# Graph Attention Spatial-Temporal Network With Collaborative Global-Local Learning for Citywide Mobile Traffic Prediction

Kaiwen He, Xu Chen, *Senior Member, IEEE*, Qiong Wu, Shuai Yu, *Member, IEEE*, and Zhi Zhou, *Member, IEEE*

**Abstract**—With the rapid development of mobile cellular technologies and the increasing popularity of mobile and Internet of Things (IoT) devices, timely mobile traffic forecasting with high accuracy becomes more and more critical for proactive network service provisioning and efficient network resource allocation in smart cities. Traditional traffic forecasting methods mostly rely on time series prediction techniques, which fail to capture the complicated dynamic nature and spatial relations of mobile traffic demand. In this paper, we propose a novel deep learning framework, graph attention spatial-temporal network (GASTN), for accurate citywide mobile traffic forecasting, which can capture not only local geographical dependency but also distant inter-region relationship when considering spatial factor. Specifically, GASTN considers spatial correlation through our constructed spatial relation graph and utilizes structural recurrent neural networks to model the global near-far spatial relationships as well as the temporal dependencies. In the framework of GASTN, two attention mechanisms are designed to integrate different effects in a holistic way. Besides, in order to further enhance the prediction performance, we propose a collaborative global-local learning strategy for the training of GASTN, which takes full advantage of the knowledge from both the global model and local models for individual regions and enhance the effectiveness of our model. Extensive experiments on a large-scale real-world mobile traffic dataset demonstrate that our GASTN model dramatically outperforms the state-of-the-art methods. And it reveals that a significant enhancement in the prediction performance of GASTN can be obtained by leveraging the collaborative global-local learning strategy.

**Index Terms**—Mobile traffic prediction, collaborative learning, spatial-temporal network, smart city services

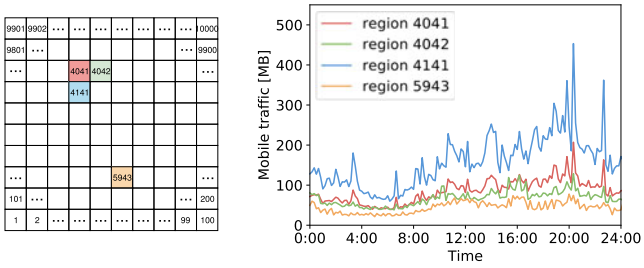
## 1 Introduction

Smart cities, enpowered by a wide variety of Internet of Things (IoT) devices, such as smart phones and wearables, promise to provide intelligent services for the citizens so as to improve quality of service (QoS) and user experience [2], [3]. With the unprecedented development of mobile communication technology, users' demand for mobile services has been increasing rapidly. As the arrival of the fifth generation (5G) era, the demand is forecasted to bloom and increase by threefold over the next five years [4]. To meet users' fast growing demand on mobile services and enhance users' experience, artificial intelligence (AI)-driven cognitive network management for automated resource orchestration and intelligent service management is envisioned as a promising paradigm. To embrace such vision, machine learning based large-scale mobile traffic prediction will be one key building block for precise perception of the fluctuations of users' demand, in order to facilitate smart network resource allocation and proactive service provisioning [5], [6], [7]. For example, based on the precise prediction results of citywide mobile traffic demand in a near future time, service operators are able to proactively optimize the network resource allocation in advance [8], so that regions with low demand can release extra resources while more network resources can be allocated to the regions with huge demand during the peak time. Also, accurate mobile traffic prediction would be critical for efficient base station sleeping control for green cellular networking and dynamic edge computing service deployment in 5G/6G networks.

Even though network traffic exhibits strong dynamics over time, its periodic patterns make it possible for mobile traffic forecasting. Autoregressive integrated moving average (ARIMA) and recurrent neural network (RNN) are widely used for traffic prediction, which are time series forecasting models that focus on leveraging temporal factors [9]. Besides, spatial correlation can also be utilized to promote citywide mobile traffic prediction. Intuitively, areas with the same function have similar traffic patterns, for example, residential areas would have small traffic demand during the day and great demand at night while shopping districts have particularly high demand for mobile traffic on weekends compared to weekdays. In addition, suburbs generally present analogous demand patterns though they are far apart geographically. Thus, this kind of spatial correlation inherent in mobile traffic would contribute to mobile traffic forecasting. Nevertheless, most of the existing studies [10], [11] focus on characterizing spatial correlation of mobile traffic based on physical

• *The authors are with the School of Data and Computer Science, Sun Yat-Sen University, Guangzhou, Guangdong 510006, China. E-mail: {hekaiw, wuqiong23}@mail2.sysu.edu.cn, {chenxu35, yushuai, zhouzhi9}@mail.sysu.edu.cn.*

*Manuscript received 29 Feb. 2020; revised 28 July 2020; accepted 25 Aug. 2020. Date of publication 1 Sept. 2020; date of current version 4 Mar. 2022. (Corresponding author: Xu Chen.)*
*Digital Object Identifier no. 10.1109/TMC.2020.3020582*




(a) Geographical location (b) Traffic volume in one day

Fig. 1. The relationship between geographical location and traffic volume.

proximity, without considering the fact that two regions with similar traffic patterns can be far apart geographically. As an illustration, the real-world data shown in Fig. 1 indicates that the mobile traffic demand of region 4041 is similar to that of its adjacent region 4042 as well as the far region 5943 but it is quite different from that of its another adjacent region 4141. Therefore, it is critical to characterize spatial correlation based on such near-far effects for achieving accurate mobile traffic prediction.

With the important insights above, in this paper we propose a novel deep learning framework, Graph Attention Spatial-Temporal Network (GASTN) which integrates temporal features and spatial correlations of different regions in a city for traffic demand forecasting. For spatial domain, existing works [10], [11] tend to treat the traffic volume over a whole area as an image to capture the spatial dependency based on the geographical relationship, which cannot get the best result because the grid structure fails to capture actual spatial relations about mobile traffic across all the regions and will introduce irrelevant regions as neighbors that degrades the performance. To tackle this issue, we first build a spatial relation graph by Dynamic Time Warping (DTW) algorithm to capture spatial relationship between regions instead of utilizing the given grid structure. Then, in order to integrate the graph structural information and fully utilize spatial information, we construct a novel attention-based structural RNN which is able to extract spatial correlation while capturing temporal dependency.

The GASTN model trained by all the data samples in the city (i.e., the global model) is equipped with generalization ability but fails on model personalization, which may result in a certain performance degradation for each region in the city. Instead, the prediction model trained by the data of a single region, denoted as a local model, is regarded to have the capacity to perform personalized prediction for the region but it would be overfitting due to the limited training data samples [12], [13]. Therefore, the global model in cooperation with local models of all regions is able to take full advantage of model's generalization and personalization abilities, further promoting the prediction performance of the global model. Specifically, we conduct collaborative global-local learning for mobile traffic prediction that the global model and all local models are jointly trained iteratively and learn from each other in the training process. In this way, the global model can obtain regional personalized knowledge from local models and thus leverage the useful knowledge for performance improvement.

In summary, the major contributions of this paper are as follows:

- We propose a novel deep learning model named Graph Attention Spatial-Temporal Network (GASTN) for precise mobile traffic prediction that jointly integrates temporal and spatial factors in a holistic manner.
- We build a spatial relation graph according to time series similarity of traffic demand across both near and far regions using the DTW algorithm. We then construct a novel attention-based structural RNN algorithm based on the graph to capture temporal dependency and spatial relationship simultaneously.
- We design a collaborative global-local learning scheme that leverages the generalization capability of the global model and the personalization ability of local models to boost the training performance of GASTN for citywide mobile traffic prediction.
- Extensive experiments are conducted using a large-scale realistic dataset, which reveal that our GASTN model significantly outperforms the state-of-the-art methods. Moreover, a performance gain of 18 percent in terms of MAPE can be achieved with our collaborative global-local learning strategy over the standard model learning scheme.

The rest of this paper is organized as follows. Section 2 reviews the related work of mobile traffic prediction and collaborative learning. Section 3 introduces the details of our proposed model GASTN. Based on our prediction model, we design a collaborative learning scheme in Section 4. The experimental results are discussed in Section 5. The conclusion is given in Section 6.

## 2 Related Work

### 2.1 Mobile Traffic Prediction

Mobile traffic forecasting has recently attracted great attention due to its wide application in mobile network management. Lots of time series models have been proposed considering the temporal patterns of mobile traffic demands. For example, autoregressive integrated moving average (ARIMA) [9], [14], and its variants [15] are widely used for network traffic forecasting by exploring the correlation between timestamps. However, as ARIMA merely depicts a linear relationship of mobile traffic demand and make predictions based on classical statistics, it fails to capture the complicated fluctuations of traffic volume.

In recent years, deep learning has made remarkable achievements in many prediction tasks [16], [17]. RNN is proposed to model the sequence patterns in many areas such as natural language processing [18] and speech recognition [19]. Long-Short Term Memory (LSTM) [20], a variant of RNN, is utilized for mobile traffic forecasting and shows its superiority over traditional time series prediction methods [21]. Nevertheless, all of the above approaches are limited to characterize temporal correlation of mobile traffic.

Another line of studies applied convolutional structures to extract spatial correlations of mobile traffic demand. Huang *et al.* [10] propose an effective multitask learning (MTL) architecture with the combination of convolutional neural network (CNN) and RNN to capture spatial-temporal patterns. Zhang *et al.* [11] introduce a Spatio-Temporal neural Network (STN) architecture that simultaneously exploits spatial and temporal

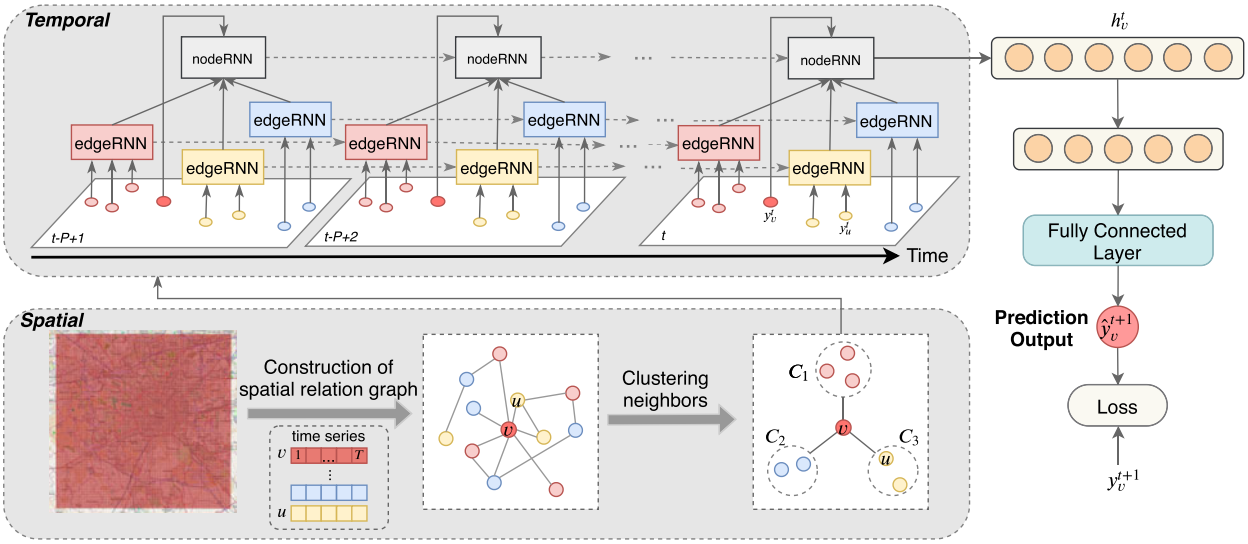



Fig. 2. The architecture of GASTN which is composed of two parts: spatial module and temporal module. The inputs of the spatial module are the mobile traffic sequences of regions in the training data, which are used to calculated the similarity between regions. And the inputs of the temporal module are the the time series with $P$ time slots of the target node $v$ and its neighbors.

correlations of traffic patterns by ConvLSTM [22] and 3D-ConvNet [23]. These CNN-based methods map the spatial distribution to an image and only consider spatial dependency with adjacent areas. In addition, a graph-based method, which employs the graph convolutional LSTM based on a distance-based graph, is proposed to simultaneously characterize both the spatial and temporal aspects of mobile demand forecasting [24]. In this way, the impact of distant regions which have similar traffic demand patterns with the target area is ignored. On the other hand, lots of spatiotemporal forecasting approaches based on graph convolutional network are designed for traffic forecasting on road networks [25], [26], [27] but not for mobile traffic prediction. In general, graph convolutional network is computationally very heavy, and we will explore the effectiveness of such approach for mobile traffic prediction in the future work.

Different from these works, our proposed GASTN model considers both temporal dependency and spatial correlation of mobile traffic demands in different regions. Specifically, to capture near-far spatial correlation, we construct a spatial relation graph according to the similarity of network traffic patterns in different areas, regardless of the actual distance.

### 2.2 Collaborative Learning

Collaborative learning technique is recently gaining much attention in edge computing for AI applications to conduct cooperative training among cloud and edge devices. Daga *et al.* [28] devise a collaborative transfer learning framework among the edges which first figures out logical neighbors for each edge and then collaboratively learns the edge models with their neighbors via knowledge transfer. Besides, Zhang *et al.* propose a collaborative cloud-edge computation system for personalized driving behavior modeling that trains and prunes global model in the cloud and then transfers the learned model to the edges for further personalization [29]. In [30], Lu *et al.* propose to allow cloud and edge devices to learn collectively from each other through continuous dual knowledge distillation process. These methods are mainly designed to improve the robustness and personalization of local models while taking the performance of global model as secondary. With the goal of boosting the overall prediction performance of a city in this paper, we devise a novel collaborative learning method to enhance the global model for citywide mobile traffic prediction.

## 3 GASTN Framework for Mobile Traffic Prediction

In this section, we first formulate the problem of mobile traffic prediction and then introduce our proposed Graph Attention Spatial-Temporal Network (GASTN) in detail. The whole architecture of GASTN is shown in Fig. 2, which consists of two major parts: spatial module and temporal module. In the spatial module, we construct a spatial relation graph to characterize the relationship among the traffic demands in different regions and then figure out their neighbors. Besides, we propose attention-based structural RNN (S-RNN) in the temporal module to capture spatiotemporal features for mobile traffic prediction.

### 3.1 Problem Formulation

In this study, the geographical area of a city can be divided into a $n \times n$ grid map, denoted as $\mathcal{M} = \{1, 2, \ldots, M\}$ with totally $M$ grids, based on the longitude and latitude. Each grid maps to a region in the city. For each region $v$, the traffic volume at time slot $t$ can be represented as $y_v^t$. To make precise mobile traffic forecasting, we aim to predict the traffic demand in region $v$ at time slot $t+1$ given the previous $P$ observed traffic values $\mathcal{Y}_v^{t-P+1:t} = \{y_v^{t-P+1}, \ldots, y_v^{t-1}, y_v^t\}$. Since mobile traffic patterns are affected by both temporal and spatial factors, our mobile traffic prediction problem can be formulated as

$$\hat{y}_v^{t+1} = \mathcal{F}(\mathcal{Y}_v^{t-P+1:t}, \mathcal{X}_v^{t-P+1:t}), \tag{1}$$

for $v \in \mathcal{M}$, where $\mathcal{X}_v^{t-P+1:t} = \{\mathcal{Y}_u^{t-P+1:t} | u \in NB(v)\}$ are the traffic sequences of $v$'s neighbors $NB(v)$ in the same time period, and next we will introduce how to select the neighbors.

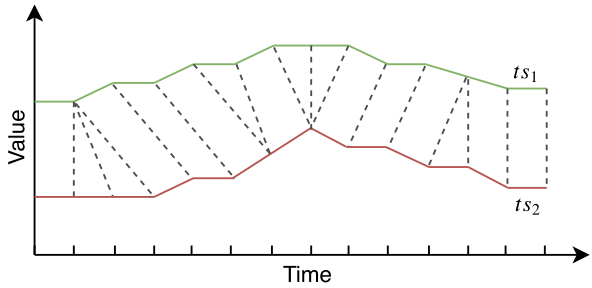



Fig. 3. The alignment process between two time series under DTW algorithm.

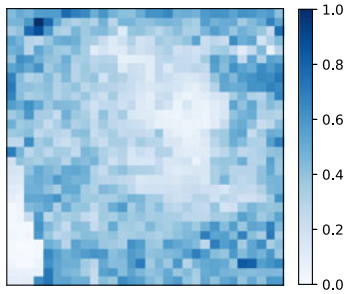



Fig. 4. The normalized average geographic distance between each node and its neighboring nodes.

### 3.2 Construction of Spatial Relation Graph

The grid structure of a city has been widely used to identify the geospatially nearby regions of a given region for mobile traffic forecasting [10], [11]. However, the grid structure can only find the neighbors of the target region based on physical proximity, while ignoring the fact that two regions with similar traffic patterns can be far apart geographically. Thus, to find the neighbors based on traffic similarity, we discard the given grid structure and figure out a novel graph based spatial structure with the divided grids of the city as the nodes in the graph.

To characterize the spatial correlation of mobile traffic demands among the regions, we put forward to construct a weighted graph $G = (\mathcal{V}, \mathcal{E}, \mathcal{W})$, where $\mathcal{V}$ represents the set of all nodes (i.e., regions), $\mathcal{E}$ is the set of edges connecting each two nodes and $\mathcal{W}$ denotes the set of weights for all the edges, and thus $G$ is able to represent the relevancy between every two regions. Intuitively, regions are closely related in terms of mobile traffic when they have similar traffic patterns. As frequent changing dynamics and time shifts exist commonly in mobile traffic demands, traditional static distance metrics such as euclidean distance, are often not suitable to calculate the similarity between two traffic sequences [31] Thus, we propose to adopt Dynamic Time Warping (DTW) algorithm [32], which is competent to capture the pattern similarities between two time series and has been commonly used in various domains such as speech recognition, motion detection, and signal processing [33], [34].

As illustrated in Fig. 3, DTW allows a time series to be "stretched" or "compressed" to provide a better pattern match with another time series. By offering a modified distance metric with alignment, DTW provides an optimal match between two given sequences, and then measures the summation of distances between each pair of the matching points as the similarity score. To be specific, suppose there are two time series $X$ and $Y$ with length $L_X$ and $L_Y$, respectively. A distance matrix $\boldsymbol{D} \in \mathbb{R}^{L_X \times L_Y}$ is first constructed for the two sequences, and the matrix element $\boldsymbol{D}(i, j)$ represents the distance between the $i$th point in $X$ and the $j$th point in $Y$, which is calculated by using some usual distance metrics, such as euclidean distance, Manhattan distance, etc. To find the shortest path from point (1, 1) to point $(L_X, L_Y)$, we define the cumulative distance from point (1, 1) to point $(i, j)$ as $\gamma(i, j)$, which can be calculated by the following equation:

$$\gamma(i,j) = \boldsymbol{D}(i,j) + \min[\gamma(i-1,j), \gamma(i,j-1), \gamma(i-1,j-1)]. \tag{2}$$

As Eq. (2) can be solved by dynamic programming technique, the shortest distance $\gamma(L_X, L_Y)$ can be recursively calculated. Thus, the similarity between the two sequences $X$ and $Y$ is obtained.

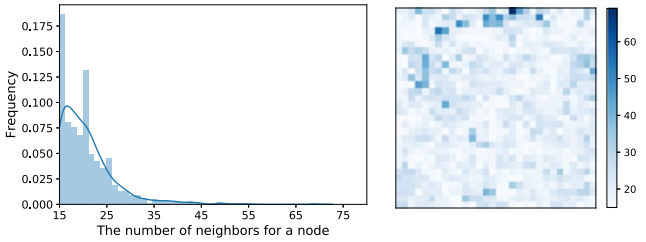
Here, we treat each region as a node and the weight of an edge connecting two nodes is calculated based on DTW. The weight $w_{uv}$ of the edge connecting node $u$ and node $v$ is estimated as

$$w_{uv} = \exp(-\mathrm{DTW}(ts_u, ts_v)), \tag{3}$$

where $\mathrm{DTW}(ts_u, ts_v)$ is the normalized dynamic time warping distance between the traffic sequences $ts_u$ and $ts_v$ for region $u$ and region $v$. Then we get a dense weighted graph where every two nodes are connected by a weighted edge.

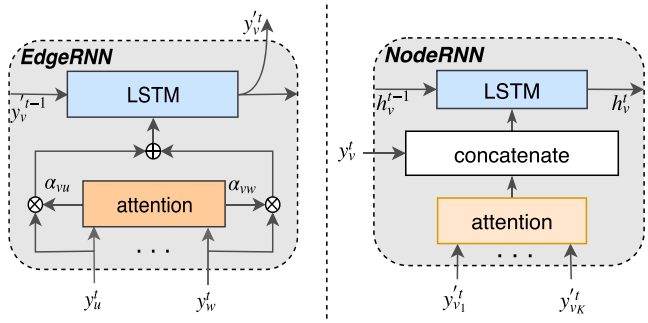
Besides, we analyze the distribution of neighboring nodes. As shown in Fig. 4, we calculate the normalized average value of the geographic distance between each region and all its neighbors on the spatial relation graph. It can be seen that for the central regions they are geographically close to their neighbors while suburbs have relatively long distance with their neighbors. This confirms our inference that suburbs generally present analogous demand patterns though they are far apart geographically and it can also prove the effectiveness of our spatial relation graph to capture the near-far spatial correlation. In addition, we further display the distribution of the number of neighbors for each region in a city in Fig. 5. It reveals that most regions have exactly $N$ neighboring nodes, especially the central regions. And only a few regions have more than $N$ neighbors, for example, a suburban region would have more neighbors since it is affected by many other nearby or remote suburbs.

In order to focus on the strong correlation as well as reduce computation complexity, it is essential to find out the closely related neighbors for each region based on the edge weights and then build a sparse graph by removing low-weight edges in the dense graph. To be specific, we select top-$N$ nearest neighboring nodes for each node according to the weights of its edges and consider the relationship between nodes is symmetric that node $u$ would also be chosen as a neighbor of node $v$ if $v$ is $u$'s neighbor even though $u$ is not in the top-$N$ nearest neighbors of $v$. In this case, some nodes may have more than $N$ neighbors. To analyse the characteristics of neighbors generated by DTW algorithm, we first calculate the normalized average value of the geographic distance between each region and all its




(a) Probability distribution (b) Geographical distribution

Fig. 5. The distribution of the number of neighbors for each region in a city.




(a) edgeRNN (b) nodeRNN

Fig. 6. The architecture of edgeRNN and nodeRNN. (a) The inputs $\{y_u^t, \ldots, y_w^t\}$ of edgeRNN are the traffic volumes of neighbors of target node $v$ at time interval $t$. (b) The output of edgeRNN $i$ for target node $v$, denoted as $e_{v_i}^t$, acts as an input of nodeRNN.

neighbors when $N = 15$ as an example. As illustrated in Fig. 4, the normalized average geographic distance in regions of the city center is much lower than that in the city fringe, which means that the neighbors of central regions are likely to geographically close to their neighbors while the neighbors of regions in city fringe (eg., suburbs) can be far away. This observation further confirms our motivation for the design of spatial relation graph which captures the near-far spatial correlation of mobile traffic demands in different regions. To investigate the number of neighbors for each region after the symmetry operation, we further analyse the probability distribution and geographical distribution of the number of neighbors for the regions in a city as depicted in Fig. 5. It reveals that a portion of regions will have more than $N$ regions with symmetry consideration. Many regions in the city center are more likely to have exactly $N$ neighboring nodes while some regions in city fringe would tend to have a large number of neighbors. Although the number of neighbors varies by regions, there is no significant difference for these regions in subsequent model learning operations.

For ease of computation, we eliminate the edges between each node and its unselected neighbors and then obtain a sparse weighted graph. The resulted spatial relation graph captures the spatial relationship of mobile traffic demands among regions in the city rather than the geo-distance relation. Note that besides for mobile traffic prediction, the similarity based spatial relation graph can be also useful for other applications, e.g., edge resource planning such that we should deploy similar resource amount for the regions with high similarity.

### 3.3 Attention-Based Structural RNN for Traffic Forecasting

Given the spatial relation graph defined above, we propose a graph-based method, attention-based structural RNN (S-RNN) which is inspired by [35], for mobile traffic forecasting. Traditionally, S-RNN is proposed to cast a spatio-temporal graph (st-graph) as a rich RNN mixture and is exploited mainly for computer vision, such as modeling human motion and detecting object interactions [35], and to the best of our knowledge, we are the first to leverage S-RNN for mobile traffic prediction. Moreover, we design two attention mechanisms in the S-RNN architecture for capturing the diverse influence of neighbors and distinct importance of different kinds of edges.

For the S-RNN part, nodes in the graph are first clustered into $K$ classes $\{C_1, \ldots, C_K\}$ according to their overall mobile traffic demand and accordingly there are $\frac{K(K+1)}{2}$ kinds of edges. For each kind of edges, we use $E_{ij} = \{(u, v)|u \in C_i, v \in C_j\}$ to denote the set of the edges connecting a node $u \in C_i$ and a node $v \in C_j$. For the nodes in each class $C_i$, there is a corresponding nodeRNN $R_{C_i}$ to handle node information. Likewise, for each kind of edges $E_{ij}$, an edgeRNN $R_{E_{ij}}$ is used for extracting edge information (i.e., neighbor information) so as to capture spatial dependency. The architecture of edgeRNN and nodeRNN is shown in Fig. 6, which will be described in detail next.

EdgeRNN is designed to characterize the importance of different neighbors in one class. Specifically, for a target node $v \in C_i$, the temporal sequences of its neighbors in class $C_j$, denoted as $TS_{v_j} = \{\mathcal{Y}_u^{t-P+1:t} | u \in NB(v) \cap C_j\}$ are input to edgeRNN $R_{E_{ij}}$ and integrated via the following attention mechanism. Since distinct neighbors have different influence on the target node, we first propose a soft attention module to explore different degrees of importance of neighbors. Specifically, for node $v$ in the spatial relation graph, the attention coefficient $\alpha_{vu}^t$ of its neighbor $u$ is calculated as follows:

$$e_{vu}^t = v_a^{\mathsf{T}} \tanh(\boldsymbol{W_a}[\mathcal{Y}_v^{t-P+1:t}; \mathcal{Y}_u^{t-P+1:t}]), \tag{4}$$

$$\alpha_{vu}^t = \frac{\exp(e_{vu}^t)}{\sum_{k \in NB(v) \cap C_j} \exp(e_{vk}^t)}, \tag{5}$$

where $v_a^{\mathsf{T}}$ and $\boldsymbol{W_a}$ are trainable parameters and $\tanh(\cdot)$ is tanh activation function which represents the importance of neighboring node $u$ in class $C_j$ for target node $v$.

According to Eqs. (4) and (5), each neighboring node $u$ is assigned an attention weight $\alpha_{vu}^t$ and then the spatial dependency of $v$ on neighboring nodes of class $C_j$ is represented as

$$Y_{v_j}^{t-P+1:t} = \sum_{u \in NB(v) \cap C_j} \alpha_{vu}^t \cdot \mathcal{Y}_u^{t-P+1:t}, \tag{6}$$

which is fed to the time series module in $R_{E_{ij}}$. We utilize LSTM as the time series module to capture the useful information of neighbors and get the hidden representation as $Y'^{t-P+1:t}_{v_j} \in \mathbb{R}^{P \times H}$, where $H$ is the dimension of hidden representation of LSTM. As the neighbors of a node are dispersed in different categories, the information of different kinds of neighbors will be assigned to different edgeRNNs. Then we get outputs from edgeRNNs $R_{E_{ik}}$, $k = 1, \ldots, K$, which are associated with the target node $v \in C_i$.

It's worth noting that the number of neighbors in different categories may vary from node to node, which implies that the input length of the edgeRNN are not fixed. To deal with this problem, we first figure out the maximum number of neighbors an edgeRNN should accommodate at one time and use it as the fixed input size of edgeRNN. Then, if the number of neighbors fed to the edgeRNN is less than the specified input size, zero vectors are used to fill the empty part. Besides, to effectively characterize the importance of actual neighbors and avoid the negative impact of padding values, we adopt a mask mechanism to filter the corresponding values when calculating the attention coefficients.

As described above, edgeRNN is used to extract the neighbor information in a specified class, while nodeRNN is designed to integrate the useful neighbor information in different classes captured by all the edgeRNNs with the temporal features of the target node itself for mobile traffic prediction. Another attention module, which is a component of nodeRNN, is suggested to solve the problem that different kinds of edges exert different effects for the target node. We derive the attention weight $\beta_{v_j}^t$ of edgeRNN $R_{E_{ij}}$ for the target node $v \in C_i$ as:

$$s_{v_j}^t = v_b^\mathsf{T} \tanh(\boldsymbol{W_b}[\mathcal{Y}_v^{t-P+1:t}; Y'^{t-P+1:t}_{v_j}]), \tag{7}$$

$$\beta_{v_j}^t = \frac{\exp(s_{v_j}^t)}{\sum_{k=1}^{K} \exp(s_{v_k}^t)}, \tag{8}$$

where $v_b^\mathsf{T}$ and $\boldsymbol{W_b}$ are parameters to be learned. The coefficient $\beta_{v_j}^t$ indicates the significance of the information captured by edgeRNN $R_{E_{ij}}$ for $v \in C_i$, and the output of attention module is computed by weighted summation

$$X_v^{t-P+1:t} = \sum_{j=1}^{K} \beta_{v_j}^t \cdot Y'^{t-P+1:t}_{v_j}. \tag{9}$$

The output $X_v^{t-P+1:t} \in \mathbb{R}^{P\times H}$ and the target node $v$'s historical time series $\mathcal{Y}_v^{t-P+1:t} \in \mathbb{R}^{P\times 1}$ are concatenated and then fed into the time series module LSTM in $R_{C_i}$. The output $h_v^t \in \mathbb{R}^H$ for node $v$ at time slot $t$ can be formulated as

$$h_v^t = \mathrm{LSTM}([\mathcal{Y}_v^{t-P+1:t}; X_v^{t-P+1:t}]). \tag{10}$$

In the end, $h_v^t$ is fed into the fully connected layer to predict the mobile traffic demand $\hat{y}_v^{t+1}$ of node $v$ at time slot $t+1$.

The combination of nodeRNN and edgeRNN in our attention-based structural RNN approach considers both self temporal dependency and complicated spatial impact simultaneously, and the attention mechanisms provide a fine-grained importance quantification of different neighbors for a target region, which together enable a precise mobile traffic prediction. Since our proposed prediction model is a data-driven approach and does not make any assumption on the mobile traffic demands of the target city, our method is generalizable for implementation at different cities, by training specific prediction models based on the given data traces of different target cities.

## 4 Collaborative Global-Local Learning

Our proposed GASTN model is able to train a global mobile traffic prediction model with the datasets of the citywide mobile traffic sequences for all the regions in a city. Such a global model can well capture the common knowledge of traffic patterns among all the regions in the city and hence process the nice merit in terms of model generalization. Nevertheless, it is prone to failures when performing on a particular region. To further improve the prediction performance, we would like to account for the specific characteristics of individual regions, which is called personalization in machine learning [12]. Since a local model trained with the local data is targeted at learning personal information for one region, it can better capture the personalized features of the target region so as to conduct personalization [36], [37]. However, the local data of one region is always limited to train a satisfactory prediction model. Thus, we further propose a collaborative global-local learning approach for mobile traffic forecasting, which considers both generalization ability and personalization ability of the model simultaneously and balances the trade-off between generalization and personalization of the prediction model, so as to further promote the performance of citywide mobile traffic prediction. In this section, we elaborate on the whole training process of collaborative learning between the global model and all local models.

### 4.1 Overview of Workflow for Collaborative Learning

Although the GASTN model trained with the global dataset (i.e., the data of all regions) can well capture common characteristics among regions and possess great generalization, it fails to learn personalized traffic features tailored to a specific region since there is distribution difference between the local data and the global data. Thus, directly deploying the trained global model would limit the prediction performance of citywide mobile traffic prediction. On the other hand, building a local GASTN model for each region with its limited local data will lead to an even worse model as it cannot reap the benefits of rich traffic knowledge from other regions in the city. In a word, such methods would be detrimental to prediction performance to varying degree.

To address these issues, we propose an innovative collaborative global-local learning scheme in this paper. As shown in Fig. 7, the learning scheme is operated between the citywide global model and local models of regions in the city. Specifically, the collaborative learning process mainly consists of the following four stages: (1) train a model based on aforementioned GASTN framework using the global dataset of mobile traffic to attain a well-initialized global model for the follow-up operations, (2) build a personalized local model for each region based on the pre-trained global model via transfer learning and then refine the local model for each region with its own data, (3) fine-tune the global model with the help of knowledge distillation from local models, and (4) obtain a fine-tuned global model for citywide mobile traffic prediction by iterating the operations of stage (2) and (3) until it converges. It is worth mentioning that the iteration between the second and third stages is to achieve better performance and improve the robustness of the model. These two stages are the core parts of our approach which will be elaborated in detail next.

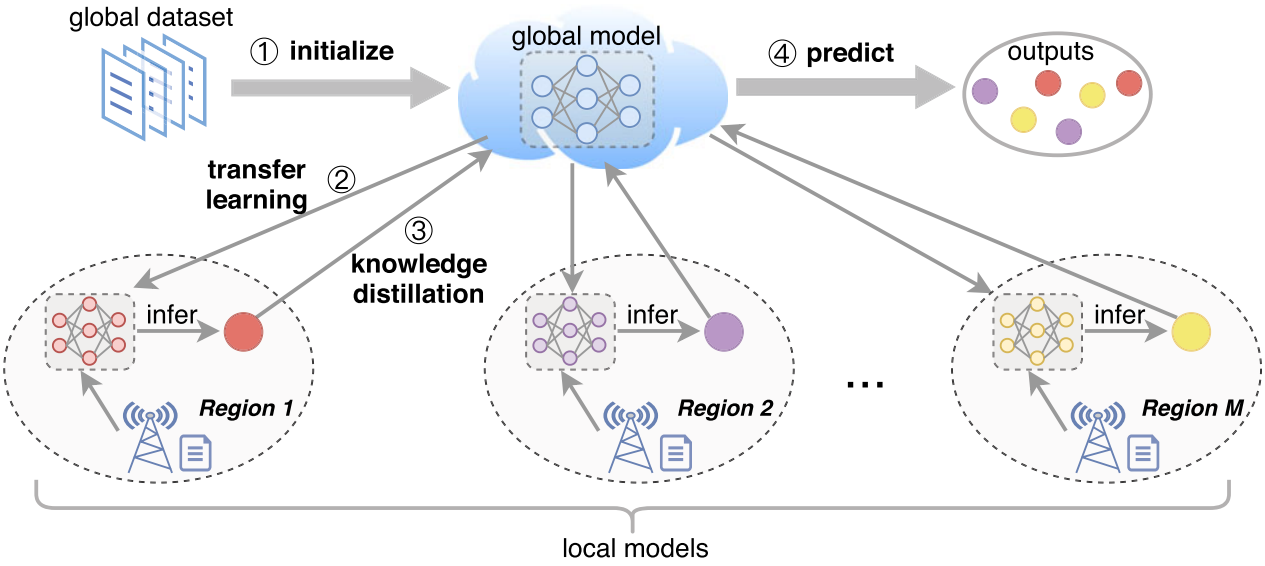



Fig. 7. The training process of our collaborative global-local learning scheme, which mainly consists of four stages operating between the global model and local models.

### 4.2 Local Model Learning via Transfer Learning

To effectively capture the specific characteristics and achieve personalization for each region in the city, we advocate a local model learning stage in the collaborative learning process. However, it would face high resource demand (e.g., energy, computation resources) when directly training a local GASTN model from scratch with its local data (traffic sequence of the target region and a set of its neighbors' traffic sequences) for each region. Moreover, insufficient local data samples and local data shifts will lead to poor performance of the trained local model.

Since the trained global model reaps rich information from all the regions in a city, we propose to utilize the global model as a model initialization for the local model of each region and then perform local model adaptation for each region with its local data samples. We first transfer the useful knowledge from the global model to the local models by adopting transfer learning technique [38]. Specifically, for each region, we select the related learning structures in the global GASTN model (i.e., the nodeRNN of the target region and edgeRNNs with its neighboring regions in GASTN) as its local model architecture. Then, the corresponding parameters of the trained global model, treated as the source knowledge, are transfered to the local GASTN model of the target region for model initialization. To capture the fine-grained information of the target region, the local model is fine-tuned with its local data samples based on the transferred model parameters. As a consequence, local model learning via transfer learning can achieve fast adaptation and effective personalization to the target region comparing with training from scratch.

### 4.3 Global Model Fine-Tuning via Knowledge Distillation

After the local model learning at the second stage, each region in the city is equipped with a personalized local model for accurate mobile traffic forecasting. However, it is inconvenient and impractical to manage a local prediction model for each region, especially when there are a large number of regions in the city. Considering the fact that the global model is able to improve the holistic prediction performance for a city by parameter alignment, we intend to employ the global model for our final prediction. In order to constitute a high-precision global model for all regions and take full advantage of the personalized local models, we propose to fine-tune the original global GASTN model with local models via knowledge distillation, which is also a technique for knowledge transfer [39].

The process of global model fine-tuning at the third stage is shown in Fig. 8. First of all, the trained local models are separately used to make predictions for their own training data and then the inference (e.g., the predicted values of mobile traffic at the next time slot) for all the regions can be obtained, which is regarded as the distilled knowledge. Such knowledge includes regional personalized information distilled from its local data by the local model of each region, and it is considered to be useful for improving the personalization ability of our global model.

Thus, we adopt knowledge distillation to retrain the global model under the support of knowledge distilled from

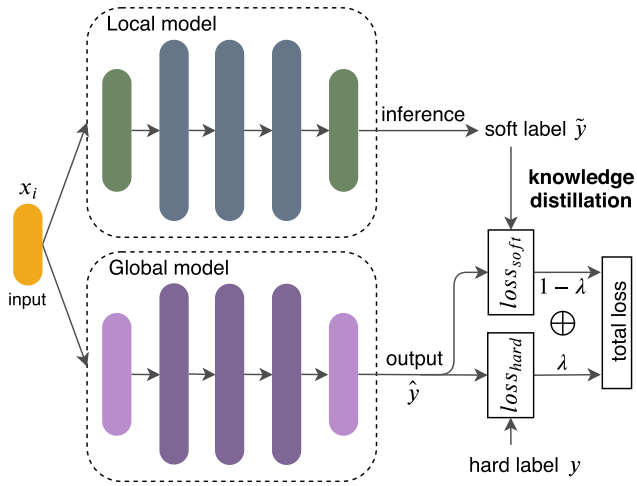



Fig. 8. The process of global model fine-tuning via knowledge distillation.

local models of all the regions in the city. Generally, the ground truth of a sample is constant and given beforehand, which is thought to be a hard target for training. Hence, we treat the ground truth as a *hard label* of the sample. On the other hand, an inferred result is generated by the local model inference, so the inferred result is regarded as a soft label to assist the training of our global model. In this way, our global model can not only learn from the ground truths, but also benefit from the personalized information extracted from local models, thereby further improving the model performance. Note that the terms of hard and soft labels are commonly used in the literature of knowledge distillation [39].

In the knowledge distillation process, the global model is fine-tuned based on its latest parameters and its objective is to match each output $\hat{y}$ to both the corresponding *hard label* $y$ as well as the *soft label* $\tilde{y}$ during training. Therefore, the training loss of the global model consists of two parts: the original loss and the distillation loss, which can be formulated as

$$Loss = \lambda L(y, \hat{y}) + (1 - \lambda) L(\tilde{y}, \hat{y}), \tag{11}$$

where $L(\cdot)$ can be a common loss function for regression problems, such as root mean squared error, mean absolute error, etc. $\lambda$ is a hyperparameter that controls the impact of knowledge distillation on the whole model. With the operation of knowledge distillation, the original global model can learn personalized mobile traffic information of various regions from local models, thus enhancing its ability to extract personalized features.

In summary, the whole collaborative learning between the global model and local models is achieved through different forms of knowledge transfer techniques so that these models can learn fully and promote mutually. As a result, our global model has the power to realize generalization as well personalization in equilibrium so as to boost the performance of mobile traffic prediction for the city compared to a single trained global model.

## 5 Experiments

### 5.1 Data Description

In this paper, we evaluate our proposed method on a large-scale real-world telecommunications dataset, which is provided by Telecom Italia and publicly available [40]. The dataset contains records of mobile traffic volume observed over 10-minute intervals for the city of Milan where the geographical area is partitioned into $100 \times 100$ regions and the size of each region is $235m \times 235m$. Each record in the dataset includes region ID, the beginning of the time interval and the mobile traffic volume during the time interval. In the experiment, we choose data from 01/11/2013 to 20/11/2013 (20 days) as training data and the rest 10 days as test data. Actually, the Milan dataset includes the mobile traffic data in November and December, 2013. Since our data sample is based on the interval of 10-minutes, we can generate a large amount of samples by a sliding window only on a month's data, which is sufficient to well train our proposed model and achieve outstanding performance. Due to the computing resource limitation of our research lab, we only use the data for one month (i.e., 1st Nov. 2013 to 30th Nov. 2013) in our experiments.

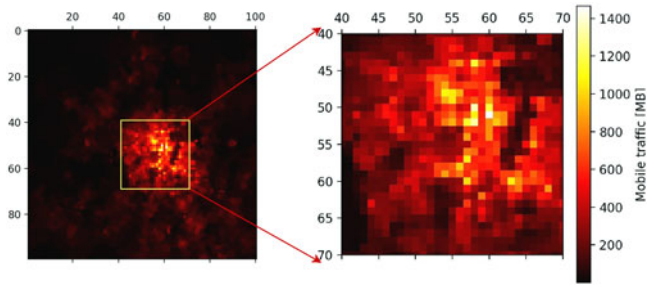



Fig. 9. Milan mobile traffic heat map and the coverage area with $30 \times 30$ grids is selected to be studied.

### 5.2 Experimental Settings

*Preprocessing.* Fig. 9 shows a heat map of the total cellular traffic volume in Milan during November 2013. It indicates that the central part of Milan generally maintains a great demand for mobile traffic while the traffic volume is relatively small in other regions. Since people don't care about the low-demand situation in practical applications, we focus on the regions ($30 \times 30$ grids) with great demand to evaluate our proposed method. For the traffic data, we scale them to $[0, 1]$ by Min-Max Normalization before prediction and anti-normalize the predicted values for evaluation after prediction. Samples for training and testing are generated by a sliding window on the data.

*Implementation Details.* We set the length of input sequence $P = 6$ (i.e., previous 1 hour) and apply our model to predict the traffic volume in the next time step. The threshold of the size of each neighborhood is set as $N = 15$ and the number of classes for nodes is set to $K = 3$. In our model, we utilize LSTMs as the recurrent neural networks in edgeRNN and nodeRNN, and the number of hidden units of each LSTM is 64. For our collaborative global-local learning scheme, the number of iterations and the coefficient of loss function are respectively set as $E = 3$ and $\lambda = 0.9$ through the experimental validation. Besides, all local models have the same structure as the global model based on GASTN framework.

We should emphasize that the training of the prediction model is the most time-consuming part, and fortunately the training is delay insensitive and can be done in an offline manner by collecting the historical data traces and computing in the cloud datacenter. After training the model, we can then deploy the model inference for real-time prediction, which is much lightweight and can be done very fast. To further address latency issue caused by data transmission and post-processing, big data processing framework for mobile cellular networks such as [41] can be adopted. Moreover, to reduce the volume of data transmission, we can leverage the edge computing at each base station to implement the NodeRNN module locally and in this case only model parameters are transmitted instead of the massive raw traffic data.

On the other hand, in our study we use the cellular traffic volume observed over 10-minute intervals for mobile traffic prediction as in [5] and [11]. It's worthnoting that our model allows the time interval length can be specified by the telecom operators according to their requirements. For example, by simply aggregating data from three adjacent 10-minute intervals, our model can conduct half-hour mobile traffic forecasting.

TABLE 1
Comparison With Different Baselines

| Method | NRMSE | MAPE | MAE |
|---|---|---|---|
| HA | 0.4338 | 0.4955 | 69.9062 |
| ARIMA | 0.1817 | 0.3783 | 32.6042 |
| MLP | 0.2089 | 0.2442 | 37.9893 |
| LSTM | 0.1771 | **0.2064** | 32.6483 |
| CNN-LSTM | 0.2089 | 0.2313 | 37.7972 |
| STN | 0.1750 | 0.3114 | 33.0419 |
| **GASTN** | **0.1705** | 0.2143 | **30.9307** |

*Baselines.* We compare our GASTN with the following widely used methods:

- *Historical average (HA)*: HA predicts the value using the average of previous mobile traffic demand of the given region in the same relative time interval (i.e., the same time of the day) [42].
- *ARIMA*: ARIMA is a well-known time series model and widely used for mobile traffic prediction [15].
- *Multiple layer perceptron (MLP)*: MLP is a neural network with three fully connected layers that the number of hidden units are 64, 128 and 64, respectively.
- *LSTM*: LSTM is able to capture short-term and long-term temporal dependency and it has been widely used in time series forecasting problems including mobile traffic prediction [21], [43].
- *CNN-LSTM*: CNN and LSTM are integrated into a model to capture spatial dependency and temporal relationship in a holistic manner [10].
- *STN*: STN is a deep spatio-temporal network that combines ConvLSTM and 3D-ConvNet to capture temporal and spatial relationships simultaneously. It obtains the state-of-the-art result in mobile traffic forecasting [11].

*Evaluation Metrics.* We evaluate our model with three commonly used metrics: Normalized Root Mean Square Error (NRMSE) [11], Mean Absolute Percentage Error (MAPE) and Mean Absolute Error (MAE) [10], which are defined as follows:

$$NRMSE = \frac{1}{\bar{y}}\sqrt{\frac{1}{z}\sum_{i=1}^{z}(\hat{y}_i - y_i)^2}, \tag{12}$$

$$MAPE = \frac{1}{z}\sum_{i=1}^{z}\left|\frac{\hat{y}_i - y_i}{y_i}\right|, \tag{13}$$

$$MAE = \frac{1}{z}\sum_{i=1}^{z}|\hat{y}_i - y_i|, \tag{14}$$

where $\hat{y}_i$ and $y_i$ are the predicted value and ground truth of mobile traffic demand for each region, $z$ is the total number of regions in the city, and $\bar{y}$ is the mean of all ground truth values. Thus, each experimental result is calculated by the average of prediction errors of all the test samples.

### 5.3 Prediction Results of GASTN

*Performance Comparison.* As a fair comparison, similar to the baselines, our GASTN model is trained with the global dataset directly without the collaborative global-local learning strategy. The evaluation of the effectiveness of collaborative global-local learning strategy is provided later on. As shown in Table 1, GASTN generally outperforms all the baselines, achieving the lowest NRMSE (0.1705) and the lowest MAE (30.9307). Specifically, the traditional time series models (HA and ARIMA) don't perform well. For deep learning methods, with the ability of capturing temporal dependency, LSTM performs better than MLP. Although CNN-LSTM and STN consider the relations of both space and time, their performance is worse than GASTN. This is because that these methods capture spatial relation based on the grid structure and regard all nearby areas of a region as its neighbors, which may introduce noise from some actually irrelevant neighbors. With the designed spatial relation graph, GASTN captures the spatial relationship of mobile traffic demands in different regions and achieves the best performance compared with all state-of-the-art baselines. The experimental results imply the essentiality of selecting neighbors by constructing the spatial relation graph and demonstrate the effectiveness of our proposed GASTN model.

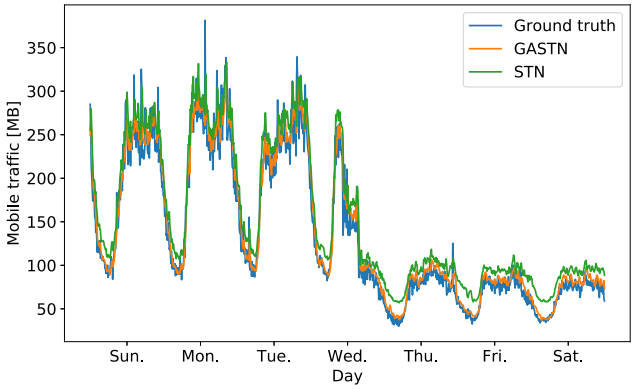



Fig. 10. The ground truth and predicted values of mobile traffic volume in the last week of November for region 6861.

In Fig. 10, we plot the ground truth and predicted values for mobile traffic in the last week of November for a given region and the predicted values are generated by our GASTN model and the up-to-date method STN respectively. It illustrates that the curve generated by GASTN model fits the real traffic curve better while STN is unable to grasp the dynamics well especially when traffic demand declines. As for the computation complexity, Fig. 11 shows that the training time of GASTN is always less than STN for each epoch under the same experimental equipment. Moreover, as the number of samples increases, the gap between them becomes more and more obvious. This observation implies that the effectiveness of GASTN depends on its reasonable architecture design rather than model complexity. In addition, we give a global view of the prediction performance comparison between GASTN and STN in Fig. 12, which indicates that the traffic distribution predicted by GASTN is much closer to the ground truth compared with STN.

*Parameter Analysis.* In order to figure out the impact of the number of neighbors on the model performance, we analyze the setting of parameter $K$ as illustrated in Table 2. We can see that when $K$ has a small value (e.g., $K = 2, 5$), the prediction errors in terms of NRMSE and MAE are relatively large. On the other hand, when we set the number of neighbors as a large value (e.g., $K = 50, 100$), there is no significant

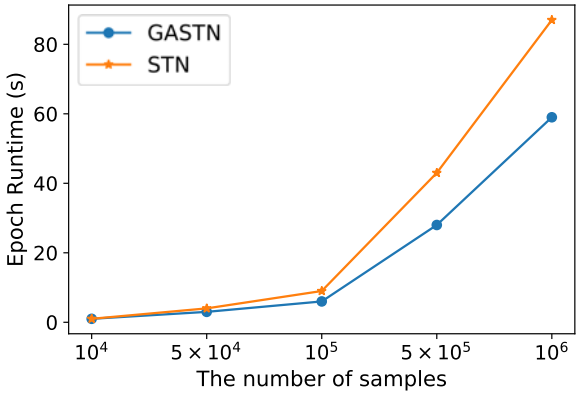
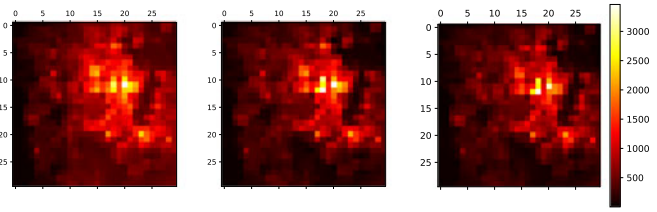



Fig. 11. The training time of GASTN and STN for each epoch under different sample sizes.



(a) STN (b) GASTN (c) Ground truth

Fig. 12. An example of the predicted mobile traffic distributions using STN and GASTN respectively and the corresponding ground truth.

TABLE 2
Comparison With Different Values of Parameter $K$

| K | NRMSE | MAPE | MAE |
|---|---|---|---|
| 2 | 0.1731 | 0.2019 | 31.1379 |
| 5 | 0.1712 | 0.1970 | 30.9496 |
| 10 | 0.1709 | 0.2230 | 30.9979 |
| 15 | 0.1705 | 0.2143 | 30.9307 |
| 20 | 0.1705 | 0.2278 | 30.7733 |
| 25 | 0.1705 | 0.2172 | 30.8969 |
| 50 | 0.1703 | 0.2393 | 30.4550 |
| 100 | 0.1704 | 0.2109 | 30.5060 |

improvement in the model performance while increasing the training burden greatly with more neighboring nodes. That is, it is not a good choice of setting the value of $K$ too small or too large. Thus, we select a relatively reasonable value for $K$ (i.e., $K = 15$) such that it is able to obtain a good prediction performance of GASTN as well as reduce the computation time as much as possible.

*Effect of Attention Mechanisms.* To analyze the effectiveness of the attention mechanisms in GASTN, several variants are designed as follows:

- *GASTN-AT1*: GASTN-AT1 removes the first attention mechanism in GASTN that just averages the temporal sequences of neighbors in edgeRNN.
- *GASTN-AT2*: instead of using the second attention mechanism, we simply concatenate the outputs of edgeRNNs and then feed them into nodeRNN.
- *GASTN-ATs*: GASTN-ATs removes the two attention mechanisms in the origion GASTN model.

TABLE 3
Comparison With Variants of GASTN

| Method | NRMSE | MAPE | MAE |
|---|---|---|---|
| GASTN-AT1 | 0.1729 | 0.2215 | 32.2622 |
| GASTN-AT2 | 0.1730 | 0.2268 | 32.5218 |
| GASTN-ATs | 0.1734 | 0.2301 | 32.5469 |
| **GASTN** | **0.1705** | **0.2143** | **30.9307** |

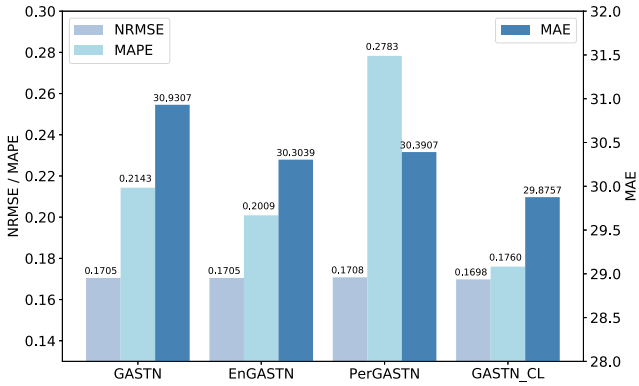



Fig. 13. The performance of different learning strategies.

The results of these variants are listed in Table 3. It reveals that GASTN outperforms its variants because GASTN-AT1 overlooks the different effects of different neighbors for a target node and GASTN-AT2 ignores the discrepancy of the importance of distinct edgeRNNs. GASTN-ATs performs worst among these variants since it pays no attention to the differences in neighbors or edge types. The results manifest the effectiveness of the attention mechanisms in GASTN to collectively capture the impact of spatial dependency on mobile traffic prediction.

### 5.4 Performance of Collaborative Global-Local Learning

The well-performed GASTN obtained above is trained by the whole dataset, so we treat it as the initialization of our global model for collaborative global-local learning and then evaluate the GASTN model equipped with our learning strategy, which is named GASTN_CL. To further compare with GASTN_CL, we build two new baselines as follows. One is EnGASTN that first selects the regions with prediction errors in the top 20 percent generated by the initial global model and then fine-tunes the global model with the data set of those regions. The other is PerGASTN, which adopts collaborative global-local learning for training but makes predictions by the local models. Fig. 13 reveals the experimental results of our proposed method and baselines. As we can see, GASTN_CL has the best performance among all the state-of-the-art baselines, achieving the lowest prediction error in all evaluation metrics. Especially, GASTN_CL achieves a performance gain of 18 percent over GASTN in terms of MAPE. Compared with GASTN, the superior performance of GASTN_CL demonstrates the effectiveness of our collaborative global-local learning scheme which makes full use of the potential of local models. While EnGASTN is useful to enhance the performance of GASTN by fine-tuning GASTN with high-error regions. it still performs worse than

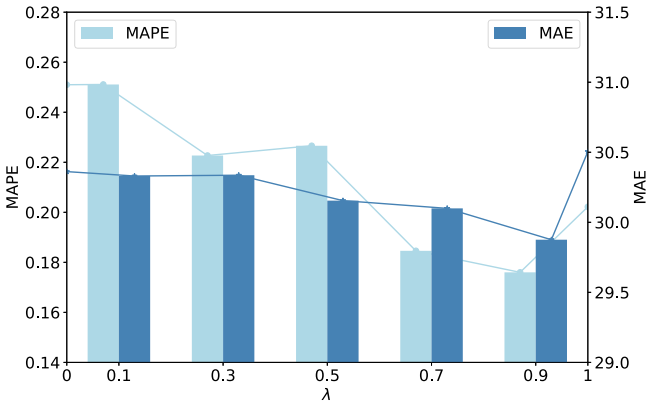



Fig. 14. The prediction performance of GASTN_CL under different parameter settings of $\lambda$

GASTN_CL. This is because that EnGASTN cannot capture personalized features of each region for further learning, which illustrates the significance of collaborative learning. Moreover, though collaborative learning scheme is applied for PerGASTN, each local model has limited prediction ability due to the limited data of a single region, leading to performance degradation. Instead, the global model of GASTN_CL trained with a large amount of data possesses rich knowledge so as to conduct a precise mobile traffic prediction. Therefore, it indicates the essentiality of taking the well-trained global model as our ultimate goal to make precise citywide prediction. Owing to our delicate design, GASTN_CL outperforms all the baselines as demonstrated by the extensive experimental results.

Furthermore, we investigate the influence of hyperparameter $\lambda$ in knowledge distillation on the performance of collaborative learning. Since the changing trends of the three evaluation metrics are similar, we select two of them for further study. As shown in Fig. 14, the prediction error generally presents a declining trend in terms of both MAPE and MAE as the value of $\lambda$ increases. If $\lambda$ is too small, the global model will rely too much on the knowledge distilled from local models while neglecting the ground truths. In this way, once the local models do not work well, the global model will also be negatively affected and the predicted values will deviate more and more from the ground truths after collaborative learning. Especially when $\lambda = 0$, the training of our global model depends entirely on the inferred results of local models without considering the effect of ground truths, so the prediction error of the global model will be superimposed on the error of the local models, leading to the worse results. Hence, it is critical to set a relatively high value for $\lambda$ so that the global model can avoid deviation by grasping global knowledge while leveraging useful local knowledge. However, when $\lambda$ is set as 1, GASTN_CL will degenerate into the original GASTN with just more training epoches. In this case, the prediction performance is degraded compared to the case of $\lambda = 0.9$, which verifies the rationality of our parameter setting for $\lambda$. In practice, depending on how diversified the traffic patterns of the regions in a city are, a reasonable value is able to make a trade-off between the global and local knowledge so as to perform superior prediction performance for a city.

## 6 Conclusion

In this paper, we study the mobile traffic prediction, which will contribute to mobile network service in a smart city. We propose a novel Graph Attention Spatial-Temporal Network (GASTN) that integrates spatial and temporal patterns of mobile traffic together to extract important information for prediction. The near-far spatial correlation is captured by the spatial relation graph and the spatial-temporal factors are modeled by attention-based structural RNN. Besides, we put forward a collaborative global-local learning scheme to further enhance the performance of our prediction model. We evaluate our method on a large-scale mobile traffic dataset and the experimental results demonstrate that our proposed model GASTN can outperform the state-of-the-art baselines with faster running time. In addtion, the outstanding forecasting results of GASTN_CL further illustrate the effectiveness of our designed collaborative global-local learning strategy. For the future work, we will explore the graph convolutional network approach for mobile traffic prediction based on our spatial relation graph construction.

## Acknowledgments

This work was supported in part by the National Science Foundation of China (No. U1711265 and No. 61972432), the Program for Guangdong Introducing Innovative and Entrepreneurial Teams (No. 2017ZT07X355), and the Pearl River Talent Recruitment Program (No. 2017GC010465). Part of results in this submission has been presented in 2019 IEEE GLOBECOM [1].

## References

[1] K. He, Y. Huang, X. Chen, Z. Zhou, and S. Yu, "Graph attention spatial-temporal network for deep learning based mobile traffic prediction," in *Proc. IEEE Global Commun. Conf.*, 2019, pp. 1–6.
[2] K. Su, J. Li, and H. Fu, "Smart city and the applications," in *Proc. Int. Conf. Electron. Commun. Control*, 2011, pp. 1028–1031.
[3] A. Nitu, C. Stirbu, and F. M. Enescu, "Mobile application for a smart city," in *Proc. 10th Int. Conf. Electron. Comput. Artif. Intell.*, 2018, pp. 1–4.
[4] Cisco Systems Inc., "Cisco Visual Networking Index: Forecast and Trends, 2017-2022," [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white-paper-c11-741490.html
[5] F. Xu *et al.*, "Big Data driven mobile traffic understanding and forecasting: A time series approach," *IEEE Trans. Services Comput.*, vol. 9, no. 5, pp. 796–805, Sep./Oct. 2016.
[6] A. Furno, M. Fiore, and R. Stanica, "Joint spatial and temporal classification of mobile traffic demands," in *Proc. IEEE Conf. Comput. Commun.*, 2017, pp. 1–9.
[7] R. Li *et al.*, "Intelligent 5G: When cellular networks meet artificial intelligence," *IEEE Wireless Commun.*, vol. 24, no. 5, pp. 175–183, Oct. 2017.
[8] I. Donevski, G. Vallero, and M. A. Marsan, "Neural networks for cellular base station switching," in *Proc. IEEE Conf. Comput. Commun. Workshops*, 2019, pp. 738–743.
[9] H. W. Kim, J. H. Lee, Y. H. Choi, Y. U. Chung, and H. Lee, "Dynamic bandwidth provisioning using arima-based traffic forecasting for mobile wimax," *Comput. Commun.*, vol. 34, no. 1, pp. 99–106, 2011.
[10] C. W. Huang, C. T. Chiang, and Q. Li, "A study of deep learning networks on mobile traffic forecasting," in *Proc. IEEE Int. Symp. Pers.*, 2018, pp. 1–6.
[11] C. Zhang and P. Patras, "Long-term mobile traffic forecasting using deep spatio-temporal neural networks," in *Proc. 19th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2018, pp. 231–240.
[12] G. Adomavicius and A. Tuzhilin, "Personalization technologies: a process-oriented perspective," *Commun. ACM*, vol. 48, no. 10, pp. 83–90, 2005.

[13] S. Kim, Y. Chon, S. Lee, and H. Cha, "Prediction-based personalized offloading of cellular traffic through WiFi networks," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun.*, 2016, pp. 1–9.
[14] A. Adas, "Traffic models in broadband networks," *Commun. Magazine*, vol. 35, no. 7, pp. 82–89, Jul. 1997.
[15] Y. Shu, M. Yu, J. Liu, and O. W. W. Yang, "Wireless traffic modeling and prediction using seasonal arima models," in *Proc. IEEE Int. Conf. Trans. Commun.*, 2003, pp. 1675–1679.
[16] J. Liu and Y. L. Huang, "Nonlinear network traffic prediction based on bp neural network," *J. Comput. Appl.*, vol. 27, no. 7, pp. 1770–1772, 2007.
[17] J. Zhang, Y. Zheng, and D. Qi, "Deep spatio-temporal residual networks for citywide crowd flows prediction," in *Proc. AAAI Conf. Artif. Intell.*, 2017, pp. 1655–1661.
[18] T. Mikolov, M. Karafiát, L. Burget, J. Černockỳ, and S. Khudanpur, "Recurrent neural network based language model," in *Proc. Interspeech*, 2010, pp. 1045–1048.
[19] S. Kombrink, T. Mikolov, M. Karafiát, and L. Burget, "Recurrent neural network based language modeling in meeting recognition," in *Proc. Interspeech*, 2011, pp. 2877–2880.
[20] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[21] H. D. Trinh, L. Giupponi, and P. Dini, "Mobile traffic prediction from raw data using LSTM networks," in *Proc. 29th IEEE Annu. Int. Symp. Pers. Indoor Mobile Radio Commun.*, 2018, pp. 1827–1832.
[22] X. Shi, Z. Chen, H. Wang, D. Yeung, W. Wong, and W. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, 2015, pp. 802–810.
[23] S. Ji, W. Xu, M. Yang, and K. Yu, "3D convolutional neural networks for human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 221–231, Jan. 2013.
[24] L. Fang, X. Cheng, H. Wang, and L. Yang, "Mobile demand forecasting via deep graph-sequence spatiotemporal modeling in cellular networks," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 3091–3101, Aug. 2018.
[25] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, 2018, pp. 3634–3640.
[26] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," in *Proc. Int. Conf. Learn. Representations*, 2018, pp. 1–16.
[27] Z. Cui, K. Henrickson, R. Ke, and Y. Wang, "Traffic graph convolutional recurrent neural network: A deep learning framework for network-scale traffic learning and forecasting," *IEEE Trans. Intell. Transp. Syst.*, Nov. 2019, doi: 10.1109/TITS.2019.2950416.
[28] H. Daga, P. K. Nicholson, A. Gavrilovska, and D. Lugones, "Cartel: A system for collaborative transfer learning at the edge," in *Proc. ACM Symp. Cloud Comput.*, 2019, pp. 25–37.
[29] X. Zhang, M. Qiao, L. Liu, Y. Xu, and W. Shi, "Collaborative cloud-edge computation for personalized driving behavior modeling," in *Proc. 4th ACM/IEEE Symp. Edge Comput.*, 2019, pp. 209–221.
[30] Y. Lu *et al.*, "Collaborative learning between cloud and end devices: an empirical study on location prediction," in *Proc. 4th ACM/IEEE Symp. Edge Comput.*, 2019, pp. 139–151.
[31] H. Ding, G. Trajcevski, P. Scheuermann, X. Wang, and E. Keogh, "Querying and mining of time series data: experimental comparison of representations and distance measures," *Proc. VLDB Endowment*, vol. 1, no. 2, pp. 1542–1552, 2008.
[32] E. J. Keogh and M. J. Pazzani, "Derivative dynamic time warping," in *Proc. 1st SIAM Int. Conf. Data Mining*, 2001, pp. 1–11.
[33] M. K. Brown and L. R. Rabiner, "Dynamic time warping for isolated word recognition based on ordered graph searching techniques," in *Proc. IEEE Int. Conf. Acoustics Speech Signal Process.*, 1982, pp. 1255–1258. [Online]. Available: https://doi.org/10.1109/ICASSP.1982.1171695
[34] Y. Yuan and M. Raubal, "Extracting dynamic urban mobility patterns from mobile phone data," in *Proc. 7th Int. Conf. Geographic Inf. Sci.*, 2012, pp. 354–367. [Online]. Available: https://doi.org/10.1007/978-3-642-33024-7_26
[35] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena, "Structural-RNN: Deep learning on spatio-temporal graphs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 5308–5317.
[36] Y. Chen, X. Qin, J. Wang, C. Yu, and W. Gao, "Fedhealth: A federated transfer learning framework for wearable healthcare," *IEEE Intell. Syst.*, vol. 35, no. 4, pp. 83–93, Jul./Aug. 2020.
[37] M. G. Arivazhagan, V. Aggarwal, A. K. Singh, and S. Choudhary, "Federated learning with personalization layers," *CoRR*, vol. abs/1912.00818, 2019. [Online]. Available: http://arxiv.org/abs/1912.00818
[38] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.
[39] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *Comput. Sci.*, vol. 14, no. 7, pp. 38–39, 2015.
[40] G. Barlacchi *et al.*, "A multi-source dataset of urban life in the city of milan and the province of trentino," *Sci. Data*, vol. 2, 2015, Art. no. 150055.
[41] S. Brdar *et al.*, "Big data processing, analysis and applications in mobile cellular networks," in *High-Performance Modelling and Simulation for Big Data Applications*, Cham: Springer, 2019, pp. 163–185.
[42] H. Yao *et al.*, "Deep multi-view spatial-temporal network for taxi demand prediction," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 2588–2595.
[43] R. Vinayakumar, K. Soman, and P. Poornachandran, "Applying deep learning approaches for network traffic prediction," in *Proc. Int. Conf. Advances Comput. Commun. Inform.*, 2017, pp. 2353–2358.

**Kaiwen He** received the BS and MS degrees from the School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China, in 2018 and 2020 respectively. She is currently an algorithm engineer at the Interactive Entertainment Group, Alibaba Inc., Guangzhou, China. Her current research interests include data analysis, data mining, and natural language processing.

**Xu Chen** (Senior Member, IEEE) received the PhD degree in information engineering from the Chinese University of Hong Kong, in 2012. He is currently a full professor with Sun Yat-sen University, Guangzhou, China, and the vice director of National and Local Joint Engineering Laboratory of Digital Home Interactive Applications. He worked as a postdoctoral research associate with Arizona State University, Tempe, from 2012 to 2014, and a Humboldt scholar fellow with the Institute of Computer Science, University of Goettingen, Germany from 2014 to 2016. He received the prestigious Humboldt research fellowship awarded by the Alexander von Humboldt Foundation of Germany, 2014 Hong Kong Young Scientist Runner-up Award, 2017 IEEE Communication Society Asia-Pacific Outstanding Young Researcher Award, 2017 IEEE ComSoc Young Professional Best Paper Award, Honorable Mention Award of 2010 IEEE international conference on Intelligence and Security Informatics (ISI), Best Paper Runner-up Award of 2014 IEEE International Conference on Computer Communications (INFOCOM), and Best Paper Award of 2017 IEEE Intranational Conference on Communications (ICC). He is currently an Area Editor of the *IEEE Open Journal of the Communications Society*, an associate editor of the *IEEE Transactions Wireless Communications*, the *IEEE Internet of Things Journal* and the *IEEE Journal on Selected Areas in Communications* (JSAC) Series on Network Softwarization and Enablers.

**Qiong Wu** received the BS and ME degrees from the School of Data and Computer Science, Sun Yat-sen University (SYSU), Guangzhou, China, in 2017 and 2019, respectively. She is currently working toward the PhD degree from the School of Data and Computer Science, SYSU. Her primary research interests include social data analysis, mobile edge computing and federated learning.

**Shuai Yu** (Member, IEEE) received the BS degree from the Nanjing University of Post and Telecommunications (NJUPT), Nanjing, China, in 2009, the MS degree from the Beijing University of Post and Telecommunications (BUPT), Beijing, China, in 2014, and the PhD degree from University Pierre and Marie Curie (now Sorbonne Université), Paris, France, in 2018, He is currently a post-doctoral research fellow with the School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China. His research interests include wireless communications, mobile computing and machine learning.

**Zhi Zhou** (Member, IEEE) received the BS, ME and PhD degrees from the School of Computer Science and Technology, Huazhong University of Science and Technology (HUST), Wuhan, China, in 2012, 2014, and 2017, respectively. He is currently a research associate fellow at the School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China. In 2016, he has been a visiting scholar with University of Goettingen. His research interests include edge computing, cloud computing, and distributed systems.