Deep Multi-View Channel-Wise Spatio-Temporal Network for Traffic Flow Prediction

Hao Miao[1], **Senzhang Wang**[1], **Meiyue Zhang**[1], **Diansheng Guo**[2], **Funing Sun**[2], **Fan Yang**[2]
[1]Nanjing University of Aeronautics and Astronautics, [2]Tencent
[1]{miaohao,szwang,meiyuezhang}@nuaa.edu.cn, [2]{dguo,funingsun,wendellyang}@tencent.com

Abstract

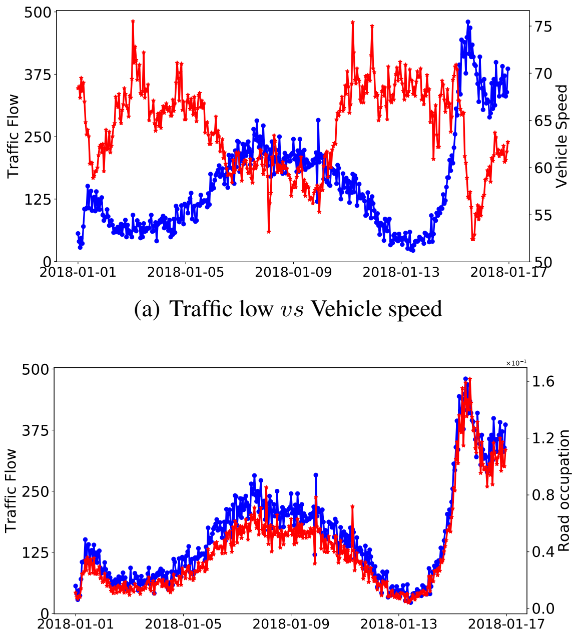
Accurately forecasting traffic flows is critically important to many real applications including public safety and intelligent transportation systems. The challenges of this problem include both the dynamic mobility patterns of the people and the complex spatial-temporal correlations of the urban traffic data. Meanwhile, most existing models ignore the diverse impacts of the various traffic observations (e.g. vehicle speed and road occupancy) on the traffic flow prediction, and different traffic observations can be considered as different channels of input features. We argue that the analysis in multiple-channel traffic observations might help to better address this problem. In this paper, we study the novel problem of multi-channel traffic flow prediction, and propose a deep Multi-View Channel-wise Spatio-Temporal Network (MVC-STNet) model to effectively address it. Specifically, we first construct the localized and globalized spatial graph where the multi-view fusion module is used to effectively extract the local and global spatial dependencies. Then LSTM is used to learn the temporal correlations. To effectively model the different impacts of various traffic observations on traffic flow prediction, a channel-wise graph convolutional network is also designed. Extensive experiments are conducted over the PEMS04 and PEMS08 datasets. The results demonstrate that the proposed MVC-STNet outperforms state-of-the-art methods by a large margin.

Introduction

As a typical spatio-temporal prediction task, traffic prediction has drawn considerable research attention in recent years due to the increasing amount of urban traffic and its significant impacts on real-world application. Accurate traffic forecasting is particularly useful to support Intelligent Transportation Systems (ITS), and can facilitate many real applications such as travel route planning(Huang, Xu, and Weng 2020), and travel time estimation(Wang et al. 2018).

Traditional statistics-based prediction approaches, such as ARIMA (Williams and Hoel 2003) and VAR (Chandra and Al-Deek 2009) have been widely used in road segment-level traffic flow prediction, and achieved promising performance. Predicting the traffic flows over a large road network, however, is a much more difficult task due to the very complex and non-linear spatio-temporal dependencies among the traffic data over different road links. Thus statistics-based models become less effective to handle the road network-level traffic prediction. With the recent advances of deep learning techniques, various deep learning models (Wang, Cao, and Yu 2020) are employed for traffic flow prediction and has achieved remarkable performance gains. ST-ResNet (Zhang, Zheng, and Qi 2017) was proposed to collectively forecast the inflow and outflow in each region of a city. (Yao et al. 2019) proposed a Spatio-Temporal Dynamic Network (STDN) model for road network based traffic prediction. Diffusion convolution recursive neural network (DCRNN) (Li et al. 2018) integrated diffusion convolution and Seq2Seq structure for traffic flow prediction. Graph WaveNet (Wu et al. 2019) combined GCN with dlated casual convolution to capture the spatial-temporal dependencies of the traffic flows. STGCN (Yu, Yin, and Zhu 2018) used ChebNet graph convolution and 1D convolution to forecast the traffic flows in each road of a road network. GMAN (Zheng et al. 2020) proposed to utilize the attention mechanism to effectively extract spatial and temporal features for traffic prediction.

However, existing works mainly focus on capturing the local spatial correlations of the traffic data, which follows the First Law of Geography (Tobler 1970): "*Near things are more related than distant things*", but cannot fully reflect the global spatial dependencies. For example, previous work (Yao et al. 2018) showed that two regions with similar POI distribution or functionality(i.e., both commercial area), even though they are not geographically close to each other, can present very similar patterns on the semantic space of the spatio-temporal data (e.g., traffic flow). Such global spatial dependencies are not carefully considered by existing deep learning based approaches. Another limitation of existing methods is that the diverse impacts of the traffic observations on the traffic flow prediction task is largely ignored. As shown in Figure 1, different types of traffic observations may have different impacts on traffic flows. Figure 1(a) shows the relationship between traffic flow and traffic speed. From the picture1(a), one can see that traffic flow and speed are negatively correlated. That is to say the traffic speed will slow



(a) Traffic low *vs* Vehicle speed

(b) Traffic flow *vs* Road occupancy

Figure 1: Relationship between traffic flow and two traffic observation features (vehicle speed and road occupancy)

down with the increase of traffic flows, and vice versa. Figure 1(b) shows the relationship between traffic flow and the road occupancy where the change of road occupancy follows the change of traffic flows. Both traffic speed and road occupancy have impacts on traffic flows, but the two relationships are quite different which is like the idea in MMRate(Wang et al. 2014). However, it is not considered in previous works how to model the influence of different traffic observation on the task of traffic flow prediction.

To address the above issues, in this paper we propose a deep multi-view channel-wise spatial-temporal network model named MVC-STNet for traffic flow prediction. To model the different relations between various input channels representing traffic flow, road occupy and vehicle speed, and the prediction, channel-wise graph convolutional network (CGCN) is proposed. CGCN can learn the data representations of each channel first, and then fuse them in a parametric-matrix-based way (Zhang, Zheng, and Qi 2017). To effectively capture the local and global spatial correlations, localized and globalized spatio-temporal graphs are constructed to learn the two spatial representations, and then a multi-view fusion module is proposed to fuse the localized and globalized data representations. Additionally, LSTM is also used to learn the sequential dependency of the traffic flows. Considering the external context features including holidays and weather can also significantly influence traffic flows, the external features are also incorporated into our model. We summarize our main contributions as follows:

- A novel deep learning framework MVC-STNet is presented to perform spatio-temporal knowledge extraction for traffic flow prediction. By considering the spatial and temporal features exhaustively, the proposed model can effectively capture the complex spatial and temporal correlations.
- We propose a channel-wise graph neural network that can adaptively learn the different influences of input channels on traffic prediction. To the best of our knowledge, this is the first graph neural network based model for traffic flow prediction that considers the divergence between different input channels.
- We conduct experiments on two real world traffic datasets. The results show that our model is consistently and significantly better than existing state-of-the-art methods.

The remainder of the paper is organized as follows. Section 2 will review related work. Section 3 will give a formal problem definition. Section 4 will first show the model framework and then introduce the model in detail. Evaluations are given in Section 5, followed by the conclusion in Section 6.

Related Work

This work is highly relevant to the research topics of traffic flow prediction and graph convolutional network. Next, we will review related works from the above two aspects.

Traffic flow prediction. As a typical spatio-temporal prediction task, traffic flow prediction has been studied for decades in intelligent transportation systems (Wang, Cao, and Yu 2020; Yao et al. 2019). Traditionally, statistic-based time series prediction models such as ARIMA (Williams and Hoel 2003) and SVR (Castro-Neto et al. 2009) are widely used for predicting traffic flows on a single road. Due to the limited learning capacity, these statistic-based approaches cannot effectively capture the complex spatio-temporal dependencies of the traffic data over a large road network. Recently, various deep learning based methods are broadly applied in traffic flow prediction and these models have achieved much better performance than traditional statistic-based shallow models, such as ST-ResNet (Zhang, Zheng, and Qi 2017) and ConvLSTM (Xingjian et al. 2015). SeqST-GAN (Wang et al. 2020) was proposed to perform multi-step traffic flow prediction of a city in an adversarial learning way. Above mentioned works mainly applied CNN to capture the spatial correlation by treating the traffic data of an entire city as images, or combined CNN and RNN models to capture both the spatial and temporal correlations.

However, CNN based models cannot well model the semantic spatial correlation of the traffic data and the global spatial structure of a city. To address this issue, some recent works try to use graph neural network to perform road network level traffic prediction. (Li et al. 2018) proposed the diffusion Convolutional Recurrent Neural Network (DCRNN) to model the traffic flow as a diffusion process on a directed road graph. A Spatio-Temporal Graph Convolutional Networks was proposed to tackle the time series prediction problem in traffic forecasting. (Zheng et al. 2020) proposed a graph multi-attention network (GMAN) to predict traffic conditions for time steps ahead at different locations on a road network graph. However, complex spatial dependencies which contains local and global spatial features are not well considered in these models.

Graph Convolutional Network. Graph convolutional network (GCN) was widely studied in recent years (Qu, Bengio, and Tang 2019). (Estrach et al. 2014) first proposed the graph convolution operation in Fourier domain through the graph Laplacian. Then Chebyshev expansion of the graph Laplacian was employed to improve the inference efficiency (Defferrard, Bresson, and Vandergheynst 2016). (Kipf and Welling 2017) simplified the convolution operation which only aggregated the node features from their neighbors. GAT (Veličković et al. 2018) introduced the attention mechanism to aggregate node features with the learned wights. GraphSAGE (Hamilton, Ying, and Leskovec 2017) proposed a general, inductive framework that leveraged node feature information (e.g., text attributes) to efficiently generate node embedding. However, these GCN based models treat the input features equally, but ignore the different impacts of node features on prediction tasks.

Problem Formulation

In this section, we will first give some definitions to help us state the studied problem. Then a formal problem formulation will be given.

Definition 1 ***Traffic network*** *$\mathcal{G}$ we denote the traffic network at time slot t as $G^t = \{V^t, E^t, A^t\}$, where $|V^t| = N$ is the set of vertices, N denotes the number of vertices that represent road segments in real world, E^t denotes the set of edges, indicating the connectivity between the nodes, for example, if two segments in real world are connected, an edge exists between the corresponding nodes. Note that the traffic network can be either directed or undirected, and A^t represents the adjacency matrix.*

Definition 2 ***Traffic network feature matrix*** *$\mathcal{X}$ we define the traffic network feature matrix at time slot t as $X^t \in \mathcal{R}^{N\times C}$, where C is the dimension of the node features. The traffic network feature matrix represents the observations of $\mathcal{G}$ at the time step t.*

Problem Definition 1 *Given the spatio-temporal traffic network and graph feature matrix $\{X^t, G^t | t = t_1, \dots, t_T\}$ over T time slots, and external context data matrix E (e.g., weather, holiday, etc.), our goal is to to predict the traffic flow Y^t in the next time slot.*

Deep Multi-View Channel-wise Spatio-Temporal Network

Figure 2 shows the framework of the proposed MVC-STNet model. As shown in the figure, the model contains four major steps. First, we split the raw input traffic network data into n parts with each part indicating a channel. The local and global spatial-temporal graphs are constructed from the geographical and semantic views, respectively. Next, several channel-wise GCN (CGCN) layers are stacked to learn the hidden feature of each channel which represents traffic flow, vehicle speed or road occupancy. From the picture 1, we assume that different channels may have different impacts on task prediction. The stacked CGCN layers aim to capture the spatial dependencies of the data and map the channel-wise data into a high-dimensional embedding latent space. Meanwhile, it also can adaptively learn the hidden relations between input channels and prediction. Third, after the multi-view fusion, the learned spatial features are input into stacked LSTM layers to learn the temporal dependencies. Finally, we concatenate the learned spatio-temporal representations and the learned external features for prediction. Next, we will elaborate these steps in detail in the following subsections.

Local and Global Spatial-Temporal Graph Construction

We intend to build a model that can directly capture the influence of each node on its local neighbors and global neighbors. We use $A_g \in \mathcal{R}^{N\times N}$ and $A_s \in \mathcal{R}^{N\times N}$ to denote the local and global adjacency matrix of the spatial graph, respectively. Based on the first Law of Geography (Tobler 1970): "*Near things are more related than distant things*", we first construct the local spatial graph. If two nodes are connected with each other geographically, there is an edge between them. The corresponding value in the adjacency matrix is set to be the reciprocal of the distance between the two nodes. The geographical adjacency matrix can be formulated as follows:

$$A_g^{i,j} = \begin{cases} \frac{1}{dis}, if\ v_i\ connects\ to\ v_j \\ 0,\ otherwise \end{cases} \tag{1}$$

where dis denotes the geographical distance between two nodes v_i and v_j.

However, the First Law of Geography may not fully reflect the spatial correlations of the traffic flows in urban areas that is known as the global spatial correlations or semantic spatial correlations. For example, a commercial area may have few traffic flows coming from a park, although they are geographically close to each other; while a residential district far away may have a large number of people flowing into the commercial area. To tackle this problem, we construct the adaptive global adjacency matrix which was used in AGCRN (Bai et al. 2020). First, we randomly initialize a learnable node embedding: $E_A \in \mathcal{R}^{N\times d_e}$ for all nodes, where d_e denotes the dimension of node embedding, and each row of E_A represents the embedding of a specific node. Then we can infer the semantic spatial dependencies between each pair of nodes by multiplying E_A and E_A^T as follows:

$$A_s = Softmax(ReLU(E_A E_A^T) \tag{2}$$

where $Softmax$ is used to normalize the adaptive matrix. Note that the globalized adjacency matrix changes over time. This process is similar as constructing the graph based on nodes similarity.

Multi-View Channel-wise Graph Convolutional Network

The localized spatial graphs and the globalized spatial graphs are next input into the Multi-View Channel-wise Graph Convolutional Network for modeling the geographical and semantic spatial correlations. As the input features

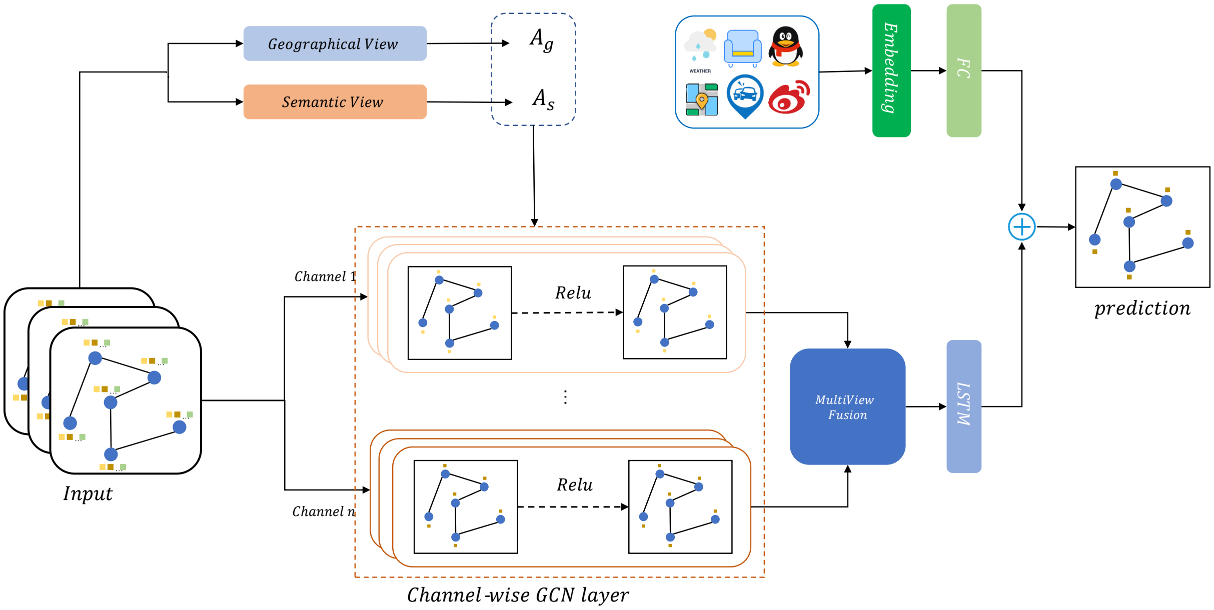


Figure 2: The framework of MVC-STNet.(Relu is a non-linear activation function)

are treated equally by the traditional GCN (Kipf and Welling 2017), the different influences of various features on the prediction is ignored. To address this issue, we propose a channel-wise graph convolutional network (CGCN) to first learn the data representation of each channel separately, and then fuse them together. Moreover, we propose a multi-view fusion module for modeling both localized and globalized spatial correlations, which can be formulated as follows:

$$\begin{aligned} H_g^t &= CGCN(X^t, A_g) \\ H_s^t &= CGCN(X^t, A_s) \\ H_{mv}^t &= H_g^t + H_s^t \end{aligned} \tag{3}$$

where H_g^t, H_s^t and H_{mv}^t are localized spatial hidden features, globalized spatial hidden features and multi-view fused spatial features respectively, $CGCN(\cdot)$ denotes the proposed channel-wise graph convolution network.

Channel-wise Graph Convolutional Network Recently, generalizing convolutional networks to graph data have attracted considerable research interest. In this paper, we consider to use spectral convolutions (Kipf and Welling 2017) on the constructed spatial graphs, which can be formulated as follows:

$$f(X^t, A) = \sigma(D^{-\frac{1}{2}}\widetilde{A}D^{-\frac{1}{2}}X^tW^t) \tag{4}$$

where $f(\cdot)$ represents the GCN operation, $\widetilde{A} = A + I_N$ is the adjacency matrix of $\mathcal{G}$ with added self-connections, $\widetilde{D}_{ii} = \sum_j \widetilde{A}_{ij}$ is the degree matrix , W^t represents the learnable weight matrix, and $\sigma(\cdot)$ is the activation function.

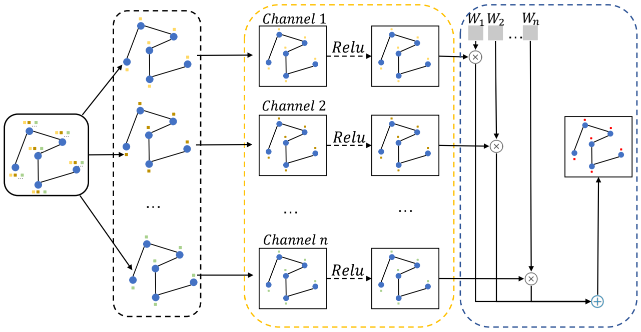


Figure 3: The illustration of channel-wise graph convolutional network.(Relu is a non-linear activation function)

However, the above GCN operation cannot model the different influences of various input features on the final prediction. To address this issue, we propose a channel-wise graph convolutional network (CGCN) to separately model relations between each input features and output. Figure 3 shows the framework of the proposed CGCN. We first split the input graph features $X^t \in \mathcal{R}^{N\times C}$ by channels to get the channel-wise spatial graph features $X_i^t \in \mathcal{R}^{N\times 1}$. Then we input the X_i^t into the stacked GCN layer which is shared by all channel-wise graph features. The learned i-th localized spatial hidden features can be formulated as follows:

$$H_{i,g,l+1}^t = \sigma(D^{-\frac{1}{2}}\widetilde{A_g}D^{-\frac{1}{2}}H_{i,g,l}^tW_l^t) \tag{5}$$

where $H_{g,0}^t$ is equal to X_i^t, W_l^t is the a trainable matrix of filter parameters in the l-th graph convolutional layer. We use $CGCN_i(X_i^t, A_g)$ to denote the localized GCN operation. Similarly, the globalized graph convolutional operation for $i\text{-}th$ channel can be defined as:

$$H^t_{i,s,l+1} = \sigma(D^{-\frac{1}{2}} A_s D^{-\frac{1}{2}} H^t_{i,s,l} W^t_l) \tag{6}$$

We use $CGCN_i(X_i^t, A_s)$ to represent the globalized GCN operation.

As shown in Figure 1, with the change of traffic flow, the speed and occupancy show different trends. Inspired by this observation, We propose a novel channel-fusion method to fuse the latent representations of all channel features. This method aims to automatically learn the relations between outputs and each input features. We employ the parametric-matrix-based (Zhang, Zheng, and Qi 2017) fusion method to fuse the n outputs of stacked CGCN layers as follows:

$$\begin{aligned} H^t_g &= W_1 \odot H^t_{1,g} + W_2 \odot H^t_{2,g} + \cdots + W_n \odot H^t_{n,g} \\ H^t_s &= W_1 \odot H^t_{1,s} + W_2 \odot H^t_{2,s} + \cdots + W_n \odot H^t_{n,s} \end{aligned} \tag{7}$$

where $W_i (i \in \{1, \ldots, n\})$ are the learnable parameters that adjust the degrees affected by n channels, $H^t_{\alpha\beta}(\alpha \in \{1, \ldots, n\}, \beta \in \{g, s\})$ are the learned spatial representations of n channels' feature.

Learning Temporal Dependencies with LSTM

Besides the spatial correlations, traffic flow forecasting also involves complex temporal correlations. We input the extracted spatial features into LSTM layers as follows for temporal feature learning.

$$\begin{aligned} & f^t = \sigma(W_f[h^{t-1}, H^t_{mv}] + b_f), \\ & i^t = \sigma(W_i[h^{t-1}, H^t_{mv}] + b_i), \\ & c^t = f_t \odot c^{t-1} + i_t \odot tanh(W_c[h^{t-1}, H^t_{mv}] + b_c), \\ & o^t = \sigma(W_o[h^{t-1}, H^t_{mv}] + b_o), \\ & h^t = o^t \odot tanh(c^t). \end{aligned} \tag{8}$$

where f^t, i^t, c^t, o^t, h^t are forget gate, input gate, memory cell, output gate and hidden state, respectively.

Integrating the External Features

External context features may significantly affect the traffic flows. For example, the patterns of traffic on weekdays and weekends can be quite different, while rainstorms may dramatically decrease the traffic flows. By considering the external context features including weather conditions and holidays, we design an external feature extractor which is defined as :

$$e^t = FC(Embedding(E^t)) \tag{9}$$

where e^t is the learned external feature representation, $FC(\cdot)$ represents the multi-layer perception model. Finally, we concatenate the learned spatial-temporal graph representations and external feature representation for traffic flow prediction.

$$\hat{Y}^t = FC(Concat(h^t, e^t)) \tag{10}$$

where $Concat(\cdot)$ denotes the concatenation operation.

Algorithm 1 Deep Multi-View Channel-wise Spatio-Temporal Network

```
Input: $\mathcal{G}$: Spatial-temporal traffic network $X$: Spatial-
    temporal graph signal matrix
Output: Parameter set $\Theta$
Initialize parameters $\Theta$
while $not\ converge$ do
    $0 \leftarrow t$, $t$ is the $t\text{-}th$ time slot
    while $t < T$ do
        Sample $X^T \in X, \mathcal{G}^t \in \mathcal{G}$
        $H^t_{mv}$ ← Learning spatial correlations with
            multi-view channel-wise GCN through Eq. 3
        $h_t$ ← Learning temporal correlations with
            LSTM by Eq. 8
        $e^t$ ← External feature learning by Eq.9
        $\hat{Y}^t$ ← Predict traffic flows with Eq.10
        update $\Theta$ based on Objective function
        $t \leftarrow t + 1$
    end while
    return $\Theta$
end while
```

Table 1: Dataset Description

Dataset	PeMS04	PeMS08
Area	San Francisco Bay	San Bernaridino
Detectors	3848	1979
Time span	01/01/2018∼28/02/2018	01/07/2016∼31/08/2016
Time interval	5 mins	5 mins
Nodes	307	170
# of time intervals	16992	17847
External Features		
Days	Weekday, weekend, holiday, etc	
Weather conditions	Temperature, rain, snow, etc.	

Overall Objective Function

The final loss function of MVC-STNet is as follows:

$$Loss = \frac{1}{L} \sum_{t=1}^{L} ||\hat{Y}^t - Y^t||^2 \tag{11}$$

where L is the training sample size, $\hat{Y}^t$ is the prediction and Y^t is the ground truth. The pseudo-code of the algorithm is shown in Algorithm 1.

Experiment

Dataset and Experiment Setup

Dataset We use two datasets that are widely used in graph traffic flow prediction for evaluation: PeMS04, and PeMS08. The details of the datasets are introduced as follows.

PeMS04 This traffic dataset is collected in San Francisco Bay Area. It contains the traffic data collected by 3848 detectors on 29 roads. The time span of this dataset is from January to February in 2018. We use the first 54 days data for training and validating, and the remaining data are for testing.

PeMS08 This dataset is collected in San Bernaridino from July to August in 2016. It contains the traffic data collected by 1979 detectors on 8 roads. We use the first 55 days for

Table 2: RMSE and MAE comparison among different methods

Model	PeMS04		PeMS08	
	RMSE	MAE	RMSE	MAE
HA	47.80	32.73	43.32	30.09
ARIMA	55.18	36.84	48.88	34.27
FC-LSTM	43.18	29.52	39.94	26.80
STGCN	42.37	30.69	37.83	27.66
DCRNN	39.86	26.78	33.62	25.44
T-GCN	39.45	27.69	34.43	24.92
MVC-STNet	**37.61**	**23.83**	**32.88**	**21.40**

training and validating,and the remaining 7 days data for testing.

We also use some external features including weather, holiday and weekends. Whether the day is weekday, weekend or holiday is also considered as the people mobility patterns on holidays and regular days are quite different. The descriptions on the two datasets and external features are shown in Table 1.

Baselines We compare the proposed MVC-STNet with the following baseline methods.

- **HA** Historical Average(HA) uses the average historical data as the prediction of the future.
- **ARIMA** Auto-Regressive Integrated Moving Average (ARIMA) (Williams and Hoel 2003) is a classic statistics-based method for time series prediction.
- **FC-LSTM** FC-LSTM (Sutskever, Vinyals, and Le 2014) uses RNN with fully connected LSTM hidden units to capture the non-linear temporal dependencies for traffic prediction.
- **STGCN** STGCN (Yu, Yin, and Zhu 2018) applies ChebNet-GCN and 1D convolution to extract spatial and temporal dependencies for traffic prediction.
- **DCRNN** DCRNN (Li et al. 2018) uses graph convolution networks to capture the spatial correlations and the encoder-decoder architecture with scheduled sampling to capture the temporal correlations for traffic prediction.
- **T-GCN** T-GCN (Zhao et al. 2019) combines the graph convolutional network (GCN) and the gated recurrent unit (GRU) for traffic prediction

To further evaluate the effectiveness of different components in our model, we also compare the full version MVC-STNet with the following variants:

- **GCN-STNet** This model removes the Channel-wise Convolutional network and the multi-view fusion module. It only considers the local spatial correlations. By comparing with it, we test whether the proposed $CGCN(\cdot)$ and multi-view fusion is useful for improving the prediction performance.
- **CGCN-STNet** This model does not consider the features of the globalized spatial-temporal graph. Through comparing with this model, we test whether integrating the global spatial graph is helpful to capture the complex spatial features.

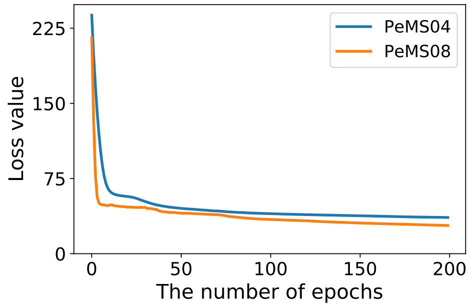


Figure 4: Loss curves of MVC-STNet on the two datasets

- **MV-STNet** This model drops the CGCN layer while only use the tradiction GCN layer (Kipf and Welling 2017) to capture the spatial correlations. Through comparing with it, we test whether the proposed CGCN method can model the divergence between channels.

Evaluation metric We adopt mean absolute error (MAE) and root mean square error (RMSE) defined as follows as the evaluation metrics.

$$MAE = \frac{1}{n}\sum_{t=1}^{n}|\hat{Y}_t - Y_t|, RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}||\hat{Y}_t - Y_t||^2} \quad (12)$$

where $\hat{Y}^t$ is the prediction, and Y^t is the ground truth.

Implement details We implement our model with Pytorch framework on NVIDIA RTX3090 GPU. The model parameters are set as follows. The input data size for PeMS04 dataset is $3\times307\times3$ where the first 3 is the previous time slot length used for prediction, 307 is the number of the nodes, the last 3 is the number of channels that represent flow, speed and occupy, for PeMS08 dataset is $3 \times 170 \times 3$, where 170 represents the number of the nodes. The CGCN model contains 3 layers whose hidden feature dimensions are $16, 32$, and 64. The final LSTM contains 2 layers, whose hidden dimension are all 256. The output of PeMS04 is $1 \times 307 \times 1$, and the output of PeMS08 is $1 \times 170 \times 1$. The baseline methods are implemented based on the original papers or we use the publicly available code. The parameters of baseline methods are set based on the original papers. Note that we normalize the traffic data into $[0, 1]$ to facility the feature learning.

Dataset	Methods	RMSE	MAE
PeMS04	GCN-STNet	40.77	26.83
	CGCN-STNet	38.47	24.83
	MV-STNet	38.91	25.12
	MVC-STNet	37.61	23.83
PeMS08	GCN-STNet	37.33	25.63
	CGCN-STNet	34.48	22.93
	MV-STNet	33.90	22.44
	MVC-STNet	32.88	21.40

Table 3: RMSE and MAE comparison with variant methods

Convergence of the algorithm Figure 4 shows the training loss curves of the algorithm on the two datasets. One can see that MVC-STNet converges after about 50 epochs on the two datasets, and then becomes stable. It shows the proposed model can quickly converge. In the following experiments, we run MVC-STNet with 50 epochs on both datasets.

Comparison with Baselines

Table 2 shows the performance comparison among different methods over the two datasets. The best results are highlighted with bold font. It shows that the proposed MVC-STNet achieves the best performance over both datasets. Traditional statistics-based methods ARIMA and HA achieve the worse performance among all the methods. It is not surprising because ARIMA and HA only use the time series data of each node, but ignore the spatial dependency. On PeMS04 dataset, compared with the best results achieved by baselines, MVC-STNet reduces RMSE of the traffic flow prediction from 39.45 achieved by T-GCN to 37.61, and MAE from 26.78 achieved by DCRNN to 23.83. MVC-STNet reduces the RMSE from 33.62 to 32.88, and MAE from 24.92 to 21.40. The decrease trends of RMSE and MAE over PeMS08 dataset are smaller than that over PeMS04. This is because the number of nodes in the traffic network of PeMS08 is smaller than that of PeMS04, which means that the spatial correlations is more complex in PeMS04. The results in Table 2 show that the proposed MVC-STNet is superior to existing state-of-the-art spatio-temporal learning approaches.

Comparison with Variant Models

To study whether the components in MVC-STNet are all helpful to the prediction task, we compare with its variants GCN-STNet, CGCN-STNet, and MV-STNet. The result is shown in Table 3. One can that CGCN and multi-view fusion module are all useful to the model as removing any one of them will increase the prediction error. On PeMS04 dataset, considering the divergence between channels seems more important, while the multi-view fusion operation is more important on PeMS08 dataset. Combining these components together achieves the lowest RMSE and MAE, demonstrating that all of them are useful to the studied problem.

Case Study

To further intuitively illustrate how accurately our model can predict the traffic flows, we visualize the predicted traffic flows and the ground truth in Figure 5. Due to space limitation, we only show a case study on the selected four nodes of the traffic network on the two datasets. From top to down, the upper two figures show the predicted traffic flows and the ground truth on nodes 7 and 15 of the PeMS04 dataset, and the lower two figures show the prediction and the ground truth on the nodes 20 and 29 of the PeMS08 dataset. One can see that the red curves of prediction can accurately trace the blue curves of the ground truth. The figure also shows that the two datasets present obvious periodical change characteristics, which is consistent with the traffic mobility patterns in cities. Our model can perfectly capture the periodicity of the data due to the usage of LSTM. Furthermore, the results show that our proposed MVC-STNet model can capture the spatial and temporal dependencies effectively.

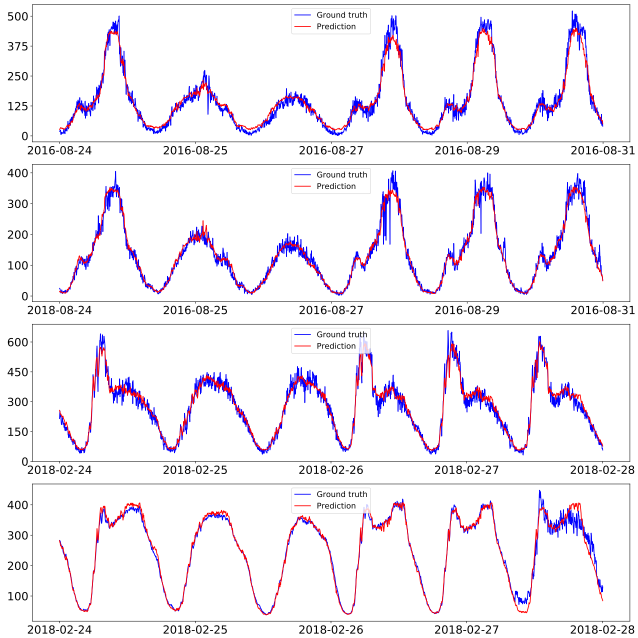


Figure 5: Prediction vs ground truth on two datasets(top to down: node 7 and node 15 in PeMS04, node 20 and node 29 in PeMS08)

Conclusion

In this paper, we propose a novel deep multi-view channel-wise spatio-temporal network named MVC-STNet. The novelty of the model lies in the proposed channel-wise graph convolutional network to learn the divergence between channels. The local and global spatio-temporal graph are constructed to capture both the geographical and semantic spatial correlations. Extensive evaluations on two real datasets show that the proposed model can mutually enhance the performance of both tasks, and also outperforms state-of-the-art graph based deep learning methods for traffic flow prediction. In the future, we will further study how to learn more temporal dependencies to capture sudden changes of traffic flows. In addition, it would be also interesting to further study how the proposed MVC-STNet framework can be applied to other prediction tasks such as traffic congestion prediction and traffic accident prediction.

Acknowledgement

This work is supported by CCF-Tencent Open Research Fund, and the Fundamental Research Funds for the Central Universities (No.: NZ2020014).

References

[Bai et al. 2020] Bai, L.; Yao, L.; Li, C.; Wang, X.; and Wang, C. 2020. Adaptive graph convolutional recurrent network for traffic forecasting. In *34th Conference on Neural Information Processing Systems*.

[Castro-Neto et al. 2009] Castro-Neto, M.; Jeong, Y.-S.; Jeong, M.-K.; and Han, L. D. 2009. Online-svr for short-term traffic flow prediction under typical and atypical traffic conditions. *Expert systems with applications* 36(3):6164–6173.

[Chandra and Al-Deek 2009] Chandra, S. R., and Al-Deek, H. 2009. Predictions of freeway traffic speeds and volumes using vector autoregressive models. *Journal of Intelligent Transportation Systems* 13(2):53–72.

[Defferrard, Bresson, and Vandergheynst 2016] Defferrard, M.; Bresson, X.; and Vandergheynst, P. 2016. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in neural information processing systems*, 3844–3852.

[Estrach et al. 2014] Estrach, J. B.; Zaremba, W.; Szlam, A.; and LeCun, Y. 2014. Spectral networks and deep locally connected networks on graphs. In *2nd International Conference on Learning Representations, ICLR 2014*.

[Hamilton, Ying, and Leskovec 2017] Hamilton, W.; Ying, Z.; and Leskovec, J. 2017. Inductive representation learning on large graphs. In *Advances in neural information processing systems*, 1024–1034.

[Huang, Xu, and Weng 2020] Huang, F.; Xu, J.; and Weng, J. 2020. Multi-task travel route planning with a flexible deep learning framework. *IEEE Transactions on Intelligent Transportation Systems*.

[Kipf and Welling 2017] Kipf, T. N., and Welling, M. 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*.

[Li et al. 2018] Li, Y.; Yu, R.; Shahabi, C.; and Liu, Y. 2018. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In *International Conference on Learning Representations*.

[Qu, Bengio, and Tang 2019] Qu, M.; Bengio, Y.; and Tang, J. 2019. Gmnn: Graph markov neural networks. In *International Conference on Machine Learning*, 5241–5250.

[Sutskever, Vinyals, and Le 2014] Sutskever, I.; Vinyals, O.; and Le, Q. V. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, 3104–3112.

[Tobler 1970] Tobler, W. R. 1970. A computer movie simulating urban growth in the detroit region. *Economic geography* 46(sup1):234–240.

[Veličković et al. 2018] Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2018. Graph attention networks. In *International Conference on Learning Representations*.

[Wang et al. 2014] Wang, S.; Hu, X.; Yu, P. S.; and Li, Z. 2014. Mmrate: inferring multi-aspect diffusion networks with multi-pattern cascades. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 1246–1255.

[Wang et al. 2018] Wang, D.; Zhang, J.; Cao, W.; Li, J.; and Zheng, Y. 2018. When will you arrive? estimating travel time based on deep neural networks. In *AAAI*, volume 18, 1–8.

[Wang et al. 2020] Wang, S.; Cao, J.; Chen, H.; Peng, H.; and Huang, Z. 2020. Seqst-gan: Seq2seq generative adversarial nets for multi-step urban crowd flow prediction. *ACM Transactions on Spatial Algorithms and Systems (TSAS)* 6(4):1–24.

[Wang, Cao, and Yu 2020] Wang, S.; Cao, J.; and Yu, P. 2020. Deep learning for spatio-temporal data mining: A survey. *IEEE Transactions on Knowledge and Data Engineering*.

[Williams and Hoel 2003] Williams, B. M., and Hoel, L. A. 2003. Modeling and forecasting vehicular traffic flow as a seasonal arima process: Theoretical basis and empirical results. *Journal of transportation engineering* 129(6):664–672.

[Wu et al. 2019] Wu, Z.; Pan, S.; Long, G.; Jiang, J.; and Zhang, C. 2019. Graph wavenet for deep spatial-temporal graph modeling. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, 1907–1913. AAAI Press.

[Xingjian et al. 2015] Xingjian, S.; Chen, Z.; Wang, H.; Yeung, D.-Y.; Wong, W.-K.; and Woo, W.-c. 2015. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Advances in neural information processing systems*, 802–810.

[Yao et al. 2018] Yao, H.; Wu, F.; Ke, J.; Tang, X.; Jia, Y.; Lu, S.; Gong, P.; Ye, J.; and Li, Z. 2018. Deep multi-view spatial-temporal network for taxi demand prediction. In *2018 AAAI Conference on Artificial Intelligence (AAAI'18)*.

[Yao et al. 2019] Yao, H.; Tang, X.; Wei, H.; Zheng, G.; and Li, Z. 2019. Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 5668–5675.

[Yu, Yin, and Zhu 2018] Yu, B.; Yin, H.; and Zhu, Z. 2018. Spatio-temporal graph convolutional networks: a deep learning framework for traffic forecasting. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 3634–3640.

[Zhang, Zheng, and Qi 2017] Zhang, J.; Zheng, Y.; and Qi, D. 2017. Deep spatio-temporal residual networks for citywide crowd flows prediction. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, 1655–1661.

[Zhao et al. 2019] Zhao, L.; Song, Y.; Zhang, C.; Liu, Y.; Wang, P.; Lin, T.; Deng, M.; and Li, H. 2019. T-gcn: A temporal graph convolutional network for traffic prediction. *IEEE Transactions on Intelligent Transportation Systems*.

[Zheng et al. 2020] Zheng, C.; Fan, X.; Wang, C.; and Qi, J. 2020. Gman: A graph multi-attention network for traffic prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 1234–1241.