Origin-Destination Convolution Recurrent Network A Novel OD Matrix Prediction Framework

Presentation · October 2023

4 authors, including:

Tian Liang
The University of Manchester

Wanzhi Xiao
Central South University

Origin-Destination Convolution Recurrent Network: A Novel OD Matrix Prediction Framework

Jiayu Chang, Tian Liang, Wanzhi Xiao, Li Kuang

School of Computer Science and Engineering, Central South University,
Changsha, China

Content

01 Introduction

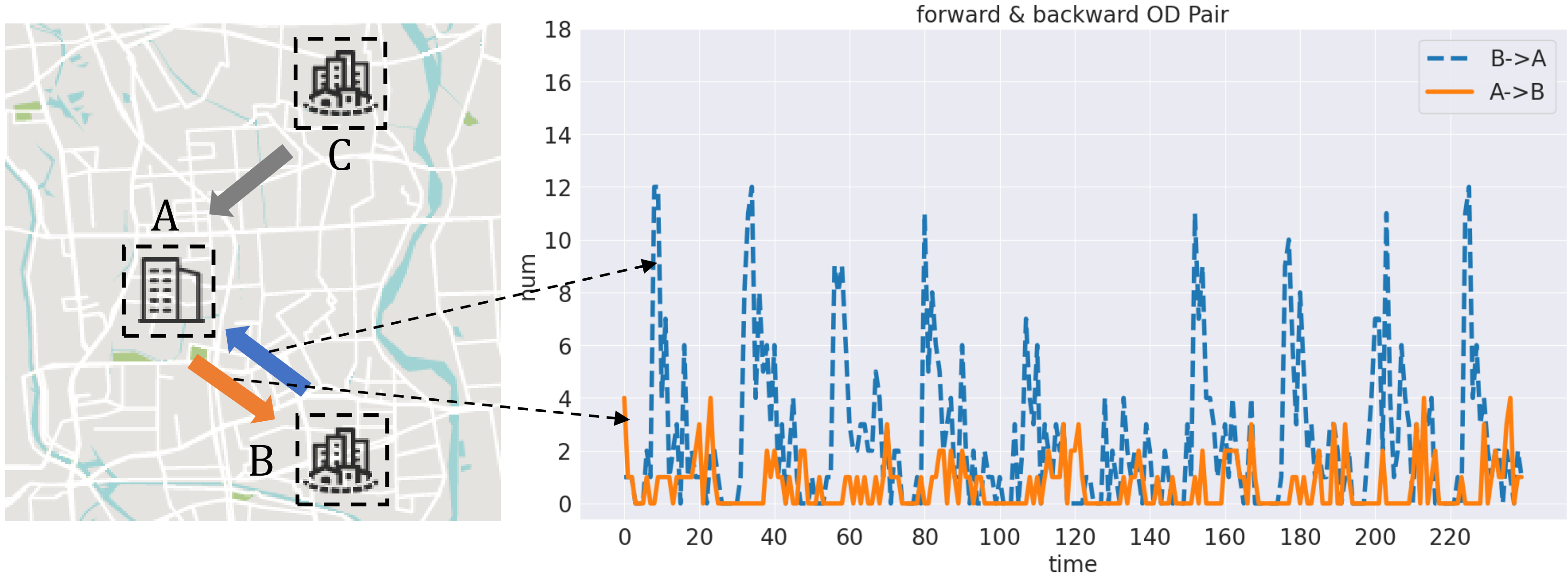
02 Problem Definition

03 Methods

04 Experiments and Analysis

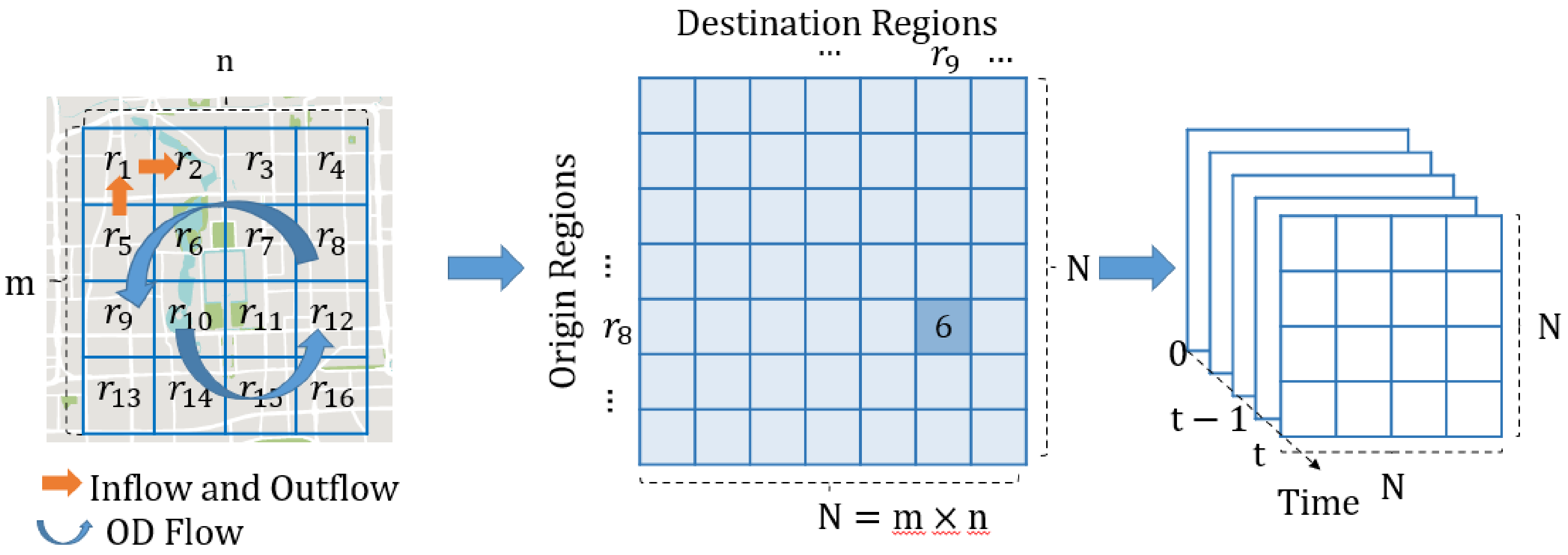
05 Conclusion

Introduction



- Ride-hailing applications are becoming prevalent choices for daily commutes, such as Didi, UCAR, and Uber, which aim to provide passengers with convenient ride services and improve the efficiency of public transportation.
- Directed semantics of travel demand
- Various Traffic Context
- OD matrix prediction has received increasing attention

Problem Definition



Spatial Neighborhood

OD Matrix

OD Matrices Sequence

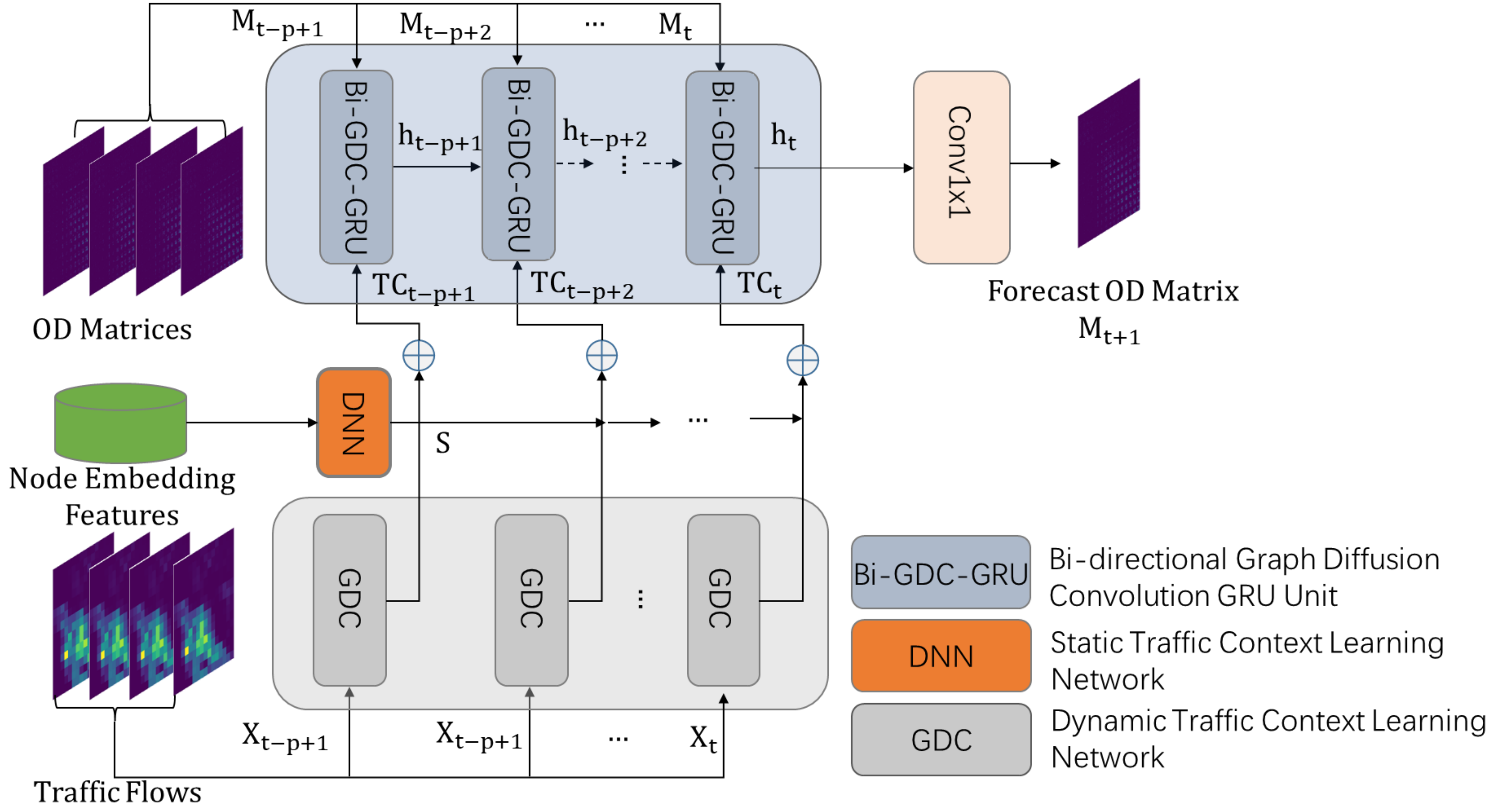
Definition 2.1.3 Static Traffic Context We define the traffic graph of city as $G = \{V, A\}$ where $V = [v_1, \cdots, v_{N_l}]$ and $|V| = N_l$. $v_i \in V$ is a node on the traffic graph which represents a region in the city. 2-dimensional tensor $S \in R^{N_l \times F}$ represents the static traffic context on all regions, such as POIs and embedding features.

$M = [M_1, \cdots, M_{Nt}] \in R^{N_t \times N_l \times \hat{N_l}}$, where N_t is
historical traffic data.

Defi
from
parti

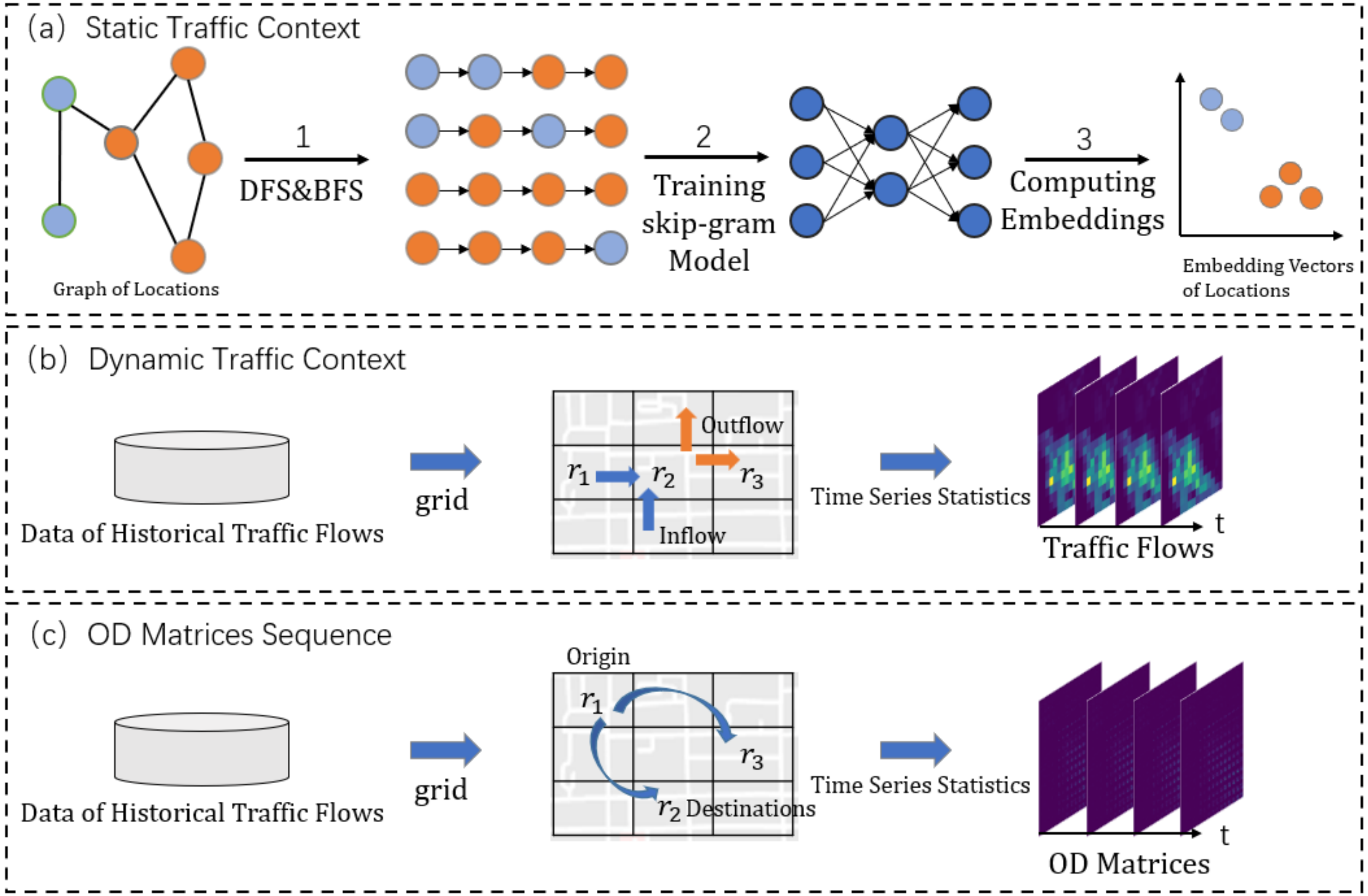
Definition 2.1.4 Dynamic Traffic Context We define the dynamic traffic context in regions as a 3-dimensional tensor $X = [X_1, \cdots, X_{Nt}] \in R^{Nt \times Nl \times Dt}$, where the D_t represents the features' dimension of dynamic traffic context. For example, when we use inflow and outflow as features of context, features' dimension $D_t = 2$.

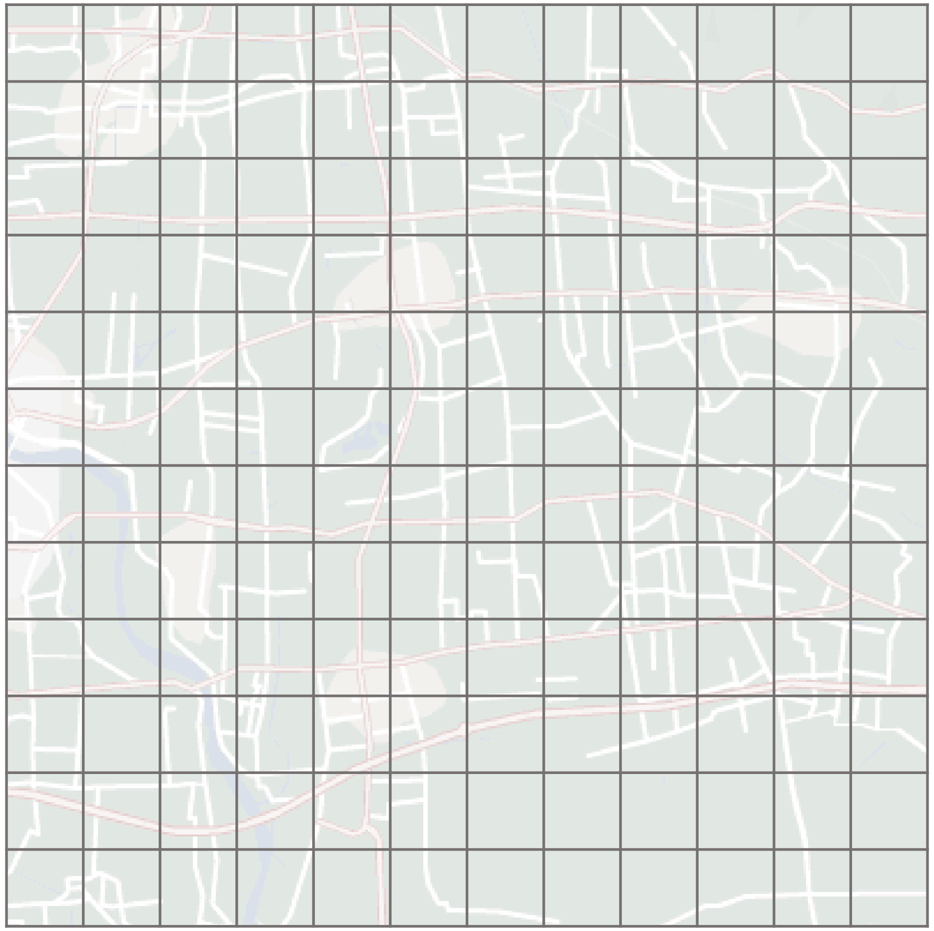
Methodology



Overview of the framework ODGCN

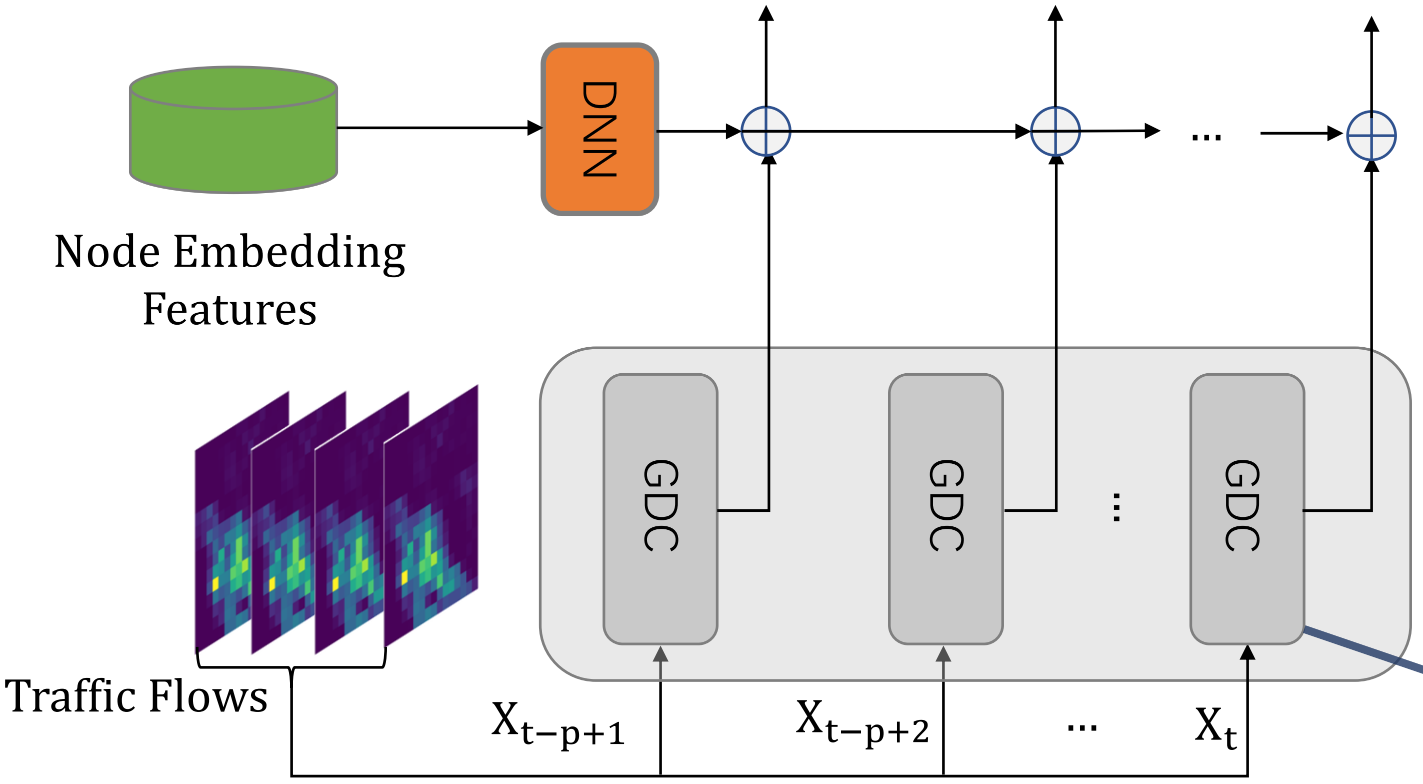
Data Preprocess





The entire city has been divided into m×n grids

Static & Dynamic Traffic Context Learning Network



- Node embedding features represent the structural properties of the nodes on the graph and the similarity between nodes etc. After constructing the static traffic context, we use a deep neural network DNN as a Static Context Learner (SCL) to map the static traffic context into node hidden states.

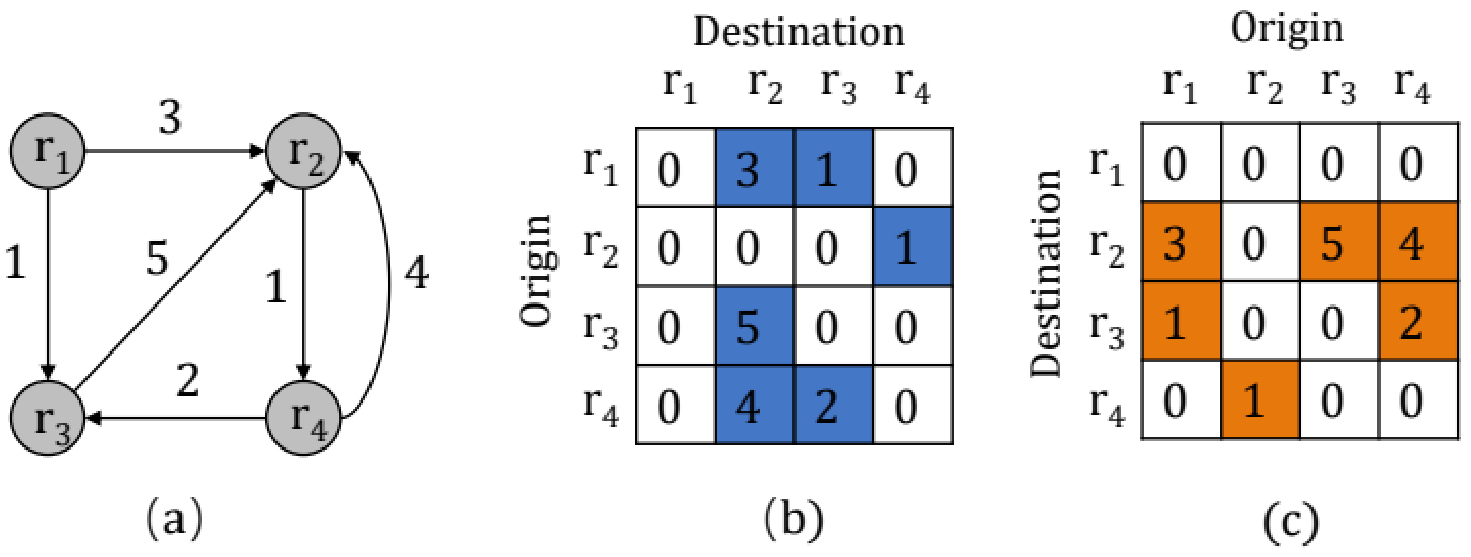
$$h_1 = SCL(S) = DNN(S)$$

- we use a graph convolution network as a Dynamic Context Learner (DCL) network to capture spatial correlations. DCL maps the dynamic traffic context of all nodes in each time slot into hidden state representations:

$$h_2 = DCL(X_t) = GDC(A, X_t, W) = \sum_{k=0}^{K} A^k X_t W_k$$

- Each GDC processes a dynamic traffic context for a time segment, which is then spliced or fused with a static context and then fed into **Bi-GDC-GRU Unit**

Bi-GDC-GRU Unit



- Since there is some correlations between the region set as the origin and set as the destination, using the forward and backward adjacency matrix to simultaneously aggregate the semantic neighbor node features on the OD pair starting from node vi and arriving at node vi.

$$A_f^{(t)} = Norm\left(M_t\right)$$

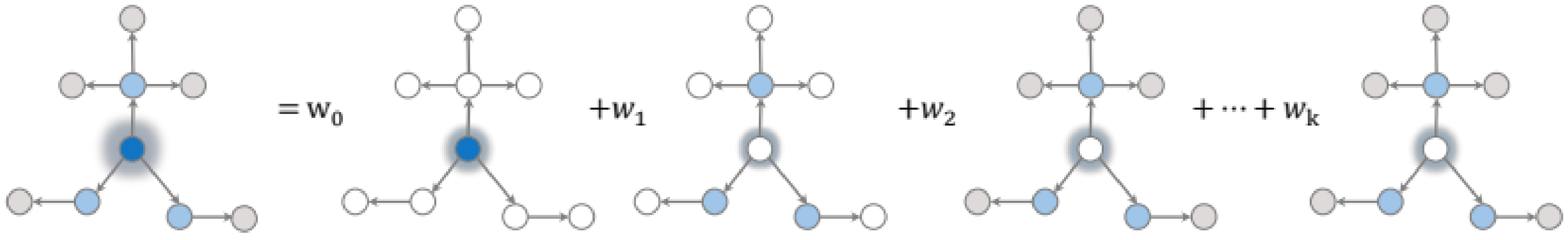
$$A_b^{(t)} = A_f^{(t)}.T$$

- we contact these two matrix and use bidirectional diffusion graph convolution to capture the spatial correlation between OD pairs.

$$h^{(t)} = \sum_{k=0}^{K} \left(A_f^{(t)}\right)^k \mathrm{TC}_t W_{k1} + \left(A_b^{(t)}\right)^k \mathrm{TC}_t W_{k2}$$

Advantages

- capture both temporal and spatial correlations
- Alleviate the sparsity of OD matrix to some extent.



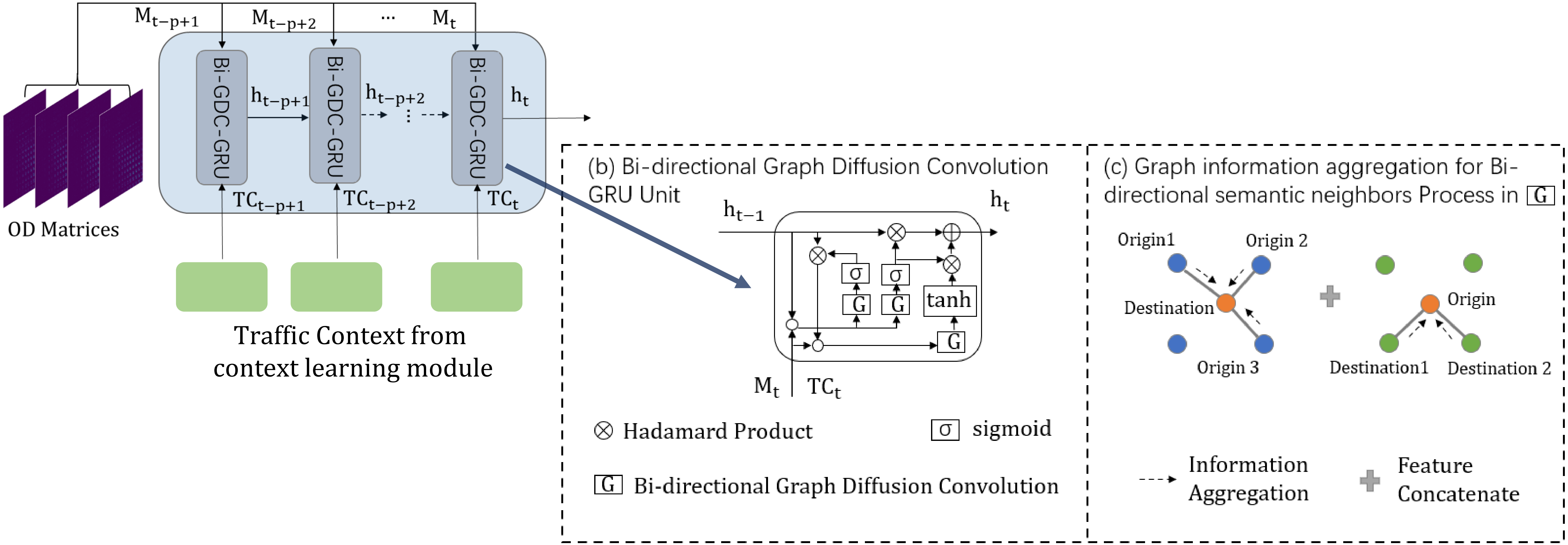
The graph convolution diffusion process of the node in the center

Step 0

Step 1

Step 2

Step k



- The time correlations of the OD matrix at different moments are captured by the GRU units based on the recurrent neural network. Specifically, we use GRU to capture the temporal correlation between OD matrix, and replace the matrix multiplication operation in GRU with the bi-directional diffusion graph convolution operation in the following Equation:

$$r^{(t)} = \sigma\left(\Theta_r *_G \left[TC_t, H^{(t-1)}\right] + b_r\right)$$

$$u^{(t)} = \sigma\left(\Theta_u *_G \left[TC_t, H^{(t-1)}\right] + b_u\right)$$

$$c^{(t)} = \tanh\left(\Theta_c *_G \left[TC_t, \left(r^{(t)} \odot H^{(t-1)}\right)\right] + b_c\right)$$

$$H^{(t)} = u^{(t)} \odot H^{(t-1)} + \left(1 - u^{(t)}\right) \odot c^{(t)}$$

Experiments & Analysis

Dataset	TaxiNYC	TaxiCD
Data type	Taxi Orders	Taxi Orders
City	New York	Chengdu
Longitude Range	-74.02 ~ 73.95	104.02 ~ 104.12
Latitude Range	40.67 ~ 40.77	30.62 ~ 30.70
Time Range	2015/1/1-2015/4/30	2016/11/1-2016/11/30
Total Number of time slots	2880	720
Length of unit time period	1 Hour	1 Hour
Number of grids	16×16	16×16
Static Traffic Context Information	Node2vec Node Embedding	Node2vec Node Embedding
Dynamic Traffic Context Information	Regional inflow/outflow	Regional inflow/outflow

TaxiNYC: https://www.nyc.gov/site/tlc/about/tlc-trip-record-data.page
TaxiCD: https://outreach.didichuxing.com/research/opendata/

Static Context: Two weeks of order data are preprocessed into a graph and then trained using Node2Vec as the static context.

Dynamic Context: Count the number of times each area is picked up and dropped off per hour as the inflow and outflow of the area.

OD Matrix: The historical OD matrix predicts the OD matrix for the next hour using the past 8 hours of historical data.

Experiments & Analysis

Models	RMSE(NYC)	MAE(NYC)	RMSE(CD)	MAE(CD)
HA	1.085	0.163	0.440	0.114
ARIMA	4.027	2.713	0.813	0.328
ST-ResNet	0.647	0.195	0.325	0.146
MDL	1.081	0.993	0.413	0.319
GEML	0.865	0.156	0.332	0.090
ODCRN(ours)	**0.622**	**0.153**	**0.322**	**0.116**

- **HA and ARIMA** have poor prediction results, due to the randomness of OD travel demand between some regions.
- **ST-ResNet** which considers the trend, period, proximity, and other temporally more relevant historical data, and deposits a residual network to enhance the network depth, improves the prediction effect.
- **MDL** designs node network and edge network branches to extract edge flow (OD) and node flow (inflow and outflow) features respectively, and uses a multi-task learning approach to predict both flows. But its performance is inferior to that of the single-task ST-ResNet.
- **GEML** is a GNN-based OD matrix prediction model that uses a grid embedding approach and graph convolution to aggregate information about the semantic and geographic neighbors of OD pairs to pre-weight the importance of neighbor nodes in the OD matrices, and adapts LSTM to capture temporal correlations. But the model neither considers the matrix prediction in terms of dynamic and static contextual information of the starting and ending points, nor does it explicitly distinguish the direction of the edges of the adjacency matrix.

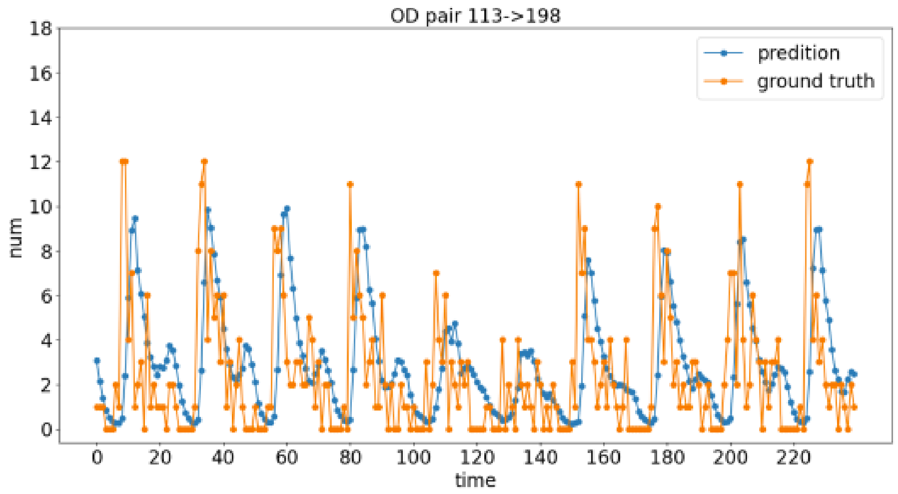
Experiments & Analysis

Table 3. Comparison of the number of Parameters on the Model and Speed

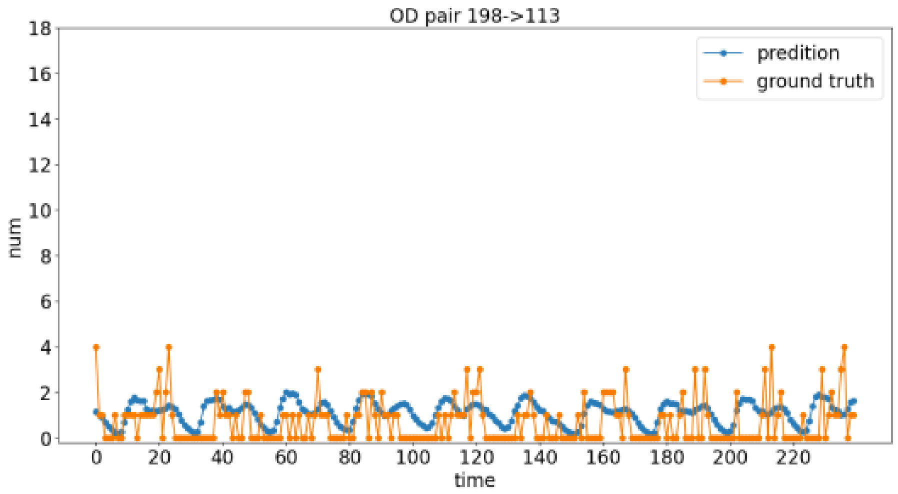
Model	The Number of Trainable Parameters	Training Speed	Reasoning Speed
MDL	7,694,106	81	238
ST-ResNet	7,202,579	93	323
GEML	395,264	53	138
ODCRN(ours)	108,640	108	324

Our ODCRN model, compared with the GEML model, is an OD matrix prediction model with an LSTM network which has **a much smaller number of parameters**. For inputs at different time steps, **the parameters in the LSTM units are shared**, greatly reducing the number of parameters. In contrast, since the ODCRN model integrates both GRU and GCN, the number of parameters in the whole model is mainly the convolutional kernel parameters in the bi-direction diffusion graph convolution in the GRU units, which further reduces the number of parameters compared with the serial structure of GCN and LSTM in GEML. The design of the recurrent unit in ODCRN can capture both temporal and spatial correlations, which improves the training and inference speed while reducing the parameters.

Experiments & Analysis



(a) from region 113 to region 198



(b) from region 198 to region 113

- The true values of the OD curves change irregularly with time, which shows an overall cyclical nature and a more obvious morning and evening peak. The trend of the curve and the range of values are different, which indicates that the travel demand between regions is asymmetric, and the forward OD matrix and backward OD matrix we constructed aggregates the characteristics of neighbor nodes with OD semantic relationship through a directed and weighted adjacency matrix.
- The ODCRN fits the real data well and is generally consistent with the real values in terms of trend and numerical magnitude. We can also find that the OD curves show dramatic local fluctuations due to the dynamic changes in taxi demand between regions. And this randomly changing curve increases the difficulty of prediction, so the model fails to give a more accurate prediction in detail.

Ablation Study

Model Variant	RMSE(NYC)	MAE(NYC)	RMSE(CD)	MAE(CD)
w/o static context learner	0.628	0.157	0.322	0.118
w/o dynamic context learner	0.687	0.204	0.327	0.123
w/o forward GDC	0.663	0.161	0.328	0.121
w/o backward GDC	0.631	0.159	0.322	0.119
ODCRN	0.622	0.153	0.322	0.116

- W/O Static Context Learner: without inputting the static traffic context information, i.e., Node2vec node embedding features.
- W/O Dynamic Context Learner: without inputting the dynamic traffic context information, i.e., The inflow and outflow for each region at each moment.
- W/O Forward GDC: removing the graph aggregation from the central node, the forward adjacency matrix is an $N \times N (N = m \times n)$ matrix from Origin to Destination.
- W/O Backward GDC: removing the graph aggregation to the central node, the backward adjacency matrix is an $N \times N (N = m \times n)$ matrix from Destination to Origin.

Conclusion

- In this paper we propose ODCRN, a novel model that integrates traffic context and spatial-temporal information of OD matrices. We take advantage of the bi-directional semantic information of each travel demand’s semantic neighbors to capture both the inflow and outflow of one region. Then we construct traffic context information by static and dynamic traffic contexts to coordinate with OD matrices in the prediction task. We conduct extensive experiments in two datasets and the results demonstrate that our method outperforms baseline methods in terms of prediction accuracy and model complexity.

- In future work, we will explore predicting traffic spikes or dips caused by rush hours in the day, holidays, extreme weathers and social events more accurately. Also, we will consider integrating more traffic features into the model elegantly, which requires more experiments on model design and data mining.

Thanks for

Your Time