Long-term Spatio-Temporal Forecasting via Dynamic Multiple-Graph Attention

Wei Shao\textsuperscript{1,\ast}, Zhiling Jin\textsuperscript{2\dagger}, Shuo Wang\textsuperscript{3\dagger}, Yufan Kang\textsuperscript{1}, Xiao Xiao\textsuperscript{2}, Hamid Menouar\textsuperscript{5}, Zhaofeng Zhang\textsuperscript{1}, Junshan Zhang\textsuperscript{6} and Flora Salim\textsuperscript{7}

\textsuperscript{1}School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, United States
\textsuperscript{2}School of Telecommunications Engineering, Xidian University, Xi’an, China
\textsuperscript{3}D-ITET, ETH Zurich, Zurich, Switzerland
\textsuperscript{4}School of Computing Technologies, RMIT University, Melbourne, Australia
\textsuperscript{5}Qatar Mobility Innovations Center, Qatar University, Doha, Qatar
\textsuperscript{6}ECE Department at University of California Davis, Davis, United States
\textsuperscript{7}School of Computer Science and Engineering, University of New South Wales, Sydney, Australia

\{phdweishao, shawnwang.tech\}@gmail.com, jinzhiling@stu.xidian.edu.cn,
s3711739@student.rmit.edu.au, xiaoxiao@xidian.edu.cn, hamidm@qmic.com, zzhan199@asu.edu,
jazh@ucdavis.edu, flora.salim@unsw.edu.au

Abstract

Many real-world ubiquitous applications, such as parking recommendations and air pollution monitoring, benefit significantly from accurate long-term spatio-temporal forecasting (LSTF). LSTF makes use of long-term dependency structure between the spatial and temporal domains, as well as the contextual information. Recent studies have revealed the potential of multi-graph neural networks (MGNNs) to improve prediction performance. However, existing MGNN methods do not work well when applied to LSTF due to several issues: the low level of generality, insufficient use of contextual information, and the imbalanced graph fusion approach. To address these issues, we construct new graph models to represent the contextual information of each node and exploit the long-term spatio-temporal data dependency structure. To aggregate the information across multiple graphs, we propose a new dynamic multi-graph fusion module to characterize the correlations of nodes within a graph and the nodes across graphs via the spatial attention and graph attention mechanisms. Furthermore, we introduce a trainable weight tensor to indicate the importance of each node in different graphs. Extensive experiments on two large-scale datasets demonstrate that our proposed approaches significantly improve the performance of existing graph neural network models in LSTF prediction tasks.

1 Introduction

Recently, various spatio-temporal prediction tasks have been investigated, including traffic flow [Li \textit{et al.}, 2018; Huang \textit{et al.}, 2020; Yu \textit{et al.}, 2018], parking availability [Zhang \textit{et al.}, 2020a], and air pollution [Wang \textit{et al.}, 2020b; Wen \textit{et al.}, 2019; Liu \textit{et al.}, 2021]. All the scenarios above benefit from an accurate forecast by leveraging historical data in the long run, namely, long-term spatio-temporal forecasting (LSTF).

One main challenge in LSTF is to effectively capture the long-term spatio-temporal dependency and extract contextual information. Recently, multi-graph neural networks (MGNNs) [Wang \textit{et al.}, 2021] have received increasing attention for spatio-temporal forecasting problems. Specifically, as shown in Figure 1, each node’s value $V_i$ is estimated in the long run using historical data and correlations across nodes of a distance graph, where each edge denotes the correlation or dependency between two different nodes. Furthermore, the functionality similarities of surrounding areas, which represent contextual information, can also be used for prediction purposes. Compared to the single graph approach, which may not comprehensively capture all the relationships, the MGNN-based approach is appropriate for...
leveraging more information and features by integrating different graphs. Thus, in this work, we choose the MGNN-based approach to infer how information about each node evolves over time.

Although MGNNs show potential for extracting contextual information around prediction sites, four significant limitations remain when solving the LSTM problem:

1. **Most existing MGNN studies consider only the spatial similarity of nodes, such as the distance similarity and neighborhood correlation.** Previous studies have shown that the distance similarity is insufficient to represent correlations among nodes with spatio-temporal attributes [Geng et al., 2019]. Wu et al. [Wu et al., 2019] proposed an adaptive adjacency matrix to discover hidden spatial dependencies directly from historical records of each node in an end-to-end fashion by computing the inner product of the nodes’ learnable embedding. However, these works did not utilize well the existing prior knowledge encoded as an adjacency matrix, which may result in missing vital information.

2. **Fusing different graph models is challenging.** For multi-graph-based problems, the graph models differ with different scales; thus, it is inappropriate to simply merge them using weighted sum or other averaging approaches. Additionally, how to align each node in different graphs is challenging since nodes in different graphs are associated with different spatio-temporal information.

3. **Existing multi-graph fusion approaches rely heavily on specific models.** The current MGNNs lack generalizability. Specifically, the existing graph construction approaches and fusion methods need to be strictly bonded, assuming specific graph neural network structures. Although such an end-to-end framework provides a convenient method, it induces various difficulties in examining the importance of each graph to find a better combination of each module.

4. **Long-term spatio-temporal dependency is not considered.** Usually, MGNNs tend to learn the spatio-temporal dependency by projecting mapping from data within the observation window and the prediction horizon. However, due to the limitation of data sources, existing graph models, such as the distance graph [Li et al., 2018] or the neighbor graph [Geng et al., 2019] represent only the static spatial information, which cannot capture the long-term spatio-temporal dependency.

To address the issues above, we investigate graph construction and fusion mechanisms, and make improvements to each component. Specifically, we take advantage of human insights to build a new graph model namely ‘heuristic graph’, for the LSTM problem, which can represent the long-term spatio-temporal dependency from historical data or human insights and can be widely used for various graph neural networks.

- We design a novel graph model fusion module called a dynamic graph fusion block to integrate various graph models with graph attention and spatial attention mechanisms, aiming to align nodes within graphs and across different graphs. We further construct a learnable weight tensor for each node to flexibly capture the dynamic correlations between nodes.
- We conduct extensive experiments on two large-scale public real-world spatio-temporal datasets. We validate the effectiveness of the proposed new graph models and fusion approaches using ablation studies.

## 2 Methodologies

As shown in Figure 2, the proposed framework consists of three major components: the graph construction module, the dynamic multi-graph fusion module, and the spatio-temporal graph neural network (ST-GNN). We designed five graphs to represent different aspects of the spatio-temporal information in the graph construction module. In the dynamic multi-graph fusion module, we align spatial and temporal dependency using an adaptive trainable tensor and introduce graph and spatial attention mechanisms to calculate the correlations among nodes located in different graphs. We then obtain the prediction results with existing ST-GNN models.

### 2.1 Graph Construction

In this section, we describe in detail two new graph models we proposed named the heuristic graph \( G^H = \{V, E, W^H\} \) and the functionality graph \( G^F = \{V, E, W^F\} \), combined with other three existing graphs, the distance graph \( G^D = \{V, E, W^D\} \), neighbor graph \( G^N = \{V, E, W^N\} \), and temporal pattern similarity graph \( G^T = \{V, E, W^T\} \), into a multiple graph set \( G = \{G^D, G^N, G^F, G^H, G^T\} \).

**Distance Graph.** The element of distance matrix \( W^D \) is defined with a thresholded Gaussian kernel as follows [Shuman et al., 2013]:

\[
W^D_{ij} := \begin{cases} \exp \left( -\frac{d_{ij}^2}{\sigma_D^2} \right), & \text{for } i \neq j \text{ and } \exp \left( -\frac{d_{ij}^2}{\sigma_D^2} \right) \geq \varepsilon, \\ 0, & \text{otherwise,} \end{cases} \tag{1}
\]

where \( d_{ij} \) is the Euclidean distance between \( v_i \) and \( v_j \), \( \varepsilon \) and \( \sigma_D^2 \) are used to control the sparsity and distribution of \( W^D \).

**Neighbor Graph.** The element of neighbor matrix \( W^N \) is defined as follows:

\[
W^N_{ij} := \begin{cases} 1, & \text{if } v_i \text{ and } v_j \text{ are adjacent,} \\ 0, & \text{otherwise.} \end{cases} \tag{2}
\]

**Functionality Graph.** Usually, places with similar functionalities or utilities, such as factories, schools, and hospitals, have strong correlations. In this paper, different from the functionality graph proposed by [Geng et al., 2019], we propose a new functionality graph using Pearson correlation
coefficients to capture the global contextual function similarity. Denote the total number of functions is $K$; then the vector of the number of these functions of vertex $v_i$ is denoted as $F_i = \{f_{i,1}, f_{i,2}, \ldots, f_{i,k}, \ldots, f_{i,K}\}$. The functionality matrix can be obtained using Pearson correlation coefficients [Zhang et al., 2020b] by

$$W_{ij}^F := \begin{cases} \frac{\sum_{k=1}^{K}(f_{i,k} - F_i)(f_{j,k} - F_j)}{\sqrt{\sum_{k=1}^{K}(f_{i,k} - F_i)^2} \sqrt{\sum_{k=1}^{K}(f_{j,k} - F_j)^2}}, & \text{if } i \neq j, \\ 0, & \text{otherwise}. \end{cases}$$

(3)

Note that we consider all functions that contribute equally to the relationships of nodes.

**Heuristic Graph.** To leverage heuristic knowledge and human insights, we propose a new graph model called the heuristic graph. We create a histogram to represent the overview of the spatio-temporal training data, where each bin indicates a predefined temporal range, and the bar height measures the number of data records that fall into each bin. Then we apply a function $f(x) = \alpha e^{-\beta x}$ to approximate the histogram. For a vertex $v_i$, we can obtain its fitted parameters $\alpha_i$ and $\beta_i$. The distribution distance is calculated using the Euclidean distance $d_{ij}^H = (\alpha_i - \alpha_j)^2 + (\beta_i - \beta_j)^2$. The element of the heuristic matrix $W^H$ can be defined as follows:

$$W_{ij}^H := \begin{cases} \exp \left( -\frac{\|d_{ij}^H\|^2}{\sigma_{H}^2} \right), & \text{for } i \neq j, \\ 0, & \text{otherwise}. \end{cases}$$

(4)

where $\sigma_{H}^2$ is a parameter to control the distribution of $W^H$. Kullback-Leibler (KL) divergence [Van Erven and Harremos, 2014] can be also used to create this graph, which usually quantifies the difference between two probability distributions.

**Temporal Pattern Similarity Graph.** For a vertex $v_i$, the vector of the time-series data used for training is described as...
Dynamic Multi-graph Attention Block

To model inner node correlations, we design a multi-level attention mechanism. When acting on multiple graphs, these impacts are denoted as \( H^{(l+1)}_v \) and \( H^{(l+1)}_G \), respectively. Spatial Attention. Inspired by [Zheng et al., 2020], we capture the contextual correlations of nodes by proposing a spatial attention mechanism (shown in Figure 3a). Different from the previous spatial attention mechanism, which acts on the hidden state of the batch of temporal data, our method acts on the hidden state of the weight tensor. Then we can calculate the next hidden state in the graph node as follows:

\[
\tilde{h}_i^{(l)} = \sum_{v_k \in \mathcal{V}_i} \alpha_{i,v_k} \cdot h_{v_k}^{(l)}
\]

(6)

where \( \mathcal{V}_i \) is all the vertices on the graph except the \( v_i \), \( \alpha_{i,v_k} \) is the attention score respecting the importance of \( v_k \) to \( v_i \).

In the real world, the vertices are influenced not only by other vertices on the same graph but also other graphs. For example, the parking occupancy rate of one place is affected not only by the distance from another place but also by the functionality of another place. To this end, we concatenate the hidden state with MGSE to extract both the spatial and graph attention mechanisms as denoted by \( H^{(l+1)}_S \) and \( H^{(l+1)}_G \), respectively.

Multi-graph Spatial Embedding

We denote the input of the \( l \)-th block \( H^{(l)}_v \) and denote the hidden state of the vertex \( v_i \) on graph \( G_i \) in \( H^{(l)}_v \) as \( h^{(l)}_{v_i,G_i} \). We then define this as follows:

\[
\tilde{h}_i^{(l+1)} = \mathbb{E}_{v_k \in \mathcal{V}_i} \alpha_{i,v_k} \cdot h_{v_k}^{(l)}
\]

(6)

\[
\alpha_{i,v_k} = \exp (s_{i,v_k}) / \sum_{v_k \in \mathcal{V}_i} \exp (s_{i,v_k}).
\]

To stabilize the learning process, we concatenate \( M \) parallel attention mechanisms to extend them to the multi-head attention mechanism [Zheng et al., 2020] with

\[
s_{i,v_k}^{(m)} = \langle \tilde{h}_{v_k,G_i}^{(l)} \| \tilde{h}_{v_i,G_i}^{(l)} \rangle / \sqrt{d_i},
\]

(7)

where \( \| \) is the concatenation operation and \( \langle | \rangle \) is the inner product operation. Then \( s_{v_i,v_k} \) is normalized by the softmax function

\[
\beta_{i,v_k} = \exp (s_{i,v_k}) / \sum_{v_k \in \mathcal{V}_i} \exp (s_{i,v_k}).
\]

Dynamic Multi-graph Attention Block

Any node in a graph is impacted by other nodes with different levels. When acting on multiple graphs, these impacts are magnified. To model inner node correlations, we design a multi-graph attention block to adaptively capture the correlations among the nodes. As shown in Figure 2, the multi-graph attention block contains spatial attention and graph attention.

Dynamic Multi-graph Fusion


Parameter: Number of batches: \( B_t \)

Output: Fused weight matrix \( W^* \)

1. Stack weight matrices to tensor \( T^{(0)} \in \mathbb{R}^{\left| G \right| \times N \times N} \).
2. Train \( T^{(0)} \) while training ST-GNN models.
3. for \( i \in [0, B_t - 1] \) do
   4. \( T^{(i+1)} \leftarrow T^{(i)} \)
   5. \( i \leftarrow i + 1 \)
6. \( W^* = \sum_{i=1}^{\left| G \right|} T^{(i)}(i, j, k) \), where \( W^* \) is the element of the weight matrix of the fused graph.
7. end for
8. return \( W^* \)
\[
\begin{align*}
    h_{v_i,G_i}^{(l+1)} &= \| M_{m=1} \left\{ \sum_{n=1}^{N} a_{v_i,v_n}^{(m)} \cdot f_s^{(m)} \left( h_{v_n,G_i}^{(l)} \right) \right\} ,
\end{align*}
\]
where \( f_s^{(m)}(\cdot) \), \( f_s^{(m)}(\cdot) \), and \( f_s^{(m)}(\cdot) \) are different ReLU functions serving as nonlinear projections in \( m \)-th head attention. \( a_{v_i,v_n}^{(m)} \) is calculated with a softmax function in the \( m \)-th head attention and \( h_{v_i,G_i}^{(l+1)} \) is the hidden state of \( v_i \in G_i \).

**Graph Attention.** We employ graph attention to obtain the self-correlations of a node in different graphs (as shown in Figure 3b). Similar to the spatial attention mechanism, we concatenate the hidden state with MGSE and employ the multi-head method to calculate the correlations. For \( v_i \), the correlation between graph \( G_j \) and \( G_k \) is defined as:
\[
    u_{G_j,G_k}^{(m)} = \frac{\langle f_{G_1}^{(m)} \left( h_{v_i,G_j}^{(l)} \right), f_{G_2}^{(m)} \left( h_{v_i,G_k}^{(l)} \right) \rangle}{\sqrt{d}},
\]
where \( \beta_{G_j,G_k}^{(m)} \) calculated with a softmax function is the attention score in the \( m \)-th head, indicating the importance of graph \( G_k \) to \( G_j \). \( f_{G_1}^{(m)}(\cdot) \), \( f_{G_2}^{(m)}(\cdot) \), and \( f_{G_3}^{(m)}(\cdot) \) are the ReLU functions in \( m \)-th head attention.

**Gated Fusion.** To further extract the correlations of nodes on different graphs, we adopt the gated fusion method [Zheng et al., 2020] to consider both effects. The spatial attention \( H_{G}^{(l)} \) and the graph attention \( H_{G}^{(l)} \) in the \( l \)-th block are fused with
\[
    H^{(l)} = z \odot H_{G}^{(l)} + (1-z) \odot H_{S}^{(l)},
\]
where the gate \( z \) is calculated by:
\[
    z = \sigma \left( H_{S}^{(l)} W_{z,1} + H_{G}^{(l)} W_{z,2} + b_z \right),
\]
where \( W_{z,1} \in \mathbb{R}^{D \times D} \), \( W_{z,2} \in \mathbb{R}^{D \times D} \), and \( b_z \in \mathbb{R}^D \) are the learnable parameters, \( \odot \) indicates the element-wise Hadamard product, and \( \sigma(\cdot) \) is the sigmoid activation function. By combining the spatial and graph attention mechanisms, we further create a spatial-graph attention (SG-ATT) block, which is shown in Figure 2.

## 3 Experiments

### 3.1 Datasets

**Parking:** The Melbourne parking dataset, collected by the Melbourne City Council in 2019, contains 42,672, 743 parking events recorded by the in-ground sensors every five minutes located in the Melbourne Central Business District (CBD) [Shao et al., 2017]. All sensors have been classified into 40 areas.

**Air Quality:** The Ministry of Ecology and Environment of China (MEE) published a large-scale air quality dataset [Wang et al., 2020b], comprising 92 air quality monitoring stations, to assess the hourly PM\(_{2.5}\) concentration in Jiangsu province in 2020.

### 3.2 Experimental Details

**Baselines.** We selected five state-of-the-art ST-GNN models as baselines: STGCN [Yu et al., 2018], ASTGCN [Guo et al., 2019], MSTDGCN [Guo et al., 2019], ST-MGCN [Geng et al., 2019], and Graph WaveNet [Wu et al., 2019].

**Platform.** All experiments were trained and tested on a Linux system (CPU: Intel(R) Xeon(R) Gold 6240 CPU @2.60GHz, GPU: NVIDIA GeForce RTX 2080 Ti).

**Hyper-parameters.** All the tests used a 24-time step historical time window, and the prediction horizons ranged from three to 24 steps. The proposed methods were optimized with the Adam optimizer. The learning rate was set to \( 1e^{-4} \). The L1 loss function was adopted to measure the performance of the proposed model. The batch size was 32, and the global seed was set to 0 for the experiment repeat. All the tests were trained for 40 epochs. The number of attention heads \( M \) and the dimension \( d \) of each attention head were set to 8 and 8 in the Parking dataset and set to 24 and 6 in the Air Quality dataset.

**Evaluation Metrics.** In our study, mean absolute error (MAE) and root mean square error (RMSE) were used.

### 3.3 Results and Analysis

Table 1 summarizes the results of all ST-GNN models based on the two datasets. The prediction horizon ranged from three time steps to 24 steps. The best evaluation results are highlighted in boldface. The number of highlighted values is three to 24 steps. The proposed methods were optimized with the existing ST-GNN models with various prediction fusion approaches. We compared results between the existing ST-GNN models. The results illus-

Table 1 shows the following: (1) When the proposed dynamic multi-graph fusion approach (marked with "+") was used, the prediction performances significantly improved. For example, when the STGCN method was used, our method had an average RMSE decrease of 9.5% (over all prediction horizons). This indicates that our multi-graph fusion methods can extract more information and be effective for various ST-GNN models. (2) When the same ST-GNN methods are used, our proposed methods outperform the original ones in winning counts under all circumstances, which demonstrates the strong generality of our approach. (3) The results illustrate that our model is more suitable for the LSTM problem. Specifically, with the increase in prediction horizons, the gaps between vanilla ST-GNN models and our proposed models become larger. Figure 4 illustrates the trends of the proposed model and existing ST-GNN models with various prediction


[https://english.mee.gov.cn/](https://english.mee.gov.cn/)
models (green line) did not show a significant drop with the increasing prediction horizons. We found that the performance of the proposed functionality graph, respectively.

### Table 2

Table 2: The predicted RMSE of each model in the Parking dataset.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Metrics</th>
<th>Model</th>
<th>STGCN</th>
<th>ASTGCN</th>
<th>ST-MGCN</th>
<th>AST-MGCN</th>
<th>MCTGCN</th>
<th>AST-MCTGCN</th>
<th>WaveNet</th>
<th>WaveNet*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parking</td>
<td>RMSE</td>
<td>3</td>
<td>0.067</td>
<td>0.061</td>
<td>0.067</td>
<td>0.067</td>
<td>0.068</td>
<td>0.068</td>
<td>0.074</td>
<td>0.074</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6</td>
<td>0.071</td>
<td>0.068</td>
<td>0.067</td>
<td>0.066</td>
<td>0.064</td>
<td>0.061</td>
<td>0.072</td>
<td>0.067</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9</td>
<td>0.087</td>
<td>0.080</td>
<td>0.079</td>
<td>0.078</td>
<td>0.074</td>
<td>0.070</td>
<td>0.083</td>
<td>0.076</td>
</tr>
<tr>
<td></td>
<td></td>
<td>12</td>
<td>0.093</td>
<td>0.090</td>
<td>0.090</td>
<td>0.089</td>
<td>0.084</td>
<td>0.077</td>
<td>0.093</td>
<td>0.079</td>
</tr>
<tr>
<td></td>
<td></td>
<td>15</td>
<td>0.110</td>
<td>0.107</td>
<td>0.099</td>
<td>0.092</td>
<td>0.111</td>
<td>0.085</td>
<td>0.104</td>
<td>0.087</td>
</tr>
<tr>
<td></td>
<td></td>
<td>18</td>
<td>0.120</td>
<td>0.110</td>
<td>0.102</td>
<td>0.102</td>
<td>0.122</td>
<td>0.091</td>
<td>0.110</td>
<td>0.094</td>
</tr>
<tr>
<td></td>
<td></td>
<td>21</td>
<td>0.130</td>
<td>0.114</td>
<td>0.117</td>
<td>0.107</td>
<td>0.130</td>
<td>0.097</td>
<td>0.122</td>
<td>0.101</td>
</tr>
<tr>
<td></td>
<td></td>
<td>24</td>
<td>0.139</td>
<td>0.127</td>
<td>0.125</td>
<td>0.114</td>
<td>0.141</td>
<td>0.103</td>
<td>0.138</td>
<td>0.107</td>
</tr>
</tbody>
</table>

### 3.4 Ablation Study

To validate the performance of each component, we further conducted ablation studies on the Parking dataset.

#### The Performance of Functionality Graphs

Table 2 shows that (1) most ST-GNN models using the proposed functionality graph (marked with ‘†’) outperformed those using the functionality graph proposed by [Geng et al., 2019]. (2) The results using the proposed functionality graph showed less drop when the prediction horizons changed from 12 to 24, which suggests that our proposed functionality graph performs well in LSTM tasks.

#### The Performance of Heuristic Graph

Figure 5 shows that graphs generated by exponential approximation function in general outperformed other approaches with prediction horizons 12 and 24, while graphs generated by the KL divergence outperformed graphs without heuristic graphs.

#### The Performance of SG-ATT

Figure 4 shows the performance of the framework with (marked with ‘*’) and without SG-ATT (marked with ‘†’). We observe that the SG-ATT
mechanism contributes considerably to the proposed framework, especially in long-term prediction.

4 Related Work

Graph convolution networks (GCN) attracts much attention in spatio-temporal data prediction tasks recently. Bruna et al. [Bruna et al., 2013] proposed convolutional neural networks on graphs for the first time, which Defferrard et al. [Defferrard et al., 2016] extended using fast localized convolutions. Using graph-based approaches, we can easily model spatial data. However, the observation from a single graph usually brings bias, while multiple graphs can offset and attenuate the bias. Chai et al. [Chai et al., 2018] designed a multi-graph convolutional network for bike flow prediction. Geng et al. [Geng et al., 2019] encoded non-Euclidean pairwise correlations among regions into multiple graphs and then modeled these correlations using multi-graph convolution for ride-hailing demand forecasting. Lv et al. [Lv et al., 2020] encoded non-Euclidean spatial and semantic correlations among roads into multiple graphs for traffic flow prediction. However, the relationships among graphs are ignored. Moreover, the input graphs are fixed and cannot be adapted to change during training and long-term temporal information is rarely considered.

5 Conclusion

In this paper, we try to solve the LSTF problem with multi-graph neural networks. We propose two new graphs to extract heuristic knowledge and contextual information from spatio-temporal data. Specifically, we designed a heuristic graph to capture the long-term pattern of the data and a functional similarity graph to represent the similarity of functionality between two areas. To align nodes in graphs and timestamps, we designed a dynamic graph multi-graph fusion module and fed them to various graph neural networks. Extensive experiments on real-world data demonstrated the effectiveness of the proposed methods for enhancing the prediction capacity in LSTF problems. In the future, we will apply the proposed framework to additional graph-based applications.

Acknowledgments

This work was made possible, in part, by grant NPRP No. NPRP12S-0313-190348 from the Qatar National Research Fund (a member of The Qatar Foundation). The statements made herein are solely the responsibility of the authors.
References


[Zhang et al., 2020b] Xu Zhang, Ruixu Cao, Zuyu Zhang, and Ying Xia. Crowd Flow Forecasting with Multi-Graph Neural Networks. In 2020 International Joint Conference on Neural Networks (IJCNN), pages 1–7, Jul. 2020.