Motif-aware Graph Neural Networks for Networked Time Series Imputation

Nourhan Ahmed [1, 2]**, Vijaya Krishna Yalavarthi** [1]**, Lars Schmidt-Thieme** [1, 2]

[1] Information Systems and Machine Learning Lab (ISMLL), University of Hildesheim, Germany
[2] VWFS Data Analytics Research Center
{ahmed, yalavarthi, schmidt-thieme}@ismll.uni-hildesheim.de

Abstract

Networked time series are time series on a graph, one for each node, with applications in traffic and weather monitoring. Graph neural networks are natural candidates for networked time series imputation and have recently outperformed existing alternatives such as recurrent and generative models for time series imputation as they utilize a relational inductive bias for imputation. However, existing GNN-based approaches fail to capture the higher-order topological structure between sensors, which are shaped by recurring substructures in the graph, referred to as temporal motifs. In addition, it remains uncertain which motifs are the most pivotal motifs guiding the imputation task in networked time series. In this paper, we fill in this gap by proposing a graph neural network designed to leverage motif structures within the network by employing weighted motif adjacency matrices to capture higher-order neighborhood information. In particular, (1) we design a motif-wise multi-view attention module that explicitly captures various higher-order structures along with an attention mechanism that automatically assigns high weights to informative ones in order to maximize the use of higher-order information. (2) We introduce a gated fusion module by merging gated recurrent networks and graph convolutional networks to capture the spatial and temporal dependency in order to reflect the intricate impacts of temporal and spatial influence. Experimental results demonstrate that when compared to state-of-the-art models for time-series imputation tasks, our proposed model can reduce the error by around 19%.

Introduction

Multivariate time series data are ubiquitous in real life, including medical logs, meteorological records, and traffic data (Che et al. 2018). In complex linked systems, missing values occur due to sensor failures, poor data collection processes, and human errors. Accordingly, missing values can cause major issues in time series analysis, and suitable values must be used to impute the missing values (Cini, Marisca, and Alippi 2022). Recent breakthroughs in deep learning have allowed new ways to deal with missing data, including Recurrent Neural Networks (RNNs) (Che et al. 2018; Cao et al. 2018; Liu et al. 2019) and Generative Adversarial Networks (GANs) (Yoon, Jordon, and Schaar

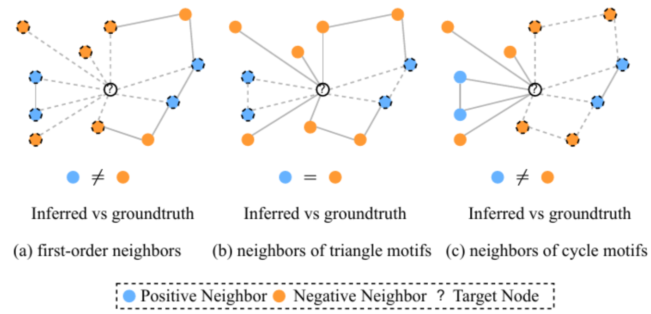


Figure 1: Demonstration using node classification task using YELP dataset (Dou et al. 2020). (a) Using only a node's direct neighbors might lead to incorrect predictions. (b) Considering groups of three connected nodes (triangles) can improve accuracy. (c) However, using less informative groups like cycles might not help.

2018; Luo et al. 2018, 2019). These approaches have been the workhorse of time series imputation. These approaches, however, do not account for the higher-order topological structures among sensors, which correspond to small, recurring subgraphs.

Graph Neural Networks (GNNs) have recently gained prominence as a powerful alternative, particularly with the introduction of the concept of networked time series (Zhu et al. 2021). This concept refers to a set of time series data that are interconnected or interdependent through an underlying network structure which was introduced for the first time in the context of time imputation by Cini, Marisca, and Alippi (2022). GNNs are particularly suitable for this task because they better capture the relational interdependence of networked time series. While these GNN-based approaches have shown promising results, they are still in their early phases. Two significant issues remain open and need to be addressed.

The first issue is about *structural diversity*. Current graph-based imputation approaches either employ edge-defined one-hop neighborhoods to propagate information (Lee et al. 2019; Cini, Marisca, and Alippi 2022) or depend on random walks and their variants (Wang et al. 2023a). These methods are effective for capturing the lower-order graph structure, such as first-order and second-order neighbors, and for identifying prominent nodes. However, they often fall short in capturing the structural diversity of networks, particularly in situations where higher-order topology structures, represented by small, recurring subgraphs, are crucial. As illustrated in Figure 1(a), the first-order neighbors of the target node mostly have different labels (negative neighbors). However, when considering higher-order relations, the target node and its same-label (positive) neighbors form triangle relations, as shown in Figure 1(b). These triangle relations indicate a stronger bond and can help correctly infer the target node's label. Motifs, recurring small patterns or subgraphs, are fundamental in graph theory and demonstrate their ability to characterize the higher-order structure of complex networks (Friggeri, Chelius, and Fleury 2011; Wang et al. 2023b)(see Figure 2). Therefore, incorporating motifs is essential for capturing higher-order neighborhood characteristics in graphs and improving tasks such as networked time series imputation (Wang et al. 2023b; Lee et al. 2019). Although motifs have been studied in contexts like graph and node classification (Chen et al. 2023), to our knowledge, no previous motif-based GNN method has specifically targeted missing value imputation for networked time series.

The second issue concerns *structural information fusion*, stemming from the complexity of networked time series data. This data consists of numerous subgraphs, but only a select subset is valuable for accurate imputation. The challenge lies in choosing the most informative subgraphs to effectively characterize the structural information of nodes and guide the imputation task. For example, consider the target node and its neighboring nodes as shown in Figure 1. Within this neighborhood, various motifs can be induced, such as cycle-shaped and triangle-shaped motifs. Figure 1(b) shows that triangle-shaped motifs are the most informative substructure for the target node. Using less informative motifs, such as cycle motifs instead of triangle motifs, as depicted in Figure 1(c), may lead to an incorrect inference of the target node's class. Therefore, to enhance the accuracy of networked time series imputation, it is crucial to identify and select key subgraphs from the extensive pool available and optimize the integration of their structural information. Effectively fusing the most informative structural information remains a significant challenge in the context of networked time series imputation.

Our approach. To address the aforementioned challenges, we propose a novel Motif-aware graph neural network model (**Motif-GNN**) aimed at jointly modeling diverse structural information, including both low-order and high-order structural features. To effectively capture and utilize network information, particularly higher-order complexities in networked time series data, our approach incorporates a temporal motif embedding module for deriving various network motifs. These motifs are then integrated using a motif-based multi-view attention module, leveraging their significance as fundamental components of complex networks. To comprehensively model the combined effects of structural and temporal influences, we employ a gated fusion layer. This layer integrates a gated recurrent design within the Graph Convolutional Network (GCN), enabling the fusion of motif representations, feature matrices, original adjacency matrices, and temporal aspects of the time series data.

The contributions are summarized as follows:

- **Novelty**. We make the first attempt to utilize higher-order structures within graphs, specifically temporal motifs, for the task of networked time series imputation.
- **Methodology**. In Motif-GNN, we introduce a novel approach to capture and utilize network information, focusing particularly on higher-order complexities within complex networks. Our method incorporates a motif-based multi-view attention module to integrate motifs, which are fundamental building blocks of complex networks, into multi-view structural representation. This module employs attention mechanisms to fuse the diverse perspectives offered by different motifs. Additionally, we implement a gated fusion layer to model the dependencies between the fused high-order structural information and temporal dynamics, thereby enhancing the accuracy of imputation tasks.
- **Experiment**. Empirical results show that Motif-GNN outperforms baseline models by approximately 19% for time series imputation across various real-world datasets.

Related Work

Time Series Imputation. Missing value imputation in time series has a considerable body of literature, including deletion-based approaches (McKnight et al. 2007; Wothke 2000), traditional interpolation methods (e.g., mean and median) (Acuna and Rodriguez 2004; Donders et al. 2006; Kantardzic 2011), and techniques such as neighbor-based (Amiri and Jensen 2016; Sun et al. 2020), constraint-based (Song and Chen 2011; Song, Chen, and Cheng 2013) and regression-based methods (Cleveland and Loader 1996; Box et al. 2015). Later works use RNNs to capture temporal information. Early methods concatenated timestamps with input data for imputation (Lipton, Kale, and Wetzel 2016; Che et al. 2018; Cao et al. 2018; Liu et al. 2019). Models like the Gated Recurrent Unit (GRU) have been used to extract long-term information, as seen in the GRU-D model (Che et al. 2018), which introduces a decay mechanism for input variables to model missing patterns. BRITS (Cao et al. 2018), another RNN-based model, uses bidirectional mechanisms for multivariate time series imputation. In addition, there are several methods introduced as an adaptation of GANs (Goodfellow et al. 2014; Luo et al. 2018) for time series imputation. Other successful works are entirely GAN architecture only (Yoon, Jordon, and Schaar 2018; Miao et al. 2021). For example, GAIN (Yoon, Jordon, and Schaar 2018) generalizes GANs (Goodfellow et al. 2014) for missing data imputation. Other works utilize attention mechanisms to leverage spatial-temporal information such as SPIN (Marisca, Cini, and Alippi 2022), SaD (Ahmed and Schmidt-Thieme 2024), and ReCTSi (Lai et al. 2024). Additionally, 2D vision backbones have been introduced for time series imputation. As an example, the TimesNet model (Wu et al. 2023) imputes missing values by transforming 1D time series into 2D tensors. A key drawback of these methods is that they either overlook the graph's topological information or depend exclusively on message passing through lower-order structures like first-order and second-order neighbors.

Networked Time Series Imputation. The concept of networked time series which refers to a set of time series data that are interconnected or interdependent through an underlying network structure was introduced by Cini, Marisca, and Alippi (2022) that presents GRIN, the first graph-based architecture for networked time series imputation, which rebuilds missing data in multivariate time series by learning spatio-temporal representations via message passing. Another work by Chen, Wang, and Xu (2023) combined pre-trained large language models with graph attention networks, leveraging their complementary strengths for enhanced performance. Later, the PoGeVon model (Wang et al. 2023a) utilized a variational autoencoder with an encoder that incorporates Random Walk with Restart (RWR)-based node position embeddings to impute missing node features and graph structures.

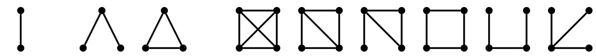
Network Motifs. Network motifs are recurring patterns or subgraphs of connections that occur in complex networks at a higher frequency than in randomized networks. These motifs are considered the building blocks of complex network structures (Milo et al. 2002; Lee et al. 2019; Benson, Gleich, and Leskovec 2016). Standard GNNs are implied to be unable to capture high-order structures since their power is limited to that of the 1-dimensional Weisfeiler-Leman graph isomorphism test (Morris et al. 2019; Lee et al. 2019). As a result, a line of study on GNN models aims to enhance representation learning by the inclusion of motifs (Lee et al. 2019; Fu, Zhou, and He 2020; Chen and Ying 2024; Sun et al. 2024). Specifically, these models learn high-order graph structure by using motif-based adjacency matrices derived from distinct motifs (Yang et al. 2022; Wang et al. 2022; Chen et al. 2023). These methods demonstrate the advantages of incorporating motifs into graph representation learning on a variety of downstream tasks. To the best of our knowledge, we are the first to incorporate motifs into the time series imputation task and aim to use high-order structure information in this task.

Preliminaries

First, we explain the problem of time series imputation, followed by representing time series as a dynamic graph, and then provide the background knowledge of the motif and motif adjacency matrix.

Time Series Imputation

Time series imputation is the problem of imputing the missing values in a partially observed multivariate time series. Formally, let $\mathbf{X} := (\boldsymbol{X}_1, \ldots, \boldsymbol{X}_T) \in \left((\mathbb{R} \cup \{\texttt{NaN}\})^{N \times D}\right)^T$ represent the partially observed multivariate time series, where $T \in \mathbb{N}$ is the total number of time steps, $N \in \mathbb{N}$ is the total number of sensors or instances, and $D \in \mathbb{N}$ is the number of channels in each sensor.Here, $X_{t,n,d} \neq \texttt{NaN}$ indicates an observed value; otherwise, it is missing. Since models cannot directly work on NaN values, we replace NaN in $\mathbf{X}$ with 0 and introduce a



(a) 2-node (b) 3-node (c) 4-node

Figure 2: Motifs of different sizes

mask sequence $\mathbf{M} := (\boldsymbol{M}_1, \ldots, \boldsymbol{M}_T) \in \left(\{0,1\}^{N \times D}\right)^T$, where $M_{t,n,d} = 1$ indicates that $X_{t,n,d}$ is observed; otherwise, it is missing.

Additionally, $\mathbf{A} := (\boldsymbol{A}_1, \ldots, \boldsymbol{A}_T) \in \left(\mathbb{R}^{N \times N}\right)^T$ represents the relationship between the sensors at all timestamps, e.g., $A_{t,m,n}$ is the distance between the sensor m and n with $m, n \in \{1, \ldots, N\}$ at timestamp t. Given $\mathbf{X}, \mathbf{M}, \mathbf{A}$, the goal is to find $\hat{\mathbf{X}} := (\hat{\boldsymbol{X}}_1, \ldots, \hat{\boldsymbol{X}}_T) \in \left(\mathbb{R}^{N \times D}\right)^T$ where $\hat{X}_{t,n,d}$ is the imputed observation of $X_{t,n,d}$ when $M_{t,n,d} = 0$ (when $M_{t,n,d} = 1$ we assume $\hat{X}_{t,n,d} = X_{t,n,d}$).

Time Series as Dynamic Graph

A partially observed multivariate time series, as defined above, can be conceptualized as a graph evolving over time, $\mathbb{G} := (\mathcal{G}_1, \ldots, \mathcal{G}_T)$. At each timestamp t, $\mathcal{G}_t := (\boldsymbol{X}_t, \boldsymbol{M}_t, \boldsymbol{A}_t, \mathcal{E}_t)$ is an undirected graph of N many nodes where: $\mathcal{E}_t \subseteq \{1, \ldots, N\} \times \{1, \ldots, N\}$ is set of edges, $\boldsymbol{X}_t \in \mathbb{R}^{N \times D}$ is the node attribute matrix (partially observed), $\boldsymbol{M}_t \in \mathbb{R}^{N \times D}$ is the mask matrix corresponding to $\boldsymbol{X}_t$, and $\boldsymbol{A}_t \in \mathbb{R}^{N \times N}$ is the weighted adjacency matrix, with $A_{t,m,n}$ indicating the distance between nodes m and n at timestamp t. We assume an edge exists between nodes m and n at timestamp t if and only if $A_{t,m,n} \neq 0$ i.e., $\{m, n\} \in \mathcal{E}_t \iff A_{t,m,n} \neq 0$.

Motif and Motif-induced Adjacency Matrix

A network motif $\mathcal{G}^{\text{MOT}}$ is a connected graph of $M \geq 2$ nodes that represent recurring patterns or subgraphs within a network. In Figure 2, we illustrate 2, 3, and 4-node motifs.

Following Chen et al. (2020), we construct a motif-induced adjacency matrix. To do this, we first create a motif instance set for a motif $\mathcal{G}^{\text{MOT}}$ on a graph $\mathcal{G}$ which is the set of all the edge subsets that can form a motif instance (we ignore subscript t when the context is clear):

$$\mathbb{M} := \{\mathcal{E}' \subseteq \mathcal{E} \mid \mathcal{G}[\mathcal{E}'] \simeq \mathcal{G}^{\text{MOT}}\} \tag{1}$$

where $\mathcal{G}[\mathcal{E}']$ is induced subgraph of $\mathcal{G}$ on edges $\mathcal{E}'$, and $\simeq$ indicates isomorphism.

Now, the motif-based adjacency matrix $\boldsymbol{B} \in \mathbb{R}^{N \times N}$ of graph $\mathcal{G}$ for motif $\mathcal{G}^{\text{MOT}}$ is given as:

$$\boldsymbol{B}_{ij} = \sum_{\mathcal{E}' \in \mathbb{M}} I(\{i, j\} \in \mathcal{E}') \tag{2}$$

where $I(\cdot)$ is an indicator function, i.e., $I(x) = 1$ if the statement x is true and 0 otherwise.

We assume a total of K many motifs hence we have an K many adjacency matrices. $\boldsymbol{B}^k$ is $\mathcal{G}^{\text{MOT}k}$-induced adjacency matrix for graph $\mathcal{G}$:

$$(\boldsymbol{B}^1, \ldots, \boldsymbol{B}^K) = \text{mot-ind}(\mathcal{G}, \mathcal{G}^{\text{MOT}1}, \ldots, \mathcal{G}^{\text{MOT}K}) \tag{3}$$

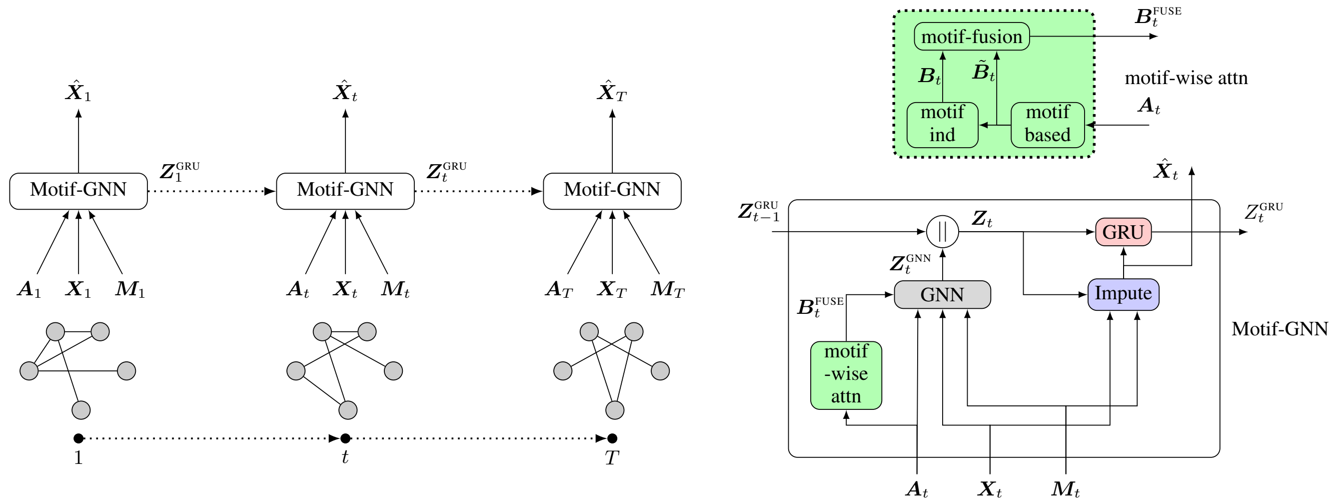


Figure 3: The architecture of Motif-GNN. The terms "motif-ind," "motif-based," "motif-wise attn," and "impute" refer to the Motif-induced adjacency matrix, Motif-based adjacency matrix, Motif-wise multi-view attention, and Imputation network, respectively, as described in Sections 3.3, 3.4, 4.1, and 4.3.

Motif-based adjacency matrix

Different motifs create distinct local neighborhoods for nodes in a network, as shown in Figure 2. The connections within these neighborhoods, represented by weights in the motif-induced adjacency matrix $\boldsymbol{B}$, can vary because the frequency of specific motifs differs between node pairs. To generalize these connections, Lee et al. (2019) and Rossi, Ahmed, and Koh (2018) proposed motif matrix functions to create motif-based adjacency matrices, $\tilde{\boldsymbol{B}} \in \mathbb{R}^{N\times N}$.

1. Motif Transition Matrix. In a random walk on the graph with adjacency matrix $\boldsymbol{B}$, the transition probabilities are given by $P_{ij} = \frac{B_{ij}}{B_i}$, where $B_i = \sum_j B_{ij}$. Thus, the random walk motif transition matrix is calculated as:

$$\tilde{\boldsymbol{B}}^{\text{TRANS}} := \boldsymbol{D}^{-1}\boldsymbol{B} \tag{4}$$

where $\boldsymbol{D}$ is the diagonal degree matrix of $\boldsymbol{B}$.

2. Absolute Motif Laplacian. The absolute Laplacian matrix can be defined as follows:

$$\tilde{\boldsymbol{B}}^{\text{LAPL}} := \boldsymbol{D} + \boldsymbol{B} \tag{5}$$

3. Symmetric Normalized Motif Matrix with Row-wise Max. A symmetric normalized matrix, similar to the normalized Laplacian, is calculated as:

$$\tilde{\boldsymbol{B}}^{\text{SYMM}} := \boldsymbol{D}^{-1/2}(\boldsymbol{B} + \boldsymbol{R})\boldsymbol{D}^{-1/2} \tag{6}$$

where $\boldsymbol{R}$ is a diagonal matrix with $\boldsymbol{R}_{ii} = \max_j \boldsymbol{B}_{ij}$.

There are a total of $3K$ many motif-based adjacency matrices:

$$\tilde{\boldsymbol{B}} := \text{motif-based}(\boldsymbol{B}) \tag{7}$$

Proposed Architecture

Our networked time series is represented as a dynamic graph. GNNs are effective at learning embeddings from graph data, while RNN-based architectures such as GRUs excel at capturing sequential information in time series. To leverage both approaches, we propose a novel model with two components, as illustrated in Figure 3. The first component is a GNN that encodes the static graph at each time point. The second component is a GRU that predicts the latent representation of the feature matrix for the next time step. Finally, a decoder is used to predict the missing features in the feature matrix in an auto-regressive manner.

Temporal Motif Embedding

GNNs embed the input graph into latent representation by passing information between neighbor nodes via connected edges specified by adjacency matrices. Here, we have 3 different kinds of adjacency matrices: 1) The directly provided adjacency matrix with the time series: $\boldsymbol{A}$, 2) K many motif-induced adjacency matrices: $\boldsymbol{B}^1, \ldots, \boldsymbol{B}^K$, and 3) Three motif-based adjacency matrices for each motif-induced adjacency matrix with a total of $3K$: $\tilde{\boldsymbol{B}}^1, \ldots, \tilde{\boldsymbol{B}}^{3K}$. A simple approach could be to use separate GNNs on each adjacency matrix and then concatenate their output. However, $1 + K + 3K$ many graph neural networks would be computationally expensive. Therefore, first we combine the adjacency matrices derived from motifs to one motif-fused adjacency matrix $\boldsymbol{B}^{\text{FUSE}} \in \mathbb{R}^{M\times M}$. Then, we concatenate the GNN outputs for the original adjacency matrix $\boldsymbol{A}_t$ and fused matrix $\boldsymbol{B}_t^{\text{FUSE}}$ at every timepoint t.

Motif-wise Multi-view Attention. To achieve this, we use separate attention mechanisms for the motif-based and motif-induced matrices as follows:

$$\boldsymbol{B}^{\text{FUSE}} := \sum_{k=1:K} \alpha_k \boldsymbol{B}^k + \sum_{k=1:3K} \beta_k \tilde{\boldsymbol{B}}^k \tag{8}$$

where α_k and β_k are the attention coefficients calculated for each motif-induced matrix $\boldsymbol{B}_k$ and each motif-based matrix

$\tilde{\boldsymbol{B}}^k$, respectively. This fusion mechanism ensures that the most relevant neighborhood information from both types of motif matrices is integrated into the final structural representation. The attention coefficients, α_k and β_k, are calculated using the following softmax function to create a probability distribution over the motifs:

$$\alpha_k = \frac{\exp(e_k)}{\sum_{m=1}^{K} \exp(e_m)}, \quad \beta_k = \frac{\exp(f_k)}{\sum_{n=1}^{3K} \exp(f_n)}, \tag{9}$$

where e_k and f_k are the intermediate attention scores computed as follows:

$$e_k = \boldsymbol{w}_1^T \tanh(\boldsymbol{B}^k \boldsymbol{w}_2), \quad f_k = \boldsymbol{w}_3^T \tanh(\tilde{\boldsymbol{B}}^k \boldsymbol{w}_4) \tag{10}$$

Here, $\boldsymbol{w}_1, \boldsymbol{w}_2, \boldsymbol{w}_3, \boldsymbol{w}_4 \in \mathbb{R}^{N\times 1}$ are learnable weight vectors.

Now, the GNN embedding at layer l at timestep t is given as:

$$\begin{aligned} \boldsymbol{Z}_t^{\text{GNN},l} &= \text{GNN}_t^l(\boldsymbol{A}_t, \boldsymbol{B}_t^{\text{FUSE}}, \boldsymbol{Z}_t^{\text{GNN},l-1}) \\ &:= \phi(\boldsymbol{A}_t' \boldsymbol{Z}_t^{\text{GNN},l-1} \boldsymbol{W}_{\text{A}}^l \| \boldsymbol{B}_t^{\text{FUSE}} \boldsymbol{Z}_t^{\text{GNN},l-1} \boldsymbol{W}_{\text{B}}^l) \end{aligned} \tag{11}$$

where $\boldsymbol{A}_t' = \boldsymbol{D}_t^{-\frac{1}{2}} \boldsymbol{A}_t \boldsymbol{D}_t^{-\frac{1}{2}}$ denotes the normalized symmetric adjacency matrix at time step t, $\boldsymbol{Z}_t^{\text{GNN},l-1}$ is the embedding from the $l-1$-th layer, and $\boldsymbol{W}_{\text{A}}^l$ and $\boldsymbol{W}_{\text{B}}^l$ are the weight matrices for the l-th layer, $\|$ denotes the concatenation operator, and ϕ is an activation function. The input to the first layer $\boldsymbol{Z}_t^{\text{GNN},0}$ is the concatenation of original feature matrix $\boldsymbol{X}_t$ and mask $\boldsymbol{M}_t$. Note that, the parameters of GNN are shared across all the timepoints. The final embedding, after L many GNN layers, $\boldsymbol{Z}_t^{\text{GNN},L}$ is considered as $\boldsymbol{Z}_t^{\text{GNN}}$.

Gated Fusion Layer

While GNN can provide embedding for a static graph at each timepoint, it cannot accommodate the information from the previous timesteps. For this, we use a simple GRU modified to incorporate three things: $\boldsymbol{X}_t$, embedding from GNN $\boldsymbol{Z}_t^{\text{GNN}}$, and previous GRU hidden state $\boldsymbol{Z}_{t-1}^{\text{GRU}}$. Note that $\boldsymbol{X}_t$ consists of missing values. Similar to GNN we can concatenate $\boldsymbol{M}_t$ with $\boldsymbol{X}_t$ and pass it as an input to the GRU. However, owning to the ideas of kalman filtering, we impute the missing values of $\boldsymbol{X}_t$, calling $\hat{\boldsymbol{X}}_t$ using the past GRU hidden state and the current GNN embedding.

$$\boldsymbol{Z}_t^{\text{GRU}} := \text{GRU}(\hat{\boldsymbol{X}}_t, \boldsymbol{Z}_t) \quad \text{with} \quad \boldsymbol{Z}_t = \boldsymbol{Z}_{t-1}^{\text{GRU}} \| \boldsymbol{Z}_t^{\text{GNN}}$$

Imputation Network

For the imputation of $\boldsymbol{X}_t$, we use, $\boldsymbol{Z}_t$, the concatenation of GRU past hidden state and the latent embedding given by the GNN. First, we use a simple feed-forward layer (FF) to transform $\boldsymbol{Z}_t$ from embedding space to $\tilde{\boldsymbol{X}}_t$ in input space:

$$\tilde{\boldsymbol{X}}_t = \text{FF}(\boldsymbol{Z}_t) \tag{12}$$

Now, the final imputation of the missing values in the time series is:

$$\begin{aligned} \hat{\boldsymbol{X}}_t &= \text{Imputation Network}(\boldsymbol{Z}_t, \boldsymbol{X}_t, \boldsymbol{M}_t) \\ &:= \boldsymbol{M}_t \odot \boldsymbol{X}_t + (1 - \boldsymbol{M}_t)\tilde{\boldsymbol{X}}_t \end{aligned} \tag{13}$$

where $\odot$ is element-wise multiplication and $\hat{X}_t$ is the final feature matrix with missing values filled in. Algorithm 1 provides the flow of the proposed model.

Algorithm 1: The Framework of Motif-GNN.

Ensure: $\mathsf{X}, \mathsf{A}, \mathsf{M}, (\mathcal{G}^{\text{MOT}\,1}, \ldots, \mathcal{G}^{\text{MOT}\,K})$, motif-ind, motif-based, motif-fusion, Imputation Network, GNN, GRU

1: **for** $t = 1 : T$ **do**
2: $\quad \boldsymbol{B}_t \leftarrow$ motif-ind$(\boldsymbol{A}_t, \mathcal{G}^{\text{MOT}\,1:K})$
3: $\quad \tilde{\boldsymbol{B}}_t \leftarrow$ motif-based$(\boldsymbol{B}_t)$
4: $\quad \boldsymbol{B}_t^{\text{FUSE}} \leftarrow$ motif-fusion$(\boldsymbol{B}_t, \tilde{\boldsymbol{B}}_t)$
5: $\quad \boldsymbol{Z}_t^{\text{GCN}} \leftarrow \text{GNN}(\boldsymbol{A}_t, \boldsymbol{X}_t, \boldsymbol{M}_t, \boldsymbol{B}_t^{\text{FUSE}})$
6: $\quad \boldsymbol{Z}_t \leftarrow \boldsymbol{Z}_{t-1}^{\text{GRU}} \| \boldsymbol{Z}_t^{\text{GNN}}$
7: $\quad \hat{\boldsymbol{X}}_t \leftarrow$ Imputation Network$(\boldsymbol{Z}_t, \boldsymbol{M}_t)$
8: $\quad \boldsymbol{Z}_t^{\text{GRU}} \leftarrow \text{GRU}(\hat{\boldsymbol{X}}_t, \boldsymbol{Z}_t)$
9: **end for**
10: $\hat{\mathsf{X}} \leftarrow (\hat{\boldsymbol{X}}_1, \ldots, \hat{\boldsymbol{X}}_T)$
11: **return** $\hat{\mathsf{X}}$

Objective Function and Training

The proposed model is trained using a joint optimization strategy that combines imputation and reconstruction losses into a unified objective function (Du, Côté, and Liu 2023) as follows:

$$\min_{\theta}\Bigg(\frac{1}{\|\boldsymbol{M}_t\|_1} \|\boldsymbol{M}_t \odot (\tilde{\boldsymbol{X}}_t - \boldsymbol{X}_t)\|_1 + \gamma \cdot \frac{1}{\|(1-\boldsymbol{M}_t)\|_1} \|(1-\boldsymbol{M}_t) \odot (\tilde{\boldsymbol{X}}_t - \boldsymbol{X}_t)\|_1 \Bigg) \tag{14}$$

Here, $\odot$ denotes element-wise multiplication (Hadamard product), $\|\cdot\|_1$ represents the element-wise ℓ_1-norm, and $\boldsymbol{M}_t$ and $(1 - \boldsymbol{M}_t)$, indicate the number of missing and observed entries, respectively. The parameter $\gamma \in [0, 1]$ controls the trade-off between imputation and reconstruction.

Experiments

In this section, we present our experiments to evaluate the proposed model, Motif-GNN for networked time series imputation. The experiments seek to answer the following research questions: (i) RQ1: How effective is Motif-GNN for networked time series imputation compared to the state-of-the-art imputation models? (ii) RQ2: How do different components of our model contribute to its overall performance?

Datasets

We assess the utility of Motif-GNN using four real-world datasets comprising one meteorological dataset and three traffic datasets.

- **Meteorological Dataset**. The air quality dataset comprises observations from 36 monitoring sites (Yi et al. 2016), collected between May 1, 2014, and April 30, 2015, totaling 8,759 timestamps (Yi et al. 2016; Zheng et al. 2015). The adjacency matrix was constructed using a thresholded Gaussian kernel, following the same method of Cini, Marisca, and Alippi (2022) and Wang et al. (2023a).

Dataset	Model	General Missing		Spatio-temporal Block Missing		Temporal Block Missing	
		MAE	MRE (%)	MAE	MRE (%)	MAE	MRE (%)
Air Quality	**MI**	59.81 ± 0.00	81.29 ± 0.00	61.45 ± 0.00	83.41 ± 0.00	62.78 ± 0.00	84.27 ± 0.00
	MF (Luo et al. 2014)	38.56 ± 0.24	52.95 ± 0.36	38.73 ± 0.24	52.81 ± 0.22	36.32 ± 0.25	49.91 ± 0.34
	GRU-D (Che et al. 2018)	36.71 ± 0.32	47.12 ± 0.35	38.87 ± 0.27	49.16 ± 0.31	38.14 ± 0.32	48.12 ± 0.35
	BRITS (Cao et al. 2018)	38.79 ± 0.44	47.58 ± 0.32	38.58 ± 0.46	46.52 ± 0.26	35.56 ± 0.62	43.24 ± 0.36
	GRIN (Cini, Marisca, and Alippi 2022)	29.51 ± 0.26	39.41 ± 0.32	33.14 ± 0.26	42.55 ± 0.31	31.22 ± 0.27	42.47 ± 0.36
	SPIN (Marisca, Cini, and Alippi 2022)	27.32 ± 0.29	32.08 ± 0.38	35.18 ± 0.23	39.55 ± 0.31	33.17 ± 0.24	37.45 ± 0.32
	TimesNet (Wu et al. 2023)	32.54 ± 0.31	37.15 ± 0.32	38.23 ± 0.24	41.93 ± 0.37	34.28 ± 0.24	38.56 ± 0.33
	PoGeVon (Wang et al. 2023a)	19.49 ± 1.10	26.26 ± 1.50	21.34 ± 0.89	27.36 ± 0.94	22.42 ± 1.28	27.92 ± 1.36
	Motif-GNN (ours)	**15.43 ± 0.32**	**20.41 ± 0.61**	**17.27 ± 0.23**	**22.16 ± 0.24**	**16.98 ± 0.31**	**22.41 ± 0.52**
PeMS-LA	**MI**	215.19 ± 0.00	40.64 ± 0.00	219.85 ± 0.00	44.09 ± 0.00	224.27 ± 0.00	44.65 ± 0.00
	MF (Luo et al. 2014)	77.23 ± 0.27	14.51 ± 0.48	79.78 ± 0.24	17.12 ± 0.36	77.87 ± 0.25	15.12 ± 0.45
	GRU-D (Che et al. 2018)	46.52 ± 0.28	9.78 ± 0.33	48.37 ± 0.44	10.05 ± 0.14	49.04 ± 0.26	10.12 ± 0.28
	BRITS (Cao et al. 2018)	47.72 ± 0.28	6.98 ± 0.36	49.77 ± 0.44	7.75 ± 0.12	49.31 ± 0.29	7.21 ± 0.32
	GRIN (Cini, Marisca, and Alippi 2022)	46.52 ± 0.22	9.18 ± 0.32	48.77 ± 0.44	10.65 ± 0.32	48.34 ± 0.26	10.76 ± 0.32
	SPIN (Marisca, Cini, and Alippi 2022)	26.75 ± 0.15	5.24 ± 0.28	32.86 ± 0.41	7.65 ± 0.42	38.88 ± 0.24	6.76 ± 0.35
	TimesNet (Wu et al. 2023)	27.45 ± 0.27	7.04 ± 0.34	34.19 ± 0.35	9.49 ± 0.32	39.62 ± 0.28	6.88 ± 0.34
	PoGeVon (Wang et al. 2023a)	23.91 ± 0.25	4.26 ± 0.02	26.91 ± 0.25	6.15 ± 0.02	25.32 ± 0.28	4.87 ± 0.04
	Motif-GNN (ours)	**18.21 ± 0.12**	**3.96 ± 0.02**	**21.81 ± 0.01**	**4.11 ± 0.02**	**20.67 ± 0.12**	**3.81 ± 0.06**
PeMS-BA	**MI**	192.05 ± 0.00	47.41 ± 0.00	195.75 ± 0.00	49.98 ± 0.00	196.05 ± 0.00	49.78 ± 0.00
	MF (Luo et al. 2014)	57.25 ± 1.15	14.11 ± 0.22	59.65 ± 1.22	16.21 ± 0.21	63.25 ± 1.26	17.15 ± 0.27
	GRU-D (Che et al. 2018)	36.56 ± 1.21	9.02 ± 0.14	39.24 ± 1.24	10.78 ± 0.14	38.52 ± 1.22	10.25 ± 0.12
	BRITS (Cao et al. 2018)	31.27 ± 1.07	7.79 ± 0.10	34.47 ± 1.24	9.25 ± 0.14	32.55 ± 1.12	8.62 ± 0.18
	GRIN (Cini, Marisca, and Alippi 2022)	30.34 ± 1.07	6.92 ± 0.30	32.65 ± 1.26	8.67 ± 0.14	33.34 ± 1.07	7.92 ± 0.70
	SPIN (Marisca, Cini, and Alippi 2022)	26.75 ± 1.23	6.15 ± 0.33	29.11 ± 1.24	6.91 ± 0.19	28.87 ± 1.09	6.39 ± 0.69
	TimesNet (Wu et al. 2023)	25.85 ± 1.15	6.46 ± 0.32	28.05 ± 1.14	6.81 ± 0.16	30.32 ± 0.94	6.26 ± 0.52
	PoGeVon (Wang et al. 2023a)	22.19 ± 0.05	5.50 ± 0.01	22.19 ± 0.46	5.51 ± 0.01	24.31 ± 0.06	6.23 ± 0.02
	Motif-GNN (ours)	**17.25 ± 0.16**	**4.21 ± 0.01**	**19.65 ± 0.26**	**5.14 ± 0.01**	**19.04 ± 0.18**	**5.47 ± 0.02**
PeMS-SD	**MI**	208.19 ± 0.00	52.91 ± 0.00	221.19 ± 0.00	55.36 ± 0.00	217.27 ± 0.00	57.63 ± 0.00
	MF (Luo et al. 2014)	45.46 ± 0.02	11.78 ± 0.01	48.72 ± 0.02	13.26 ± 0.01	45.46 ± 0.02	11.78 ± 0.01
	GRU-D (Che et al. 2018)	42.32 ± 0.22	9.88 ± 0.01	46.65 ± 0.02	12.18 ± 0.04	45.13 ± 0.21	10.92 ± 0.08
	BRITS (Cao et al. 2018)	35.32 ± 0.22	7.91 ± 0.01	39.12 ± 0.02	9.06 ± 0.01	36.86 ± 0.22	9.04 ± 0.02
	GRIN (Cini, Marisca, and Alippi 2022)	31.42 ± 0.22	6.82 ± 0.01	34.22 ± 0.02	7.29 ± 0.01	34.21 ± 0.25	9.63 ± 0.04
	SPIN (Marisca, Cini, and Alippi 2022)	32.47 ± 0.26	6.94 ± 0.01	36.06 ± 0.03	7.42 ± 0.01	36.12 ± 0.25	9.72 ± 0.04
	TimesNet (Wu et al. 2023)	33.48 ± 0.37	7.05 ± 0.02	37.83 ± 0.02	7.63 ± 0.01	38.04 ± 0.26	9.87 ± 0.05
	PoGeVon (Wang et al. 2023a)	18.99 ± 0.11	4.86 ± 0.01	20.91 ± 0.14	5.91 ± 0.01	20.73 ± 0.16	9.18 ± 0.04
	Motif-GNN (ours)	**15.87 ± 0.14**	**3.72 ± 0.01**	**17.54 ± 0.12**	**4.62 ± 0.04**	**16.92 ± 0.14**	**6.97 ± 0.03**
	Overall Improvement	**21.21%**	**18.26%**	**16.39%**	**21.49%**	**19.53%**	**19.33%**

Table 1: Performance comparison w.r.t. MAE and MRE (%) scores on different datasets for imputation task.

- **Traffic Datasets**. The traffic datasets used in this paper are **PeMS-LA**, **PeMS-BA**, and **PeMS-SD**, collected from January 1 to March 30, 2022, from highways in Los Angeles County, the Bay Area, and San Diego, respectively (Chen et al. 2001). For all traffic datasets, the adjacency matrix was calculated using pairwise distances derived from latitude and longitude values as in (Cini, Marisca, and Alippi 2022; Wang et al. 2023a).

Experimental Setup

Following previous research (Yi et al. 2016; Cini, Marisca, and Alippi 2022), we consider two missing data scenarios: missing completely at random (MCAR), also known as general or point missing (Heitjan and Basu 1996), and block-missing, which includes time-structured (temporal) and space-structured (spatial) missing data. Under the MCAR scenario, datasets are split into 70% for training, 10% for validation, and 20% for testing. In the block-missing scenario, we simulate overlapping time- and space-structured (spatio-temporal) missingness by randomly dropping 5% of the data per sensor and simulating failures with a 0.15% probability, varying the duration from 1 to 4 hours for traffic data and 2 to 6 days for air quality data, following the evaluation protocol in (Cini, Marisca, and Alippi 2022). Additionally, to ensure consistency with baseline models, we consider time-structured missingness in the air quality dataset by partitioning the 1-year data into two parts, using the months of March, June, September, and December as the test set and the rest for training (Yi et al. 2016; Cini, Marisca, and Alippi 2022; Wang et al. 2023a). For traffic datasets, we used the first half of February and the last half of March for testing and the rest for training. Evaluation is conducted using Mean Absolute Error (MAE) and Mean Relative Error

Models	Air Quality		PeMS-LA		PeMS-BA		PeMS-SD	
	MAE	MRE(%)	MAE	MRE(%)	MAE	MRE(%)	MAE	MRE(%)
Motif-GNN	**15.43 ± 0.32**	**20.41 ± 0.46**	**18.21 ± 0.12**	**3.96 ± 0.02**	**17.25 ± 0.16**	**4.21 ± 0.01**	**15.87 ± 0.14**	**3.72 ± 0.01**
Motif-GNN w/o motif-ind	17.49 ± 0.32	23.56 ± 0.38	22.24 ± 0.25	4.76 ± 0.02	21.14 ± 0.05	4.91 ± 0.04	18.18 ± 0.11	4.24 ± 0.05
Motif-GNN w/o motif-based	18.16 ± 0.38	24.35 ± 0.34	23.91 ± 0.25	4.26 ± 0.02	22.19 ± 0.05	5.50 ± 0.01	17.63 ± 0.12	4.06 ± 0.06
Motif-GNN w/o multi-view attn	19.21 ± 0.26	25.67 ± 0.51	21.87 ± 0.29	4.09 ± 0.02	22.04 ± 0.05	4.96 ± 0.08	18.45 ± 0.12	4.39 ± 0.04

Table 2: Analysis of Motif-GNN Model Components

(MRE) as metrics (Yi et al. 2016).

Baselines

- **Mean Imputation (MI)** replaces missing values with the corresponding global mean.
- **Matrix Factorization (MF)** (Luo et al. 2014) decomposes the data matrix into two low-rank matrices and fills in missing values by completing the matrices.
- **GRU-D** (Che et al. 2018) adopts a GRU with two different decay mechanisms to generate missing values.
- **BRITS** (Cao et al. 2018) uses bidirectional RNNs for imputation in both forward and backward directions.
- **GRIN** (Cini, Marisca, and Alippi 2022) generalizes GNNs for multivariate time series imputation through message passing.
- **SPIN** (Marisca, Cini, and Alippi 2022) adopts graph attention networks, leveraging hierarchical attention mechanism for time series imputation.
- **TimesNet** (Wu et al. 2023) employs an inception block to capture both intraperiod and interperiod variations within the 2D space.
- **PoGEVoN** (Wang et al. 2023a) uses variational autoencoder and node positional encodings for imputation.

Time Series Imputation Task

The goal of this experiment is to evaluate whether the Motif-GNN model can improve imputation performance compared to state-of-the-art methods. The results, as shown in Table 1, demonstrate that Motif-GNN consistently surpasses all baseline models in predicting missing values for time series within networked datasets. Specifically, Motif-GNN achieves over an 19% improvement across all datasets compared to the best performing baselines under various missing data scenarios. For example, in the air quality dataset, Motif-GNN reduces MAE by approximately 21% and MRE by around 20% compared to the previous best model, PoGeVon. Overall, Motif-GNN demonstrates improvements of 16–21% in MAE and 18–19% in MRE across the four datasets. While GRIN and PoGeVon also leverage graph topological information, they are less effective than Motif-GNN. The superior performance of Motif-GNN stems from its capability to capture recurring patterns and detailed structural information, underscoring the method's effectiveness.

Ablation Study

We conducted an in-depth investigation of the components in Motif-GNN under the MCAR scenario. The variants of our method include Motif-GNN without the motif-induced matrices, where these matrices are fused using only the first term in Equation 8 ($\text{Motif-GNN}_{\text{w/o motif-ind}}$). Another variant is Motif-GNN without motif-based adjacency matrices, where these matrices are fused using only the second term in Equation 8 ($\text{Motif-GNN}_{\text{w/o motif-based}}$). The last variant tests the importance of selecting key subgraphs from the extensive pool available. This variant includes all motif matrices, but instead of using the module motif-wise multi-view attention, it uses a simple aggregation method (i.e., mean) ($\text{Motif-GNN}_{\text{w/o multi-view attn}}$). The results, shown in Table 2, demonstrate that all three components contribute to performance improvement. In particular, the proposed Motif-GNN model is enhanced by the inclusion of different types of motif-induced and motif-based adjacency matrices.

Conclusion

In this paper, we introduce Motif-GNN, a novel time series imputation model that extends existing GNN-based approaches by learning higher-order network embeddings. The proposed model employs a motif-wise multi-view attention mechanism, which assigns high weights to informative motifs to maximize the use of higher-order structural information for guiding the imputation task. Experimental results on four real-world datasets demonstrate that Motif-GNN consistently outperforms state-of-the-art time series imputation methods, providing an efficient solution for imputing time-series data under various imputation scenarios.

References

Acuna, E.; and Rodriguez, C. 2004. The treatment of missing values and its effect on classifier accuracy. In *Classification, clustering, and data mining applications*, 639–647. Springer.

Ahmed, N.; and Schmidt-Thieme, L. 2024. Structure-aware decoupled imputation network for multivariate time series. *Data Mining and Knowledge Discovery*, 38(3): 1006–1026.

Amiri, M.; and Jensen, R. 2016. Missing data imputation using fuzzy-rough methods. *Neurocomputing*, 205: 152–164.

Benson, A. R.; Gleich, D. F.; and Leskovec, J. 2016. Higher-order organization of complex networks. *Science*, 353(6295): 163–166.

Box, G. E.; Jenkins, G. M.; Reinsel, G. C.; and Ljung, G. M. 2015. *Time series analysis: forecasting and control*. John Wiley & Sons.

Cao, W.; Wang, D.; Li, J.; Zhou, H.; Li, L.; and Li, Y. 2018. Brits: Bidirectional recurrent imputation for time series. *Advances in neural information processing systems*, 31.

Che, Z.; Purushotham, S.; Cho, K.; Sontag, D.; and Liu, Y. 2018. Recurrent neural networks for multivariate time series with missing values. *Scientific reports*, 8(1): 1–12.

Chen, C.; Petty, K.; Skabardonis, A.; Varaiya, P.; and Jia, Z. 2001. Freeway performance measurement system: mining loop detector data. *Transportation research record*, 1748(1): 96–102.

Chen, J.; and Ying, R. 2024. Tempme: Towards the explainability of temporal graph neural networks via motif discovery. *Advances in Neural Information Processing Systems*, 36.

Chen, X.; Cai, R.; Fang, Y.; Wu, M.; Li, Z.; and Hao, Z. 2023. Motif graph neural network. *IEEE Transactions on Neural Networks and Learning Systems*.

Chen, X.; Chen, S.; Yao, J.; Zheng, H.; Zhang, Y.; and Tsang, I. W. 2020. Learning on attribute-missing graphs. *IEEE transactions on pattern analysis and machine intelligence*.

Chen, Y.; Wang, X.; and Xu, G. 2023. Gatgpt: A pre-trained large language model with graph attention network for spatiotemporal imputation. *arXiv preprint arXiv:2311.14332*.

Cini, A.; Marisca, I.; and Alippi, C. 2022. Filling the G_ap_s: Multivariate Time Series Imputation by Graph Neural Networks. In *International Conference on Learning Representations*.

Cleveland, W. S.; and Loader, C. 1996. Smoothing by local regression: Principles and methods. In *Statistical Theory and Computational Aspects of Smoothing: Proceedings of the COMPSTAT'94 Satellite Meeting held in Semmering, Austria, 27–28 August 1994*, 10–49. Springer.

Donders, A. R. T.; Van Der Heijden, G. J.; Stijnen, T.; and Moons, K. G. 2006. A gentle introduction to imputation of missing values. *Journal of clinical epidemiology*, 59(10): 1087–1091.

Dou, Y.; Liu, Z.; Sun, L.; Deng, Y.; Peng, H.; and Yu, P. S. 2020. Enhancing graph neural network-based fraud detectors against camouflaged fraudsters. In *Proceedings of the 29th ACM international conference on information & knowledge management*, 315–324.

Du, W.; Côté, D.; and Liu, Y. 2023. Saits: Self-attention-based imputation for time series. *Expert Systems with Applications*, 219: 119619.

Friggeri, A.; Chelius, G.; and Fleury, E. 2011. Triangles to capture social cohesion. In *2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing*, 258–265. IEEE.

Fu, D.; Zhou, D.; and He, J. 2020. Local motif clustering on time-evolving graphs. In *Proceedings of the 26th ACM SIGKDD International conference on knowledge discovery & data mining*, 390–400.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. *Advances in neural information processing systems*, 27.

Heitjan, D. F.; and Basu, S. 1996. Distinguishing "missing at random" and "missing completely at random". *The American Statistician*, 50(3): 207–213.

Kantardzic, M. 2011. *Data mining: concepts, models, methods, and algorithms*. John Wiley & Sons.

Lai, Z.; Zhang, D.; Li, H.; Zhang, D.; Lu, H.; and Jensen, C. S. 2024. ReCTSi: Resource-efficient Correlated Time Series Imputation via Decoupled Pattern Learning and Completeness-aware Attentions. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 1474–1483.

Lee, J. B.; Rossi, R. A.; Kong, X.; Kim, S.; Koh, E.; and Rao, A. 2019. Graph convolutional networks with motif-based attention. In *Proceedings of the 28th ACM international conference on information and knowledge management*, 499–508.

Lipton, Z. C.; Kale, D.; and Wetzel, R. 2016. Directly modeling missing data in sequences with rnns: Improved classification of clinical time series. In *Machine learning for healthcare conference*, 253–270. PMLR.

Liu, Y.; Yu, R.; Zheng, S.; Zhan, E.; and Yue, Y. 2019. Naomi: Non-autoregressive multiresolution sequence imputation. *Advances in neural information processing systems*, 32.

Luo, X.; Zhou, M.; Leung, H.; Xia, Y.; Zhu, Q.; You, Z.; and Li, S. 2014. An incremental-and-static-combined scheme for matrix-factorization-based collaborative filtering. *IEEE transactions on automation science and engineering*, 13(1): 333–343.

Luo, Y.; Cai, X.; Zhang, Y.; Xu, J.; et al. 2018. Multivariate time series imputation with generative adversarial networks. *Advances in neural information processing systems*, 31.

Luo, Y.; Zhang, Y.; Cai, X.; and Yuan, X. 2019. E2gan: End-to-end generative adversarial network for multivariate time series imputation. In *Proceedings of the 28th international joint conference on artificial intelligence*, 3094–3100. AAAI Press.

Marisca, I.; Cini, A.; and Alippi, C. 2022. Learning to reconstruct missing data from spatiotemporal graphs with sparse observations. *Advances in Neural Information Processing Systems*, 35: 32069–32082.

McKnight, P. E.; McKnight, K. M.; Sidani, S.; and Figueredo, A. J. 2007. *Missing data: A gentle introduction*.

Miao, X.; Wu, Y.; Wang, J.; Gao, Y.; Mao, X.; and Yin, J. 2021. Generative semi-supervised learning for multivariate time series imputation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 8983–8991.

Milo, R.; Shen-Orr, S.; Itzkovitz, S.; Kashtan, N.; Chklovskii, D.; and Alon, U. 2002. Network motifs: simple building blocks of complex networks. *Science*, 298(5594): 824–827.

Morris, C.; Ritzert, M.; Fey, M.; Hamilton, W. L.; Lenssen, J. E.; Rattan, G.; and Grohe, M. 2019. Weisfeiler and leman go neural: Higher-order graph neural networks. In *Pro-*

ceedings of the AAAI conference on artificial intelligence, volume 33, 4602–4609.

Rossi, R. A.; Ahmed, N. K.; and Koh, E. 2018. Higher-order network representation learning. In *Companion Proceedings of the The Web Conference 2018*, 3–4.

Song, S.; and Chen, L. 2011. Differential dependencies: Reasoning and discovery. *ACM Transactions on Database Systems (TODS)*, 36(3): 1–41.

Song, S.; Chen, L.; and Cheng, H. 2013. Efficient determination of distance thresholds for differential dependencies. *IEEE Transactions on knowledge and data engineering*, 26(9): 2179–2192.

Sun, L.; Huang, Z.; Wang, Z.; Wang, F.; Peng, H.; and Philip, S. Y. 2024. Motif-aware riemannian graph neural network with generative-contrastive learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 9044–9052.

Sun, Y.; Song, S.; Wang, C.; and Wang, J. 2020. Swapping repair for misplaced attribute values. In *2020 IEEE 36th International Conference on Data Engineering (ICDE)*, 721–732. IEEE.

Wang, B.; Cheng, L.; Sheng, J.; Hou, Z.; and Chang, Y. 2022. Graph convolutional networks fusing motif-structure information. *Scientific Reports*, 12(1): 10735.

Wang, D.; Yan, Y.; Qiu, R.; Zhu, Y.; Guan, K.; Margenot, A.; and Tong, H. 2023a. Networked time series imputation via position-aware graph enhanced variational autoencoders. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2256–2268.

Wang, D.; Zhang, Z.; Zhao, Y.; Huang, K.; Kang, Y.; and Zhou, J. 2023b. Financial Default Prediction via Motif-preserving Graph Neural Network with Curriculum Learning. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2233–2242.

Wothke, W. 2000. *Longitudinal and multigroup modeling with missing data*. Psychology Press.

Wu, H.; Hu, T.; Liu, Y.; Zhou, H.; Wang, J.; and Long, M. 2023. TimesNet: Temporal 2D-Variation Modeling for General Time Series Analysis. In *The Eleventh International Conference on Learning Representations*.

Yang, Y.; Guan, Z.; Zhao, W.; Lu, W.; and Zong, B. 2022. Graph substructure assembling network with soft sequence and context attention. *IEEE Transactions on Knowledge and Data Engineering*, 35(5): 4894–4907.

Yi, X.; Zheng, Y.; Zhang, J.; and Li, T. 2016. ST-MVL: filling missing values in geo-sensory time series data. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence*.

Yoon, J.; Jordon, J.; and Schaar, M. 2018. Gain: Missing data imputation using generative adversarial nets. In *International conference on machine learning*, 5689–5698. PMLR.

Zheng, Y.; Yi, X.; Li, M.; Li, R.; Shan, Z.; Chang, E.; and Li, T. 2015. Forecasting fine-grained air quality based on big data. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, 2267–2276.

Zhu, Y.; Jiang, B.; Jin, H.; Zhang, M.; Gao, F.; Huang, J.; Lin, T.; and Wang, X. 2021. Networked Time Series Prediction with Incomplete Data via Generative Adversarial Network. *ACM Transactions on Knowledge Discovery from Data*.