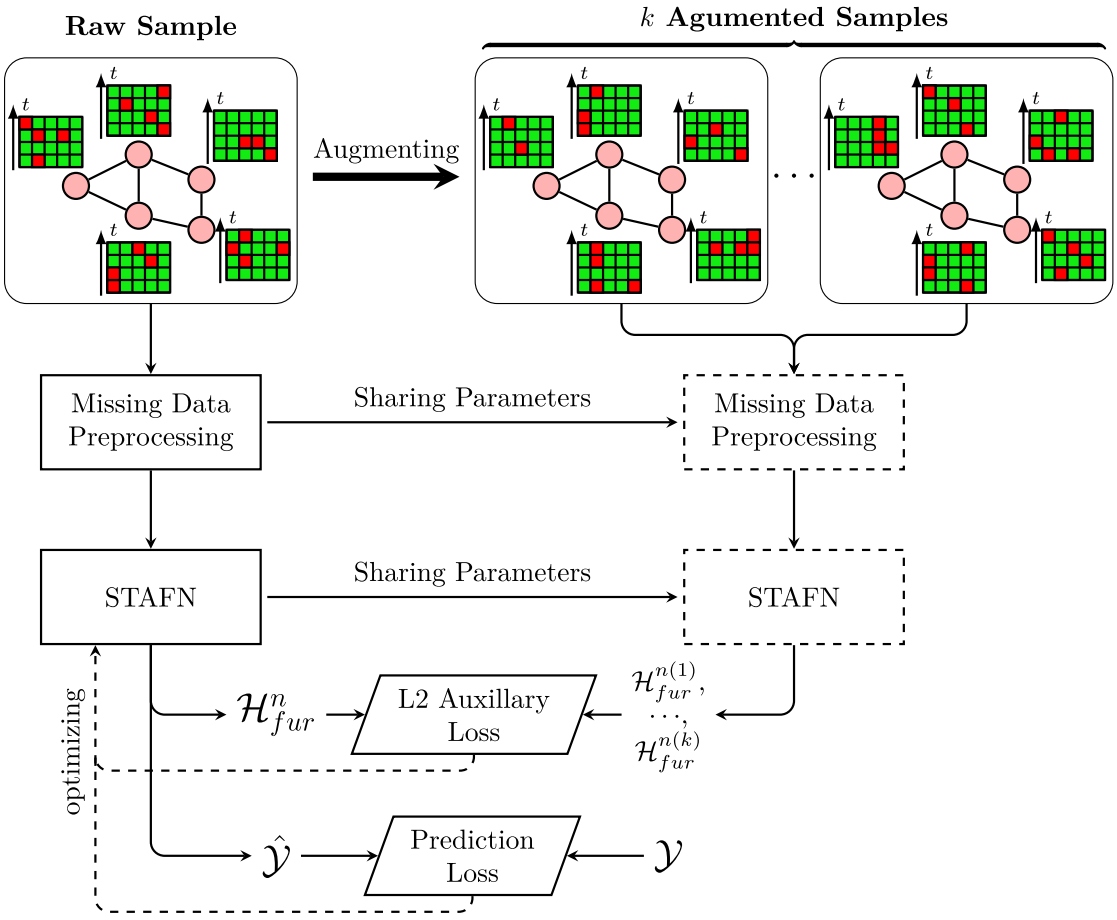


Fig. 3. The framework of GeoMAE.

Table 1
Frequently used notations.

Notations	Description
N_l	The number of nodes in a spatio-temporal graph
$N_{\text{in}}, N_{\text{out}}$	The number of historical inputs and future outputs
$\mathbf{X}_t$	The reading matrix at timestamp t
$\mathbf{M}_t$	The missing indicator matrix at timestamp t
$\mathcal{Y}$	The output prediction

rates, but it does not discuss about different missing patterns (Yu et al., 2025b).

Despite these advances, representation learning methods specifically designed for incomplete multivariate spatio-temporal data remain an area requiring further exploration, particularly models that are robust to real-world data conditions.

3. Preliminaries

Table 1 shows the frequently used notations in this article. And we formulate our research problem as follows:

Definition 1 (Spatio-Temporal Graph Prediction with Missing Values)**.** Given continuous N_{in} readings of past moments $\mathcal{X} = [\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_{\text{in}}]$, missing identification tensors $\mathcal{M} = [\mathbf{M}_1, \mathbf{M}_2, \ldots, \mathbf{M}_{\text{in}}]$ and static spacial relation $\mathcal{G}$, predicting the readings for the coming N_{out} steps, $\hat{\mathcal{Y}} = [\hat{\mathbf{Y}}_1, \hat{\mathbf{Y}}_2, \ldots, \hat{\mathbf{Y}}_{\text{out}}]$, among them, $\mathbf{X}_i \in \mathbb{R}^{N_l \times D_{\text{in}}}$, $\mathbf{M}_i \in \{0,1\}^{N_l \times D_{\text{in}}}$, $\mathbf{Y}_i \in \mathbb{R}^{N_l \times D_{\text{out}}}$. Particularly, $\mathcal{M}$ represents whether there is a corresponding missing value in $\mathcal{X}$, 1 represents missing value while non-missing value is 0.

4. Methodologies

GeoMAE consists of three major components: input data processing, spatio-temporal representation learning, and self-supervised auxiliary tasks. Fig. 3 illustrates the primary workflow. GeoMAE first uses a carefully designed input data preprocessing module to construct a hint tensor $\mathcal{M}_{hint}$ and performs necessary imputation on the input readings $\mathcal{X}$. Then, it employs the spatio-temporal attention forecasting network (STAFN) for representation learning to obtain future spatio-temporal representation $\mathcal{H}_n$. Finally, a fully connected network is used to decode $\mathcal{H}_n$ into the final prediction $\mathcal{Y}$.

Additionally, to enhance the stability and generalization ability of the spatio-temporal representation learning process, GeoMAE adopts a method similar to Masked AutoEncoder (MAE), i.e., adding extra masks to the data and incorporating an additional self-supervised loss to reduce the distance between the representations of the original data and the augmented samples.

The following subsections will introduce the input data preprocessing method, STAFN, and the multitask optimization method with self-supervised loss.

4.1. Missing data preprocessing

Unlike models based on decision trees that can handle missing values by treating them as special categorical features, deep learning models, particularly those based on neural networks, struggle to process input data with missing values, especially when missing values are represented by NaN (Not a Number).

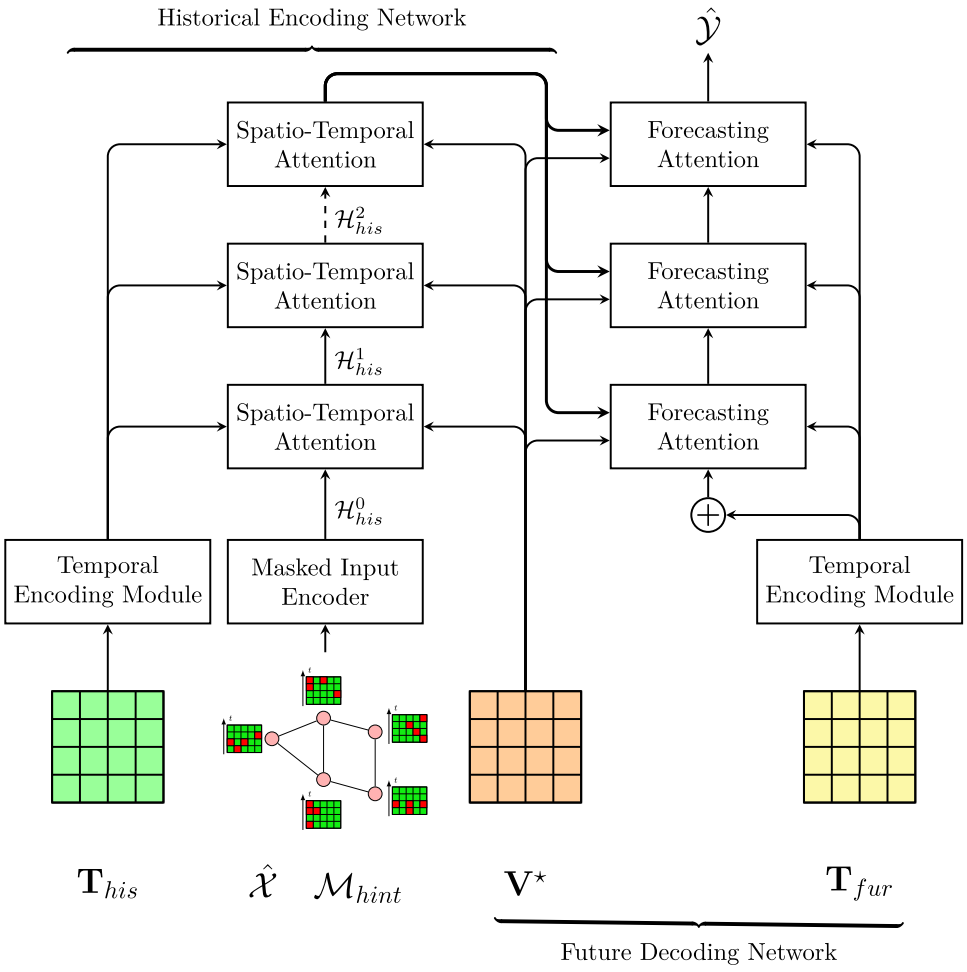


Fig. 4. The structure of STAFN.

Therefore, missing values require special preprocessing. A common approach is to fill missing values with "0". On the one hand, "0" is a passive signal in neural network computations, meaning that it does not actively activate or suppress neurons. On the other hand, after the typical standardization preprocessing, the data would follow a standard normal distribution, and assigning zeros to missing values would have minimal impact on the original data distribution. Moreover, adding extra missing values to the input data and filling them with 0s behaves similarly to Dropout in the input layer, which models uncertainty, helping the model cope with partial input loss (Gal & Ghahramani, 2016). However, this approach is only suitable when the missing data proportion is low and the actual missing rate in the data matches the training rate. This is because a high missing rate would lead to a large number of zeros in the input samples, altering the data distribution, and this bias could be learned by the model, ultimately affecting its generalization ability.

Moreover, incorporating indicators of missingness in the input data is a feasible strategy. GAIN inputs both the mask matrix and the data into the model to extract the data representation (Yoon et al., 2018), and BRITS constructs a missing matrix and inputs it into the model to better capture correlations in time series data (Cao et al., 2018). Generally, these models construct a 0/1 mask matrix, where 0 represents missing data and 1 represents existing data (or vice versa). However, this type of mask matrix faces a similar problem: the distribution of matrix values varies with different missing rates, which can affect the model's generalization ability.

Therefore, we propose a new data preprocessing method that includes two components: missing data imputation and mask matrix construction. We first use random variables that follow a normal distribution for imputation to handle missing data.

$$\hat{x}^{ijk} = \begin{cases} x^{ijk} & m^{ijk} = 0 \\ \varepsilon \sim \mathcal{N}(0, \sigma) & m^{ijk} = 1 \end{cases}, \tag{1}$$

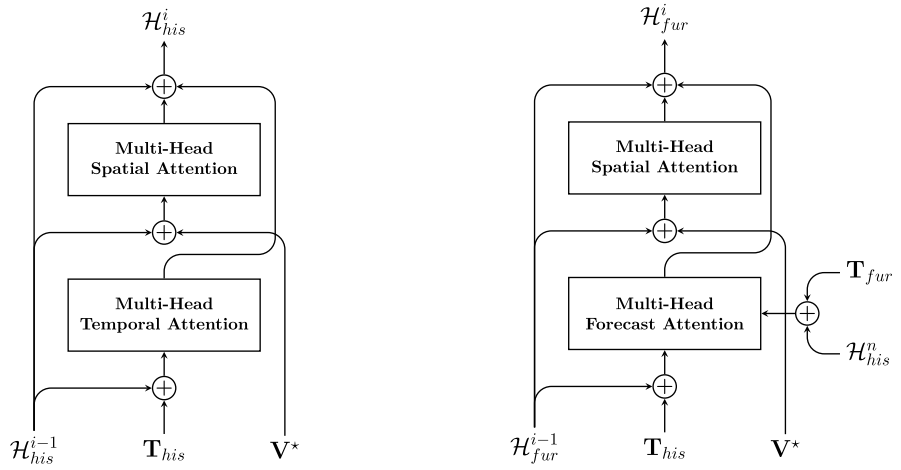
where i, j, and k are the indices of the input tensor $\mathcal{X}$ across the node, time, and feature dimensions, respectively. The variable ε is the imputed random value that follows a normal distribution with a mean of 0 and a standard deviation of σ.

It is assumed that the input features have been standardized, i.e.,

$$\mathbf{x}^i = \frac{\mathbf{x}^i_{\text{raw}} - \bar{\mathbf{x}}^i_{\text{raw}}}{\sigma(\mathbf{x}^i_{\text{raw}})}, \tag{2}$$

where $\mathbf{x}^i_{\text{raw}}$ represents the raw vector of the ith feature, $\bar{\mathbf{x}}^i_{\text{raw}}$ represents the mean of that feature, and $\sigma(\mathbf{x}^i_{\text{raw}})$ represents the standard deviation of that feature. After this standardization process, the input data should follow a distribution with a mean of 0 and a standard deviation of 1. Therefore, using random variables that follow a normal distribution with a mean of 0 and a standard deviation of σ for the imputation will minimally affect the distribution of the input features. However, to avoid excessively large random values that could cause abnormal activation of certain neurons and introduce additional noise, it is recommended to choose a relatively small value for σ (e.g., 0.2).

Additionally, based on the identification tensor $\mathcal{M}$, this study constructs a hint sensor $\mathcal{M}_{\text{hint}}$. The specific process is as follows:



(a) Spatio-temporal attention module (b) Forecasting attention module

Fig. 5. The structures of two attention modules in STAFN.

1. Constructing a balanced mask tensor $\mathcal{M}_{\text{sym}} = (m^{ijk}_{\text{sym}})$, where

$$m^{ijk}_{\text{sym}} = \begin{cases} 1 & m^{ijk} = 0 \\ -1 & m^{ijk} = 1 \end{cases} \tag{3}$$

2. Standardize and scale $\mathcal{M}_{\text{sym}}$ to obtain $\mathcal{M}_{\text{hint}}$:

$$\mathcal{M}_{\text{hint}} = \frac{\mathcal{M}_{\text{sym}} - \bar{\mathcal{M}}_{\text{sym}}}{\sigma[\mathcal{M}_{\text{sym}}]} \tag{4}$$

where $\bar{\mathcal{M}}_{\text{sym}}$ and $\sigma[\mathcal{M}_{\text{sym}}]$ represent the mean and variance of $\mathcal{M}_{\text{sym}}$ respectively.

4.2. Spatio-temporal attention forecasting network

To better capture the complex and dynamic spatio-temporal correlations in the data, we design and implement a STAFN (Spatio-Temporal Attention Forecasting Network) based on attention mechanisms. The main structure of STAFN is illustrated in Fig. 4.

STAFN adopts an Encoder-Decoder architecture, which comprises two parts: the historical encoding network (Encoder) and the future decoding network (Decoder). The temporal encoding module is shared between the encoder and the decoder. It aims to encode timestamp (e.g., month, day, and hour) into a temporal vector to help learn representation with multi-head attention mechanisms. The temporal encoding method is the common sin/cos positional encoding, as detailed in the literature (Wu et al., 2023). Moreover, STAFN also maintains a learnable node representation denoted as $\mathbf{V}^\star \in \mathbb{R}^{N_l \times D_{\text{model}}}$. $\mathbf{V}^\star$ is fed into the spatio-temporal attention and forecasting attention modules, allowing attention mechanisms to capture spatial correlations from the data.

STAFN consists of n spatio-temporal attention modules and n forecasting attention modules, where n is a hyperparameter of the network. These attention modules are composed of multi-head spatial attention, multi-head temporal attention, and multi-head forecast attention. Fig. 5 illustrates the structures of the spatio-temporal attention module and the forecasting attention module.

In the spatio-temporal attention module, the multi-head temporal attention mechanism and the multi-head spatial attention mechanism are arranged in parallel. The spatio-temporal representation $\mathcal{H}^{i-1}_{\text{his}}$ is output by the previous module is combined with the temporal vector $\mathbf{T}_{\text{his}}$ and the spatial encoding $\mathbf{V}^\star$, and then fed into the multi-head attention mechanism. For the forecasting attention module, since its initial representation $\mathcal{H}^0_{\text{fur}}$ is merely the sum of the time encoding and the spatial encoding without any information related to the current sensor readings, a serial architecture is chosen, where the multi-head predictive attention mechanism is followed by the multi-head spatial attention mechanism. This design ensures that the input to the spatial attention is not simply a combination of fixed spatial encodings and time encodings.

Here is a brief description of the multi-head spatial attention, temporal attention, and forecast attention:

4.2.1. Multi-head spatial attention

Fig. 6 illustrates the details of a spatial attention. $\mathcal{H}^{i-1}_s \in \mathbb{R}^{T \times N_l \times D_{\text{model}}}$ is the input tensor to the spatial attention mechanism, containing time, space, and feature dimensions. We split $\mathcal{H}^{i-1}_s$ into T representation matrices along the temporal dimension. Without loss of generality, we take an arbitrary $\mathbf{H}^{i-1}_{s(j)} \in \mathbb{R}^{D_{\text{model}} \times N_l}$ as an example to explain the computation

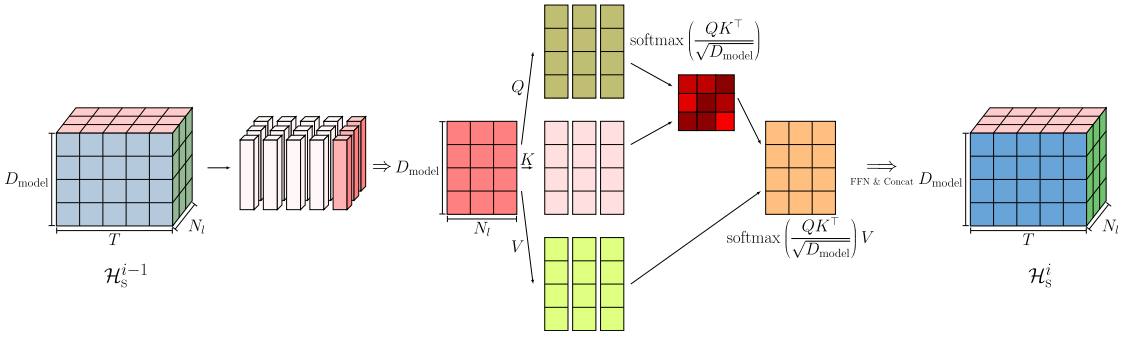


Fig. 6. The structure of multi-head spatial attention.

of the spatial attention:

$$\mathbf{Q}_s^i = \mathbf{W}_{Q(s)}^i \mathbf{H}_{s(j)}^{i-1}, \tag{5}$$

$$\mathbf{K}_s^i = \mathbf{W}_{K(s)}^i \mathbf{H}_{s(j)}^{i-1}, \tag{6}$$

$$\mathbf{V}_s^i = \mathbf{W}_{V(s)}^i \mathbf{H}_{s(j)}^{i-1}, \tag{7}$$

$$\mathbf{A}_s^i = \frac{\mathbf{Q}_s^i \mathbf{K}_s^{i\top}}{\sqrt{D_{\text{model}}}}, \tag{8}$$

$$\alpha_{ef} = \frac{\exp a_{ef}}{\sum_{g=1}^{N_l} \exp a_{eg}}, \tag{9}$$

$$\tilde{\mathbf{A}} = (\alpha_{ef}), \tag{10}$$

$$\mathcal{H}_{s(j)}^i = \text{MLP}(\tilde{\mathbf{A}} \mathbf{V}_s^i), \tag{11}$$

where, $\text{MLP}(\cdot)$ denotes a multi-layer perceptron operation, and $\mathbf{W}_{Q(s)}^i$, $\mathbf{W}_{K(s)}^i$, $\mathbf{W}_{V(s)}^i \in \mathbb{R}^{D_{\text{model}} \times D_{\text{model}}}$ represent the learnable parameters for the query (Q), key (K), and value (V) mappings, respectively.

4.2.2. Multi-head temporal attention

The temporal attention mechanism is similar to the spatial attention mechanism, except the spatial attention splits the tensor along the time axis and performs self-attention at the node dimension, while the temporal attention splits the representation along the node dimension and performs self-attention across different time steps for the same node. The computation process for a single attention head is as follows:

$$\mathcal{H}_{t(j)}^i = \text{MLP}[\text{softmax}\left(\frac{\mathbf{W}_{Q(t)}^i \mathbf{H}_{t(j)}^{i-1} \left(\mathbf{W}_{K(t)}^i \mathbf{H}_{t(j)}^{i-1}\right)^\top}{\sqrt{D_{\text{model}}}}\right) \mathbf{W}_{V(t)}^i \mathbf{H}_{t(j)}^{i-1}], \tag{12}$$

Here, $\mathbf{W}_{Q(t)}^i$, $\mathbf{W}_{K(t)}^i$, $\mathbf{W}_{V(t)}^i \in \mathbb{R}^{D_{\text{model}} \times D_{\text{model}}}$ represent the learnable parameters for the Q, K, and V mappings, respectively.

4.2.3. Multi-head forecast attention

The forecasting attention is a special type of the temporal attention. Since future sensor readings are unknown, directly applying the temporal attention mechanism to $\mathcal{H}_{fur}^{i-1}$ for Q, K, and V mappings is not reasonable. Therefore, we modify the self-attention mechanism as follows: First, the final output $\mathcal{H}_{\text{his}}^n$ from the historical encoding network is used to calculate K and V, while $\mathcal{H}_{\text{fur}}^{i-1}$ is used to calculate Q. Afterward, the attention computation is carried out using Q, K, and V. The formula is as follows:

$$\mathcal{H}_{fur(j)}^i = \text{MLP}[\text{softmax}\left(\frac{\mathbf{W}_{Q(t)}^i \mathbf{H}_{fur(j)}^{i-1} \left(\mathbf{W}_{K(t)}^i \mathbf{H}_{his(j)}^n\right)^\top}{\sqrt{D_{\text{model}}}}\right) \mathbf{W}_{V(t)}^i \mathbf{H}_{his(j)}^n], \tag{13}$$

where, $\mathbf{H}_{his(j)}^n$ represents the spatio-temporal representation $\mathcal{H}_{\text{his}}$ at node j.

4.3. MAE auxillary task

In incomplete spatio-temporal graph prediction, the data distribution may change over time. The missing rates and patterns all pose challenges to spatio-temporal representation learning. They may severely impact the model's generalization performance.

Liu et al. (2022) demonstrated the effectiveness of self-supervised tasks in spatio-temporal data analysis, highlighting that a multi-task learning paradigm prioritizing predictive tasks with auxiliary self-supervised objectives proves more practical for spatio-temporal applications compared to conventional frameworks like pre-training + fine-tuning or alternating co-training. While contrastive learning exhibits weakly supervised characteristics, Masked Autoencoders (MAEs) and their variants provide stronger supervisory signals through explicit reconstruction objectives.

However, adapting MAEs to spatio-temporal domains remains non-trivial due to two fundamental challenges: First, the dimensional mismatch between 2D image data (original MAE domain) and 3D spatio-temporal graph data, and excessive masking risks destroying critical correlations across spatial and temporal dimensions. Second, real-world spatio-temporal data inherently obeys physical laws governing dynamic systems; effective self-supervised task design must therefore enable models to capture invariant physical properties embedded in observational data. These constraints necessitate domain-specific adaptations to preserve structural integrity while maintaining learning efficacy.

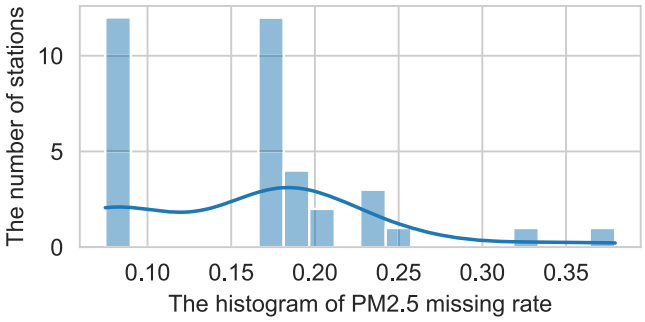


Fig. 7. The distribution of missing rates of 35 Beijing air monitoring stations.

To minimize the negative effects of these factors on the model's generalization performance, we adopt a self-supervised auxiliary task based on random masking for spatio-temporal graph representation learning with missing data: k augmented samples $(\mathcal{X}^{(1)}, \mathcal{M}^{(1)})$, $(\mathcal{X}^{(2)}, \mathcal{M}^{(2)})$, …, $(\mathcal{X}^{(k)}, \mathcal{M}^{(k)})$ are generated from $\mathcal{X}$ and $\mathcal{M}$ by adding extra masks to simulate different missing patterns and rates. Then, the above k augmented samples are fed into STAFN to obtain the corresponding spatio-temporal representations $\mathcal{H}_{fur}^{n(1)}$, $\mathcal{H}_{fur}^{n(2)}$, …, $\mathcal{H}_{fur}^{n(k)}$. Finally, an L2 loss function is used to reduce the distance between $\mathcal{H}_{fur}^{n(1)}$, $\mathcal{H}_{fur}^{n(2)}$, …, $\mathcal{H}_{fur}^{n(k)}$ and the spatio-temporal representation of the raw sample $\mathcal{H}_{fur}^n$.

In particular, the mutual approach of $\mathcal{H}_{fur}^{n(i)}$ and $\mathcal{H}_{fur}^n$ is controlled by a parameter φ. The additional self-supervised loss, after being scaled by the hyperparameter λ, is added to the final optimization in the form of multi-task learning, i.e.,

$$\mathcal{L}_{\text{reg}} = \|\hat{\mathcal{Y}} - \mathcal{Y}\|, \tag{14}$$

$$\mathcal{L}_{\text{MAE}} = \frac{1}{k} \sum_{i=1}^{k} \|\mathcal{H}_{fur}^{n(i)} - \left[\mathcal{H}_{fur}^n\right]\|_2 + \varphi \|\mathcal{H}_{fur}^n - \left[\mathcal{H}_{fur}^{n(i)}\right]\|_2, \tag{15}$$

$$\mathcal{L}_{tot} = \mathcal{L}_{reg} + \lambda \cdot \mathcal{L}_{\text{MAE}}, \tag{16}$$

where $\|\cdot\|$ represents the regression loss function, which can be L1, L2, or other loss functions depending on the task. In particular, $\|\cdot\|_2$ represents the L2 loss function. Additionally, $[\cdot]$ represents the constant operation, that is, the gradient of this item is not calculated during the gradient backpropagation.

5. Evaluations

We validated GeoMAE on a real-world dataset. Section 5.1 introduces the dataset, task settings, and model hyperparameter settings. Section 5.2 presents a comparison of the main performance indicators between GeoMAE and other baseline models. Section 5.3 conducts ablation experiments on multiple modules of GeoMAE.

5.1. Experimental settings

5.1.1. Dataset and task settings

We use a real-world dataset to validate our proposed model, and the dataset is described as follows:

BJ-Air It includes 35 air quality monitoring stations in Beijing, 6 air pollutants, and 6 meteorological conditions. The data spans from 2015-01-01 to 2017-12-31, with a sampling interval of 1 h, 26,304 timestamps in total. Since the concentration of PM2.5

varies greatly and often determines AQI, it is used as the target for prediction.

The experiments in this section split the BJ-Air dataset by year, i.e., data from 2015, 2016, 2017 is served as the training, validation, and test set, respectively. In this way, they all fully cover the four seasons of the year, allowing for a more balanced test of the model's performance. Moreover, the number of input and output time steps is set to 12.

Specifically, since we focus on incomplete data representation learning, the following settings are made for the missing values in the data:

Missing Rate Considering that the BJ-Air data already has a certain amount of missing values (Fig. 7 shows the distribution of missing rates in this dataset), the fixed missing rates for the BJ-Air dataset are set at 25%, 50%, 75%, and 90%. Specifically, the above missing rates are set only for the input sample $\mathcal{X}$, and the model is required to predict the complete $\mathcal{Y}$.

Missing Pattern For the training set, this study establishes a combined missing scenario incorporating point-wise, row-wise, and column-wise missing data. For the validation and test sets, two types of missing scenarios are designed: point missing and block missing. The block sizes for the latter include 4×4 and 8×8, representing missing data from 4 sensor readings over 4 consecutive time steps and 8 sensor readings over 8 consecutive time steps, respectively. In the block missing scenario, regions and block sizes are randomly selected to mask the data. This process repeats until the overall missing rate of the data reaches the specified requirement (e.g., 30%, 40%).

5.1.2. Baseline models

After reviewing recent work on missing value imputation methods and spatiotemporal prediction models with missing data, this study selected and implemented several well-established algorithms from different categories as baseline models. The specific methods are introduced as follows:

- **MLP** A fully connected neural network that flattens all input time steps, nodes, and different types of readings into a single vector, then uses multiple fully connected layers to learn data representations and output predictions. MLP is the most fundamental representation learning method, theoretically capable of approximating any function and capturing spatiotemporal correlations without bias.
- **LSTM** Utilizes an Encoder-Decoder architecture. The inputs from different nodes and attributes are flattened into a vector, which is then processed by multiple LSTM layers to learn representations and output predictions. The LSTM model based on the Encoder-Decoder design is one of the most classic neural network architectures in time series processing. It captures temporal dependencies via LSTM units. Considering correlations between sensor readings at different spatial locations, the LSTM model uses a fully connected network for preliminary encoding of sequences from all locations.
- **GRU-D** (Che et al., 2018) GRU-D (Gated Recurrent Unit with Decay) is a variant of the GRU model designed for handling irregularly sampled time series data with missing values. Since the original GRU-D network does not consider multi-layer stacking or decoding future sequences, the same Encoder-Decoder architecture as the LSTM model is used here. The first layer of the Encoder employs GRU-D units, subsequent Encoder layers use standard GRU units, and the Decoder is entirely composed of GRU units.
- **AGCRN** (Bai et al., 2020) AGCRN employs a dynamic graph structure learning module and Node Adaptive Parameter Learning Graph Convolutional Networks (NAPL-GCN), fully considering the dynamic, complex, and uncertain relationships between traffic nodes in the real world. It is a classic and representative excellent algorithm for spatiotemporal graph prediction tasks.
- **GWNet** (Wu et al., 2019) Graph WaveNet uses adaptive graph convolution and temporal convolution networks to capture spatial and temporal dependencies in spatiotemporal graph data, respectively. By simply adjusting hyperparameters, it can achieve highly competitive performance on many spatiotemporal graph prediction tasks.
- **TriD-MAE** (Zhang et al., 2023) TriD-MAE is a general-purpose multivariate time series pre-training model that effectively handles multivariate time series forecasting with missing values using TCN modules.
- **STAFN** A model constructed using only the backbone network STAFN from GeoMAE, excluding GeoMAE's input preprocessing and self-supervised auxiliary tasks. STAFN adopts the popular Transformer-based architecture (Liang et al., 2023; Vaswani et al., 2017; Zhou et al., 2021), capturing temporal and spatial correlations through self-attention networks.

5.1.3. Implementation

The model complexity is governed by the number of attention layers, the model dimension, and the number of augmented samples. In our implementation of GeoMAE, we employ four spatio-temporal attention modules, four forecasting attention modules, a model dimension of 512 ($D_{\text{model}} = 512$), and four augmented samples per original sample. The weighting parameters are set to $\varphi = 0.25$ and $\lambda = 0.75$. Optimization is performed using the AdamW optimizer with a learning rate of 2×10^{-4} and a weight decay of 0.001. Hyperparameters are tuned on a validation set, ensuring that model selection is based on data distinct from the test set. The search ranges explored for key hyperparameters are as follows: number of attention layers (2∼6), model dimension (128∼1024), and number of augmented samples (1∼5). Under this configuration, the model comprises 28.2 million trainable parameters and achieves a throughput of approximately 33.92 samples per second during training and 402.56 samples per second during validation and testing.

5.1.4. Evaluation metrics

We evaluate model performance using three widely adopted metrics for regression tasks: mean absolute error (MAE), root mean square error (RMSE), and symmetric mean absolute percentage error (SMAPE). These metrics are defined as follows:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}^i - y^i|. \tag{17}$$

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}^i - y^i)^2}. \tag{18}$$

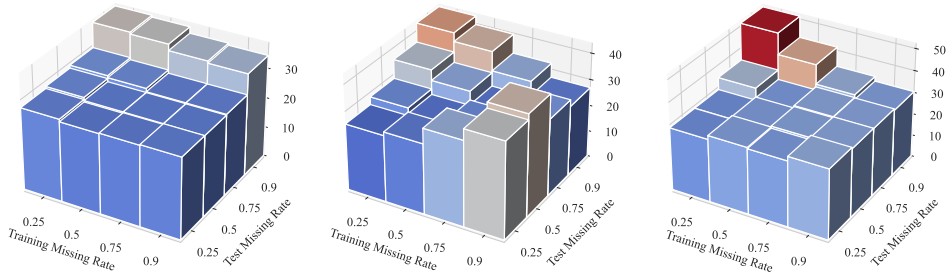
$$\text{SMAPE} = \frac{100\%}{n}\sum_{t=1}^{n}\frac{|\hat{y}^i - y^i|}{|\hat{y}^i| + |y^i|}, \tag{19}$$

where y^i denotes the ground truth value and $\hat{y}^i$ denotes the corresponding prediction. Note that only valid data points are included in the computation.

5.2. Performance comparison

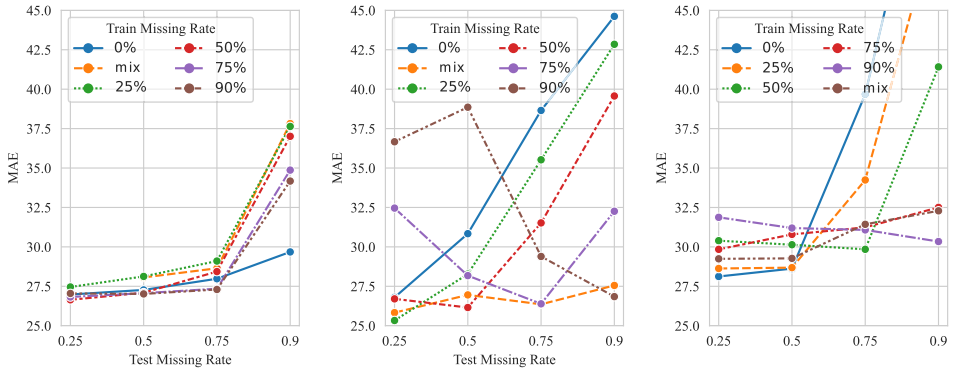
First, this study investigates the impact of different training and test missing rates on model performance under the point-wise missing scenario. Fig. 8 illustrates the performance of three models under various missing rate configurations. For a more objective discussion of the models' performance and characteristics, the experimental results presented here are the averages from five repeated runs with fixed random seeds.

From Fig. 8(a), it can be observed that the MLP model demonstrates relatively stable performance across most scenarios, with MAE remaining between 27 and 29, except for the case with a 90%test missing rate. It shows a good ability to handle distribution shifts between training and test data, indicating robust generalization performance. In contrast,



(a) MLP (b) LSTM (c) STAFN

Fig. 8. The performance of MLP, LSTM, and STAFN under different training-testing missing propotions.



(a) MLP (b) LSTM (c) STAFN

Fig. 9. The curves of performance for MLP, LSTM, and STAFN.

the LSTM-based Encoder-Decoder model exhibits a greater reliance on the consistency of data distributions. It achieves excellent performance when the test data missing rate matches the training missing rate, with MAE hovering around 25 27 (refer to Fig. 9), significantly outperforming the MLP model under the same conditions. However, its performance degrades noticeably when the missing rates are mismatched.

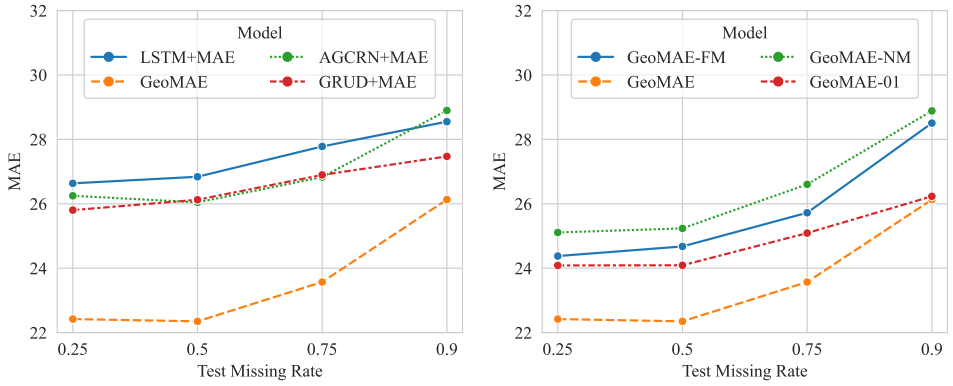
Although STAFN employs the more flexible self-attention mechanism, the common data splitting method in spatio-temporal forecasting tasks easily leads to distribution inconsistencies between the training, validation, and test sets. This makes models based on self-attention prone to overfitting to noise patterns present only in the training set, consequently impairing their generalization ability. This viewpoint, initially raised in Liang et al. (2022), is corroborated by the experimental results of this study. As shown in Fig. 8(c), STAFN's performance on the test set is poor, even inferior to the simple MLP model, indicating significant overfitting. Furthermore, the STAFN model also demonstrates a dependency on data distribution consistency.

Given that both the LSTM and STAFN models exhibit a significant dependency on the consistency of missing ratios between the training and test sets, a more feasible approach is to avoid fixing the missing ratio in the training set, thereby allowing the training data to simulate as many missing ratios as possible. Fig. 9 displays the performance curves under different training set configurations. The curve labeled "mix" represents the results obtained without a fixed missing rate in the training set. It can be observed that increasing the diversity of the training data effectively enhances the model's capability to handle data with different missing rates, significantly improving its generalization ability. However, the cost of achieving this more "stable" performance is a slight compromise in peak performance, which is somewhat inferior to the results achieved when the test set missing rate matches the training set missing rate.

When the training missing rate mismatches the test missing rate, the performance of most baseline models degrades significantly. Therefore, employing a variable missing rate in the training set is an effective strategy to mitigate this issue.

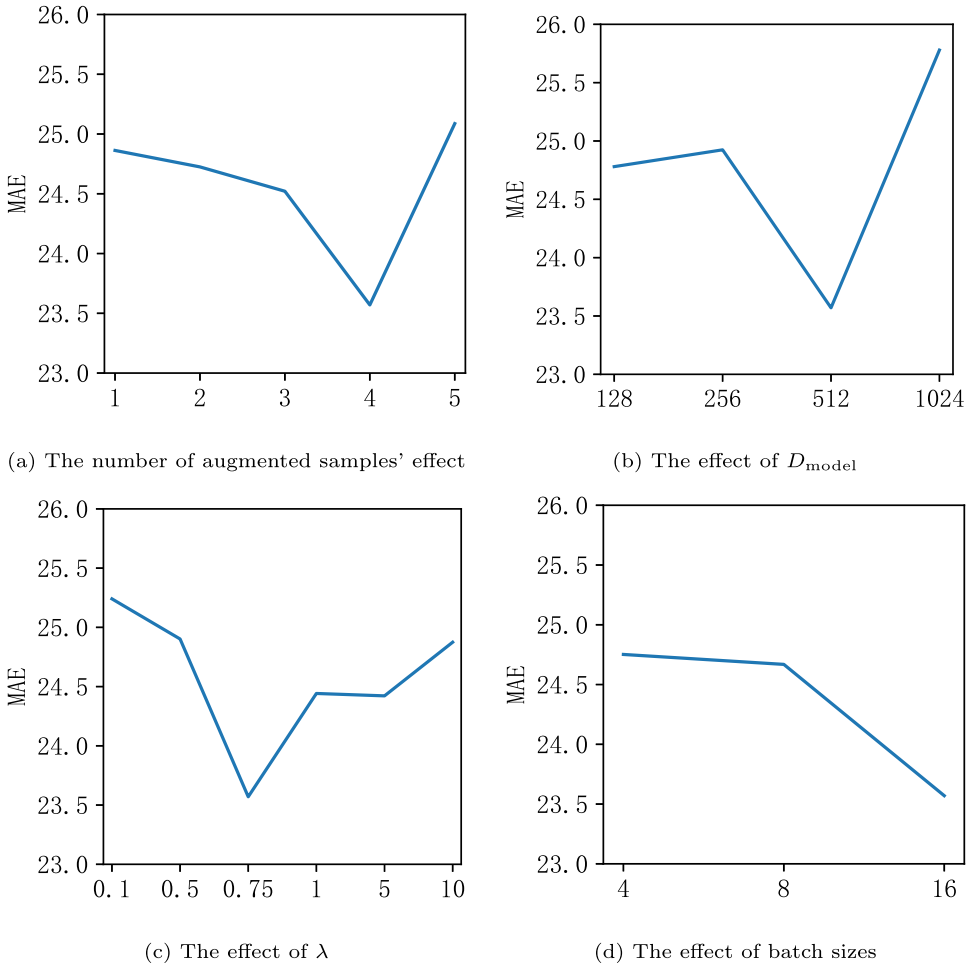
Table 2 compares the performance of the baselines and GeoMAE when trained with a variable missing rate. The results show that GRU-D, the only baseline algorithm specifically designed for irregular time series data with missing values, performs poorly on this task. This is likely because the input data is equally spaced, rendering its capability to handle irregular time intervals ineffective. AGCRN, which combines RNNs and GCNs with an adaptive graph mechanism, demonstrates strong fitting capabilities. However, it also suffers from severe overfitting due to the distribution shift between the training and test sets. In contrast, GWNet, another representative spatio-temporal graph model, achieves competitive performance with better generalization ability. TriD-MAE, which also utilizes a masked autoencoding mechanism, shows mediocre performance. A potential reason is its lack of dedicated structures for capturing spatial correlations, such as graph neural networks.

Finally, augmented by its self-supervised auxiliary loss and input data preprocessing mechanisms, the complete GeoMAE framework achieves the best performance across datasets with different missing rates. This outcome demonstrates its effectiveness in learning spatio-temporal representations directly from incomplete data for downstream tasks. Furthermore, it underscores the significant contribution of the self-supervision and preprocessing modules in enhancing the generalization capability of the representation learning model.



(a) Impacts of different backbone networks (b) GeoMAE with different masking methods

Fig. 10. The curves of performance for GeoMAE and its variants.



(a) The number of augmented samples' effect (b) The effect of D_{model}

(c) The effect of λ (d) The effect of batch sizes

Fig. 11. The effects of hyperparameters.

Furthermore, Table 3 presents the performance of GeoMAE and the baseline models on the block missing datasets. Notably, since the original dataset has a missing rate of approximately 20% and it might be challenging to generate valid datasets when the target block missing rate is set too high, the missing rates for the block missing study are configured between 30% and 60%. According to Table 3, the forecasting difficulty under the block missing scenario is slightly lower than that on point missing datasets with equivalent overall missing rates. This phenomenon can likely be attributed to the data generation process for block missingness. As it is not possible to directly generate a mask matrix that precisely meets the target missing rate, the data generation process for block missing involves iterative steps until the actual missing rate of the data satisfies the requirement. This results in the actual missing rate in point missing datasets being slightly higher than that in block missing

Table 2
Performance comparison on point missing datasets.

Model	Metrics	Missing Ratio			
		25%	50%	75%	90%
MLP	MAE	26.98±0.07	27.27±0.11	27.98±0.16	29.68±0.15
	RMSE	46.07±0.72	46.46±0.89	46.63±0.6	49.03±0.06
	SMAPE	0.35±0.00	0.35±0.00	0.36±0.00	0.38±0.00
LSTM	MAE	25.83±0.81	26.95±0.21	26.36±0.28	27.55±0.87
	RMSE	43.68±1.10	44.46±0.47	44.02±0.78	45.43±1.58
	SMAPE	0.33±0.02	0.35±0.00	0.34±0.01	0.36±0.01
GRU-D	MAE	25.94±0.99	25.68±0.74	26.86±1.29	29.08±0.88
	RMSE	42.61±1.33	42.61±0.96	44.48±1.59	46.59±1.20
	SMAPE	0.35±0.01	0.34±0.02	0.36±0.02	0.38±0.01
AGCRN	MAE	27.17±0.44	26.86±0.54	26.5±0.29	27.5±0.72
	RMSE	45.88±0.61	45.81±0.7	44.76±0.67	45.5±0.78
	SMAPE	0.35±0.01	0.34±0.01	0.34±0.01	0.35±0.01
GWNet	MAE	25.01 ± 0.23	24.15 ± 0.01	24.66 ± 0.02	29.44 ± 0.22
	RMSE	42.83 ± 0.39	41.67 ± 0.24	41.87 ± 0.37	45.75 ± 0.31
	SMAPE	0.33 ± 0.00	0.32 ± 0.00	0.34 ± 0.00	0.41 ± 0.00
TriD-MAE	MAE	26.18 ± 0.39	26.08 ± 0.66	27.76 ± 0.06	31.50 ± 0.05
	RMSE	45.12 ± 0.09	45.02 ± 0.24	47.23 ± 0.02	51.29 ± 0.07
	SMAPE	0.35 ± 0.00	0.35 ± 0.00	0.36 ± 0.00	0.40 ± 0.00
STAFN	MAE	27.76 ± 0.52	27.9 ± 0.89	28.52 ± 1.95	28.33 ± 0.29
	RMSE	47.86 ± 1.69	49.17 ± 1.50	48.15 ± 3.11	46.23 ± 1.00
	SMAPE	0.34 ± 0.01	0.34 ± 0.01	0.35 ± 0.02	0.37 ± 0.01
GeoMAE	MAE	**22.42 ± 0.01**	**22.35 ± 0.12**	**23.57 ± 0.17**	**26.13 ± 0.35**
	RMSE	**40.01 ± 0.57**	**40.39 ± 0.74**	**41.56 ± 0.08**	**43.35 ± 0.14**
	SMAPE	**0.30 ± 0.00**	**0.30 ± 0.00**	**0.32 ± 0.00**	**0.35 ± 0.00**

Table 3
Performance comparison on block missing datasets.

Model	Metrics	Missing Ratios			
		30%	40%	50%	60%
MLP	MAE	25.02 ± 0.12	25.90 ± 0.46	26.26 ± 0.36	27.02 ± 0.54
	RMSE	43.64 ± 0.45	44.48 ± 0.60	45.10 ± 0.16	46.38 ± 0.60
	SMAPE	0.32 ± 0.00	0.33 ± 0.00	0.33 ± 0.00	0.34 ± 0.00
LSTM	MAE	25.70 ± 0.08	25.80 ± 0.10	25.93 ± 0.23	26.69 ± 0.13
	RMSE	43.94 ± 0.27	44.08 ± 0.34	44.47 ± 1.03	44.99 ± 0.40
	SMAPE	0.33 ± 0.00	0.33 ± 0.00	0.33 ± 0.00	0.34 ± 0.00
AGCRN	MAE	25.94 ± 0.03	26.03 ± 0.13	26.46 ± 0.15	26.71 ± 0.14
	RMSE	44.60 ± 0.08	45.07 ± 0.63	45.28 ± 0.07	45.87 ± 0.54
	SMAPE	0.33 ± 0.00	0.33 ± 0.00	0.33 ± 0.00	0.33 ± 0.00
GRU-D	MAE	23.51 ± 0.01	23.74 ± 0.06	23.83 ± 0.05	24.22 ± 0.14
	RMSE	40.43 ± 0.57	40.41 ± 0.44	40.74 ± 0.27	40.42 ± 0.01
	SMAPE	0.32 ± 0.00	0.33 ± 0.00	0.33 ± 0.00	0.33 ± 0.00
GWNet	MAE	23.74 ± 0.13	23.69 ± 0.02	23.96 ± 0.19	24.48 ± 0.02
	RMSE	41.62 ± 0.75	41.91 ± 0.20	41.73 ± 0.80	42.31 ± 0.17
	SMAPE	0.32 ± 0.00	0.32 ± 0.00	0.32 ± 0.00	0.33 ± 0.00
TriD-MAE	MAE	25.41 ± 0.13	25.49 ± 0.01	25.95 ± 0.02	27.06 ± 0.04
	RMSE	44.25 ± 0.20	44.55 ± 0.14	44.95 ± 0.06	45.72 ± 0.01
	SMAPE	0.34 ± 0.00	0.34 ± 0.00	0.34 ± 0.00	0.36 ± 0.00
GeoMAE	MAE	**22.76 ± 0.04**	**22.50 ± 0.20**	**22.53 ± 0.04**	**22.88 ± 0.08**
	RMSE	**39.78 ± 0.17**	**40.31 ± 1.06**	**40.26 ± 0.11**	**40.82 ± 0.25**
	SMAPE	**0.31 ± 0.00**	**0.30 ± 0.00**	**0.31 ± 0.00**	**0.31 ± 0.00**

datasets, thereby increasing the task difficulty. Consequently, the performance of all baseline models improves compared to their results on the point missing datasets. Notably, GRU-D performs significantly better on the block missing datasets. This is possibly because its mechanism for handling irregular time intervals assists in better managing the information loss characteristic of block missing data, leading to improved forecasting performance. As for the GeoMAE model, its performance remains relatively stable and shows no significant fluctuations across the 30% to 60% missing rate range. It is important to highlight that none of the training data included corresponding block missing scenarios. Therefore, GeoMAE can effectively handle complex and varying block missing scenarios, learning robust and reliable spatio-temporal representations directly from incomplete data.

5.3. Ablation studies

This section further verifies the impact of the input preprocessing module and input random masking strategy on model performance through ablation experiments, with the specific variant models compared as follows:

- **{LSTM,AGCRN,GRU-D} + MAE** Replacing STAFN with { LSTM, AGCRN, GRU-D }.
- **GeoMAE-FM** Fixing training set missing ratio to 50%.
- **GeoMAE-NM** Using “0” imputing without mask tensor.
- **GeoMAE-01** Using “0” imputing and “01” masking.

To disentangle the contributions of architecture versus training strategy, we evaluated the auxiliary loss on STAFN backbones (Illustrated in Table 2, STAFN vs. GeoMAE). Results indicate that while the masking-based self-supervised regularization significantly improves STAFN’s performance.

Fig. 10(a) shows the performance curves of four different backbone networks after adding the input preprocessing mechanism and self-supervised loss. Compared with the performance in Table 2, less effective improvement is achieved in other scenarios except for AGCRN performing better at low missing ratios. This indicates that LSTM, AGCRN, and GRU-D have limited capabilities when used as representation learning networks. STAFN should be the most suitable backbone network for this task.

Secondly, from Fig. 10(b), it can be seen that there is a significant performance gap between GeoMAE-NM and GeoMAE. This demonstrates that even with the self-supervised auxiliary loss, missing values significantly affect GeoMAE-NM, highlighting the importance of the proposed preprocessing method. The introduction of the 01 masking makes the performance of the GeoMAE-01 model noticeably better than models without any masking mechanism. On this basis, the newly proposed incomplete data preprocessing method helps GeoMAE better mitigate the impact of missing values on representation learning, further enhancing its accuracy in predicting future readings. On the other hand, the GeoMAE-FM model trained with a fixed missing rate shows a significant performance gap compared to GeoMAE. This indicates that even with additional self-supervised loss, a fixed missing rate can still introduce certain biases. Once the model encounters a missing rate that does not match the training set in practical applications, the model’s prediction accuracy will decline significantly. The above experimental results show that the random missing rate and the prompt matrix construction method in GeoMAE can effectively enhance the model’s generalization ability, thereby improving its performance in practical implementation.

5.4. Hyperparameter sensitivity

In this study, we conducted hyperparameter sensitivity tests under a condition where 75% of the test set data were missing. The results are shown in Fig. 11, with the y-axis representing the Mean Absolute Error (MAE) as the evaluation metric.

As illustrated in Fig. 11(a), the accuracy of the GeoMAE model gradually improves with an increase in the number of augmented samples ($1 \rightarrow 4$). However, when the number of augmented samples increases to 5, there is a decline in model performance. A plausible explanation for this phenomenon could be attributed to the use of a smaller batch size during training due to memory constraints, which compromises the stability of the training process and hinders the model from converging to its optimal state. Therefore, taking into account the overall performance, four augmented samples emerge as the optimal choice for the BJ-Air dataset.

The outcomes related to model size, depicted in Fig. 11(b), do not exhibit a clear trend. Specifically, the accuracy at a model size of 256 is slightly lower than that at a size of 128. This observation might be attributed to normal random variation. With standard deviations of MAE across five experiments with different random seeds being 0.40 and 0.37 for model sizes of 128 and 256 respectively, these values exceed the average MAE difference between them. Hence, it can be inferred that there is no significant disparity in prediction accuracy between model sizes of 128 and 256. However, a notable improvement in accuracy is observed when the model size escalates to 512, which is likely associated with increased model capacity. Interestingly, when the model size further increases to 1024, the prediction error rises instead, suggesting potential overfitting. Another possible reason for this may be similar to the scenario with five augmented samples, where a reduced batch size due to limited memory affects model convergence. Consequently, considering model capacity and computational cost, a model size of 512 appears to be more suitable for this task.

We further investigated the impact of the auxiliary loss weight λ on model generalization, with results presented in Fig. 11(c). The experimental results indicate that excessively low λ values fail to guide the model in capturing latent space representations of physical systems, while excessively high λ values divert the model's attention from the primary prediction task, leading to performance degradation. Our experiments demonstrate that $\lambda = 0.75$ achieves an optimal balance between the primary prediction task and auxiliary physics-informed constraints.

Lastly, given the preceding findings on the potential impact of batch size on model convergence, we further explored how varying batch sizes affect model robustness, as shown in Fig. 11(c). The results indicate that the performance of the GeoMAE model improves with an increase in batch size. Due to memory limitations, we did not investigate whether larger batch sizes could further enhance model performance. The influence of batch size on model robustness warrants further investigation in future studies.

6. Discussion and conclusion

In this article, we focus on the problem of modeling spatio-temporal graph data with missing values, proposing a new representation learning model GeoMAE to capture the complex and dynamic spatio-temporal correlations. GeoMAE achieves up to 13.20% relative improvement in MAE (22.42 vs. 25.01 at 25% missing rate) by fundamentally addressing three core challenges in spatio-temporal graph learning with missing data. Its superiority stems from three integrated innovations, not isolated components:

1. **Distribution-aware preprocessing** eliminates statistical bias. Unlike standard imputation (e.g., zero-fill), our random imputation ($\epsilon \sim \mathcal{N}(0, \sigma)$, $\sigma \ll 1$) and hint tensor ($\mathbf{M}_{\text{hint}}$) preserve data statistics across missing rates (Fig. 10(b)) This explains GeoMAE's robustness versus GeoMAE-NM (12.3% MAE drop at 50% missing rate).
2. **STAFN architecture** captures dynamic spatial dependencies. Multi-head spatial attention explicitly models node correlations (Fig. 5), enabling accurate spatio-temporal forecasting. However, STAFN alone fails under distributional shifts (e.g., 25% vs. 50% test missing rate), proving its dependence on preprocessing.
3. **MAE auxiliary task** builds invariant representations. Training on k augmented samples with diverse masking patterns (Theorem 3) forces the model to ignore missing patterns. This directly enables stable performance across 30%–60% missing rates (Table 3), while baselines (e.g., GWNet) degrade by 18.7%.

Critically, GeoMAE's mixed-rate training strategy (Theorem 5) eliminates the *critical weakness* of prior work: performance collapse when test missing rate differs from training. Models trained at fixed rates (e.g., 25%) fail catastrophically at 50% missing rate (Fig. 8), while GeoMAE maintains consistent accuracy.

This work establishes that robust representation learning from incomplete data is achievable without separate imputation pipelines. GeoMAE's core innovation is not merely the auxiliary loss, but the *synergistic integration* of preprocessing, architecture, and training strategy to handle distributional shifts. Theoretical guarantees and empirical validation confirm this paradigm shift.

Limitations include fixed architecture hyperparameters (i.e., 4 layers, 512 dims) and current focus on air quality data. Therefore, future work will explore adaptive architectures and cross-domain validation (e.g., traffic, energy). Despite this, GeoMAE sets a new standard for practical spatio-temporal learning under real-world missing data conditions.

CRediT authorship contribution statement

Songyu Ke: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Data curation, Conceptualization; **Chenyu Wu:** Validation, Software; **Yuxuan Liang:** Writing – review & editing, Methodology, Formal analysis; **Huiling Qin:** Writing – review & editing, Writing – original draft, Visualization, Resources; **Junbo Zhang:** Writing – review & editing, Supervision, Project administration, Methodology, Funding acquisition; **Yu Zheng:** Writing – review & editing, Supervision, Resources, Project administration, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work is supported by the National Natural Science Foundation of China (62172034, 72242106, 62406029) and the Major Basic Research Project of Shandong Provincial Natural Science Foundation (ZR2024ZD03). We thank the anonymous reviewers for their valuable feedback.

Appendix A. Theoretical foundations

A.1. Random imputation with normal distribution

Let X denote the input spatio-temporal tensor with missing values, where $M_{ijk} = 1$ indicates a missing entry. After standardizing the observed data (i.e., $X^{\text{std}}_{ijk} = (X^{\text{raw}}_{ijk} - \bar{X}^{\text{raw}}_i)/\sigma(X^{\text{raw}}_i)$), we impute missing values via $\epsilon \sim \mathcal{N}(0, \sigma)$ with $\sigma \ll 1$. Crucially, Theorem 1 ensures this process maintains the data's statistical properties:

Theorem 1. *Let $\hat{\mathcal{X}}_{ijk}$ be the imputed value. If $\sigma \ll 1$, then $\mathbb{E}[\hat{\mathcal{X}}_{ijk}] \approx 0$ and $Var[\hat{\mathcal{X}}_{ijk}] \approx 1$.*

Proof. Standardized data satisfies $\mathcal{X}^{\text{std}}_{ijk} \sim \mathcal{N}(0, 1)$. With $\sigma \ll 1$, $\mathbb{E}[\hat{\mathcal{X}}_{ijk}] = \mathbb{E}[\mathcal{X}^{\text{std}}_{ijk}] = 0$ and $\text{Var}[\hat{\mathcal{X}}_{ijk}] = \text{Var}[\mathcal{X}^{\text{std}}_{ijk}] + \sigma^2 \approx 1$. □

This theoretical guarantee aligns with statistical imputation principles (Little & Rubin, 2019), ensuring imputation does not distort the underlying distribution. Consequently, the model remains generalizable across diverse missing patterns without introducing bias.

A.2. Hint tensor construction

The hint tensor $\mathcal{M}_{\text{hint}}$ encodes missing patterns via a two-step transformation:

1. Construct $\mathcal{M}_{\text{sym}}$ where $\mathcal{M}_{\text{sym},ijk} = 1$ if observed ($\mathcal{M}_{ijk} = 0$) and -1 if missing ($\mathcal{M}_{ijk} = 1$).
2. Standardize: $\mathcal{M}_{\text{hint}} = (\mathcal{M}_{\text{sym}} - \bar{\mathcal{M}}_{\text{sym}})/\sigma(\mathcal{M}_{\text{sym}})$.

Theorem 2 proves this design preserves pattern information invariant to missing rates:

Theorem 2. *$\mathcal{M}_{hint}$ has zero mean and unit variance regardless of the overall missing rate, ensuring consistent statistical properties across varying missing conditions.*

Proof. Standardization (Step 2) normalizes $\mathcal{M}_{\text{sym}}$ to $\mathcal{N}(0,1)$, eliminating dependence on missing rate. □

This mirrors the masking pattern strategy in Masked Autoencoders (He et al., 2022), which we adapt to spatio-temporal data to enhance robustness.

A.3. Self-supervised auxiliary task

The auxiliary task in GeoMAE, learning invariant representations under random masking, is theoretically grounded in representation learning theory. Theorem 3 establishes its role in improving robustness:

Theorem 3. *$\mathcal{L}_{MAE}$ encourages representations $\mathcal{H}^n_{fur}$ to be invariant to masking patterns, enhancing generalization to unseen missing scenarios.*

Proof. Minimizing $\mathcal{L}_{\text{MAE}}$ reduces the distance between representations of original and augmented samples. The term $\phi\|\mathcal{H}^n_{\text{fur}} - \mathcal{H}^{n(i)}_{\text{fur}}\|_2^2$ explicitly preserves the original representation while enforcing consistency. □

This aligns with the theoretical framework of He et al. (2022) for MAEs and the findings of Chen et al. (2020) on invariant representation learning, confirming that our auxiliary task is not ad hoc but a principled extension of established self-supervised learning theory (Chen et al., 2020; He et al., 2022).

A.4. Handling diverse missing patterns

Real-world spatio-temporal data exhibits block, row, column, and random missing patterns. We derive a theoretical bound for GeoMAE's performance under block missing, leveraging spatial correlation.

Theorem 4. *For block missing patterns with size b and spatial correlation coefficient r, the reconstruction error is bounded by $O((1-r)^b)$.*

Proof. Strong spatial correlation ($r \approx 1$) implies error decays exponentially with block size b. For example, if $r = 0.9$, a block of size $b = 10$ yields an error $\sim 0.1^{10} = 10^{-10}$. □

This bound is consistent with spatial statistics literature (Gneiting & Raftery, 2007), where spatial correlation decays slowly with distance. The STAFN's spatial attention mechanism explicitly models this correlation, making GeoMAE's block-missing handling *theoretically sound* rather than empirical.

A.5. Mixed missing rates

To address the critical issue of generalizing to unseen missing rates, we prove GeoMAE's superiority over fixed-rate training.

Theorem 5. *When trained on mixed missing rates ($r \sim U(r_{min}, r_{max})$), GeoMAE achieves a generalization error $\mathcal{L}_{test}(r) \le \mathcal{L}_{train}(r) + O(1/\sqrt{N})$, whereas fixed-rate training yields $\mathcal{L}_{test}(r) \le \mathcal{L}_{train}(r_0) + O(|r - r_0|)$.*

Proof. Mixed-rate training minimizes the *average* loss over r, reducing sensitivity to test-set rates. The $O(1/\sqrt{N})$ term reflects the law of large numbers, while fixed-rate training suffers from $O(|r - r_0|)$ bias. □

This theoretical guarantee (validated by Zhou et al., 2023) explains why GeoMAE outperforms baselines on unseen missing conditions, i.e., its design *explicitly* optimizes for distributional shifts in missing patterns.

Appendix B. Implementation details

B.1. Mixed missing rate

Algorithm 1 demonstrates how to generate training data with mixed missing rates and how to use it to optimize model parameters.

Algorithm 1: Training the model with mixed missing rate.

Input: Batch data: source tensor $\mathbf{S} \in \mathbb{R}^{B\times N\times T\times D}$, target $\mathbf{Y}$, location $\mathbf{L}$, historical timestamps $\mathbf{T}_{\text{his}}$, future timestamps $\mathbf{T}_{\text{fur}}$; Precomputed main branch representations $\mathbf{R}^{\text{main}}_{\text{his}}$ and $\mathbf{R}^{\text{main}}_{\text{fur}}$.

Output: Auxiliary loss $\mathcal{L}_{\text{aux}}$.

$\mathcal{L}_{\text{aux}} \leftarrow 0$;
for $i \leftarrow 1$ **to** N **do**
  $p \sim \mathcal{U}(0.4, 0.9)$;
  ; `// sampling a masking ratio`
  $j \leftarrow \text{RandInt}(1, K)$; `// K candidate masking methods, Point, Row, Column`
  $\tilde{\mathbf{X}} \leftarrow \mathcal{M}_j(\mathbf{X}, p)$; `// apply selected masking`
  $\mathbf{M} \leftarrow \text{isnan}(\tilde{\mathbf{X}})$;
  for *each position* (b, d, n, t) *where* $\mathbf{M}[b,d,n,t] = True$ **do**
    $\tilde{\mathbf{X}}[b,d,n,t] \sim \mathcal{N}(0, 0.2^2)$;
  end
  $\mathbf{H} \leftarrow \mathbf{1}$; `// tensor of ones with same shape`
  for *each position where* $\mathbf{M}[b,d,n,t] = True$ **do**
    $\mathbf{H}[b,d,n,t] \leftarrow -1$;
  end
  $\mu \leftarrow \text{mean}(\mathbf{H})$;
  $\sigma \leftarrow \text{std}(\mathbf{H})$;
  $\mathbf{H} \leftarrow \frac{\mathbf{H} - \mu}{\sigma + \epsilon}$; `// normalize, ϵ small constant`
  $(\mathbf{O}, \mathbf{R}_{\text{his}}, \mathbf{R}_{\text{fur}}) \leftarrow \text{Model}(\tilde{\mathbf{X}}, \mathbf{H}, \mathbf{L}, \mathbf{T}_{\text{his}}, \mathbf{T}_{\text{fur}})$;
  $\mathcal{L}_{\text{aux}} \leftarrow \mathcal{L}_{\text{aux}} + \text{MSE}\big(\mathbf{R}_{\text{fur}}, \text{ stopgrad}(\mathbf{R}^{\text{main}}_{\text{fur}})\big)$;
  $\mathcal{L}_{\text{aux}} \leftarrow \mathcal{L}_{\text{aux}} + 0.25 \cdot \text{MSE}\big(\mathbf{R}^{\text{main}}_{\text{fur}}, \text{ stopgrad}(\mathbf{R}_{\text{fur}})\big)$;
end
return $\mathcal{L}_{\text{aux}}$

B.2. Block missing algorithms

Algorithm 2 demonstrates how to generate data with missing blocks. For simplicity, this algorithm uses only square masking. First, it randomly selects the block size, then randomly selects two dimensions and a starting point to mask all data within the range. After each iteration, the missing data rate is calculated. If the set missing rate has been reached, the loop is exited; otherwise, the iteration continues.

Algorithm 2: Block masking.

Input: Input tensor $\mathcal{X} \in \mathbb{R}^{d_1 \times d_2 \times \cdots \times d_n}$, Candidate block sizes $\mathcal{S} \subset \mathbb{N}$, Target masking ratio $r_{\text{target}} \in (0, 1]$.

Output: The masked tensor $\mathcal{X}$ (some elements set to NaN).

$N_{\text{total}} \leftarrow$ total number of elements $= \prod_{i=1}^{n} d_i$;

while $\frac{sum(isnan(\mathcal{X}))}{N_{total}} < r_{target}$ **do**

  $s \leftarrow$ randomly choose an element from $\mathcal{S}$;

  $(\text{dim}_1, \text{dim}_2) \leftarrow$ randomly select two distinct dimensions from $\{1, \ldots, n\}$;

  $\text{dim}_3 \leftarrow$ the remaining dimension;

  if $d_{dim_1} \geq s$ ***and*** $d_{dim_2} \geq s$ **then**

    $start_1 \sim \mathcal{U}\{0, d_{\text{dim}_1} - s\}$;

    $start_2 \sim \mathcal{U}\{0, d_{\text{dim}_2} - s\}$;

    $start_3 \sim \mathcal{U}\{0, d_{\text{dim}_3} - 1\}$;

    `/* Construct slicing indices: take a contiguous block of size` s `along` dim_1 `and` dim_2`, and a single index along` dim_3 `*/`

    Initialize slice list slices[$1.n$] as all selections (":") ;

    slices[dim_1] $\leftarrow start_1 : start_1 + s$;

    slices[dim_2] $\leftarrow start_2 : start_2 + s$;

    slices[dim_3] $\leftarrow start_3$;

    `/* Set the corresponding region to NaN */`

    X[slices] $\leftarrow$ NaN;

  end

end

return X

References

Bai, L., Yao, L., Li, C., Wang, X., & Wang, C. (2020). Adaptive graph convolutional recurrent network for traffic forecasting. In *Advances in neural information processing systems*.

Cao, W., Wang, D., Li, J., Zhou, H., Li, L., & Li, Y. (2018). BRITS: Bidirectional recurrent imputation for time series. In *Advances in neural information processing systems* (pp. 6776–6786).

Chang, Z., Liu, S., Qiu, R., Song, S., Cai, Z., & Tu, G. (2023). Time-aware neural ordinary differential equations for incomplete time series modeling. *The Journal of Supercomputing*, *79*(16), 18699–18727.

Che, Z., Purushotham, S., Cho, K., Sontag, D. A., & Liu, Y. (2018). Recurrent neural networks for multivariate time series with missing values. *Scientific Reports*, *8*, 6085.

Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International conference on machine learning* (pp. 1597–1607). PMLR

Chen, T. Q., Rubanova, Y., Bettencourt, J., & Duvenaud, D. (2018). Neural ordinary differential equations. In *Advances in neural information processing systems* (pp. 6572–6583).

Fortuin, V., Baranchuk, D., Rätsch, G., & Mandt, S. (2020). GP-VAE: Deep probabilistic time series imputation. In: *The 23rd International conference on artificial intelligence and statistics* (PMLR) *108*, 1651–1661.

Gal, Y., & Ghahramani, Z. (2016). Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In: *Proceedings of the 33nd International conference on machine learning*, *48*, 1050–1059. JMLR.org.

Gao, H., Shen, W., Qiu, X., Xu, R., Yang, B., & Hu, J. (2025). SSD-TS: Exploring the potential of linear state space models for diffusion models in time series imputation. In *Proceedings of the 31st ACM SIGKDD Conference on knowledge discovery and data mining V.2* (pp. 649–660). ACM.

Gao, N., Xue, H., Shao, W., Zhao, S., Qin, K. K., Prabowo, A., Rahaman, M. S., & Salim, F. D. (2022). Generative adversarial networks for spatio-temporal data: A survey. *ACM Transactions on Intelligent Systems and Technology (TIST)* (p. 13).

Gneiting, T., & Raftery, A. E. (2007). Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, *102* (477), 359–378.

Habiba, M., & Pearlmutter, B. A. (2020). Neural odes for informative missingness in multivariate time series. In *2020 31st Irish Signals and Systems Conference (ISSC)* (pp. 1–6).

Haworth, J., & Cheng, T. (2012). Non-parametric regression for space-time forecasting under missing data. *Computers, Environment and Urban Systems*, *36*, 538–550.

He, K., Chen, X., Xie, S., Li, Y., Dollár, P., & Girshick, R. (2022). Masked autoencoders are scalable vision learners. In *2022 IEEE/CVF Conference on computer vision and pattern recognition (CVPR)* (pp. 15979–15988).

Henrickson, K., Zou, Y., & Wang, Y. (2015). Flexible and robust method for missing loop detector data imputation. *Transportation Research Record*, *2527* (1), 29–36.

Hsu, M., Chien, Y., Wang, W., & Hsu, C. J. (2020). A convolutional fuzzy neural network architecture for object classification with small training database. *International Journal of Fuzzy Systems*, *22* (1), 1–10.

Ku, W. C., Jagadeesh, G. R., Prakash, A., & Srikanthan, T. (2016). A clustering-based approach for data-driven imputation of missing traffic data. In *2016 IEEE Forum on integrated and sustainable transportation systems (FISTS)* (pp. 16–21). Institute of Electrical and Electronics Engineers Inc.

Li, S. C., Jiang, B., Marlin, B.M (2019). MISGAN: Learning from incomplete data with generative adversarial networks. In *7th International conference on learning representations*. ICLR.

Liang, Y., Shao, Z., Wang, F., Zhang, Z., Sun, T., & Xu, Y. (2022). BasicTS: An open source fair multivariate time series prediction benchmark. In *International symposium on benchmarking, measuring and optimization* (pp. 87–101). Springer.

Liang, Y., Xia, Y., Ke, S., Wang, Y., Wen, Q., Zhang, J., Zheng, Y., & Zimmermann, R. (2023). Airformer: Predicting nationwide air quality in china with transformers. In *Proceedings of the thirty-seventh AAAI conference on artificial intelligence* (pp. 14329–14337). AAAI Press.

Little, R., & Rubin, D. (2019). Statistical Analysis with Missing Data, Third Edition. Wiley Series in Probability and Statistics. (1st ed.). Wiley.

Liu, X., Liang, Y., Huang, C., Zheng, Y., Hooi, B., & Zimmermann, R. (2022). When do contrastive learning signals help spatio-temporal graph forecasting?. In: *the 30th International conference on advances in geographic information systems* (pp. 5:1-5:12). ACM.

Nasiri, H., & Ebadzadeh, M. M. (2022). MFRFNN: Multi-functional recurrent fuzzy neural network for chaotic time series prediction. *Neurocomputing*, *507*, 292–310.

Nazábal, A., Olmos, P. M., Ghahramani, Z., & Valera, I. (2020). Handling incomplete heterogeneous data using vaes. *Pattern Recognition*, *107*, 107501.

Nie, T., Qin, G., Ma, W., Mei, Y., & Sun, J. (2024). Imputeformer: Low rankness-induced transformers for generalizable spatiotemporal imputation. In *Proceedings of the 30th ACM SIGKDD conference on knowledge discovery and data mining, kdd '24* (pp. 2260–2271). Association for Computing Machinery.

Qin, H., Zhan, X., Li, Y., Yang, X., & Zheng, Y. (2021). Network-wide traffic states imputation using self-interested coalitional learning. In *Proceedings of the ACM SIGKDD International conference on knowledge discovery and data mining* (pp. 1370–1378). Association for Computing Machinery.

Stekhoven, D. J., & Bühlmann, P. (2012). Missforest-non-parametric missing value imputation for mixed-type data. *Bioinformatics*, *28*, 112–118.

Tan, H., Feng, G., Feng, J., Wang, W., Zhang, Y. J., & Li, F. (2013). A tensor-based method for missing traffic data completion. *Transportation Research Part C: Emerging Technologies*, *28*, 15–27.

Tang, K., Chen, S., Liu, Z., & Khattak, A. J. (2018). A tensor-based bayesian probabilistic model for citywide personalized travel time estimation. *Transportation Research Part C: Emerging Technologies*, *90*, 260–280.

Tashiro, Y., Song, J., Song, Y., & Ermon, S. (2021). CSDI: Conditional score-based diffusion models for probabilistic time series imputation. In *Advances in neural information processing systems* (pp. 24804–24816).

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems* (pp. 5998–6008).

Wu, H., Hu, T., Liu, Y., Zhou, H., Wang, J., & Long, M. (2023). TimesNet: Temporal 2d-variation modeling for general time series analysis. In *The eleventh international conference on learning representations* ICLR 2023.

Wu, Z., Pan, S., Long, G., Jiang, J., & Zhang, C. (2019). Graph wavenet for deep spatial-temporal graph modeling. In *Proceedings of the twenty-eighth international joint conference on artificial intelligence* (pp. 1907–1913). International Joint Conferences on Artificial Intelligence Organization.

Yi, X., Zheng, Y., Zhang, J., & Li, T. (2016). ST-MVL: Filling missing values in geo-sensory time series data. In *Proceedings of the twenty-fifth international joint conference on artificial intelligence* (pp. 2704–2710).

Yoon, J., Jordon, J., & Schaar, M. V. (2018). Gain: Missing data imputation using generative adversarial nets. In: *35th international conference on machine learning, ICML 2018*, PMLR, *13*, 9052–9059.

Yu, C., Wang, F., Shao, Z., Qian, T., Zhang, Z., Wei, W., An, Z., Wang, Q., & Xu, Y. (2025a). GinAR+: A robust end-to-end framework for multivariate time series forecasting with missing values *IEEE Transactions on Knowledge and Data Engineering 37*(8) 4635–4648.

Yu, C., Wang, F., Shao, Z., Qian, T., Zhang, Z., Wei, W., & Xu, Y. (2024). GinAR: An end-to-end multivariate time series forecasting model suitable for variable missing. Association for Computing Machinery. New York, NY, USA. https://doi.org/10.1145/3637528.3672055

Yu, C., Wang, F., Yang, C., Shao, Z., Sun, T., Qian, T., Wei, W., An, Z., & Xu, Y. (2025b). Merlin: Multi-view representation learning for robust multivariate time series forecasting with unfixed missing rates. In *Proceedings of the 31st ACM SIGKDD conference on knowledge discovery and data mining v.2, kdd '25* (pp. 3633–3644). Association for Computing Machinery.

Zhang, J., Zheng, Y., & Qi, D. (2017). Deep spatio-temporal residual networks for citywide crowd flows prediction. In *Proceedings of the thirty-first aaai conference on artificial intelligence* (pp. 1655–1661).

Zhang, K., Li, C., & Yang, Q. (2023). Trid-MAE: A generic pre-trained model for multivariate time series with missing values. In *Proceedings of the 32nd ACM international conference on information and knowledge management* (pp. 3164–3173). ACM.

Zhao, Q., Zhang, L., & Cichocki, A. (2015). Bayesian CP factorization of incomplete tensors with automatic rank determination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *37* (9), 1751–1763.

Zhong, M., Lingras, P., & Sharma, S. (2004). Estimation of missing traffic counts using factor, genetic, neural, and regression techniques. *Transportation Research Part C: Emerging Technologies*, *12*, 139–166.

Zhou, H., Balakrishnan, S., & Lipton, Z. (2023). Domain adaptation under missingness shift. In *Proceedings of the 26th international conference on artificial intelligence and statistics* (pp. 9577–9606). PMLR.

Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., & Zhang, W. (2021). Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the thirty-fifth aaai conference on artificial intelligence* (pp. 11106–11115). AAAI Press.