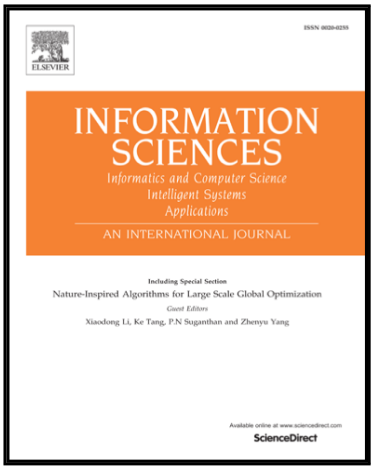

Spatial Temporal Incidence Dynamic Graph Neural Networks for Traffic Flow Forecasting

Hao Peng, Hongfei Wang, Bowen Du, Md Zakirul Alam Bhuiyan, Hongyuan Ma, Jianwei Liu, Lihong Wang, Zeyu Yang, Linfeng Du, Senzhang Wang, Philip S. Yu

PII: S0020-0255(20)30045-1
DOI: https://doi.org/10.1016/j.ins.2020.01.043
Reference: INS 15174

To appear in: *Information Sciences*

Received date: 31 March 2019
Revised date: 17 December 2019
Accepted date: 19 January 2020

Please cite this article as: Hao Peng, Hongfei Wang, Bowen Du, Md Zakirul Alam Bhuiyan, Hongyuan Ma, Jianwei Liu, Lihong Wang, Zeyu Yang, Linfeng Du, Senzhang Wang, Philip S. Yu, Spatial Temporal Incidence Dynamic Graph Neural Networks for Traffic Flow Forecasting, *Information Sciences* (2020), doi: https://doi.org/10.1016/j.ins.2020.01.043

Highlights

- The paper proposes a novel dynamic graph recurrent convolutional neural network model, named Dynamic-GRCNN, to deeply capture the spatio-temporal traffic flow features for more accurately predicting urban passenger traffic flows.
- The paper presents incidence dynamic graph structures based on historically passenger traffic flows to model traffic station relationships. Different from existing traffic transportation network topological structures based graph relationships between stations, the incidence dynamic graph structures firstly model the traffic relationships from historical passenger flows.
- For real urban passenger traffic flows, the paper demonstrates that dynamic spatial-temporal incidence graphs are more suitable to model external changes and influences.
- The paper compares Dynamic-GRCNN with state-of-the-art deep learning approaches on three benchmark datasets which contain different types of passenger traffic flows for evaluation. The results show that Dynamic-GRCNN significantly outperforms all the baselines in both effectiveness and efficiency in urban passenger traffic flows prediction.

Spatial Temporal Incidence Dynamic Graph Neural Networks for Traffic Flow Forecasting

Hao Peng[a], Hongfei Wang[b], Bowen Du[b,*], Md Zakirul Alam Bhuiyan[c,*], Hongyuan Ma[d,*], Jianwei Liu[a], Lihong Wang[d], Zeyu Yang[e], Linfeng Du[b], Senzhang Wang[f], Philip S. Yu[g]

[a]*School of Cyber Science and Technology, Beihang University, Beijing 100083, China*
[b]*School of Computer Science and Engineering, Beihang University, Beijing 100083, China*
[c]*Department of Computer and Information Sciences, Fordham University, NY 10458, USA*
[d]*National Computer Network Emergency Response Technical Team/Coordination Center of China, Beijing 100029, China*
[e]*Department of Electronic Engineering, Tsinghua University, Beijing 100084, China*
[f]*College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China*
[g]*Department of Computer Science, University of Illinois at Chicago, Chicago 60607, USA*

Abstract

Accurate and real-time traffic passenger flows forecasting at transportation hubs, such as subway/bus stations, is a practical application and of great significance for urban traffic planning, control, guidance, etc. Recently deep learning based methods are promised to learn the spatial-temporal features from high non-linearity and complexity of traffic flows. However, it is still very challenging to handle so much complex factors including the urban transportation network topological structures and the laws of traffic flows with spatial and temporal dependencies. Considering both the static hybrid urban transportation network structures and dynamic spatial-temporal relationships among stations from historical traffic passenger flows, a more effective and fine-grained spatial-temporal features learning framework is necessary. In this paper, we propose a novel spatial-temporal incidence dynamic graph neural networks framework for urban traffic passenger flows prediction. We first model dynamic traffic station relationships over time as spatial-temporal incidence dynamic graph structures based on historically traffic passenger flows. Then we design a novel dynamic graph recurrent convolutional neural network, namely Dynamic-GRCNN, to learn the spatial-temporal features representation for urban transportation network topological structures and transportation hubs. To fully utilize the historical passenger flows, we sample the short-term, medium-term and long-term historical traffic data in training, which can capture the periodicity and trend of the traffic passenger flows at different stations. We conduct extensive experiments on different types of traffic passenger flows datasets including subway, taxi and bus flows in Beijing. The results show that the proposed Dynamic-GRCNN effectively captures comprehensive spatial-temporal correlations significantly and outperforms both traditional and deep learning based urban traffic passenger flows prediction methods.

Keywords: Traffic Passenger Flows Prediction, Graph Convolutional Neural Network, LSTM, Importance Sampling, Urban Computing

1. Introduction

The goal of traffic passenger flow forecasting on transportation hubs is to predict the future traffic passenger inflow and outflow of traffic stations based on previous traffic flow measured by sensors [1]. Traffic passenger flows forecasting plays an important role in urban traffic route planning, traffic control, management and everybody's daily life, which is also one of main functions of the smart city system. Widely used intelligent transportation services, such as travel recommendation and navigation for traveler, also rely heavily on a high-quality traffic flows evaluation. Even with the development of urbanization and urban population expansion, intelligent transportation systems are increasingly affecting everyone's travel and even social security [2, 3]. An accurate and real-time traffic flows forecasting can not only provide a scientific basis for traffic managers to sense traffic congestions and limit vehicles in advance but also provide security for urban travelers to choose appropriate travel routes and improve travel efficiency.

However, accurate and real-time traffic flows forecasting has always been a challenge task due to its complex spatial and temporal dependencies, and inherent and dynamic difficulties with hybrid urban transportation network topological structures

*Corresponding author
Email addresses: penghao@act.buaa.edu.cn (Hao Peng), gabrielle_w@buaa.edu.cn (Hongfei Wang), dubowen@buaa.edu.cn (Bowen Du), mbhuiyan3@fordham.edu (Md Zakirul Alam Bhuiyan), mahongyuan@foxmail.com (Hongyuan Ma), liujianwei@buaa.edu.cn (Jianwei Liu), wlh@isc.org.cn (Lihong Wang), yangzeyu16@mails.tsinghua.edu.cn (Zeyu Yang), fred9918@buaa.edu.cn (Linfeng Du), szwang@nuaa.edu.cn (Senzhang Wang), psyu@uic.edu (Philip S. Yu)

in the long term forecasting. On the one hand, the changes in traffic passenger flows consist of all transportation hubs, such as subway/bus stations, in the topological structure of the urban transportation network. Moreover, the traffic passenger flows change dynamically over time and are reflected in closeness, periodicity and trend [4, 5]. On the other hand, it's hard to model the traffic passenger flows generated by tens of thousands of passengers in a urban traffic network at anytime. Since different passengers of inflows travel to different traffic stations to form different outflows, the relationship among different traffic passenger stations will dynamically change with external influences such as time, social events, traffic accidents, etc. Furthermore, different urban transportation network topological structures also affect the modeling of traffic passenger flows, such as square transportation network topological structures and irregular transportation network topological structures.

There are many existing traffic forecasting methods, some of which consider temporal dependence, including the Autoregressive Integrated Moving Average (ARIMA) model [6], historical average (HA), vector Autoregressive (VAR), gaussian process based [7], Kalman filtering model [8], etc, and others consider multi-source data fusion, including mixed gaussian probability model [9], coupled matrix and tensor factorization model [10, 11], etc. Despite the above methods consider the dynamic change of traffic conditions and the compelling results achieve by these studies, their prediction accuracies remain unsatisfactory for building reliable and more complicated traffic passenger flows forecasting systems in practice. Recent years, deep learning methods have shown promising results in dynamic prediction over spatial-temporal traffic flows data [1, 12, 5]. For example, there are three mainstream deep learning architectures which have attracted more researchers' attention in spatial-temporal data mining, i.e., recurrent neural networks (RNNs) based models [13, 14, 15], convolutional neural networks (CNNs) based models [2, 4, 16, 17] and graph convolution networks (GCNs) based models [18, 19, 20] due to their powerful ability in learning different-levels of features of spatial-temporal traffic data. The above methods consider the temporal dependency and spatial dependency, respectively, but ignore modeling the dynamic traffic station relationships from historical traffic passenger flows. So that considering both traffic transportation network topological structures and potential relationships among stations from historical traffic passenger flows will benefit to accurately traffic passenger flows forecasting.

In this paper, we propose a novel spatial-temporal incidence dynamic graph neural networks framework for urban traffic passenger flows prediction. We first model dynamic traffic station relationships over time as spatial-temporal incidence dynamic graph structures based on historically streaming traffic passenger flows of the same type of vehicle. The graph structures consider incidence dynamic relationships of both inflows and outflows. Then we design a novel dynamic graph recurrent convolutional neural network model, namely Dynamic-GRCNN, to learn the spatial-temporal features representation for urban transportation network topological structures and transportation hubs. Different from previous deep learning based models [4, 21, 18, 14], they either take external influences such as holidays, weekends, events, etc. as a hyper-parametric variable of the model, or ignore external influences, and we model the traffic flows affected by external influences as dynamic graph structures of traffic stations with specific periodic features. To fully utilize the historical passenger flows, we also use periodic sampling, including the short-term, medium-term and long-term historical traffic flows, and importance sampling of historical traffic flows in training, which can capture the periodicity and trend of the traffic passenger flows for different transportation hubs. Specifically, the whole architecture of the proposed model is as follows.

Graph Structures Input. Instead of modeling the urban traffic passenger flows as traffic transportation network based matrix [4, 5, 2, 22] or topological graph [21, 20, 23, 24, 18], we propose to utilize historical traffic passenger flows to construct dynamic incidence relationships among traffic transportation hubs as graph structures, where the node refers to the traffic stations, and the edge refers to the interaction between the two traffic stations. Even, we use the dynamic graph structures to model the different influences of different time, festivals and events on the relationships among traffic stations. For the traffic passenger inflows and outflows, we take them as two attributes of the node in the above graph structures. Then, we use the spatial structure information of the graph to convert the graph into a matrix of traffic stations. So, we can utilize standard deep convolutional neural network to learn different-levels of features.

Graph Recurrent Deep Convolutional Neural Network Layers. In order to effectively learn the spatial and temporal features of traffic passenger flows, we propose to utilize recurrent deep convolutional neural networks to capture the feature representation. For each graph, we use the deep convolutional neural networks to extract the spatial features. Then, we sample the short-term, medium-term and long-term historical traffic flows, such as week periodicity, daily trendiness and recentness, to learn temporal features of traffic passenger flows by multiple LSTM units. Even we sample the recent samples with a larger probability than periodicity and trend's since the most recent traffic passenger flows of a region are highly correlated to the traffic passenger flows of the region in the next time slot. It's an importance sampling instance when extracting training traffic flows with different probabilities.

Output. After extracting spatial and temporal features of traffic passenger flows, we employ both two-layer fully connected networks and one Sigmoid layer as the final output layer for the prediction of each traffic's inflow and outflow. Unlike the matrix-based output that contains a large number of non-traffic regions, the output of the proposed model is directly the traffic inflow and outflow for each station. We use RMSE and MAP to measure the loss between the predicted flows and ground truth flows, respectively. We calculate the accumulated loss for each station in mini-batch.

The main contributions of this paper are:

- We propose a novel dynamic graph recurrent convolutional neural network model, named Dynamic-GRCNN, to deeply capture the spatio-temporal traffic flow features for more accurately predicting urban traffic passenger flows. Different from previous deep learning models, Dynamic-GRCNN models graph convolutional neural network, LSTM units, periodic sampling and importance sampling to better fit the historical traffic passenger flows and predict the future traffic flows.
- We propose incidence dynamic graph structures based on historically traffic passenger flows to model traffic station relationships. Different from existing traffic transportation network topological structures based graph relationships between stations, the incidence dynamic graph structures firstly model the traffic relationships from historical passenger flows among stations.
- For real urban traffic passenger flows, we demonstrate that dynamic spatial-temporal incidence graphs are more suitable to model external changes and influences. This can be a general framework for deep learning model to be applied in modeling real traffic passenger flows data.
- We compare Dynamic-GRCNN with traditional methods and state-of-the-art deep learning methods on three benchmark datasets which contain different types of traffic passenger flows for evaluation. The results show that Dynamic-GRCNN significantly outperforms all the baselines in both effectiveness and efficiency in urban traffic passenger flows prediction.

The rest of the paper is organized as follows. We first review related work in Section 2. Second, we present how to build incidence dynamic graph in Section 3. Third, the detail of the Dynamic-GRCNN framework is introduced in Sections 4. Then, we present the introduce of datasets, baseline methods and experimental settings in Sections 5. Next, we evaluate the effectiveness comparison and efficiency analysis of the proposed model in Section 6. Last, we give the conclusion of this work in Section 7. All source codes of this work will publicly available at `https://github.com/RingBDStack/GCNN-In-Traffic`.

2. Related Work

In this section, we briefly review the related work. traffic passenger flows prediction models can be roughly categorized into traditional machine learning based prediction models and recent deep learning based prediction models. Next we review the related work in the following two categories.

2.1. Traditional traffic passenger Flows Prediction Models

Traditional traffic passenger flows prediction models rely on feature engineering and selection from history data and existing traffic transportation networks to extract features for prediction task. Generally, traditional traffic passenger flows prediction methods can be categorized into classic statistical models, including Auto-Regressive Integrated Moving Average (ARIMA) based methods [25, 26] and parametric learning methods, including K-nearest neighbor (KNN) nonparametric regression methods, historical average(HA), vector Autoregressive (VAR) [27], gaussian process based [7, 28], support vector machine (SVM), neural networks (NN), etc [29, 30]. On the one hand, traditional works naturally focus on predicting the traffic passenger flows of one particular region, such as a street or a local region, and they do not generate the city-level traffic passenger flows prediction. For instance, ARIMA-based models are not suitable for analyzing time series with missing data, since they rely on uninterrupted time series data. HA model cannot effectively capture dynamic changes of the traffic data, such as incidents or social events. VAR model can capture the linear inter-dependencies among inter-related time series, but the correlation between the predicted values is neglected. On the other hand, these shallow models depend on hand-crafted patterns and can not fully explore complex spatial-temporal features among the big traffic data, which greatly limits their performances. In addition to parametric learning based predictive models, there are some researchers having attempted to integrate multi-source traffic data including external factors such as traffic accidents, festival, weather, etc [31, 32, 33, 10, 11].

2.2. Deep Learning based traffic passenger Flows Prediction Models

With the growing popularity of deep learning techniques and the success of various deep learning algorithms in many fields such as computer version, natural language processing and speech recognition, recent years, some works also try to apply various neural network architectures in traffic prediction tasks and have achieved state-of-the-art results. To extract spatial patterns, some methods [4, 2, 34, 12, 35, 36, 37, 15] first model traffic flows as multi-channel matrices representation, then employ deep convolution neural networks to learn hierarchical of features. Other methods [18, 21, 20, 23] model traffic flows as traffic transportation networks and stations based graph representation, and employ kinds of graph convolution networks to learn different-levels of features. To extract temporal patterns, similar to object tracking in video, some advanced methods [22, 18, 13, 14] either employ recurrent neural networks or residual based periodic sampling techniques [4, 5] to learn temporal features. For example, Deep CNNs based DeepST [38], ST-ResNet [4], AttConvLSTM [39], DMVST-Net [40], and GCNs based DCRNN [18], STGCN [23] and G-CNN [21], have been proved to achieve the state-of-art performance. ST-ResNet [4] introduces four major components to model the temporal closeness, period, trend and external information, and utilizes the residual neural network to predict the crowd flows of a city. Although ST-ResNet can incorporate prior impacts to sample the input historical traffic flow data in different time periods for forecasting the future traffic flows, it still lacks the analysis of the potential relationship among traffic flows and stations. AttConvLSTM [39] employs convolutional LSTM units and the attention mechanism to emphasize the effects of representative citywide demand patterns on each-step prediction. However, it ignores the periodicity of the traffic flow data. DMVST-Net [40] employs local CNN, LSTM and semantic graph embedding to integrate the spatial, temporal, and semantic views, respectively. DCRNN [18] captures the spatial dependency by using bidirectional random walks on the transportation network and the temporal dependency by using the encoder-decoder architecture with scheduled sampling. STGCN [23] integrates traffic transportation networks based graph convolution and gated temporal convolution. However, both DCRNN [18] and STGCN [23] models ignore the fact that the relationship among traffic flows in different stations is not equivalent to the traffic transportation network topological structures. Different from the above two graph models, G-CNN [21] quantifies the relationship among traffic flows by measuring traffic flows between different stations. Compared to the proposed Dynamic-GRCNN model, the G-CNN [21] belongs to static graph model, and ignores both external influences and periodic laws of traffic flows.

3. Incidence Dynamic Graph Construction

In this section, we first give some definitions, concepts and necessary notations. Then, we give a brief introduction on the traffic passenger flow data studied in this work, and present how to convert urban transportation network topological structure to the spatial-temporal incidence graph of traffic stations representation based on historical traffic passenger flows records.

Definition 1 *Incidence Dynamic Graph.* In this study, we model real-world urban transportation network topological structure as an incidence dynamic graph function $G = f(V, E, W, A, t)$, where vertex V refers to traffic station, directed edge E refers to an incidence relationship between stations at time t, W refers to the weights of the edges at time t, and A refers to the inflow and outflow of the stations.

Different from the static line relationship between stations in the actual urban transportation network topological structure, the incidence relationship between stations refers to a probability relationship between the stations in the flow of passengers in and out. For example, in the Beijing subway transportation network, a large number of passengers travel between Huoying subway station (residence) and Xizhimen subway station (work) every day. Although the two subway stations are separated by at least 8 stations on the subway transportation network, there are two direct connections between the two stations on the incidence graph, and the weight of this interaction relationship is the probability of passenger flow distribution, which checked in from one station and checked out in the other station. In historical traffic flow, the more frequent the interaction between stations, the greater the weight of the edges. In this study, the raw data is the travel records of the anonymous passengers collected from Beijing public transport system. Each record contains the traffic route number/train number, the entrance station ID and time, the exit station ID and time and the locations (longitude, latitude).

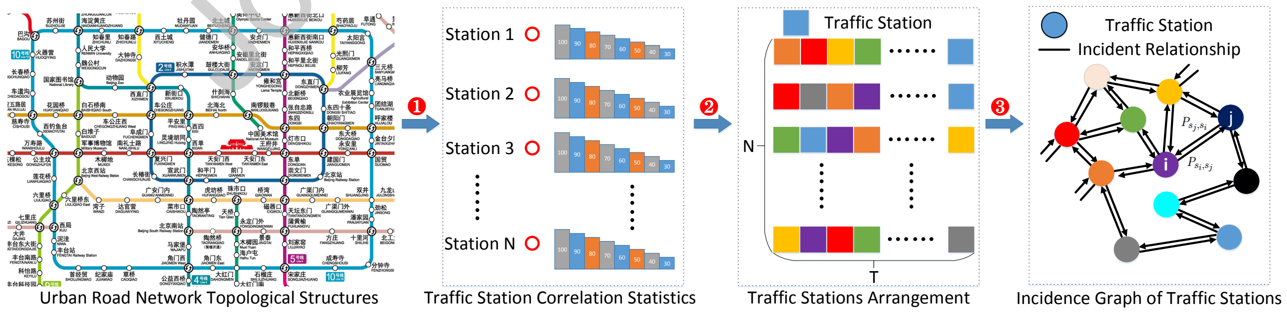

Figure 1: Illustration of converting urban transportation network topological structure to an incidence graph of traffic stations representation based on historical traffic passenger flows.

Next, we present how to build an incidence dynamic graph of traffic stations representation from historical traffic passenger flows. First, we assume the total number of station is N, and select the historical traffic flows F from start time t_s to end time t_e. The traffic flows of any station $s_i, i \in [1, N]$ include the total numbers of check-in passengers X_{in,s_i} and check-out passengers X_{out,s_i}, respectively, in the historical traffic flows. Second, we count the total number of traffic passenger flows

T_{s_i,s_j} from any source transportation station s_i to any destination station s_j in the historical traffic flows. Therefore, we can calculate a probability from the station s_i to the station s_j, as following:

$$\widetilde{p}_{s_i,s_j} = \frac{\sum_{t=t_s}^{t_e} X_{out,s_j}(t)}{\sum_{t=t_s}^{t_e} X_{in,s_i}(t)}. \tag{1}$$

So, for each traffic station, we can count the above probability between traffic stations from the historical passenger flows. For example, the Huilongguan community, having Huilongguan Subway Station, in Beijing suburb is home to IT engineer families of 500,000 people, and most IT engineers work in Zhongguancun, having Zhongguancun Subway Station, in Beijing city. There is a large amount of daily commute traffic passenger flows between the two subway stations. Therefore, the probability between passengers in both Huilongguan Subway Station and Zhongguancun Subway Station are relatively large. As shown in the step 1 of Figure 1, we select the top T relevant traffic stations for each traffic station from the historical traffic passenger flows, to represent its neighbor nodes. Note that the number of T depends on the connectivity of urban traffic flows between traffic stations. In general, more than 80% of the traffic flow needs to be associated with selected stations. Third, we give the normalized probability as

$$P_{s_i,s_j} = \frac{\widetilde{p}_{s_i,s_j}}{\sum_{k=1}^{T} \widetilde{p}_{s_i,s_k}}, j \in [1, T] \tag{2}$$

for the traffic station s_i to the top T relevant traffic stations, respectively, where $\sum_{k=1}^{T} P_{s_i,s_k} = 1$. Then, we rank the T relevant traffic stations in descending order of the normalized probability, and we get a $N \times T$ traffic stations arrangement matrix, as shown in the step 2 of Figure 1. Here, the same color represents the same station in the step 2 of Figure 1. Therefore, we can build a weighted and directed traffic station passenger flows network, namely incidence graph, where the vertex refers to traffic station, the directed edge refers to one relevant relationship, and the weight is incidence probability P_{s_i,s_j} from traffic station i to traffic station j, as shown in the step 3 of Figure 1.

However, the structure of the incidence graph will be affected by many external factors, such as weekends, concerts, gatherings, sports events, etc. For example, at weekends, tourist attractions and fitness and entertainment venues, such as the Bird's Nest Subway Station and the Xiangshan Subway Station, have a relatively large traffic flows from the residential area, such as Huilongguan community, Tiantongyuan community, Wangjing community, etc. So, the incidence graph will be dynamically changing over time and other external factors. Different from external parameters estimation and naive learning based models [4, 11], we employ the incidence dynamic graph to model the more adaptable and flexible relationship among traffic stations. Therefore, our incidence dynamic graph can be adjusted automatically or semi-automatically based on external factors, such as weekdays and weekends, holidays, large gatherings, extreme weather, etc. At last, similar to the multi-channel representation in image or text, for any traffic station $s_j, i \in [1, N]$, there are two properties of check-in and check-out passenger flows as $X_{in,s_j}(t)$ and $X_{out,s_j}(t)$ at time slot t. Next, we introduce how to implement traffic flow prediction model based on the spatial-temporal dynamic incidence graph.

4. Dynamic Graph Recurrent Convolutional Neural Network

After modeling both the complex traffic passenger flows and urban road network topological structures as dynamic incidence graph, we introduce the proposed dynamic graph recurrent convolutional neural network framework in detail. Before using the dynamic graph recurrent convolutional neural network, we first convert the incidence graph into matrix based traffic passenger flows representation. Then, we make use of the dynamic graph recurrent convolutional neural network to learn different levels of spatial and temporal features of traffic flows to predict the future traffic passenger flows.

4.1. Graph Processing

Similar to the standard process in learning convolutional neural networks for graphs [41, 42, 43], the converting from graph data to matrix contains four major parts, the incidence graph generation unit, the subgraph generation unit, normalized matrix unit and the traffic flows matrix representation unit. The illustration of converting spatial-temporal incidence graph to matrix based representation is shown in Figure 2.

First, when converting the real traffic passenger flows of stations to the weighted and directed incidence graph, we can follow the steps of data processing and graph modeling presented in Section 3. We have denoted the incidence graph as $G = f(V, E, W, A, t)$. We also see that the weighted and directed incidence graph can be updated by changing of the relationship of passenger flows between traffic stations and external urban events, such as weekends, concerts, gatherings, sports events, etc. Therefore, the incidence graph will dynamically change over external urban events and time.

Second, we extract top-K central vertexes (traffic stations) from the incidence graph G based on vertex's *closeness centrality* feature. Here, in order to calculate *closeness centrality* for each vertex (traffic station), we use $d(v, u)$ to denote the

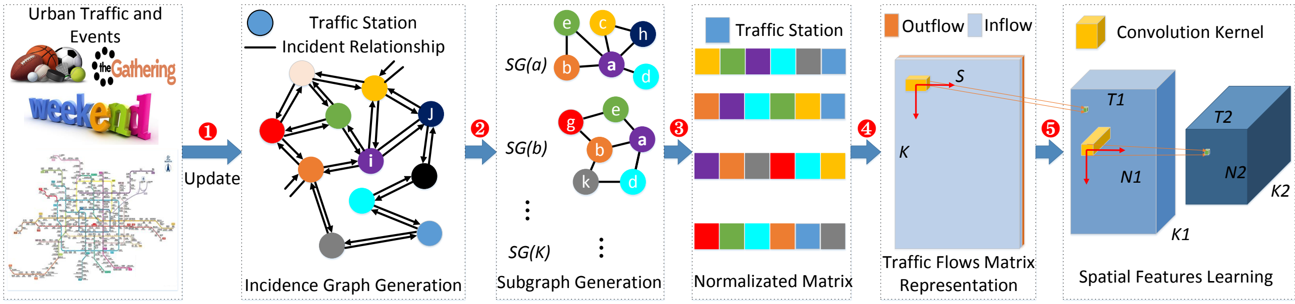

Figure 2: Illustration of converting urban traffic passenger flows to matrix based representation and spatial features learning by convolutional neural networks.

weighted and directed based shortest-path distance between any vertexes v and u. For each vertex v, its *closeness centrality* can be calculated as following,

$$C_v = (n-1)/\sum_{u\in V, u\neq v} d(v,u). \tag{3}$$

We sort the vertexes according to their *closeness centrality* features in descending order, and then select the Top-K central vertexes (traffic stations). Considering both the computational complexity and the representative traffic stations are guaranteed, the value of K should be more than half of the total number of stations in our work.

Third, we extract the vertexes and edges from the neighborhood of each central vertex in the order of breadth first search (BFS), depth first search (DFS) and the vertex's *closeness centrality* feature to build a subgraph. Meanwhile, we limit the number of vertexes in the subgraph to be no more than S, as shown in the step 2 of Figure 2. In this way, the sub-graph $SG(i), i \in [1, K]$ contains both the non-consecutive and long-distance information of the i-th central vertex in the incidence graph. Given a sub-graph, we want to have an order of vertexes for a convolution mask to convolve. Thus, a labeling of stations is expected to make the convolution consistent over all sub-graphs and across graphs. An optimal labeling is defined as follows. Suppose graphs GA and GA' with S vertexes are in a collection of graphs $\mathcal{GA}$. Given the labeling s of a graph GA, we can construct an adjacency matrix $\mathbf{A}^s(GA)$. Then an optimal labeling is defined as

$$s^* = \arg\min_s E[D_{\mathbf{A}}(\mathbf{A}^s(GA), \mathbf{A}^s(GA')) - D_{GA}(GA, GA')], \tag{4}$$

where $D_{\mathbf{A}}(\cdot,\cdot)$ is a distance measure of two matrices, such as $||\mathbf{A} - \mathbf{A}'||_{L1}$, and $D_{GA}(\cdot,\cdot)$ is a edit distance measure of two graphs. However, such labeling is NP-hard and thus we follow [41] to have an alternative labeling. Starting from the root, which is the node that triggered the sub-graph in previous step, we first follow breadth-first-search to use the depth to rank the stations. Then in the same tier (depth) of the graph based spanning tree, we use the degree to rank the stations. Then if two stations in the same tier have the same degree, we further use other factors to break the tier, such as the edges used in previous step. Then after the above guarantee, we have S stations for each sub-graph. In general, the above extracted subgraphs must cover all traffic stations. For the sub-graphs with more than S stations in the previous step, we simply use the rank filter out them. For the sub-graphs with less than S stations, we add some dummy stations disconnected to any stations in the graph. We can easily see that, in this way, the normalization applied to a 2-D lattice such as an image, will be exactly the same as the way CNN's first layer is applied to images. The complexity of this sub-graph normalization is given by [41]. Practically, the complexity can be at most $O(KS^2)$ where K is the number of selected central stations and S is the size of the sub-graph. To further save the above information of the subgraph, we order the stations in the sub-graph $SG(i)$ by their node's (traffic station) *closeness centrality* feature. As a result, we can normalize each subgraph as a sequence of nodes (traffic stations) that keeps the same length S. Here, the same traffic station may appear in different sequences. If the number of stations in the sequence is less than S, it is padded with zeros. Finally, we concatenate all the normalized sequences of the K central stations into an arranged normalized matrix, as shown in step 3 of Figure 2.

Fourth, for better representing the original traffic passenger inflows and outflows in the matrix, we use 2 channels of matrix to represent them. In this way, we have a 3-dimensional tensor $K \times S \times 2$ representation for one time slot of traffic passenger flows, where the padded vectors are zero vectors with the same dimension, as shown in step 4 of Figure 2. So far, we can use convolutional neural networks or recurrent neural networks to learn different levels of spatial-temporal features. Then the deep convolution and recurrent neural networks introduced in the next section will be operated over the unified representations of the urban traffic passenger flows.

4.2. Dynamic Recurrent Convolutional Neural Network

As shown in step 5 of Figure 2, we employ two-layers of convolutional neural networks to learn spatial features for each time slot of traffic passenger flows. In the first layer of convolutional neural network, the size of convolution kernel is

$1 \times 3 \times 2$ with a horizontal stride of 2 elements and a vertical stride of 1 element to extract local features among incidence traffic stations. Here, the first layer uses $K1$ convolution kernels to generate a $T1 \times N1 \times K1$ feature map. Similar, in the second layer of convolutional neural network, the size of convolution kernel is $1 \times 3 \times K1$ with a horizontal stride of 2 elements and a vertical stride of 1 element, and it uses $K2$ convolution kernels to generate a $T2 \times N2 \times K2$ feature map. We use *ReLU* as the activation function to speed up the training process and avoid over-fitting. Here, convolution kernel serves as a composition of the semantics in the receptive field to extract the higher level spatial features. Next, we employ the recurrent neural operators, such as LSTM units, periodic sampling and importance sampling to capture high-levels of temporal features for traffic passenger flows of stations.

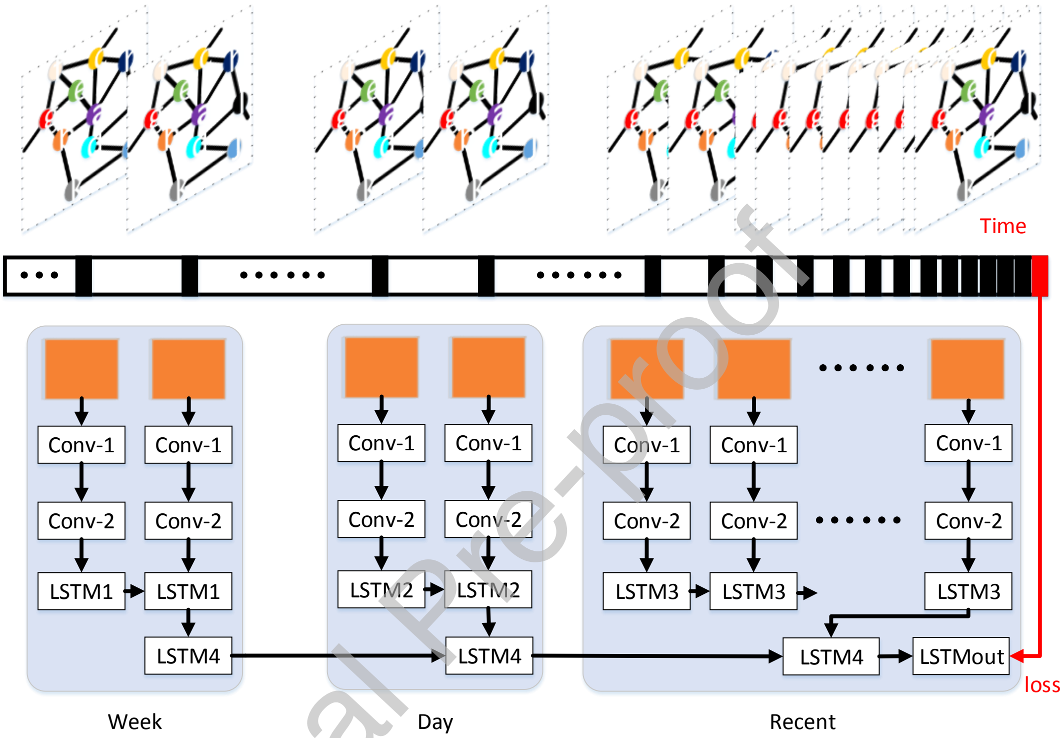

Figure 3: The framework of the proposed Dynamic-GRCNN, Conv: Convolution neural network; LSTM: Long Short Term Memory; LSTMout: the output of LSTM.

The proposed dynamic graph recurrent convolutional neural network framework, namely Dynamic-GRCNN, is shown in Figure 3. The framework includes both weekly spatial-temporal features learning (STFL) component, daily STFL component and recently STFL component, which consists of periodic sampling based traffic passenger flows features learning. In addition to the above periodic sampling, we also employ an importance sampling to select more recent samples to optimize the temporal features learning. More specifically, the closer the time, the more samples are sampled. As shown in Figure 3, more dense samples are chosen to train the recently STFL. For the short-term (recent) historic traffic passenger flows component, we sample 12 snapshots of the traffic passenger flows data matrix in different time intervals. Here, the sampled recent data consists of 6 traffic passenger flows samples in the last 6 time slots, 3 samples in the middle 6 time slots, 3 samples in the front 3 time slots. For the medium-term (daily) historical traffic flows component, we sample 2 samples of previous days in the same time interval. Similarly, the week component also samples 2 samples of previous week in the same time interval. More formally, the convolution operator can be defined as

$$x_j^l = f(\sum_{i \in M_j} x_i^{l-1} \cdot k_{ij}^l + b_j^l), \tag{5}$$

where x_j^l represents the j-th feature map of the l-th layer of the convolution network. This formula shows the convolution operation and the summation for all the associated feature maps x_i^{l-l} and the j-th convolution kernel k_{ij}^l of layer l, and then add an offset parameter b_j^l. Here, a *ReLU* activation function f is applied. The output of the convolution network is input to a

LSTM unit, and output of the LSTM unit is the feature map. The LSTM unit can be defined as:

$$\begin{aligned}
f_t &= \sigma_g(W_f x_t + U_f c_{t-1} + b_f),\\
i_t &= \sigma_g(W_i x_t + U_i c_{t-1} + b_i),\\
o_t &= \sigma_g(W_o x_t + U_o c_{t-1} + b_o),\\
c_t &= f_t c_{t-1} + i_t \sigma_c(W_c x_t + b_c),\\
h_t &= o_t \sigma_h(c_t),
\end{aligned} \tag{6}$$

where t refers to the index of the current feature map generated by convolutional sequences. f_t refers to the forgotten gate, i_t refers to the input gate, o_t refers to the output gate, and c_t is the cell state based on the previous time $t-1$. Finally, although the proposed incidence dynamic graph will change over time, the LSTM units can still learn the temporal impacts based on the states before and after with supervised learning.

For each traffic passenger flows at a given interval, we first convert it to incidence graph, as presented in Section 3. Second, we apply the proposed graph modeling technology, as presented in Section 4.1, to convert the incidence graph to the matrix representation. Third, we use the two layers of convolutional neural networks to learning the $T2 \times N2 \times K2$ size of spatial features. Then, we use three LSTM modules to integrate the spatial-temporal features of long-term, medium-term and short term of traffic passenger flows, respectively. As shown in Figure 3, the LSTM1, LSTM2 and LSTM3 units refer to the three LSTM modules. Next, we employ the fourth LSTM module to integrate the spatial-temporal features of the above three components. The output of the fourth LSTM module is two layers of full-connected neural networks, and the number of neurons in the output layer is equal to $2N$, where each neuron refers to one inflow or outflow for traffic station, respectively. Since this is a supervised model, we use the RMSE and MAE to measure the loss between the predicted values and ground truth flows of stations, as shown in Figure 3. Here the traffic passenger flows is also normalized to the range 0 to 1. We calculate the accumulated loss in min-batch of 32. To speed up training, we choose the momentum [44] method on multiple GPUs.

5. Transportation Mode Setting Through Traffic Passenger Flows Datasets

In this section, we first introduce the three different traffic passenger flows datasets in detail, including subway, bus and taxi in Beijing city, used in this work, and describe how to divide the training set and test set. Second, we present a set of state-of-the-art models and associated datasets.

5.1. Traffic Flows Datasets for Transportation Mode Setting

We choose the following three datasets for evaluation, including the Beijing subway, Beijing bus and Beijing taxi datasets. A summarization of the statistics of these three datasets is shown in Table 1. We can see that the three datasets represent different types of transportation modes, respectively.

Table 1: Datasets Statistics

Dataset	SubwayBJ	BusBJ	TaxiBJ
Data type	e-card	e-card	Taxi GPS
Location	Beijing	Beijing	Beijing
Time Span	7/1/2016-30/10/2016	7/1/2016-30/10/2016	7/1/2013-30/10/2016
Stations and Lines	329, 18	2321, 1020	-
Data frequency	15 minutes	15 minutes	30 minutes

• **Beijing Subway (SubwayBJ)**: The Beijing subway data is collected from anonymous passengers' check-in and check-out records of their Beijing metro-card system. The time span of this data is from 1st Jul.2016 to 30th Oct. 2016. Beijing subway has 329 traffic stations, 18 lines, and the time interval between statistical inflows and outflows is 15 minutes. For this work, we obtain both inflows and outflows from the 329 traffic stations, and construct the corresponding incidence graph structures with 329 nodes for this dataset. The data in the first three and a half months are used for training, and the remaining data are used for testing.

• **Beijing Bus (BusBJ)**: The Beijing Bus data is also collected from anonymous passengers' getting on and getting off records of Beijing buses by their bus e-cards system. The time span is from 1st Jul.2016 to 30th Oct. 2016. Beijing bus has 2321 traffic stations, 1020 lines, and the time interval between statistical inflows and outflows is also 15 minutes. We obtain two types of crowd flows, and construct the corresponding incidence graph structures with 2321 nodes for this dataset. The first two months data are used for training, and the remaining one month data are used for testing.

• **Beijing Taxi (TaxiBJ)**: The Beijing Taxi data is collected from the taxicab GPS data in Beijing in four time intervals: 1st Jul.2013 - 30th Oct. 2013, 1st Mar. 2014 - 30th Jun. 2014, 1st Mar. 2015 - 30th Jun. 2015, 1st Nov. 2015 - 10th Apr.2016. We map the traffic passenger flows of this data into 2 channels, and construct the corresponding incidence graph structures with 128×128 nodes according to latitude and longitude coordinates for this dataset. The data of the last four weeks are the testing data, and the other data are training data.

5.2. Candidate Models and Benchmark

We consider a set of existing models as candidate models for comparison and associated datasets in order to perceive the benefits of Dynamic-GRCNN.

• **Historical Average (HA)** [45]: It simply uses the historical average of the same time period and same traffic station as the prediction. For example, to predict the traffic passenger flows of station s_i in 9:00am-9:30am, we use the average traffic passenger flows of station s_i in all the previous days in the same time interval 9:00am-9:30am as the prediction.

• **Auto-Regressive Integrated Moving Average (ARIMA)** [6]: It is a well-known model for understanding and predicting the future trends of time series data, and widely used in traffic flow prediction.

• **SARIMA** [46]: It is a seasonal ARIMA model, and considers the seasonal terms, capable of both learning closeness and periodic deeyond ARIMA.

• **Vector Auto-Regressive (VAR)** [27]: It captures the pairwise relationships among all flows, which is an advanced spatio-temporal model and has heavy computational costs due to the large number of parameters.

• **ST-ANN** [47]: It extracts spatial (nearby 8 traffic station values) and temporal (8 previous time intervals) correlated traffic passenger flows data as the input, and then they are fed into an artificial neural network.

• **DeepST** [38]: It is a deep neural network based prediction model, and models the spatial-temporal data as temporal closeness, period and seasonal trend. This model shows state-of-the-art results on the crowd flow prediction.

• **ST-ResNet** [4]: It is currently the state-of-the-art deep convolution-based residual networks for predicting the future urban traffic passenger flows.

• **AttConvLSTM** [39]: It employs an encoder-decoder framework based on convolutional and attentional LSTM to capture the spatial-temporal features. It is a state-of-the-art model for multi-step passenger demands prediction in the mobility-on-demand services.

• **DMVST-Net** [40]: DMVST-Net is a deep multi-view spatial-temporal neural network model for taxi demand prediction. It incorporates information of the following three views: temporal view, spatial view and semantic view.

• **DCRNN** [18]: DCRNN is a diffusion convolutional recurrent neural network based model for traffic forecasting. It uses bidirectional graph random walk to model the spatial dependency and recurrent neural network to capture the temporal dynamics.

• **STGCN** [23]: STGCN model integrates both graph convolution network and gated temporal convolution network to form spatio-temporal convolutional blocks.

• **T-GCN** [24]: T-GCN model combines the graph convolution network and the GRU. The graph network models the urban transportation network in which the nodes on the graph represent roads, the edges represent the connection relationships between roads.

• **GCNN** [21]: GCNN model quantifies the relationship among traffic flows by measuring traffic flows between different stations. Compared to the proposed Dynamic-GRCNN model, the GCNN model belongs to static graph model, and ignores both external influences and periodic laws of traffic flows.

We use both the Rooted Mean Square Error (RMSE) and the Mean Average Error (MAE) as the evaluation metrics,

$$MAE = \frac{1}{z} \sum_i ||x_i - \hat{x}_i||, \tag{7}$$

$$RMSE = \sqrt{\frac{1}{z} \sum_i (x_i - \hat{x}_i)^2}, \tag{8}$$

where $\hat{x}$ and x are the predicted value and the ground truth of passenger flows of traffic stations, respectively, and z is the number of all the samples for prediction. For the sake of fairness, we directly calculate the error between the predicted and ground truth values of sum of traffic passenger inflows and outflows for all models. The smaller the value is, and the better the prediction effect is.

6. Experiments and Performance Analysis

In this section, we conduct quantitative evaluation of various models over the three datasets, and analysis the experimental results. First, we give experimental settings including the default model parameters and the experiment environments. Then, we conduct quantitative evaluation of various models over the three datasets, and show the experimental results and analysis. To demonstrate the efficiency of proposed models, we also show the running time of proposed models. Next, we give a visualization analysis in a subway station.

6.1. Experimental Settings

For the input of the proposed Dynamic-GRCNN, we build incidence graph structures based on the number of the traffic stations, and dynamically update the incidence graph by considering weekday, weekends, festivals, concerts, gatherings and large sports events. All of our experiments were performed on 64 core Intel Xeon CPU E5-2680 v4@2.40GHz with 512GB RAM and 4×NVIDIA Tesla P100-PICE GPUs. The operating system and software platforms are Debian 7.0, TensorFlow r0.12 and Python 3.4. The K to be 200, 1500 and 10000 in the subway, bus and taxi datasets, respectively. The S to be 20, 50 and 100 in the subway, bus and taxi datasets, respectively. The numbers of convolution kennels $K1$ to be 128, $K2$ to be 256. The parameters of the baseline methods are set following the setting in their original papers. We also use early-stop in all the experiments.

6.2. Experiment Results on SubwayBJ

The experimental results of these methods over the Beijing subway dataset are shown in Table 2. We can observe that the proposed Dynamic-GRCNN model can significantly outperform all the baselines in terms of RMSE and MAE. Compared to the previous state-of-the-art models, Dynamic-GRCNN achieves the lowest RMSE 0.0016 and the lowest MAE 0.0005 among all the methods, and the performance improvements are both significant on the two metrics.

Considering the traffic transportation network topological structures based graph representation, such as STGCN, T-GCN, DCRNN, the experimental comparison on the subway dataset shows the advantage of traffic station relationships based on historical traffic passenger flows. STGCN, T-GCN, DCRNN, GCNN and Dynamic-GRCNN have the same size of graph structure with 329 vertex. The Dynamic-GRCNN model reduces the RMSE from 0.0065 to 0.0016 and the MAE from 0.0013 to 0.0005. Compared to the same traffic station relationships but lacking the dynamically updating of GCNN model, our proposed Dynamic-GRCNN model reduces the RMSE from 0.0058 to 0.0016 and the MAE from 0.0012 to 0.0005. This comparison again demonstrates the advantage of incidence dynamic graph with the external urban events. We also test the Dynamic-GRCNN model without importance sampling (Dynamic-GRCNN (N-IS)) in selecting recent samples, and the RMSE is 0.0042 and the MAE is 0.0011. In summary, the Dynamic-GRCNN model achieves the best results in the subway traffic passenger flows forecasting.

Table 2: Comparison among different methods on SubwayBJ

Models	RMSE	MAE
HA	0.0245	0.0077
ARIMA	0.0193	0.0060
SARIMA	0.0217	0.0071
VAR	0.0174	0.0057
ST-ANN	0.0142	0.0043
DeepST	0.0113	0.0034
STGCN	0.0095	0.0031
AttConvLSTM	0.0081	0.0024
ST-ResNet	0.0079	0.0023
DMVST-Net	0.0072	0.0020
T-GCN	0.0068	0.0017
DCRNN	0.0065	0.0013
GCNN	0.0058	0.0012
Dynamic-GRCNN (N-IS)	**0.0042**	**0.0011**
Dynamic-GRCNN	**0.0016**	**0.0005**

6.3. Experiment Results on BusBJ

Compared to the traffic passenger flows of subway, there is no bus passenger who needs to transfer to different lines or buses in records. Although there are more traffic stations, both more longer intervals between buses and more less numbers of passengers are benefit for accurately passenger flows forecasting. In STGCN, T-GCN, DCRNN, GCNN and Dynamic-GRCNN models, the nodes refer to traffic stations. The experimental results of these methods over the Beijing bus dataset are shown in Table 3.

Generally, previous deep learning models including DeepST, STGCN, AttConvLSTM, ST-ResNet, DMVST-Net, T-GCN, DCRNN and GCNN perform better than traditional shallow methods such as HA, ARIMA, SARIMA, VAR, and ST-ANN. Compared to the traffic transportation network topological structures based graph representation, such as STGCN, T-GCN, DCRNN, the experimental comparison on the subway dataset again shows the advantage of the incidence graph modeling. Here, STGCN, T-GCN, DCRNN, GCNN and Dynamic-GRCNN have the same size of graph structure with 2321 vertex. The Dynamic-GRCNN model reduces the RMSE from 0.0017 to 0.0008 and the MAE from 0.0055 to 0.00031. Compared to the same traffic station relationships but lacking the dynamically updating of GCNN model, our proposed Dynamic-GRCNN model reduces the RMSE from 0.0017 to 0.0008 and the MAE from 0.0054 to 0.00031. This comparison again demonstrates the advantage of incidence dynamic graph with the external urban events. We also test the Dynamic-GRCNN model without importance sampling (Dynamic-GRCNN (N-IS)) in selecting recent samples, and the RMSE is 0.0011 and the MAE is 0.0004.

In summary, the Dynamic-GRCNN model achieves the best results in the bus passenger flows forecasting. It shows again that the incidence dynamic graph based graph convolutional neural networks and LSTM models can better capture the spatio-temporal features than traditional convolution kernel, graph convolution or recurrent networks.

Table 3: Comparison among different methods on BusBJ

Models	RMSE	MAE
HA	0.0082	0.0026
ARIMA	0.0067	0.0022
SARIMA	0.0088	0.0028
VAR	0.0041	0.0012
ST-ANN	0.0034	0.0011
DeepST	0.0028	0.00075
STGCN	0.0026	0.00073
AttConvLSTM	0.0025	0.00072
ST-ResNet	0.0022	0.00070
DMVST-Net	0.0018	0.00057
T-GCN	0.0018	0.00056
DCRNN	0.0017	0.00055
GCNN	0.0017	0.00054
Dynamic-GRCNN (N-IS)	**0.0011**	**0.00040**
Dynamic-GRCNN	**0.0008**	**0.00031**

Table 4: Comparison among different methods on TaxiBJ

Models	RMSE	MAE
HA	57.69	18.91
ARIMA	22.78	7.25
SARIMA	26.88	8.51
VAR	22.88	7.47
ST-ANN	19.57	6.23
DeepST	18.18	6.21
STGCN	18.09	6.03
AttConvLSTM	17.41	6.04
ST-ResNet	16.69	5.41
DMVST-Net	15.57	5.28
T-GCN	15.17	5.06
DCRNN	15.04	5.01
GCNN	14.33	4.74
Dynamic-GRCNN (N-IS)	**13.88**	**4.59**
Dynamic-GRCNN	**10.25**	**3.42**

6.4. Experiment Results on TaxiBJ

Table 4 shows the experiment results of various methods on the Bejing taxi dataset. Different from subway and bus passenger flows having fixed routes, taxi can stay on any road, and it's difficult to build a fixed graph modeling the relationship and vertex of taxi passenger flows. Even, the taxi route is instantly determined by passenger and traffic environments, such as congestion status, passenger's preferences and limited choices of traffic lines, etc. For the sake of simplicity, we divide the different regions according to the latitude and longitude coordinates to represent the vertices, and then use the historical passenger flow data to construct the relationship between the regions to build the incidence dynamic graph structures. For a fair comparison, we take the regions based matrix representation or graph representation as input data for all the deep models.

We can see that the proposed Dynamic-GRCNN model still achieves the best performance with the smallest RMSE value 10.25 and MAE value 3.42. Because of the uncertainty and the randomness of the taxi flows, the traditional regressive and average based models cannot achieve satisfied performance. The matrix representation based deep learning models, such as DeepST, ST-ResNet and DMVST-Net, have significantly reduced errors. Both traffic transportation network topological structures based graph representation, such as STGCN, T-GCN, DCRNN, and historical passenger flows based graph representation, such as GCNN and Dynamic-GRCNN, have better ability to learn spatio-temporal features than matrix representation based deep learning models. Considering the above two types of graph models, the Dynamic-GRCNN model reduces the RMSE from 15.04 to 10.25 and the MAE from 5.01 to 3.42. Compared to the same traffic station relationships but lacking the dynamically updating of GCNN model, our proposed Dynamic-GRCNN model reduces the RMSE from 14.33 to 10.25 and the MAE from 4.75 to 3.42.

This comparison again demonstrates the advantage of incidence dynamic graph with the external urban events. We also test the Dynamic-GRCNN model without importance sampling (Dynamic-GRCNN (N-IS)) in selecting recent samples, and the RMSE is 13.88 and the MAE is 4.59.

6.5. Computational Efficiency Analysis

We also compare our models trained with different devices with different settings on Dynamic-GRCNN model, shown in Table 5. It shows that GPUs can speed up the training time by at least 7 times for the three datasets and also achieve comparable performance than CPUs. Based on the same batch size, the train time of 1-Batch depends on the size of the incidence graph and normalizated Matrix. The 1-Batch training time for the TaxiBJ data is much more than others, because the number of vertex is 128×128. The subway traffic passengers flows prediction is the most efficient and effective than other situations with less transportation hubs and more regular flows.

Table 5: Comparison of training time based on GPUs and CPUs. (Test evaluations for all the models were performed by CPUs.)

Type	Datasets	Batch(s)	Train(h)	RMSE	MAE
CPUs	SubwayBJ	280	2.6	0.0016	0.0005
GPUs	SubwayBJ	35	0.3	0.0016	0.0005
CPUs	BusBJ	680	4.5	0.0008	0.00031
GPUs	BusBJ	86	0.5	0.0008	0.00031
CPUs	TaxiBJ	1740	29	10.28	3.43
GPUs	TaxiBJ	280	4.2	10.25	3.42

6.6. Visualization Analysis

To have a better understanding on the prediction performance of Dynamic-GRCNN model, we visualize the ground-truth and forecasting results in the Xizhimen subway station within 24 hours, as shown in Figure 4. Xizhimen subway station is a busy transfer station with three traffic lines 2, 4 and 13. From the figure, we can have the following observations. First, the highest peak period of traffic flows is the working time period around 8:30 in the morning, and our Dynamic-GRCNN model can accurately predict passenger flow peaks in rush hours of a day as well as multiple local peaks. Second, Dynamic-GRCNN also performs well in predicting some sudden changes in the real passenger flows from 12:00 to 21:00. This is probably because Dynamic-GRCNN can well capture the spatial-temporal features of passenger flows in associated neighborhood from multi-channel lines.

7. Conclusion

In this paper, we have proposed a novel spatial-temporal incidence dynamic graph neural networks framework for forecasting the flows of crowds in transportation stations of a city. For the first time, we have modeled an incidence dynamic graph

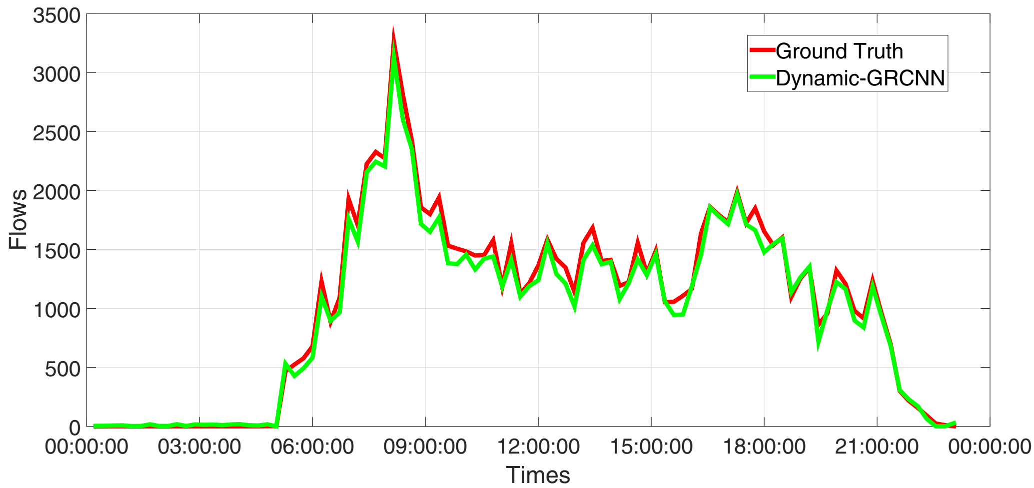

Figure 4: Ground-truth and predicted daily traffic passengers flows of Xizhimen Subway Station in 28/10/2016.

structures by the statistical historical traffic passengers inflows and outflows among traffic stations to consider both internal relationships between traffic stations and external changes and influences. Our Dynamic-GRCNN model has integrated the relationship of passengers, graph convolutional neural network and LSTM units to learn complex traffic spatial-temporal features, and made use of an importance sample strategy to optimize the temporal feature learning. We have evaluated our model on three types of crowd flows in Beijing city, achieving performances which are significantly beyond 13 mainstream baseline methods, confirming that our model is better and more applicable to the crowd flows prediction tasks.

In the future, we plan to apply our dynamic-GRCNN model to real-time and fine-grained urban traffic flows forecasting and warning system, and extend our model to more complex traffic order forecasting and route planning tasks.

Acknowledgment

This work is supported by National Natural Science Foundation of China under Grant (51822802, 51778033, 61772151, U1811463), National Key R&D Program of China through grant (2018YFB2101003, 2017YFB0803305, 2017YFB1400702), and the Science and Technology Major Project of Beijing under Grant No. Z191100002519012. Md Zakirul Alam Bhuiyan is supported by Fodham University Faculty Startup Grant. Philip S. Yu is supported by NSF through grants III-1526499, III-1763325, III-1909323, SaTC-1930941, and CNS-1626432.

References

References

[1] Y. Lv, Y. Duan, W. Kang, Z. Li, F.-Y. Wang, Traffic flow prediction with big data: a deep learning approach, IEEE Transactions on Intelligent Transportation Systems 16 (2015) 865–873.

[2] X. Ma, Z. Dai, Z. He, J. Ma, Y. Wang, Y. Wang, Learning traffic as images: a deep convolutional neural network for large-scale transportation network speed prediction, Sensors 17 (2017) 818.

[3] B. Du, C. Liu, W. Zhou, Z. Hou, H. Xiong, Catch me if you can: Detecting pickpocket suspects from large-scale transit records, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2016, pp. 87–96.

[4] J. Zhang, Y. Zheng, D. Qi, Deep spatio-temporal residual networks for citywide crowd flows prediction., in: AAAI, 2017, pp. 1655–1661.

[5] J. Zhang, Y. Zheng, D. Qi, R. Li, X. Yi, T. Li, Predicting citywide crowd flows using deep spatio-temporal residual networks, Artificial Intelligence 259 (2018) 147 – 166.

[6] M. M. Hamed, H. R. Al-Masaeid, Z. M. B. Said, Short-term prediction of traffic volume in urban arterials, Journal of Transportation Engineering 121 (1995) 249–254.

[7] S. Thajchayapong, J. A. Barria, E. Garcia-Trevino, Lane-level traffic estimations using microscopic traffic variables, in: International IEEE Conference on Intelligent Transportation Systems, 2010, pp. 1189–1194.

[8] I. Okutani, Y. J. Stephanedes, Dynamic prediction of traffic volume through kalman filtering theory, Transportation Research Part B: Methodological 18 (1984) 1–11.

[9] Z. Zhu, B. Peng, C. Xiong, L. Zhang, Short-term traffic flow prediction with linear conditional gaussian bayesian network, Journal of Advanced Transportation 50 (2016) 1111–1123.

[10] S. Wang, X. Zhang, J. Cao, L. He, L. Stenneth, P. S. Yu, Z. Li, Z. Huang, Computing urban traffic congestions by incorporating sparse gps probe data and social media data, ACM Transactions on Information Systems (TOIS) 35 (2017) 40.

[11] S. Wang, L. He, L. Stenneth, S. Y. Philip, Z. Li, Z. Huang, Estimating urban traffic congestions with multi-sourced data, in: 2016 17th IEEE International conference on mobile data management (MDM), volume 1, IEEE, 2016, pp. 82–91.

[12] N. G. Polson, V. O. Sokolov, Deep learning for short-term traffic flow prediction, Transportation Research Part C: Emerging Technologies 79 (2017) 1–17.

[13] Y. Tian, L. Pan, Predicting short-term traffic flow by long short-term memory recurrent neural network, in: 2015 IEEE international conference on smart city/SocialCom/SustainCom (SmartCity), IEEE, 2015, pp. 153–158.

[14] Z. Zhao, W. Chen, X. Wu, P. C. Chen, J. Liu, Lstm network: a deep learning approach for short-term traffic forecast, IET Intelligent Transport Systems 11 (2017) 68–75.

[15] Z. Zhene, P. Hao, L. Lin, X. Guixi, B. Du, M. Z. A. Bhuiyan, Y. Long, D. Li, Deep convolutional mesh rnn for urban traffic passenger flows prediction, in: 2018 IEEE SmartWorld, IEEE, 2018, pp. 1305–1310.

[16] M. Fouladgar, M. Parchami, R. Elmasri, A. Ghaderi, Scalable deep traffic flow neural networks for urban traffic congestion prediction, in: 2017 International Joint Conference on Neural Networks (IJCNN), IEEE, 2017, pp. 2251–2258.

[17] B. Du, H. Peng, S. Wang, M. Z. A. Bhuiyan, L. Wang, Q. Gong, L. Liu, J. Li, Deep irregular convolutional residual lstm for urban traffic passenger flows prediction, IEEE Transactions on Intelligent Transportation Systems (2019).

[18] Y. Li, R. Yu, C. Shahabi, Y. Liu, Diffusion convolutional recurrent neural network: Data-driven traffic forecasting, in: International Conference on Learning Representations (ICLR '18), 2018.

[19] Q. Zhang, Q. Jin, J. Chang, S. Xiang, C. Pan, Kernel-weighted graph convolutional network: A deep learning approach for traffic forecasting, in: 2018 24th International Conference on Pattern Recognition (ICPR), IEEE, 2018, pp. 1018–1023.

[20] M. Wang, B. Lai, Z. Jin, X. Gong, J. Huang, X. Hua, Dynamic spatio-temporal graph-based cnns for traffic prediction, arXiv preprint arXiv:1812.02019 (2018).

[21] J. Li, H. Peng, L. Liu, G. Xiong, B. Du, H. Ma, L. Wang, M. Z. A. Bhuiyan, Graph cnns for urban traffic passenger flows prediction, in: 2018 IEEE SmartWorld, IEEE, 2018, pp. 29–36.

[22] X. Shi, Z. Chen, H. Wang, W. C. Woo, W. C. Woo, W. C. Woo, Convolutional lstm network: a machine learning approach for precipitation nowcasting, in: International Conference on Neural Information Processing Systems, 2015, pp. 802–810.

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

[24] L. Zhao, Y. Song, M. Deng, H. Li, Temporal graph convolutional network for urban traffic flow prediction method, arXiv preprint arXiv:1811.05320 (2018).

[25] B. L. Smith, B. M. Williams, R. K. Oswald, Comparison of parametric and nonparametric models for traffic flow forecasting, Transportation Research Part C Emerging Technologies 10 (2002) 303–321.

[26] B. Williams, P. Durvasula, D. Brown, Urban freeway traffic flow prediction: Application of seasonal autoregressive integrated moving average and exponential smoothing models, Transportation Research Record 1644 (1998) 132–141.

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

[28] L. Lin, J. Li, F. Chen, J. Ye, J. Huai, Road traffic speed prediction: a probabilistic model fusing multi-source data, IEEE Transactions on Knowledge and Data Engineering 30 (2018) 1310–1323.

[29] W. Lv, B. Du, D. Ma, T. Zhu, C. Wang, Applied research of data sensing and service to ubiquitous intelligent transportation system, Frontiers of Computer Science in China 4 (2010) 417–426.

[30] S. Wang, L. He, L. Stenneth, P. S. Yu, Z. Li, Citywide traffic congestion estimation with social media, in: Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems, ACM, 2015, p. 34.

[31] E. Mai, R. Hranac, Twitter interactions as a data source for transportation incidents, Transportation Research Board Annual Meeting (2013).

[32] T. H. Maze, M. Agarwai, G. Burchett, Whether weather matters to traffic demand, traffic safety, and traffic operations and flow, Transportation Research Record Journal of the Transportation Research Board 1948 (2006) 170–176.

[33] A. Schulz, P. Ristoski, The car that hit the burning house: Understanding small scale incident related information in microblogs, in: When the City Meets the Citizen Workshop, the International Conference on Weblogs and Social Media, 2013.

[34] S. Manoharan, Short Term Traffic Flow Prediction Using Deep Learning Approach, Ph.D. thesis, Dublin, National College of Ireland, 2016.

[35] Y.-J. Wu, F. Chen, C.-T. Lu, S. Yang, Urban traffic flow prediction using a spatio-temporal random effects model, Journal of Intelligent Transportation Systems 20 (2016) 282–293.

[36] S. Zhang, G. Wu, J. P. Costeira, J. M. F. Moura, Fcn-rlstm: Deep spatio-temporal neural networks for vehicle counting in city cameras (2017).

[37] J. Ke, H. Zheng, H. Yang, Xiqun, Chen, Short-term forecasting of passenger demand under on-demand ride services: A spatio-temporal deep learning approach (2017).

[38] J. Zhang, Y. Zheng, D. Qi, R. Li, X. Yi, Dnn-based prediction model for spatio-temporal data, in: ACM Sigspatial International Conference on Advances in Geographic Information Systems, 2016, p. 92.

[39] X. Zhou, Y. Shen, Y. Zhu, L. Huang, Predicting multi-step citywide passenger demands using attention-based neural networks, in: Eleventh ACM International Conference on Web Search and Data Mining, 2018, pp. 736–744.

[40] H. Yao, F. Wu, J. Ke, X. Tang, Y. Jia, S. Lu, P. Gong, J. Ye, Z. Li, Deep multi-view spatial-temporal network for taxi demand prediction, in: AAAI, 2018.

[41] M. Niepert, M. Ahmed, K. Kutzkov, Learning convolutional neural networks for graphs, in: International conference on machine learning, 2016, pp. 2014–2023.

[42] H. Peng, J. Li, Y. He, Y. Liu, M. Bao, L. Wang, Y. Song, Q. Yang, Large-scale hierarchical text classification with recursively regularized deep graph-cnn, in: Proceedings of the 2018 World Wide Web Conference on World Wide Web, International World Wide Web Conferences Steering Committee, 2018, pp. 1063–1072.

[43] H. Peng, J. Li, Q. Gong, S. Wang, L. He, B. Li, L. Wang, P. S. Yu, Hierarchical taxonomy-aware and attentional graph capsule rcnns for large-scale multi-label text classification, arXiv preprint arXiv:1906.04898 (2019).

[44] J. Ngiam, A. Coates, A. Lahiri, B. Prochnow, Q. V. Le, A. Y. Ng, On optimization methods for deep learning, in: Proceedings of the 28th international conference on machine learning (ICML-11), 2011, pp. 265–272.

[45] B. L. Smith, M. J. Demetsky, Traffic flow forecasting: comparison of modeling approaches, Journal of transportation engineering 123 (1997) 261–266.

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

[47] H. T. Abdelwahab, M. A. Abdel-Aty, Development of artificial neural network models to predict driver injury severity in traffic accidents at signalized intersections, Transportation Research Record 1746 (2001) 6–13.

Author Contribution Statement:

Peng Hao: Writing - Original Draft, Methodology, Software

Hongfei Wang: Validation, Methodology

Bowen Du: Conceptualization, Formal analysis, Writing - Review & Editing

Md Zakirul Alam Bhuiyan: Conceptualization, Writing - Review & Editing

Hongyuan Ma: Conceptualization, Data Curation

Jianwei Liu: Resources, Investigation

Lihong Wang: Data Curation, Investigation

Zeyu Yang: Visualization, Investigation

Linfeng Du: Visualization, Software

Senzhang Wang: Project administration, Software

Philip S. Yu: Supervision

Conflicts of Interest Statement

Manuscript title: Spatial Temporal Incidence Dynamic Graph Neural Networks for Traffic Flow Forecasting

---

The authors listed in the paper certify that they have NO affiliations with or involvement in any organization or entity with any financial interest (such as honoraria; educational grants; participation in speakers' bureaus; membership, employment, consultancies, stock ownership, or other equity interest; and expert testimony or patent-licensing arrangements), or non-financial interest (such as personal or professional relationships, affiliations, knowledge or beliefs) in the subject matter or materials discussed in this manuscript.

Yours Sincerely,
On behalf of the all authors of the manuscript

Md Zakirul Alam Bhuiyan, PhD, SMIEEE