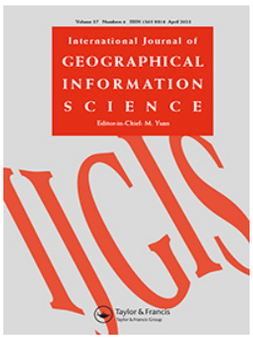

International Journal of Geographical Information Science

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

# Urban traffic flow prediction: a dynamic temporal graph network considering missing values

Peixiao Wang, Yan Zhang, Tao Hu & Tong Zhang

**To cite this article:** Peixiao Wang, Yan Zhang, Tao Hu & Tong Zhang (2023) Urban traffic flow prediction: a dynamic temporal graph network considering missing values, International Journal of Geographical Information Science, 37:4, 885-912, DOI: 10.1080/13658816.2022.2146120

**To link to this article:** https://doi.org/10.1080/13658816.2022.2146120

INTERNATIONAL JOURNAL OF GEOGRAPHICAL INFORMATION SCIENCE
2023, VOL. 37, NO. 4, 885–912
https://doi.org/10.1080/13658816.2022.2146120

Taylor & Francis
Taylor & Francis Group

RESEARCH ARTICLE



# Urban traffic flow prediction: a dynamic temporal graph network considering missing values

Peixiao Wang[a], Yan Zhang[a], Tao Hu[b] and Tong Zhang[a]

[a]State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan, China; [b]Department of Geography, Oklahoma State University, Stillwater, OK, USA

**ABSTRACT**

Accurate traffic flow prediction on the urban road network is an indispensable function of Intelligent Transportation Systems (ITS), which is of great significance for urban traffic planning. However, the current traffic flow prediction methods still face many challenges, such as missing values and dynamic spatial relationships in traffic flow. In this study, a dynamic temporal graph neural network considering missing values (D-TGNM) is proposed for traffic flow prediction. First, inspired by the Bidirectional Encoder Representations from Transformers (BERT), we extend the classic BERT model, called Traffic BERT, to learn the dynamic spatial associations on the road structure. Second, we propose a temporal graph neural network considering missing values (TGNM) to mine traffic flow patterns in missing data scenarios for traffic flow prediction. Finally, the proposed D-TGNM model can be obtained by integrating the dynamic spatial associations learned by Traffic BERT into the TGNM model. To train the D-TGNM model, we design a novel loss function, which considers the missing values problem and prediction problem in traffic flow, to optimize the proposed model. The proposed model was validated on an actual traffic dataset collected in Wuhan, China. Experimental results showed that D-TGNM achieved good prediction results under four missing data scenarios (15% random missing, 15% block missing, 30% random missing, and 30% block missing), and outperformed ten existing state-of-the-art baselines.

**ARTICLE HISTORY**
Received 24 April 2022
Accepted 7 November 2022

**KEYWORDS**
Traffic flow missing; traffic flow prediction; graph neural networks; dynamic graph; Traffic BERT

## 1. Introduction

With the continuous increase of urban vehicles, traffic congestion has gradually become a common problem in almost all modern metropolises, and has seriously disrupted the normal travel of humans (Y. Wang *et al.* 2019, Lin *et al.* 2020, Shi *et al.* 2021). Traffic flow prediction technology, a fundamental objective of the intelligent transportation system (ITS), can dynamically guide traffic flow based on the traffic state predicted by the model, which is of great significance for urban traffic planning

**CONTACT** Tong Zhang zhangt@whu.edu.cn

Supplemental data for this article is available online at https://doi.org/10.1080/13658816.2022.2146120.

(Fang *et al.* 2021, F. Zhou *et al.* 2021a). In recent years, the rapid development of sensors provide essential data sources for traffic flow prediction (J. Yu *et al.* 2020, P. Wang *et al.* 2022a). However, due to technical problems of data collection, data missing is common in real-world applications, which severely limits the use of traffic data (Chen *et al.* 2020, D. Xu *et al.* 2020, Furtlehner *et al.* 2022, P. Wang *et al.* 2022a).

In the early years, some scholars did not consider the missing phenomenon in traffic flow, and simply established prediction models without considering missing values. This kind of classical prediction model mainly includes the spatiotemporal k-nearest-neighbor model (ST-KNN) (S. Wu *et al.* 2014, B. Yu *et al.* 2016a), spatiotemporal residual network (ST-ResNet) (J. Zhang *et al.* 2017), and HIDLST network (Ren *et al.* 2020). In addition, considering that the traffic road network is a non-Euclidean data structure in nature, graph convolutional networks (GCNs) have been applied to traffic flow modeling and achieved state-of-the-art (SOTA) prediction performance (Kipf and Welling 2017), such as temporal graph convolutional network (T-GCN) (Zhao *et al.* 2020), spatiotemporal graph convolutional network (ST-GCN) (B. Yu *et al.* 2018), and residual graph convolutional long short-term memory network (RGC-LSTM) (Y. Zhang *et al.* 2020). Although the above models have achieved good prediction results, there are still shortcomings. Specifically, the above prediction models do not have the ability to deal with missing values but convert data with missing values into data without missing values through preprocessing. The above models adopt two preprocessing methods to deal with the missing values. One is to estimate missing data before constructing the prediction models, and the other is to delete the time series with missing data (Yang *et al.* 2021). The preprocessing methods not only increase the computational complexity of the prediction model, but also may lead to insufficient training data to obtain a reliable prediction model. In addition, traffic flow collected in the real world may contain multiple missing patterns (characteristics of the missing pattern are described in Supplementary Appendix A), which also seriously restrict the performance of the prediction models (Chen *et al.* 2020, D. Xu *et al.* 2020, Furtlehner *et al.* 2022, P. Wang *et al.* 2022b).

In recent years, some scholars have tried to establish traffic prediction models considering missing values, which directly use raw data to predict the future traffic state without preprocessing missing values. For example, Che *et al.* (2018) proposed the gate recurrent unit with decay mechanism (GRU-D), and Tian *et al.* (2018) proposed the long short-term memory network with missing data (LTSM-M). However, GRU-D and LTSM-M are pure time series models that ignore the spatial patterns of traffic flow. In addition, matrix and tensor factorization models also provide a solution for traffic flow prediction based on missing data (H.-F. Yu *et al.* 2016b, Chen and Sun 2022). However, the matrix and tensor factorization models are mainly designed for the Euclidean structure, limiting the modeling ability on non-Euclidean datasets. In recent years, relevant scholars have also applied GCNs to traffic flow modeling with missing data and obtained good prediction results, such as spectral graph Markov network (SGMN) (Cui *et al.* 2020) and Heterogeneous Spatiotemporal graph convolution network (RIHGCN) (Zhong *et al.* 2021). However, the SGMN and RIHGCN models rely heavily on the predefined adjacency matrix. In the natural traffic environment, due to the influence of signal lights and traffic events, the spatial association between roads is not fixed, but dynamically changes with time. Therefore, the predefined adjacency

matrix cannot accurately describe the spatial association between roads (Z. Wu *et al.* 2021, F. Zhou *et al.* 2021a).

In general, the current traffic flow forecasting methods still face two challenges. One is that missing data affects the performance of prediction model, and the other is that fixed adjacency matrixes cannot accurately describe the dynamic spatial association on the traffic network. This study proposes a novel solution to address the above two challenges. Specifically, a dynamic temporal graph network considering missing values (D-TGNM) is proposed for traffic flow prediction. The main contributions are summarized as follows:

1. We propose a novel temporal graph neural network considering missing values (TGNM) for traffic flow prediction under missing data scenarios. In the TGNM model, we designed a missing data processing component to capture missing patterns in traffic flow automatically. Supported by the proposed component, the TGNM model does not need to impute the missing values in the traffic flow in advance, but directly captures the complex spatiotemporal relationships in the traffic flow for traffic flow prediction.
2. Inspired by Bidirectional Encoder Representations from Transformers (BERT) (Devlin *et al.* 2019, Bao *et al.* 2021), we develop a novel self-supervised learning method called Traffic BERT, which enables the TGNM model to capture dynamic spatial relationships in traffic flow.
3. A novel loss function is proposed to optimize the parameters of proposed model. The proposed loss function divides the optimization function into temporal-based imputation tasks, spatial-based imputation tasks, and prediction tasks. That is, the loss function considers not only the problem of missing value, but also the problem of traffic flow prediction.

## 2. Related works

In this section, we systematically review the work related to this study. Existing traffic flow prediction methods can be roughly divided into two categories: spatiotemporal prediction based on complete data, and spatiotemporal prediction based on incomplete data, which are discussed in Sections 2.1–2.2, respectively. Among them, complete data refers to data without missing values, and incomplete data refers to data containing missing values.

### *2.1. Spatiotemporal prediction models based on complete data*

As the prediction models based on complete data are challenging to deal with the missing values in the data, the prediction models based on complete data are often divided into multi-stage modeling. More specifically, relevant scholars first used imputation methods, such as matrix factorization (Asif *et al.* 2016, H.-F. Yu *et al.* 2016b), tensor factorization (Chen *et al.* 2019, Chen and Sun 2022), and spatiotemporal interpolation (S. Cheng and Lu 2017, S. Cheng *et al.* 2020), to estimate the missing values in traffic flow, and then model the complete traffic flow to predict the future

traffic state (Liu *et al.* 2018, Ge *et al.* 2019, Q. Li *et al.* 2020). For example, S. Cheng and Lu (2017) adopted a two-step spatiotemporal interpolation (ST-2SMR) to impute the missing values of traffic flow, and proposed adaptive spatiotemporal k-nearest neighbor (adaptive-STKNN) (S. Cheng et al. 2018) and dynamic spatiotemporal k-nearest neighbor (D-STKNN) (S. Cheng et al. 2021) to model the complete traffic flow. P. Wang et al. (2021) used the linear interpolation method (P. Cai et al. 2016) to impute the missing values in the dataset, and proposed an improved deep belief network (Improved-DBN) to predict the traffic state in 30, 60, and 120-min intervals. In addition, some scholars directly deleted the incomplete data series to model the remaining complete traffic flow data. For instance, J. Zhang et al. (2017) proposed the spatiotemporal residual networks (ST-ResNet) to forecast the inflow and outflow in every city region. L. Cai et al. (2020) proposed a novel deep learning framework called Traffic Transformer to capture the continuity and periodicity of time series and to model spatial dependency. Although above models have achieved good prediction performance, there are still shortcomings. More specifically, vehicles can only travel along road networks, which manifest non-Euclidean geometry and topology structures. Above models are mainly designed for Euclidean structured datasets, limiting the modeling ability for non-Euclidean datasets (Z. Wu et al. 2021, J. Zhou et al. 2021b).

Fortunately, the rapid development of graph convolutional neural networks (GCNs) provides a solution for non-Euclidean data modeling (Y. Zhang *et al.* 2020, M. Li *et al.* 2021, Yi *et al.* 2021). As mentioned above, the traffic road network is a non-Euclidean data structure in nature, the GCNs are naturally applied to traffic flow modeling and achieved SOTA performance (Y. Zhang *et al.* 2020, K. Zhang *et al.* 2021a, S. Zhang *et al.* 2022). For example, Zhao *et al.* (2020) proposed a temporal graph convolutional network (T-GCN), which uses the GRU and GCN models to mine the temporal patterns and spatial patterns of traffic flow, respectively. B. Yu *et al.* (2018) integrated the one-dimensional CNN into the GCN and proposed a spatiotemporal graph convolutional network (ST-GCN) to mine the spatiotemporal patterns of traffic flow. K. Zhang *et al.* (2021a) integrated the temporal convolutional network (TCN) into the GCN and proposed a novel graph attention temporal convolutional network (GATCN) to predict future traffic state. T. Zhou *et al.* (2022) proposed an attention-based hybrid spatiotemporal model for city-wide traffic flow forecasting, accounting for spatial and feature heterogeneity of traffic flows. Although these graph-based prediction models have the capability to handle non-Euclidean data, most of them can only make predictions based on complete data. The prediction models based on complete data either estimate missing data before constructing the prediction models, or delete the time series with missing data. The former adds an extra computational burden and the imputation accuracy directly affects the performance of the prediction models (S. Cheng *et al.* 2018, 2019, 2021), while the latter may lead to insufficient training data for the models and fail to obtain reliable traffic flow patterns (L. Cai *et al.* 2020, Y. Zhang *et al.* 2020, Yi *et al.* 2021).

### 2.2. Spatiotemporal prediction models based on incomplete data

Compared with the prediction models based on complete data, the prediction models based on incomplete data integrate missing patterns into the prediction models and

directly use raw data to predict the future traffic state. For example, Che *et al*. (2018) proposed the GRU-D model based on the gated recurrent unit (GRU), which effectively integrates *masking* and *time interval* into the deep model architecture to capture the long-term time dependence in the missing time series. Tian *et al*. (2018) integrated the multi-scale time information of traffic flow into the long short term memory network (LSTM), and proposed the LSTM-M to model missing traffic flow. Although GRU-D and LSTM-M models can model missing traffic sequences, there are still deficiencies. Specifically, GRU-D and LSTM-M models are simple time series models, which ignore the impact of spatial information on traffic flow prediction (Ermagun and Levinson 2018, Medrano and Aznarte 2021). In addition, matrix/tensor factorization models provide a natural solution to address traffic flow prediction with missing data. For instance, H.-F. Yu *et al*. (2016b) proposed the temporal regularized matrix factorization model (TRMF) for traffic flow prediction with missing data based on the rolling prediction scheme and the traditional matrix factorization model. To improve the nonlinear fitting ability of the matrix factorization, Yang *et al*. (2021) integrated LSTM and graph Laplacian (GL) into the solution of the matrix factorization, and proposed the LSTM-GL-ReMF model for traffic flow prediction with missing data.

Considering the non-Euclidean structure of the traffic road network, relevant scholars also applied GCNs to the traffic flow prediction with missing values. For example, Cui *et al*. (2020) proposed a novel spectral graph Markov network (SGMN), which gradually infers the missing data and predicts the future traffic state by defining the Markov process on the graph. Zhong *et al*. (2021) proposed a heterogeneous spatio-temporal graph convolution network (RIHGCN) for traffic forecasting with missing values. However, compared with the prediction models based on complete data, limited research explored the graph-based prediction model considering missing values. In addition, SGMN and RIHGCN models are a kind of spectral GCN, which rely heavily on the predefined static adjacency matrix. In other words, the spatial relationship between road segments is constant in SGMN and RIHGCN models. However, in the actual scene, the spatial relationship between road segments often changes dynamically with time, which makes it difficult for the static matrix to capture the dynamic spatial relationship (Diao *et al*. 2019, M. Xu *et al*. 2021).

Therefore, to address above two challenges (dynamic spatial associations and missing value), we propose a dynamic temporal graph network considering missing values (D-TGNM) for traffic flow prediction. The D-TGNM model does not rely on the predefined static adjacency matrix but captures the dynamic associations in the traffic flow. In addition, the D-TGNM model also can to predict traffic flow under missing scenarios.

## 3. Preliminaries and problem definitions

In this section, we first introduce the definitions of several key concepts, and then present the traffic flow prediction problem based on the definitions.

**Definition 1** (**Traffic Road Network**), As shown in Figure 1(a), a traffic road network can be abstracted as a graph structure $G =< V, E, \boldsymbol{A} >$, where $V = \{v_i\}_{i=1}^{n}$ represents $n$ nodes in the graph $G$ (such as $n$ intersections or detectors); $E$ represents the set of

edges in $G$, i.e., the relationships of nodes. For simplicity, the connection relationships between nodes can be represented by an association matrix $\boldsymbol{A} \in \mathcal{R}^{n \times n}$. **Note:** The connection relationship between nodes is not the same as the topological relationship in the physical world, but an implicit relationship dynamically learned through the states between nodes.

**Definition 2 (Traffic State),** As shown in Figure 1(b), the traffic state on the entire road network $G$ within a specific time window can be expressed as a spatiotemporal state matrix $\boldsymbol{X} \in \mathcal{R}^{n \times g}$, where $x_i^t$ represents the traffic state of node $v_i$ in the time window $\tau_t$, $\boldsymbol{x}^t = \{x_i^t\}_{i=1}^n \in \mathcal{R}^{n \times 1}$ represents a spatial sequence of all nodes in the time window $\tau_t$.

**Definition 3** (**Zero-one Matrix**), Zero-one matrix $\boldsymbol{M} \in \mathcal{R}^{n \times g}$ is a matrix containing only elements 0 and 1, whick is used to distinguish missing values and observed values in the spatiotemporal state matrix $\boldsymbol{X}$. If $\boldsymbol{M}_i^t = 0$, it means that the traffic state of node $v_i$ in the time window $\tau_t$ is missing, i.e., $x_i^t = \phi$. Similarly, $\boldsymbol{m}^t = \boldsymbol{M}^t \in \mathcal{R}^{n \times 1}$ is used to distinguish missing values and observed values in the spatial sequence $\boldsymbol{x}^t$.

**Definition 4** (**Dynamic Temporal Graph Sequence**), A dynamic temporal graph sequence $DTGS = \{G^t\}_{t=1}^m$ represents a dynamic graph sequence in which graph information changes over time. Specifically, the dynamic graph means that the traffic state and the relationships between nodes in the graph change over time, i.e., $G^t = < V, E^t, \boldsymbol{A}^t, \boldsymbol{x}^t, \boldsymbol{m}^t >$. As shown in Figure 1(c), $G^t$ graph information in time window $\tau_t$, where $\boldsymbol{x}^t$ and $\boldsymbol{m}^t$ have the same meaning as in Definition 2 and Definition 3, $E^t$ and $\boldsymbol{A}^t$ represent the associations between nodes in the time window $\tau_t$, i.e., the relationships between nodes also changes over time.

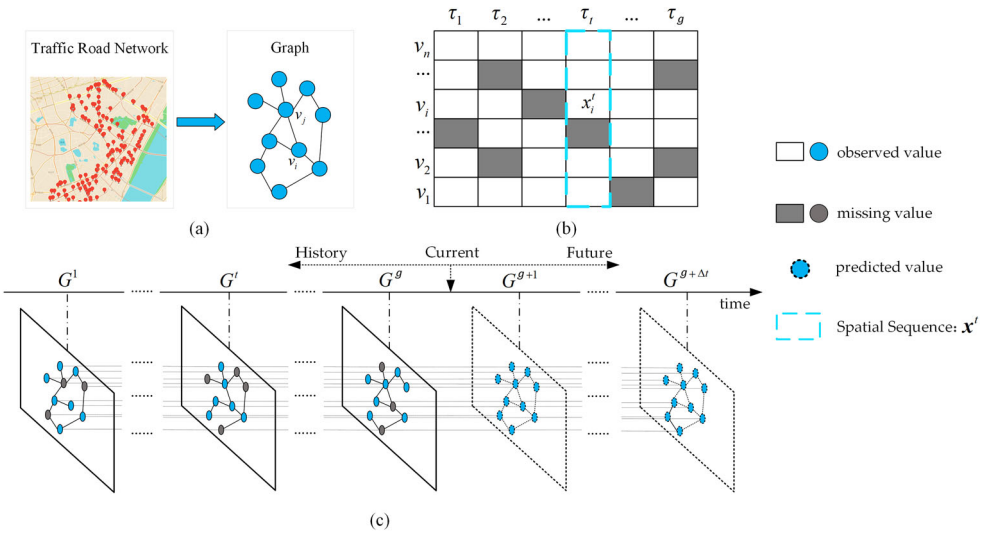


**Figure 1.** Preliminary definitions: (a) traffic road network can be abstracted as a graph structure, (b) the traffic state on the entire road network can be expressed as a spatiotemporal state matrix, and (c) the studied traffic flow prediction task in missing.

Our work aims to build a function $\mathcal{F}(\cdot)$ that can mine the spatiotemporal correlations of traffic flow from the dynamic temporal graph sequence with missing values to forecast future traffic flow accurately. Given a dynamic temporal graph sequence, the modeling process is shown in Formula (1).

$$\{\hat{\boldsymbol{x}}^t\}_{t=g+1}^{g+\Delta t} = \mathcal{F}\left(\{G^t\}_{t=1}^{g}; \boldsymbol{\Theta}\right) \tag{1}$$

where $\{G^t\}_{t=1}^{g}$ represents the historical dynamic temporal graph sequence with missing values; $\{\hat{\boldsymbol{x}}^t\}_{t=g+1}^{g+\Delta t}$ represents the future dynamic temporal graph sequence; $\mathcal{F}(\cdot)$ represents the prediction model proposed in this study, i.e., D-TGNM model; $\Delta t$ represents the prediction step, $\Delta t = 1$ represents single step prediction, $\Delta t > 1$ represents multi-step prediction; $\boldsymbol{\Theta}$ indicates the learnable parameter in the model.

## 4. Methodology

In this section, we describe the proposed D-TGNM model for traffic flow prediction considering missing values. The structure of D-TGNM is presented in Figure 2. The components of D-TGNM model are introduced in Sections 4.1–4.3, respectively. First, we construct a dynamic graph sequence to describe the traffic states of the studied road network. Then, to address two challenges (i.e., dynamic spatial associations and missing value) in the dynamic graph sequence, we propose a novel self-supervised learning method called Traffic BERT and a novel temporal graph network considering

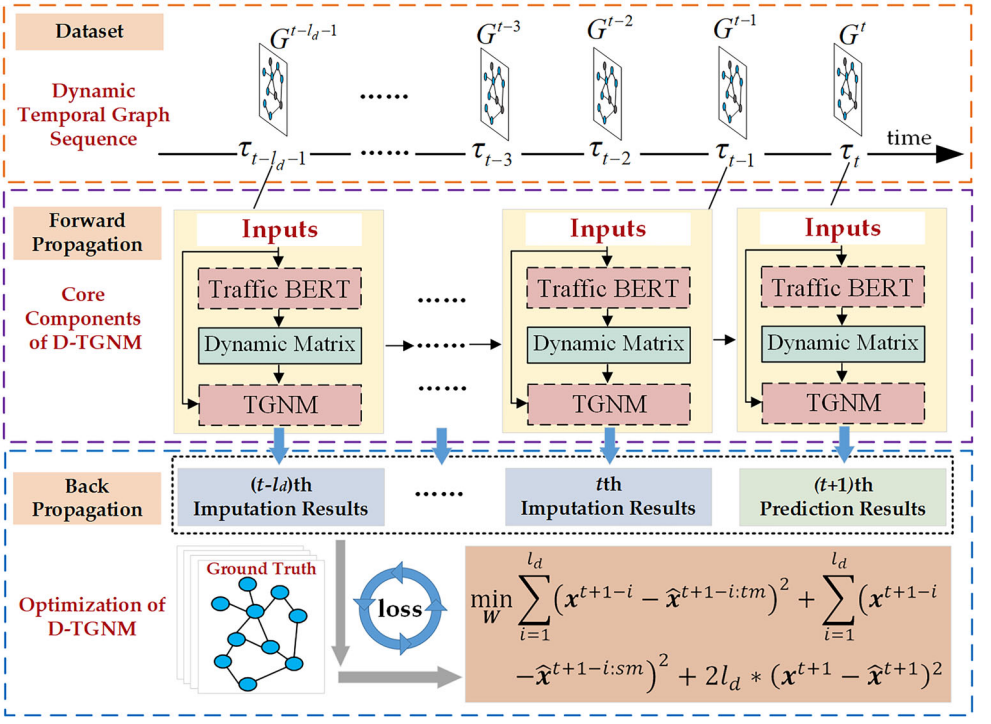


**Figure 2.** Workflow of the D-TGNM model.

missing values (TGNM), respectively. Finally, the proposed D-TGNM model can be obtained by integrating the dynamic spatial associations learned by Traffic BERT into the TGNM model. More specifically, the dynamic graph sequence is used as the input of Traffic BERT to obtain the dynamic spatial associations in traffic flow. Then, the dynamic graph sequence and dynamic spatial associations are used as the input of D-TGNM model to obtain the final output. To train the D-TGNM model, we design a novel loss function, which considers the missing values problem and prediction problem in traffic flow, to optimize the proposed model.

### 4.1. Construction of dynamic association matrix

Due to signal timing and real-time traffic states, the spatial associations between road segments are often dynamic rather than static, i.e., the spatial associations between roads change over time (Diao *et al.* 2019). For example, when the signal light is red, vehicles will be prohibited from passing, even if there is a topological connection between the two roads in the physical world. However, existing graph convolutional networks mostly use the predefined static matrix to encode spatial associations on the road network, ignoring the dynamics of spatial associations (Zhang *et al.* 2022). Compared with the explicit static spatial associations, the implicit dynamic spatial associations may more accurately describe the spatial relationships between roads. Therefore, we attempt to integrate the dynamic spatial associations into the graph convolutional networks to further improve the accuracy of traffic flow prediction. Inspired by BERT, we extended the classic BERT model to learn the implicit dynamic spatial associations on the traffic road network (aka., Traffic BERT).

The BERT model is a pre-training model for natural language processing, which learns the implicit semantic associations between words based on text data (Y. Zhang *et al*. 2021b; Y. Zhang *et al.* 2022). By analogy with the traffic environment, the Traffic BERT model learns the implicit semantic associations between spatial locations based on the spatial sequences (Definition 2) of traffic states, i.e., the implicit spatial associations. Compared with the traditional BERT model, the Traffic BERT model has two main differences. First, Traffic BERT model is used to process continuous traffic flow data rather than discrete text data. Second, Traffic BERT focuses on the implicit spatial associations (i.e., attention matrix) between traffic flows rather than the output of the model, i.e., Traffic BERT model explicitly uses attention matrix to describe the similarity between different nodes (intersections or detectors). As shown in Figure 3, the Traffic BERT model is mainly composed of two stages: model training and dynamic matrix generation. The training stage is mainly used to optimize the parameters of the Traffic BERT model. The dynamic matrix generation stage is used to obtain the corresponding dynamic spatial association matrix of the spatial sequence.

As shown in Figure 3, the spatial sequence containing the missing values obtains the final output through state embedding, position embedding, and $L$ encoders in turn. In the Traffic BERT model, the encoder is the key to obtain the implicit dynamic spatial associations on the traffic network. Therefore, taking the spatial sequence $\boldsymbol{x}^t$ of the $t$ th time window as an example, we describe the operation process of a single

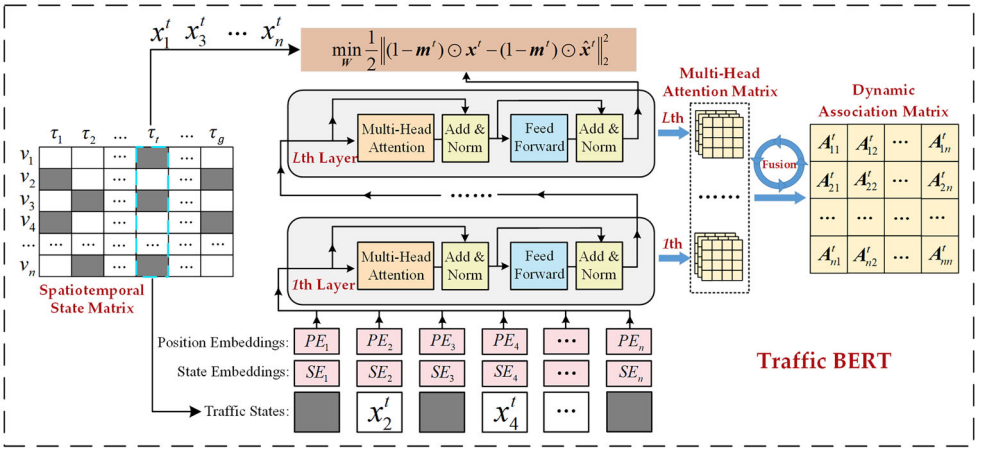


**Figure 3.** Illustration of the Traffic BERT.

encoder, which is shown in Formulas (2) and (3).

$$\begin{cases} \boldsymbol{I}^t = (\boldsymbol{x}^t \odot \boldsymbol{m}^t)\boldsymbol{W}_{se} + \boldsymbol{W}_{pe} \\ \boldsymbol{I}^t = \boldsymbol{I}^t + \text{Concat}\left(\boldsymbol{U}_1^t, \boldsymbol{U}_2^t, \ldots, \boldsymbol{U}_p^t\right)\boldsymbol{W}_u \\ \boldsymbol{O}^t = \boldsymbol{I}^t + \text{gelu}(\boldsymbol{I}^t\boldsymbol{W}_{ff})\boldsymbol{W}_f \\ \hat{\boldsymbol{x}}^t = \boldsymbol{O}^t\boldsymbol{W}_o \end{cases} \tag{2}$$

$$\begin{cases} \boldsymbol{U}_p^t = \boldsymbol{T}_p^t\left(\boldsymbol{I}^t\boldsymbol{W}_{v:p}\right) \\ \boldsymbol{T}_p^t = \text{softmax}\left(\dfrac{\left(\boldsymbol{I}^t\boldsymbol{W}_{q:p}\right)\left(\boldsymbol{I}^t\boldsymbol{W}_{k:p}\right)^T}{d_k}\right) \end{cases} \tag{3}$$

where $\boldsymbol{I}^t \in \mathcal{R}^{n\times d_e}$ and $\boldsymbol{O}^t \in \mathcal{R}^{n\times d_e}$ respectively represent the input and output of the encoder ($d_e$ represents the matrix dimensions required by the input and output of the encoder. In general, the output of the former encoder can be used as the input of the latter encoder); $\boldsymbol{m}^t \in \mathcal{R}^{n\times 1}$ is used to identify observed and missing values in $\boldsymbol{x}^t$, and $\boldsymbol{x}^t \odot \boldsymbol{m}^t$ means that only the observed data is input into the Traffic BERT model; $\hat{\boldsymbol{x}}^t \in \mathcal{R}^{n\times 1}$ represents the output of the Traffic BERT model, which is used to calculate the loss function. $\boldsymbol{U}_1^t, \boldsymbol{U}_2^t, \ldots, \boldsymbol{U}_p^t \in \mathcal{R}^{n\times d_k}$ indicate the results of input $\boldsymbol{I}^t$ through multi-head attention mechanism ($d_k$ represents the matrix dimensions required by the multi-head attention mechanism); In the multi-headed attention mechanism (Formula 3), $\boldsymbol{T}_p^t \in \mathcal{R}^{n\times n}$ represents the attention matrix of the $p$-head, which can be used to obtain the dynamic spatial association matrix of the spatial sequence $\boldsymbol{x}^t$; $\boldsymbol{W}_{se} \in \mathcal{R}^{1\times d_e}$, $\boldsymbol{W}_{pe} \in \mathcal{R}^{1\times d_e}$, $\boldsymbol{W}_u \in \mathcal{R}^{(p\times d_k)\times d_e}$, $\boldsymbol{W}_{ff} \in \mathcal{R}^{d_e\times 4d_e}$, $\boldsymbol{W}_f \in \mathcal{R}^{4d_e\times d_e}$, $\boldsymbol{W}_o \in \mathcal{R}^{d_e\times 1}$, $\boldsymbol{W}_{v:p} \in \mathcal{R}^{d_e\times d_k}$, $\boldsymbol{W}_{q:p} \in \mathcal{R}^{d_e\times d_k}$, and $\boldsymbol{W}_{k:p}$ respectively represent the learnable parameters, which are mainly used for the dimension alignment in the calculation process; $\text{Concat}(\cdot)$ represents the matrix connection function; $gelu(\cdot)$ represents the activation function of gaussian error linear units; $\text{softmax}(\cdot)$ represents the normalized exponential function.

After obtaining the output $\hat{\boldsymbol{x}}^t$, the parameters of the Traffic BERT model can be optimized by minimizing the mean square error between the output values $\hat{\boldsymbol{x}}^t$ and the actual values $\boldsymbol{x}^t$. The optimization process is shown in Formula (4).

$$\mathcal{L}(\boldsymbol{W}) = \min_{\boldsymbol{W}} \frac{1}{2}(\boldsymbol{x}^t \odot (1 - \boldsymbol{m}^t) - \hat{\boldsymbol{x}}^t \odot (1 - \boldsymbol{m}^t))^2 \tag{4}$$

where $\boldsymbol{W}$ represents the learnable parameter of the Traffic BERT; $\boldsymbol{m}^t \in \mathcal{R}^{n\times 1}$ is used to identify the observed values and missing values in $\boldsymbol{x}^t$; $\odot$ represents the Hadamard product; $\boldsymbol{x}^t \odot (1 - \boldsymbol{m}^t) - \hat{\boldsymbol{x}}^t \odot (1 - \boldsymbol{m}^t)$ means that only the output value of the missing position is used to optimize the parameters of the Traffic BERT. As shown in Figure 4, from a spatial perspective, the essence of the Traffic BERT model is to use potential spatial associations to estimate missing values in the spatial sequence. Therefore, the process of optimizing the Traffic BERT model is the process of learning the potential spatial associations in the spatial sequence.

After obtaining the optimized Traffic BERT model, we obtain the final dynamic spatial association matrix by fusing the multi-head attention matrix in the encoder. Although Traffic BERT model contains $L$-layer encoder, we only fuse the attention matrix in the last layer encoder in order to facilitate calculation. Taking the spatial sequence $\boldsymbol{x}^t$ of the $t$ th time window as an example, the fusion process is shown in Formula (5).

$$\boldsymbol{A}^t = \sum_{i=1}^{p} \frac{\boldsymbol{T}_i^t}{p} \tag{5}$$

where $\boldsymbol{A}^t \in \mathcal{R}^{n\times n}$ represents the implicit spatial associations in the spatial sequence $\boldsymbol{x}^t \in \mathcal{R}^{n\times 1}$, and mainly used for subsequent traffic flow prediction; $\boldsymbol{T}_i^t \in \mathcal{R}^{n\times n}$ represents the similarity matrix of the $i$ th head, and its calculation method is the same as that of Formula (3).

### 4.2. Construction of the TGNM

By replacing the adjacency matrix in traditional GCNs with the dynamic association matrix obtained by Traffic BERT, traditional GCNs can capture the dynamic spatial association in traffic flow. Compared with static adjacency matrix, the time-varying dynamic association matrix is more suitable for describing the spatial associations in the graph. However, the GCNs integrating Traffic BERT still have two shortcomings.

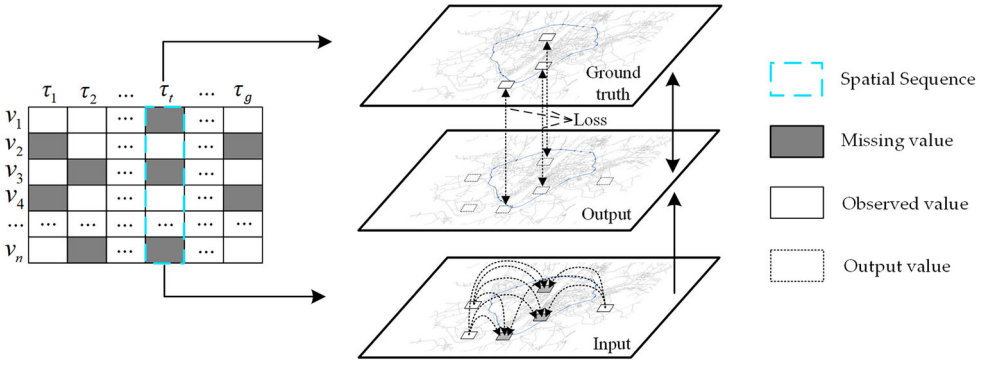


Figure 4. Optimization of the Traffic BERT: the essence of the Traffic BERT model is to use potential spatial associations to estimate missing values in the spatial sequence.

First, the traditional GCNs are mainly designed for spatial data, and it is difficult to mine the temporal patterns in dynamic temporal graph sequence (Rossi *et al.* 2020). Second, the traditional GCNs cannot effectively deal with missing values in traffic flow data (Cui *et al.* 2020). To solve the above shortcomings, we propose a novel temporal graph network considering missing values (TGNM) for traffic flow modeling.

The TGNM model defines graph convolution operations under time constraints from the perspective of neighborhood feature aggregation. As shown in Figure 5, inspired by the recurrent neural network, we define two memory states for each node in the graph. Based on the memory state, the basic idea of the TGNM model can be simply described as three steps. First, the temporal memory state $\boldsymbol{H}^{t:tm}$ of the $t$ th time window is obtained from $\boldsymbol{H}^{t-1:tm}$ and $G^t$. Then, the spatial memory state $\boldsymbol{H}^{t:sm}$ of the $t$ th time window is obtained from $\boldsymbol{H}^{t-1:sm}$ and $G^t$. Finally, $\boldsymbol{H}^{t:tm}$ and $\boldsymbol{H}^{t:sm}$ are integrated to obtain the final prediction results. The advantage of the TGNM model is that it captures the temporal and spatial dependence of graph nodes at the same time. By analogy with the traffic environment, the traffic state of the current road is not only affected by its own historical states, but also by the historical states of its adjacent roads.

### 4.2.1. Forward propagation of the TGNM

Figure 6 describes the forward propagation process of the TGNM model in detail. The DTGM model iteratively updates the spatial memory state and the time memory state of the previous moment to obtain the final memory state. As the missing pattern will be automatically captured before the traffic flow enters the neuron interior (discussed later), the neuron interior directly operates on the complete data. In addition, there may be long-term dependence in temporal graph sequences, and the gating mechanism of GRU (Chung *et al.* 2014) is integrated into the neuron interior. Taking the $t$ th time window as an example, the forward propagation process of the neuron interior is shown in Formula (6).

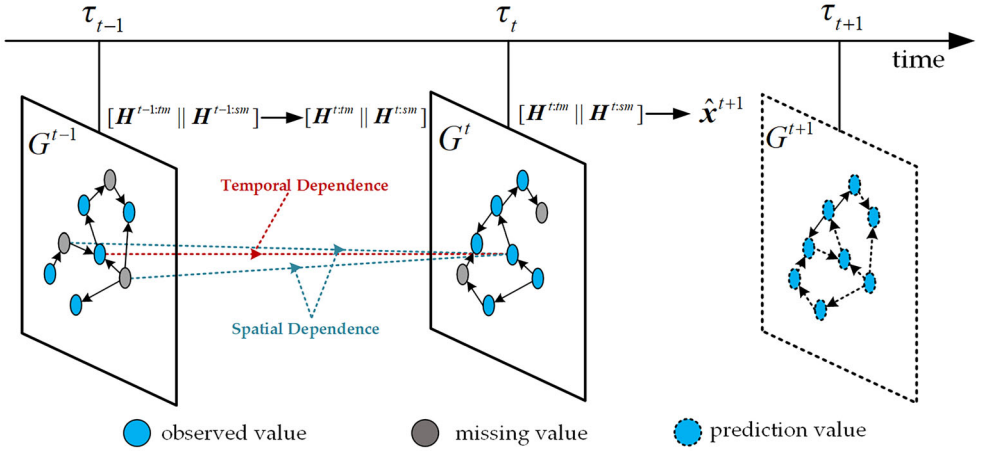


**Figure 5.** Illustration of memory state in the TGNM model: $\boldsymbol{H}^{t:tm} = \left\{\boldsymbol{h}_i^{t:tm}\right\}_{i=1}^{n}$ represents the memory state in the time dimension, and $\boldsymbol{H}^{t:sm} = \left\{\boldsymbol{h}_i^{t:sm}\right\}_{i=1}^{n}$ represents the memory state in the space dimension. The red dashed line indicates the temporal dependency of the target node, and the blue dashed line indicates the spatial dependency of the target node.

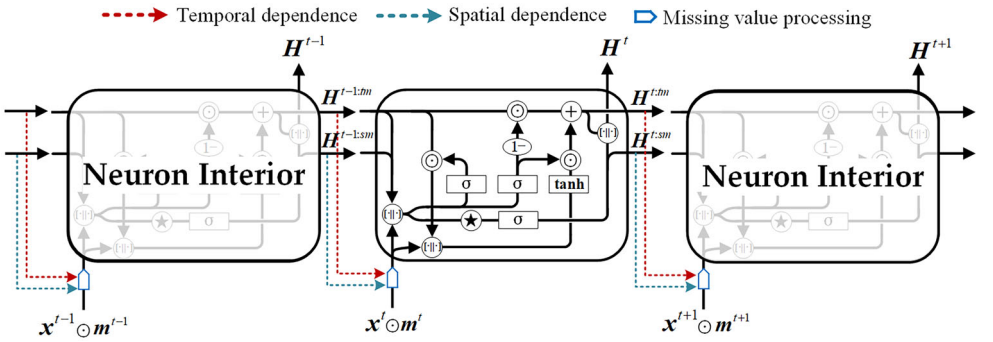


**Figure 6.** Forward propagation illustration of TGNM model: the traffic flow containing missing values enters the neuron interior through the missing value processing, thereby updating the spatial memory state and temporal memory state of the previous moment.

$$
\begin{cases}
\boldsymbol{r}_i^{t:tm} = \sigma\left(\left[\boldsymbol{h}_i^{t-1:tm} || x_i^t\right]\boldsymbol{W}_r\right) \\
\boldsymbol{z}_i^{t:tm} = \sigma\left(\left[\boldsymbol{h}_i^{t-1:tm} || x_i^t\right]\boldsymbol{W}_z\right) \\
\tilde{\boldsymbol{h}}_i^{t:tm} = \tanh\left(\left[\left(\boldsymbol{r}_i^{t:tm} \odot \boldsymbol{h}_i^{t-1:tm}\right) || x_i^t\right]\boldsymbol{W}_{\hat{h}}\right) \\
\boldsymbol{h}_i^{t:tm} = \left(1 - \boldsymbol{z}_i^{t:tm}\right) \odot \boldsymbol{h}_i^{t-1:tm} + \boldsymbol{z}_i^{t:tm} * \tilde{\boldsymbol{h}}_i^{t:tm} \\
\left(\star v_i\right)^t = \sum_{j \in N_i^t} \boldsymbol{A}_{ij}^t \left[\boldsymbol{h}_j^{t-1:sm} || x_j^t\right] \\
\boldsymbol{h}_i^{t:sm} = \sigma\left(\sigma\left(\left(\star v_i\right)^t \boldsymbol{W}_{sm}\right)\boldsymbol{W}_{ssm}\right) \\
\hat{x}_i^{t+1} = \boldsymbol{h}_i^t \boldsymbol{W}_{out} = \left[\boldsymbol{h}_i^{t:tm} || \boldsymbol{h}_i^{t:sm}\right]_{\boldsymbol{W}_{out}}
\end{cases}
\tag{6}
$$

where $\hat{x}_i^{t+1} \in \mathcal{R}^{1\times 1}$ represents the prediction value of node $v_i$ in the $(t+1)$th time window; $x_i^t \in \mathcal{R}^{1\times 1}$ represents the observed (or pre-processed) value of node $v_i$ in the $t$th time window; $x_j^t \in \mathcal{R}^{1\times 1}$ represents the observed (or pre-processed) value of node $v_j$ in the $t$th time window; $\boldsymbol{h}_i^{t-1:tm} \in \mathcal{R}^{1\times d_h}$ represents the temporal memory state of the node $v_i$ in the ($t$-1)th time window; $\boldsymbol{h}_i^{t-1:sm} \in \mathcal{R}^{1\times d_h}$ represents the spatial memory state of the node $v_i$ in the ($t$-1)th time window; $[\cdot||\cdot]$ represents the vector connection function; $\sigma(\cdot)$ represents the sigmoid activation function; $(\star v_i)^t$ represents the graph convolution operation of the node $v_i$ in the $t$th time window; $\boldsymbol{A}^t$ represents the spatial association matrix of the graph structure (when $\boldsymbol{A}^t$ is dynamically generated by Traffic BERT, the TGNM model evolves into the D-TGNM model); $N_i^t$ represents the set of spatial neighbors of node $v_i$ in the $t$th time window; $\boldsymbol{W}_r \in \mathcal{R}^{(d_h+1)\times d_h}$, $\boldsymbol{W}_z \in \mathcal{R}^{(d_h+1)\times d_h}$, $\boldsymbol{W}_{\hat{h}} \in \mathcal{R}^{(d_h+1)\times d_h}$, $\boldsymbol{W}_{sm} \in \mathcal{R}^{(d_h+1)\times 4d_h}$, $\boldsymbol{W}_{ssm} \in \mathcal{R}^{4d_h\times d_h}$, and $\boldsymbol{W}_{out} \in \mathcal{R}^{2d_h\times 1}$ indicate the learnable parameters in the TGNM model.

### *4.2.2. Handling missing values in TGNM*

As mentioned above, the neuron interior of the TGNM is mainly used to capture the spatiotemporal correlations in the complete (or pre-processed) traffic flow, and does not process the missing values in the traffic flow. For the problem of missing values in traffic flow, we design a component for mining the missing patterns in traffic flow automatically and iteratively impute the missing values of traffic flow. As shown in Figure 7, the motivation for missing pattern mining component comes from two points. First, in the single-step prediction of traffic flow, the contribution of temporal correlation to prediction results is more significant than the contribution of spatial

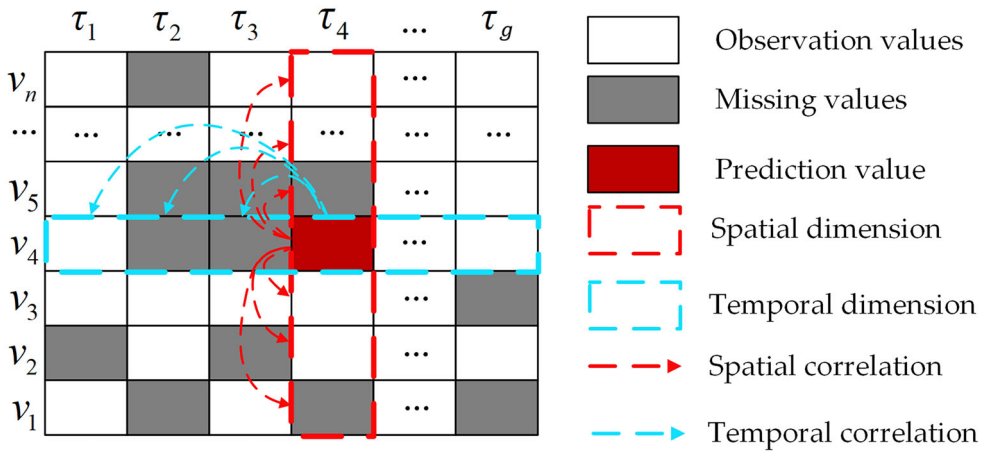


**Figure 7.** Motivation for missing pattern mining component.

correlation to prediction results. Second, with the increase of the prediction steps (i.e., multi-step prediction), the contribution of temporal correlation to the prediction results will decrease, while the contribution of spatial correlation to the prediction results will increase.

The single-step prediction and multi-step prediction correspond to random missing and block missing, respectively. Specifically, the random missing value in the traffic flow can be directly imputed by the temporal memory state $\boldsymbol{H}^{t-1:tm}$ at the previous time, while the block missing value in the traffic flow should further introduce the spatial memory state $\boldsymbol{H}^{t-1:sm}$ to improve the imputation result. To make TGNM model can identify missing patterns in traffic flow, we introduce an auxiliary quantity $\boldsymbol{c}^t = \{c_i^t\}_{i=1}^n$, and its calculation method is shown in Formula (7).

$$c_i^t = \begin{cases} \tau_t - \tau_{t-1} + c_{i-1}^t & t > 1, m_i^{t-1} = 0 \\ \tau_t - \tau_{t-1} & t > 1, m_i^{t-1} = 1 \\ 0 & t = 1 \end{cases} \tag{7}$$

where $c_i^t$ represents the time step of node $v_i$ in the t time window from the nearest observed value; when $c_i^t = 1$ and $m_i^t = 0$, the missing pattern of node $v_i$ in the $t$ th time window is likely to be random missing; when $c_i^t > 1$ and $m_i^t = 0$, the missing pattern of node $v_i$ in the $t$ th time window is likely to be block missing.

Based on $\boldsymbol{c}^t$, we further define the process for handling missing values. The greater the $\boldsymbol{c}^t$, the greater the probability of using spatial memory state to estimate missing values. The smaller the $\boldsymbol{c}^t$, the greater the probability of using the temporal memory state to estimate the missing value. Based on this principle, the calculation method that deals with missing values is shown in Formula (8).

$$\begin{cases} \boldsymbol{x}^t = \boldsymbol{m}^t \odot \boldsymbol{x}^t + (1 - \boldsymbol{m}^t) \odot (\boldsymbol{\delta}^t \odot \hat{\boldsymbol{x}}^{t:tm} + (1 - \boldsymbol{\delta}^t) \odot \hat{\boldsymbol{x}}^{t:sm}) \\ \boldsymbol{\delta}^t = \exp\left(-\max(0,\ \boldsymbol{W}_{\delta} \boldsymbol{c}^t + \boldsymbol{b}_{\delta})\right) \\ \hat{\boldsymbol{x}}^{t:tm} = \boldsymbol{H}^{t-1:tm} \boldsymbol{W}_{tm_{out}} \\ \hat{\boldsymbol{x}}^{t:sm} = \boldsymbol{H}^{t-1:sm} \boldsymbol{W}_{sm_{out}} \end{cases} \tag{8}$$

where $\hat{\boldsymbol{x}}^{t:tm} \in \mathcal{R}^{n \times 1}$ represents the prediction value of all nodes obtained by temporal

memory state; $\hat{\boldsymbol{x}}^{t:sm} \in \mathcal{R}^{n\times 1}$ represents the prediction value of all nodes obtained by spatial memory state; $\exp(\cdot)$ means exponential function; max means maximum value function; $\boldsymbol{\vartheta}^t \in \mathcal{R}^{n\times 1}$ represents the missing pattern probability calculated by $\boldsymbol{c}^t$. When $\boldsymbol{\vartheta}^t$ approaches 1, the missing pattern approaches random missing, and when $\boldsymbol{\vartheta}^t$ approaches 0, the missing pattern approaches block missing. $\boldsymbol{W}_{tm_{out}} \in \mathcal{R}^{d_h\times 1}$, $\boldsymbol{W}_{sm_{out}} \in \mathcal{R}^{d_h\times 1}$, and $\boldsymbol{W}_{\vartheta} \in \mathcal{R}^{1\times 1}$ respectively represent the learnable parameters in missing value processing.

### 4.2.3. Optimization of TGNM

In the process of model optimization, a time-dependent step $l_d$ is set to reduce the computational complexity. That is, the traffic flow in the future $\hat{\boldsymbol{x}}^{t+1}$ can be predicted by the traffic flow $\{\boldsymbol{x}^j\}_{j=t-l_d-1}^{t}$ in the previous $l_d$ time windows. Theoretically, the prediction model can be obtained by minimizing the square loss between the actual traffic state $\boldsymbol{x}^{t+1}$ and the predicted traffic state $\hat{\boldsymbol{x}}^{t+1}$. However, if we only optimize the square loss between $\boldsymbol{x}^{t+1}$ and $\hat{\boldsymbol{x}}^{t+1}$, the impact of missing values on prediction performance will be ignored. Therefore, a loss function that takes into account the impact of missing values on the prediction results is designed.

As shown in Figure 8, the loss function proposed in this study is mainly composed of three parts: the temporal-based imputation task loss, the spatial-based imputation task loss, and the prediction task loss. Among them, the prediction task loss is mainly used to ensure the accuracy of prediction results, while the imputation task loss mainly considers the influence of missing value on prediction results. The loss function proposed in this study is shown in Formula (9).

$$\mathcal{L}(\boldsymbol{W}) = \min_{\boldsymbol{W}} \left( \sum_{i=1}^{l_d} \left(\boldsymbol{x}^{t+1-i} - \hat{\boldsymbol{x}}^{t+1-i:tm}\right)^2 + \sum_{i=1}^{l_d} \left(\boldsymbol{x}^{t+1-i} - \hat{\boldsymbol{x}}^{t+1-i:sm}\right)^2 + 2l_d * \left(\boldsymbol{x}^{t+1} - \hat{\boldsymbol{x}}^{t+1}\right)^2 \right) \tag{9}$$

where $\left(\boldsymbol{x}^{t+1-i} - \hat{\boldsymbol{x}}^{t+1-i:tm}\right)^2$ represents the temporal-based imputation task loss; $\left(\boldsymbol{x}^{t+1-i} - \hat{\boldsymbol{x}}^{t+1-i:sm}\right)^2$ represents the spatial-based imputation task loss; $\left(\boldsymbol{x}^{t+1} - \hat{\boldsymbol{x}}^{t+1}\right)$ represents the prediction task loss. To guarantee the equal weight of three parts loss, we

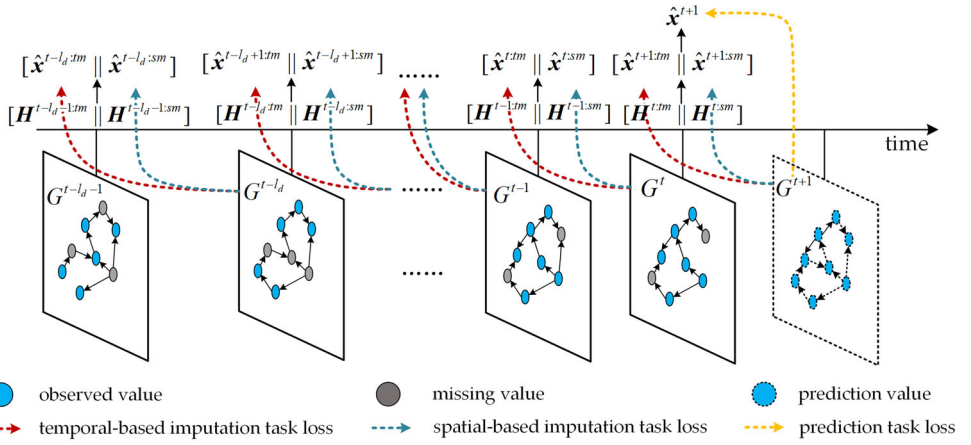


**Figure 8.** Loss function illustration of the TGNM: the loss function is mainly composed of the temporal-based imputation task loss, the spatial-based imputation task loss, and the prediction task loss.

enlarge the prediction task loss by $2l_d$ times. **Note:** In fact, it is more common to add a set of adjustable hyperparameters as weighting factors for each loss. However, the increase of hyperparameters will increase the difficulty of model calibration. To simplify the model calibration process, we set only one hyperparameter.

#### 4.2.4. Relationship between TGNM and D-TGNM

In this subsection, we further describe the relationship between TGNM and D-TGNM. As shown in Figure 9, the D-TGNM model can be regarded as a special case of the TGNM model. Specifically, the adjacency relationship between nodes in D-TGNM model is dynamically generated by Traffic BERT, while the adjacency relationship between nodes in TGNM model can be a static adjacency matrix, such as a spatial topological adjacency matrix or spatial distance matrix. In other words, the Traffic BERT model enables the D-TGNM model to capture the dynamic spatial association in the Traffic flow. In addition, although the TGNM model is a part of the D-TGNM model, the TGNM model is essentially an independent component for traffic flow prediction considering missing values. The advantages of designing the TGNM model as an independent component are as follows: when the TGNM model does not rely on Traffic BERT, the TGNM model can still run using a static spatial association matrix, such as a fixed spatial distance matrix or spatial topological adjacency matrix.

### 4.3. Training and algorithm

In sections 4.1 and 4.2, we have discussed in detail the two core components in D-TGNM, i.e., Traffic BERT and TGNM. In this section, we further introduce the training process of the D-TGNM model. The basic principle of the D-TGNM model is to establish a supervised learning model that uses the missing patterns and spatiotemporal correlation in the traffic flow to predict the future traffic state. Considering that the D-TGNM model relies on the dynamic association matrix estimated by Traffic BERT, two models will be obtained after model training, i.e., $\mathcal{M}_{\text{TrafficBERT}}$ and $\mathcal{M}_{\text{DTGNM}}$. The training process is shown in Algorithm 1. First, we build two different training instances (lines 1–6), i.e., $\mathcal{D}^{\text{TrafficBERT}}$ and $\mathcal{D}^{\text{DTGNM}}$. Then, based on $\mathcal{D}^{\text{TrafficBERT}}$, the Traffic BERT

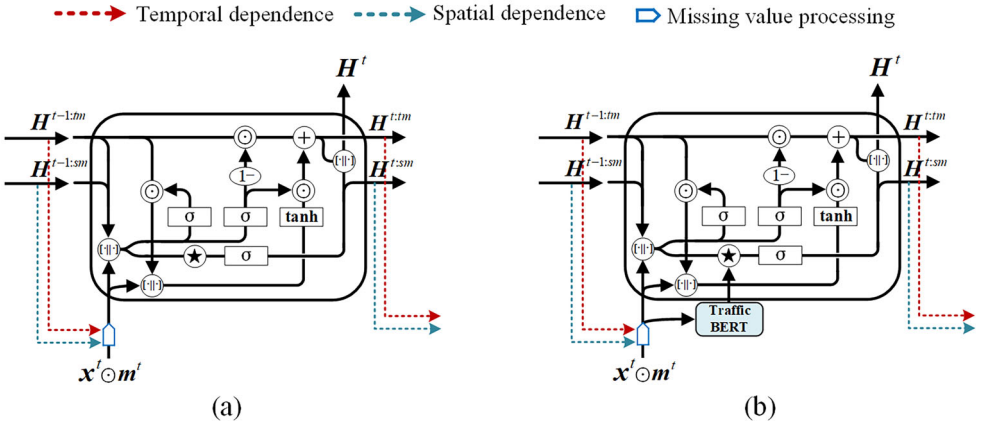


**Figure 9.** Illustration of the TGNM and D-TGNM models: (a) TGNM model, and (b) D-TGNM model.

model was trained (lines 7–12). Finally, based on $\mathcal{D}^{\text{TrafficBERT}}$ and $\mathcal{D}^{\text{DTGNM}}$, the final prediction model was obtained (lines 13–19). In the model testing stage, we only need to use the forward propagation process of D-TGNM to obtain the final prediction results.

**Algorithm 1.** Training Process of D-TGNM

**Require**: Dynamic temporal graph sequence: $\{G^t\}_{t=1}^{g}$
Number of encoders in Traffic BERT: $L$
Number of multi-headed attention in Traffic BERT: $p$
Time dependent step: $l_d$
**Ensure**: Traffic BERT model: $\mathcal{M}_{\text{TrafficBERT}}$
D-TGNM model: $\mathcal{M}_{\text{DTGNM}}$
*//construct training instances of Traffic BERT*
**1:** $\mathcal{D}^{\text{TrafficBERT}} \leftarrow \emptyset$
**2: for** each $G^t \in \{G^t\}_{t=1}^{g}$ **do**
**3:** put a training instance $\{\boldsymbol{x}^t, \boldsymbol{m}^t\}$ into $\mathcal{D}^{\text{TrafficBERT}}$ based $G^t$
*//construct training instances of D-TGNM*
**4:** $\mathcal{D}^{\text{DTGNM}} \leftarrow \emptyset$
**5: for** next $t \in [l_d,\ l_d+1,\ \ldots\ \ldots, g]$ **do**
**6:** put a training instance $\left(\{\boldsymbol{m}^j\}_{j=t-l_d-1}^{t}, \{\boldsymbol{x}^j\}_{j=t-l_d-1}^{t}, \{\boldsymbol{m}^j\}_{j=t-l_d}^{t+1}, \{\boldsymbol{x}^j\}_{j=t-l_d}^{t+1}\right)$ into $\mathcal{D}^{DTGNM}$
*//train Traffic BERT model*
**7:** initialize the parameters $\boldsymbol{W}$ of Traffic BERT
**8: repeat**
**9:** randomly select a training instance $\mathcal{D}_b^{\text{TrafficBERT}}$ from $\mathcal{D}^{\text{TrafficBERT}}$
**10:** obtain results $\hat{\boldsymbol{x}}^t$ by Formulas (2) and (3) with $L$ and $p$
**11:** find $\boldsymbol{W}$ by minimizing the Formula (4)
**12: until** $\mathcal{M}_{\text{TrafficBERT}}$ converges
*//train D-TGNM model*
**13:** initialize the parameters $\boldsymbol{W}$ of D-TGNM
**14: repeat**
**15:** randomly select a training instance $\mathcal{D}_b^{\text{DTGNM}}$ from $\mathcal{D}^{\text{DTGNM}}$
**16:** obtain dynamic matrix $\{\hat{\boldsymbol{A}}^{t-l_d-1}, \hat{\boldsymbol{A}}^{t-l_d}, \ldots\ldots, \hat{\boldsymbol{A}}^{t}\}$ by $\mathcal{M}_{\text{TrafficBERT}}$ and $\{\hat{\boldsymbol{x}}^j\}_{j=t-l_d-1}^{t}$
**17:** obtain results $\{\hat{\boldsymbol{x}}^j\}_{j=t-l_d}^{t+1}$ by Formulas (6), (7), and (8)
**18:** find $\boldsymbol{W}$ by minimizing the Formula (9)
**19: until** $\mathcal{M}_{\text{DTGNM}}$ converges
**20:** output the learned models $\mathcal{M}_{\text{TrafficBERT}}$ and $\mathcal{M}_{\text{DTGNM}}$

# 5. Experimental results and discussions

## 5.1. Data preparation

### 5.1.1. Data sources

Traffic flow data in Wuhan, China, was used to evaluate the prediction performance of the D-TGNM model. Traffic flow data was obtained by Automatic Vehicle Identification (AVI) technology, which automatically identifies spatial coordinates of vehicles through photos taken by cameras. We counted the traffic flow of a single camera at intervals of 5 min, i.e., the time window size is 5 min. The period of traffic flow data was from 1 March 2021 to 28 March 2021. Figure 10 shows the spatial distribution of the experimental cameras. In this study, a total of 71 experimental cameras were selected, i.e., we built a graph with 71 nodes (each camera represents a graph node, and the association between any two cameras is determined by Traffic BERT). Table 1 shows the traffic information counted by each camera. Each record contains the unique identification ID of the camera, the time window, the latitude and longitude of the camera, and the traffic flow in the time window.

### 5.1.2. Data preprocessing

To support the research of this work, we further preprocessed traffic volume data, and the preprocessing process is mainly as follows:

1. There are natural missing data in the raw traffic flow collected by cameras, and the traffic flow collected by different cameras may have different missing rates. Since different missing rates may affect the fairness of subsequent experiments, we imputed the natural missing values in the raw traffic flow. Referring to P. Wang et al. (2022b), the Multi-BiSTGN model achieved SOTA imputation

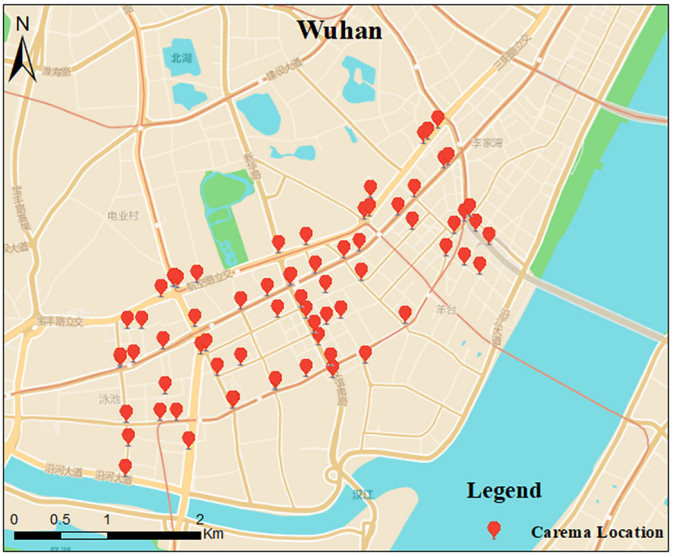


**Figure 10.** Spatial distribution of cameras.

**Table 1.** Traffic volume sample of single camera.

| Camera ID | Time window | Latitude | Longitude | Traffic volume |
|---|---|---|---|---|
| DE28HN*** | 2021-03-01 00:00 ~ 2021-03-01 00:05 | 30.6*** | 114.1*** | 26 |
| DE28HN*** | 2021-03-01 00:05 ~ 2021-03-01 00:10 | 30.6*** | 114.1*** | 18 |
| DE28HN*** | 2021-03-01 00:10 ~ 2021-03-01 00:15 | 30.6*** | 114.1*** | 18 |
| … … | … … | … … | … … | … … |
| DE28HN*** | 2021-03-28 23:55 ~ 2021-03-29 00:00 | 30.6*** | 114.1*** | 13 |

***means the content is omitted.

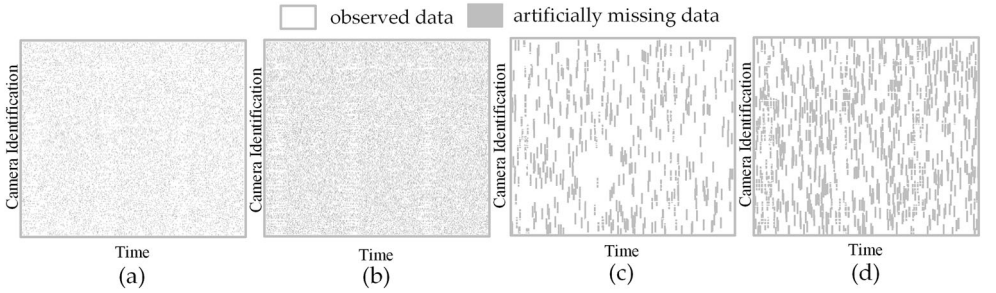


**Figure 11.** Traffic volume information after preprocessing: (a) 15% random missing, (b) 30% random missing, (c) 30% block missing, and (d) 30% block missing.

performance compared with existing multiple models. Therefore, we use the Multi-BiSTGN model to impute natural missing values in the raw traffic flow.

2. Referring to Cui *et al*. (2020), Tian *et al*. (2018), and Yang *et al*. (2021), based on two missing types (random missing and block missing), partial traffic state is deleted at 15% and 30% missing rates by artificial experience. Figure 11 shows the traffic flow information for three days after manual processing, and the details of the missing traffic flow are described in Supplementary Appendix A.
3. The data processed manually were divided into training samples and test samples. According to the 20–80 criterion, the training samples account for 80%, and the test samples account for 20%.

### *5.2. Evaluation metrics*

In traffic flow forecasting, a key issue is how to evaluate the performance of the forecasting model. In this study, the Mean Absolute Error (MAE), Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE) were used as quantitative indicators to verify the prediction accuracy of the proposed model (S. Cheng *et al*. 2021). The calculation methods of MAE, RMSE, and MAPE are shown in Formulas (10), (11), and (12).

$$MAE = \frac{1}{n*\Delta t}\sum_{t=1}^{\Delta t}\sum_{i=1}^{n}\left|x_i^t - \hat{x}_i^t\right| \tag{10}$$

$$RMSE = \sqrt{\frac{1}{n*\Delta t}\sum_{t=1}^{\Delta t}\sum_{i=1}^{n}\left(x_i^t - \hat{x}_i^t\right)^2} \tag{11}$$

$$MAPE = \frac{100\%}{n*\Delta t}\sum_{t=1}^{\Delta t}\sum_{i=1}^{n}\left|\frac{x_i^t - \hat{x}_i^t}{x_i^t}\right| \tag{12}$$

where $x_i^t$ represents the ground truth of node $v_i$ in the $t$ th time window in the future; $\hat{x}_i^t$ represents the predicted traffic state of node $v_i$ in the $t$ th time window in the future; $n$ represents the total number of nodes in the graph; $\Delta t$ represents the prediction step.

### 5.3. Estimation of hyper-parameters

In this study, the historical traffic flow is processed on a PC (CPU: Intel(R) Xeon(R) E-2224G @ 3.50 GHz, memory: 16.0GB). Moreover, we built our model based on PyTorch and Python3.7 on a Graphics Processing Unit (GPU) platform with 24GB of GPU memory.

The hyper-parameters of the D-TGNM model mainly include the number of encoders $L$ and the number of multi-head attention $p$ in the Traffic BERT component, and the time-dependent step $l_d$ in the TGNM component. As Traffic BERT and TGNM are two independent components, we only calibrate the hyper-parameters of one single component at a time. In the process of model training, the control variable method is used to obtain the optimal combination of hyperparameters, where the value range of $L$ is [1,8], the value range of $p$ is [1, 7], and the value range of $l_d$ is [1,10]. Figure 12 shows the process of parameter calibration in the missing scenario. The results show that with the increase of $L$ and $p$, the MAE of Traffic BERT decreases first and then stabilizes. When $L$=6 and $p$=4, the Traffic BERT component achieves better accuracy. Similarly, with the increase of $l_d$, the MAE of TGNM also decreases first and then stabilizes. When $l_d$=7, the TGNM component obtains a reasonable accuracy.

### 5.4. Comparison with baselines

Classical statistical methods always perform worse than data driven methods on many traffic forecasting tasks, due to their inability to handle complex spatiotemporal information (Fang *et al.* 2021, Yi *et al.* 2021). We directly compare the D-TGNM with popular data-driven methods. In this study, the D-STGM model was compared with ten baseline methods. The baseline methods can be roughly divided into four categories. The first category is the ST-KNN model (S. Wu *et al.* 2014, B. Yu *et al.* 2016a), which is

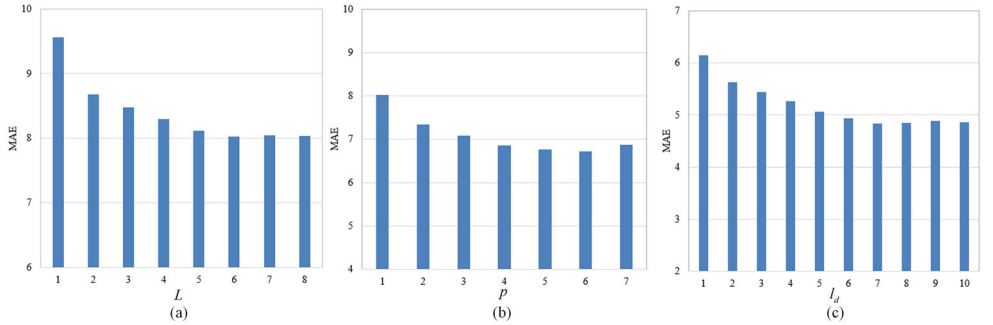


**Figure 12.** Impact of parameters $L$, $p$ and $l_d$ on the prediction performance.

regarded as a shallow machine learning method that does not consider missing values. The second category includes the TCN (Bai *et al.* 2018), T-GCN (Zhao *et al.* 2020), and ST-GCN (B. Yu *et al.* 2018) models, which are regarded as deep learning models that do not consider missing values. The third category includes the TRMF (H.-F. Yu *et al.* 2016b), BTMF (Chen and Sun 2022), and BTTF (Chen and Sun 2022) methods, which are regarded as shallow machine learning models that consider missing values. The fourth category includes the GRU-D (Che *et al.* 2018), LSTM-M (Tian *et al.* 2018), and SGMN (Cui *et al.* 2020) methods which are regarded as deep learning models that consider missing values.

### 5.4.1. Comparison results on complete data

In this section, we compare the prediction results of the D-TGNM model and baseline methods on the complete data. The complete data represents the dataset after the natural missing value is imputed, and the comparison results are shown in Table 2. On the complete data, the difference in prediction accuracy between the four categories of models is relatively small, and the prediction accuracy of the deep learning models is slightly higher than that of the machine learning models. In addition, in the deep learning models, the prediction performance of graph-based spatiotemporal prediction models such as T-GCN, ST-GCN, and SGMN is lower than that of time prediction models such as TCN, GRU-D, and LSTM-M. There are two main reasons for the above inconsistent results with common sense. First, T-GCN, ST-GCN, and SGMN models are graph convolution networks based on static graph structure. When the static graph structure is challenging to describe the spatial relationship of traffic flow, the graph convolution network based on static graph structure may introduce more errors, making the prediction performance of the prediction model lower than that of the simple time graph network model. Second, T-GCN, ST-GCN, and SGMN models integrate graph convolution in series, making it more challenging to mine the nonlinear relationship in traffic flow data. Compared with the baseline methods, the D-TGNM model not only considers the dynamic spatial relationship in traffic flow data, but also integrates graph convolution in parallel. Therefore, the D-TGNM model achieves the highest prediction performance on the complete data.

### 5.4.2. Comparison results on incomplete data

In this section, we further compare the prediction results of the D-TGNM model and baseline methods on the incomplete data. The incomplete data represents the dataset after the complete data is artificially missing. Due to space limitations, we only show

**Table 2.** Comparison results (in MAE/RMSE/MAPE) of D-TGNM and baseline methods without missing data.

| Models | 1-step (5-min) | 3-steps (15-min) | 5- steps (25-min) | 7-steps (35-min) |
|---|---|---|---|---|
| TCN | 4.85/7.82/20.29% | 5.38/9.05/21.68% | 5.85/10.04/23.24% | 6.41/11.32/24.71% |
| ST-KNN | 5.92/10.11/25.84% | 6.18/10.70/26.93% | 6.53/11.35/28.52% | 6.86/11.94/30.37% |
| T-GCN | 6.65/11.86/32.18% | 7.21/12.92/34.16% | 8.12/14.94/36.44% | 8.73/16.38/38.86% |
| ST-GCN | 6.99/12.78/33.14% | 7.96/14.65/36.33% | 8.73/16.38/38.86% | 8.96/16.46/39.52% |
| TRMF | 5.49/8.85/25.11% | 6.77/10.97/29.43% | 6.79/11.19/29.43% | 6.91/11.24/29.63% |
| BTMF | 5.34/8.52/27.53% | 5.38/8.84/27.68% | 5.68/9.24/28.28% | 6.84/10.90/31.80% |
| BTTF | 5.32/8.46/28.17% | 5.53/9.01/30.11% | 5.96/9.82/29.16% | 6.31/10.63/32.37% |
| GRU-D | 4.83/7.84/26.08% | 5.34/8.99/27.70% | 5.81/9.94/29.48% | 6.36/11.19/31.21% |
| LSTM-M | 4.86/7.93/27.00% | 5.35/9.02/28.69% | 5.81/9.91/30.64% | 6.38/11.17/32.69% |
| SGMN | 5.38/8.51/24.85% | 5.84/9.57/26.26% | 6.22/10.31/27.89% | 6.70/11.43/29.41% |
| **D-TGNM** | **4.62/7.49/19.81%** | **5.08/8.57/21.48%** | **5.46/9.25/23.18%** | **5.99/10.39/24.87%** |

Bold indicates best results.

**Table 3.** Comparison results (in MAE/RMSE/MAPE) of D-TGNM and baseline methods on random missing data.

| | Missing rate: 15% | | Missing rate: 30% | |
|---|---|---|---|---|
| Models | 1-step (5-min) | 3-steps (15-min) | 1-step (5-min) | 3-steps (15-min) |
| TCN | 7.89/12.23/101.52% | 9.29/13.66/132.54% | 9.32/15.45/107.86% | 10.65/16.32/140.2% |
| ST-KNN | 10.97/19.95/33.24% | 11.69/20.84/33.40% | 16.80/29.57/42.13% | 17.07/30.82/42.70% |
| T-GCN | 7.32/13.17/27.38% | 8.87/16.72/31.60% | 9.15/16.89/32.10% | 10.39/19.58/37.44% |
| ST-GCN | 7.43/13.37/34.76% | 8.73/16.38/38.86% | 8.96/16.46/39.52% | 10.16/19.25/45.21% |
| TRMF | 5.67/9.23/24.95% | 7.35/12.04/30.31% | 6.01/9.95/26.26% | 8.39/13.87/32.06% |
| BTMF | 5.49/8.60/26.33% | 6.39/10.55/30.84% | 5.60/9.11/27.27% | 6.24/9.88/28.35% |
| BTTF | 5.41/8.52/25.64% | 5.70/9.28/29.06% | 5.47/8.76/26.80% | 5.89/9.58/29.10% |
| GRU-D | 5.46/11.70/21.75% | 5.91/12.38/23.70% | 5.65/11.93/23.67% | 6.11/12.60/25.75% |
| LSTM-M | 5.47/11.64/21.29% | 5.90/12.27/23.10% | 5.68/11.89/22.52% | 6.12/12.53/24.19% |
| SGMN | 5.99/9.18/33.54% | 6.41/10.12/34.86% | 6.07/9.49/33.16% | 6.47/10.41/34.41% |
| **D-TGNM** | **4.84/7.99/20.25%** | **5.29/8.98/21.80%** | **5.06/8.42/21.27%** | **5.52/9.42/22.88%** |

Bold indicates best results.

**Table 4.** Comparison results (in MAE/RMSE/MAPE) of D-TGNM and baseline methods on block missing data.

| | Missing rate: 15% | | Missing rate: 30% | |
|---|---|---|---|---|
| Models | 1-step (5-min) | 3-steps (15-min) | 1-step (5-min) | 3-steps (15-min) |
| TCN | 9.78/20.17/103.15% | 10.82/19.88/133.4% | 13.11/28.43/109.5% | 14.07/27.99/140.8% |
| ST-KNN | 14.32/27.27/41.59% | 15.62/30.28/43.00% | 22.17/39.32/54.82% | 24.02/42.62/56.61% |
| T-GCN | 9.43/17.07/43.25% | 11.34/25.38/49.34% | 12.61/23.62/44.19% | 13.51/25.12/49.29% |
| ST-GCN | 9.27/16.85/41.25% | 11.18/24.12/47.13% | 12.36/23.11/52.92% | 13.17/24.55/58.06% |
| TRMF | 6.84/10.73/30.50% | 8.18/13.35/32.69% | 6.55/11.55/28.93% | 8.60/14.13/32.16% |
| BTMF | 5.93/10.02/26.66% | 7.30/11.82/36.15% | 6.79/10.55/31.12% | 7.77/13.15/36.99% |
| BTTF | 5.67/9.01/26.14% | 5.86/9.28/28.60% | 6.34/10.24/29.31% | 7.12/11.26/33.16% |
| GRU-D | 5.84/12.31/24.10% | 6.30/12.98/26.40% | 6.37/13.29/27.46% | 6.89/14.05/30.39% |
| LSTM-M | 5.87/12.30/23.21% | 6.35/12.98/25.33% | 6.45/13.39/25.86% | 7.01/14.17/28.22% |
| SGMN | 7.26/12.45/41.43% | 7.61/12.97/42.81% | 8.96/17.32/46.68% | 9.32/17.86/47.77% |
| **D-TGNM** | **5.24/9.02/22.68%** | **5.71/9.99/24.44%** | **5.95/11.00/26.03%** | **6.45/11.88/27.85%** |

Bold indicates best results.

the comparison results of one-step prediction and three-step prediction. Tables 3 and 4 show the comparison results of the prediction performance under the missing scenario. When data missing happens, the prediction models such as TCN, ST-KNN, T-GCN, and ST-GCN that do not consider missing values have poor prediction performance, while TRMF, BTMF, BTTF, GRU-D, LSTN-M, and SGMN that consider missing values have better prediction performance. In addition, in different missing data scenarios, the prediction performance of the BTTF, GRU-D, and LSTN-M models is relatively stable, while the prediction performance of the TRMF, BTMF, and SGMN models is greatly affected by the missing data scenarios. Compared with the TRMF, BTMF, and SGMN models, the D-TGNM model also has stable prediction performance and the highest prediction results. Overall, the D-TGNM model has obvious advantages compared to the baseline methods.

### *5.5. Qualitative analysis of prediction results*

In addition to quantitative analysis, in this section, scatter plots are used to describe the prediction performance of the D-TGNM model qualitatively. Figure 13 visually shows the difference between the prediction and actual values under the four missing

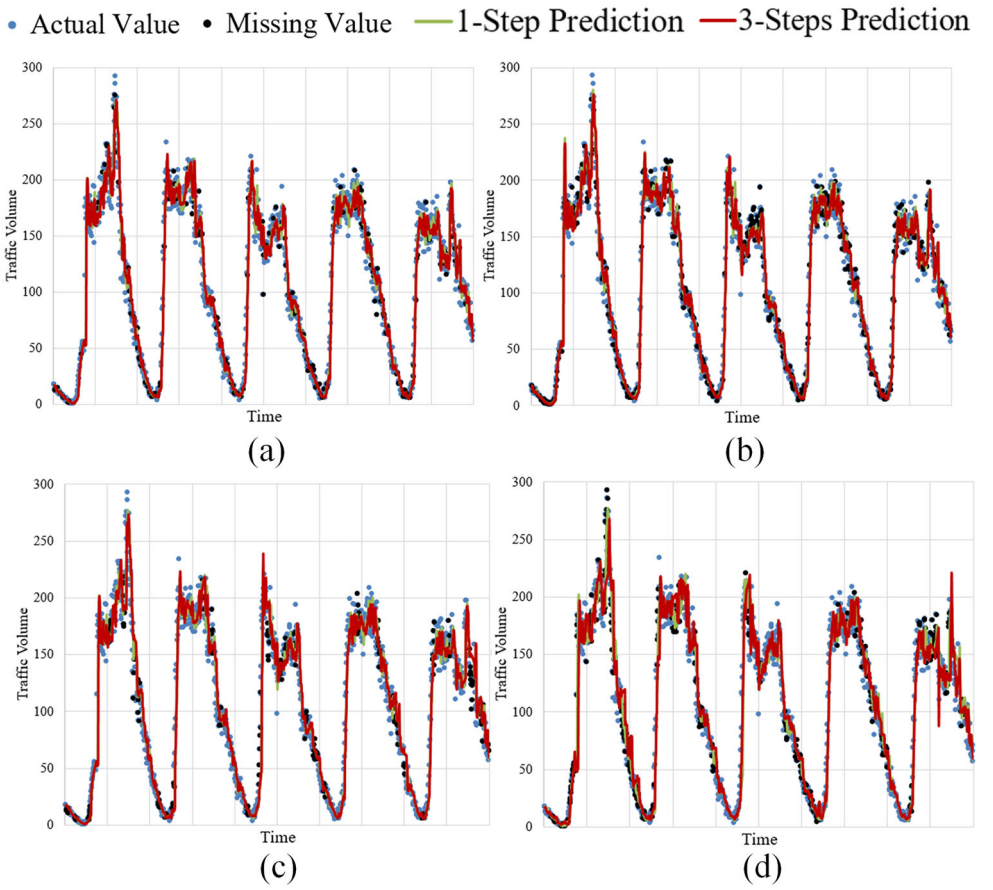


**Figure 13.** Difference between prediction values and actual values: (a) 15% random missing, (b) 30% random missing, (c) 15% block missing, and (d) 30% block missing.

**Table 5.** Impact of dynamic association matrix on prediction results (in MAE/RMSE/MAPE).

| | Missing rate: 15% | | Missing rate: 30% | |
|---|---|---|---|---|
| Models | 1-step (5-min) | 3-steps (15-min) | 1-step (5-min) | 3-steps (15-min) |
| TGNM (R) | 5.09/8.80/23.69% | 5.59/9.74/26.12% | 5.30/9.20/24.67% | 5.78/10.09/26.66% |
| TGNM (B) | 5.58/9.87/26.59% | 6.07/10.74/29.04% | 6.24/11.21/32.16% | 6.75/12.01/34.77% |
| **D-TGNM (R)** | **4.84/7.99/20.25%** | **5.29/8.98/21.80%** | **5.06/8.42/21.27%** | **5.52/9.42/22.88%** |
| **D-TGNM (B)** | **5.24/9.02/22.68%** | **5.71/9.99/24.44%** | **5.95/11.00/26.03%** | **6.45/11.88/27.85%** |

Bold indicates best results.

data scenarios. Combined with the quantitative analysis in Tables 3 and 4, it can be found that although there are differences in the prediction performance under the four different missing data scenarios, the actual values are always close to the prediction values, i.e., the D-TGNM model can accurately predict changes in traffic flow over time. The results further prove that the D-TGNM model still has good predictive performance under missing scenarios.

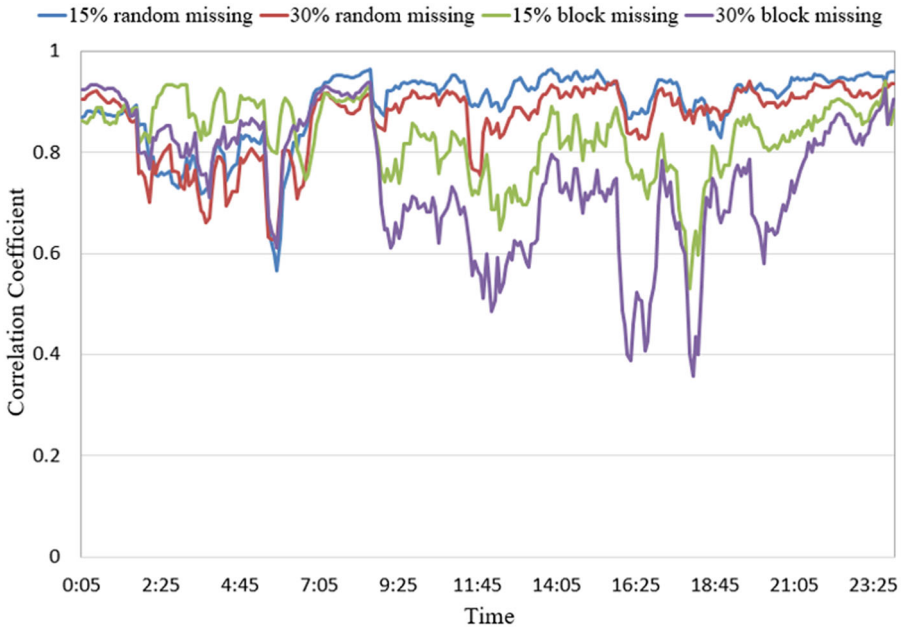


**Figure 14.** Correlation coefficient of the dynamic spatial relationship estimated from the missing data and the non-missing data.

**Table 6.** Impact of loss function on prediction results (in MAE/RMSE/MAPE).

| | Missing rate: 15% | | Missing rate: 30% | |
|---|---|---|---|---|
| Models | 1-step (5-min) | 3-steps (15-min) | 1-step (5-min) | 3-steps (15-min) |
| D-TGNM-I (R) | 5.07/9.25/20.88% | 5.50/10.05/22.57% | 5.29/9.61/23.50% | 5.72/10.41/25.50% |
| D-TGNM-I (B) | 5.55/10.18/25.76% | 6.02/11.01/28.39% | 6.45/12.29/34.71% | 6.97/13.09/38.49% |
| D-TGNM-P (R) | 7.36/18.92/33.75% | 7.88/19.37/39.13% | 7.92/19.27/44.63% | 8.45/19.74/49.78% |
| D-TGNM-P (B) | 8.07/19.73/37.67% | 8.65/20.27/43.46% | 9.38/21.39/48.60% | 9.97/22.00/53.32% |
| **D-TGNM (R)** | **4.84/7.99/20.25%** | **5.29/8.98/21.80%** | **5.06/8.42/21.27%** | **5.52/9.42/22.88%** |
| **D-TGNM (B)** | **5.24/9.02/22.68%** | **5.71/9.99/24.44%** | **5.95/11.00/26.03%** | **6.45/11.88/27.85%** |

D-TGNM-I represents the model optimized only by imputation task loss, D-TGNM-P represents the model optimized only by predicting task loss, (R) represents the random missing data, and (B) represents the block missing data. Bold indicates best results.

### *5.6. Impact of dynamic association matrix*

In the D-TGNM model, we use the dynamic matrix instead of the static matrix to describe the implicit spatial association in traffic flow. Therefore, we analyze the impact of the dynamic association matrix on the prediction performance. Table 5 shows the impact of the dynamic spatial matrix on prediction performance, where TGNM represents the model using the static spatial matrix, (R) represents the random missing data, and (B) represents the block missing data. The results show that the introduction of the dynamic spatial matrix improves the prediction ability of the model. Figure 14 further shows the correlation coefficient of the dynamic spatial association estimated from the missing and non-missing data. The results show that the spatial associations estimated at different times change dynamically with time, proving the rationality of modeling dynamic spatial associations in the proposed method. In addition, the results show that the ability of Traffic BERT to identify spatial associations

declines under block missing, which is also the reason that explains the low prediction performance of the model under block missing.

### 5.7. Impact of loss function

In the optimization process, the loss function consists of imputation task loss and predicting task loss. Therefore, in this section, we analyze the impact of the loss function on the prediction performance of the model. Table 6 shows the impact of the loss function on the prediction results. The results show that the prediction performance of the D-TGNM-P model is lower than that of the D-TGNM-I model, and the prediction performance of the D-TGNM-I model is lower than that of D-TGNM. That is, the D-TGNM model integrating multiple loss tasks has the optimal prediction ability, which further proves the necessity of introducing multiple loss tasks. In addition, the results show that if the impact of missing values on the prediction results is not considered, i.e., only the loss of the prediction task is optimized, the worst prediction result will be obtained.

### 5.8. Analysis of incremental learning

In real-world applications, the spatial structure of sensors on urban road networks is not static. In general, the number of sensors on urban road networks tends to increase gradually over time. When the number of sensors increases, the GCNs based on predefined matrix often need to be re-trained, which greatly wastes the computing resources. Compared with GCNs based on predefined matrix, the D-TGNM model does not need to be re-trained, but can be fine-tuned to regain the optimal prediction ability. This is, the D-TGNM model has the ability of incremental learning. Figure 15 shows the results of incremental learning for the D-TGNM model. As shown in Figure 15(a), the *Whole Study Area* is divided into *Study Area A* and *Study Area A*. Fine-tune means that the optimal model in *Study Area A* is used to learn the traffic flow pattern of the *Whole Study Area*, and re-train means that an initial model is used to directly learn the traffic flow pattern of the *Whole Study Area*. Figure 15(b,c) show the results of fine-tune and re-train the Traffic BERT and TGNM components, respectively. The result proves that the D-TGNM model has good incremental learning ability.

## 6. Conclusions and future work

In this study, a dynamic temporal graph network considering missing values (D-TGNM) was proposed for traffic flow prediction under missing scenarios. In the experimental

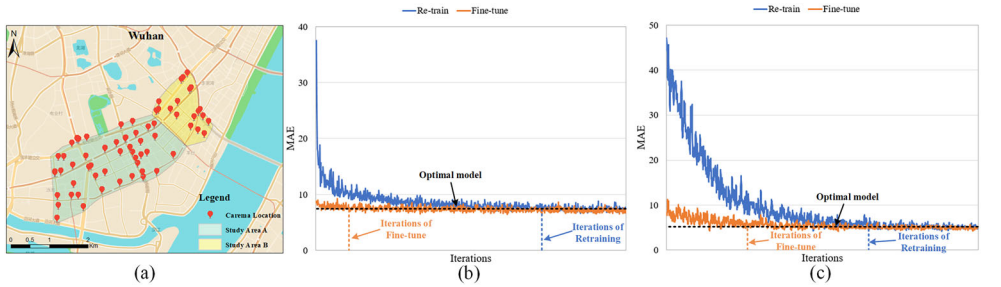


**Figure 15.** Illustration of incremental learning of D-TGNM model: (a) regional division, (b) Traffic BERT component, and (c) TGNM component.

section, the actual traffic dataset collected in Wuhan, China, was used to verify the prediction performance of D-TGNM. Experimental results showed that D-TGNM still had good prediction results under four missing data scenarios (15% random missing, 15% block missing, 30% random missing, and 30% block missing), and outperformed ten existing baselines. Second, we tested the impact of dynamic matrix and loss function on the prediction accuracy of D-TGNM, further proving that the proposed method is suitable for traffic flow prediction with missing values.

The limitations of this study are as follows: (1) We only verified the prediction performance of the D-TGNM model under two missing rates (15% and 30%), and did not analyze the upper and lower bounds of the missing rate when the model performance was within an acceptable range; (2) The D-TGNM model needs to calculate the dynamic matrix for each forward propagation, which is computationally expensive; (3) The D-TGNM model is essentially a general prediction model considering missing values, but we only use traffic datasets to verify the imputation performance of the proposed model. Given the above problems, future work will focus on two aspects. First, we will try to determine the missing rate bounds when the D-TGNM model performance is within an acceptable range. Second, the dynamic matrix in the time window can be calculated in advance to reduce the computational complexity of the model. Finally, multi-source data, such as air quality data and meteorological data, will be further collected to improve the imputation performance of the D-TGNM model.

## Acknowledgments

The numerical calculations in this paper have been done on the supercomputing system in the Supercomputing Center of Wuhan University.

## Disclosure statement

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

## 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.19642575.

## 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].

## 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, spatiotemporal prediction, social computing, and public health.

***Yan Zhang*** is a PhD candidate of State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University. He is also a joint PhD candidate at School of Design and Environment, National University of Singapore. His research focus on collaborative sensing and geographic knowledge services.

***Tao Hu*** is an Assistant Professor in Department of Geography at Oklahoma State University (OSU). 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.

***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.

## ORCID

Peixiao Wang http://orcid.org/0000-0002-1209-6340
Yan Zhang http://orcid.org/0000-0002-2059-4171
Tao Hu http://orcid.org/0000-0002-8557-8017
Tong Zhang http://orcid.org/0000-0002-0683-4669

## References

Asif, M.T., *et al.* 2016. Matrix and tensor based methods for missing data estimation in large traffic networks. *IEEE Transactions on Intelligent Transportation Systems*, 17 (7), 1816–1825.

Bai, S., Kolter, J.Z., and Koltun, V., 2018. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *ArXiv:1803*.01271 [Cs]. http://arxiv.org/abs/1803.01271

Bao, H., Dong, L., and Wei, F., 2021. BEiT: BERT pre-training of image transformers. *ArXiv:2106*.08254 [Cs]. http://arxiv.org/abs/2106.08254

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.

Che, Z., *et al.* 2018. Recurrent neural networks for multivariate time series with missing values. *Scientific Reports*, 8 (1), 6085.

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., and Sun, L., 2022. Bayesian temporal factorization for multidimensional time series prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44 (9), 4659–4673.

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., *et al.* 2021. Short-term traffic forecasting by mining the non-stationarity of spatiotemporal patterns. *IEEE Transactions on Intelligent Transportation Systems*, 22 (10), 6365–6383.

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.

Chung, J., *et al.* 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *ArXiv:1412*.3555 [Cs]. http://arxiv.org/abs/1412.3555

Cui, Z., *et al.* 2020. Graph Markov network for traffic forecasting with missing data. *Transportation Research Part C: Emerging Technologies*, 117, 102671.

Devlin, J., *et al.* 2019. BERT: pre-training of deep bidirectional transformers for language understanding. *ArXiv:1810*.04805 [Cs]. http://arxiv.org/abs/1810.04805

Diao, Z., *et al.* 2019. Dynamic spatial-temporal graph convolutional neural networks for traffic forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33 (01), 890–897.

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

Fang, Z., *et al.* 2021. Spatial-temporal graph ODE networks for traffic flow forecasting. *In*: *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, 364–373.

Furtlehner, C., *et al.* 2022. Short-term forecasting of urban traffic using spatio-temporal Markov field. *IEEE Transactions on Intelligent Transportation Systems*, 23 (8), 10858–10867.

Ge, L., *et al.* 2019. Traffic speed prediction with missing data based on TGCN. *In*: *2019 IEEE SmartWorld, ubiquitous intelligence computing, advanced trusted computing, scalable computing communications, cloud big data computing, internet of people and smart city innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, 522–529.

Kipf, T.N., and Welling, M., 2017. Semi-supervised classification with graph convolutional networks. *In*: *International conference on learning representations*. http://arxiv.org/abs/1609.02907

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, Q., *et al.* 2020. Traffic flow prediction with missing data imputed by tensor completion methods. *IEEE Access.*, 8, 63188–63201.

Lin, C., *et al.* 2020. Spatiotemporal congestion-aware path planning toward intelligent transportation systems in software-defined smart city IoT. *IEEE Internet of Things Journal*, 7 (9), 8012–8024.

Liu, A., *et al.* 2018. Real-time traffic prediction: a novel imputation optimization algorithm with missing data. *In*: *2018 IEEE Global Communications Conference (GLOBECOM)*, 1–7.

Medrano, R. d., and Aznarte, J.L., 2021. On the inclusion of spatial information for spatio-temporal neural networks. *Neural Computing and Applications*, 33 (21), 14723–14740.

Ren, Y., *et al.* 2020. A hybrid integrated deep learning model for the prediction of citywide spatio-temporal flow volumes. *International Journal of Geographical Information Science*, 34 (4), 802–823.

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

Shi, Y., *et al.* 2021. Detecting spatiotemporal extents of traffic congestion: a density-based moving object clustering approach. *International Journal of Geographical Information Science*, 35 (7), 1449–1473.

Tian, Y., *et al.* 2018. LSTM-based traffic flow prediction with missing data. *Neurocomputing*, 318, 297–305.

Wang, P., Hao, W., and Jin, Y., 2021. Fine-grained traffic flow prediction of various vehicle types via fusion of multisource data and deep learning approaches. *IEEE Transactions on Intelligent Transportation Systems*, 22 (11), 6921–6930.

Wang, P., *et al.* 2022a. A hybrid data-driven framework for spatiotemporal traffic flow data imputation. *IEEE Internet of Things Journal*, 9 (17), 16343–16352.

Wang, P., *et al.* 2022b. A multi-view bidirectional spatiotemporal graph network for urban traffic flow imputation. *International Journal of Geographical Information Science*, 36 (6), 1231–1257.
Wang, Y., *et al.* 2019. Optimizing the spatial relocation of hospitals to reduce urban traffic congestion: a case study of Beijing. *Transactions in GIS*, 23 (2), 365–386.
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, D., *et al.* 2020. GE-GAN: a novel deep learning framework for road traffic state estimation. *Transportation Research Part C: Emerging Technologies*, 117, 102635.
Xu, M., *et al.* 2021. Spatial-temporal transformer networks for traffic flow forecasting. *ArXiv: 2001*.02908 [*Cs, Eess]*. http://arxiv.org/abs/2001.02908
Yang, J.-M., Peng, Z.-R., and Lin, L., 2021. Real-time spatiotemporal prediction and imputation of traffic status based on LSTM and Graph Laplacian regularized matrix factorization. *Transportation Research Part C: Emerging Technologies*, 129, 103228.
Yi, Z., *et al.* 2021. Inferencing hourly traffic volume using data-driven machine learning and graph theory. *Computers, Environment and Urban Systems*, 85, 101548.
Yu, B., *et al.* 2016a. 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. *In*: *Proceedings of the twenty-seventh international joint conference on artificial intelligence*, 3634–3640.
Yu, H.-F., Rao, N., and Dhillon, I.S., 2016b. Temporal regularized matrix factorization for high-dimensional time series prediction. *In*: *30th Conference on neural information processing systems (NIPS 2016)*, 15.
Yu, J., *et al.* 2020. Urban network-wide traffic speed estimation with massive ride-sourcing GPS traces. *Transportation Research Part C: Emerging Technologies*, 112, 136–152.
Zhang, J., Zheng, Y., and Qi, D., 2017. Deep spatio-temporal residual networks for citywide crowd flows prediction. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31 (1), 10735.
Zhang, K., *et al.* 2021a. Graph attention temporal convolutional network for traffic speed forecasting on road networks. *Transportmetrica B: Transport Dynamics*, 9 (1), 153–171.
Zhang, S., *et al.* 2022. A graph-based temporal attention framework for multi-sensor traffic flow forecasting. *IEEE Transactions on Intelligent Transportation Systems*, 23 (7), 7743–7758.
Zhang, Y., *et al.* 2021b. Extracting the location of flooding events in urban systems and analyzing the semantic risk using social sensing data. *Journal of Hydrology*, 603, 127053.
Zhang, Y., Zhang, F., and Cheng, N., 2022. Migratable urban street scene sensing method based on vision language pre-trained model. *International Journal of Applied Earth Observation and Geoinformation*, 113 (9), 102989.
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.
Zhong, W., *et al.* 2021. Heterogeneous spatio-temporal graph convolution network for traffic forecasting with missing values. In: *2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS)*, 707–717.
Zhou, T., *et al.* 2022. An attention-based deep learning model for citywide traffic flow forecasting. *International Journal of Digital Earth*, 15 (1), 323–344.
Zhou, F., *et al.* 2021a. Urban flow prediction with spatial–temporal neural ODEs. *Transportation Research Part C: Emerging Technologies*, 124, 102912.
Zhou, J., *et al.* 2021b. Graph neural networks: a review of methods and applications. *AI Open*. https://arxiv.org/abs/1812.08434v5