**Traffic Network Imputation and Congestion Recognition via Attentive Graph Neural Processes**

Meng Xu1, Hongxing Ding1, Yining Di1,\*

1. Department of Civil and Environmental Engineering, Hong Kong University of Science and Technology, Hong Kong, China

\* Corresponding Author

**Abstract:**

The data-driven Intelligent Transportation System (ITS) provides great support to travel decisions and system management but inevitably encounters the issue of data missing. Hence, network imputation can be critical to recognizing the whole traffic state in a transportation network. Abundant research works adopt numerous deep learning approaches for traffic prediction and imputation. However, previous methods ignore the reliability analysis of the predicted/imputed traffic information. Thus, this study proposes an attentive graph neural process (AGNP) method for short-term traffic network imputation by considering reliability. Firstly, the gaussian process (GP) is used to model the changes in the observed traffic state on arterial highways. Such a stochastic process is further learned by the proposed AGNP method. The proposed AGNP can finally indicate the congestion state on the remaining road segments. Data from a transportation network in Anhui Province, China, is used to test the AGNP. The results show that the proposed AGNP model can impute traffic networks and predict traffic congestion with high-level performance.

*Keywords:* traffic network imputation; neural processes; congestion prediction; graph neural networks

# **1. Introduction**

With advances in the Intelligent Transportation System (ITS), traffic information can be quickly collected, analyzed, and transmitted, supporting travel decisions and systems management (Sumalee and Ho, 2018). The ITS improves traffic mobility by incorporating various technological devices into vehicles and road infrastructure (El Hamdani et al., 2020). However, the data-driven ITS suffers from the problem of missing data (Kaur et al., 2022). Due to a tight budget or privacy concerns, limited coverage of monitoring systems in a transportation network would lead to missing data issues on some road segments; the occasional sensor failure can also exaggerate the problem. Missing patterns can be divided into three categories with respect to randomness. Random missing is the case where missing values are independent of each other; temporally/spatially correlated missing is the case where missing values are temporally/spatially correlated; when missing values have both temporal and spatial correlation, it is block missing (Li et al., 2019). The latter two cases are the main research targets of traffic data imputation works.

Traffic data imputation thus is critical to enhancing the prediction performance of ITSs (Zhang et al., 2021) based on temporal or/and spatial patterns. Temporal-pattern-based imputation relies on temporal information and regards traffic data as time series. For example, the autoregressive integrated moving average (ARIMA) model assumes that the imputed data are linearly dependent on the prevailing data (Smith et al., 2002). Chan et al. (2021) imputed the missing information by a weighted combination of short-term and long-term historical data, and incorporated spatiotemporal traffic information into a deep learning neural network for traffic congestion prediction. Since the imputation technique from the temporal view only focuses on the correlation among timestamps but fails to use the correlation among sensors (Li et al., 2019), spatial information is further incorporated into data imputation models to improve the imputation accuracy (Liang et al., 2021). Li et al. (2013) found that spatiotemporal dependencies can reduce imputing errors in both probabilistic principal component analysis (PPCA) and kernel probabilistic principal component analysis (KPPCA) methods. Moreover, the KPPCA-based imputation method performs better when spatiotemporal dependence is nonlinear. However, the matrix-based imputation methods (e.g., PPCA and KPPCA) may fail to solve the problem with the large missing ratio (Tan et al., 2013), which is often the case in many traffic datasets. Tan et al. (2013) formulated traffic data in a tensor pattern to cover all spatial-temporal information and multi-mode correlations, and further applied a weighted optimization algorithm based on tensor decomposition to impute missing values. Duan et al. (2016) adopted a deep neural network to represent spatiotemporal features and correlations in traffic data, and showed that the hidden nodes in the stacked layers can be trained to detect missing data points in a data vector.

Some studies incorporate data imputation and traffic prediction into an integrated task (Cui et al., 2020b; Tian et al., 2018; Yang et al., 2021; Zhang et al., 2021). Cui et al. (2020b) designed a graph Markov network for traffic prediction with missing values. The traffic state transition process is regarded as a Markov process in a topological network. Yang et al. (2021) developed a spatiotemporal data imputation and traffic prediction framework; the long short-term memory (LSTM) network is utilized to capture temporal dependencies, and the graph Laplacian (GL) captures spatial dependence. Zhang et al. (2021) proposed a graph convolutional bidirectional recurrent neural network to simultaneously address data imputation and traffic prediction problems; by passing spatiotemporal messages and topology information in the road network, missing traffic data was estimated.

Since a graph structure can effectively model the topology and spatial correlations among road links, advanced spatiotemporal relationships can be regressed by graph neural networks (de Medrano and Aznarte, 2020; Salamanis et al., 2016). Hence, the graph-based deep learning method has been widely used for traffic prediction and congestion recognition (Cui et al., 2020a; Luan et al., 2022; Peng et al., 2021; Ta et al., 2022; Zhu et al., 2022). Spatiotemporal GNN techniques rely on the graph convolution operation to handle graph-structured and temporal traffic data, like simple temporal convolution (Lee and Rhee, 2022; Zhu et al., 2022), gated temporal convolution layers (Ta et al., 2022; Yu et al., 2017), LSTM (Peng et al., 2021), and attention-based encoder/decoder networks (Huang et al., 2022; Ye et al., 2021). Cui et al. (2020a) proposed a traffic graph convolutional LSTM-based neural network (TGC-LSTM) model to learn spatiotemporal features and conduct predictions. Ye et al. (2021) combined attention encoder networks (AEN) with GNN to extract spatial and temporal features from traffic flow data. Huang et al. (2022) incorporated the inhomogeneous Poisson process into graph neural networks (GNN) to predict traffic demand. However, previous methods focus on traffic prediction and data imputation but ignore the reliability analysis of the predicted/imputed values.

Reliability analysis is vital for the decision-making in ITSs, while the existing data-driven techniques often fail to provide confidence in traffic prediction/imputation accuracy (Rodrigues and Pereira, 2018; Yuan et al., 2021). The Gaussian process (GP) can be a desirable alternative to address the issue of reliability in traffic prediction/imputation. The GP can provide closed-form expressions of the posterior distribution where the mean and covariance functions characterize the stochastic process of the observations (Williams and Rasmussen, 2006). Yuan et al. (2021) encoded the physical traffic state information as the shadow GP when applying Bayesian inference to model the traffic flow. Spana and Du (2022) exploited GPs to extract potential relationships from training data and further incorporated them into a coordinated routing mechanism. The key in GPs is to identify the mean and covariance (i.e., kernel) function, which can be computationally expensive. Thus, Garnelo et al. (2018) proposed a neural process (NP) that maintains the properties of GPs to model distributions over functions and estimate uncertainties over predictions, and has linear computation complexity. Kim et al. (2019) further improved the training efficiency of NP by introducing the attention mechanism and proposed the attentive neural process (ANP) model.

As a novel method, NPs have not received much attention in intelligent transportation fields, although GPs have been widely used to model traffic patterns and predict traffic congestion (Liu et al., 2013; Rodrigues et al., 2019; Rodrigues and Pereira, 2018). Therefore, this study aims to incorporate ANP into graph neural networks and proposes an attentive graph neural process (AGNP) method for short-term traffic network imputation. After identifying the spatiotemporal patterns of the traffic data, we model the traffic data as a stochastic process which the AGNP will further learn. Based on observed traffic state information on arterial highways, we can give the congestion probability on the remaining road segments by our proposed AGNP.

The remainder of this study is organized as follows. In **Section 2**, the research problem and the spatiotemporal features of traffic data are introduced. The AGNP model is then formulated in **Section 3**. **Section 4** shows the results and discussions based on a real case. **Section 5** concludes the study.

# **2. Research Problem**

A road segment is defined as a directed connection between two intersections in the source data. Specifically, each road segment has a starting intersection, an ending intersection, and a corresponding free-flow speed. In addition, each road consists of two parts: in the longer mixed zone, vehicles can move and switch lanes freely, while in the channelized zone, vehicles must join the corresponding lane to complete a specified movement, either a left turn, a thorough movement or a right turn. The speed of vehicles in the mixed zone and the channelized zone is recorded separately. A detailed description of the dataset is provided in **Section 4.1**.

Based on the data features stated above, we first specified the definition of nodes. Intuitively, when constructing a directed graph in a transportation network, intersections are defined as nodes, and roads are defined as directed links between nodes. However, our study is more concerned with the connectivity between road segments. Therefore, we treat the lanes as nodes because of the channelized features that road segments naturally carry. The definition of a node is given as:

|  |  |
| --- | --- |
|  | (1) |
|  | (2) |

where is the starting intersection ID, denotes the ending intersection ID, is a collection of intersection ID, and is a collection of all nodes. is the target direction (mixed 0/ left 1/ thorough 2/ right 3).

Two datasets are constructed based on nodes in a fixed order. Firstly, features are recorded for . Features of node including the number of lanes , free-flow speed , length of the channelized zone , length of the mixed zone , turn rate and green time ratio (or equivalent terms).

|  |  |
| --- | --- |
|  | (3) |

These feature values are only related to the node and are constant values that do not change over time. Secondly, a connectivity-speed matrix is constructed with the dimension of , where the first dimension denotes the timestamp of the record, the second dimension means the current nodewhere the speed is recorded, and the third dimension represents the target node of vehicles where the vehicle flow is heading. Recorded values of speed are assigned to the matrix with the coordinate given by:

|  |  |
| --- | --- |
|  | (4) |

The very same value of speed is assigned when the target node has the same , and ; that is

|  |  |
| --- | --- |
|  | (5) |

In other words, we assume that the speed of vehicles is equal in the different channelized lanes of the target node. For speed on the mixed lane, we make equal to . That is

|  |  |
| --- | --- |
|  | (6) |

The connectivity of the road network is thus expressed as whether the velocity between any two nodes is zero. The matrix is then split into layers according to the first dimension, i.e., each layer contains connectivity information of the road network at the same timestamp, and the link weights (expressed in terms of speed) of the links connected. Each timestamp layer would be utilized to conduct imputation from speed information on arterial lanes to congestion probability on branch lanes.

The road network at a given timestamp *t* is modeled as a directed network ,in which nodes , feature of node , and speed of connected nodes at given timestamp *t.* As aforementioned, if link between node and exists at a given time *t*, then We define congestion condition as that speed on a certain lane is below 20. The network imputation problem on a single layer is formulated as follows:

|  |  |
| --- | --- |
|  | (7) |

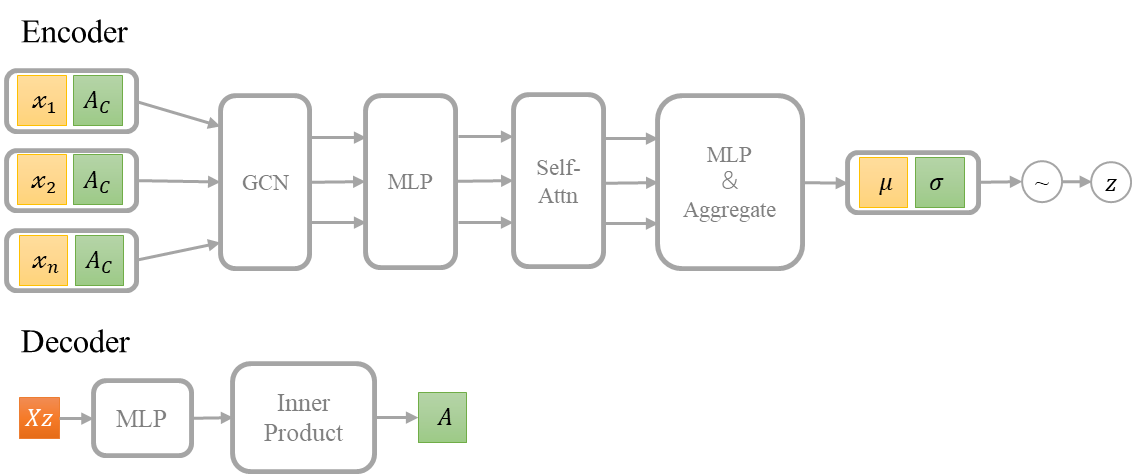
where indicates all arterial nodes, represents all branch nodes, is the matrix of arterial node speed and is the matrix of probability of congestion.

# **3. Attentive Graph Neural Processes**

In this section, we comprehensively introduce the proposed AGNP model. We display the overall model framework in **Section 3.1**, and **Section 3.2** describes each module in detail. The overall learning procedure is introduced in **Section 3.3**.

## *3.1 Overview of the Model Framework*

The proposed attentive graph neural process (AGNP), which exploits an encoder-decoder structure, is depicted in **Figure 1** briefly. The framework starts with a graph convolutional module (Kipf et al., 2016); it helps to aggregate node features with their neighbors’ features according to the adjacent matrix and convert the features of each node into a latent representation. The representations go through a multi-layer perceptron (MLP) module with three hidden layers and then a self-attention module with a multi-head attention mechanism to further encode their information. Next, the representations encoded will be fed into an MLP module and an aggregation module to obtain the mean and the variance for the multivariate Gaussian distribution. Finally, with the mean and variance, we sample a , concentrate it with node features of all nodes, and go through a Decoder with MLP and inner product modules to calculate the whole adjacent matrix for the complete graph.



**Figure 1**. Architecture of AGNP

## *3.2 Detailed Modules*

### *3.2.1 Graph Convolutional Module*

Graph convolutional module (GCM) is the first among our proposed AGNP framework; it takes advantage of graph structural information, helps each node integrate information from itself and its neighbor nodes, and thus generates more informative representations. In this work, we apply a two-layer graph convolutional network (Kipf et al., 2016) as our first module, and each layer has the following layer-wise expression:

|  |  |
| --- | --- |
|  | (8) |

where denotes the revised adjacent matrix of the small context graph with added self-connections. represents the identity matrix, denotes the degree matrix of , is the trainable weight matrix of layer and is the adopted activation function. is the matrix fed into the layer and .

### *3.2.2 MLP and Self-Attention Module*

After graph convolutional layers, the generated representation vectors will go through a 3-layer MLP module first, where each layer has identical expressions shown as follows.

|  |  |
| --- | --- |
|  | (9) |

where is the representations fed into the MLP layer.

Next, the representation will be sent into a multi-head self-attention mechanism. Generally, the attention mechanism’s input usually consists of queries and key-value pairs and , with the output being a weighted sum of the values. We derive the weight for each key-value pair by computing the similarity between the query and corresponding key. Moreover, the output is a weighted sum functioned on values , with the as weighted coefficients. The similarity function adopted in this work is scaled dot-product operation, and the computation expression is presented as follows.

|  |  |
| --- | --- |
|  | (10) |

Scaling the result by helps to offset extremely small gradient issues. It is more effective to perform multi-head attention mechanisms than single-head attention, with different learned queries, keys, and values. The formula is provided below.

|  |  |
| --- | --- |
|  | (11) |
|  | (12) |

where concat() is a concatenation function. Self-attention mechanisms denote , and to be the same matrix.

After these two modules, the representation can be further embedded by MLPs and converted based on interactions between features with the aid of the self-attention module. For example, if numerous features overlap, the self-attention mechanism helps to save efforts from paying attention to all of them, but merely to one or a few. In other words, the self-attention mechanism will guide to obtain a more informative representation of features that encode these sorts of relationships.

### *3.2.3 Aggregation Module and Decoder Module*

After embedding with MLPs and self-attention mechanisms, each node will have a latent representation. The latent representations will be further passed through another MLP to generate the latent probability distribution of the global representation which is parameterized by mean and standard variance . Specifically, and are obtained by:

|  |  |
| --- | --- |
|  | (13) |
|  | (14) |

After the aggregation module, we sample with the latent probability distributions obtained, and stack it with each feature to get . In this way, the new representations contains global information which is embedded in . The is next passed through two-layer MLP and take inner product with sigmoid function to calculate the probability of link existence between two nodes.

## *3.3 Overall Training Procedure*

Referring to NPs, the variational lower bound of AGNP can be expressed as follows.

|  |  |
| --- | --- |
|  | (15) |

where is the output of the decoder introduced in **Section 3.2**, is parameterized by mean and variance calculated in the aggregation modules based on the whole observed graph and is similarly parameterized based on randomly generated subgraph from . In this way, and in the training phase can be viewed as whole graph and subgraph, and further in predicting phase, are counterparts for whole graph with unobserved links and observed graph . Hence, maximizing the variational lower bound is to maximize the conditional log likelihood and meanwhile, force the latent distribution inferred from and to be close to the latent distribution inferred from and the whole graph with unobserved links. Conclusively, parameters of AGNPs are learned by maximizing Eq. (15) via the reparameterization trick (Kingma and Welling, 2014). In this manner, the hyper level training procedure to optimize the proposed AGNP model is summarized in **Algorithm 1**.

**Algorithm 1.** Training Procedure for AGNP

|  |
| --- |
| **1**. Input: observed graph with features  **2**. for to do  **3**. Generate a subgraph randomly with features  **4**. ,  **5**. ,  **6**. Calculate  **7**. Sample  **8**.  **9**. Compute , and update and  **10**. End |

# **4. Numerical Studies**

### *4.1 Data Descriptions*

The dataset we use is sampled from the road network in the central city of Xuancheng, Anhui Province, China. Through dense checkpoints set up in the road network, this dataset records 374 valid road sections, each of which possesses a free-flow speed, length of the mixed zone and the channelized zones (constant between different lanes). After counting the lanes of each road section, the road sections were transformed into 1196 valid lanes, containing both mixed and channelized lanes. The average speed and flow were recorded at a five-minute interval for each lane, producing 5220 valid time stamps. The statistics also summarize the number of lanes in each lane, the turn rate, and the green time ratio for each direction of the channelized lanes. The recording. All recorded data is arranged in a fixed lane order, which is given in a separate file, containing the start and end intersection ID, the direction of the turning (0 for mixed lane, 1 for turning left, 2 for thorough movement, 3 for turning right), the target intersection ID after turning (0 if the lane denotes the mixed lane).

In our experiment, based on the processing described in **Section 2**, model is tested each time on a single matrix layer with 1196 nodes, 2241 node features and 1563 valid links at any given timestamp. To evaluate the performance of proposed model, we introduce criteria of the Area Under a receiver operating characteristic Curve (AUC) and Average Precision (AP). 10 iterations of the model running on different choice of links are conducted and mean of 10 folds is kept for evaluation.

### *4.2 Results and Discussion*

Based on one five-minute network graph obtained in Xuancheng, we randomly mask 10% links for testing, and 10% for validation, and the rest links are utilized as the partial adjacent matrix forthe observed graph in the training phase. When generating the context subgraph based on the observed graph, we further randomly mask 10% training links to produce the adjacent matrix for the subgraph with. In prediction phase, we use the learned AGNP model to produce whole adjacent matrix conditioned on the partial adjacent matrix and features for nodes.

The results are summarized in **Table 1.** It demonstrates the performance of our proposed model in AUC and AP scores.

**Table 1**. Prediction Results

|  |  |  |
| --- | --- | --- |
|  | ROC | AP |
| Mean | 0.918 | 0.930 |
| Std | 0.03 | 0.02 |

# **5. Conclusion**

In urban areas, it is unrealistic for monitoring systems to cover every road due to the limited budget issue and privacy concerns. Hence, traffic network imputation is of great importance and values for real-world transportation systems. In this work, we treat the traffic speed state on arterial highways as a stochastic process and propose an attentive graph neural process (AGNP) method for short-term network imputation problems. The proposed model is capable of conducting reliability analysis and providing confidence in its predictions. The predicted results can be further summarized into congestion knowledge on the unobserved road segments. Real-world traffic speed states collected in Xuancheng, Anhui, China, is utilized to test the performance of AGNPs. The results demonstrate a high-level predictive performance in traffic network imputation and traffic congestion recognition.

There are some future works that can be conducted to extend this work. First, at this moment, we dismiss the temporal dependency of the transportation networks; we will attempt to further design a model that can simultaneously consider spatiotemporal correlations. Second, the model is designed for short-term predictions which is sometimes unrealistic in real-world; after all, it is not always possible to obtain data statistics in such a short time, and make predictions based on it. In the future, we will consider to design a relatively long-term model that can summarize the information of past few hours and predict traffic states in the long term. Third, the reliability analysis experiments is unfinished; we are still finding an appropriate method to illustrate the convenience and promising values of the AGNPs.

# **Acknowledgments**

The work described in this paper was partially supported by Hong Kong Research Grants Council under project HKUST16208920, National Natural Science Foundation of China under projects 71922019, 72171210, Alibaba-Zhejiang University Joint Research Institute of Frontier Technologies, and the Hong Kong University of Science and Technology–DiDi Chuxing (HKUST-DiDi) Joint Laboratory.

# **Reference**

Chan, R. K. C., Lim, J. M. Y., & Parthiban, R. (2021). A neural network approach for traffic prediction and routing with missing data imputation for intelligent transportation system. *Expert Systems with Applications*, *171*, 114573.

Cui, Z., Henrickson, K., Ke, R., & Wang, Y. (2019). Traffic graph convolutional recurrent neural network: A deep learning framework for network-scale traffic learning and forecasting. *IEEE Transactions on Intelligent Transportation Systems*, *21*(11), 4883-4894.

Cui, Z., Lin, L., Pu, Z., & Wang, Y. (2020). Graph Markov network for traffic forecasting with missing data. *Transportation Research Part C: Emerging Technologies*, *117*, 102671.

de Medrano, R., & Aznarte, J. L. (2020). A spatio-temporal attention-based spot-forecasting framework for urban traffic prediction. *Applied Soft Computing*, *96*, 106615.

Duan, Y., Lv, Y., Liu, Y. L., & Wang, F. Y. (2016). An efficient realization of deep learning for traffic data imputation. *Transportation research part C: emerging technologies*, *72*, 168-181.

El Hamdani, S., Benamar, N., & Younis, M. (2020). Pedestrian support in intelligent transportation systems: challenges, solutions and open issues. *Transportation research part C: emerging technologies*, *121*, 102856.

Garnelo, M., Schwarz, J., Rosenbaum, D., Viola, F., Rezende, D. J., Eslami, S. M., & Teh, Y. W. (2018). Neural processes. *arXiv preprint arXiv:1807.01622*.

Huang, F., Yi, P., Wang, J., Li, M., Peng, J., & Xiong, X. (2022). A dynamical spatial-temporal graph neural network for traffic demand prediction. *Information Sciences*, *594*, 286-304.

Kaur, M., Singh, S., & Aggarwal, N. (2022). Missing traffic data imputation using a dual-stage error-corrected boosting regressor with uncertainty estimation. *Information Sciences*, *586*, 344-373.

Kim, H., Mnih, A., Schwarz, J., Garnelo, M., Eslami, A., Rosenbaum, D., ... & Teh, Y. W. (2019). Attentive neural processes. *arXiv preprint arXiv:1901.05761*.

Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Lee, K., & Rhee, W. (2022). DDP-GCN: Multi-graph convolutional network for spatiotemporal traffic forecasting. *Transportation Research Part C: Emerging Technologies*, *134*, 103466.

Li, L., Li, Y., & Li, Z. (2013). Efficient missing data imputing for traffic flow by considering temporal and spatial dependence. *Transportation research part C: emerging technologies*, *34*, 108-120.

Li, L., Zhang, J., Wang, Y., & Ran, B. (2018). Missing value imputation for traffic-related time series data based on a multi-view learning method. *IEEE Transactions on Intelligent Transportation Systems*, *20*(8), 2933-2943.

Liang, Y., Zhao, Z., & Sun, L. (2021). Dynamic spatiotemporal graph convolutional neural networks for traffic data imputation with complex missing patterns. *arXiv preprint arXiv:2109.08357*.

Liu, S., Yue, Y., & Krishnan, R. (2013, August). Adaptive collective routing using gaussian process dynamic congestion models. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 704-712).

Luan, S., Ke, R., Huang, Z., & Ma, X. (2022). Traffic congestion propagation inference using dynamic Bayesian graph convolution network. *Transportation Research Part C: Emerging Technologies*, *135*, 103526.

Peng, H., Du, B., Liu, M., Liu, M., Ji, S., Wang, S., ... & He, L. (2021). Dynamic graph convolutional network for long-term traffic flow prediction with reinforcement learning. *Information Sciences*, *578*, 401-416.

Rodrigues, F., Henrickson, K., & Pereira, F. C. (2018). Multi-output Gaussian processes for crowdsourced traffic data imputation. *IEEE Transactions on Intelligent Transportation Systems*, *20*(2), 594-603.

Rodrigues, F., & Pereira, F. C. (2018). Heteroscedastic Gaussian processes for uncertainty modeling in large-scale crowdsourced traffic data. *Transportation research part C: emerging technologies*, *95*, 636-651.

Salamanis, A., Kehagias, D. D., Filelis-Papadopoulos, C. K., Tzovaras, D., & Gravvanis, G. A. (2015). Managing spatial graph dependencies in large volumes of traffic data for travel-time prediction. *IEEE Transactions on Intelligent Transportation Systems*, *17*(6), 1678-1687.

Smith, B. L., Williams, B. M., & Oswald, R. K. (2002). Comparison of parametric and nonparametric models for traffic flow forecasting. *Transportation Research Part C: Emerging Technologies*, *10*(4), 303-321.

Spana, S., & Du, L. (2022). Optimal information perturbation for traffic congestion mitigation: Gaussian process regression and optimization. *Transportation Research Part C: Emerging Technologies*, *138*, 103647.

Sumalee, A., & Ho, H. W. (2018). Smarter and more connected: Future intelligent transportation system. *Iatss Research*, *42*(2), 67-71.

Ta, X., Liu, Z., Hu, X., Yu, L., Sun, L., & Du, B. (2022). Adaptive Spatio-temporal Graph Neural Network for traffic forecasting. *Knowledge-Based Systems*, 108199.

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

Tian, Y., Zhang, K., Li, J., Lin, X., & Yang, B. (2018). LSTM-based traffic flow prediction with missing data. *Neurocomputing*, *318*, 297-305.

Williams, C. K., & Rasmussen, C. E. (2006). *Gaussian processes for machine learning* (Vol. 2, No. 3, p. 4). Cambridge, MA: MIT press.

Yang, J. M., Peng, Z. R., & 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.

Ye, J., Xue, S., & Jiang, A. (2021). Attention-based spatio-temporal graph convolutional network considering external factors for multi-step traffic flow prediction. *Digital Communications and Networks*.

Yu, B., Yin, H., & Zhu, Z. (2017). Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint arXiv:1709.04875*.

Yuan, Y., Zhang, Z., Yang, X. T., & Zhe, S. (2021). Macroscopic traffic flow modeling with physics regularized Gaussian process: A new insight into machine learning applications in transportation. *Transportation Research Part B: Methodological*, *146*, 88-110.

Zhang, Z., Lin, X., Li, M., & Wang, Y. (2021). A customized deep learning approach to integrate network-scale online traffic data imputation and prediction. *Transportation Research Part C: Emerging Technologies*, *132*, 103372.

Zhu, K., Zhang, S., Li, J., Zhou, D., Dai, H., & Hu, Z. (2022). Spatiotemporal multi-graph convolutional networks with synthetic data for traffic volume forecasting. *Expert Systems with Applications*, *187*, 115992.