MMNet: Missing-Aware and Memory-Enhanced Network for Multivariate Time Series Imputation

Xiaoye Miao[1] , **Han Shi**[1] , **Yi Yuan**[2] , **Daozhan Pan**[2] , **Yangyang Wu**[2,3] and **Xiaohua Pan**[2,3]
[1]Center for Data Science, Zhejiang University
[2]Software College, Zhejiang University
[3]Binjiang Institute of Zhejiang University
{miaoxy, 22421063, icecens, 22251221, zjuwuyy}@zju.edu.cn, panxh@zju-bj.com

Abstract

Multivariate time series (MTS) data in real-world scenarios are often incomplete, which hinders effective data analysis. Therefore, MTS imputation has been widely studied to facilitate various MTS tasks. Existing imputation methods primarily initialize missing values with zeros in order to perform effective incomplete MTS encoding, which impede the model's capacity to precisely discern the missing distribution. Moreover, these methods often overlook the global similarity in time series but are limited in the use of local information within the sample. To this end, we propose a novel multivariate time series imputation network model, named MMNet. MMNet introduces a Missing-Aware Embedding (MAE) approach to adaptively represent incomplete MTS, allowing the model to better distinguish between missing and observed data. Furthermore, we design a Memory-Enhanced Encoder (MEE) aimed at modeling prior knowledge through memory mechanism, enabling better utilization of the global similarity within the time series. Building upon this, MMNet incorporates a Multi-scale Mixing architecture (MSM) that leverages information from multiple scales to enhance the final imputation. Extensive experiments on four public real-world datasets demonstrate that, MMNet yields a more than 25% gain in performance, compared with the state-of-the-art methods.

1 Introduction

Multivariate time series(MTS) represent a crucial type of data in the real world, providing valuable insights for decision-making across various fields [Yi *et al.*, 2023b], including energy [Pérez-Lombard *et al.*, 2008], healthcare [Esteban *et al.*, 2017], weather [Bright *et al.*, 2015], and transportation [Li *et al.*, 2018]. However, in practical scenarios, multivariate time series are typically incomplete due to various factors, such as sensor malfunctions [Miao *et al.*, 2021], unstable system environments [Wu *et al.*, 2022; Wu *et al.*, 2023b], or privacy concerns [Wu *et al.*, 2023c; Wu *et al.*, 2022]. For instance, the average missing rate of public real-world medical dataset PhysioNet exceeds 80% [Silva *et al.*, 2012], which significantly hinders time series analysis. Consequently, addressing the problem of imputing missing values in incomplete multivariate time series data has become a critical challenge.

Early approaches for MTS imputation typically relied on statistical or machine learning methods, such as mean averaging, ARIMA [Nelson, 1998], and KNN [Peterson, 2009]. While these methods are generally effective in capturing the linear characteristics of time series, they struggle to model the complex and nonlinear temporal dependencies. In recent years, imputation methods based on deep neural networks have achieved remarkable success in both deterministic and probabilistic imputation. These approaches commonly employ autoregressive models to process incomplete MTS data.

Nevertheless, to effectively encode missing data, these methods often require a preliminary step of fully filling in the missing components. Currently, most deep learning methods [Ma *et al.*, 2019; Du *et al.*, 2023; Bansal *et al.*, 2021] primarily initialize missing values with zeros, and a missing mask matrix is used to mark the locations of the missing data. However, this approach will unavoidably introduce noise, complicating the model's ability to differentiate between missing and observed data, while also failing to capture variations in missingness across multiple time series.

Moreover, the inherent non-stationarity and variability of time series data pose significant challenges for accurately imputing historical observations. Although a relationship exists between the observed and missing components, this relationship is complex and further exacerbated in incomplete scenarios. In the presence of missing data, the available information for time series imputation is sparse. Existing methods [Chen *et al.*, 2023; Li *et al.*, 2018; Luo *et al.*, 2019] typically rely on limited local information within the sample for imputation. However, these methods overlook the fact that similarity in time series data is not limited to local samples, but also extends globally, as similar time segments can be identified beyond the sample.

In this paper, we propose a Missing-Aware and Memory-Enhanced NetWork named MMNet for incomplete multivariate time series imputation. Specifically, MMNet introduces a *missing-aware embedding* (MAE) module, which uses learnable vectors to represent missing components, enabling the model to differentiate between missing and observed data while adapting to varying missing patterns. To capture global patterns in MTS, MMNet introduces a *memory-enhanced encoder* (MEE) that leverages persistent memory vectors to integrate global prior information while modeling temporal and variable dependencies. Moreover, MMNet employs a *multi-scale mixing* (MSM) architecture for more robust imputation. MSM encodes incomplete multivariate time series data across multiple scales, progressively merging coarse-fine information, thereby integrating both macroscopic and detailed data to improve imputation performance. We conduct extensive evaluations on four real-world MTS datasets. The results show the superior performance of our proposed method by comparison with current SOTA models. Our main contributions are summarized as follows:

- We propose MMNet, a Missing-Aware and Memory-Enhanced Network for multivariate time series imputation. MMNet adaptively learns the missingness in MTS and utilizes a memory mechanism to effectively capture and leverage the global information in the time series.
- We propose a MAE module for incomplete MTS embedding, utilizing learnable vectors to adaptively represent missing components in incomplete MTS, thus avoiding uncontrollable noise.
- We introduce a MEE module that captures temporal and variable dependencies while utilizing persistent memory vectors to model global similarities in time series data.
- Extensive experiments using four real-life MTS datasets demonstrate that, MMNet substantially outperforms the state-of-the-art methods.

2 Related Work

Multivariate time series imputation has attracted increasing attention, driven by the growing availability of time series data and its diverse applications. In the early stages, statistical methods were widely employed to address missing values in multivariate time series data. These approaches typically substitute missing values with statistical estimates, such as zero, the mean, or the last observed value, or utilize simple statistical models like ARIMA [Nelson, 1998], ARFIMA [Hamzaçebi, 2008], and SARIMA [Hamzaçebi, 2008]. However, the reliance on linear assumptions limit their ability to capture the dynamic and nonlinear features of time series data. Recently, various deep learning-based methods have been developed to address the limitations of traditional approaches in time series imputation, which can be broadly categorized into two classes: forecasting-based methods and generating-based methods.

Forecasting-based methods provide consistent imputation results but often overlook the inherent uncertainties in the data. RNN-based methods have become integral to time series analysis, demonstrating their effectiveness in handling missing data, e.g., GRU-D [Chung *et al.*, 2014], M-RNN [Yoon *et al.*, 2019], Brits [Cao *et al.*, 2018]. CNN-based methods have also gained prominence in MTS imputation tasks, offering robust performance with TimesNet [Wu *et al.*, 2023a] and ModernTCN [Luo and Wang, 2024] being notable examples. GNN-based methods utilize graph structures to model MTS [Yi *et al.*, 2023a; Zhu *et al.*, 2023], leveraging learned node representations to reconstruct missing values, e.g., GRIN [Cini *et al.*, 2022], SPIN [Marisca *et al.*, 2022]. Transformer-based methods [Wen *et al.*, 2023] are also used to model time series, leveraging self-attention mechanisms to effectively capture complex dependencies, e.g., CDSA [Ma *et al.*, 2019], DeepMVI [Bansal *et al.*, 2021], SAITS [Du *et al.*, 2023], OFA [Zhou *et al.*, 2023].

Generating-based methods utilize generative models to simulate and generate missing data. These methods are notable for their ability to produce diverse outputs for missing observations, quantifying computational uncertainty as probability distributions by learning the distribution from the observed values. VAE-based methods typically employ an encoder-decoder structure to approximate the true data distribution, optimizing the evidence lower bound (ELBO) to achieve this goal, e.g., GP-VAE [Fortuin *et al.*, 2020], V-RIN [Mulyadi *et al.*, 2022], supnotMIWAE [Kim *et al.*, 2023]. GAN-based methods leverage a min-max game between the generator and discriminator: the generator simulates the real data distribution, while the discriminator distinguishes between real and generated data, e.g., GRUI-GAN [Luo *et al.*, 2018], E2GAN [Luo *et al.*, 2019], SSGAN [Miao *et al.*, 2021]. Diffusion model use Markov chain process to capture complex data distribution, e.g., CSDI [Tashiro *et al.*, 2021], SSSD [Alcaraz and Strodthoff, 2022], CSBI [Chen *et al.*, 2023], Score-CDM [Zhang *et al.*, 2024].

However, current deep learning methods exhibit certain limitations. A major drawback is that they often require the initialization of missing values, which may degrade the performance of imputation precision. In addition, they frequently fail to effectively capture global patterns in time series data and incorporate prior knowledge.

3 Problem Definition

Given a MTS dataset $\mathcal{D} = \{\mathbf{X}_1, \cdots, \mathbf{X}_N\}$, each sample $\mathbf{X}$ with d variables and L time steps can be formally denoted as $\mathbf{X} = (\mathbf{x}_1, \cdots, \mathbf{x}_d)^\top = (\mathbf{x}_{:1}, \cdots, \mathbf{x}_{:L}) \in \mathbb{R}^{d\times L}$, where $\mathbf{x}_i$ can be represented as $(x_{i1}, \cdots, x_{iL})$. Specifically, x_{ij} is the value of the i-th variable at the j-th time step in $\mathbf{X}$, and this value may be missing in the incomplete multivariate time series dataset.

Incomplete multivariate time series. To encode the missing information in each sample, a mask matrix $\mathbf{M} = (\mathbf{m}_1, \cdots, \mathbf{m}_d)^\top = (\mathbf{m}_{:1}, \cdots, \mathbf{m}_{:L}) \in \{0, 1\}^{d\times L}$ is introduced to indicate whether a value in $\mathbf{X}$ is missing. Here, $\mathbf{m}_i = (m_{i1}, \cdots, m_{iL})$, and m_{ij} equals 0 or 1, indicating that x_{ij} is either missing or observable.

Multivariate time series imputation. Given an incomplete multivariate time series sample $\mathbf{X}$, the goal of incomplete multivariate time series imputation is to construct an imputation model $\mathcal{M}_\theta$ parameterized by θ, which can accurately estimates the missing values in the data. The imputed time series $\hat{\mathbf{X}}$ can be defined as:

$$\hat{\mathbf{X}} = \mathbf{M} \odot \mathbf{X} + (1 - \mathbf{M}) \odot \bar{\mathbf{X}}, \tag{1}$$

where $\odot$ denotes element-wise multiplication, and $\bar{\mathbf{X}} = \mathcal{M}_\theta(\mathbf{X})$ is the reconstructed time series.

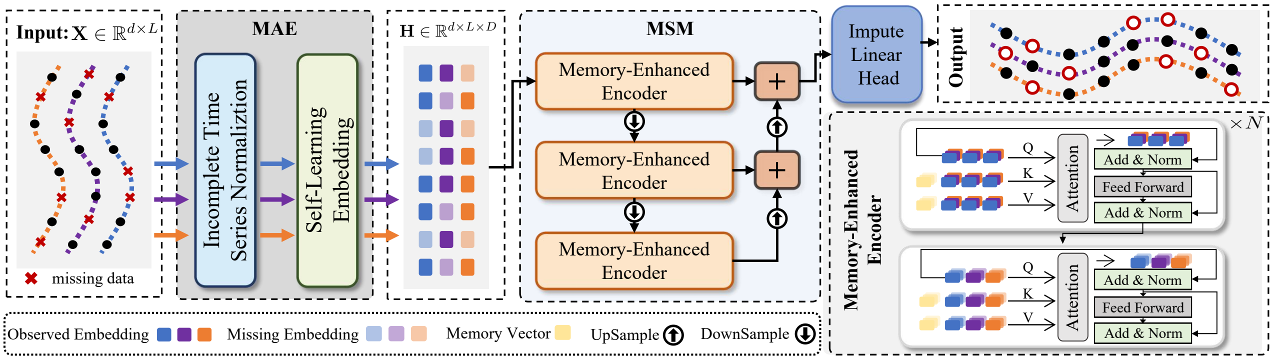


Figure 1: The overall framework of the MMNet. It consists of three main modules: *the missing-aware embedding* (MAE) module, *the multi-scale mixing* (MSM) module, and the *memory-enhanced encoder* (MEE) module. Firstly, the MAE Module is used to project incomplete multivariate time series data $\mathbf{X} \in \mathbb{R}^{d\times L}$ into a hidden representation $\mathbf{H} \in \mathbb{R}^{d\times L\times D}$. Secondly, *the multi-scale mixing* architecture, which incorporates the MEE module at each scale, is applied to $\mathbf{H}$ to obtain the multi-scale encoding set. Finally, a coarse-to-fine mixing strategy is employed to combine encoding across multiple scales, followed by a linear layer to impute missing values.

4 Methodology

In this section, we present the proposed model, MMNet, with its architecture depicted in Figure 1. The model consists of two key components: the *missing-aware embedding* (MAE) module and the *multi-scale mixing* (MSM) module, both specifically designed for incomplete MTS. First, the incomplete multivariate time series data are processed through the MAE module to generate an incomplete time series embedding. This embedding is then passed to the MSM module, where each scale's encoder incorporates a *memory-enhanced encoder* (MEE) to effectively capture both variable and temporal dependencies via a memory-enhanced mechanism. Finally, a coarse-to-fine mixing approach integrates the encodings across multiple scales, and a linear layer is applied to the mixed encoding to impute the missing values.

4.1 Missing-Aware Embedding

The *missing-aware embedding* (MAE) module of MMNet aims to project incomplete MTS into hidden representations, allowing the model to explicitly distinguish between missing and observed values. To achieve this, it first normalizes the data to mitigate the impact of non-stationarity. Subsequently, the *self-learning missing embedding* is introduced to adaptively encode the normalized incomplete MTS.

Incomplete Time Series Normalization. To address the impact of inherent non-stationarity in time series on the model, we first normalize the input incomplete time series sample $\mathbf{X}$ using Z-score scaling prior to data embedding [Patro and Sahu, 2015]. This normalization helps mitigate the distribution shift effects within incomplete multivariate time series data. Specifically, for each incomplete time series $\mathbf{x}_i \in \mathbf{X}$ and its corresponding mask vector $\mathbf{m}_i$, the model first computes the mean μ_i and standard deviation σ_i based on the observed data:

$$\mu_i = \frac{\sum_{j=1}^{L} x_{ij} \cdot m_{ij}}{\sum_{j=1}^{L} m_{ij}}, \sigma_i^2 = \frac{\sum_{j=1}^{L} (x_{ij} - \mu_i)^2 \cdot m_{ij}}{\sum_{j=1}^{L} m_{ij}}. \quad (2)$$

Next, the normalized time series $\ddot{\mathbf{x}}_i \in \ddot{\mathbf{X}}$ can be defined as: $\ddot{\mathbf{x}}_i = \left(\frac{x_{i1}-\mu_i}{\sigma_i}, \cdots, \frac{x_{iL}-\mu_i}{\sigma_i}\right)$. To preserve the non-stationarity of the time series, the model applies inverse normalization on the output data. Specifically, the predicted time series is restored to its original scale using the stored mean and standard deviation.

Self-Learning Missing Embedding. Subsequently, to prevent pre-filled values from interfering with the model and introducing noise, we propose a *self-learning missing embedding* method to adaptively represent the missing values in multivariate time series. Regarding the observed data, the model employs a trainable linear projection to generate its embedded representation. Furthermore, for each time series $\mathbf{x}_i$, a unique missing embedding vector $[\text{MASK}]_i$ is learned to represent the missing components in the sequence. In this way, the model can more effectively represent and distinguish the missing states of different time series. Additionally, a positional encoding is utilized to capture the temporal order, with the specific formula provided below:

$$\mathbf{h}_{ij} = \begin{cases} \mathbf{W}^{\text{o}} \cdot \ddot{\mathbf{x}}_{ij} + \mathbf{W}_j^{\text{pos}} & m_{ij} = 1, \\ [\text{MASK}]_i + \mathbf{W}_j^{\text{pos}} & m_{ij} = 0, \end{cases} \quad (3)$$

where $\mathbf{W}^{\text{o}} \in \mathbb{R}^{D\times 1}$ represents the learnable projection matrix, $\mathbf{W}_j^{\text{pos}} \in \mathbb{R}^{D}$ represents the absolute positional embedding for position j, and $[\text{MASK}]_i \in \mathbb{R}^{D}$ denotes the learnable missing embedding vector for the i-th time series. Finally, the model produces a 2D vector array $\mathbf{H} = (\mathbf{h}_1, \cdots, \mathbf{h}_d)^\top = (\mathbf{h}_{:1}, \cdots, \mathbf{h}_{:L}) \in \mathbb{R}^{d\times L\times D}$, where $\mathbf{h}_{ij}$ represents the embedded observable or missing data for the i-th time series at the j-th timestamp.

4.2 Memory-Enhanced Encoder

Given an incomplete multivariate time series as input, self-attention operations, commonly used in Transformers, are capable of capturing both temporal and inter-variable dependencies, thereby enabling more effective imputation. Notably, the basic self-attention mechanism focuses primarily on local relationships within the input sample, as the attention weights are determined solely on the pairwise similarities between the linear projections of the input. Nevertheless, the similarity in time series lies not only within the sample itself but also is reflected in the global similarity across samples.

To fully exploit the global similarity inherent in time series, we introduce a *memory-enhanced attention* (MEA), which encodes global prior knowledge of time series by incorporating additional learnable memory vectors into the attention computation. It can be formally defined as:

$$\begin{aligned}&\mathrm{MEA}(X) = \mathrm{MSA}(W_q Q(X), W_k K(X), W_v V(X)),\\ &Q(X) = X, K(X) = V(X) = [X \| \mathrm{Mem}],\end{aligned} \tag{4}$$

where W_q, W_k, W_v are learnable attention weights in the attention mechanism, and $\mathrm{Mem} \in \mathbb{R}^{n_m \times D}$ is a matrix containing n_m learnable memory vectors. The symbol $\|$ denotes concatenation. Intuitively, by adding additional learnable memory vectors, the attention mechanism becomes capable of retrieving and capturing global relationships beyond the individual sample.

Based on this, we propose a *memory-enhanced encoder* (MEE) to capture both temporal and inter-variable relationships in MTS. The encoder is composed of multiple stacked two-stage memory-enhanced attention layers, each receiving a two-dimensional vector matrix $\mathbf{Z} = (\mathbf{Z}_1, \cdots, \mathbf{Z}_d)^\top = (\mathbf{Z}_{:1}, \cdots, \mathbf{Z}_{:l}) \in \mathbb{R}^{d \times l \times D}$ as input, where l and d represent the dimension of the time and variable, respectively.

To capture the temporal dependencies of each time series, the model applies the *memory-enhanced attention* layer to the time dimension of each time series $\mathbf{Z}_i$ $(1 \le i \le d)$, i.e.

$$\begin{aligned}\bar{\mathbf{A}}_i &= \mathrm{LN}(\mathbf{Z}_i + \mathrm{MEA}(\mathbf{Z}_i)), \quad 1 \le i \le d,\\ \bar{\mathbf{Z}}_i &= \mathrm{LN}(\bar{\mathbf{A}}_i + \mathrm{MLP}(\bar{\mathbf{A}}_i)), \quad 1 \le i \le d,\end{aligned} \tag{5}$$

where LN denotes layer normalization, MLP$(\cdot)$ refers to a multi-layer perceptron, and $\mathrm{MEA}(\cdot)$ denotes the memory-enhanced attention layer. The output of the above process is denoted as $\bar{\mathbf{Z}}$. Next, to capture the inter-variable relationships, the model applies the memory-enhanced attention mechanism along the variable dimension of the output $\bar{\mathbf{Z}}_{:i}$ $(1 \le i \le l)$, as formulated in the following equations:

$$\begin{aligned}\hat{\mathbf{A}}_{:i} &= \mathrm{LayerNorm}(\bar{\mathbf{Z}}_{:i} + \mathrm{MEA}(\bar{\mathbf{Z}}_{:i})), \quad 1 \le i \le l,\\ \hat{\mathbf{Z}}_{:i} &= \mathrm{LayerNorm}(\hat{\mathbf{A}}_{:i} + \mathrm{MLP}(\hat{\mathbf{A}}_{:i})), \quad 1 \le i \le l,\end{aligned} \tag{6}$$

Then the output $\hat{\mathbf{Z}}$ fully integrates information from both inter-variable and temporal dimension. Additionally, by leveraging the memory-enhanced mechanism, the model can implicitly capture global similarities in the time series, resulting in more informative representations.

4.3 Multi-scale Mixing

Time series exhibit distinct characteristics at different scales, making multi-scale modeling essential for capturing their inherent patterns. In MTS imputation, coarse scales capture general trends, while fine scales reconstruct finer details.

Building on the memory-enhanced encoder, MMNet utilizes a *multi-scale mixing*(MSM) architecture to extract and integrate multi-scale features from incomplete multivariate time series. The encoder at each scale transforms the input into a hidden representation, capturing features specific to that scale. For an architecture with S scales, the input and output of the encoder at each scale can be defined as:

$$\begin{aligned}&\mathbf{Z}^{\mathrm{in},i} = \begin{cases}\mathbf{H} & i = 1,\\ \mathrm{DownSample}(\mathbf{Z}^{\mathrm{out},i-1}) & i \in (1, S],\end{cases}\\ &\mathbf{Z}^{\mathrm{out},i} = \mathrm{MEE}(\mathbf{Z}^{\mathrm{in},i}) \quad i \in [1, S],\end{aligned} \tag{7}$$

where $\mathrm{DownSample}(\cdot)$ denotes the downsampling layer implemented by using a convolution operation in the time dimension (kernel=3, stride=2, and padding=1). $\mathrm{MEE}(\cdot)$ denotes the memory-enhanced encoder. $\mathbf{Z}^{\mathrm{in},i}$ and $\mathbf{Z}^{\mathrm{out},i}$ respectively represent the input and output of the encoder at the i-th scale, and $\mathbf{H}$ denotes the output of the MAE. We can obtain a multi-scale encoding set $[\mathbf{Z}^{\mathrm{out},1}, \mathbf{Z}^{\mathrm{out},2}, \cdots, \mathbf{Z}^{\mathrm{out},S}]$.

Considering that coarse-scale time series provide clearer macro-level information compared to fine-scale time series, the model adopts a coarse-to-fine mixing strategy to combine the encoding of S scales, leveraging macro knowledge from coarser scale to guide the imputation modeling at finer scale:

$$\begin{aligned}&\text{for } s : (S-1) \to 1 \text{ do:}\\ &\qquad \mathbf{Z}^{out,s} = \mathbf{Z}^{out,s} + \text{ UpSample } \left(\mathbf{Z}^{out,s+1}\right)\end{aligned} \tag{8}$$

where $\mathrm{UpSample}(\cdot)$ is the upsampling layer implemented by a deconvolution operation in the time dimension.

Finally, a linear head is applied to the mixed encoding $\mathbf{Z}^{out,1}$ to obtain the reconstructed multivariate time series. By combining the observable values of the original sample with the reconstructed series, the imputed time series is obtained and formally expressed as:

$$\begin{aligned}\bar{\mathbf{X}} &= \mathbf{W}^z \mathbf{Z}^{out,1},\\ \hat{\mathbf{X}} &= \mathbf{M} \odot \mathbf{X} + (1 - \mathbf{M}) \odot \bar{\mathbf{X}},\end{aligned} \tag{9}$$

where $\mathbf{W}^z \in \mathbb{R}^{1 \times D}$ is a learnable linear projection matrix, $\odot$ denotes element-wise multiplication, $\mathbf{X}$ and $\mathbf{M}$ represent the input sample data and its missing matrix, respectively. $\bar{\mathbf{X}} \in \mathbb{R}^{d \times L}$ and $\hat{\mathbf{X}} \in \mathbb{R}^{d \times L}$ denote the reconstructed and the imputed time series, respectively.

4.4 Overview Training Process

To train the imputation model more effectively on incomplete multivariate time series, we employs multi-task training optimization referring to SAITS [Du *et al.*, 2023].

In general, the objective of MMNet contains two types of loss: i) *masked reconstruction loss*, which aims to reconstruct the masked values within the incomplete historical multivariate time series. and ii) *observed reconstruction loss*, which focuses on accurately reconstructing the observed values. Here, *masked reconstruction loss* encourages the model to impute missing values accurately, while *observed reconstruction loss* ensures the model aligns with the distribution of observed data. The final objective can be formalized as:

$$\mathcal{L} = \frac{||(\mathbf{X} - \bar{\mathbf{X}}) \odot \bar{\mathbf{M}}||_2^2}{||\bar{\mathbf{M}}||_2^2} + \lambda \frac{||(\mathbf{X} - \bar{\mathbf{X}}) \odot (\mathbf{M} - \bar{\mathbf{M}})||_2^2}{||(\mathbf{M} - \bar{\mathbf{M}})||_2^2}, \tag{10}$$

where $|| \cdot ||_2$ represents the L2 norm function, $\odot$ denotes element-wise multiplication, and λ is a hyper-parameter.

Specifically, during model training, we randomly mask γ percent of the time series. To encode the artificial masking information, an artificial mask matrix $\bar{\mathbf{M}} \in \mathbb{R}^{d \times L}$ is used, where $\bar{\mathbf{m}}_{ij}$ equals 0 (resp. 1) means $\mathbf{x}_{ij}$ is unmasked (resp. masked). This objective function is used to train the multivariate time series imputation model in an unsupervised manner. Eventually, the MMNet is trained by minimizing $\mathcal{L}$ to obtain the optimal imputation model $\bar{\mathcal{M}}_\theta$. The training pipeline is shown in Algorithm 1.

Algorithm 1: MMNet Algorithm Flow

Input: Incomplete multivariate time series $\mathbf{X}$, missing mask matrix $\mathbf{M}$, number of scales S, training epochs $epochs$

Output: Optimized time series imputation model $\mathcal{M}_\theta$

for $epoch = 1$ to $epochs$ **do**
- Mask the observable values in $\mathbf{X}$ with a ratio γ
- Normalize the masked data
- obtain the incomplete data embedding $\mathbf{H}$
- `/* Multiscale Encoding */`
- $\mathbf{Z}^{in,1} = \mathbf{H}$
- **for** $s = 1$ to S **do**
  - $\mathbf{Z}^{out,s} = \text{MEE}(\mathbf{Z}^{in,s})$
  - $\mathbf{Z}^{in,s+1} = \text{DownSample}(\mathbf{Z}^{out,s})$
- `/* Multiscale Mixing */`
- **for** s = S-1 to 1 **do**
  - $\mathbf{Z}^{out,s} = \mathbf{Z}^{out,s} + \text{Upsample}(\mathbf{Z}^{out,s+1})$
- Obtain the reconstructed sequence $\bar{\mathbf{X}}$ and the imputed sequence $\hat{\mathbf{X}}$ using Equation 9
- Calculate the training loss using Equation 10
- Update parameters θ using the ADAM optimizer with $\nabla_\theta \mathcal{L}$.

return Optimized imputation model $\mathcal{M}_\theta$

5 Experiment

In this section, we evaluate the performance of MMNet and eight state-of-the-art methods for multivariate time series imputation . All approaches were implemented in Python. The experiments were conducted on an Intel Core 2.80GHz server with TITAN Xp 12GiB (GPU) and 192GB RAM, running Ubuntu 18.04 system.

5.1 Experiment Setup

Dataset. (i) PhysioNet 2012 Mortality Prediction Challenge Dataset(**PhysioNet**) [Silva *et al.*, 2012] contains 12,000 multivariate clinical time series samples with 37 measurements collected from ICU (Intensive Care Unit) patients. Each sample represents data recorded during the first 48 hours following a patient's admission to the ICU. The dataset contains 80.67% missing values. (ii) Beijing Multi-site Air Quality Dataset(**Air-Quality**) [Zhang *et al.*, 2017] includes hourly air pollutant data from 12 monitoring stations in Beijing, collected between March 1, 2013, and February 28, 2017 (48 months). For each station, 11 continuous time series variables (e.g., PM2.5, PM10, SO2) were measured. The dataset contains 1.6% missing values. (iii) Electric Transformer Temperature Dataset(**ETT**) [Zhou *et al.*, 2021] records hourly electricity consumption from 370 clients across two regions in China. Each data sample includes 7 features, including temperature and 6 different types of external power load characteristics. This dataset consists of four subsets (ETTh1, ETTh2, ETTm1, ETTm2), with the ETTh1 subset used in our experiments. (iv) Italy Air Quality Dataset (**Italy-Air**) [De Vito *et al.*, 2008] provides hourly average response values from five metal oxide sensors, alongside hourly average pollutant concentration data from certified analyzers. The train/val/test splits for all four datasets follow the same setup as in [Du *et al.*, 2024]. The sequence length for each sample across all four datasets is 48, 24, 48, and 24, respectively.

Baselines. In the experiments, we compare MMNet with eight baseline time series imputation methods. These include one static imputation method (Last) and seven advanced deep learning algorithms: SAITS [Du *et al.*, 2023], Brits [Cao *et al.*, 2018], GPVAE [Fortuin *et al.*, 2020], CSDI [Tashiro *et al.*, 2021], TimesNet [Wu *et al.*, 2023a], Trans[Vaswani *et al.*, 2017] and ModernTCN [Luo and Wang, 2024].

Evaluation metrics. We apply the *mean absolute error* (MAE) [Chai and Draxler, 2014] and *mean square error* (MSE) [Jeffery *et al.*, 2006] to evaluate the effectiveness of imputation models. Smaller metric values indicate better imputation performance. For datasets without missing values (*ETT* and *Italy-Air*), 50% of the observed values are randomly removed before model training. To compute the evaluation metrics, 20% of the observed values in both the validation and test datasets are randomly masked, and these masked values are treated as ground truth to assess imputation performance. In downstream experiment, classification accuracy is evaluated using ROC-AUC, PR-AUC, and F1-score, with higher values reflecting better performance.

Implementation details. All experiments were implemented using PyTorch [Paszke *et al.*, 2019]. To ensure a fair comparison, the number of training epochs was uniformly set to 500 across all models, with an early stopping strategy employed using a patience value of 10. Additionally, hyperparameter optimization was performed on key parameters of the deep learning-based imputation algorithms. For MMNet, to balance efficiency and accuracy, the number of scales S is set to 3, the number of layers in the encoder at each scale is set to 2, the number of memory vectors n_m is set to 12, the loss weight factor λ is set to 1, the artificial masking rate γ for self-supervised training is set to 20%, and the hidden layer dimension is set to 64. To ensure the reliability of the results, all reported metrics represent the average performance across five runs with different random seeds.

Missing patterns. In the experiments, we inspect six missing mechanisms simulated using PyGrinder[1]. The foundational classification scheme proposed by Rubin [Strawderman, 1989] delineates missing data mechanisms into three specific categories: Missing Completely At Random (MCAR), Missing At Random (MAR), and Missing Not At Random (MNAR). Despite this, in real-world scenarios,

[1] https://github.com/WenjieDu/PyGrinder

Models	*Physionet*		*Air Quality*		*ETT*		*Italy-Air*	
	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
Last	0.400 ± 0.000	0.651 ± 0.000	0.192 ± 0.000	0.276 ± 0.000	0.271 ± 0.000	0.183 ± 0.000	0.262 ± 0.000	0.321 ± 0.000
SAITS	<u>0.223</u> ± 0.002	0.344 ± 0.005	0.160 ± 0.001	0.130 ± 0.004	0.275 ± 0.010	0.173 ± 0.012	0.205 ± 0.004	0.128 ± 0.006
Brits	0.273 ± 0.001	0.417 ± 0.026	0.162 ± 0.003	0.144 ± 0.019	0.359 ± 0.005	0.284 ± 0.003	0.291 ± 0.008	0.222 ± 0.012
GPVAE	0.440 ± 0.004	0.583 ± 0.032	0.293 ± 0.001	0.276 ± 0.004	0.494 ± 0.017	0.525 ± 0.030	0.415 ± 0.010	0.492 ± 0.013
CSDI	0.229 ± 0.002	<u>0.332</u> ± 0.040	<u>0.085</u> ± 0.001	<u>0.110</u> ± 0.001	0.225 ± 0.006	0.111 ± 0.010	<u>0.182</u> ± 0.007	<u>0.126</u> ± 0.008
TimesNet	0.284 ± 0.002	0.387 ± 0.029	0.155 ± 0.003	0.230 ± 0.025	<u>0.223</u> ± 0.001	<u>0.101</u> ± 0.002	0.183 ± 0.001	0.131 ± 0.002
ModernTCN	0.305 ± 0.002	0.382 ± 0.004	0.182 ± 0.001	0.135 ± 0.001	0.333 ± 0.004	0.202 ± 0.003	0.297 ± 0.002	0.194 ± 0.003
Trans	0.229 ± 0.001	0.350 ± 0.026	0.139 ± 0.001	0.118 ± 0.003	0.254 ± 0.007	0.144 ± 0.009	0.245 ± 0.002	0.159 ± 0.001
MMNet	**0.210 ± 0.001**	**0.329** ± **0.019**	**0.079 ± 0.002**	**0.106 ± 0.005**	**0.150 ± 0.002**	**0.055 ± 0.002**	**0.075 ± 0.001**	**0.051 ± 0.001**

Table 1: Imputation performance comparison under different datasets

missing values in time series often exhibit continuity, where a missing point is typically followed by a sequence of consecutive missing points, termed as "block" missing. To account for this characteristic, we extend the three fundamental missing mechanisms by incorporating block missing, resulting in six distinct missing mechanisms: MCAR, MCAR_B, MAR, MAR_B, MNAR and MNAR_B. To ensure fairness, the missing rate R_m, representing the proportion of dropped values in the test, validation, or training data, is set to 50% by default for all generated datasets under various missing mechanisms.

5.2 Comparison Study

Table 1 reports the experimental results of multivariate time series imputation methods over four datasets. The best-performing results are highlighted in bold, while the second-best results are underlined. It can be observed that MMNet consistently outperforms all baseline methods across all datasets and metrics. Specifically, the imputation performance of MMNet, measured by MSE and MAE, is on average 26.87% and 28.62% better than the best baseline method CSDI, and it even increases up to 58.79% and 59.52% on the *Italy-Air* dataset. This is because MMNet adopts an effective imputation model with the *missing-aware embedding* and *the memory-enhanced encoder*, while improving the imputation accuracy by incorporating a *multi-scale mixing* strategy. Additionally, we observe that among the baseline methods, static imputation algorithms (Last) generally perform worse than most deep learning-based models. Among the deep learning-based approaches, the diffusion-based method CSDI achieves the best performance on most datasets, whereas the VAE-based model GPVAE performs significantly poorer compared to other methods. Therefore, the Last impuation and GPVAE imputation methods will be excluded from the subsequent parts of the experiments.

5.3 Downstream Result

In this experiment, we evaluate the performance of downstream classification analysis using imputed datasets, which indirectly reflects the performance of each imputation method. The classification task is performed on the *PhysioNet* dataset, where each sample is labeled to indicate whether the patient has died. Specifically, the imputation methods are first employed to infer missing values in original incomplete datasets. The imputed datasets are then used to train a simple RNN classifier optimized with cross-entropy loss. All hyperparameters are set as follows: the learning rate is set to 0.005, the hidden state dimension is set to 128, the number of training epochs is set to 100, and early stopping is employed with a patience of 20 epochs.

Models	PR-AUC	ROC-AUC	F1-score
SAITS	0.463 ± 0.009	<u>0.822</u> ± 0.004	<u>0.447</u> ± 0.011
Brits	0.422 ± 0.012	0.807 ± 0.005	0.415 ± 0.016
CSDI	<u>0.466</u> ± 0.006	0.821 ± 0.004	0.439 ± 0.005
TimesNet	0.394 ± 0.005	0.765 ± 0.004	0.387 ± 0.012
ModernTCN	0.436 ± 0.010	0.813 ± 0.006	0.418 ± 0.012
Trans	0.448 ± 0.007	0.810 ± 0.004	0.415 ± 0.009
MMNet	**0.468** ± 0.006	**0.831** ± 0.005	**0.451** ± 0.005

Table 2: Downstream classification evaluation on *Physionet* Dataset

The experimental results are shown in Table 2. It can be observed that datasets with better imputation results tend to exhibit superior performance in downstream classification tasks. Notably, the classifier trained on the dataset imputed by MMNet achieved the best results across all evaluation metrics. Compared to the suboptimal performance, it improved by 0.42%, 1.09%, and 0.89% in PR-AUC, ROC-AUC, and F1-Score, respectively. These findings further highlight that MMNet provides the best imputation quality among all methods and that accurate imputation can improve the performance of downstream classification tasks on incomplete multivariate time series datasets.

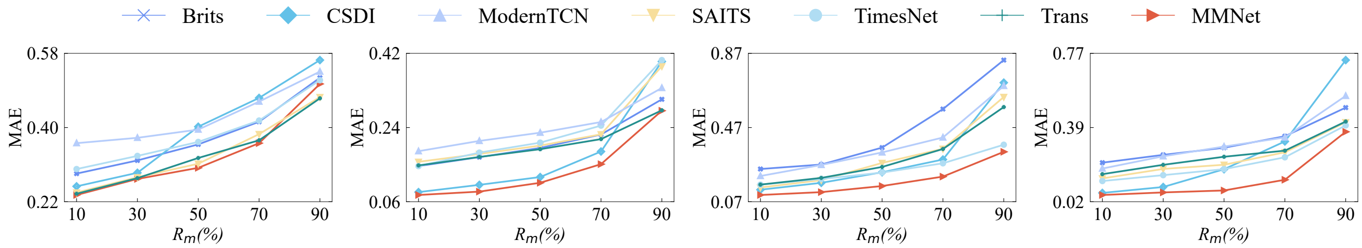
5.4 Parameter Evaluation

Effect of Missing Rate. When varying the missing rate R_m (i.e., the proportion of additional observed values in multivariate time series data are dropped) from 10% to 90%, the corresponding experimental results, measured by MAE, are depicted in Figure 2. As observed, the accuracy of all imputation algorithms consistently declines as the missing rate increases, reflected by a continuous rise in MAE. This is because higher missing rates reduce the amount of available information for inference, thereby diminishing the effectiveness of imputation algorithms. Among these algorithms, MMNet consistently outperforms all others in every case and maintains more stable accuracy as the missing rate increases. In other words, MMNet is more robust with the increasing missing rate R_m than others. The underlying reason is that MMNet incorporates a *memory-enhanced encoder*, enabling it to capture global information beyond the sample, thereby better addressing data sparsity issue.

Effect of Missing Mechnisam. In this set of experiments, we evaluate the effect of different missing mechanisms (i.e., *MCAR*, *MAR*, *MNAR*, *MCAR_B*, *MAR_B*, and *MNAR_B*) on the imputation algorithm performance. The results, mea-

Models	*Physionet*		*Air-Quality*		*ETT*		*Italy-Air*	
	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
MMNet-no-MAE	0.274 ± 0.002	0.386 ± 0.016	0.101 ± 0.002	0.116 ± 0.003	0.169 ± 0.003	0.069 ± 0.002	0.097 ± 0.004	0.074 ± 0.001
MMNet-no-MEE	0.218 ± 0.001	0.338 ± 0.013	0.084 ± 0.003	0.114 ± 0.005	0.157 ± 0.001	0.059 ± 0.001	0.085 ± 0.004	0.069 ± 0.001
MMNet-no-MSM	0.221 ± 0.002	0.342 ± 0.011	0.087 ± 0.001	0.117 ± 0.007	0.161 ± 0.004	0.062 ± 0.001	0.090 ± 0.003	0.061 ± 0.003
MMNet	**0.210** ± **0.001**	**0.329** ± **0.019**	**0.079** ± **0.002**	**0.106** ± **0.005**	**0.150** ± **0.002**	**0.055** ± **0.002**	**0.075** ± **0.001**	**0.051** ± **0.001**

Table 3: The ablation study of MMNet



(a) *Physionet* (b) *Air Quality* (c) *ETT* (d) *Italy Air*

Figure 2: The performance of multivariate time series imputation algorithms vs. R_m

sured by MAE, over the *ETT* dataset are shown in Figure 3. The results indicate that the performance of imputation algorithms is relatively insensitive to different missing mechanisms. Furthermore, "block" missing patterns (i.e., *MCAR_B*, *MAR_B*, and *MCAR_B*) leads to a further decrease in imputation performance compared to the "value" missing patterns (i.e., *MCAR*, *MAR*, and *MCAR*). This is because the "block" pattern hinders the model's ability to infer missing points from adjacent ones, while also lead to more substantial information loss over extended periods, thereby affecting the model's imputation performance. Additionally, as expected, it is evident that MMNet consistently attains the optimal imputation performance in every scenario, further validating its robustness. Specifically, MMNet surpasses the best baseline CSDI with an average improvement of 24.81%, and even increases up to 33.33% on the MCAR missing mechanism.

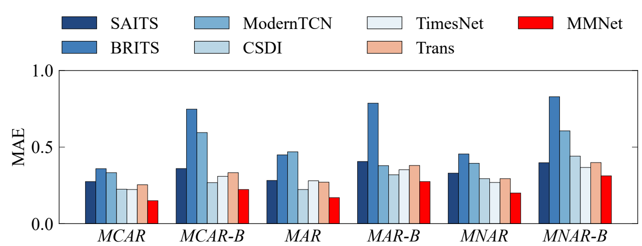


Figure 3: The performance of different missing mechanisms

Effects of Scales. We investigate the impact of the number of scales on the model's imputation performance, with the results presented in Table 4. The results clearly show that as the number of scales increases, the imputation performance improves overall, further validating the importance of integrating information from multiple scales to enhance accuracy.

5.5 Ablation Study

We investigate the influence of different components of MMNet on the imputation performance. The corresponding experimental results are shown in Table 4. MMNet-no-MAE

S	*Physionet*	*Air-Quality*	*ETT*	*Italy-Air*
1	0.221± 0.002	0.087± 0.001	0.161± 0.004	0.090± 0.004
2	0.218± 0.003	0.081± 0.002	0.154± 0.001	0.079± 0.003
3	**0.210**± 0.001	0.079± 0.002	**0.150**± 0.002	0.075± 0.001
4	0.211± 0.001	**0.078**± 0.003	0.151± 0.003	**0.072**± 0.001

Table 4: The performance (MAE) of different number of scale

refers to the variant of MMNet without the *missing-aware embedding* module, where zero-filling is used instead. MMNet-no-MEE refers to the variant of MMNet without using *memory-enhanced attention*, and MMNet-no-MSM refers to the variant of MMNet without *multi-scale mixing*. It can be observed that each module in the MMNet model do contribute positively to imputation performance. Specially, the average imputation accuracy (i.e. MSE and MAE) decreases by 24.70%, 11.17%, and 10.12% without the *missing-aware embedding*, *multi-scale mixing* and *memory-enhanced attention*, respectively. Among these, the *missing-aware embedding* module has the most significant impact on MMNet. This proves that simple pre-filling strategies indeed interfere with the model's ability. The MMNet model, by introducing MAE module, allows the model to better learn the missing distributions across multiple time series while avoiding noise.

6 Conclusion

In this paper, we propose a Missing-Aware and Memory-Enhanced Network, named MMNet, for multivariate time series imputation. Specifically, MMNet employs the *missing-aware embedding* (MAE) module to adaptively represent incomplete multivariate time series data. The *memory-enhanced encoder* (MEE) is designed to capture both temporal and variable dependencies while modeling the prior knowledge within the time series. Building on these modules, the *multi-scale mixing* (MSM) architecture is employed to combine information across multiple scales to enhance imputation performance. Extensive evaluations on four real-world datasets demonstrate the effectiveness of the proposed model, highlighting its superiority in addressing incomplete multivariate time series data.

Acknowledgments

This work is partly supported by the National Key R&D Program of China under Grant No. 2024YFF1400900, the National NSFC under Grants No. 62372404, the Leading Goose R&D Program of Zhejiang under Grant No. 2024C01109, and the Fundamental Research Funds for the Central Universities under Grant No. 226-2024-00030. Yangyang Wu is the corresponding author of the work.

References

[Alcaraz and Strodthoff, 2022] Juan Miguel Lopez Alcaraz and Nils Strodthoff. Diffusion-based time series imputation and forecasting with structured state space models. *arXiv preprint arXiv:2208.09399*, 2022.

[Bansal *et al.*, 2021] Parikshit Bansal, Prathamesh Deshpande, and Sunita Sarawagi. Missing value imputation on multidimensional time series. *Proc. VLDB Endow.*, 14(11):2533–2545, July 2021.

[Bright *et al.*, 2015] JM Bright, CJ Smith, PG Taylor, and R Crook. Stochastic generation of synthetic minutely irradiance time series derived from mean hourly weather observation data. *Solar Energy*, 115(1):229–242, 2015.

[Cao *et al.*, 2018] Wei Cao, Dong Wang, Jian Li, Hao Zhou, Lei Li, and Yitan Li. Brits: Bidirectional recurrent imputation for time series. *Advances in Neural Information Processing Systems*, 31:6776–6786, 2018.

[Chai and Draxler, 2014] Tianfeng Chai and Roland R Draxler. Root mean square error (RMSE) or mean absolute error (MAE)?–Arguments against avoiding RMSE in the literature. *Geoscientific Model Development*, 7(3):1247–1250, 2014.

[Chen *et al.*, 2023] Yu Chen, Wei Deng, Shikai Fang, Fengpei Li, Nicole Tianjiao Yang, Yikai Zhang, Kashif Rasul, Shandian Zhe, Anderson Schneider, and Yuriy Nevmyvaka. Provably convergent schrödinger bridge with applications to probabilistic time series imputation. In *Proceedings of the 40th International Conference on Machine Learning*. JMLR.org, 2023.

[Chung *et al.*, 2014] Junyoung Chung, Çaglar Gülçehre, Kyunghyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *ArXiv*, abs/1412.3555, 2014.

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

[De Vito *et al.*, 2008] Saverio De Vito, Ettore Massera, Marco Piga, Luca Martinotto, and Girolamo Di Francia. On field calibration of an electronic nose for benzene estimation in an urban pollution monitoring scenario. *Sensors and Actuators B: Chemical*, 129(2):750–757, 2008.

[Du *et al.*, 2023] Wenjie Du, David Côté, and Yan Liu. SAITS: self-attention-based imputation for time series. *Expert Syst. Appl.*, 219:119619, 2023.

[Du *et al.*, 2024] Wenjie Du, Jun Wang, Linglong Qian, Yiyuan Yang, Zina Ibrahim, Fanxing Liu, Zepu Wang, Haoxin Liu, Zhiyuan Zhao, Yingjie Zhou, et al. Tsi-bench: Benchmarking time series imputation. *arXiv preprint arXiv:2406.12747*, 2024.

[Esteban *et al.*, 2017] Cristóbal Esteban, Stephanie L Hyland, and Gunnar Rätsch. Real-valued (medical) time series generation with recurrent conditional gans. *ArXiv Preprint ArXiv:1706.02633*, 2017.

[Fortuin *et al.*, 2020] Vincent Fortuin, Dmitry Baranchuk, Gunnar Rätsch, and Stephan Mandt. GP-VAE: deep probabilistic time series imputation. In *The 23rd International Conference on Artificial Intelligence and Statistics*, pages 1651–1661. PMLR, 2020.

[Hamzaçebi, 2008] Coşkun Hamzaçebi. Improving artificial neural networks' performance in seasonal time series forecasting. *Information Sciences*, 178(23):4550–4559, 2008.

[Jeffery *et al.*, 2006] Shawn R Jeffery, Minos Garofalakis, and Michael J Franklin. Adaptive cleaning for RFID data streams. In *VLDB*, pages 163–174, 2006.

[Kim *et al.*, 2023] Seunghyun Kim, Hyunsu Kim, Eunggu Yun, Hwangrae Lee, Jaehun Lee, and Juho Lee. Probabilistic imputation for time-series classification with missing data. In *Proceedings of the 40th International Conference on Machine Learning*, Proceedings of Machine Learning Research, pages 16654–16667. PMLR, 2023.

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

[Luo and Wang, 2024] Donghao Luo and Xue Wang. Moderntcn: A modern pure convolution structure for general time series analysis. In *The Twelfth International Conference on Learning Representations*, 2024.

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

[Luo *et al.*, 2019] Yonghong Luo, Ying Zhang, Xiangrui Cai, and Xiaojie Yuan. E²gan: End-to-end generative adversarial network for multivariate time series imputation. In *IJCAI*, pages 3094–3100, 2019.

[Ma *et al.*, 2019] Jiawei Ma, Zheng Shou, Alireza Zareian, Hassan Mansour, Anthony Vetro, and Shih-Fu Chang. CDSA: cross-dimensional self-attention for multivariate, geo-tagged time series imputation. *CoRR*, abs/1905.09904, 2019.

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

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

[Mulyadi *et al.*, 2022] Ahmad Wisnu Mulyadi, Eunji Jun, and Heung-Il Suk. Uncertainty-aware variational-recurrent imputation network for clinical time series. *IEEE Transactions on Cybernetics*, 52(9):9684–9694, 2022.

[Nelson, 1998] Brian K Nelson. Time series analysis using autoregressive integrated moving average (arima) models. *Academic emergency medicine*, 5(7):739–744, 1998.

[Paszke *et al.*, 2019] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, 32, 2019.

[Patro and Sahu, 2015] SGOPAL Patro and Kishore Kumar Sahu. Normalization: A preprocessing stage. *ArXiv Preprint ArXiv:1503.06462*, 2015.

[Pérez-Lombard *et al.*, 2008] Luis Pérez-Lombard, José Ortiz, and Christine Pout. A review on buildings energy consumption information. *Energy and Buildings*, 40(3):394–398, 2008.

[Peterson, 2009] Leif E Peterson. K-nearest neighbor. *Scholarpedia*, 4(2):1883, 2009.

[Silva *et al.*, 2012] Ikaro Silva, George Moody, Daniel J Scott, Leo A Celi, and Roger G Mark. Predicting in-hospital mortality of icu patients: The physionet/computing in cardiology challenge 2012. In *CinC*, pages 245–248, 2012.

[Strawderman, 1989] William E. Strawderman. Statistical analysis with missing data (roderick j. a. little and donald b. rubin). *SIAM Rev.*, (2):348–349, 1989.

[Tashiro *et al.*, 2021] Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. CSDI: conditional score-based diffusion models for probabilistic time series imputation. In *Advances in Neural Information Processing Systems*, pages 24804–24816, 2021.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017.

[Wen *et al.*, 2023] Qingsong Wen, Tian Zhou, Chaoli Zhang, Weiqi Chen, Ziqing Ma, Junchi Yan, and Liang Sun. Transformers in time series: A survey. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23*, pages 6778–6786, 2023.

[Wu *et al.*, 2022] Yangyang Wu, Jun Wang, Xiaoye Miao, Wenjia Wang, and Jianwei Yin. Differentiable and scalable generative adversarial models for data imputation. *ArXiv Preprint ArXiv:2201.03202*, 2022.

[Wu *et al.*, 2023a] Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. In *International Conference on Learning Representations*, 2023.

[Wu *et al.*, 2023b] Yangyang Wu, Xiaoye Miao, Xinyu Huang, and Jianwei Yin. Jointly imputing multi-view data with optimal transport. In *AAAI*, volume 37, pages 4747–4755, 2023.

[Wu *et al.*, 2023c] Yangyang Wu, Xiaoye Miao, Zilinghan Li, Shilan He, Xinkai Yuan, and Jianwei Yin. An efficient generative data imputation toolbox with adversarial learning. In *ICDE*, pages 3651–3654, 2023.

[Yi *et al.*, 2023a] Kun Yi, Qi Zhang, Wei Fan, Hui He, Liang Hu, Pengyang Wang, Ning An, Longbing Cao, and Zhendong Niu. Fouriergnn: rethinking multivariate time series forecasting from a pure graph perspective. In *37th Conference on Neural Information Processing Systems (NeurIPS 2023)*, pages 1–23, 2023.

[Yi *et al.*, 2023b] Kun Yi, Qi Zhang, Wei Fan, Shoujin Wang, Pengyang Wang, Hui He, Defu Lian, Ning An, Longbing Cao, and Zhendong Niu. Frequency-domain mlps are more effective learners in time series forecasting. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, 2023.

[Yoon *et al.*, 2019] Jinsung Yoon, William R. Zame, and Mihaela van der Schaar. Estimating missing data in temporal data streams using multi-directional recurrent neural networks. *IEEE Transactions on Biomedical Engineering*, 66(5):1477–1490, 2019.

[Zhang *et al.*, 2017] Shuyi Zhang, Bin Guo, Anlan Dong, Jing He, Ziping Xu, and Song Xi Chen. Cautionary tales on air-quality improvement in beijing. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 473(2205):20170457, 2017.

[Zhang *et al.*, 2024] Shunyang Zhang, Senzhang Wang, Hao Miao, Hao Chen, Changjun Fan, and Jian Zhang. Score-cdm: Score-weighted convolutional diffusion model for multivariate time series imputation. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI 2024, Jeju, South Korea, August 3-9, 2024*, pages 2551–2560, 2024.

[Zhou *et al.*, 2021] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Thirty-Fifth AAAI Conference on Artificial Intelligence*, pages 11106–11115, 2021.

[Zhou *et al.*, 2023] Tian Zhou, Peisong Niu, Xue Wang, Liang Sun, and Rong Jin. One fits all: Power general time series analysis by pretrained LM. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems*, 2023.

[Zhu *et al.*, 2023] Yifan Zhu, Fangpeng Cong, Dan Zhang, Wenwen Gong, Qika Lin, Wenzheng Feng, Yuxiao Dong, and Jie Tang. Wingnn: Dynamic graph neural networks with random gradient aggregation window. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, page 3650–3662, 2023.