Research Article

Spatiotemporal DeepWalk Gated Recurrent Neural Network: A Deep Learning Framework for Traffic Learning and Forecasting

Jian Yang 1,2, Jinhong Li 1,2, Lu Wei 1, Lei Gao 1,2, and Fuqi Mao 2

1Beijing Key Lab of Urban Road Traffic Intelligent Technology, North China University of Technology, Beijing 100144, China
2School of Computer Science and Technology, North China University of Technology, Beijing 100144, China

Correspondence should be addressed to Jian Yang; yanj200045@163.com

Received 12 August 2021; Revised 7 March 2022; Accepted 11 March 2022; Published 18 April 2022

Academic Editor: Ming Xu

Copyright © 2022 Jian Yang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

As a typical spatiotemporal problem, there are three main challenges in traffic forecasting. First, the road network is a nonregular topology, and it is difficult to extract complex spatial dependence accurately. Second, there are short- and long-term dependencies between traffic dates. Third, there are many other factors besides the influence of spatiotemporal dependence, such as semantic characteristics. To address these issues, we propose a spatiotemporal DeepWalk gated recurrent unit model (ST-DWGRU), a deep learning framework that fuses spatial, temporal, and semantic features for traffic speed forecasting. In the framework, the spatial dependency between nodes of an entire road network is extracted by graph convolutional network (GCN), whereas the temporal dependency between speeds is captured by a gated recurrent unit network (GRU). DeepWalk is used to extract semantic information from road networks. Three publicly available datasets with different time granularities of 15, 30, and 60 min are used to validate the short- and long-time prediction effect of this model. The results show that the ST-DWGRU model significantly outperforms the state-of-the-art baselines.

1. Introduction

With the advancement of society and economy, traffic congestion is an inevitable common problem in big cities, which affects the development of cities and people’s travel safety. Traffic speed prediction is a basic and important function of the intelligent transportation system (ITS). Accurate and real-time traffic speed prediction can effectively save travel costs and provide a reliable traffic management support for traffic management and route recommendation. However, traffic forecasting has many challenges because of traffic changes in different locations throughout the different timestamps and weather and accidents, etc., and factors will also have an impact on traffic. Therefore, there are complex spatiotemporal relationships in traffic data.

Several temporal- and spatial-dependent prediction methods are reported in the existing literature. The analysis methods are represented by the time-series analysis model “auto-regressive integrated moving average” (ARIMA) [1, 2] and the Kalman filtering model [3, 4]. However, these models cannot effectively respond to the nonlinear and uncertain characteristics of the traffic data. Meanwhile, with the development of deep learning, numerous intelligent traffic prediction models have emerged [5, 6]. But, these models ignore the spatial characteristics in the traffic network. Several studies [7, 8] have used convolutional neural networks (CNN) to learn the spatial feature of road networks according to historical traffic maps. But, CNN is suitable for extracting Euclidean spatial features while the traffic road network involves a complex non-Euclidean spatial network. To solve this problem, graph convolution-based methods have been extensively developed recently [9, 10]. In summary, most of the existing literature considers only the traffic temporal and spatial correlations for traffic prediction, without considering the semantic information of the traffic road network; for example, similar functional areas have similar traffic patterns.

To better capture the complex spatiotemporal dependencies and semantic correlation hidden in traffic data, we
further propose a spatiotemporal DeepWalk gated recurrent unit model (ST-DWGRU), which can simultaneously learn the temporal, spatial, and semantic correlations of the road network. When validated using three real traffic datasets, the proposed model outperforms the state-of-the-art traffic forecasting baselines.

The main contributions of our work are as follows:

1. The ST-DWGRU model learns the spatial correlation between traffic speeds by GCN. The connectivity between traffic roads is represented by the adjacency matrix. Spatial correlation in traffic speed can be learned effectively by GCN.
2. GRU was introduced in order to learn the temporal correlation between traffic speed data. Traffic speed itself is time-series, and historical traffic speed data and future data have time-series relationship, and GRU is used to obtain the hidden time-series relationship between traffic speed.
3. Since the traffic patterns are different between different locations, DeepWalk is a position-aware graph embedding algorithm. DeepWalk can effectively obtain semantic information and enhance the accuracy of traffic speed prediction.
4. Spatial features, temporal features, and semantic features are fused to improve the predictive power of the model. Traffic speed data are affected by other factors besides spatial and temporal correlation, such as semantic correlation, and the ST-DWGRU model that incorporates semantic correlation can effectively improve prediction accuracy.

2. Literature Review

Traffic forecasting problem is to predict traffic indicators such as traffic volume, speed, and travel time at a certain location at a certain time. Forecasting methods can generally be divided into two types of approaches: model-based and data-driven forecasting methods. The model-based forecasting methods rely on queueing theory [11] and traffic velocity models [12]. These models require some assumptions and prior knowledge. However, in reality, there are many factors that can have an impact on traffic such as weather and unforeseen events. Therefore, it is difficult for the model-based methods to accurately predict.

The data-driven methods are more flexible because they discover patterns from historical data and automatically deduce intrinsic connections between data without requiring many assumptions. These methods are divided into statistical prediction and machine learning methods. The main statistical prediction methods include ARIMA, linear regression models, the Kalman filtering model, and exponential smoothing (ES). For example, Ahmed and Cook [13] first introduced ARIMA to the traffic-forecasting problem. Hamed and others [14] used a simple ARIMA model to predict traffic volumes on urban arterials with good results. Ding and others [15] proposed a spatiotemporal STARIMA model for predicting traffic volumes. Moreover, there are various variants of the ARIMA model for predicting traffic, such as Kohonen map ARIMA [16] and seasonal ARIMA [17]. Sun et al. [18] used a local linear predictor to address the issue of interval forecasting. Guo et al. [19] used an adaptive Kalman filtering model to predict traffic flow. Hinsbergen et al. [20] used an extended Kalman filtering model to estimate traffic state. Williams et al. [21] used seasonal ARIMA and winter’s ES model to perform traffic flow prediction.

Statistical prediction methods do not respond well to traffic uncertainty. Compared with the statistical methods, the machine learning methods are more flexible. The machine learning methods are mainly used to learn traffic patterns in the road network using large amounts of historical data, and the main methods include the K-nearest neighbor algorithm model (KNN) [22], SVM model [23–25], and neural network (NN) model [26]. KNN predicts traffic speed by the distance between features; however, it has high time complexity and space complexity. SVM uses kernel functions for traffic prediction, but it is relatively difficult to find the right kernel function.

Recently, deep learning methods have evolved rapidly. Huang et al. [27] used a deep belief network (DBN) to learn features, a multitask regression layer for traffic flow prediction. Tan et al. [28] used DBN based on restricted Boltzmann machines for traffic flow prediction. However, none of these methods considered long-time dependencies in traffic; therefore, to solve this problem, an RNN and its variants LSTM and GRU were used to learn temporal features in traffic. Tian and others [29] used LSTM NN to predict short-term traffic flow; Fu and others [5] used LSTM and GRU NN to predict traffic flow. All these methods consider only the temporal dependency but ignore the spatial dependency. Wu et al. [30] proposed a novel deep architecture that combined CNN and LSTM to predict traffic flow (CLTFP). A 1-dimension CNN is used to capture spatial features of traffic flow, and two LSTMs are used to mine short-term variability and traffic flow periodicities. Cao et al. [31] proposed an interactive temporal recurrent convolution network for traffic prediction, where the CNN part learns network traffic as images to capture network-wide services’ correlations, and the GRU part learns temporal features to help the interactive network traffic prediction. Although the above methods have made considerable progress in spatiotemporal learning, the CNNs are more effective in Euclidean space and cannot obtain spatial features more accurately for complex road network structures. Recently, with the development of graph CNN, it can acquire the spatial features of complex road networks more effectively. Yu et al. [32] proposed an ST-NCNN model for traffic prediction and achieved good results; Cui et al. [33] proposed a TGC-LSTM model to learn the interactions between roadways in the traffic network and proposed a network-wide traffic state; Zhao et al. [10] combined the GCN and GRU models for traffic prediction; Bogaerts et al. [34] combined the GCN and LSTM for long- and short-term traffic prediction using trajectory (GPS) data to achieve good results. However, the abovementioned methods only considered spatial and temporal dependencies, ignoring other
3. Methodology

3.1. Problem Definition. This study aims to predict future traffic speed after a time step \( T \) based on historical traffic speed information. To predict the traffic speed more accurately, this study not only considers the traffic-timing information and the spatial structure characteristics of the road network but also adds the semantic information of the road network, where similar functional areas have similar traffic patterns, as shown in the objective function.

\[
y_{t+T} = f(x_{t-h,...,t}; m_{t-h,...,t}; A),
\]

where \( T \) denotes the predicted time step, \( x_t \) denotes the historical traffic speed data, \( m_{t-h,...,t} \) denotes the semantic information of the road network, and \( A \) denotes the road network structure information. The objective of the model is to derive the function \( f \) from the complex temporal, spatial, and semantic information. As shown in Figure 1, the inputs to the model are the historical traffic speed data \( x_1,\ldots,x_t \), the road network structure matrix, and the final traffic semantic matrix.

3.2. Traffic Speed Prediction Based on ST-DWGRU. In this section, we elaborate on the architecture of the ensemble deep learning framework (ST-DWGRU). As shown in Figure 2, ST-DWGRU is composed of several ST-DWGRU layers, each of which contains three components: spatial feature extraction component, temporal feature extraction component, and semantic feature extraction component.

The final prediction component outputs the prediction results.

In addition, each layer of ST-DWGRU learns spatiotemporal feature and semantic feature, which is different from the previous prediction model. Since most graph neural networks compute node embedding by aggregating information from each node’s q-hop neighborhood and are thus structure-aware, GCNs cannot fully learn the location information of nodes [41]. The role of different location intersections in the road network is different, so we need to consider the position information embedded in the nodes to represent the semantic information of different intersections. The proposed ST-DWGRU learns spatiotemporal feature and semantic feature, which is more suited to the actual situation. The details of each component are described as follows.

3.3. Spatial Feature Extraction Component. A CNN extracts spatial effects by summing the weights of surrounding pixels, which is particularly effective for Euclidean space. However, a CNN cannot be directly used for extracting the spatial features of road network structure because the number of neighboring intersections or road segments around the intersection or road segment in the actual road network is not fixed. Deferrard et al. [42] defined the convolution operation for graph structures, based on spectral theory. The urban road network is considered an undirected graph \( G = (V, E) \), where \( V \) is the set of vertices in the graph and \( E \) is the set of the graph edges. Using the road network adjacency matrix as input, the graph convolution operation provides the road network structure features, and a two-layer GCN can be represented as [43].

\[
Z = f(X, A) = \text{softmax}(\tilde{A}\text{ReLU}(\tilde{A}XW^{(0)})W^{(1)}),
\]

where \( \tilde{A} = D^{-1/2}\tilde{A}D^{-1/2} \), \( \tilde{A} = A + I_N \) is the adjacency matrix of the road network with its connectivity, \( A \) is the adjacency matrix of the road network, \( D_{ij} = \sum A_{ij} \), \( W^{(0)} \) is the weight parameter of the first layer, and \( W^{(1)} \) denotes the weight parameter of the second layer.

3.4. Temporal Feature Extraction Component. GRU is an LSTM network variant with only two gates: an updated gate and resets gate, as compared with the input gate, output gate, and forget gate of LSTM. It has only two gates: update gate and reset gate. Let the input sequence be \( X = (x_1, x_2, \ldots, x_t) \) and then input to GRU to learn temporal features by GCN in...
The specific calculation formula is shown as follows.

\[ s_t = \text{GCN}(x_t), \]
\[ r_t = \sigma(W_r \cdot [h_{t-1}, s_t \cdot x_t] + b_r), \]
\[ z_t = \sigma(W_z \cdot [h_{t-1}, s_t \cdot x_t] + b_z), \]
\[ \tilde{h}_t = \tanh(W_h \cdot [r_t \odot h_{t-1}, s_t \cdot x_t] + b_h), \]
\[ h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t, \]

where \( s_t \) is the GCN output at time \( t \), \( h_t \) is the hidden state at time \( t \), \( x_t \) is the current input, \( h_{t-1} \) is the hidden state at the previous time, \( r_t \) is the reset gate, \( z_t \) is the update gate, \( \tilde{h}_t \) is the current memory content, and \( h_t \) is the current hidden state. Where \( W \) and \( b \) denote the weights and biases of the network, respectively, \( \sigma \) denotes the sigmoid activation function, and \( \tanh \) is the tanh activation function. Therefore, the final GRU output has temporal and spatial features.

3.5. Semantic Feature Extraction Component. Generally, similar functional attribute areas exhibit similar traffic patterns. For example, traffic speed near city parks will be lower on weekends, while industrial parks will have smaller traffic speed during morning and evening peak traffic periods. Therefore, the traffic pattern correlation for similar areas is relatively high, and this study uses the graph embedding technique to learn the representation of similar functions.

DeepWalk [44] is a position-aware node embedding representation learning algorithm used widely recently [41]. The algorithm is divided into two main steps: random walk and update procedure. A path with the root node is \( s_i \) denoted by \( W_s \), and the nodes in the path are labeled as \( \{W_1^i, W_2^i, W_3^i, \ldots, W_k^i\} \), where \( W_k^i \) denotes the \( k \)th intersection or section in the path. All random walks are the same length in a truncated random walk, and the random walk sequence matrix of the entire path is obtained after traversing all intersections or road sections. The corresponding vector representation is \( s_i \) obtained from the Skip-gram algorithm, and the optimization objective is

\[ \min_{\phi} - \log p(v_i \mid v_{i+w_1} \ldots v_{i+w_d}) \phi(v_i). \]

This study obtains the final semantic information representation through a fully connected layer after obtaining the vector representation.
\[ \hat{s} = f(W_i \cdot s_i + b_i), \]  
(5)

where \( W_i \) and \( b_i \) are the learnable weights and biases.

### 3.6. Prediction Component.

This study predicts traffic speed change information in the future based on the historical traffic speed information. Because the output of GRU has spatial and temporal features, the outputs of the three components are stitched together, and the concatenate operation is expressed as follows:

\[ o^i_t = h^i_t \odot \hat{s}_i, \]  
(6)

The concatenated result is then passed through a fully connected network to obtain the predicted output of the model \( y^i_{i+T} \), where \( T \) is the predicted time step, of the following form:

\[ y^i_{i+T} = \text{sigmoid}(W_s o^i_t + b_s), \]  
(7)

where \( W_s \) and \( b_s \) are the learnable weights and biases.

## 4. Experiments

### 4.1. Dataset Description.

In the experiments of this paper, three real-world traffic datasets are used for the experiments. They are PeMSD4, PeMSD8 [45], and PeMS-BAY [46]. The three datasets are collected by California Performance of Transportation (PeMS). Details of the dataset are shown in Table 1. The sensor network of PeMS is shown in Figure 3.

PeMSD4 [45]: PeMSD4 has 307 detectors distributed across 29 roadways spanning January to February 2018. The dataset was collected from Caltrans Performance Measurement System (PeMS) and traffic data from the San Francisco Bay Area.

PeMSD8 [45]: PeMSD8 has 170 detectors distributed on 8 roads, between July and August 2016 in San Bernardino.

PeMS-BAY [46]: It is collected from Caltrans PeMS too, which has 325 sensors in the Bay Area, spanning from Jan 1, 2017, to May 31, 2017.

### 4.2. Experimental Settings.

A PC (CPU: Intel(R) Xeon(R) CPU E5-2620 v4 @ 2.10 GHz, Memory: 64 GB, GPU: NVIDIA TiTAN XP) was used as the experimental platform, and TensorFlow1.14 was used to build the model.

In this study, the data are divided into two parts, respectively, 80% of the data as the training set, and the remaining as the test set. Predictions are performed for 15, 30, and 60 min, respectively.

The model’s hyperparameters include learning rate, training rounds, number of hidden units, window size, and length of prediction time, where the learning rate is 0.001, the number of training epoch is 800, the number of hidden units is 64, the window size is 10, and the prediction time is 15, 30, and 60 min, respectively.

### 4.3. Evaluation.

The root mean square error (RMSE), mean absolute error (MAE), and mean absolute percentage errors (MAPE) are used to evaluate the model. RMSE, MAE, and MAPE are defined as

\[
\text{RMSE} = \sqrt{\frac{1}{m} \sum_{i=1}^{m} (Y_i - \bar{Y}_i)^2},
\]

\[
\text{MAE} = \frac{1}{m} \sum_{i=1}^{m} |Y_i - \bar{Y}_i|,
\]

\[
\text{MAPE} = \frac{1}{m} \sum_{i=1}^{m} \left| \frac{Y_i - \bar{Y}_i}{Y_i} \right|,
\]

where \( m \) denotes the number of road sections and \( Y_i, \bar{Y}_i \) denote the true and predicted values of speed, respectively.

### 4.4. Baselines.

The baseline methods for comparison with the ST-DWGRU are as follows:

(i) HA: historical average

(ii) ARIMA [17]: auto-regressive integrated moving average, which is often used in time series prediction

(iii) STGCN [32]: spatiotemporal graph convolutional networks (STGCN), which combines graph convolution with gated CNNs

(iv) DCRNN [47]: diffusion convolutional recurrent neural network (DCRNN), which combines bidirectional random walks with gated recurrent units

(v) GWN [48]: graph neural network GraphWaveNet, which consists of stacked spatial-temporal layers and an output layer

(vi) ASTGCN [49]: attention-based spatial-temporal graph convolutional network (ASTGCN), which mainly consists of three independent components to, respectively, model recent, daily-periodic, and weekly periodic dependencies

(vii) LSGCN [50]: long short-term graph convolutional networks, which integrate both cosAtt and graph convolution networks (GCN) to handle spatial dependency and use gated linear units convolution (GLU) to capture complex temporal feature

(viii) USTGCN [51]: unified spatiotemporal graph convolution network, which performs both spatial and temporal aggregation through spectral graph convolution on a spatiotemporal graph

### 4.5. Experiment Results.

As shown in Table 2, the ST-DWGRU model performs the best in both short- and long-term predictions on all evaluation metrics. In particular, all metrics are optimal on the PeMSD4 dataset, while MAE is the second best on the PeMS-BAY dataset, the GWN model is the best, RMSE is second only to the USTGCN model on the PeMSD8 dataset, and the other metrics are still the best.
Table 1: The details for PeMSD4, PeMSD8, and PEMS-BAY.

<table>
<thead>
<tr>
<th>Model</th>
<th>PeMSD4</th>
<th>PeMSD8</th>
<th>PEMS-BAY</th>
</tr>
</thead>
<tbody>
<tr>
<td>#Nodes</td>
<td>307</td>
<td>170</td>
<td>325</td>
</tr>
<tr>
<td>#Edges</td>
<td>340</td>
<td>295</td>
<td>2369</td>
</tr>
<tr>
<td>#Time steps</td>
<td>16992</td>
<td>17856</td>
<td>52116</td>
</tr>
<tr>
<td>Time interval</td>
<td>5 mins</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Daily range</td>
<td>0:00–24:00</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Performance comparison of ST-DWGRU and other approaches on the datasets PeMSD4, PeMSD8, and PEMS-BAY.

<table>
<thead>
<tr>
<th>Model</th>
<th>PeMSD4 (15/30/60 min)</th>
<th>PeMSD8 (15/30/60 min)</th>
<th>PEMS-BAY (15/30/60 min)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MAE RMSE MAPE (%)</td>
<td>MAE RMSE MAPE (%)</td>
<td>MAE RMSE MAPE (%)</td>
</tr>
<tr>
<td>HA</td>
<td>2.54/4.96/5.56</td>
<td>5.72/3.75/4.12</td>
<td>5.32/6.56/6.96</td>
</tr>
<tr>
<td>ARIMA (2003)</td>
<td>1.90/2.75/3.21</td>
<td>2.94/4.06/5.28</td>
<td>2.68/3.71/5.10</td>
</tr>
<tr>
<td>DCRNN (2018)</td>
<td>1.19/1.77/2.26</td>
<td>3.01/4.21/5.65</td>
<td>2.92/3.98/5.39</td>
</tr>
<tr>
<td>STGCN (2018)</td>
<td>1.17/2.42/2.73</td>
<td>3.96/4.59/5.21</td>
<td>4.16/4.80/5.46</td>
</tr>
<tr>
<td>ASTGCN (2019)</td>
<td>1.00/1.70/2.03</td>
<td>2.68/3.82/4.65</td>
<td>2.67/3.73/4.60</td>
</tr>
<tr>
<td>GWN (2019)</td>
<td>1.35/1.82/2.22</td>
<td>2.93/3.92/4.83</td>
<td>2.90/3.84/4.85</td>
</tr>
<tr>
<td>LSGCN (2020)</td>
<td>1.40/1.64/2.03</td>
<td>2.69/3.19/4.25</td>
<td>2.81/3.23/4.32</td>
</tr>
<tr>
<td>USTGCN (2021)</td>
<td>1.20/1.48/1.90</td>
<td>2.40/3.12/4.01</td>
<td>2.21/2.75/3.53</td>
</tr>
<tr>
<td>ST-DWGRU (ours)</td>
<td>1.00/1.25/1.57</td>
<td>2.08/2.70/3.49</td>
<td>1.81/2.24/2.78</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Model</th>
<th>PeMSD4 (15/30/60 min)</th>
<th>PeMSD8 (15/30/60 min)</th>
<th>PEMS-BAY (15/30/60 min)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MAE RMSE MAPE (%)</td>
<td>MAE RMSE MAPE (%)</td>
<td>MAE RMSE MAPE (%)</td>
</tr>
<tr>
<td>HA</td>
<td>1.98/4.11/3.94</td>
<td>4.11/5.24/6.22</td>
<td>5.11/5.21/5.62</td>
</tr>
<tr>
<td>ARIMA (2003)</td>
<td>1.90/2.12/2.79</td>
<td>2.59/3.56/4.50</td>
<td>2.32/3.21/4.28</td>
</tr>
<tr>
<td>DCRNN (2018)</td>
<td>1.17/1.49/1.87</td>
<td>2.62/3.61/4.68</td>
<td>2.34/2.44/5.45</td>
</tr>
<tr>
<td>STGCN (2018)</td>
<td>1.19/1.59/2.25</td>
<td>3.18/3.69/4.13</td>
<td>3.16/3.59/4.22</td>
</tr>
<tr>
<td>ASTGCN (2019)</td>
<td>1.49/1.67/1.89</td>
<td>2.45/3.28/4.11</td>
<td>2.24/3.02/3.89</td>
</tr>
<tr>
<td>GWN (2019)</td>
<td>1.16/1.46/1.81</td>
<td>2.15/2.58/3.27</td>
<td>2.07/2.35/3.22</td>
</tr>
<tr>
<td>LSGCN (2020)</td>
<td>1.14/1.25/1.70</td>
<td>2.08/2.70/3.49</td>
<td>1.81/2.24/2.78</td>
</tr>
<tr>
<td>USTGCN (2021)</td>
<td>1.00/1.25/1.57</td>
<td>2.08/2.70/3.49</td>
<td>1.81/2.24/2.78</td>
</tr>
<tr>
<td>ST-DWGRU (ours)</td>
<td>1.00/1.25/1.57</td>
<td>2.08/2.70/3.49</td>
<td>1.81/2.24/2.78</td>
</tr>
</tbody>
</table>

Bold is the best; underline is the second best.
From the results, it is shown that the ST-DWGRU model can capture spatiotemporal features and semantic features effectively for traffic prediction.

In contrast, traditional statistical methods such as HA and ARIMA perform the worst in short- and long-term prediction on three datasets because they cannot effectively capture complex spatiotemporal and semantic features.

Figure 4: Trends in MAE and RMSE on PeMSD4, PeMSD8, and PeMS-BAY. (a) MAE of PeMSD4. (b) RMSE of PeMSD4. (c) MAPE of PeMSD4. (d) MAE of PeMSD8. (e) RMSE of PeMSD8. (f) MAPE of PeMSD8. (g) MAE of PeMS-BAY. (h) RMSE of PeMS-BAY. (i) MAPE of PeMS-BAY

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a)</td>
<td>MAE of PeMSD4</td>
</tr>
<tr>
<td>(b)</td>
<td>RMSE of PeMSD4</td>
</tr>
<tr>
<td>(c)</td>
<td>MAPE of PeMSD4</td>
</tr>
<tr>
<td>(d)</td>
<td>MAE of PeMSD8</td>
</tr>
<tr>
<td>(e)</td>
<td>RMSE of PeMSD8</td>
</tr>
<tr>
<td>(f)</td>
<td>MAPE of PeMSD8</td>
</tr>
<tr>
<td>(g)</td>
<td>MAE of PeMS-BAY</td>
</tr>
<tr>
<td>(h)</td>
<td>RMSE of PeMS-BAY</td>
</tr>
<tr>
<td>(i)</td>
<td>MAPE of PeMS-BAY</td>
</tr>
</tbody>
</table>
The GWN model uses stacked spatiotemporal layers to handle spatial dependencies at different temporal levels, while the DCRNN uses diffusion convolution to effectively obtain complex spatial dependencies. Besides, it uses bidirectional graph random walk in combination with a sequence-to-sequence learning framework and scheduled sampling to obtain long-term temporal dependency. Because of the cumulative error of STGCN, STGCN does not perform as well as GWN and DCRNN.

The USTGCN performs both spatial and temporal aggregation through spectral graph convolution on a spatiotemporal graph. In addition to the relationship of spatiotemporal, USTGCN considers the important historical and current-day pattern. So, it performs better than GWN and LSGCN in the short- and long-term prediction of PeMSD4 and PeMSD8.

The ST-DWGRU model has the best overall performance on three datasets in addition to the second best MAE on the PeMS-BAY and RMSE on the PeMSD8. It is implied that in traffic prediction, in addition to spatiotemporal correlations, semantic correlation also has an important impact.

As can be seen from Figure 4, the errors of all models grow in the 15-, 30-, and 60-minute predictions on the three datasets, except for the GWN model in which the 60-minute MAE is smaller than the 30-minute on the PeMS-BAY dataset. However, the ST-DWGRU model has more flat growth. This indicates that the ST-DWGRU model is more robust on the short- and long-term forecasts.

Especially in the extreme moments of the morning and evening peaks, as the ST-DWGRU model can effectively obtain the spatiotemporal features and semantic features, the ST-DWGRU model can well predict the changes of traffic in extreme moments. The ST-DWGRU model can accurately predict extreme situations, as shown in Figure 5.

4.6 Parameter Sensitivity Analysis. As shown in Figure 6, to determine the number of hidden units, we experimentally validate the hidden units as 8, 16, 32, 64, and 128, respectively. The results are shown in Figure 6, where the X-axis
represents the number of hidden units and the Y-axis represents the values of RMSE and MAE. As shown in the figure, when the number of hidden units is 64, the values of RMSE and MAE are minimum.

To determine how different DeepWalk window size in short- and long-term prediction, we have listed the performance of ST-DWGRU with different DeepWalk window size (W) in Table 3. In Table 3, we can observe that in short- and long-term prediction, the performance of ST-DWGRU with $W=10$ is the best. With the increase of DeepWalk window size, it is not that the ST-DWGRU model is getting better. When $W=10$, the ST-DWGRU model is more accurate in obtaining semantic information.

5. Conclusion

This paper proposes an urban road network traffic speed prediction model, which explores the potential spatio-temporal relationships and semantic information in the traffic speed data. The model is validated by applying three public datasets. Experimental results verify that compared with traditional HA and ARIMA prediction methods, the ST-DWGRU model has better prediction performance. Compared with the state-of-the-art traffic prediction methods DCRNN, STGCN, ASTGCN, GWN, LSGCN, and USTGCN, the ST-DWGRU model is also performed better. Due to the complexity of traffic, future research will focus on combining the attention mechanism, considering the influence of key road sections or intersections on traffic speed, and further exploring the effective acquisition of spatio-temporal relationships and the interpretability of the model in complex networks.

Data Availability

Previously reported traffic data used to support this study are available. These prior studies (and datasets) are cited at relevant places within the text as references [45, 46].

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was financially supported by the project supported by the “National Natural Science Foundation of China” (61977001).

References


