Mobile Demand Forecasting via Deep Graph-Sequence Spatiotemporal Modeling in Cellular Networks

Luoyang Fang, *Student Member, IEEE*, Xiang Cheng, *Senior Member, IEEE*, Haonan Wang, and Liuqing Yang, *Fellow, IEEE*

***Abstract*—The demand forecasting plays a crucial role in the predictive physical and virtualized network management in cellular networks, which can effectively reduce both the capital and operational expenditures by fully exploiting the network infrastructure. In this paper, we study the per-cell demand forecasting in cellular networks. The success of demand forecasting relies on the effective modeling of both the spatial and temporal aspects of the per-cell demand time series. However, the main challenge of the spatial relevancy modeling in the per-cell demand forecasting is the irregular spatial distribution of cells in a network, where applying grid-based models (e.g., convolutional neural networks) would lead to degradation of spatial granularity. In this paper, we propose to model the spatial relevancy among cells by a dependency graph based on spatial distances among cells without the loss of spatial granularity. Such spatial distance-based graph modeling is confirmed by the spatiotemporal analysis via semivariogram, which suggests that the relevancy between any two cells declines as their spatial distance increases. Hence, the graph convolutional networks and long short-term memory (LSTM) from deep learning are employed to model the spatial and temporal aspects, respectively. In addition, the deep graph-sequence model, graph convolutional LSTM, is further employed to simultaneously characterize both the spatial and temporal aspects of mobile demand forecasting. Experiments demonstrate that our proposed graph-sequence demand forecasting model could achieve a superior forecasting performance compared with the other two proposed models as well as the traditional auto regression integrated moving average time series model.**

***Index Terms*—Communication system traffic, mobile learning, statistical learning.**

Manuscript received February 10, 2018; revised April 3, 2018; accepted April 19, 2018. Date of publication May 1, 2018; date of current version August 9, 2018. This work was supported in part by the National Natural Science Foundation of China under Grant 61622101 and Grant 61571020, in part by the National Science and Technology Major Project under Grant 2018ZX03001031, and in part by the National Science Foundation under Grant DMS-1521746 and Grant DMS-1737795. *(Corresponding author: Xiang Cheng.)*

L. Fang and L. Yang are with the Department of Electrical and Computer Engineering, Colorado State University, Fort Collins, CO 80523 USA (e-mail: luoyang.fang@colostate.edu; lqyang@engr.colostate.edu).

X. Cheng is with the State Key Laboratory of Advanced Optical Communication Systems and Networks, School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China (e-mail: xiangcheng@pku.edu.cn).

H. Wang is with the Department of Statistics, Colorado State University, Fort Collins, CO 80523 USA (e-mail: wanghn@stat.colostate.edu).

Digital Object Identifier 10.1109/JIOT.2018.2832071

I. Introduction

With the explosive growth of wireless data traffic and large-scale penetration of mobile devices into our everyday life, the massive data generated from mobile devices and mobile networks, termed as mobile big data [1], could significantly reveal the human activity patterns, which is valuable in both data-driven personalized applications and data-driven public services. In [2], one of the highlighted characteristics of mobile big data is its spatiotemporal feature. In fact, the cell towers or base stations of a mobile network spatially distributed in an area could be regarded as sensors, recording the location of the network subscribers without the proactive location update via GPS by subscribers.

The mobile big data collected by mobile network operators can also benefit the management of mobile networks. In fact, mobile big data could help uncover and understand user' behavior patterns [3] via effective data mining techniques, which could benefit to the resource-constraint network optimization, from network planning, network traffic monitoring to network management. In recent years, self-organizing networks (SONs) is widely studied to automatically manage and organize networks without manual intervention [4], [5]. One motivation to employ SONs in cellular networks is the reduction of network operational expenditures and capital expenditures, which requires full exploitation of the capability of network infrastructure. The demand forecasting will play an important role of providing predictive knowledge [6] in various cellular SON functions, especially for the future cellular networks with the virtualization and cloudization of network functions [7], [8].

In addition, the studied mobile demand forecasting is not only a critical problem in cellular networks, but also closely related to the domain of Internet of Things (IoT). First, each base station of cellular networks can be regarded as an IoT device to track and monitor network subscribers' behavior in terms of the aggregated traffic demands. This makes the studied demand forecasting a widely useful IoT application. Second, the interconnection between IoT devices and control centers will be realized by the machine-type communications in cellular networks via various low power wide area network technologies [9], where the IoT-type traffic demand forecasting in cellular networks will be critical to facilitate an efficient network resource schedule and management for IoT services,

especially in the context of service-oriented network operation in 5G networks [10]. Third, the proposed graph-sequence models of mobile demand forecasting may be extended to other forecasting problems in the domain of IoT, e.g., IoT-enabled load forecasting in smart grid [11], [12] and air quality forecasting [13].

In this paper, we study the mobile demand forecasting, the foundation of predictive mobile network management. In the literature, the mobile traffic/demand forecasting schemes have been studied for traffic apprehension and prediction via the Holt–Winter's exponential smoothing technique [14], information theory [15], and the seasonal auto regression integrated moving average (ARIMA) model [16]. However, all these demand forecasting models only consider the temporal aspect via various time series models without taking into account the spatial relevancy of cells. Models of mobile demand forecasting accounting for the spatial relevancy have been recently studied based on deep learning [17], [18]. In these models, the temporal aspect of demand time series is commonly studied via the recurrent neural networks (RNNs), while the spatial relevancy is captured by various grid-based spatial models.

However, the main challenge of applying grid-based spatial models to per-cell demand forecasting is the irregular spatial distribution of cells in the real-world setting. Generally, the cell towers are distributed in a network covered area according to the population density. That is, the distance between two cell towers is about 500 m. in the urban area, but can reach 2000 m in the rural area. Hence, grid-based models [17], [18] do note directly apply. To utilize the grid-based models, one first needs to redivide the network covered area into a uniform square grid, and then predict the aggregated demands of multiple cell towers residing in each lattice. Such spatial area redivision and demand aggregation will lead to the loss of the spatial granularity and will significantly limit the applications to future cellular network management that requires variable spatial granularity.

To this end, we propose a flexible *graph-based spatial model* for the per-cell demand forecasting without any spatial resolution degradation and data aggregation. First, we realize that the spatiotemporal analysis of the per-cell demand time series via the semivariogram [19] reveals that the relevancy between the demands of two cells relies on the spatial distance of the two cells. That is, the dependency level of two cells would decrease when their spatial distance increases. Hence, we can build a dependency graph characterizing the relevancy of cells based on their spatial distances. In others words, the per-cell demands generated at each cell tower could be regarded as signals generated at the vertices of a graph. In addition, not only the recent demand history is applied to forecast the future demands, but also the periodic history [e.g., day(s) ahead demands] are considered in order to obtain an accurate demand predictor.

With the dependency graph formulation, the recently developed graph convolutional networks (GCNs) [20], [21] and the long short-term memory (LSTM) neural networks [22] are employed to characterize the spatial aspect and the temporal aspect for demand forecasting, respectively. The LSTM is a gated version of RNNs in deep learning, which is well known for its good performance on sequence modeling. In GCNs, the graph convolution operation, originated from signal processing theory on graphs [23]–[25], is employed to replace the matrix multiplication in the feedforward neural networks. The power of graph convolution results from the ideas of parameter sharing and sparse interaction as in the traditional convolutional neural networks [26]. The sparse interaction in per-cell demand prediction means that the demand prediction of one cell is only related to itself and its nearest neighbors in the dependency graph. The parameter sharing assumes that the model parameters are shared across all cells of the network.

In this paper, we first formulate the demand forecasting problem as a one-step ahead demand prediction problem. The demand forecasts after one step in the future are dynamically generated by the one-step ahead predictor. Three models, namely the spatial-only (GCNs), the temporal-only (LSTM), and the spatiotemporal [graph convolutional LSTM (GCLSTM)], are studied. The GCLSTM [27] is the model replacing the matrix multiplication operation with the graph convolution operation in LSTM, inspired by the convolutional LSTM [28]. Compared with GCLSTM, LSTM without the embedded spatial information will predict the demand of one cell based on all other cells in the network, which would lead to an inferior generalization performance. Experiments show that the temporal-only LSTM could achieve a superior performance for the very-short-term demand forecasting for its much larger model capacity, but rapidly deteriorates when the forecast horizon increases. This results from the inferior generalization performance of LSTM and the accumulated errors in the generated predicts. The GCLSTM with the spatial and the temporal aspects modeled will generally have a superior forecast performance except for the very-short-term one. Main contributions of this paper are summarized as follows.

1) To the best of the authors' knowledge, this is the first work modeling the spatial relevancy among cells by a dependency graph. The graph-based spatial modeling could completely retain the spatial granularity without any data aggregation.
2) The periodicity of the per-cell demand time series is explicitly taken into account by adding past periodic observations as input features in our studied models so that the accuracy of demand forecasting could be enhanced without significantly increasing model size.
3) The graph convolutional and RNNs are proposed to simultaneously characterize both the spatial and temporal attributes with parameter sharing, which could lead to a superior generalization performance of demand forecasting.

The rest of this paper is organized as follows. In Section II, the studied dataset is described and the per-cell demand is defined. The per-cell demand forecasting problem in Section III is identified with the spatial and temporal aspects modeled. In Section IV, three demand prediction models based on GCN, LSTM, and GCLSTM are proposed. In Section V, experiment results are compared to demonstrate the superior performance of GCLSTM. Finally, concluding remarks are made in Section VI.

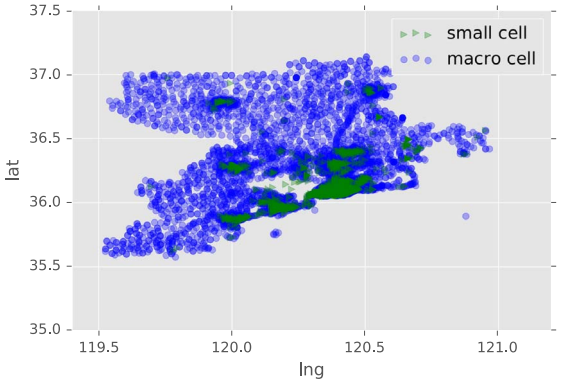


Fig. 1. Cell distribution heatmap.

II. Dataset

A. Signaling Dataset

The signaling data is collected near the radio access networks in a cellular network, which records communication events as well as location update events on all active subscribers in mobile networks. Data fields of the signaling data include: 1) subscriber's anonymized identifier; 2) time stamp (e.g., 20160101184312); 3) location coordinates (i.e., the longitude and latitude of cell towers); 4) event type; and 5) cell type (i.e., small cell or macro cell). The longitude and latitude coordinates where the cell tower is located are accurate to 6 decimal places and time stamps are accurate to seconds. The signaling data logs event type as well as the direction of the event (e.g., initiating a call or being called). In the studied dataset, more than 6000 cells in total including small and macro cells with millions of subscribers are recorded in the studied dataset, as shown in Fig. 1. In the studied dataset, the average daily active subscribers is about three million. The time period of the studied signaling data is 104 days, from August 22, 2016 to December 3, 2016.

B. Per-Cell Demands

Based on the studied signaling dataset, two categories of service demands could be extracted, namely communication demands and tracking demands. The communication demands including the first 4 events on calls and texts recorded in the signaling dataset, to forecast which is the very task of this paper. The tracking demands could be obtained based on the location update events, which is closely related to crowd mobility. The location update frequency is once per hour, which may be too coarse to exactly describe the crowd flow, especially in the urban area (where cells are densely distributed). Hence, we focus on the communication demand forecasting in this paper.

With the spatiotemporal information of each event recorded, we define the per-cell demand as the number of communication events occurring within a cell during an event counting time window ΔT. Hence, a per-cell demand time series could be generated as follows:

$$\left[x_t^n, x_{t-1}^n, x_{t-2}^n, \ldots, x_{t-l+1}^n, \ldots\right] \tag{1}$$

where $x_t^n = \ln(1+c_t^n)$ denotes the per-cell demand within time window $[t-\Delta T, t)$, where c_t^n is the number of communication events of the nth cell. Here, we utilize the commonly used logarithm function $\ln(1+x)$ to convent the integer event number domain to the real number domain of demands. In this paper, we mainly study the demand forecasting in terms of the 10-min counting time windows, i.e., $\Delta T = 10$.

It can be clearly observed that small cells are densely deployed in the studied urban area (green areas as shown in Fig. 1). In a heterogeneous cellular networks, small cells are designed to assist their corresponding macro cell by offloading data traffic, whose coverage is also relatively much smaller than that of macro cells. As a result, the communication demands of small cells is sparse, which is not of interest in this paper. Hence, we aggregate the demand of small cells to its corresponding macro cell, which is determined by their spatially closest macro cell based on the location information (i.e., the longitude and latitude of cell towers). In other words, we study the per-cell aggregated demands within a spatial area covered by a macro cell.

In Fig. 2, the per-cell demands with different cell types are illustrated, namely business, entertainment, and residence. In each subfigure, three demand time series with different counting time window are plotted, $\Delta T = 5$ min, $\Delta T = 10$ min, and $\Delta T = 20$ min. One can easily observe that the large counting time window could significantly reduce the noise of the per-cell demand time series, as the larger counting time window acts like a smoothing filter applied on the one generated by the small counting time window. However, such noise reduction is at the cost of lowering the temporal resolution of demand time series. In addition, it can be easily observed that per-cell demands are strongly periodic in terms of calendar days, regardless of cell types. Another periodic effect, that the demands during weekends is obviously less than those during weekdays, could be observed from the demand time series of the business type [Fig. 2(a)]. Such effects would inspire the feature engineering for demand forecasting, which will be discussed in detail later.

III. Demand Prediction Problem Formulation

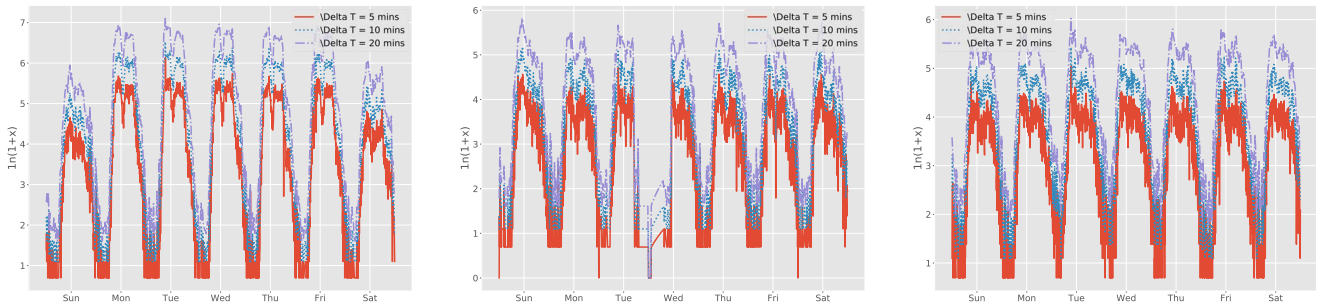
With the definition of per-cell demands, the demand forecasting is aimed to predict the per-cell demands of all cells in a mobile network based on its history. In this paper, demand forecasting is studied as the one-step ahead prediction problem as follows:

$$\hat{\boldsymbol{x}}_{t+1} = f(\boldsymbol{x}_t, \boldsymbol{x}_{t-1}, \ldots, \boldsymbol{x}_{t-l+1}, \ldots) \tag{2}$$

where $\boldsymbol{x}_t = [x_t^1, x_t^2, \ldots, x_t^N]^T$ denotes the per-cell demands of cells across the covered area at time t and N is the total number of macro cells in the network. Hence, the prediction problem essentially amounts to the estimation of a function or predictor f based on the collected history data and the knowledge of cell locations. In this section, we will discuss the one-step ahead demand prediction with the innovative spatiotemporal modeling.

A. Graph-Based Spatial Formulation

By the spatiotemporal analysis of multiple per-cell demand time series (see Section V-B, Appendix B), it can be concluded



(a) (b) (c)

Fig. 2. Demand time series of various cell type, where the 7-day demands are recorded from November 27, 2016 to December 3, 2016 and 24-h demands are recorded on November 27, 2016. (a) Business cell is located in the central business district (CBD), (b) entertainment-type cell is located in a public park, and (c) residence area is located in a large residential area.

that the demand relevancy between two cells declines when their spatial distance increases. Hence, we first propose to model the spatial relevancy between cells in the network by a dependency graph. The adjacency matrix $\boldsymbol{A}$ of the dependency graph can be obtained based on the spatial distance between cells as follows:

$$\boldsymbol{A}_{ij} = \begin{cases} 1, & \text{dist}(s_i, s_j) \le \zeta \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

where s_i denotes the location of cell i and ζ is the threshold, a hyperparameter that could be tuned. We set $\zeta = 2$ km in this paper. In fact, the threshold suggests that any two cells whose distance is beyond the threshold will be considered irrelevant. Such graph modeling could successfully make the cell relevancy sparse (from N^2 to $\sum_{i,j} \boldsymbol{A}_{i,j}$), which can lead to a good demand forecasting generalization performance with the graph modeled in the predictor as detailed in Sections IV and V. As a result, each cell could be regarded as a vertex in the spatial dependency graph and the per-cell demand $\boldsymbol{x}_t$ is viewed as the signal observed at each vertex of the graph at time t.

B. Periodicity-Based Temporal Features

As shown in Fig. 2, it is obvious that the per-cell demand time series is periodic with respect to calendar days or weeks. In fact, such periodicity could provide valuable information for one-step ahead per-cell demand prediction at time t. Accordingly, we could reformulate the per-cell demand time series in terms of calendar days at time t as follows:

$$\begin{bmatrix} x_t^i & x_{t-1}^i & \cdots & x_{t-L+1} & \cdots \\ x_{t-n_d}^i & x_{t-1-n_d}^i & \cdots & x_{t-L+1-n_d} & \cdots \\ x_{t-2n_d}^i & x_{t-1-2n_d}^i & \cdots & x_{t-L+1-2n_d} & \cdots \\ \ddots & \ddots & \ddots & \ddots & \ddots \\ x_{t-7n_d}^i & x_{t-1-7n_d}^i & \cdots & x_{t-L+1-7n_d} & \cdots \\ \ddots & \ddots & \ddots & \ddots & \ddots \end{bmatrix}$$

where n_d denotes the number of per-cell demand observations in one calendar day. To predict x_{t+1}^i, not only the recent demand history $[x_t^i, x_{t-1}^i, \ldots, x_{t-L+1}^i]$ of cell i is taken into accounts, but also their corresponding days ahead demand observations will be regarded as input features for a predictor. Here, we only take the one-day ahead and 6-day ahead observations as the extra features in order to make the predictor more dependent on the current trend. Hence, the input features of all cells in the network at time t take the form

$$\boldsymbol{Z}_t = \left[\boldsymbol{z}_t^1, \boldsymbol{z}_t^2, \ldots, \boldsymbol{z}_t^N\right]^T \tag{4}$$

where $\boldsymbol{z}_t^i$ denotes the input features of cell i at time t, i.e., $\boldsymbol{z}_t^i = [x_t^i, x_{t-n_d}^i, x_{t-7n_d}^i]$.

C. Graph-Sequence Demand Prediction Formulation

Based on the spatial and temporal modeling discussed above, the one-step ahead demand prediction problem could be further expressed as

$$\hat{\boldsymbol{x}}_{t+1} = f(\boldsymbol{Z}_t, \boldsymbol{Z}_{t-1}, \ldots, \boldsymbol{Z}_{t-L+1}; \boldsymbol{A}) \tag{5}$$

where L is the length of recent history used for demand prediction. We will discuss the selection of L in Section V. In this paper, we employ the commonly used mean absolute predicted error (MAE) as the evaluation criterion and cost function. Hence, the demand prediction problem could be expressed as follows:

$$\min_f \frac{\mathbb{1}^T E\left[\left|\boldsymbol{x}_{t+1} - \hat{\boldsymbol{x}}_{t+1}\right|\right]}{N}. \tag{6}$$

Next, we will discuss the proposed per-cell demand predictor with effective graph and sequence information embedded based on deep learning.

IV. Deep Graph-Sequence Spatiotemporal Modeling

In this paper, the graph-based (GCN) model and the sequence-based model (LSTM) are first proposed to individually capture the spatial and temporal aspects, respectively. In addition, we study their integrated version (GCLSTM), which embeds the graph information in the sequence model.

A. Spatial Modeling—GCNs

The graph convolution is the convolution operation in graph signal processing (GSP) domain, defined as $g_\theta(\boldsymbol{L}) \star \boldsymbol{x}_t$, where $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{A}$ denotes the graph Laplacian and $g_\theta(\boldsymbol{L})$ denotes

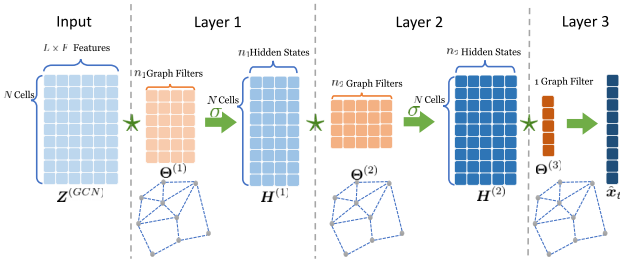


Fig. 3. Spatial modeling: GCNs.

a filter with respect to the graph $\boldsymbol{L}$. The graph convolution would relate the signal of one vertex to others in terms of the graph topology, where the corresponding graph filter coefficients could be trainable based on data. Details of the graph convolution and graph filter description refer to Appendix A.

As only the nearest neighbors are considered in this paper, the first-order graph filter based on (19), $g_\theta^{(1)}(\boldsymbol{\Lambda}) = \theta_0 + \theta_1 \boldsymbol{\Lambda}$, is considered. Kipf and Welling [20] proposed a simple first-order graph filer approximation based on Chebyshev polynomials of first kind [21] by forcing $\theta = \theta_0 = -\theta_1$ as follows:

$$g_\theta^{(1)}(\widetilde{\boldsymbol{L}}) \star \boldsymbol{x}_t = \widetilde{\boldsymbol{D}}^{-\frac{1}{2}} \widetilde{\boldsymbol{A}} \widetilde{\boldsymbol{D}}^{-\frac{1}{2}} \boldsymbol{x}_t \theta \tag{7}$$

where $\widetilde{\boldsymbol{A}} = \boldsymbol{I} + \boldsymbol{A}$ and $\widetilde{\boldsymbol{D}}$ is a diagonal matrix, $\widetilde{\boldsymbol{D}}_{ii} = \sum_j \boldsymbol{A}_{ij}$.

Therefore, a GCN could be built based on the approximated first-order graph convolution operation to replace the matrix multiplication in the feedforward neural networks, which embeds the prior knowledge of graph topology into the learning model. As a result, each layer of GCNs is defined as[1]

$$\boldsymbol{H}^{l+1} = \sigma\left(\widetilde{\boldsymbol{D}}^{-\frac{1}{2}} \widetilde{\boldsymbol{A}} \widetilde{\boldsymbol{D}}^{-\frac{1}{2}} \boldsymbol{H}^l \boldsymbol{\Theta}^l\right) \tag{8}$$

where $\sigma(\cdot)$ denotes the activation function for nonlinearity modeling. $\boldsymbol{H}^l \in \mathcal{R}^{N \times n_l}$ denotes the inputs of the lth layer and $\boldsymbol{\Theta}^l \in \mathcal{R}^{n_l \times n_{l+1}}$ is the trainable parameters in the model. Again, N denotes the number of vertices of the graph. In each graph convolution operation, the $\boldsymbol{H}^l \boldsymbol{\Theta}^l$ in (8) is first to learn the pattern in a cell-wise manner with shared parameters $\boldsymbol{\Theta}^l$. The product of $\boldsymbol{H}^l \boldsymbol{\Theta}^l$ and $\widetilde{\boldsymbol{D}}^{-(1/2)} \widetilde{\boldsymbol{A}} \widetilde{\boldsymbol{D}}^{-(1/2)}$ is essentially equivalent to the weighted sum over the cell and its first-order neighbors.

In the context of the per-cell demand prediction problem, we propose a three-layer GCN as the demand predictor f as detailed in Model 1 and Fig. 3.

Model 1 (GCN): A per-cell demand predictor is approximated by a three-layer GCN, $\hat{\boldsymbol{x}}_{t+1} = \hat{f}(\boldsymbol{Z}_t^{(\mathrm{GCN})}, A)$, i.e.,

$$\begin{aligned}
&\text{Layer 1:} \quad \boldsymbol{H}^{(1)} = \sigma\left(\widehat{\boldsymbol{A}} \boldsymbol{Z}_t^{(\mathrm{GCN})} \boldsymbol{\Theta}^{(1)}\right), \quad \boldsymbol{\Theta}^{(1)} \in \mathcal{R}^{(L \times F) \times n_1} \\
&\text{Layer 2:} \quad \boldsymbol{H}^{(2)} = \sigma\left(\widehat{\boldsymbol{A}} \boldsymbol{H}^{(1)} \boldsymbol{\Theta}^{(2)}\right), \quad \boldsymbol{\Theta}^{(2)} \in \mathcal{R}^{n_1 \times n_2} \\
&\text{Layer 3:} \quad \hat{\boldsymbol{x}}_{t+1} = \widehat{\boldsymbol{A}} \boldsymbol{H}^{(2)} \boldsymbol{\Theta}^{(3)}, \quad \boldsymbol{\Theta}^{(3)} \in \mathcal{R}^{n_2 \times 1}
\end{aligned} \tag{9}$$

where $\widehat{\boldsymbol{A}} = \widetilde{\boldsymbol{D}}^{-(1/2)} \widetilde{\boldsymbol{A}} \widetilde{\boldsymbol{D}}^{-(1/2)}$ and $\boldsymbol{Z}_t^{(\mathrm{GCN})}$ denotes the input of the GCN with L-length window.

[1]For simplicity, we ignore the bias terms in the presentation of each studied model.

Here, $\boldsymbol{Z}_t^{(\mathrm{GCN})}$ is the L-length demand history with days ahead features as the input, i.e.,

$$\boldsymbol{Z}_t^{(\mathrm{GCN})} = \left[\boldsymbol{Z}_t, \ldots, \boldsymbol{Z}_{t-L+1}\right].$$

In other words, the L-length demand history and extra days ahead features of each cell are regarded as its input features of GCNs without the explicit sequence modeling. As a result, the total number of free trainable parameters in the proposed three-layer GCN is $n_{h_1}(L \times F) + n_{h_1} n_{h_2} + n_{h_2}$.

B. Temporal Modeling—LSTM

In the literature, the RNNs is proved to be an effective sequence model [29], which is designed to capture the sequential information inherited in data, e.g., audio, nature language, etc. Essentially, RNNs adds a feedback path in the feedforward neural networks, which could provide the information of the previous inputs so that the current output is not only dependent on the current inputs but also relies on the hidden state learned from previous inputs as follows:

$$\boldsymbol{h}_t = \sigma(\boldsymbol{W} \boldsymbol{z}_t + \boldsymbol{V} \boldsymbol{h}_{t-1}) \tag{10}$$

where $\boldsymbol{h}_{t-1}$ denotes the hidden states updated previously.

The LSTM networks is one of special designed RNNs, which has a capability of controlling the updating process by adding three gates, namely input gate $\boldsymbol{g}_i$, forget gate $\boldsymbol{g}_f$, and output gate $\boldsymbol{g}_o$ in a LSTM cell

$$\begin{aligned}
\boldsymbol{g}_i &= \sigma(\boldsymbol{W}_i \boldsymbol{z}_t + \boldsymbol{V}_i \boldsymbol{h}_{t-1}) \\
\boldsymbol{g}_f &= \sigma\left(\boldsymbol{W}_f \boldsymbol{z}_t + \boldsymbol{V}_f \boldsymbol{h}_{t-1}\right) \\
\boldsymbol{g}_o &= \sigma(\boldsymbol{W}_o \boldsymbol{z}_t + \boldsymbol{V}_o \boldsymbol{h}_{t-1})
\end{aligned} \tag{11}$$

where $\sigma(\cdot)$ denotes the sigmoid function. These gates control how much information should be passed through in different places of LSTM cells as follows:

$$\begin{aligned}
\boldsymbol{c}_t &= \boldsymbol{g}_f \circ \boldsymbol{c}_{t-1} + \boldsymbol{g}_i \circ \tanh(\boldsymbol{W}_c \boldsymbol{x}_t + \boldsymbol{V}_c \boldsymbol{h}_{t-1}) \\
\boldsymbol{h}_t &= \boldsymbol{g}_o \circ \tanh(\boldsymbol{c}_t)
\end{aligned} \tag{12}$$

where $\boldsymbol{c}_t$ and $\boldsymbol{h}_t$ denote the cell state and the hidden state at time t, respectively. Here, the operator "$\circ$" denotes the element-wise multiplication. In LSTM, the cell state is employed to remember the current state of the cell and the hidden state records the output of the LSTM cell, which could be further inputted to next layer of the network.

In this paper, we propose a three-layer LSTM network as a per-cell demand predictor as described in Model 2, which regards the per-cell demand of all cells at each time stamp as inputs.

Model 2 (LSTM): A per-cell demand predictor is approximated by a three-layer LSTM network with two LSTM layers and one full-connection layer. The LSTM sequence model is demonstrated in Fig. 4 and illustrated mathematically as follows:

$$\begin{aligned}
&\text{Layer 1:} \quad \left(\boldsymbol{h}_t^{(1)}, \boldsymbol{c}_t^{(1)}\right) = \eta_{\mathrm{lstm}}^{(1)}\left(\boldsymbol{z}_t^{(\mathrm{LSTM})}, \boldsymbol{h}_{t-1}^{(1)}, \boldsymbol{c}_{t-1}^{(1)}\right) \\
&\text{Layer 2:} \quad \left(\boldsymbol{h}_t^{(2)}, \boldsymbol{c}_t^{(2)}\right) = \eta_{\mathrm{lstm}}^{(2)}\left(\boldsymbol{h}_t^{(1)}, \boldsymbol{h}_{t-1}^{(2)}, \boldsymbol{c}_{t-1}^{(2)}\right) \\
&\text{Layer 3:} \quad \hat{\boldsymbol{x}}_{t+1} = \boldsymbol{W}^{(3)} \boldsymbol{h}_t^{(2)}
\end{aligned} \tag{13}$$

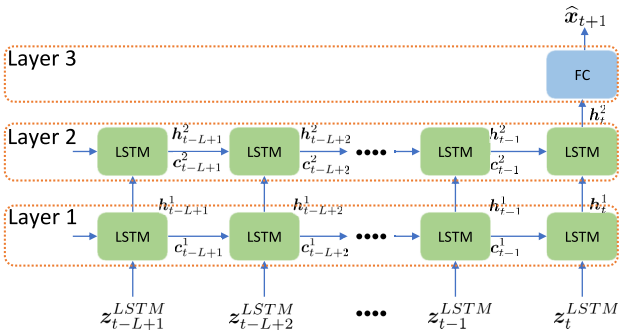


Fig. 4. Temporal modeling: LSTM.

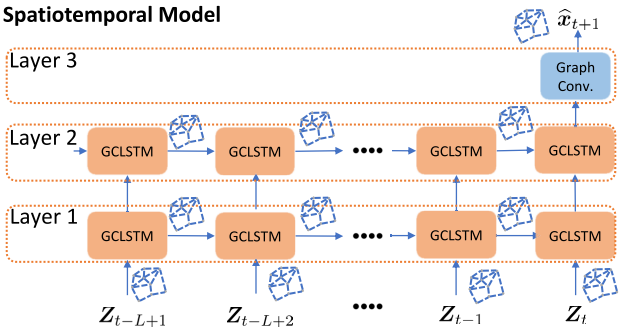


Fig. 5. Spatiotemporal modeling: GCLSTM.

where $\eta^{(i)}_{\text{lstm}}(\cdot,\cdot,\cdot)$ denotes the updating function of the layer i LSTM cell as described in (11) and (12), in which the trainable parameters are listed as follows:

$$\begin{aligned}
&\text{Layer 1:} \quad \boldsymbol{W}^{(1)}_{i,o,f,c} \in \mathcal{R}^{(N\times F)\times n_{h_1}}, \boldsymbol{V}^{(1)}_{i,o,f,c} \in \mathcal{R}^{n_{h_1}\times n_{h_1}}\\
&\text{Layer 2:} \quad \boldsymbol{W}^{(2)}_{i,o,f,c} \in \mathcal{R}^{n_{h_1}\times n_{h_2}}, \boldsymbol{V}^{(2)}_{i,o,f,c} \in \mathcal{R}^{n_{h_2}\times n_{h_2}}\\
&\text{Layer 3:} \quad \boldsymbol{W}^{(3)} \in \mathcal{R}^{n_{h_1}\times n_{h_2}}
\end{aligned}$$

where n_{h_1} and n_{h_2} denote the size of hidden states in layers 1 and 2, respectively.

Here, the input $\boldsymbol{z}_t^{(\text{LSTM})}$ is a vector that contains features of all cells at time t, whose size is $(N \times F) \times 1$. As a result, the number of trainable parameters in Model 2 is $4n_{h_1}(N \times F + n_{h_1}) + 4n_{h_2}(n_{h_1} + n_{h_2}) + n_{h_2}N$. In LSTM, we only model the temporal aspect of the per-cell demand data, but omit the spatial information. In other words, the spatial local dependence is not considered in the LSTM model, but the full connection from one cell to all other cells are taken into account, which may lead to overfitting issue in the LSTM model.

C. Spatiotemporal Modeling—GCLSTM

With the spatial and temporal information modeled, the LSTM and GCN can be integrated to utilize both the spatial and temporal information, which is termed as GCLSTM. In GCLSTM, the global connection among vertices (matrix multiplication in LSTMs) is replaced by the local graph convolution (8) in each gates as follows:

$$\begin{aligned}
\boldsymbol{G}_i &= \sigma\big(\widehat{\boldsymbol{A}}(\boldsymbol{Z}_t\boldsymbol{\Theta}_i + \boldsymbol{H}_{t-1}\boldsymbol{\Psi}_i)\big)\\
\boldsymbol{G}_f &= \sigma\big(\widehat{\boldsymbol{A}}(\boldsymbol{Z}_t\boldsymbol{\Theta}_f + \boldsymbol{H}_{t-1}\boldsymbol{\Psi}_f)\big)\\
\boldsymbol{G}_o &= \sigma\big(\widehat{\boldsymbol{A}}(\boldsymbol{Z}_t\boldsymbol{\Theta}_o + \boldsymbol{H}_{t-1}\boldsymbol{\Psi}_o)\big)
\end{aligned} \tag{14}$$

where $\boldsymbol{G}_{i,f,o} \in \mathcal{R}^{N\times n_h}$. Also, the hidden states are also updated locally as follows:

$$\begin{aligned}
\boldsymbol{C}_t &= \boldsymbol{G}_f \circ \boldsymbol{C}_{t-1} + \boldsymbol{G}_i \circ \tanh\big(\widehat{\boldsymbol{A}}(\boldsymbol{Z}_t\boldsymbol{\Theta}_c + \boldsymbol{H}_{t-1}\boldsymbol{\Psi}_c)\big)\\
\boldsymbol{H}_t &= \boldsymbol{G}_o \circ \tanh(\boldsymbol{C}_t).
\end{aligned} \tag{15}$$

Accordingly, a per-cell demand predictor based on GCLSTM is proposed to model both the spatial and temporal dimension of the per-cell demand time series as illustrated in Model 3.

Model 3 (GCLSTM): A per-cell demand predictor is approximated by a three-layer GCLSTM with two layers of GCLSTM cells and one graph convolutional layer (as demonstrated in Fig. 5), i.e.,

$$\begin{aligned}
&\text{Layer 1:} \quad (\boldsymbol{H}^{(1)}_t, \boldsymbol{C}^{(1)}_t) = \eta^{(1)}_{\text{gclstm}}\Big(\boldsymbol{Z}_t, \boldsymbol{H}^{(1)}_{t-1}, \boldsymbol{C}^{(1)}_{t-1}\Big)\\
&\text{Layer 2:} \quad (\boldsymbol{H}^{(2)}_t, \boldsymbol{C}^{(2)}_t) = \eta^{(2)}_{\text{gclstm}}\Big(\boldsymbol{H}^{(1)}_t, \boldsymbol{H}^{(2)}_{t-1}, \boldsymbol{C}^{(2)}_{t-1}\Big)\\
&\text{Layer 3:} \quad \hat{\boldsymbol{x}}_{t+1} = \widehat{\boldsymbol{A}}\boldsymbol{H}^{(2)}_t\boldsymbol{\Theta}^{(3)}
\end{aligned} \tag{16}$$

where $\eta^{(i)}_{\text{gclstm}}(\cdot,\cdot,\cdot)$ denotes the layer i GCLSTM cell based on (14) and (15), where the trainable parameters are illustrated as follows:

$$\begin{aligned}
&\text{Layer 1:} \quad \boldsymbol{\Theta}^1_{i,f,o,c} \in \mathcal{R}^{F\times n_{h_1}}, \boldsymbol{\Psi}^1_{i,f,o,c} \in \mathcal{R}^{n_{h_1}\times n_{h_1}}\\
&\text{Layer 2:} \quad \boldsymbol{\Theta}^2_{i,f,o,c} \in \mathcal{R}^{n_{h_1}\times n_{h_2}}, \boldsymbol{\Psi}^2_{i,f,o,c} \in \mathcal{R}^{n_{h_2}\times n_{h_2}}\\
&\text{Layer 3:} \quad \boldsymbol{\Theta}^3 \in \mathcal{R}^{n_{h_2}\times 1}.
\end{aligned}$$

Again, n_{h_1} and n_{h_2} denote the size of hidden states in layers 1 and 2, respectively.

Here, the input $\boldsymbol{Z}_t$ at time t is a matrix with the shape $N \times F$ defined by (4). The number of trainable parameters is $4n_{h_1}(n_{h_1}+F)+4n_{h_2}(n_{h_2}+n_{h_1})+n_{h_2}$. Compared with LSTM, the number of trainable parameters could be largely reduced, since the parameters are shared across the graph with local dependence modeled. Such parameter sharing could mitigate the overfitting problem by structurally shrinking the capacity of the model. Details of model comparisons are summarized in Table I.

V. Experiments

In this section, we verify three proposed spatial, temporal, and spatiotemporal models based on the extracted per-cell demand data of 718 cell towers in the mobile network. The per-cell demands are first normalized by their mean and standard deviation in a cell-wise manner. The demand predictors proposed in this paper are implemented by PyTorch [30], which is a deep learning framework with automatic differentiation and dynamic computational graph. The training dataset is from August 22, 2016 to November 26, 2016 and the test dataset is from November 27, 2016 to December 3, 2016.

TABLE I
COMPARISONS OF THREE PER-CELL DEMAND PREDICTION MODELS

	Model Type	Input Dimension	Feature Size	Trainable Param. Num. (Example*)
GCN	Spatial	2-D Matrix $\boldsymbol{Z} \in \mathcal{R}^{N\times(L\times F)}$	$L \times F$	$n_{h_1}(L \times F) + n_{h_1} n_{h_2} + n_{h_2}$ $(2,208)$
LSTM	Temporal	1-D Vector $\boldsymbol{z}_t \in \mathcal{R}^{(N\times F)\times 1}$	$N \times F$	$4n_{h_1}(N \times F + n_{h_1}) + 4n_{h_2}(n_{h_1} + n_{h_2}) + n_{h_2} N$ $(310,976)$
GCLSTM	Spatiotemporal	2-D Matrix $\boldsymbol{Z}_t \in \mathcal{R}^{N\times F}$	F	$4n_{h_1}(n_{h_1} + F) + 4n_{h_2}(n_{h_2} + n_{h_1}) + n_{h_2}$ $(12,704)$

*$n_{h_1} = n_{h_2} = 32$, $F = 3$, $N = 718$ and $L = 12$

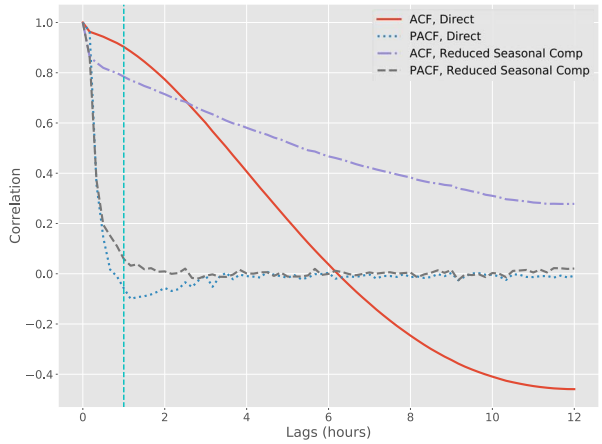


Fig. 6. ACF and partial ACF of demand time series with event counting time windows $\Delta T = 10$ min.

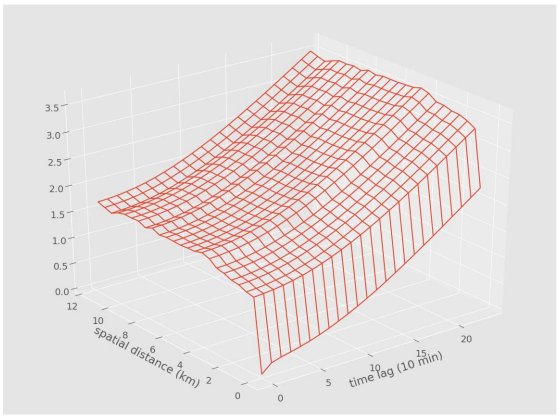


Fig. 7. Spatiotemporal semivariogram.

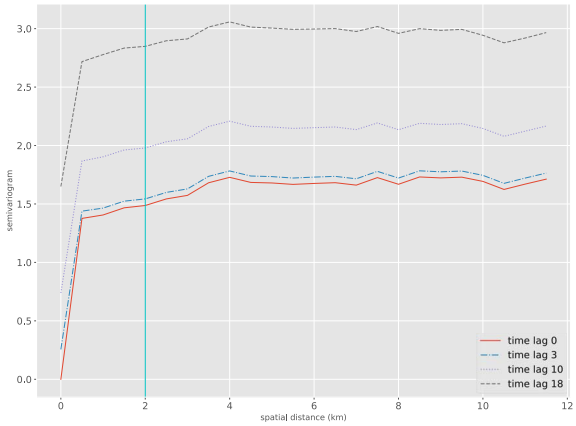


Fig. 8. Semivariogram in terms of spatial distance.

A. *Per-Cell Demands Autocorrelation Analysis*

We first investigate the autocorrelation analysis of the per-cell demands in a cell-wise manner, in order to determine the window length L should be taken into account for one-step ahead prediction. In the literature, the autocorrelation analysis and its partial derivative are commonly adopted to determine the order of ARIMA model. Specially, the autocorrelation function (ACF) would decide the order of the moving average, while the partial ACF could shed lights on the order selection for the autoregression. While the proposed time series model is quite different from ARIMA, the autocorrelation analysis could still be employed to suggest the window-length L selection.

Fig. 6 shows the correlation analysis on the per-cell demand time series with the counting time window, $\Delta T = 10$. As the per-cell demand is strongly periodic with respect to calendar days as shown in Fig. 2, the per-cell demand of cell i can be further decomposed into two parts, mean and its random component

$$\boldsymbol{x}_d^i = \bar{\boldsymbol{x}}_d^i + \boldsymbol{\epsilon}^i$$

where $\bar{\boldsymbol{x}}_d^i$ is the periodic component. Hence, Fig. 6 shows two kinds of curves, namely the direct and the periodic (seasonal) component reduced, which demonstrate the (partial) autocorrelation analysis directly on the per-cell demand time series and the random component, respectively. It could be clearly observed that the PACF curves rapidly decrease to zero with the time lag increased, while the ACF curves are slowly decreasing, especially the direct one. One can conclude that one hour history is sufficient for one-step ahead prediction, but longer history could benefit to capture the random component in the time series. As a result, we compare the different history lengths (half-hour, 1-hour, 2-hour, and 3-hour) for all three proposed models in various settings.

B. *Spatiotemporal Analysis*

The objective of the spatiotemporal analysis on the multiple per-cell demand signals is to evaluate how demand signals vary in space and time. In other words, the correlation between two signals in terms of both the time lags and the spatial distance is of significant interest. Such spatiotemporal analysis would lead to our critical spatial modeling of demands observed by many cells irregularly spatially distributed.

In this paper, the semivariogram, originated from spatial statistics, is employed to analyze the per-cell demands. Details of semivariogram refer to Appendix B. Fig. 7 shows the

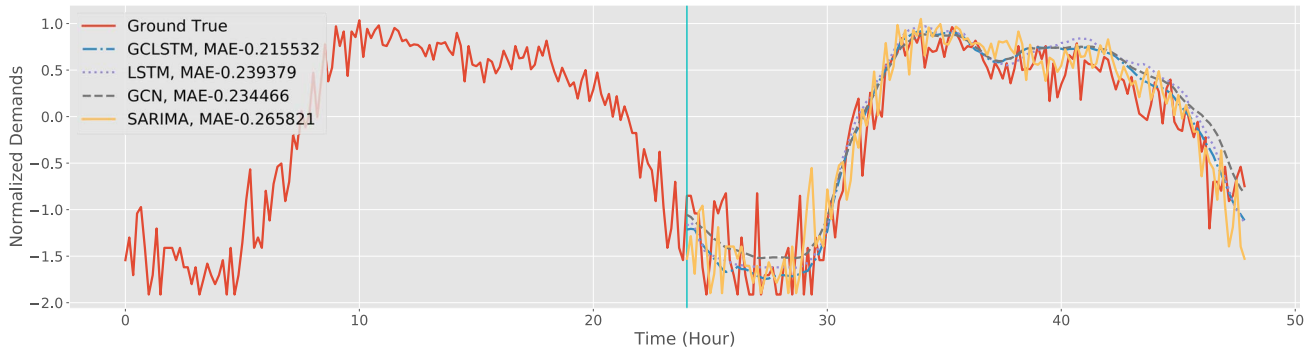


Fig. 9. Example of dynamic per-cell demand forecasting.

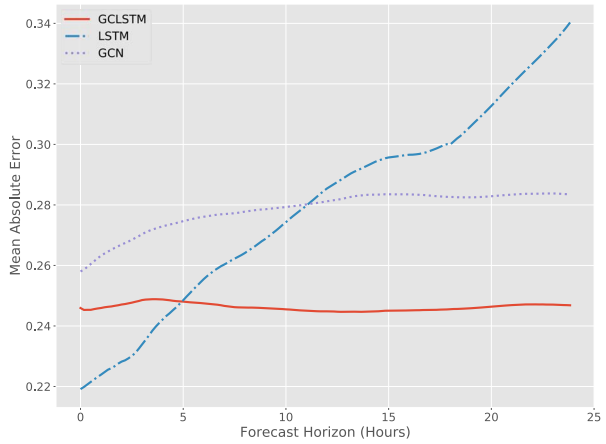


Fig. 10. MAE performance of dynamic forecasting over all cells.

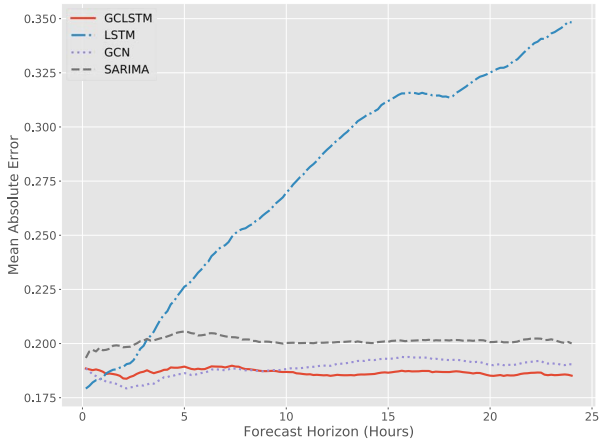


Fig. 11. Compared with SARIMA $(1, 0, 1) \times (1, 1, 1)$.

semivariogram of per-cell demand time series with different counting time window lengths $\Delta T = 10$. Based on the definition of semivariogram (22), the small value of semivariogram indicates the high dependence between signals separated at distance h and time lag τ. It could be observed that the semivariogram slowly grows along the time lag axis when $h = 0$, which suggests that the current per-cell demand is high correlated with its own history.

As for the spatial dependence, it can be observed in Fig. 8 that the value of semivariogram will stay the same after the spatial distance is 4 km. Such flat curve suggests that any two cells with the distance larger than 4 km could be considered as irrelevant. In this paper, two-layer graph convolution operations are employed in each graph-based model, to mimic second order graph filter based on the simple first-order graph filter approximation. Accordingly, we set the threshold ζ to be 2 km to capture the neighbors within 4 km after two-layer graph convolution operations.

C. Prediction Performance

In this paper, we employ the MAE as the criterion to evaluate the predictors studied in this paper. Though the forecast problem is formulated as a one-step ahead prediction problem (2), the per-cell demand predictor should be capable of forecasting the demands of a future time window. In fact, the demand forecasting is fulfilled by the dynamic prediction via the one-step ahead predictor, which would take predicted demands as inputs to further forecast the future demands, e.g., $\hat{\boldsymbol{x}}_{t+2} = f(\hat{\boldsymbol{x}}_{t+1}, \boldsymbol{x}_t, \ldots)$.

As a results, two parameters, forecast horizon and forecast resolution, are important for a forecasting problem. The forecast resolution relies on the length of event counting time window, which is a predict per 10 min in this paper. In this paper, we focus on the studied models with the forecasting horizon of 24 h. In [16], a seasonal ARIMA model is proposed to predict the per-cell demands of a single cell with seasonal component modeled, SARIMA $(1, 0, 3) \times (1, 1, 1)$

$$\begin{aligned}&\left(1 - ar_1 z^{-1}\right)\left(1 - sar_1 z^{-n_d}\right)\left(1 - z^{-n_d}\right)x_t^i \\ &= \left(1 + ma_1 z^{-1} + ma_2 z^{-2} + ma_3 z^{-3}\right)\left(1 + sma_1 z^{-n_d}\right)\epsilon_t\end{aligned}$$

where ϵ_t denotes the noise component and z_{-1} denotes the operation of one time lag. Though SARIMA cannot model the spatial correlation among cells nor simultaneously predict the per-cell demands across the entire network, we could still perform the comparisons in a cell-wise manner. In Fig. 9, an example of 24-hour demand forecasting of a cell is showed, including the proposed models and the SARIMA. It could be clearly observed that the predicts by the SARIMA is more fluctuate than that of our models, while

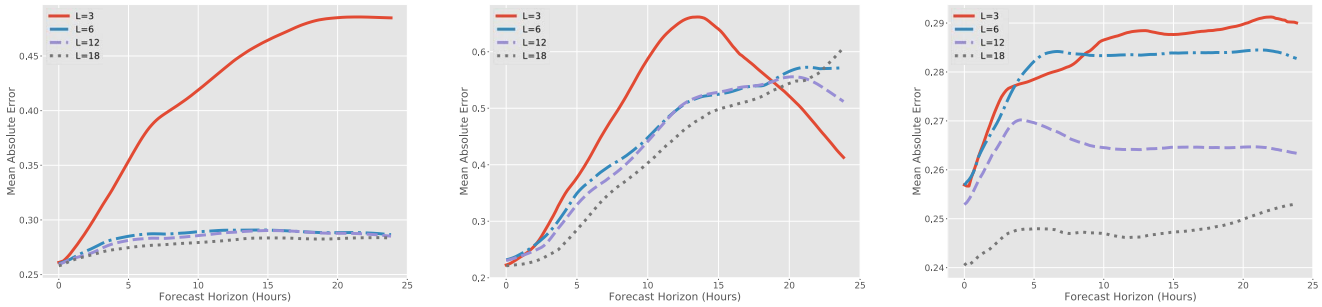


Fig. 12. MAE comparison between different window length L, where the event count time window is 10 min. (a) GCN. (b) LSTM. (c) GCLSTM.

our proposed models smoothly trace the ground truth curve. In Fig. 10, the average predicted MAE comparisons among three proposed models over all cells in the network is demonstrated. Overall, the spatiotemporal model (GCLSTM) is the best except for the case that forecast horizons are less than 5 h. As the capacity of the LSTM model without parameter sharing and locality modeling is much larger than the one of GCLSTM, demonstrated by their number of trainable parameters (see Table I), the LSTM can well capture the insight for one-step ahead prediction. However, the LSTM also easily models the noise into the predictor during training, which could lead to the overfitting issue and worsen the forecasting performance of the model. Fig. 11 also demonstrates the our proposed GCLSTM model performs better than the SARIMA.

Fig. 12 illustrates the differences of demand history length for per-cell demand prediction. Overall, the longer demand history could definitely improve the accuracy for large forecast horizon, especially for the LSTM-based models, which may result from the hidden states of LSTM-based models could remember more information when their hidden states are updated longer. On the other hand, the GCN model is not sensitive to the demand history length when $L \geq 6$ (longer than or equal to 1 h) due to the lack of explicit temporal modeling, as shown in Fig. 12(a).

D. Discussion

As demonstrated in the experiment results, the LSTM model could always have the best performance for the very-short-term demand forecasting, namely less than 3 h. However, due to accumulated error during dynamic prediction and week generalization of the LSTM model, the GCLSTM model is more capable for the short-term, mid-term, and day ahead demand forecasting. The GCN is also stable for such forecast horizons but is less accurate, while the number of trainable parameters is much smaller as illustrated in Table I. The SARIMA model performs well for the per-cell demands prediction task, but it is modeled in a cell-wise manner. That is, the per-cell demand needs to be predicted cell-by-cell. As a result, the parameters of SARIMA is linearly scaling with the number of cells in the network, while our proposed GCLSTM takes both the spatial and temporal into accounts with fixed number trainable parameters and could have a relative small trainable parameters for a large mobile network.

VI. Conclusion

In this paper, we study the per-cell demand forecasting in cellular networks. To deal with the irregular cell spatial distribution for spatial relevancy modeling among cells, we proposed to model the spatial relevancy among cells as a dependency graph based on spatial distances among cells without losing spatial granularity. Accordingly, we studied three models for demand forecasting, the spatial only (graph), the temporal only (sequence), and the spatiotemporal model (graph-sequence) based on deep learning. The spatiotemporal model simultaneously could capture both the spatial and temporal aspects in demand forecasting, which could achieve a superior forecasting performance demonstrated by experiment results.

Appendix A
Graph Filters and Graph Convolution

The GSP [23]–[25] is recently developed to deal with signals generated from a graph, such as social networks and sensor networks, which is a general extension of the traditional signal processing techniques from regular sampled data (e.g., audio or image) to the irregular data (social network data). The GSP combines both the signal processing and graph spectral theory, to fulfill the standard signal processing operations on the graph, e.g., convolution, filtering, translation, etc.

The main motivation of building a spatial dependence graph in this paper is to predict the demand of one cell not only based on the its own demand history but also taking the demand history of its neighbors into account. In the GSP theory, such motivation could be captured by the graph Laplacian operation

$$(\boldsymbol{L} \cdot \boldsymbol{x}_t)_i = \sum_{j \in \mathcal{N}_i} \left[x_t^i - x_t^j \right] \tag{17}$$

where $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{A}$ is the graph Laplacian and $\boldsymbol{D}$ is the diagonal matrix, i.e., $\boldsymbol{D}_{ii} = \sum_j A_{ij}$, recording the connectivity of each vertex in the graph. Intuitively, the graph Laplacian operation is essentially to capture the information of one vertex and its nearest neighbors.

Analogous to the filter design in the traditional signal processing, a graph filter could be expressed as polynomials in terms of the graph Laplacian [23]

$$g_\theta(\widetilde{\boldsymbol{L}}) = \theta_0 \boldsymbol{I} + \theta_1 \widetilde{\boldsymbol{L}} + \theta_2 \widetilde{\boldsymbol{L}}^2 + \cdots + \theta_K \widetilde{\boldsymbol{L}}^K \tag{18}$$

where $\widetilde{\boldsymbol{L}}$ is the normalized graph Laplacian, i.e., $\widetilde{\boldsymbol{L}} = \boldsymbol{I} - \boldsymbol{D}^{-(1/2)}\boldsymbol{A}\boldsymbol{D}^{-(1/2)}$. And θ_k is the filter coefficient of tap k. The order of graph filters would determine the order of neighbors of vertices in the graph affected by the filter.

By the eigendecomposition on the graph Laplacian, $\widetilde{\boldsymbol{L}} = \boldsymbol{U}\Lambda\boldsymbol{U}^T$, any graph signal could be transformed to the corresponding graph spectral domain, $\boldsymbol{X} = \boldsymbol{U}\boldsymbol{x}$, analogous to the discrete Fourier transform [23]–[25], where the eigenvectors $\boldsymbol{U}$ are viewed as a basis. As a result, the graph filter could be further expressed in the graph spectral domain

$$g_\theta(\boldsymbol{\Lambda}) = \theta_0 + \theta_1 \boldsymbol{\Lambda} + \theta_2 \boldsymbol{\Lambda}^2 + \cdots + \theta_K \boldsymbol{\Lambda}^K. \tag{19}$$

Hence, the graph convolution operation $g_\theta(\widetilde{\boldsymbol{L}}) * \boldsymbol{x}_t$ can be calculated as multiplication operations in the graph spectral domain

$$g_\theta(\widetilde{\boldsymbol{L}}) \star \boldsymbol{x}_t = \boldsymbol{U} g_\theta(\boldsymbol{\Lambda}) \boldsymbol{U}^T \boldsymbol{x}_t. \tag{20}$$

APPENDIX B
SPATIOTEMPORAL SEMIVARIOGRAM

The per-cell demand time series (1) could be further expressed in terms of both the spatial and temporal aspects as follows:

$$z(s_n,\ t) = x_t^n \tag{21}$$

where s_n represents the detailed spatial information of the nth cell (i.e., location coordinates). The semivariogram $\gamma(h)$ is a function to describe the spatial dependence of two stochastic processes generated in two locations s_n and s_m separated at h distance

$$\gamma(h) = E\Big[(z(s_n) - z(s_m))^2 \Big|\ \text{dist}\ (s_n, s_m) = h\Big].$$

With the temporal dependence considered, the time lag τ should be further considered atop the spatial variogram $\gamma(h)$

$$\gamma(h, \tau) = E\Big[(z(s,\ t) - z(s+h, t+\tau))^2\Big].$$

However, the cell towers are distributed irregularly in the covered area according to the population density. Hence, we analyze the multiple per-cell demand processes in terms of the empirical spatiotemporal semivariogram [19], [31] as follows:

$$\gamma(h(l),\ \tau) = \frac{1}{|\mathcal{N}(h(l), \tau)|} \times \sum_{(n,m,t,t') \in \mathcal{N}(h(l),\tau)} \big[z(s_n, t) - z\big(s_m, t'\big)\big]^2 \tag{22}$$

where

$$\mathcal{N}(h(l), \tau) = \big\{\big(n, m, t, t'\big) | \text{dist}\ (s_n, s_m) \in h(l),\ \ \big|t - t'\big| = \tau\big\}.$$

The $\mathcal{N}(h(l), \tau)$ is a set to collect any signal pairs spatially separated at distance within the distance tolerance $h(l)$ and temporally separated at τ. The distance tolerance $h(l)$ is employed to discretize the continuous spatial distance. In this paper, we utilize a linear uniform discretization with the spatial resolution 0.5 km. As a result, $h(l) = [(l-1) \times 0.5, l \times 0.5)$.

REFERENCES

[1] X. Cheng, L. Fang, X. Hong, and L. Yang, "Exploiting mobile big data: Sources, features, and applications," *IEEE Netw.*, vol. 31, no. 1, pp. 72–79, Jan./Feb. 2017.

[2] X. Cheng, L. Fang, L. Yang, and S. Cui, "Mobile big data: The fuel for data-driven wireless," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1489–1516, Oct. 2017.

[3] F. Malandrino, C.-F. Chiasserini, and S. Kirkpatrick, "Understanding the present and future of cellular networks through crowdsourced traces," in *Proc. 18th Int. Symp. World Wireless Mobile Multimedia Netw. (WoWMoM)*, Macau, China, Jun. 2017, pp. 1–9.

[4] M. Peng, D. Liang, Y. Wei, J. Li, and H.-H. Chen, "Self-configuration and self-optimization in LTE-advanced heterogeneous networks," *IEEE Commun. Mag.*, vol. 51, no. 5, pp. 36–45, May 2013.

[5] O. G. Aliu, A. Imran, M. A. Imran, and B. Evans, "A survey of self organisation in future cellular networks," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 1, pp. 336–361, 1st Quart., 2013.

[6] R. Li *et al.*, "Intelligent 5G: When cellular networks meet artificial intelligence," *IEEE Wireless Commun.*, vol. 24, no. 5, pp. 175–183, Oct. 2017.

[7] E. J. Kitindi, S. Fu, Y. Jia, A. Kabir, and Y. Wang, "Wireless network virtualization with SDN and C-RAN for 5G networks: Requirements, opportunities, and challenges," *IEEE Access*, vol. 5, pp. 19099–19115, 2017.

[8] H. Zhang *et al.*, "Network slicing based 5G and future mobile networks: Mobility, resource management, and challenges," *IEEE Commun. Mag.*, vol. 55, no. 8, pp. 138–145, Aug. 2017.

[9] U. Raza, P. Kulkarni, and M. Sooriyabandara, "Low power wide area networks: An overview," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 2, pp. 855–873, 2nd Quart., 2017.

[10] Z. Chang, Z. Zhou, S. Zhou, T. Chen, and T. Ristaniemi, "Towards service-oriented 5G: Virtualizing the networks for everything-as-a-service," *IEEE Access*, vol. 6, pp. 1480–1489, 2017.

[11] L. Li, K. Ota, and M. Dong, "When weather matters: Iot-based electrical load forecasting for smart grid," *IEEE Commun. Mag.*, vol. 55, no. 10, pp. 46–51, Oct. 2017.

[12] A. Y. Saber and T. Khandelwal, "IoT based online load forecasting," in *Proc. IEEE Green Technol. Conf. (GreenTech)*, Denver, CO, USA, Mar. 2017, pp. 189–194.

[13] C. Xiaojun, L. Xianpeng, and X. Peng, "IoT-based air pollution monitoring and forecasting system," in *Proc. Int. Conf. Comput. Comput. Sci. (ICCCS)*, Noida, India, Jan. 2015, pp. 257–260.

[14] D. Tikunov and T. Nishimura, "Traffic prediction for mobile network using Holt–Winter's exponential smoothing," in *Proc. 15th Int. Conf. Softw. Telecommun. Comput. Netw.*, Dubrovnik, Croatia, Sep. 2007, pp. 1–5.

[15] R. Li, Z. Zhao, X. Zhou, J. Palicot, and H. Zhang, "The prediction analysis of cellular radio access network traffic: From entropy theory to networking practice," *IEEE Commun. Mag.*, vol. 52, no. 6, pp. 234–240, Jun. 2014.

[16] 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.

[17] J. Zhang, Y. Zheng, D. Qi, R. Li, and X. Yi, "Dnn-based prediction model for spatio-temporal data," in *Proc. 24th ACM SIGSPATIAL Int. Conf. Adv. Geograph. Inf. Syst.*, Burlingame, CA, USA, Oct./Nov. 2016, pp. 1–4.

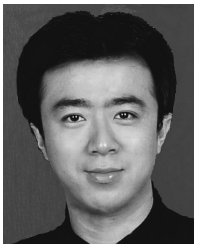
[18] J. Wang *et al.*, "Spatiotemporal modeling and prediction in cellular networks: A big data enabled deep learning approach," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Atlanta, GA, USA, May 2017, pp. 1–9.

[19] N. Cressie and H.-C. Huang, "Classes of nonseparable, spatio-temporal stationary covariance functions," *J. Amer. Stat. Assoc.*, vol. 94, no. 448, pp. 1330–1339, 1999.

[20] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. ICLR*, Paris, France, Apr. 2017, pp. 1–14.
[21] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proc. Conf. Neural Inf. Process. Syst. (NIPS)*, Barcelona, Spain, Dec. 2016, pp. 3844–3852.
[22] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.
[23] A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs," *IEEE Trans. Signal Process.*, vol. 61, no. 7, pp. 1644–1656, Apr. 2013.
[24] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Process. Mag.*, vol. 30, no. 3, pp. 83–98, May 2013.
[25] A. Sandryhaila and J. M. F. Moura, "Big data analysis with signal processing on graphs: Representation and processing of massive data sets with irregular structure," *IEEE Signal Process. Mag.*, vol. 31, no. 5, pp. 80–90, Sep. 2014.
[26] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
[27] Y. Seo, M. Defferrard, P. Vandergheynst, and X. Bresson, "Structured sequence modeling with graph convolutional recurrent networks," *eprint arXiv:1612.07659*.
[28] X. Shi *et al.*, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. Conf. Neural Inf. Process. Syst. (NIPS)*, Montreal, QC, Canada, Dec. 2015, pp. 802–810.
[29] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015.
[30] A. Paszke *et al.*, "Automatic differentiation in PyTorch," in *Proc. 31st Conf. Neural Inf. Process. Syst. (NIPS)*, Long Beach, CA, USA, Dec. 2017, pp. 1–4.
[31] X. Jian, R. A. Olea, and Y.-S. Yu, "Semivariogram modeling by weighted least squares," *Comput. Geosci.*, vol. 22, no. 4, pp. 387–397, May 1996.

Luoyang Fang (S'12) received the B.S. degree from the Department of Electronics and Information Engineering, Huazhong University of Science and Technology, Wuhan, China, in 2011. He is currently pursuing the Ph.D. degree at the Department of Electrical and Computer Engineering, Colorado State University, Fort Collins, CO, USA.

His current research interests include big data, mobile data, location privacy, data mining, distributed storage systems, and information-centric networking.



Xiang Cheng (S'05–M'10–SM'13) received the Ph.D. degree from Heriot-Watt University, Edinburgh, U.K., and the University of Edinburgh, Edinburgh, in 2009,

He is currently a Professor with Peking University, Beijing, China. He has authored or co-authored over 160 journal and conference papers, 3 books, and 6 patents. His current research interests include channel modeling and mobile communications.

Dr. Cheng was a recipient of the IEEE Asia–Pacific Outstanding Young Researcher Award in 2015, a co-recipient of the 2016 IEEE JSAC Best Paper Award, Leonard G. Abraham Prize, the NSFC Outstanding Young Investigator Award, the Second-Rank Award in Natural Science, Ministry of Education in China, the Best Paper Award of IEEE ITST'12, ICCC'13, ITSC'14, ICC'16, and ICNC'17, and the Post-Graduate Research Thesis Prize of the University of Edinburgh. He has served as the Symposium Leading-Chair, the Co-Chair, and a member of the Technical Program Committee for several international conferences. He is currently an Associate Editor for the IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS.

Haonan Wang received the Ph.D. degree in statistics from the University of North Carolina, Chapel Hill, NC, USA, in 2003.

He is currently a Professor of statistics with Colorado State University, Fort Collins, CO, USA. His current research interests include object-oriented data analysis, statistical analysis on tree-structured objects, functional dynamic modeling of neuron activities, and spatio-temporal modeling.

Liuqing Yang (S'02–M'04–SM'06–F'15) received the Ph.D. degree from the University of Minnesota, Minneapolis, MN, USA, in 2004.

Her current research interests include communications and signal processing.

Dr. Yang was a recipient of the Office of Naval Research Young Investigator Program Award in 2007, the National Science Foundation Career Award in 2009, the IEEE GLOBECOM Outstanding Service Award in 2010, the George T. Abell Outstanding Mid-Career Faculty Award, the Art Corey Outstanding International Contributions Award of CSU in 2012 and 2016, respectively, and the Best Paper Award of IEEE ICUWB'06, ICCC'13, ITSC'14, GLOBECOM'14, ICC'16, and WCSP'16. She has been actively serving of Technical Committees, including the organization of many IEEE international conferences, and on the Editorial Boards of a number of journals, including the IEEE TRANSACTIONS ON COMMUNICATIONS, the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, the IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, and the IEEE TRANSACTIONS ON SIGNAL PROCESSING.