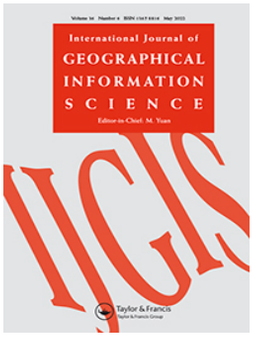

International Journal of Geographical Information Science

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/tgis20

A multi-view bidirectional spatiotemporal graph network for urban traffic flow imputation

Peixiao Wang, Tong Zhang, Yueming Zheng & Tao Hu

To cite this article: Peixiao Wang, Tong Zhang, Yueming Zheng & Tao Hu (2022) A multi-view bidirectional spatiotemporal graph network for urban traffic flow imputation, International Journal of Geographical Information Science, 36:6, 1231-1257, DOI: 10.1080/13658816.2022.2032081

To link to this article: https://doi.org/10.1080/13658816.2022.2032081

Published online: 28 Feb 2022.

INTERNATIONAL JOURNAL OF GEOGRAPHICAL INFORMATION SCIENCE
2022, VOL. 36, NO. 6, 1231–1257
https://doi.org/10.1080/13658816.2022.2032081

Taylor & Francis
Taylor & Francis Group

RESEARCH ARTICLE

A multi-view bidirectional spatiotemporal graph network for urban traffic flow imputation

Peixiao Wang[a], Tong Zhang[a], Yueming Zheng[b] and Tao Hu[c]

[a]State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan, China; [b]Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing, China; [c]Department of Geography, Oklahoma State University, Stillwater, OK, USA

ABSTRACT

Accurate estimation of missing traffic data is one of the essential components in intelligent transportation systems (ITS). The non-Euclidean data structure and complex missing traffic flow patterns make it challenging to capture nonlinear spatiotemporal correlations of missing traffic flow, which are critical for the imputation of missing traffic data. In this study, we propose a novel multi-view bidirectional spatiotemporal graph network called Multi-BiSTGN to impute urban traffic data with complex missing patterns. First, three spatiotemporal graph sequences are constructed to comprehensively describe traffic conditions from different temporal correlation views, i.e. temporal closeness view, daily periodicity view, and weekly periodicity view. Then, three bidirectional spatiotemporal graph networks are fused by a parametric-matrix-based method to obtain the final imputation results. To train the Multi-BiSTGN model, a novel loss function that considers the interactions between three temporal correlation views is designed to optimize the parameters of the Multi-BiSTGN model. The proposed model was validated on real-world traffic datasets collected in Wuhan, China. Experimental results showed that Multi-BiSTGN outperformed ten existing baselines under different missing types (random missing, block missing, and mixed missing) and missing rates.

ARTICLE HISTORY
Received 7 October 2021
Accepted 17 January 2022

KEYWORDS
Graph neural networks; missing data imputation; multi-view learning; spatiotemporal correlations

1. Introduction

Accurately monitoring traffic conditions of urban road networks is an indispensable part of an intelligent transportation system (ITS), and is of great significance to alleviate traffic congestion and pollutant emission (Ermagun and Levinson 2018, Jia and Yan 2021, S. Zhang *et al.* 2021). With the rapid development of the Internet of Things (IoT), many fixed-position sensors have been installed on urban roads to continuously monitor traffic conditions (Zhan *et al.* 2017, Deng *et al.* 2018, Hara *et al.* 2018). However, due to technical problems in data collection, data missing is common, which seriously restricts the performance of traffic flow modeling and traffic management

CONTACT Tong Zhang zhangt@whu.edu.cn

practices (Cheng *et al.* 2020, Tang *et al.* 2021, D. Xu *et al.* 2020). Therefore, how to estimate missing data (aka., traffic data imputation), is one of the essential preparatory works for traffic flow modeling (Cheng *et al.* 2018, Yang *et al.* 2021, Yi *et al.* 2021).

At present, the existing imputation methods not only face the challenge of modeling nonlinear spatiotemporal correlations of traffic flow, but also the challenge of complex missing patterns, such as random missing, block missing, and mixed missing (Li *et al.* 2019). To capture nonlinear spatiotemporal correlations, data-driven models have gradually become the mainstream methods of missing data imputation due to their excellent performance and generalization capabilities (Cheng *et al.* 2020, L. Li *et al.* 2020, Tang *et al.* 2021). To deal with complex missing patterns, multi-view learning has been integrated into traffic flow modeling (Cheng *et al.* 2019). For example, some tensor factorization models explore the spatiotemporal correlations between closeness and daily periodicity views to improve the imputation accuracy (X. Chen *et al.* 2019, 2020, X. Chen and Sun 2021). Despite these efforts on missing traffic flow imputation, the imputation performance is still not satisfactory due to non-Euclidean traffic flow on urban road networks since vehicles can only travel along road networks, which manifest non-Euclidean geometry and topology structures. Existing imputation models designed for Euclidean data structure are difficult to obtain satisfactory performance on road networks, which can be intuitively abstracted as graphs (Cai *et al.* 2020). Fortunately, the rapid development of graph neural networks (GNNs) significantly facilitates non-Euclidean data modeling (Zhang *et al.*, 2020, Li *et al.*, 2021, Yi *et al.* 2021). At present, GNNs have demonstrated state-of-art performance in ITS, including traffic speed prediction (Zhao *et al.* 2020), ride-hailing demand prediction (Geng *et al.* 2019), and subway passenger flow prediction (Ye *et al.* 2020). Therefore, we adopt GNNs for missing traffic data modeling to further improve the performance of missing data imputation. There are two main types of GNNs, which are spatial-based and spectral-based methods (Wu *et al.* 2021, Zhou *et al.* 2021). In spectral-based GNNs, graph convolution operations are defined as specific filters from the perspective of graph signal processing, i.e. factorization of graph Laplacian matrix (Yu *et al.* 2018). Spatial-based GNNs define graph convolution operations from the perspective of neighborhood aggregation which is typically realized via the message-passing neural network (MPNN) paradigm (Li *et al.* 2017, Tailor *et al.* 2021). Compared with spectral-based GNNs, spatial-based GNNs do not depend on the Laplace matrix, and are more suitable for directed graph structures, such as road networks (Defferrard *et al.* 2017, Kipf & Welling 2017). However, there are still challenges to directly use spatial-based GNNs to improve the imputation accuracy of missing traffic flow. These challenges are as follows:

1. Traffic flow data, which are constrained by road networks, are subject to dynamic changes over time. If we model a road network as a graph, traffic conditions on the graph nodes change continuously. However, the vanilla GNNs are mainly used for static graphs, and they cannot easily adapt to dynamic graphs (Rossi *et al.* 2020).
2. Missing traffic flow data contains complex missing patterns, such as random missing, block missing, and mixed missing. Existing studies leverage prior traffic

pattern knowledge to improve the imputation performance, such as daily or weekly periodicity of traffic flows. However, how to make better use of this pattern information in GNNs is still an open problem.

Some research efforts have been made to cope with the above challenges. For example, the temporal graph convolutional network (T-GCN) (Zhao *et al.* 2020) and the spatiotemporal graph convolutional network (ST-GCN) (B. Yu *et al.* 2018) are used for dynamic graph modeling. However, T-GCN and ST-GCN models are designed based on spectral GNN and rely heavily on Laplace matrices, which are determined by network topologies, limiting the representative capabilities of learned embeddings given new graphs with unknown topological structures. Therefore, based on the spatial-based GNNs, we propose a novel multi-view bidirectional spatiotemporal graph network called Multi-BiSTGN to impute missing traffic flow of urban road networks. This study makes the following three contributions:

1. Based on the MPNN paradigm, we propose a novel bidirectional temporal graph network called BiSTGN by defining specific message, aggregation, and update functions that are well-suited for traffic flow imputation under dynamic traffic environments on directed road networks;
2. We integrate a multi-view learning scheme into the Multi-BiSTGN model to improve the imputation performance of traffic flow under complex missing patterns, i.e. temporal closeness view, daily periodicity view, and weekly periodicity view. Based on the multi-view learning scheme, a novel loss function that considers the interactions between multiple views is designed to optimize the parameters of the Multi-BiSTGN model.
3. The performance of the Multi-BiSTGN model was evaluated using two real-world traffic flow data, demonstrating the advantages of our model compared with ten baseline methods.

2. Related works

Existing missing traffic data imputation methods can be roughly divided into two categories: statistical methods and data-driven methods.

2.1. Spatiotemporal traffic data imputation based on statistical methods

Statistical methods assume that the distributions of missing data are constrained by specific a prior knowledge in spatiotemporal dimensions, and establish specific parametric models to describe the traffic patterns of missing data. For example, the inverse distance weighting (IDW) model (Bartier and Keller 1996) is based on the first law of geography (Tobler 1970), and imputes missing values by calculating the distance between missing data and observed data. Kriging interpolation methods assume that the spatial distribution of observation data satisfies the second-order stability, and use covariance functions to obtain the optimal linear unbiased estimation of missing data (Pesquer *et al.* 2011). Based on the time stability assumption, autoregressive integrated

moving average (ARIMA) (Yozgatligil *et al.* 2013), seasonal ARIMA (SARIMA) (Williams and Hoel 2003), and simple exponential smoothing (SES) are used to infer the missing data based on the observations in previous time windows (Gardner 2006). In addition, based on the above models, some scholars proposed spatiotemporal statistical models based on the characteristics of missing data, such as spatiotemporal IDW (ST-IDW) (L. Li *et al.* 2014), Spatiotemporal ARIMA (ST-ARIMA) (Peibo Duan *et al.*, 2016), Spatiotemporal Kriging (ST-Kriging) (Aryaputera *et al.* 2015), and P-BSHADE (Hu *et al.* 2013, C.-D. Xu *et al.* 2013). Although statistical methods have been widely used in missing data modeling, imputation accuracy is still unsatisfactory. On the one hand, statistical methods are based on strict a priori assumptions, and the actual traffic environments are overly too complex to meet the pre-conditions of the models; on the other hand, traffic data has complex nonlinear spatiotemporal patterns, which are difficult to describe with specific parametric models (Cheng and Lu 2017, L. Li *et al.* 2020).

2.2. Spatiotemporal traffic data imputation based on data-driven methods

The data-driven methods do not require the data distributions to obey specific prior knowledge. They can automatically mine the spatiotemporal patterns from observed data (Li *et al.* 2014). For example, Wu *et al.* (2014) and Yu *et al.* (2016) applied the spatiotemporal k-nearest neighbor (ST-KNN) algorithm to traffic flow data modeling. Chang and Ge (2011) applied a variety of traditional matrix factorization models to impute missing traffic flow. The matrix factorization models consider temporal closeness constraints, such as time regularized matrix factorization (TRMF) and Bayesian temporal matrix factorization (BTMF) (Yu *et al.* 2016). Over the past few years, many deep learning algorithms have been used to impute missing traffic data, and achieved good performance. For instance, Cheng *et al.* (2020) used extreme learning machine (ELM) to integrate IDW and SES algorithms, and proposed a lightweight ensemble spatiotemporal interpolation model (ST-ISE). Li *et al.* (2017) combined deep neural network and the P-BSHADE algorithm to develop a hybrid spatiotemporal two-step missing data reconstruction framework (ST-2SMR). To deal with complex missing patterns, a series of tensor decomposition methods were developed to mine missing patterns from multiple views, such as Bayesian temporal tensor factorization (BTTF) (Chen and Sun 2021), Bayesian Gaussian CANDECOMP/PARAFAC (BGCP) (Chen *et al.* 2019), and low-rank tensor completion with truncated nuclear norm (LRTC-TNN) (Chen *et al.* 2020). Compared with classical statistical methods, data-driven methods do not require prior knowledge and explicit mathematical equations, and tend to have better data imputation results. Although the existing data-driven models have achieved better imputation performance, there are still shortcomings. Most existing models are mainly designed for imputing data under Euclidean space, and their imputation performance is still limited in non-Euclidean space (Zhang *et al.*, 2020, 2021). In order to extract traffic flow patterns from traffic data with non-Euclidean geometries and topologies, a variety of GNN-based models were applied to traffic flow modeling, such as T-GCN (Zhao *et al.* 2020) and ST-GCN (B. Yu *et al.* 2018). However, the T-GCN and ST-GCN models are mainly designed for traffic flow forecasting, and are not used for missing traffic data imputation.

3. Preliminaries and definitions

3.1. Definition 1 (traffic road network)

A traffic road network can be abstracted as a directed graph structure $G =< S, E, A >$, where $S = \{s_i\}_{i=1}^{n}$ represents n nodes in the graph G; E represents the set of edges in G, i.e. the connection relationships of nodes. In this study, the connection relationship mainly refers to the topological relationship. For simplicity, the topological relationships in the road network can be represented by an adjacency matrix $A \in \mathcal{R}^{n \times n}$. If there is an edge that starts from a node s_i to node s_j, then $A_{i,j}$=1, otherwise $A_{i,j}$=0. As the traffic road network is a directed graph, $A_{i,j}$ is not necessarily equal to $A_{j,i}$.

3.2. Definition 2 (traffic state)

Traffic state represents the quantitative traffic features of the nodes within a road network during a specific time window. As shown in Figure 1(a), the traffic state on the entire road network can be expressed as a tensor $\mathcal{X} \in \mathcal{R}^{n \times m \times d}$, where n represents the number of nodes on G, m represents the number of time windows (if the size of the time window is 15 min, there are 96 time windows in a day), d represents the

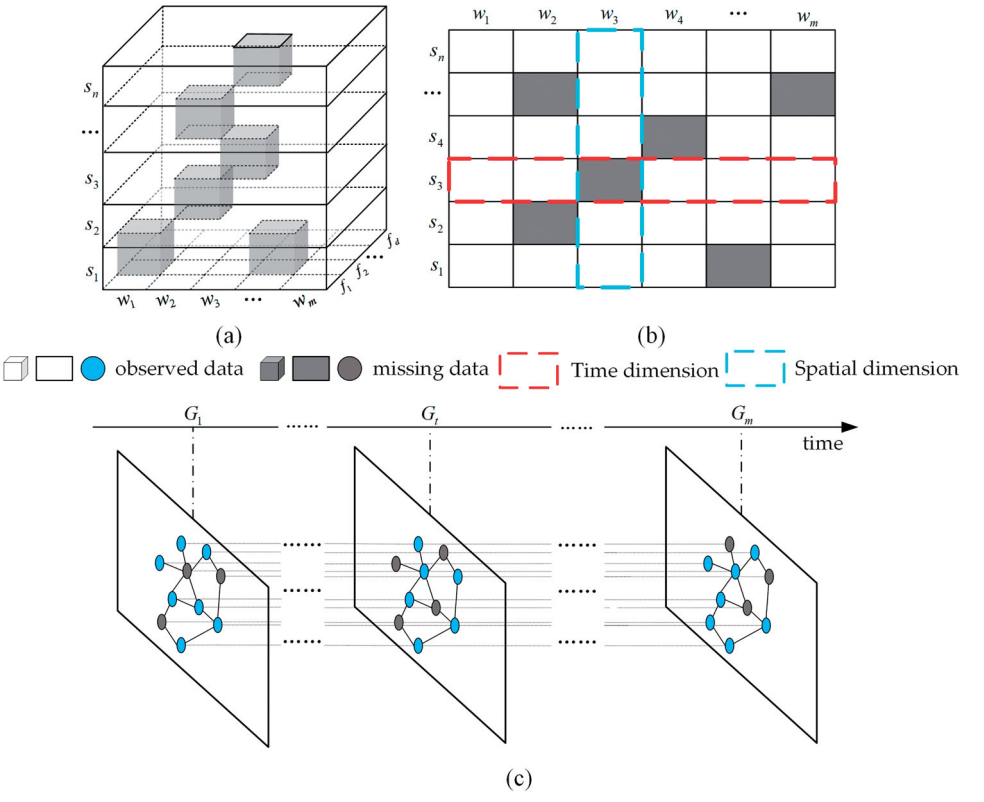

Figure 1. Preliminary definitions: s_i represents the ith node on the road network G, w_t represents the tth time window, and G_t represents the traffic state information of G within the time window w_t.

number of traffic states to be studied. As shown in Figure 1(b), the studied object of this work is a single traffic state on the traffic road network, i.e. traffic speed or traffic volume. Therefore, the tensor $\mathcal{X}$ can be further simply represented by the matrix $\boldsymbol{X} \in \mathcal{R}^{n\times m}$, i.e. d=1.

3.3. Definition 3 (spatiotemporal graph sequence)

A spatiotemporal graph sequence $SGS = \{G_t\}_{t=1}^{m}$ represents a series of temporally-order graphs, each of which represents ever-changing traffic state information on the road network over time. Specifically, the spatiotemporal graph sequence encodes dynamic traffic state on the road network, i.e. $G_t =< S, E, A, \boldsymbol{x}_t >$. As shown in Figure 1(c), G_t represents graph information in time window w_t, where $\boldsymbol{x}_t = \left\{x_t^i\right\}_{i=1}^{n}$ represents the traffic states of graph G_t, x_t^i represents the traffic state of node s_i in time window w_t.

In real-world applications, the traffic state information in a spatiotemporal graph sequence is often missing. This study aims to impute the missing information in a specific time window by using spatial-based graph neural networks. Taking a target graph $G_g =< S, E, A, \boldsymbol{x}_g >$ of the g th time window as an example, the imputation tasks can be formulated by Equations (1) and (2).

$$\begin{cases} \hat{G}_g = \mathcal{M} \longleftarrow \{\mathrm{G_t}\}_{\mathrm{t=1}}^{\mathrm{m}} \quad 1 \le g \le m \\ \hat{G}_g = <S, E, A, \hat{\boldsymbol{x}}_g> \end{cases} \tag{1}$$

$$\begin{cases} x_g^i = \phi \quad \forall i \in \boldsymbol{\Omega}_g \\ \hat{x}_g^i \neq \phi \quad \forall i \in \boldsymbol{\Omega}_g \end{cases} \tag{2}$$

where $\mathcal{M}$ represents the imputation model proposed in this work, i.e. Multi-BiSTGN model; $\{G_t\}_{t=1}^{m}$ represents a spatiotemporal graph sequence with missing data; $\hat{G}_g$ represents the graph of the target time window; $x_g^i = \phi$ indicates that the traffic state information of the node s_i in the time window w_g is missing; $\hat{\boldsymbol{x}}_g = \left\{x_g^i\right\}_{i=1}^{n}$ represents the traffic state vector after imputation; $\boldsymbol{\Omega}_g$ represents the index of the missing data in the graph G_g.

4. Methodology

In this section, we introduce the proposed Multi-BiSTGN model for missing traffic data imputation. The schematic structure of Multi-BiSTGN is presented in Figure 2. The Multi-BiSTGN model consists of three main components: a spatiotemporal graph data model, the construction of the BiSTGN, and the fusion of multi-view results, which are introduced in Sections 4.1–4.3, respectively. First, three spatiotemporal graph sequences are constructed to comprehensively describe the traffic states of the studied road network. Then, three BiSTGN models are developed to obtain the imputation results of three temporal correlation views by defining specific message, aggregation, and update functions based on the classical MPNN modeling paradigm (Li *et al.* 2017, Tailor *et al.* 2021). Finally, the imputation results from the three views are fused via a parametric matrix to obtain the final imputation result. In addition, the interactions

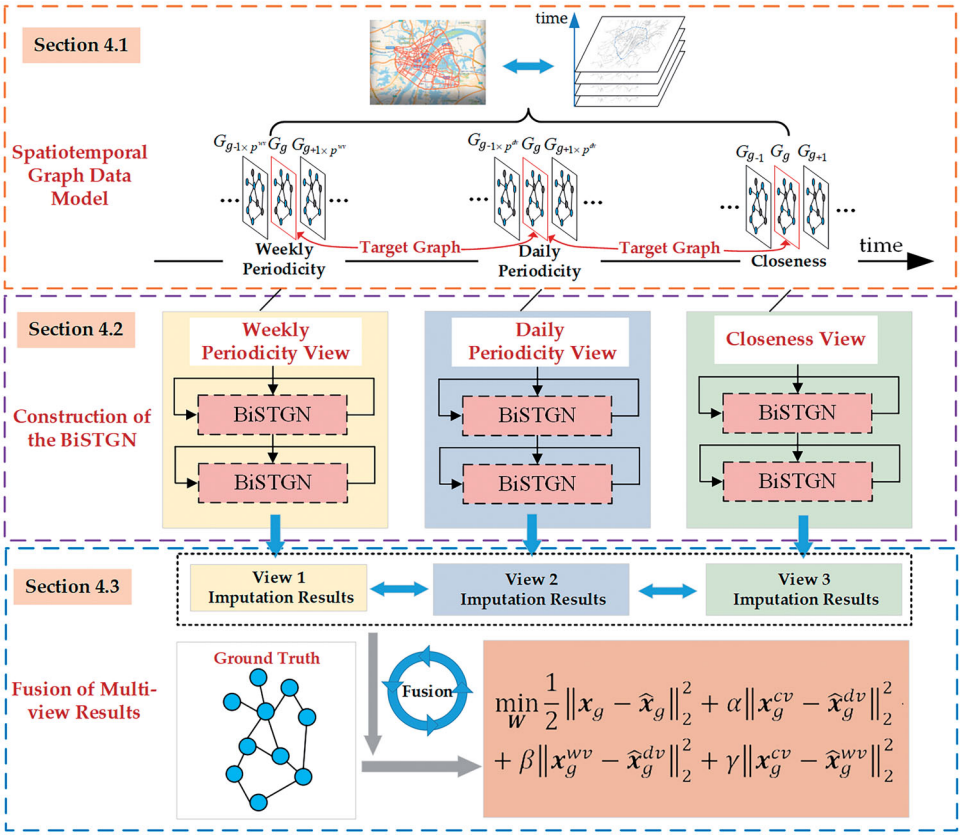

Figure 2. Overview of the Multi-BiSTGN model.

between multiple views are considered in the parameter optimization of the Multi-BiSTGN model, with the goal to improve the imputation performance of the Multi-BiSTGN model

4.1. Spatiotemporal graph data model

Traffic flow data manifest complex traffic patterns, making it difficult to directly model the traffic flow data. Therefore, multi-view learning can be used to decompose complex patterns into multiple separate simple patterns, which alleviates the burdens of describing complex nonlinear traffic patterns (Sun *et al.* 2020, Zhang *et al.*, 2020). The key of graph-based multi-view learning is to build a spatiotemporal graph data model. We reorganize the corresponding spatiotemporal traffic states from multiple views by multiple spatiotemporal graph sequences, describing the spatiotemporal correlations under different views.

Existing studies have shown that traffic flow data have apparent closeness and periodicity characteristics (Cheng *et al.* 2019). Among them, the closeness characteristics indicate that the traffic state of a specific road segment at a specific time is affected

by the traffic states of neighboring road segments and of previous time windows. The periodic characteristics indicate that the traffic patterns of a specific road segment usually repeat on a daily or weekly basis. Therefore, we construct spatiotemporal graph sequences from three views: temporal closeness, daily periodicity, and weekly periodicity. Furthermore, to capture the impacts of contextual information on missing data, spatiotemporal graph sequences are constructed from two directions along time (*forward* or *backward*). Taking the imputation of the missing data in the graph G_g as an example, its corresponding spatiotemporal closeness graph sequence SGS_g^{cv}, spatiotemporal daily periodic periodicity sequence SGS_g^{dv}, and spatiotemporal weekly periodicity graph sequence SGS_g^{wv} are defined in Equations (3) (4), and (5).

$$SGS_g^{cv} = \{G_{g+t}\}_{t=-l^{cv}}^{l^{cv}} = \{G_{g-l^{cv}}, G_{g-(l^{cv}-1)}, \ldots, G_{g+(l^{cv}-1)}, G_{g+l^{cv}}\} \tag{3}$$

$$SGS_g^{dv} = \{G_{g+t*p^{dv}}\}_{t=-l^{dv}}^{l^{dv}} = \{G_{g-l^{dv}*p^{dv}}, G_{g-(l^{dv}-1)*p^{dv}}, \ldots, G_{g+(l^{dv}-1)*p^{dv}}, G_{g+l^{dv}*p^{dv}}\} \tag{4}$$

$$SGS_g^{wv} = \{G_{g+t*p^{wv}}\}_{t=-l^{wv}}^{l^{wv}} = \{G_{g-l^{wv}*p^{wv}}, G_{g-(l^{wv}-1)*p^{wv}}, \ldots, G_{g+(l^{wv}-1)*p^{wv}}, G_{g+l^{wv}*p^{wv}}\} \tag{5}$$

where G_g represents the target graph to be imputed; l^{cv}, l^{dv} and l^{wv} respectively represent the time-dependent steps of the clossness view, the daily periodicity view, and weekly periodicity view. As we capture the spatiotemporal dependence of missing data from two directions along time, the target graph G_g is located in the middle of the spatiotemporal graph sequence. p^{dv} represents the number of time windows between two adjacent days; p^{wv} represents the number of time windows between two adjacent weeks. If the time window is 15 minutes, p^{dv} is equal to 96 (1440/15) and p^{wv} is equal to 672 (1440/15 $\times$ 7).

4.2. Construction of the BiSTGN

A spatiotemporal graph sequence under a particular view contains spatiotemporal constraints in two directions along time (*forward* and *backward*). Compared with static graphs, the spatiotemporal graph sequence contains more complex spatiotemporal correlations, which pose a greater challenge for GNNs to mine the traffic patterns from the spatiotemporal graph sequence. As the traditional GNN models cannot be directly used to model spatiotemporal graph sequences, a novel bidirectional spatiotemporal graph network called BiSTGN is proposed to capture spatiotemporal patterns in spatiotemporal graph sequences.

The BiSTGN is a spatial-based graph neural network, which follows the MPNN paradigm to define message propagation and update operations under time constraints from two directions. Inspired by RNN that models time series data (Schmidhuber 1992), we define two memory states for each node in the graph in order to apply the GNN models to the spatiotemporal graph sequences. As shown in Figure 3, based on the memory state, the basic idea of the BiSTGN model can be simply described in three steps. First, along the *forward* direction of the time axis, the memory state $\boldsymbol{H}_{g:f}$ of the g th time window is obtained from $\boldsymbol{H}_{g-1:f}$ and G_{g-1}. Then, along the *backward* direction of the time axis, the memory state $\boldsymbol{H}_{g:b}$ of the g th time window is obtained from $\boldsymbol{H}_{g+1:b}$ and G_{g+1}. Finally, $\boldsymbol{H}_{g:b}$ and $\boldsymbol{H}_{g:f}$ are integrated to obtain the imputation results for a specific view. Taking imputing the missing data in the target graph G_g as

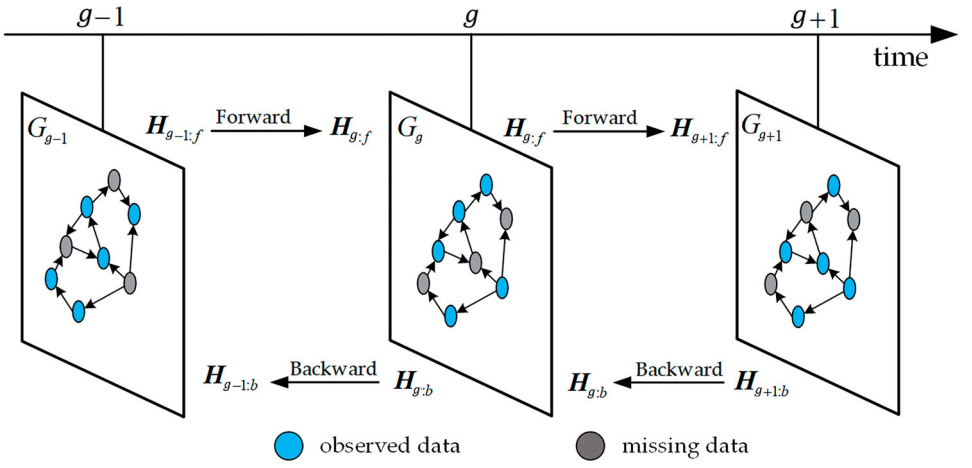

Figure 3. Illustration of memory state in BiSTGN: $\boldsymbol{H}_{g:f} = \left\{\boldsymbol{h}^i_{g:f}\right\}_{i=1}^{n}$ and $\boldsymbol{H}_{g:b} = \left\{\boldsymbol{h}^i_{g:b}\right\}_{i=1}^{n}$ respectively memorize the traffic states of the target time window (g th) at the historical time and the future time, i.e. the compressed representations of the traffic states at the historical time or the future time.

an example, the imputation value of node s_i in g th time window can be defined by Equations (6) (7) and (8).

$$forward: \begin{cases} \boldsymbol{m}^{j\to i}_{g:f} = \text{message}_{j\in N^f_i}\left(\boldsymbol{h}^j_{g-1:f}, x^j_{g-1}\right) \\ \boldsymbol{a}^i_{g:f} = \text{aggregate}\left(\left\{\boldsymbol{m}^{j\to i}_{g:f}\right\}_{j\in N^f_i}\right) \\ \boldsymbol{h}^i_{g:f} = update\left(\boldsymbol{a}^i_{g:f}\right) \end{cases} \tag{6}$$

$$backward: \begin{cases} \boldsymbol{m}^{j\to i}_{g:b} = \text{message}_{j\in N^b_i}\left(\boldsymbol{h}^j_{g+1:b}, x^j_{g+1}\right) \\ \boldsymbol{a}^i_{g:b} = \text{aggregate}\left(\left\{\boldsymbol{m}^{j\to i}_{g:b}\right\}_{j\in N^b_i}\right) \\ \boldsymbol{h}^i_{g:b} = update\left(\boldsymbol{a}^i_{g:b}\right) \end{cases} \tag{7}$$

$$\hat{x}^i_g = \left[\boldsymbol{h}^i_{g:f} || \boldsymbol{h}^i_{g:b}\right] \boldsymbol{W}^{out} \tag{8}$$

where $\hat{x}^i_g \in \mathcal{R}^{1\times 1}$ represents the imputation value of the node s_i in the gth time window; $x^j_{g-1} \in \mathcal{R}^{1\times 1}$ represents the observation value of the node s_j in the $(g-1)$th time window; $x^j_{g+1} \in \mathcal{R}^{1\times 1}$ represents the observation value of the node s_j in the $(g+1)$th time window; the message, aggregate, and update represent the redefined functions in MPNN paradigm, respectively; $[\cdot||\cdot]$ denotes the vector connection function; N^f_i and N^b_i denote the collection of spatiotemporal neighbors in two time directions of the node s_i, respectively; $\boldsymbol{m}^{j\to i}_{g:f}$ and $\boldsymbol{m}^{j\to i}_{g:b} \in \mathcal{R}^{1\times d}$ denote the message that the node s_j transmits to the node s_i in two time directions, respectively; $\boldsymbol{a}^i_{g:b}$ and $\boldsymbol{a}^i_{g:f}$ represent the compressed representation of multiple messages received by the node s_i in two time directions; $\boldsymbol{h}^i_{g:f}$ and $\boldsymbol{h}^i_{g:b} \in \mathcal{R}^{1\times d}$ denote the two memory states of the node s_i in the gth time window, and d represents the dimension of the memory state; $\boldsymbol{W}^{out} \in$

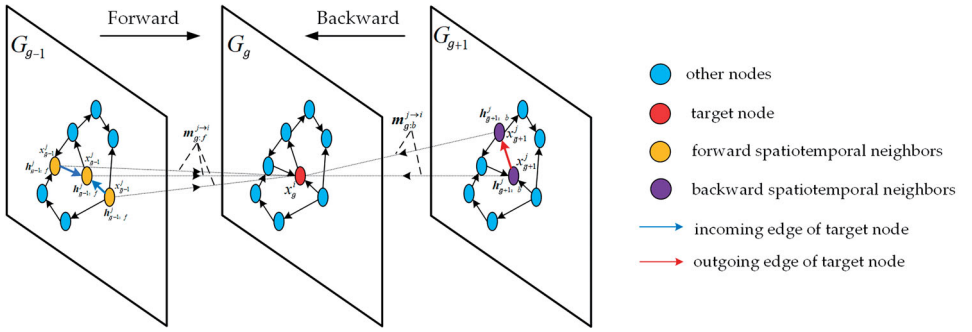

Figure 4. Illustration of messaging passing in BiSTGN: Along the *forward* direction, the target node s_i receives the traffic state $x^j_{g-1} \in \mathcal{R}^{1\times 1}$ of the spatiotemporal neighbors in the previous time window, and the memory state $\boldsymbol{h}^j_{g-1:f} \in \mathcal{R}^{1\times d}$ of the neighbors in the historical time window. Along the *backward* direction, the target node s_i receives the traffic state $x^j_{g+1} \in \mathcal{R}^{1\times 1}$ of the spatiotemporal neighbors in the next time window, and the memory state $\boldsymbol{h}^j_{g+1:b} \in \mathcal{R}^{1\times d}$ of the neighbors in the next time window.

$\mathcal{R}^{2d\times 1}$ represents a learnable conversion matrix, which converts two memory states into imputation values. According to Equations (6)–(8), the key of BiSTGN is to define the message, aggregation and update functions in the model.

4.2.1 Message function

The message function of the BiSTGN model is used to define the message that the target node s_i receives in the g th time window. Figure 4 shows the BiSTGN messaging process in both directions. In the BiSTGN model, a specific node not only receives messages from its own node (temporal neighbors), but also receives messages from surrounding nodes (spatial neighbors). As the traffic road network is a directed graph, the spatiotemporal neighbors in the *forward* direction only include nodes connected by the incoming edge, and the spatiotemporal neighbors in the *backward* direction only include nodes connected by the outgoing edge.

Since each spatiotemporal neighbor transmits two kinds of messages with different dimensions, the message function in the BiSTGN needs to merge the two types of messages. Inspired by the GGNN model (Li *et al.* 2017), we integrate the GRU model into the BiSTGN to capture the long-term dependence in the spatiotemporal graph sequence. In the BiSTGN model, the message function of the node s_i in the g th window is defined by Equations (9) and (10).

$$
forward: \begin{cases} \boldsymbol{r}^j_{g:f} = \sigma\left(\boldsymbol{W}^r_f[\boldsymbol{h}^j_{g-1:f}||x^j_{g-1}]\right) \\ \boldsymbol{z}^j_{g:f} = \sigma\left(\boldsymbol{W}^z_f[\boldsymbol{h}^j_{g-1:f}||x^j_{g-1}]\right) \\ \tilde{\boldsymbol{m}}^{j\to i}_{g:f} = \tanh\left(\boldsymbol{W}^{\tilde{m}}_f[\boldsymbol{r}^j_{g:f} * \boldsymbol{h}^j_{g-1:f}||x^j_{g-1}]\right) \\ \boldsymbol{m}^{j\to i}_{g:f} = \left(1-\boldsymbol{z}^j_{g:f}\right) * \boldsymbol{h}^j_{g-1:f} + \boldsymbol{z}^j_{g:f} * \tilde{\boldsymbol{m}}^{j\to i}_{g:f} \end{cases} \tag{9}
$$

$$backward: \begin{cases} \boldsymbol{r}^j_{g:b} = \sigma\left(\boldsymbol{W}^r_b[\boldsymbol{h}^j_{g+1:b}||x^j_{g+1}]\right) \\ \boldsymbol{z}^j_{g:b} = \sigma\left(\boldsymbol{W}^z_f[\boldsymbol{h}^j_{g+1:b}||x^j_{g+1}]\right) \\ \tilde{\boldsymbol{m}}^{j\to i}_{g:b} = \tanh\left(\boldsymbol{W}^{\tilde{m}}_b[\boldsymbol{r}^j_{g:b} * \boldsymbol{h}^j_{g+1:b}||x^j_{g+1}]\right) \\ \boldsymbol{m}^{j\to i}_{g:b} = \left(1-\boldsymbol{z}^j_{g:b}\right) * \boldsymbol{h}^j_{g+1:b} + \boldsymbol{z}^j_{g:b} * \tilde{\boldsymbol{m}}^{j\to i}_{g:b} \end{cases} \tag{10}$$

where $\boldsymbol{m}^{j\to i}_{g:f} \in \mathcal{R}^{1\times d}$ denotes the message sent by the node s_j to the node s_i along the *forward* direction, i.e. the fused message; $\boldsymbol{m}^{j\to i}_{g:b} \in \mathcal{R}^{1\times d}$ denotes the message sent by the node s_j to the node s_i along the *backward* direction; x^j_{g-1} and x^j_{g+1} denote the observation data of neighbor nodes in the previous and next time window, respectively; $\boldsymbol{h}^j_{g-1:f}$ and $\boldsymbol{h}^j_{g+1:b}$ denote the memory state of neighbor nodes in historical and future time windows, respectively; $\boldsymbol{W}^r_f$, $\boldsymbol{W}^z_f$, $\boldsymbol{W}^{\tilde{m}}_f$, $\boldsymbol{W}^r_b$, $\boldsymbol{W}^z_f$, and $\boldsymbol{W}^{\tilde{m}}_b \in \mathcal{R}^{d\times(d+1)}$ denote the learnable parameter matrices; $\sigma(x)$ denotes the sigmoid activation function; $[\cdot||\cdot]$ denotes the vector connection function; $\tanh(x)$ denotes the hyperbolic tangent activation function.

4.2.2. Aggregation function

The aggregation function of the BiSTGN model is used to aggregate the messages received by the target node s_i in the g th time window, i.e. aggregate $\left\{\boldsymbol{m}^{j\to i}_{g:f}\right\}_{j\in N^f_i}$ and $\left\{\boldsymbol{m}^{j\to i}_{g:b}\right\}_{j\in N^b_i}$. The most common aggregation function is the mean function. However, the mean function assigns equal weights to the messages of different nodes and ignores the differences between nodes (Hechtlinger *et al.* 2017, Chen *et al.* 2020). In real-world scenarios, traffic states of different roads, such as spatial neighbors and temporal neighbors, should have different effects on the traffic states of the examined node. Therefore, inspired by the graph attention mechanism (Veličković *et al.* 2018), we aggregate the messages from spatial and temporal neighbors via a weighted scheme. The aggregation function of the target node s_i in the g th time window is shown in Equations (11) and (12).

$$forward: \begin{cases} \boldsymbol{a}^i_{g:f} = [\boldsymbol{a}^{i:t}_{g:f}||\boldsymbol{a}^{i:s}_{g:f}] \\ \boldsymbol{a}^{i:t}_{g:f} = \boldsymbol{m}^{i\to i}_{g:f} \\ \boldsymbol{a}^{i:s}_{g:f} = \sum_{j\in N^f_i, j!=i} \lambda^{j\to i}_f \boldsymbol{m}^{j\to i}_{g:f} \\ \lambda^{j\to i}_f = \dfrac{\exp\left(\text{LeakyReLU}\left([\boldsymbol{m}^{j\to i}_{g:f}||\boldsymbol{m}^{i\to i}_{g:f}]\boldsymbol{W}^e_f\right)\right)}{\sum_{k\in N^f_i, k!=i}\exp\left(\text{LeakyReLU}\left([\boldsymbol{m}^{k\to i}_{g:f}||\boldsymbol{m}^{i\to i}_{g:f}]\boldsymbol{W}^e_f\right)\right)} \end{cases} \tag{11}$$

$$backward: \begin{cases} \boldsymbol{a}^i_{g:b} = [\boldsymbol{a}^{i:t}_{g:b}||\boldsymbol{a}^{i:s}_{g:b}] \\ \boldsymbol{a}^{i:t}_{g:b} = \boldsymbol{m}^{i\to i}_{g:b} \\ \boldsymbol{a}^{i:s}_{g:b} = \sum_{j\in N^b_i, j!=i} \lambda^{j\to i}_b \boldsymbol{m}^{j\to i}_{g:b} \\ \lambda^{j\to i}_b = \dfrac{\exp\left(\text{LeakyReLU}\left([\boldsymbol{m}^{j\to i}_{g:b}||\boldsymbol{m}^{i\to i}_{g:b}]\boldsymbol{W}^e_b\right)\right)}{\sum_{k\in N^b_i, k!=i}\exp\left(\text{LeakyReLU}\left([\boldsymbol{m}^{k\to i}_{g:b}||\boldsymbol{m}^{i\to i}_{g:b}]\boldsymbol{W}^e_b\right)\right)} \end{cases} \tag{12}$$

where $\boldsymbol{a}_{g:f}^{i:t}$ and $\boldsymbol{a}_{g:f}^{i:s}$ denote the aggregation values of the spatiotemporal neighbors in the *forward* direction, respectively; $\boldsymbol{a}_{g:b}^{i:t}$ and $\boldsymbol{a}_{g:b}^{i:s}$ denote the aggregation values of the spatiotemporal neighbors in the *backward* direction, respectively; N_i^f and N_i^b denote the neighbor collection of the target node s_i in two directions, respectively; $\boldsymbol{m}_{g:f}^{j\to i}$ and $\boldsymbol{m}_{g:b}^{j\to i} \in \mathcal{R}^{1\times d}$ denote messages sent by neighbor node s_j to target node s_i in two directions, respectively; $\lambda_f^{j\to i}$ and $\lambda_b^{j\to i}$ denote the message weight coefficient; $\boldsymbol{W}_f^e$ and $\boldsymbol{W}_b^e \in \mathcal{R}^{2d\times 1}$ are the conversion matrices that are used to obtain the weight coefficients; $\exp(x)$ denotes the exponential function; $[\cdot||\cdot]$ denotes the vector connection function; $\mathrm{LeakyReLU}(x)$ denotes the nonlinear activation function.

4.2.3. Update function

The update function of the BiSTGN model is used to obtain the memory state $\boldsymbol{h}_g^i \in \mathcal{R}^{1\times d}$ of the target node s_i in a specific time window. Take the g th time window as an example, along the *forward* direction, the update function is used to update the aggregated value $\boldsymbol{a}_{g:f}^i \in \mathcal{R}^{1\times 2d}$ to the memory state $\boldsymbol{h}_{g:f}^i \in \mathcal{R}^{1\times d}$ of the current time window. Similarly, along the *backward* direction, the update function is used to update the aggregated value $\boldsymbol{a}_{g:b}^i \in \mathcal{R}^{1\times 2d}$ to the memory state $\boldsymbol{h}_{g:b}^i \in \mathcal{R}^{1\times d}$ of the current time window. For simplicity, in the BiSTGN model, the update function of node s_i in the g th window is shown in Equations (13) and (14).

$$\boldsymbol{h}_{g:f}^i = \boldsymbol{a}_{g:f}^i \boldsymbol{W}_f^c \tag{13}$$

$$\boldsymbol{h}_{g:b}^i = \boldsymbol{a}_{g:b}^i \boldsymbol{W}_b^c \tag{14}$$

where $\boldsymbol{W}_f^c$ and $\boldsymbol{W}_b^c \in \mathcal{R}^{2d\times d}$ represent the learnable transformation matrices. After transformation, along the *forward* direction, the memory state $\boldsymbol{h}_{g:f}^i \in \mathcal{R}^{1\times d}$ at the current time has the same dimension as the memory state $\boldsymbol{h}_{g-1:f}^i \in \mathcal{R}^{1\times d}$ at the previous time. Similarly, along the *backward* direction, the memory state $\boldsymbol{h}_{g:b}^i \in \mathcal{R}^{1\times d}$ at the current time has the same dimension as the memory state $\boldsymbol{h}_{g+1:b}^i \in \mathcal{R}^{1\times d}$ at the next time. Therefore, the BiSTGN model can iteratively learn the missing patterns in spatiotemporal graph sequences from two time directions.

4.2.4. Advantage analysis of BiSTGN

The RNN, GNN, and BiSTGN can all be regarded as a kind of MPNN. In this section, the advantages of BiSTGN are further analyzed from the perspective of MPNN. As shown in Figure 5, in the RNN model (GRU and LSTM models can be regarded as special RNNs), the node in the graph is regarded as an independent sample. The target node can only receive the messages transmitted by its historical nodes, ignoring the topological connection of the graph (Figure 5(a)). Compared with the RNN model, the GNN model considers the spatial connection of graph nodes and receives messages from surrounding nodes. However, the GNN model does not have the concept of time (Figure 5(b)). Compared with the RNN and GNN models, the BiSTGN model combines the advantages of the RNN and GNN models from message propagation. The target node receives the messages delivered by its historical node and receives the messages delivered by surrounding nodes (Figure 5(c)). Specifically, RNN and GNN can be

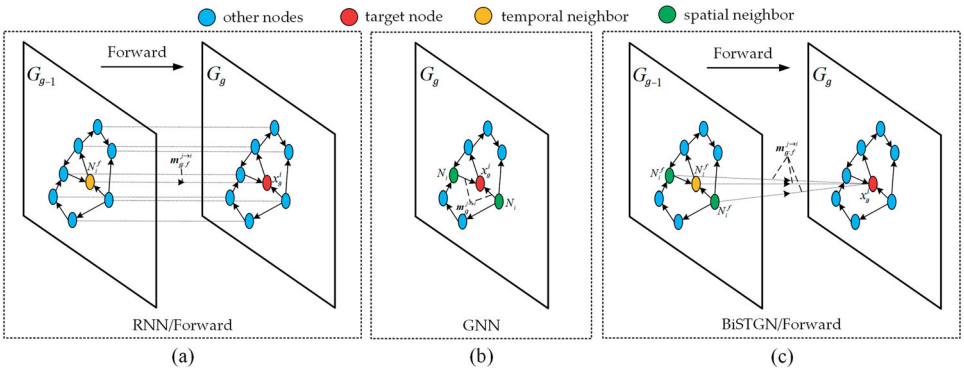

Figure 5. Difference between BiSTGN and GNN and RNN from the perspective of MPNN: $\boldsymbol{m}_{g:f}^{j\to i}$ and $\boldsymbol{m}_{g}^{j\to i} \in \mathcal{R}^{1\times d}$ represent the message received by the target node from the neighbor nodes.

regarded as a particular case of BiSTGN. When the nodes in the graph are only self-connected (the adjacency matrix is the identity matrix), the BiSTGN model degenerates to the RNN model. When there is only a single moment of graph information (the length of the spatiotemporal graph sequence is 1), the BiSTGN model degenerates to the GNN model.

4.3. Fusion of multi-view results

Three BiSTGN models are established to obtain imputation results from different views. The imputation results from multiple views are essentially different manifestations of multiple traffic patterns. The imputation results under each view have limited capabilities in capturing missing patterns of traffic flow. To improve the performance of missing data imputation, inspired by Li *et al.* (2021), we use the parametric-matrix-based fusion method to fuse the three views to obtain the final imputation result. Taking filling the missing data in the graph G_g as an example, the fusion process is shown in Equation (15).

$$\hat{\boldsymbol{x}}_g = \hat{\boldsymbol{x}}_g^{cv} \odot \boldsymbol{W}^{\mathrm{cv}} + \hat{\boldsymbol{x}}_{\mathrm{g}}^{\mathrm{dv}} \odot \boldsymbol{W}^{\mathrm{dv}} + \hat{\boldsymbol{x}}_{\mathrm{g}}^{\mathrm{wv}} \odot \boldsymbol{W}^{\mathrm{wv}} \tag{15}$$

where $\hat{\boldsymbol{x}}_g \in \mathcal{R}^{d\times 1}$ represents the final imputation results of the gth time window; $\hat{\boldsymbol{x}}_g^{cv}$, $\hat{\boldsymbol{x}}_g^{dv}$ and $\hat{\boldsymbol{x}}_g^{wv} \in \mathcal{R}^{d\times 1}$ represent the imputation results from the closeness view, daily periodicity view and weekly periodicity view, respectively; $\boldsymbol{W}^{cv}$, $\boldsymbol{W}^{dv}$ and $\boldsymbol{W}^{wv} \in \mathcal{R}^{d\times 1}$ represent the parameter fusion matrices of the closeness view, daily periodicity view and weekly periodicity view, respectively; $\odot$ represents Hadmard product.

Theoretically, the final model can be obtained by minimizing the square loss between the real traffic state $\boldsymbol{x}_g$ and the imputation traffic state $\hat{\boldsymbol{x}}_g$. However, only optimizing the square loss between $\boldsymbol{x}_g$ and $\hat{\boldsymbol{x}}_g$ ignores the interactions between views, i.e. the imputation result of a single view is independent from the results of other views. When the imputation result of single view deviates greatly from the true value, the fusion result may lead to inferior imputation performance. In fact, there is an implicit relationship between the imputation results of different views. For example, different views describe the internal characteristics of traffic states from different aspects, and the imputation results of each view should be as consistent as possible.

Therefore, the internal relationship is integrated into the optimization process of Multi-BiSTGN, and the overall loss function is shown in Equation (16).

$$\mathcal{L}(\boldsymbol{W}) = \min_{\boldsymbol{W}} \frac{1}{2} \| \boldsymbol{x}_g - \hat{\boldsymbol{x}}_g \|_2^2 + \alpha \| \hat{\boldsymbol{x}}_g^{cv} - \hat{\boldsymbol{x}}_g^{dv} \|_2^2 + \beta \| \hat{\boldsymbol{x}}_g^{wv} - \hat{\boldsymbol{x}}_g^{dv} \|_2^2 + \gamma \| \hat{\boldsymbol{x}}_g^{cv} - \hat{\boldsymbol{x}}_g^{wv} \|_2^2 \quad (16)$$

where $\boldsymbol{x}_g$ represents the true value of the gth time window; $\hat{\boldsymbol{x}}_g$ represents the imputation value of the gth time window; $\boldsymbol{W}$ represents the parameters that can be learned in the Multi-BiSTGN; $\hat{\boldsymbol{x}}_g^{cv}$, $\hat{\boldsymbol{x}}_g^{dv}$ and $\hat{\boldsymbol{x}}_g^{wv} \in \mathcal{R}^{d\times 1}$ represent the imputation results from the closeness view, daily periodicity view and weekly periodicity view, respectively; α, β and γ represent the regularization terms from the clossness view, daily periodicity view and weekly periodicity view respectively, i.e. the interactions of imputation results from different views.

4.4. Algorithms and optimization

The basic principle of the Multi-BiSTGN model is to establish a semi-supervised learning method that uses the observable data in the graph to estimate the missing values in the traffic network under time constraints. To train the Multi-BiSTGN model, the observed traffic flow data were divided into training samples and test samples. The training samples were used to train the parameters of model $\mathcal{M}$, and the test samples were used to test the imputation performance of model $\mathcal{M}$. The training process of $\mathcal{M}$ is shown in Algorithm 1, where $\boldsymbol{\Omega}_g$ records the index of training data in the g th time window. First, the spatiotemporal graph sequences under three views are constructed for the target graph G_g that needs to be imputed (line 3). Then, the training sample is composed of the index of the training data and the spatiotemporal graph sequences under the three views (line 4). Finally, the parameters of the Multi-BiSTGN model are obtained by minimizing the errors between the imputation values and the true values (lines 7–12). After the imputation model is trained, the imputation problem is converted into a prediction problem and the missing values are predicted iteratively.

Algorithm 1. Training Process of Multi-BiSTGN

Require: Graph sequence that needs to be imputed: $\{G_g\}$
Index sequence of training data: $\{\boldsymbol{\Omega}_g\}$
Graph sequence of all times: $SGS = \{G_t\}_{t=1}^{m}$ graph sequence
Time dependent step: I^{cv}, I^{dv}, I^{wv}
Regularization coefficients: α, β, γ
Ensure: Multi-BiSTGN model: $\mathcal{M}$
//construct training instances samples
1: $\mathcal{D} \leftarrow \emptyset$
2: for each $G_g \in \{G_g\}$ **do**
3: construct $\left\{SGS_g^{cv}, SGS_g^{dv}, SGS_g^{wv}\right\}$ with SGS, I^{cv}, I^{dv}, and I^{wv}
4: put a training instance $\left(\left\{SGS_g^{cv}, SGS_g^{dv}, SGS_g^{wv}\right\}, \boldsymbol{\Omega}_g\right)$ into $\mathcal{D}$
//train Multi-BiSTGN model
5: initialize the parameters $\boldsymbol{W}$
6: repeat
7: randomly select a training instance $\mathcal{D}_g$ from $\mathcal{D}$

8: obtain results $\hat{\boldsymbol{x}}_g^{cv}$ of the closeness view by Formulas (6), (7) and (8)
9: obtain results $\hat{\boldsymbol{x}}_g^{dv}$ of the daily periodicity view by Formulas (6), (7) and (8)
10: obtain results $\hat{\boldsymbol{x}}_g^{wv}$ of the weekly periodicity view by Formulas (6), (7) and (8)
11: obtain fusion results $\hat{\boldsymbol{x}}_g$ by Formulas (15)
12: find $\boldsymbol{W}$ by minimizing the Formula (16) with $\boldsymbol{\Omega}_g$, α, β and γ
13: until stopping criteria is met (the loss function is less than a specific threshold)
14: output the learned Multi-BiSTGN model $\mathcal{M}$

5. Experimental results and discussions

5.1. Data preparation

5.1.1. Data sources

Two types of datasets were used to evaluate the imputation performance of the Multi-BiSTGN model: Traffic speed and volume data collected in Wuhan, China.

The traffic speed data were obtained based on the taxi trajectory data and the traffic road network data. The time window size of the traffic speed data is 15 minutes. The time span of traffic speed data is from August 1, 2018, to August 28, 2018. In addition, due to the relatively small number of people traveling in the early morning, we only calculate the average speed from 6:00 to 24:00. Figure 6(a) shows the road network used in the experiment including a total of 82 road segments. As shown in Table 1, each record contains the unique identification ID of road segment, time window, the shape of the road segment (line elements composed of coordinate sequences), and average traffic speed within the time window.

The traffic volume data were obtained based on the automatic vehicle identification (AVI) data that were derived from camera videos. We counted the traffic volume of a single camera at an intervals of 5 minutes, i.e. the time window size is 5 minutes. The time span of traffic volume data is from March 1, 2021, to March 28, 2021. Similar to the traffic flow speed, we only calculated the traffic volume from 6:00 to 24:00. Figure 6(b) shows the spatial distribution of experimental cameras. A total of 10 experimental cameras were selected in the experiment. As shown in Table 2, each record contains

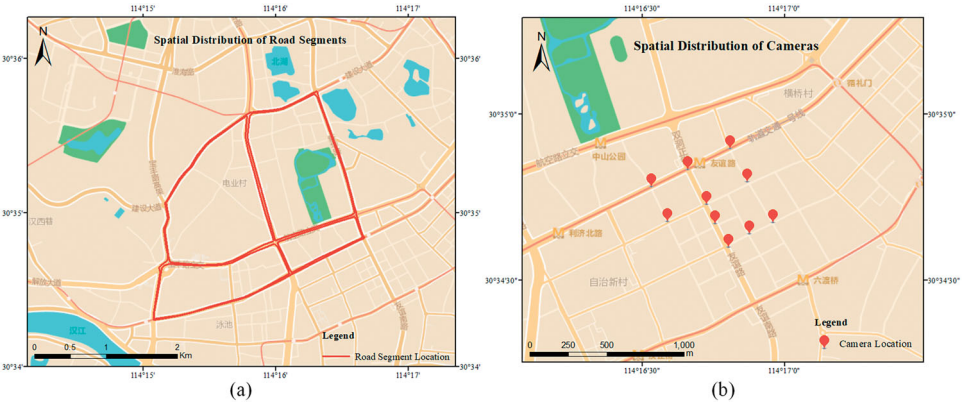

Figure 6. Sketch map of the study area.

Table 1. Traffic speed sample of single road segment.

Segment ID	Time window	Shape	Traffic speed
b5a6***	2021-08-01 06:00 ~ 2021-08-01 06:15	Geometry (Polyline)	33.1
b5a6***	2021-08-01 06:15 ~ 2021-08-01 06:30	Geometry (Polyline)	34.4
b5a6***	2021-08-01 06:30 ~ 2021-08-01 06:45	Geometry (Polyline)	34.7
… …	… …	… …	… …
b5a6***	2021-08-28 23:55 ~ 2021-08-29 00:00	Geometry (Polyline)	42.6

***Means the content is omitted.

Table 2. Traffic volume sample of single camera.

Camera ID	Time window	Latitude	Longitude	Traffic volume
9634CE***	2021-03-01 06:00 ~ 2021-03-01 06:05	30.6***	114.1***	41
9634CE***	2021-03-01 06:05 ~ 2021-03-01 06:10	30.6***	114.1***	41
9634CE***	2021-03-01 06:10 ~ 2021-03-01 06:15	30.6***	114.1***	54
… …	… …	… …	… …	… …
9634CE***	2021-03-28 23:55 ~ 2021-03-29 00:00	30.6***	114.1***	36

***Means the content is omitted.

the unique identification ID of the camera, time window, spatial position of the camera, and traffic volume in the time window.

5.1.2. Data preprocessing

We preprocessed traffic speed and traffic flow data with the following steps:

1. According to the connection relationship of the traffic network, the topological structures of the road segments and the cameras were constructed.
2. There are mainly three missing types in actual scenes, i.e. random missing, block missing, and mixed missing. Random missing means that data missing is random, which is generally caused by poor equipment signal. Block missing indicates that data missing is continuity, which is generally caused by equipment failure or power failure. Mixed missing means that the data contains not only block missing but also random missing. In this study, partial traffic states were deleted at 20% and 40% missing rates based on the three missing types, respectively. Figure 7 shows the temporal and spatial location of the artificially missing values (grey). The horizontal axis represents the time window of the missing value, and the vertical axis represents the unique identifier of the spatial location.
3. The constructed missing data in the last step was divided into training samples and test samples using an 80:20 split ratio.

5.2. Evaluation metrics

In this study, mean absolute error (MAE), root mean square error (RMSE), and mean absolute percentage error (MAPE) are used as quantitative indicators to verify the imputation accuracy of the Multi-BiSTGN. The MAE represents the average of the absolute error between the imputation value and the true value, the RMSE represents the sample standard deviation of the error between the imputation value and the true value, and the MAPE represents the average of the proportion of the error to the true

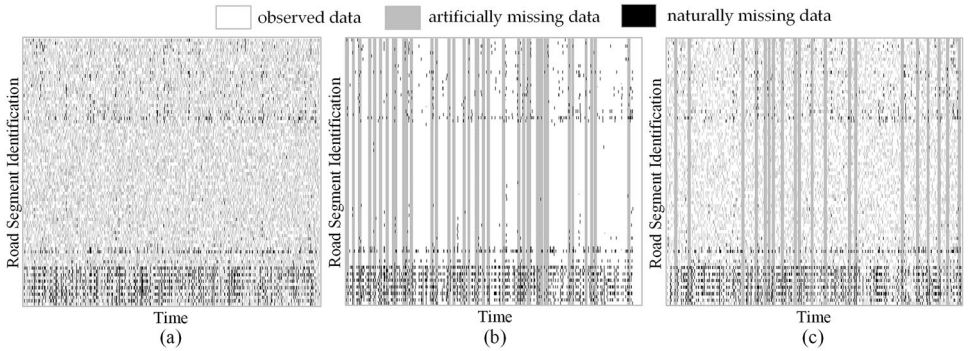

Figure 7. Traffic speed information after preprocessing: (a) random missing, (b) block missing, and (c) mixed missing.

value. The MAE, RMSE, and MAPE are calculated as shown in Equations (17) (18), and (19).

$$MAE = \frac{1}{N}\sum_{i\in\mathbf{\Omega}_g}\left|\hat{x}_g^i - x_g^i\right| \tag{17}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i\in\mathbf{\Omega}_g}\left(\hat{x}_g^i - x_g^i\right)^2} \tag{18}$$

$$MAPE = \frac{100\%}{N}\sum_{i\in\mathbf{\Omega}_g}\left|\frac{\hat{x}_g^i - x_g^i}{x_g^i}\right| \tag{19}$$

where $\mathbf{\Omega}_g$ represents the index of missing data in the target graph; N is the total number of missing data, i.e. $\left|\mathbf{\Omega}_g\right|$; x_g^i represents the actual traffic state of node s_i in time window w_g; $\hat{x}_g^i$ represents the traffic state imputed by the model of node s_i in time window w_g.

5.3. Hyper-parameter selection

The Hyper-parameters of Multi-BiSTGN model mainly include closeness dependence step l^{cv}, daily periodicity dependence step l^{dv}, weekly periodicity dependence step l^{wv}, coefficient α, coefficient β, and coefficient γ. There are two ways to calibrate these hyper-parameters of the Multi-BiSTGN model. One way is to treat the Multi-BiSTGN model as an end-to-end model, and calibrate the six hyper-parameters of the Multi-BiSTGN model simultaneously. Another way is to treat the Multi-BiSTGN model as a combination of multiple imputation components, and only calibrate the hyper-parameters of one component at a time. Considering that simultaneous calibration of multiple hyper-parameters is computionally cubersome and the model is difficult to converge, we adopt the second method. In the calibration process, the control variable method was used to obtain the optimal combination of parameters (Cheng *et al.* 2020). Taking the random missing rate of 20% as an example, Figure 8 shows the calibration process of the l^{cv}, l^{dv}, and l^{wv} on the traffic speed dataset. The results show that the RMSE of the Multi-BiSTGN model decreases first and then increases with the

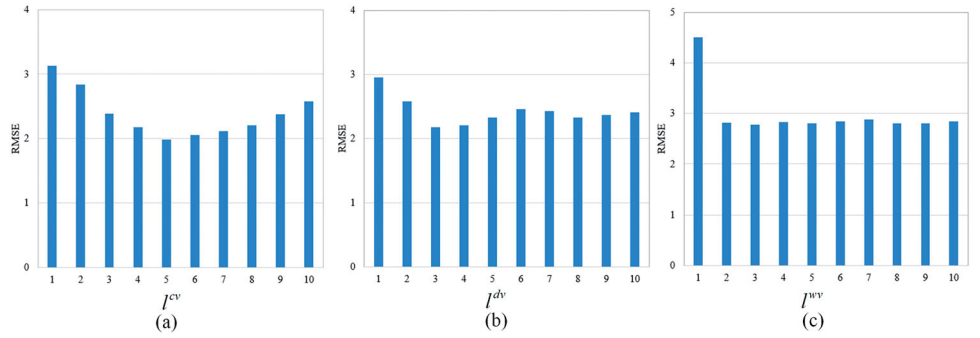

Figure 8. Parameter tuning of the Multi-BiSTGN model on traffic speed data.

increase of l^{cv}. When l^{cv}=5, the model has achieved good accuracy. Similarly, l^{dv} and l^{wv} were also calibrated: l^{dv}=3 and l^{wv}=2.

5.4. Comparison with baselines

To comprehensively evaluate the imputation performance of model proposed in this study, we used ten baseline methods for comparison, including HA (Campbell and Thompson 2008), SES (Gardner 2006), ST-KNN (P. Cai *et al.* 2016, B. Yu *et al.* 2016), ST-2SMR (Jiang *et al.* 2018), ST-ISE (Cheng *et al.* 2020), TRMF (H.-F. Yu *et al.* 2016), BTMF (X. Chen and Sun 2021), LRTC-TNN (X. Chen *et al.* 2020), BTTF (X. Chen and Sun 2021), and BGCP (X. Chen *et al.* 2019) models. These baseline methods can be roughly divided into three categories. The first category includes the HA and SES methods, which are regarded as classic statistical models. The second category includes the ST-KNN, ST-2SMR, ST-ISE, TRMF, and BTMF methods, which are data-driven models for Euclidean data structures. The third category includes the LRTC-TNN, BTTF, and BGCP methods, which are tensor factorization models for Euclidean data structures.

Table 3 shows the comparison results between the Multi-BiSTGN model and the baselines under random missing. Overall, the imputation accuracy of the second and the third category models are slightly higher than that of the first category models, i.e. the imputation performance of the data-driven models is higher than that of the statistical models. For the random missing patterns, the imputation performance of the statistical model is greatly affected by the changes of the missing rate. In contrast, the imputation performance of the data-driven models is less affected by variations of the missing rate. With the increase of missing rate, the imputation performance of the statistical models decreases obviously, while the imputation performance of the data-driven models is relatively stable. The results show that the data-driven models are easier to mine random missing patterns in traffic flow data. However, the data-driven models in the baselines are mainly designed for Euclidean datasets. Compared with the baselines, the Multi-BiSTGN model achieves the highest imputation accuracy, indicating that the Multi-BiSTGN model is more suitable for modeling on non-Euclidean data sets, such as traffic flow.

Table 4 shows the comparison results between the Multi-BiSTGN model and the baselines under block missing. The imputation performance of the three category

Table 3. Comparison results (in MAE/RMSE/MAPE) with baselines for random missing.

	Traffic Speed		Traffic Volume	
Models	MR: 20%	MR: 40%	MR: 20%	MR: 40%
HA	6.86/8.26/7.83%	8.45/10.21/21.08%	8.37/10.63/15.62%	9.83/12.17/17.68%
SES	3.41/4.29/9.83%	4.10/5.16/11.49%	5.98/7.89/12.31%	6.20/7.96/12.50%
ST-KNN	2.53/3.13/7.48%	2.64/3.31/7.62%	6.17/7.83/12.15%	6.69/8.59/12.39%
ST-ISE	2.94/3.82/9.10%	3.13/4.11/9.51%	6.02/7.64/11.45%	6.09/7.77/11.59%
ST-2SMR	3.25/4.27/10.47%	3.46/5.43/10.98%	6.76/8.99/13.16%	7.05/9.23/13.65%
TRMF	2.84/3.48/8.80%	3.29/3.92/9.74%	5.86/7.46/11.03%	5.77/7.34/10.93%
BTMF	2.39/2.92/7.20%	2.50/3.08/7.45%	5.77/7.35/11.01%	5.86/7.46/11.00%
LRTC-TNN	2.47/3.09/7.18%	2.81/3.36/7.89%	5.37/6.87/10.65%	5.48/6.95/10.49%
BTTF	2.61/3.06/7.75%	2.70/3.17/7.81%	5.83/7.39/11.31%	5.93/7.52/11.60%
BGCP	2.75/3.23/8.07%	2.82/3.25/8.12%	5.61/7.23/11.04%	5.63/7.15/10.86%
Multi-BiSTGN	**1.58/2.64/5.58%**	**2.14/2.67/6.23%**	**3.87/5.67/6.79%**	**4.03/5.64/7.22%**

Table 4. Comparison results (in MAE/RMSE/MAPE) with baselines for block missing.

	Traffic Speed		Traffic Volume	
Models	MR: 20%	MR: 40%	MR: 20%	MR: 40%
HA	13.74/17.18/31.87%	14.16/17.62/34.31%	15.92/19.53/28.26%	19.44/23.94/34.22%
SES	8.75/10.71/23.73%	9.57/12.70/24.93%	9.58/13.10/19.82%	10.00/13.58/22.45%
ST-KNN	4.78/6.67/13.65%	6.86/8.81/18.74%	10.22/13.58/19.65%	10.49/13.44/20.45%
ST-ISE	4.06/5.59/12.40%	4.30/5.89/13.07%	8.72/11.78/16.19%	9.16/12.77/16.92%
ST-2SMR	6.68/8.64/15.68%	6.14/8.07/18.13%	10.09/14.25/18.51%	11.84/17.54/21.23%
TRMF	8.34/10.92/21.06%	9.44/12.33/24.57%	6.87/9.10/12.44%	8.59/11.98/15.29%
BTMF	6.76/9.44/17.40%	7.90/9.99/20.67%	6.56/8.68/12.45%	7.25/9.76/14.17%
LRTC-TNN	3.74/4.55/9.65%	4.38/5.47/11.44%	5.97/7.65/11.10%	6.81/8.91/12.67%
BTTF	2.99/3.63/8.03%	3.10/3.79/8.52%	6.05/7.61/12.08%	6.57/8.51/13.04%
BGCP	3.06/3.53/8.60%	3.13/3.74/8.65%	6.00/7.67/11.61%	6.99/9.24/13.68%
Multi-BiSTGN	**2.02/2.56/7.26%**	**2.47/3.06/7.53%**	**4.85/6.98/8.34%**	**5.58/7.63/10.49%**

models shows apparent differences in the block missing, i.e. the imputation performance of the third category models is better than that of the second category models, and the imputation performance of the second category models is greater than that of the first category models. At the same time, in the block missing, the first and the second category models are greatly affected by the missing rate. With the increase of missing rates, the imputation performance of the first and second category models decreases significantly. The reason is that the first and second models are single view models that only consider the temporary closeness view. When there is a block missing in traffic flow, it is difficult for the first and second category models to account for the missing patterns from neighboring time windows (i.e. the closeness view). Compared with the first and second category models, the third models have more stable imputation performance in the block missing. The reason is that the third category models can be regarded as the multi-view models considering the closeness view and daily periodicity view. Even if the traffic flow data is missing in blocks, the third category models can still mine the missing patterns from the daily periodicity view. Compared with all the baselines, the Multi-BiSTGN model obtains the highest imputation performance under the block missing. The reason is that the Multi-BiSTGN model considers multiple views and the non-Euclidean data structure of traffic flow. Similarly, Table 5 shows the comparison results between the Multi-BiSTGN model and the baselines under mixed missing, and the results are similar to those in Table 4.

Table 5. Comparison results (in MAE/RMSE/MAPE) with baselines for mixed missing.

	Traffic Speed		Traffic Volume	
Models	MR: 20%	MR: 40%	MR: 20%	MR: 40%
HA	9.25/12.27/24.41%	12.12/15.41/29.41%	9.61/13.26/17.41%	11.21/14.61/20.14%
SES	7.37/10.44/22.63%	8.26/11.22/22.55%	6.66/8.92/13.99%	7.41/9.79/14.89%
ST-KNN	3.89/6.13/11.62%	4.01/5.72/10.62%	6.81/9.01/13.22%	6.82/9.13/13.79%
ST-ISE	3.41/4.67/10.72%	3.59/4.96/10.93%	6.10/7.77/11.56%	6.48/8.21/12.16%
ST-2SMR	4.68/7.98/15.68%	6.78/8.92/24.13%	7.44/9.45/14.01%	7.80/9.87/15.20%
TRMF	5.08/7.06/15.50%	5.36/7.17/14.90%	6.76/8.92/13.06%	7.74/10.40/14.42%
BTMF	4.38/6.62/13.57%	5.17/7.04/14.40%	6.53/8.45/12.59%	7.25/9.51/14.06%
LRTC-TNN	3.18/3.88/8.43%	3.19/3.92/9.00%	5.84/7.51/11.26%	6.43/8.45/12.54%
BTTF	2.64/3.19/7.42%	2.91/3.51/8.82%	5.97/8.01/11.55%	6.22/7.89/12.52%
BGCP	2.85/3.40/7.72%	3.04/3.62/8.54%	5.89/7.87/11.38%	6.21/8.05/12.16%
Multi-BiSTGN	**1.79/2.68/6.98%**	**2.26/2.69/6.73%**	**4.09/6.08/6.78%**	**5.08/6.65/9.68%**

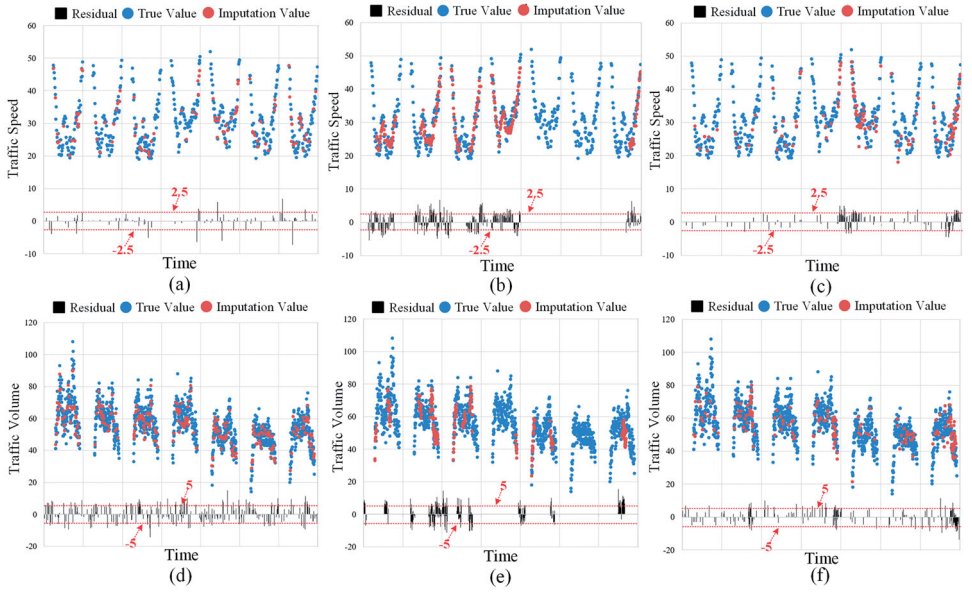

Figure 9. Imputation values and corresponding actual values under 20% missing rate: (a) random missing on traffic speed, (b) block missing on traffic speed, (c) mixed missing on traffic speed, (d) random missing on traffic volume, (e) block missing on traffic volume, and (c) mixed missing on traffic volume.

In general, the Multi-BiSTGN model shows good imputation performance under random missing, block missing, and mixed missing, proving the superiority of the Multi-BiSTGN model.

5.5. Qualitative imputation results of Multi-BiSTGN

In this section, scatter plots are used to describe the imputation performance of the Multi-BiSTGN model. Figure 9 shows the imputation results of the Multi-BiSTGN model with a 20% missing rate. The results show that the imputation values are relatively close to the true values on the two datasets. Among them, the residuals between the observed value and the imputation value are mainly concentrated between [–2.5 and

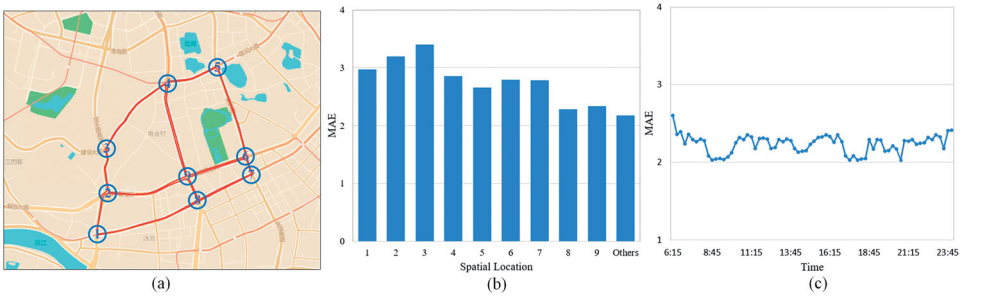

Figure 10. Spatiotemporal heterogeneity of imputation results: (a) spatial division of road segments, where 1 ∼ 9 represent road segments at different spatial locations, (b) spatial heterogeneity, and (c) temporal heterogeneity.

2.5] on the traffic speed dataset, and the residuals between the observed value and the imputation value are mainly concentrated between [–5 and 5] on the traffic volume dataset. This further proves that the imputation result of the Multi-BISTGN model has high precision.

5.6. Spatiotemporal heterogeneity of imputation results

In this subsection, the 40% mixed missing traffic speed data is used to further analyze the spatiotemporal heterogeneity of the imputation results, that is, the difference of imputation results in different spatial locations and different time windows. As shown in Figure 10, in the spatial dimension, we divide the road segments into segments near the intersections and other segments not near the intersections. The results show that the imputation results of segments near intersections are slightly lower than those of segments not near intersections. There are two main reasons for spatial heterogeneity of imputation results. First, the traffic state at the intersection changes more dramatically and the traffic pattern is difficult to capture. Second, the intersections in traffic speed data are mainly concentrated at the boundary of the study area, resulting in partial missing of the topology information of the road segments near the intersections. There are two ways to reduce the influence of the boundary effect on imputation results. First, if there are more relevant data of the boundary data, the relevant data can be added to improve the accuracy of the imputation results. Second, we can use the similarity matrix of the road segment to replace the adjacency matrix in BiSTGN, making the boundary data have more connections and improving the imputation results. In the temporal dimension, the imputation accuracy of travel peak time is slightly higher than in other time periods. The reason may be that the traffic speed changes relatively smoothly during the travel peak time, and the traffic pattern is easy to capture.

5.7. Effect of different components on imputation performance

The Multi-BiSTGN model is composed of three different components, i.e. the BiSTGN model from closeness view (BiSTGN-CV), the BiSTGN model from daily periodicity view (BiSTGN-DV), and the BiSTGN model from weekly periodicity view (BiSTGN-WV).

Table 6. Imputation results (in MAE/RMSE/MAPE) of different components.

	Traffic Speed		Traffic Volume	
Models	MR: 20%	MR: 40%	MR: 20%	MR: 40%
BiSTGN-CV (R)	1.19/1.98/3.82%	1.84/2.75/6.20%	2.20/3.29/4.54%	3.15/4.65/5.97%
BiSTGN-DV (R)	1.69/2.57/6.23%	1.97/2.95/6.77%	4.28/6.13/7.78%	4.50/6.33/8.58%
BiSTGN-WV (R)	2.32/2.82/6.86%	2.49/2.98/7.18%	4.95/6.32/9.54%	5.09/6.53/9.79%
Multi-BiSTGN (R)	**1.58/2.64/5.58%**	**2.14/2.67/6.23%**	**3.87/5.67/6.79%**	**4.03/5.64/7.22%**
BiSTGN-CV (B)	6.69/8.19/24.68%	7.14/8.43/24.13%	10.42/13.16/20.77%	11.61/15.47/21.16%
BiSTGN-DV (B)	1.74/2.53/6.47%	2.46/3.13/6.80%	4.32/6.25/8.44%	5.83/7.20/9.36%
BiSTGN-WV (B)	2.58/3.05/7.32%	2.60/3.28/7.93%	5.67/7.14/10.54%	6.34/8.53/12.40%
Multi-BiSTGN (B)	**2.02/2.56/7.26%**	**2.47/3.06/7.53%**	**4.85/6.98/8.34%**	**5.58/7.63/10.49%**
BiSTGN-CV (M)	4.94/6.61/15.36%	4.97/7.01/18.14%	4.93/7.48/9.82%	5.57/8.85/10.07%
BiSTGN-DV (M)	1.96/2.81/6.61%	2.41/2.93/6.33%	4.29/6.87/7.60%	4.41/6.71/8.65%
BiSTGN-WV (M)	2.48/3.00/6.93%	2.57/3.10/7.07%	5.22/6.68/10.29%	5.42/7.06/10.41%
Multi-BiSTGN (M)	**1.79/2.68/6.98%**	**2.26/2.69/6.73%**	**4.09/6.08/6.78%**	**5.08/6.65/9.68%**

Table 7. Effect of view alignment on imputation results (in MAE/RMSE/MAPE).

	Multi-BiSTGN-NoAligned		Multi-BiSTGN	
Models	MR: 20%	MR: 40%	MR: 20%	MR: 40%
Traffic Speed (R)	2.04/3.26/7.03%	**1.98/2.47/6.13%**	1.58/2.64/5.58%	2.14/2.67/6.23%
Traffic Speed (B)	3.42/4.37/9.72%	4.36/6.73/11.36%	2.02/2.56/7.26%	2.47/3.06/7.53%
Traffic Speed (M)	**1.65/2.53/6.83%**	3.08/3.22/8.42%	1.79/2.68/6.98%	2.26/2.69/6.73%
Traffic Volume (R)	4.32/6.15/8.27%	4.48/6.36/8.46%	3.87/5.67/6.79%	4.03/5.64/7.22%
Traffic Volume (B)	6.21/9.20/18.03%	7.57/10.65/20.50%	4.85/6.98/8.34%	5.58/7.63/10.49%
Traffic Volume (M)	5.46/8.42/18.66%	6.05/9.69/18.51%	4.09/6.08/6.78%	5.08/6.65/9.68%

Therefore, we analyzed the impact of different components on the imputation performance, and the results are shown in Table 6. The results show that the BiSTGN-CV component is more affected by the missing pattern, and the BiSTGN-DV and BiSTGN-WV components are less affected by the missing pattern. When there are only random missing in the dataset, the imputation performance of the BiSTGN-CV component is the best among the three components. When there are block and mixed missing in the dataset, the imputation performance of the BiSTGN-DV and BiSTGN-WV components is better, which further demonstrates the necessity of introducing multi-view components. In addition, the results show that the Multi-BiSTGN model may not be higher than the imputation performance of all components under specific missing patterns, but the imputation performance of the entire Multi-BiSTGN model can be better than that of most components. For example, when random missing happens, the imputation performance of the Multi-BiSTGN model is lower than that of the BiSTGN-CV component, but higher than that of the BiSTGN-DV and BiSTGN-WV components. Considering the complexity of missing patterns in real-world applications, the Multi-BiSTGN model is more practical.

In the model optimization stage, a new loss function is designed to align multiple views. Therefore, we analyzed the influence of view alignment on model imputation accuracy. Table 7 shows the effect of view alignment on imputation results, where Multi-BiSTGN-NoAligned represents the model without view alignment, and Multi-BiSTGN represents the model with view alignment. The results show that the Multi-BiSTGN has better imputation performance under most conditions compared with the Multi-BiSTGN-NoAligned. In 12 comparative experiments, the Multi-BiSTGN-NoAligned model can only achieve better imputation performance in two experiments. In

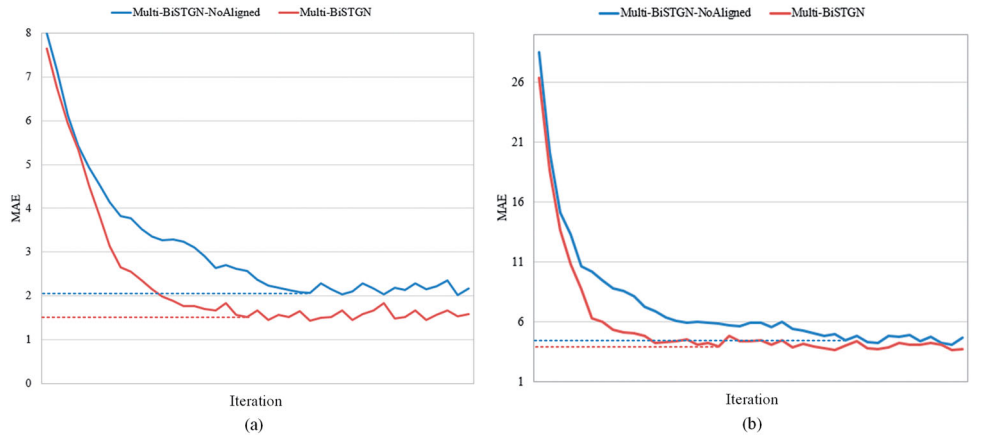

Figure 11. Effect of view alignment on model optimization: (a) traffic speed, and (b) traffic volume.

addition, we further analyzed the impact of view alignment on model optimization, as shown in Figure 11. The results show that view alignment can significantly improve the efficiency of model optimization and make the Multi-BiSTGN model converge faster.

6. Conclusions and future work

Accurately estimating the missing data under complex traffic patterns is one of the essential preparatory work in intelligent transportation systems (ITS), which is of great significance for subsequent traffic flow modeling. However, due to the non-Euclidean structure and complex missing patterns of traffic flows, existing imputation models are difficult to capture the nonlinear spatiotemporal correlations of missing traffic flows. Therefore, a novel multi-view bidirectional spatiotemporal graph network called Multi-BiSTGN is proposed to impute the missing traffic data.

In the experimental section, two real traffic datasets collected in Wuhan, China, were used to verify the imputation performance of Multi-BiSTGN, i.e. traffic speed data and traffic volume data. First, the control variable method is used to calibrate the parameters of the Multi-BiSTGN to obtain the optimal parameter combination of the model. Then, we compared ten existing baselines, including HA, SES, ST-KNN, ST-2SMR, ST-ISE, TRMF, BTMF, LRTC-TNN, BTTF, and BGCP models. Compared to the existing methods, the Multi-BiSTGN model achieved the best imputation performance on three missing types (random missing, block missing, and mixed missing) and two missing rates (20% and 40%). Finally, the influence of different components in Multi-BiSTGN on imputation accuracy was tested, further proving that the proposed method is suitable for traffic flow imputation with complex missing patterns.

The limitations of this study are as follows: (1) As the Multi-BiSTGN model has many hyper-parameters, which introduce extra computational burdens during parameter calibration and may affect the converge speed of the model; (2) The Multi-BiSTGN

model is essentially a general imputation model and can accommodate difference sources of spatiotemporal data, and we only use traffic datasets to verify the imputation performance of the model. Given the above problems, future work will focus on two aspects. First, the automatic parameter tuning will be integrated into the Multi-BiSTGN model to simplify the calibration process. Then, multi-source data, such as air quality data and meteorological data, will be collected to promote the imputation performance of the Multi-BiSTGN model.

Acknowledgments

The numerical calculations in this paper have been done on the supercomputing system in the Supercomputing Center of Wuhan University. We are also very grateful to the anonymous reviewers for their suggestions and the editor's careful revisions.

Data and codes availability statement

The data and codes that support the findings of this study are available in 'figshare.com' with the identifier https://doi.org/10.6084/m9.figshare.18489236.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This project was supported by National Key R&D Program of China (International Scientific & Technological Cooperation Program) under grant 2019YFE0106500, National Natural Science Foundation of China under grant 41871308, the Fundamental Research Funds for the Central Universities.

Notes on contributors

Peixiao Wang is a PhD candidate of State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing (LIESMARS), Wuhan University. He received the M.S. degree from The Academy of Digital China, Fuzhou University in 2020. His research focus on spatiotemporal data mining, social computing, and public health.

Tong Zhang is a Professor with the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing (LIESMARS), Wuhan University. He received the M.Eng. degree in cartography and geographic information system (GIS) from Wuhan University, Wuhan, China, in 2003, and the Ph.D. degree in geography from San Diego State University, San Diego, CA, USA, and the University of California at Santa Barbara, Santa Barbara, CA, in 2007. His research topics include urban computing and machine learning.

Yueming Zheng is currently pursuing the M.S. degree with the Aerospace Information Research Institute, Chinese Academy of Sciences. Her research focus on spatiotemporal analysis, spatiotemporal modeling, and remote sensing of environment.

Tao Hu is an Assistant Professor in Department of Geography at Oklahoma State University. Before joining OSU, he worked as a postdoc research fellow in the Center for Geographic Analysis at Harvard University and the Department of Geography at Kent State University. His

research interests include geospatial big data analysis (i.e. social media), health geography, human mobility, and crime geography.

ORCID

Peixiao Wang http://orcid.org/0000-0002-1209-6340
Tong Zhang http://orcid.org/0000-0002-0683-4669
Yueming Zheng http://orcid.org/0000-0003-0664-8409
Tao Hu http://orcid.org/0000-0002-8557-8017

References

Aryaputera, A.W., *et al.*, 2015. Very short-term irradiance forecasting at unobserved locations using spatio-temporal kriging. *Solar Energy*, 122, 1266–1278.

Bartier, P.M., and Keller, C.P., 1996. Multivariate interpolation to incorporate thematic surface data using inverse distance weighting (IDW). *Computers & Geosciences*, 22 (7), 795–799.

Cai, L., *et al.*, 2020. Traffic transformer: capturing the continuity and periodicity of time series for traffic forecasting. *Transactions in GIS*, 24 (3), 736–755.

Cai, P., *et al.*, 2016. A spatiotemporal correlative k-nearest neighbor model for short-term traffic multistep forecasting. *Transportation Research Part C: Emerging Technologies*, 62, 21–34.

Campbell, J.Y., and Thompson, S.B., 2008. Predicting excess stock returns out of sample: can anything beat the historical average? *The Review of Financial Studies*, 21 (4), 1509–1531.

Chen, M., *et al.*, 2020. Simple and deep graph convolutional networks. *ArXiv*, *2007.02133 [Cs, Stat]*. http://arxiv.org/abs/2007.02133

Chen, X., and Sun, L., 2021. Bayesian temporal factorization for multidimensional time series prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *2021*, 1–1.

Chen, X., He, Z., and Sun, L., 2019. A Bayesian tensor decomposition approach for spatiotemporal traffic data imputation. *Transportation Research Part C: Emerging Technologies*, 98, 73–84.

Chen, X., Yang, J., and Sun, L., 2020. A nonconvex low-rank tensor completion model for spatiotemporal traffic data imputation. *Transportation Research Part C: Emerging Technologies*, 117, 102673.

Cheng, S., and Lu, F., 2017. A two-step method for missing spatio-temporal data reconstruction. *ISPRS International Journal of Geo-Information*, 6 (7), 187.

Cheng, S., *et al.*, 2018. Short-term traffic forecasting: an adaptive ST-KNN model that considers spatial heterogeneity. *Computers, Environment and Urban Systems*, 71, 186–198.

Cheng, S., *et al.*, 2019. Multi-task and multi-view learning based on particle swarm optimization for short-term traffic forecasting. *Knowledge-Based Systems*, 180, 116–132.

Cheng, S., Peng, P., and Lu, F., 2020. A lightweight ensemble spatiotemporal interpolation model for geospatial data. *International Journal of Geographical Information Science*, 34 (9), 1849–1872.

Defferrard, M., Bresson, X., and Vandergheynst, P., 2017. Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering. *ArXiv:1606.09375*, *[Cs, Stat]*. http://arxiv.org/abs/1606.09375

Deng, M., *et al.*, 2018. Generating urban road intersection models from low-frequency GPS trajectory data. *International Journal of Geographical Information Science*, 32 (12), 2337–2361.

Ermagun, A., and Levinson, D., 2018. Spatiotemporal traffic forecasting: review and proposed directions. *Transport Reviews*, 38 (6), 786–814.

Chang, G., and Ge, T., 2011. Comparison of missing data imputation methods for traffic flow. *Proceedings 2011 International Conference on Transportation, Mechanical, and Electrical Engineering (TMEE)*, 639–642.

Gardner, E.S., 2006. Exponential smoothing: The state of the art—Part II. *International Journal of Forecasting*, 22 (4), 637–666.

Geng, X., *et al.*, 2019. Spatiotemporal multi-graph convolution network for ride-hailing demand forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33, 3656–3663.

Hara, Y., Suzuki, J., and Kuwahara, M., 2018. Network-wide traffic state estimation using a mixture Gaussian graphical model and graphical lasso. *Transportation Research Part C: Emerging Technologies*, 86, 622–638.

Hechtlinger, Y., Chakravarti, P., and Qin, J., 2017. A Generalization of Convolutional Neural Networks to Graph-Structured Data. *ArXiv:1704.08165*, *[Cs, Stat]*. http://arxiv.org/abs/1704.08165

Hu, M.-G., *et al.*, 2013. A B-SHADE based best linear unbiased estimation tool for biased samples. *Environmental Modelling & Software*, 48, 93–97.

Jia, T., and Yan, P., 2021. Predicting citywide road traffic flow using deep spatiotemporal neural networks. *IEEE Transactions on Intelligent Transportation Systems*, 22 (5), 3101–3111.

Jiang, L., *et al.*, 2018. A neural network method for the reconstruction of winter wheat yield series based on spatio-temporal heterogeneity. *Computers and Electronics in Agriculture*, 154, 46–53.

Kipf, T.N., and Welling, M., 2017, February 22. Semi-supervised classification with graph convolutional networks. *International Conference on Learning Representations*. International Conference on Learning Representations. http://arxiv.org/abs/1609.02907

Li, L., *et al.*, 2020. Estimation of missing values in heterogeneous traffic data: application of multimodal deep learning model. *Knowledge-Based Systems*, 194, 105592.

Li, L., *et al.*, 2014. Fast inverse distance weighting-based spatiotemporal interpolation: a web-based application of interpolating daily fine particulate matter PM2:5 in the contiguous U.S. using parallel programming and k-d tree. *International Journal of Environmental Research and Public Health*, 11 (9), 9101–9141.

Li, M., *et al.*, 2019. Reconstruction of human movement trajectories from large-scale low-frequency mobile phone data. *Computers, Environment and Urban Systems*, 77, 101346.

Li, M., *et al.*, 2021. Prediction of human activity intensity using the interactions in physical and social spaces through graph convolutional networks. *International Journal of Geographical Information Science*, 35 (12), 2489–2516.

Li, X., *et al.*, 2021. Short-term forecast of bicycle usage in bike sharing systems: a spatial-temporal memory network. *IEEE Transactions on Intelligent Transportation Systems*, *2021*, 1–12.

Li, Y., Li, Z., and Li, L., 2014. Missing traffic data: comparison of imputation methods. *IET Intelligent Transport Systems*, 8 (1), 51–57.

Li, Y., *et al.*, 2017. Gated graph sequence neural networks. *ArXiv:1511.05493*, *[Cs, Stat]*. http://arxiv.org/abs/1511.05493

Duan, P., *et al.*, 2016. STARIMA-based traffic prediction with time-varying lags. *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, 1610–1615.

Pesquer, L., Cortés, A., and Pons, X., 2011. Parallel ordinary kriging interpolation incorporating automatic variogram fitting. *Computers & Geosciences*, 37 (4), 464–473.

Rossi, E., *et al.*, 2020. Temporal graph networks for deep learning on dynamic graphs. *ArXiv: 2006.10637*, *[Cs, Stat]*. http://arxiv.org/abs/2006.10637

Schmidhuber, J., 1992. Learning complex, extended sequences using the principle of history compression. *Neural Computation*, 4 (2), 234–242.

Sun, J., *et al.*, 2020. Predicting citywide crowd flows in irregular regions using multi-view graph convolutional networks. *IEEE Transactions on Knowledge and Data Engineering*, *2020*, 1–1.

Tailor, S.A., *et al.*, 2021. Adaptive filters and aggregator fusion for efficient graph convolutions. *ArXiv:2104.01481*, *[Cs]*. http://arxiv.org/abs/2104.01481

Tang, J., *et al.*, 2021. Missing data imputation for traffic flow based on combination of fuzzy neural network and rough set theory. *Journal of Intelligent Transportation Systems*, 25 (5), 439–454.

Tobler, W.R., 1970. A computer movie simulating urban growth in the detroit region. *Economic Geography*, 46 (2), 234–240.

Veličković, P., *et al.*, 2018. Graph attention networks. *ArXiv:1710.10903*, *[Cs, Stat]*. http://arxiv.org/abs/1710.10903

Williams, B.M., and Hoel, L.A., 2003. Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: theoretical basis and empirical results. *Journal of Transportation Engineering*, 129 (6), 664–672.

Wu, S., *et al.*, 2014. Improved k-nn for short-term traffic forecasting using temporal and spatial information. *Journal of Transportation Engineering*, 140 (7), 04014026.

Wu, Z., *et al.*, 2021. A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 32 (1), 4–24.

Xu, C.-D., *et al.*, 2013. Interpolation of missing temperature data at meteorological stations using P-BSHADE*. *Journal of Climate*, 26 (19), 7452–7463.

Xu, D., *et al.*, 2020. GE-GAN: a novel deep learning framework for road traffic state estimation. *Transportation Research Part C: Emerging Technologies*, 117, 102635.

Yang, B., *et al.*, 2021. ST-LBAGAN: Spatio-temporal learnable bidirectional attention generative adversarial networks for missing traffic data imputation. *Knowledge-Based Systems*, 215, 106705.

Ye, J., *et al.*, 2020. Multi-STGCnet: a graph convolution based spatial-temporal framework for subway passenger flow forecasting. *2020 International Joint Conference on Neural Networks (IJCNN)*, 1–8.

Yi, Z., *et al.*, 2021. Inferencing hourly traffic volume using data-driven machine learning and graph theory. *Computers, Environment and Urban Systems*, 85, 101548.

Yozgatligil, C., *et al.*, 2013. Comparison of missing value imputation methods in time series: the case of Turkish meteorological data. *Theoretical and Applied Climatology*, 112 (1–2), 143–167.

Yu, B., *et al.*, 2016. K-Nearest neighbor model for multiple-time-step prediction of short-term traffic condition. *Journal of Transportation Engineering*, 142 (6), 04016018.

Yu, B., Yin, H., and Zhu, Z., 2018. Spatio-temporal graph convolutional networks: a deep learning framework for traffic forecasting. *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, 3634–3640.

Yu, H.-F., Rao, N., and Dhillon, I.S., 2016. Temporal regularized matrix factorization for high-dimensional time series prediction. *30th Conference on Neural Information Processing Systems (NIPS 2016)*, 15.

Zhan, X., *et al.*, 2017. Citywide traffic volume estimation using trajectory data. *IEEE Transactions on Knowledge and Data Engineering*, 29 (2), 272–285.

Zhang, J., *et al.*, 2020. Multi-graph convolutional network for short-term passenger flow forecasting in urban rail transit. *IET Intelligent Transport Systems*, 14 (10), 1210–1217.

Zhang, M., *et al.*, 2020. Multi-view joint graph representation learning for urban region embedding. *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, 4431–4437.

Zhang, S., *et al.*, 2021. A graph-based temporal attention framework for multi-sensor traffic flow forecasting. *IEEE Transactions on Intelligent Transportation Systems*, *2021*, 1–16.

Zhang, Y., *et al.*, 2020. A novel residual graph convolution deep learning model for short-term network-based traffic forecasting. *International Journal of Geographical Information Science*, 34 (5), 969–995.

Zhao, L., *et al.*, 2020. T-GCN: a temporal graph convolutional network for traffic prediction. *IEEE Transactions on Intelligent Transportation Systems*, 21 (9), 3848–3858.

Zhou, J., *et al.*, 2021. Graph neural networks: a review of methods and applications. *AI Open*, https://arxiv.org/abs/1812.08434v5