Dual temporal gated multi-graph convolution network for taxi demand prediction

Taoru Yang[1] · Xiaopei Tang[1] · Rong Liu[2]

Abstract

Taxi demand prediction is essential to build efficient traffic transportation systems for smart city. It helps to properly allocate vehicles, ease the traffic pressure and improve passengers' experience. Traditional taxi demand prediction methods mostly rely on time-series forecasting techniques, which cannot model the nonlinearity embedded in data. Recent studies start to combine the Euclidean spatial features through grid-based methods. By considering the spatial correlations among different regions, we can capture how the temporal events have impacts on those with adjacent links or intersections and improve prediction precision. Some graph-based models are proposed to encode the non-Euclidean correlations as well. However, the temporal periodicity of data is often overlooked, and the study units are usually constructed as oversimplified grids. In this paper, we define places with specific semantic and humanistic experiences as study units, using a fuzzy set method based on adaptive kernel density estimation. Then, we introduce dual temporal gated multi-graph convolution network to predict the future taxi demand. Specifically, multi-graph convolution is used to model spatial correlations with graphs, including the neighborhood, functional similarities and landscape similarities based on street view images. As for the temporal dependencies modeling, we design the dual temporal gated branches to capture information hidden in both previous and periodic observations. Experiments on two real-world datasets show the effectiveness of our model over the baselines.

Keywords Spatiotemporal modeling · Demand prediction · Graph convolution · Street view

1 Introduction

Taxi is one of the most common forms of commute in our daily life. Due to the advantage of the latest internet technology and its infiltration to every aspect of the society, many large-scale online platforms have emerged for taxi requesting services, such as Uber, Didi Chuxing and Grab. All these mobile apps provide a more flexible and efficient way to satisfy passengers' demand and meanwhile lead to a reduction in time the drivers spend in an empty vehicle [1]. However, both traditional taxi service and app-based ride hailing still face the challenge of supply–demand imbalance, due to the following two-sided reasons. On the supply side, most drivers depend on their practical experiences to plan their routes and look for potential pickups [2], which is a blind action in some way. On the demand side, passengers tend to travel in an aggregated and random mode. For example, the taxi demand will rise during the morning and the evening peak and near the transportation hubs. Therefore, how to utilize the available data to predict the taxi demand is the key to mitigating the supply–demand disequilibrium. It also helps to better utilize road resources [3] and enhance the traffic management [4], which is a great leap forward in smart city construction [5].

Taxi demand is defined as the number of taxi request at one location at a time point [6]. Predicting taxi demand is a

---

Taoru Yang and Xiaopei Tang contribute equally.

✉ Rong Liu
  Rong.liu@dlr.de

  Taoru Yang
  yangtr5@mail3.sysu.edu.cn

  Xiaopei Tang
  tangxp6@mail2.sysu.edu.cn

[1] School of Geography and Planning, Sun Yat-Sen University, Xingang Road, Guangzhou, China

[2] Remote Sensing Technology Institute, Germany Aerospace Center(DLR), 82234 Weßling, Germany

challenging problem because of the inherent complex properties of the taxi demand. Specifically, it can be descried as three kinds of dependencies, namely time dependencies, spatial dependencies and exogenous dependencies [7]. Large quantities of researches have been done to take advantage of the dependencies to predict taxi demand.

The time dependencies are the relationship of the taxi demand at different timestamps [8]. Taxi demand at adjacent timestamps tends to be closer for the reason that the demand is in constant. What's more, it has strong periodicity according to the people's behavioral habits. For example, the demand increases in the rush hours every day. Thus, taxi demand prediction, similar to many other traffic prediction problems, can be comprehended as a time-series prediction problem, which deduces the future demand from the historical demand data. Representatively, autoregressive integrated moving average (ARIMA) and its improved versions have been successfully applied to traffic forecasting. ARIMA uses differencing to transform data into a stationary pattern, and integrates the autoregressive part and the moving average part by choosing proper parameters based on data analysis. Li et al. [9] proposed a variant of ARIMA to forecast the human mobility patterns in an urban taxi transportation system. Luis et al. [10] combined three time-series methods including ARIMA to predict the spatial distribution of taxi passengers for a short-term time horizon. Although traditional time-series methods have simple structure and strong operability using only temporal input, the spatial dependencies and exogenous dependencies are often overlooked in these models, both of which can improve taxi demand prediction results.

In fact, spatial dependencies of the taxi demand among different places are important. Places are not isolated but are connected to each other in many ways. According to Tobler's First Law of Geography, everything is related to everything else, but near things are more related than distant things [11]. Taking the physical distance as an example, the taxi demand of a region is generally closer to adjacent regions than the distant ones. In addition, exogenous dependencies such as weather and holiday event do have impact on daily taxi demand as well. Lots of efforts have been made to utilize these two types of dependencies to modify their models. Tong et al. [12] proposed a unified linear regression model named as Linear Unit Original Taxi Demand (LinUOTD), encoding the spatial, temporal and other external features into massive features with more than 200 million dimensions. These methods simulate overall trend of the data, but are likely to fail on unusual growth or slowdown in series.

Recently, deep learning has been successfully applied in computer vision [13–15], natural language processing [16] and network analysis [17]. The different techniques of deep learning, such as transfer learning [18–21], multi-task learning [22–24], semisupervised [25] and unsupervised learning [26], enrich the application scenarios greatly. Many researchers explored the potential of neural network in the traffic predicting problems [27–31]. The data-driven deep learning methods can better model the nonlinearity of taxi demand data and the dynamic data trend, which can be categorized into two types based on model dependencies:

(1)   Methods that only simulate the temporal dependencies. As one of the most representative works of time-series forecasting, recurrent neural network (RNN) is widely used in taxi demand prediction. Xu et al. [30] used long short-term memory (LSTM) to encode the useful information of the historical data in multiple layers, then passed the results to the mixture density networks and produced the demand predictions. Vanichrujee et al. [31] combined LSTM, gated recurrent unit (GRU) and extreme gradient boosting (XGBOOST) in an ensemble model to gain the best result. Although competitive predicting results can be gained merely consider the demand data from the past, most models did not pay attention to the different impacts of the historical data and the periodicity of the data on the result. These ignored properties can make a big difference to the predicting process.

(2)   Methods that combine the spatial and exogenous dependencies with temporal correlations. How to combine the spatial, temporal and exogenous segments properly is the key to model building. Yao et al. [32] treated the traffic in a city as an image and the taxi demand for a time period as pixel values and applied convolutional neural network (CNN) on the resulting images. The output of the CNN was fed to fully connected layers and LSTM layers, for subsequent concatenation with the exogenous relevant information. Lai et al. [33] presented a LSTM-based combination model, using a spatiotemporal component to capture the spatiotemporal information. An attribute component was also used to represent the exogenous dependencies (e.g., weather, point of interest).

Despite the success of applying CNN for aggregating spatial information, most works focus on constructing a Euclidean structure to simulate the traffic process and overlook the non-Euclidean factors. They model the spatial dependencies mainly based on physical distance and distribution among different places. However, the non-Euclidean relationships are critical as well. Places that share similar functionality are more likely to have similar tendency of taxi demand. For example, taxi request orders in residential areas rise in the morning peak, because most

people leave home for work during this period. Recent researches have shed light on the potential of graph convolutional network (GCN) on extracting spatial features in a non-Euclidean way. The spatiotemporal multi-graph convolution network (ST-MGCN) proposed by Geng et al. [34] encoded the non-Euclidean pairwise correlations among regions with multi-graph convolution and re-weighted different historical observations with contextual gated recurrent neural network (CGRNN). The multi-graph setting helps to tackle complicated problems in a multi-perspective way, which can be found useful in many domains [35–38]. Therefore, the multi-graph convolution provides the possibility to consider different types of non-Euclidean correlations simultaneously.

These methods tend to mesh the city into rectangular grids and take them as basic region units, for the convenience of data partition and the application of CNN. However, it fails to describe the places in a more realistic and perceptual way. The geographical concept of place is often used as "a portion of space" [39] within which people carry out day-to-day actions and routines [40]. When we think of a thriving business area, we refer an irregular region with obscure boundaries rather than a rectangular area with distinct boundaries. Zhu et al. [41] delineated place boundaries using a kernel density estimation and studied the place characteristics in geographic contexts through GCN. The basic comprehension of places provides the support for human-centroid understanding of geographic environment and geographic analysis, which is often overlooked or simplified under the scenario of taxi demand prediction.

In this paper, we propose a dual temporal gated multi-graph convolution network (DTA-GCN) to predict taxi demand, which is based on the structure of ST-MGCN. By utilizing the POI data, we first adopt a fuzzy set method combined with adaptive kernel density estimation to delineate multiple places' footprints. These extracted places are treated as basic geographic units and graph vertexes for taxi demand prediction. Then, we construct the multi-graph structure to model three different types of correlations among places. In addition to the neighborhood graph and the functional similarity graph, we also use the street view images to depict the street landscape of places and model the landscape similarity with graph. For each graph, dual CGRNN is used to aggregated information from the historical observations. Specifically, the dual temporal gated branches take the observations from previous timestamps and periodic timestamps as input, respectively. The temporal encoded features are then passed to the multi-graph convolution, modeling the non-Euclidean correlations among places. Finally, the taxi demand predictions can be generated by a subsequent fully connected layer. Our main contributions can be summarized as follows:

(1) Based on the original ST-MGCN, we design the dual temporal gated branches, adding another temporal branch to the CGRNN to capture the periodicity of the data. The dual branches can model the long- and short-term dependencies and leverage the periodic pattern, improving the robustness of the model.

(2) We embed landscape similarity among places in a graph when predicting taxi demand, as the supplement of functional correlation. The visual features of city landscape are extracted from street view images, for the reason that it has similar perspective with pedestrian and authentic description.

(3) We define places as the basic units of taxi demand study, which are extracted with a fuzzy set method instead of simply meshing the study area. The definition of places gives us a more intuitive understanding when observing the taxi demands.

The remaining paper is organized as follows: In Sect. 2, related works are introduced, including spatiotemporal prediction in social computing, graph convolution network and urban landscape analysis with street view images. The framework and details of the proposed model are described in Sect. 3. The experimental results and discussion are reported in Sect. 4. Section 5 concludes the paper.

2 Related work

2.1 Spatiotemporal prediction in social computing

Spatiotemporal prediction is a fundamental issue in social computing. With the development of the society and technology, the explosive growth in data storage capabilities enables us to easily trace the spatial and temporal properties of any historical event. How to capture useful information from the big data resources and make a reasonable prediction is the key to social management. Therefore, when we refer to a spatiotemporal prediction method, the most distinct part of the research is how to encode the spatial and temporal information, respectively, and merge them together. Taking taxi demand prediction as an example, lots of efforts have been made in recent years. In previous works, the Euclidean structure is naturally constructed to simplify the calculation and utilize convolution. Wei et al. [42] proposed a zero-grid ensemble spatiotemporal (ZEST) model, modeling all correlations separately and combining them at last. For the temporal predictor, they analyzed the data and designed the fluctuation rate. For the spatial predictor, they made use of the target grid's neighborhood data and trained an artificial neural network. Then, gradient boosting decision tree

(GBDT) was adopted to combine the results of different predictors. Ke et al. [43] chose to transform the LSTM network with convolutional techniques into a convolutional LSTM layer and proposed the fusion long short-term memory network (FCL-Net).

Non-Euclidean structured data, however, is more common in social computing, such as users and posts on social network, or traffic flow on roads. Processing the non-Euclidean data as graphs is helpful for data analysis. However, since the number of neighbors of non-Euclidean data is not fixed, it is hard for convolution neural network to operate. We need a new form of spatial information aggregation for graphs, which is graph convolution. In the traffic forecasting area, Cui et al. [44] first encoded the spatial information with graph convolution, then the output features were fed to long short-term memory neural network (LSTM). While in the spatiotemporal graph convolutional networks (STGCNs) proposed by Yu et al. [45], entire convolutional structure was used on the time axis instead of recurrent neural network. The temporal gated convolution was combined with the spatial graph convolution to form spatiotemporal convolutional blocks.

In ST-MGCN [34], contextual gated recurrent neural network (CGRNN) was proposed to incorporate the temporal global contextual information, and multi-graph convolution was used to model multiple spatial correlations with graphs. However, ST-MGCN only takes previous observations as input and does not pay attention to the periodic property of data. Besides, the place description and place correlation need to be further explored. In this paper, the spatial and temporal information encoding is modified based on the original model.

2.2 Graph convolution network

Graph convolutional network (GCN) is a new form of spatial information aggregation for graphs, which can be categorized into spatial-based and spectral-based methods [46]. Spatial-based methods define graph convolution based on the vertexes' spatial correlations and collect information within the neighborhood of vertexes. In this paper, we apply spectral-based GCN methods with a solid mathematical foundation. Given a graph $\mathbf{G} = (V, \mathbf{A})$, where V is the set of vertices and $\mathbf{A} \in R^{|V| \times |V|}$. A normalized graph Laplacian matrix can be defined as $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}}$, where $\mathbf{D}$ is a diagonal matrix of vertex degrees. The graph convolution, taking the signal and a filter as input, is based on the graph Fourier transform, where the basis is formed by eigenvectors of the normalized graph Laplacian. Spectral-based GCN methods all follow this framework, so the key difference is how to choose the filter to reduce the computational complexity.

Defferrad et al. [47] proposed ChebNet and approximated the filter by Chebyshev polynomials of the diagonal matrix of eigenvalues. A graph convolution operation is defined as:

$$\mathbf{X}_{l+1} = \sigma \left(\sum_{k=0}^{K-1} \alpha_k \mathbf{L}^k \mathbf{X}_l \right) \tag{1}$$

where $\mathbf{X}_l$ denotes the features in the l-th layer, α_k is the trainable coefficient, $\mathbf{L}^k$ is the k-th power of the graph Laplacian matrix, σ is the activation function.

2.3 Urban landscape analysis with street view images

City streets are important representatives of urban landscapes, because they serve as the main interface for the interaction between people and the city environment and the focal point of daily activities. Street view images describe the urban landscapes at ground level and relate directly to the human perceptions of the urban environment [48]. In urban landscape analysis, Li et al. [49–51] adopted a series of landscape indexes to quantify the landscape characteristics unfolded in street view images. However, these artificially designed indexes concentrate on the visual decomposition of image, and the presented features are very limited. Recently, the development of deep learning and the availability of large street view dataset provide a more automatic and sophisticated form of city sense. Urban landscape analysis therefore has steadily moved from a surface-level description to a quantitative tool for place analysis. Zhu et al. [41] investigated the feasibility of incorporating place connections to predict place characteristics. Places extracted from multi-source geodata are treated as graph vertices, and different types of connections are measured in the graph. When quantifying place characteristics, they used ResNet [52] as feature extractor to transform the street view image into a 512-dimensional visual feature vector. Place features were further gained by taking the average of all visual vectors within a place. The resulting features and graph connections were input to graph convolution network to predict the functional properties of places. The experiment described above showed the validity of using street view images to represent the place characteristics and mapping it to functional description. In this paper, we utilize the deep learning network to generate visual features to represent the urban landscape characteristics, and the landscape similarity graph is built based on the extracted features.

3 Methodology

In this section, we first introduce the problem definition of taxi demand prediction. Then we elaborate on delineating the places' footprints based on POI data, and the structure of the proposed DTG-MGCN.

3.1 Problem definition

All places in the city are regarded as a set of graph vertexes V, and the correlation among the vertexes is formulated as an adjacency matrix $\mathbf{A}$. Together they constructed a place-based graph $\mathbf{G} = (\mathbf{V}, \mathbf{A})$. Suppose $\mathbf{X}^{(t)}$ represent the number of orders of all places at the t-th timestamp. The taxi demand prediction needs to map the historical observations with a fixed temporal length T to the taxi demand in the next timestamp with a designed function $f : R^{|V| \times T} \to R^{|V|}$.

$$[\mathbf{X}^{(t-T+1)}, \ldots, \mathbf{X}^{(t)}] \xrightarrow{f} \mathbf{X}^{(t+1)} \tag{2}$$

In most cases, the input temporal length T is designed based on different lengths and sample rates of time series.

3.2 Delineating place boundaries

Different from all the grid-based model, we carry out the taxi demand prediction based on the concept of place. A place is a geographical area with location names, humanistic feelings and other properties [53, 54]. The density of multiple point sets reflects people's recognition of places. We collect a POI dataset and adopt a fuzzy method based on adaptive kernel density estimation proposed by Wang et al. [55] to identify the places. Each POI is labeled with a place name indicating the common business area it belongs to. For the point set in each place, the adaptive kernel density estimation was first applied to obtain a intuitive boundary. Suppose $x_i(i = 1, 2, \ldots, n)$ are independent identically distributed samples. In the region centered at x_i and with a radius h, the probability of x_i occurring decays with distance. It can be modelled with a kernel function. Kernel density estimation sums up the probability density functions of all samples to gain a continuous probability density surface. The formation of kernel density estimation is as follows:

$$f(x) = \frac{1}{nh^2} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \tag{3}$$

where f is the probability density function, h stands for the bandwidth and K represents the kernel function. Here, we apply the quadratic kernel proposed by Silverman et al. [56]. The bandwidth h decides the smoothness and plays an important role in estimation result. Overlarge h value leads to over simplified result, while small value pays too much

attention to the local variations within the point set, resulting in discrete regions. Therefore, an adaptive h value is needed for diverse sizes and point density of multiple places. Because the point set belongs to a specific place, we use the area of bounding rectangle and the total number of the point set, i.e., S and N, to calculate the adaptive bandwidth.

$$h = k\sqrt{\frac{S}{N}} \tag{4}$$

where k represents an adjustable coefficient. According to the equation, when the area is fixed, the larger the total number is, the denser the point distribution will be. We can get smaller bandwidth for dense point set and vice versa. The POI kernel densities are then normalized into [0,1], indicating to what extent an area belongs to this place. However, the unevenness of POI distribution also affects the estimated density value. For example, overly congregate POI in a few places will give rise to density value, which makes the remaining values less distinguishable and likely to be overlooked. To modify this situation, a fuzzy method is applied to further define the membership related to the kernel density. We use fuzzy membership function μ to map the normalized density to place membership and perform an alpha cut.

$$\mu(x) = \frac{1}{1 + \left(\frac{x}{m}\right)^{-s}} \tag{5}$$

where s stands for divergence and m is the middle point, i.e., the value of independent variable when the membership equals to 0.5. Finally, a threshold of 0.5 is adopted to delineate the core area of the place names. These delineated polygons are used as study units and graph vertexes in subsequent experiments.

3.3 Dual temporal gated multi-graph convolution network

We first encode the places and the multiple correlations among the places with multiple graphs. The extracted places are considered as graph vertexes, while the correlations are encoded as graph edges, which can be denoted by adjacency matrix in a mathematical form. With the constructed graphs, we adopt DTG-MGCN to model the spatial and temporal characteristics of the dataset and predict future taxi demand of places. First, dual temporal gated branches are used to aggregate information from the previous and the periodic observations, respectively. Second, we use multi-graph convolution to model different types of correlations among places, taking the encoded temporal features as input. Finally, a fully connected neural network transform features into taxi demand prediction.

3.3.1 Multi-graph construction

Adjacency matrix represents the correlation among graph vertexes. It is the key to operate graph convolution and the foundation of spatial information aggregation. In this work, three types of correlations are considered and transformed into the corresponding graph, including (1) the neighborhood graph $\mathbf{G}_N = (V, \mathbf{A}_N)$, which encodes the physical distance among places, (2) functional similarity graph $\mathbf{G}_F = (V, \mathbf{A}_F)$, which encodes the functional similarity among places based on the POI data, (3) landscape similarity graph $\mathbf{G}_L = (V, \mathbf{A}_L)$, which encodes the urban landscape similarity among places with the street view data. In neighborhood graph, the spatial proximity of any two places is measured by the Euclidean distance between the center of them. A threshold is used to define whether they are adjacent.

$$\mathbf{A}_{N,ij} = \begin{cases} 1, & dis(v_i, v_j) \leq L \\ 0, & otherwise \end{cases} \tag{6}$$

The function of places fundamentally determines the taxi demand. And places that share similar function tend to have similar trend of taxi request orders. As a data source with rich properties, POI data contain the address, place name, functional categories and specific coordinates of the point. It can sufficiently represent the functional characteristics of places. Therefore, in functional similarity graph, we measure the POI similarity with the POI feature vector.

$$\mathbf{A}_{F,i,j} = sim(\mathbf{P}_{v_i}, \mathbf{P}_{v_j}) \in [0, 1] \tag{7}$$

where $\mathbf{P}_{v_i}$, $\mathbf{P}_{v_j}$ are the POI feature vectors of place v_i and place v_j, respectively. The dimension of vector is equal to the number of functional categories, and each entry is equal to the number of points belonging to the corresponding category within a specific place.

Similarly, as for the landscape similarity graph, we calculate the landscape similarity using static street view images.

$$\mathbf{A}_{L,i,j} = sim(\mathbf{S}_{v_i}, \mathbf{S}_{v_j}) \in [0, 1] \tag{8}$$

where $\mathbf{S}_{v_i}$, $\mathbf{S}_{v_j}$ are the street view feature vectors of place v_i and place v_j, respectively. In this work, ResNet-101 is used as feature extractor, generating a 2048-dimensional feature for each street view image. As shown in Fig. 1, distinct variations can be observed among areas with different landscapes, such as the historic site and business area. We input the street view images into the PSPNet [57] with the ResNet-101 as backbone, which has been pretrained on cityscapes dataset [58]. The final street view feature vectors are equal to the average of all features within the same place.

3.3.2 Spatial correlation modeling

With the constructed graphs, we apply the multi-graph convolution to model the spatial relationship as defined in Equation 9.

$$\mathbf{X}_{l+1} = \bigcup_{\mathbf{A} \in \bar{\mathbf{A}}} (\sigma f(\mathbf{A}; \theta_i) \mathbf{X}_l \mathbf{W}_l) \tag{9}$$

where $\mathbf{X}_l \in R^{|V| \times P_l}$, $\mathbf{X}_{l+1} \in R^{|V| \times P_{l+1}}$ are the feature vectors of $|V|$ places in layer l and $l + 1$, respectively. σ represents the activation function and $\bigcup$ denotes the aggregation function such as sum, max and average. $\bar{\mathbf{A}}$ represents the set of graphs, and $f(\mathbf{A}; \theta_i) \in R^{|V| \times |V|}$ represents the aggregation matrix based on graph $\mathbf{A} \in \bar{\mathbf{A}}$ parameterized by θ_i. $\mathbf{W}_l \in R^{P_l \times P_{l+1}}$ is the transformation matrix from layer l and $l + 1$.

In this work, we aggregate the multiple graph's convolution results after the activation function, while in ST-MGCN the linear results are aggregated before the activation function. This modification keeps the integrity of different correlation graphs better. The aggregation matrix $f(\mathbf{A}; \theta_i) \in R^{|V| \times |V|}$ is chosen to be the K-order polynomial function of the graph Laplacian $\mathbf{L}$.

The intricacy of polynomial form transformation not only lies in the parameter reduction. It takes full advantage of the real symmetric positive semidefinite property of the normalized graph Laplacian matrix to minimize the computing complexity. After a series of simplification, we can skip the eigen decomposition and use $\mathbf{L}$ to compute the convolution directly.

The polynomial also allows the spectral-based method to be spatial localized. It enables the parameter sharing in graph convolution to follow a local stationary pattern, which is the same as the property of graph-structured data [59]. Similar to the kernel size in CNN, k defines the size of receptive field in graph convolution. An example is given in Fig. 2. Taking vertex 1 as the centralized region, when the maximum degree of graph Laplacian K is set to 1, only the information of one-hop neighbors, colored in yellow, will be aggregated. The corresponding entry of the convolution operational matrix will be nonzero and share the same parameter. When K is set to 2, the extent of spatial feature extraction will expand to the two-hop neighbors, colored in green. All the two-hop neighbors share the same parameter, but different from those in one-hop neighborhood.

Multi-graph convolution models spatial correlation in a more flexible way due to the diversity of the correlation. Vertexes can be connected based on not only the real geographical location, but also relatively abstract features. The expression of connection can be qualitative, e.g., 0 or 1 in neighborhood graph, or quantitative such as using the
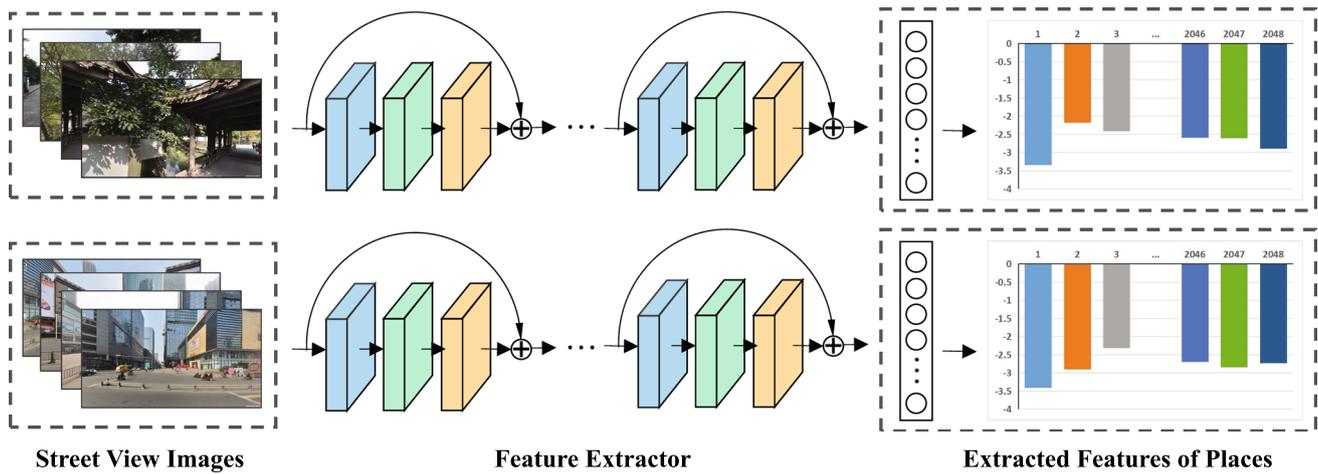
Fig. 1 Extracting landscape features from street view images
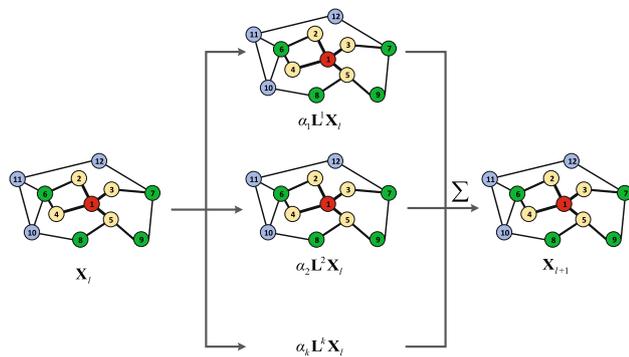
Fig. 2 An example of ChebNet graph convolution centralized at the red vertex. Left: The centralized vertex is colored red. The one-hop neighbors are colored yellow and the two-hop neighbors are colored green. Middle: When the maximum degree of graph Laplacian increases, the reception field will grow based on the hop counts. Right: The convolution result equals to a sum among graph transformations with degree value from 1 to K. The figure only shows the operation to vertex 1. In practical convolution, the same operation will be applied to all vertexes

similarity function in functional similarity graph. We take the temporal embedded features as input to the multi-graph convolution instead of the other way around. It's more logical to emphasize the spatial dependencies based on the temporal features. In this way, the demand values can be aggregated through multi-graph correlations and therefore improve the prediction results.

3.3.3 Temporal correlation modeling

We introduce the dual temporal gated branches to model the temporal dependencies among historical observations in previous timestamps and periodic timestamps, respectively, and then integrate the encoded results. Both branches are based on the contextual gated recurrent neural network (CGRNN) in ST-MGCN (Fig. 3). CGRNN

focuses on the global contextual information in temporal dimension. It captures the context by gating mechanism, i.e., a reweighting of the original sequence. Assuming that there are T temporal observations and $\mathbf{X}^{(t)} \in R^{|V| \times P}$ denotes the t-th observation, where P is the dimensionality of the feature. P will be 1 if the feature only contains the number of orders. The workflow of contextual gating mechanism will be described below.

Firstly, the historical data with its neighborhood information is concatenated to generate region descriptions, which is regarded as contextual information. The information aggregation is also obtained by a graph convolution operation $F_G^{K'}$ with max degree K' using the corresponding graph Laplacian matrix.

$$\hat{\mathbf{X}}^{(t)} = [\mathbf{X}^{(t)}, F_G^{K'}(\mathbf{X}^{(t)})] \quad t = 1, 2, \ldots T \quad (10)$$

Secondly, the global average pooling F_{pool} is used over all regions to produce the summary of each temporal observation. It further aggregates the contextual information within each timestamp (Eq. 11).

$$\mathbf{z}^{(t)} = F_{pool}(\hat{\mathbf{X}}^{(t)}) = \frac{1}{|V|} \sum_{t=1}^{|V|} \hat{\mathbf{X}}_{i,:}^{(t)} \quad t = 1, 2, \ldots T \quad (11)$$

With the summarized vector z, an attention operation (Eq. 12) is applied, where $\mathbf{W}_1$ and $\mathbf{W}_2$ are trainable weights, δ and σ denote the ReLU and sigmoid function, respectively.

$$\mathbf{s} = \sigma(\mathbf{W}_2 \delta(\mathbf{W}_1 z)) \quad (12)$$

Finally, s serves as a reweighting factor to the original historical input (Eq. 13), where ∘ denotes dot product.

$$\tilde{\mathbf{X}}^{(t)} = \mathbf{X}^{(t)} \circ \mathbf{s}^{(t)} \quad (13)$$

After the contextual gating, a shared RNN layer with

Fig. 3 Temporal correlation modeling of DTG-MGCN. (a)The framework of dual temporal branches. We select the previous observations and periodic observations from historical sequence as input to CGRNN, respectively. And the resulting features are combined through weighted sum. (b) The detailed structure of CGRNN. It first generates the place descriptions using global pooling over the historical observations and its neighborhood information. Then, it turns the extracted contextual information into a summarized vector z. The gated mechanism is operated by reweighting the original input sequence with z. Finally, the gated sequence is fed to a shared LSTM over all places

(a)

(b)
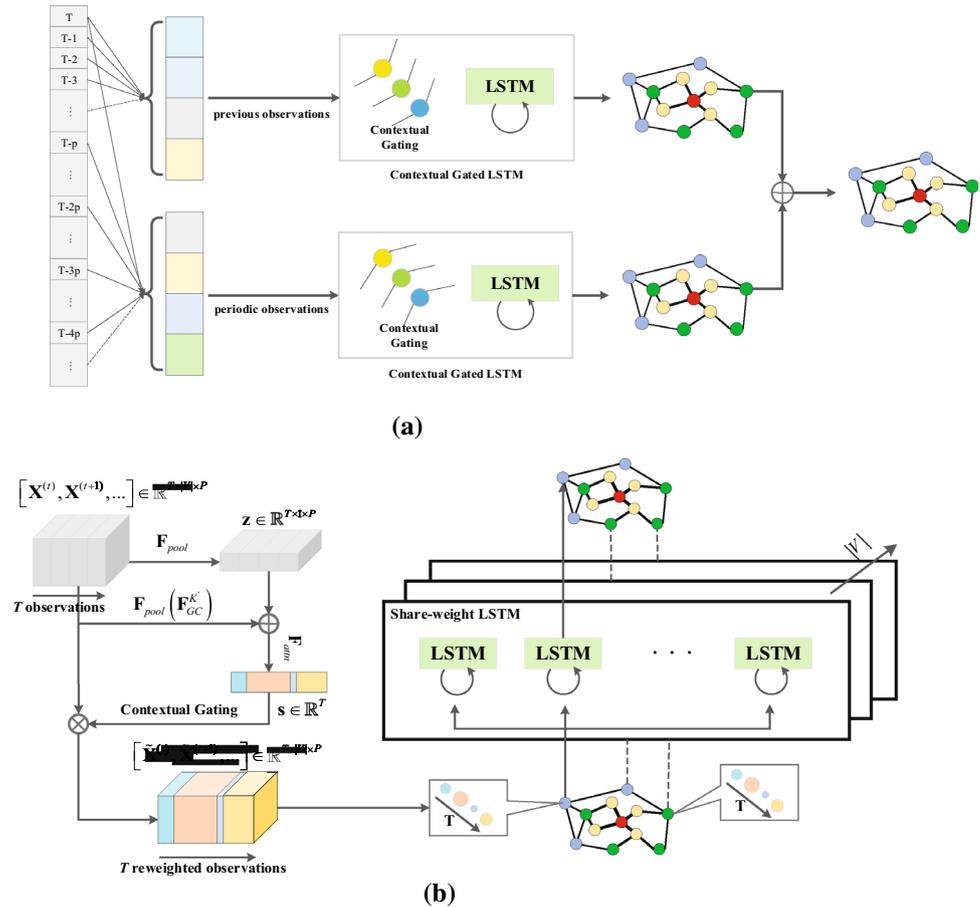
weight $\mathbf{W}_3$ across all regions is applied to encode the gated sequence in different timestamps of a region into a single vector $\mathbf{H}_{i,:}$. The basic idea of RNN is to recursively combine the current historical observation with the latest hidden state through a series of nonlinear operations. In this implementation, we choose to use long short-term memory network (LSTM), a variant of RNN, to better capture the global dependencies.

$$\mathbf{f}^{(t)} = \sigma(\mathbf{W}_f[\mathbf{h}^{(t-1)}, \mathbf{x}^{(t)}] + \mathbf{b}_f)$$

$$\mathbf{i}^{(t)} = \sigma(\mathbf{W}_i[\mathbf{h}^{(t-1)}, \mathbf{x}^{(t)}] + \mathbf{b}_i)$$

$$\mathbf{c}^{(t)} = \mathbf{f}^{(t)} \circ \mathbf{c}^{(t-1)} + \mathbf{i}^{(t)} \circ tanh(\mathbf{W}_c[\mathbf{h}^{(t-1)}, \mathbf{x}^{(t)}] + \mathbf{b}_c)$$

$$\mathbf{o}^{(t)} = \sigma(\mathbf{W}_o[\mathbf{h}^{(t-1)}, \mathbf{x}^{(t)}] + \mathbf{b}_o)$$

$$\mathbf{h}^{(t)} = \mathbf{o}^{(t)} tanh(\mathbf{c}^{(t)})$$

$$(14)$$

where σ is sigmoid function. $\mathbf{i}, \mathbf{f}, \mathbf{o}$ and $\mathbf{c}$ are input gate, forget gate, output gate and hidden cell state, respectively, parameterized with corresponding weights $\mathbf{W}$ and bias $\mathbf{b}$. Equation 14 can be further simplified as follows.

$$\mathbf{h}^{(t)} = LSTM(\mathbf{x}^t, \mathbf{h}^{t-1}) \tag{15}$$

Therefore, the generation of $\mathbf{H}_{i,:}$ through LSTM can be expressed by Eq. 16.

$$\mathbf{H}_{i,:}^G = LSTM(\tilde{\mathbf{X}}_{i,:}^{(t)}, \tilde{\mathbf{X}}_{i,:}^{(t+1)}, \tilde{\mathbf{X}}_{i,:}^{(t+2)} \\ \dots, \tilde{\mathbf{X}}_{i,:}^{(T-1)}, \tilde{\mathbf{X}}_{i,:}^{(T)}; \mathbf{W}_3) \tag{16}$$

Taking previous observations as input, LSTM can well represent the data continuity in temporal dimension. However, this learning mechanism may not fit to the taxi demand data completely for its strong periodicity. To fully demonstrate the data characteristics, we choose six days' transactions from Didi Chuxing dataset and plot the hourly taxi demand from 7:00 a.m. to 23:00 p.m. in the heart of Chengdu, China (Fig. 4). The practical demand trend not only reflects the correlation among adjacent timestamps, but also presents periodic change. The periodic interval can be diverse. When the interval is a day, i.e., 24 h, we can observe that demand values are quite similar at the same time point every day, such as the rush hours and the off-peak hours. When the interval is set to a week, the demand trends share even more similarity. For example, on Saturdays (Nov 12th and Nov 19th, 2016), the taxi demands at
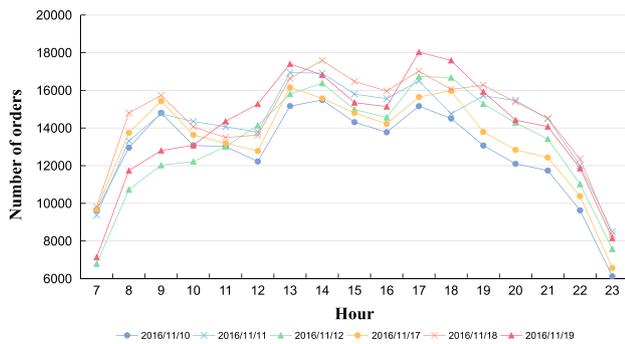
Fig. 4 The number of taxi demands w.r.t. different hours and days

normal morning peak around 8:00 a.m. are clearly lower than those on Thursdays and Fridays. This is because Saturday is the rest day for most people. While at the evening peak around 17:00 p.m., people are likely to go out for recreational activities, the request orders are slightly more than those on weekdays. Apparently, the observations from periodic timestamps play an important role in expressing the dynamic patterns of taxi demands. Therefore, we add another CGRNN branch taking periodic data as input.

$$
\begin{aligned}
\mathbf{H}_{i,:}^{PG} = LSTM(&\tilde{\mathbf{X}}_{i,:}^{(t')}, \tilde{\mathbf{X}}_{i,:}^{(t'+p)}, \tilde{\mathbf{X}}_{i,:}^{(t'+2p)}, \\
&\ldots, \tilde{\mathbf{X}}_{i,:}^{(T-p)}, \tilde{\mathbf{X}}_{i,:}^{(T)}; \mathbf{W}_4)
\end{aligned}
\tag{17}
$$

where p is the periodic interval, which can be set as a day, a week and so on according to the data characteristics.

Subsequently, we need to produce the final temporal encoded feature $\mathbf{H}_{i,:}^{D}$ by merging the output from dual branches.

$$
\mathbf{H}_{i,:}^{D} = \mathbf{W}^{G}\mathbf{H}_{i,:}^{G} + \mathbf{W}^{PG}\mathbf{H}_{i,:}^{PG} + \mathbf{b}
\tag{18}
$$

4 Experiment

4.1 Dataset

We perform experiments on two open real-life ride request datasets from leading mobile transportation platform Didi Chuxing[1] in two cities, Chengdu and Haikou, China. Both datasets include the order ID, the start time and stop time of the ride and the geographic coordinates of the pick-up and drop-off location (Tables 1, 2), with a slight difference in field format and field type. The Chengdu dataset is collected from November 1st to November 30th in urban area. We select the last week of a month as the test set and the rest is training set. The last 10% of the training set is used for validation. The Haikou dataset ranges from May 1st to

October 31st in 2017. Data from May 1st to August 31st is for training, from September 1st to September 30th is for validation, and from October 1st to October 31st is for testing. The length of the time interval is set as half an hour. We define study area as the urban area of the two cities according to the data distribution.

The POI dataset used for places definition and correlation modeling is collected from Baidu, a major source of location data. The functional categories contain canteen, company, financial services, and so on. We call the Geocoder API to get the business area that each point belongs to, most of which are the well-known place names. The obtained place areas are considered as basic study units.

Available street view images can be categorized as static images and panorama. Due to the serious distortion of panorama, we collected a total of 111,008 static street view images from Tencent, a widely accepted service provider in China. The locations of street view images lie along the streets and do not distribute at same interval. They dynamically change with the street density and the imaging conditions of moving cars. Therefore, traditional sampling at regular intervals will cause data loss or data duplication. To solve this problem, we adopted grid searching method to collect a set of ID of street view images without repetition. Four images were downloaded at every location from different horizontal viewing angles, i.e., $0°, 90°, 180°$ and $270°$, each with a resolution of 608×1110 (Fig. 5).

4.2 Experimental settings

In the experiment of identifying places' footprints, the coefficient k in adaptive kernel density estimation is set to 5. Divergence s in fuzzy method is 3, and middle point m is equal to 0.1 for Chengdu dataset and 0.3 for Haikou dataset, respectively. The places' boundaries are delineated with a membership threshold as 0.5.

To better explore the model without auxiliary data, the input of DTG-MGCN is only the taxi demand at corresponding time point. In the multi-graph convolution, $f(\mathbf{A}; \theta_i)$ in Eq. 9 is chosen as the Chebyshev polynomial function of the graph Laplacian in ChebNet [47] with the degree K being 2 and the time complexity being O(n). Aggregation function $\bigcup$ is chosen to be the sum function. In the temporal gated branches, the graph convolution degree K' is set to 1. We take the previous 8 time intervals, i.e., 4 hours, as the previous timestamps input. The periodic interval is set to one day for Chengdu and one week for Haikou according to their data size. The number of hidden layer is 3, with 64 hidden units each. We also apply an L_2 regularization with a weight decay equals being 1e-4 for each layer.

ReLU is used as the activation in the network, and the learning rate of DTG-MGCN is set to 1e-4. The network is

Table 1 Data structure of Chengdu ride request data

Field	Type	Sample	Comment
Order ID	String	mjiwdgkqmonDFvCk3ntpron5mwfrqvI	Anonymized
Ride start time	Int	1501581031	Unix timestamp, in seconds
Ride stop time	Int	1501582195	Unix timestamp, in seconds
Pick-up longitude	Float	104.11225	GCJ-02 coordinate system
Pick-up latitude	Float	30.66703	GCJ-02 coordinate system
Drop-off longitude	Float	104.07403	GCJ-02 coordinate system
Drop-off latitude	Float	30.68630	GCJ-02 coordinate system

Table 2 Data structure of Haikou ride request data

Field	Type	Sample	Comment
order_id	Int	17592719043682	Anonymized
arrive_time	String	2017-05-19 01:09:12	Date and time format as YYYY-MM-DD HH:MM:SS
departure_time	String	2017-05-19 01:05:19	Date and time format as YYYY-MM-DD HH:MM:SS
starting_lng	Float	110.3665	GCJ-02 coordinate system
starting_lat	Float	20.0059	GCJ-02 coordinate system
dest_lng	Float	110.3645	GCJ-02 coordinate system
dest_lat	Float	20.0353	GCJ-02 coordinate System
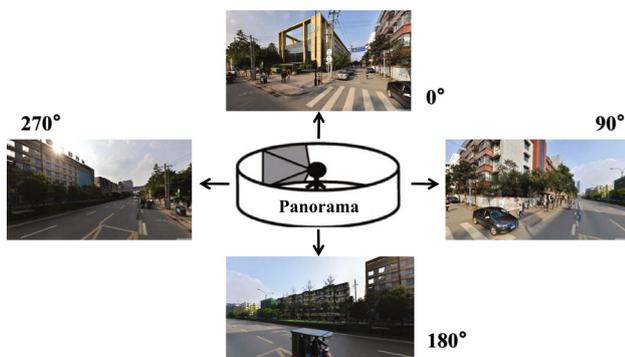
Fig. 5 The acquisition of street view images

trained using the Adam [60] optimizer for minimizing root-mean-square error (RMSE). We implement the model by Tensorflow [61] on Python 3.6, based on the existing benchmark GCN [62]. The experiments were conducted on Intel Core i7-8700K CPU and a single NVIDIA GeForce GTX 1070 Ti. The training of DTG-MGCN took 220MB RAM and 7GB GPU memory. It took 1 h to train on Chengdu dataset and 5.5 h to train on Haikou dataset, respectively.

4.3 Visualization of extracted places

The extraction results are shown in Fig. 6. We can see that the extraction method can deal with multi-place point sets properly and possess excellent adaptability to diverse point distribution. It avoids the oversimplification, meanwhile gives a crisp boundary, and provides a better understanding of the places' footprints under commercial context.

We extracted 166 places in Chengdu and 63 places in Haikou. The extracted places in Chengdu are concentrated at the center of the city, and places around the edges are relatively larger. Places distribute in a more organized pattern in Haikou. Small overlap exists among the places because extents of some business areas are originally set to be overlapped by map provider. These delineated polygons cover the urban areas where names are broadly known and used in the locals' daily lives, including residential area, business area, transportation hubs, historic sites with their surrounding area and so on.

4.4 Performance evaluation

4.4.1 Evaluation metric

We use root-mean-square error (RMSE) and mean absolute percentage error (MAPE)[2] to evaluate the performance of our model, which are defined as follows.

$$RMSE = \sqrt{\frac{1}{N}\sum_{t=1}^{N}\left(X^{(t)} - \hat{X}^{(t)}\right)^2} \tag{19}$$

$$MAPE = \frac{1}{N}\sum_{t=1}^{N}\left|\frac{X^{(t)} - \hat{X}^{(t)}}{X^{(t)}}\right| \tag{20}$$

where $X^{(t)}$ and $\hat{X}^{(t)}$ denote the real value and prediction value at timestamp t, and N represents the total number of samples.

---

[2] Following the practice in [32], the samples with demand values less than 10 are filtered when computing MAPE.

(a) Extracted places in Chengdu

(b) Extracted places in Haikou
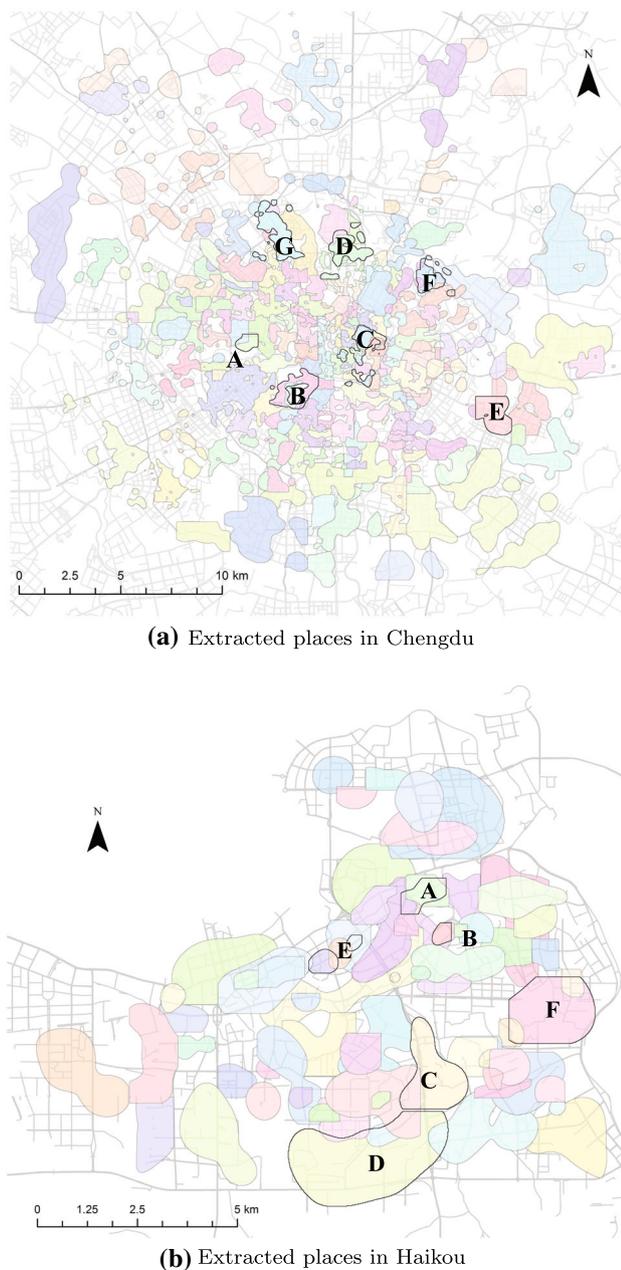
Fig. 6 Results of place identification. **a** The 166 places in Chengdu metropolitan area. Some representative places are highlighted, including historic sites **a** Du Fu's thatched cottage and **b** Wuhou Temple, business areas **c** Chunxi Road, transport hubs (D) North Station and **e** East Station, residential areas **f** Balizhuang and university **g** Southwest Jiaotong University. **b** The 63 places in Haikou urban area, including business areas **a** Jiefang West Road and **b** Pearl Square, residential areas **c** Longkun South Road and **d** Chengxi Road, and commercial districts **e** Guomao and **f** Guoxing

4.4.2 Methods for evaluation

We compare the proposed model (DTG-MGCN) with the following baselines.

Historical Average (HA) [63]: Historical average predicts the demand using the mean of the historical observations in the relative same time interval, i.e., the same time at a day.

LASSO and Ridge [64]: Linear regression takes the previous demand at different timestamps as input. We considered different versions of linear regression, including LASSO with L_1 regularization and Ridge regression with L_2 regularization.

Gradient Boosting Machine (GBM): LightGBM [65] is a gradient boosting framework that uses tree-based learning algorithm. It is designed with high efficiency for large-scale data. We set the number of trees as 50, the maximum depth as 4 and the learning rate as 2e-3.

LSTM [66]: As a variant of RNN, LSTM introduces several gates to further control the flow of information and allows the recurrent layer to capture the long-term dependencies.

Long- and short-term time-series network (LSTNet) [67]: LSTNet is a multivariate time-series prediction model, which combines both CNN and LSTM to memorize the historical information and leverages the traditional autoregressive model to tackle the scale-insensitive problem of the neural network.

ST-MGCN [34]: The network combines the multi-graph convolution with contextual temporal learning for spatiotemporal taxi demand prediction.

Trying best to ensure all methods operate under the same condition, we also consider both the previous and periodic data as independent variable in linear regression and GBM. All methods are run five times, and the best performances are reported in the results.

4.4.3 Performances analysis

Table 3 shows the quantitative results of different forecasting methods on test set. We can obtain the following observations from the table. (1) Linear regression and almost all machine learning methods outperform the HA method. However, LSTM has poor performance on Chengdu dataset with only previous observations as input. We can infer that the periodic pattern is important for predicting Chengdu taxi demand. (2) When considering dual temporal inputs, the advantage of traditional machine learning over linear regression is not so obvious. The linear regression reaches the same level of performance as GBM on both ride request datasets and achieves the best RMSE on Chengdu dataset. Besides, the poor performance of LSTNet indicates that the multivariate time-series prediction strategy is not suitable for multi-place taxi demand prediction. (3) DTG-MGCN outperforms other methods with graph-based deep learning framework, especially on large-scale Haikou dataset. Recalling the original

experiment of ST-MGCN was also implemented on large dataset ranging from March 1st to December 31st in 2017 [34], this may enlighten us that the learning-based method needs considerable data scale to support the parameter size and improve the robustness. Besides, with an additional periodic branch, DTG-MGCN also has prediction improvement compared to ST-MGCN.

Effect of periodic input To further explore the effect of the periodicity of data, we input data in three different temporal patterns to linear regression and GBM, namely previous-only (P1), periodic-only (P2) and previous-periodic (P3). Based on the structure of ST-MGCN and DTG-MGCN, we can infer the same pattern for MGCN-based methods. ST-MGCN with previous data input corresponds to P1. We modify temporal input of ST-MGCN as periodic data for P2 as well, denoted by ST-MGCN(P2). DTG-MGCN with dual temporal branches matches P3. The periodic intervals keep the same, i.e., a day for Chengdu dataset and a month for Haikou dataset. Comparison results of both datasets are shown in Tables 4 and 5.

We can observe that the dependencies between input data pattern and the prediction results vary in different datasets. Comparing the P1 and P2 pattern on Chengdu dataset, all methods tend to achieve better performance on periodic-only pattern, which demonstrates strong data periodicity. However, on Haikou dataset, more reliable results are generated through the previous data input and it seems less helpful to use periodic data only.

Although the datasets possess different properties, it is clear that all methods with combination input (P3) gain the best results. It shows the necessity of considering both the previous data and periodic data simultaneously in taxi demand prediction. This kind of combination enhances the robustness of the model, no matter the dataset are more independent on data at previous or periodic timestamps. In relatively simple models such as linear regression, we can

turn the combination into parallel independent variables. And in more sophisticated models, like MGCN-based method, dual temporal branches in DTG-MGCN are more helpful to capture the temporal features accurately.

In addition, we can observe that MGCN-based methods can always outperform over all patterns, which further proves their stability at various data scales.

Effect of different periodic intervals We further investigate the effect of choosing different periodic intervals. To ensure adequate data for training, we only conducted the comparison on Haikou dataset. From intuitive knowledge, if we want to predict the taxi demand at 10:00 a.m. Friday, it is more likely to have a closer result by referring to the demand at 10:00 a.m. on weekdays, especially last Fridays. Therefore, we mainly chose two periodic intervals, i.e., a day and a week. Specifically, historical observations were input to DTG-MGCN at the same time every day within a week (e.g., 10:00 a.m. from last Thursday to this Thursday), or the same time on the same day of the week within a month (e.g., 10:00 a.m. on every Friday within four weeks). We also tried to remove the previous temporal branch, as the P2 input pattern mentioned above, to see how different periodic intervals work independently, indicated as P2-day and P2-week. Experiment results are shown in Table 6.

We can see that periodic-only pattern does not work well on Haikou dataset. However, with the dual temporal gated branches, the error decreases greatly. When the periodic interval is set as a week, the model performs best, which indicates that data at the same time on "every Friday" have more significance to the prediction.

Effect of multi-graph construction To study the effect of multi-graph construction in spatial correlation modeling, several variants of DTG-MGCN were evaluated by removing different graphs from the model. Considering the neighborhood graph is the basic correlation of graph modeling in geographical study, we mainly remove the other two graphs. The results are shown in Table 7. Both the functional similarities graph and the landscape similarities graph have positive effects to improve the predictions. Multi-graph construction successfully encoded the region-wise correlation in different aspects.

4.5 Visualization of demand prediction results

In order to give intuitive presentation, we randomly selected a weekday in Chengdu dataset and depicted the prediction results at different timestamps in Fig. 7. The transition from blue to red denotes the taxi demand goes from low to high. From the figure, observations can be drawn as follows.

Figure 7a shows the demand during 7:00-7:30 in the morning. People tend to transit to work during this period.

Table 3 Performance comparison of different methods for taxi demand prediction

Method	Chengdu		Haikou	
	RMSE	MAPE(%)	RMSE	MAPE(%)
HA	9.81	20.45	19.17	26.97
LASSO	8.22	18.55	15.51	28.06
Ridge	**8.21**	18.58	15.51	28.11
GBM	8.31	18.04	15.28	27.54
LSTM	12.04	25.35	16.61	30.18
LSTNet	9.76	22.09	16.36	28.39
ST-MGCN	9.27	20.55	14.98	25.45
DTG-MGCN	8.58	**17.59**	**14.50**	**24.76**

Table 4 Effect of periodic input on Chengdu dataset

Method	RMSE P1	MAPE(%)	RMSE P2	MAPE(%)	RMSE P3	MAPE(%)
LASSO	13.49	27.45	8.46	18.92	8.22	18.55
Ridge	13.48	27.59	8.45	18.95	8.21	18.58
GBM	12.41	26.09	8.63	18.31	8.31	18.04
	ST-MGCN		ST-MGCN(P2)		DTG-MGCN	
MGCN-based method	9.27	20.55	8.83	18.07	**8.58**	**17.59**

Table 5 Effect of periodic input on Haikou dataset

Method	RMSE P1	MAPE(%)	RMSE P2	MAPE(%)	RMSE P3	MAPE(%)
LASSO	16.75	31.38	20.41	32.49	15.51	28.06
Ridge	16.76	31.43	20.42	32.52	15.51	28.11
GBM	16.51	30.20	20.06	31.88	15.28	27.54
	ST-MGCN		ST-MGCN(P2)		DTG-MGCN	
MGCN-based method	14.98	25.45	19.70	33.10	**14.50**	**24.76**

Table 6 Effect of different periodic intervals on Haikou dataset

Method	RMSE	MAPE(%)
ST-MGCN(P2-day)	19.75	29.63
ST-MGCN(P2-week)	19.70	33.10
DTG-MGCN(day)	17.90	29.11
DTG-MGCN(week)	**14.50**	**24.76**

Demands are low in most places except the business area Jianshe Road and residential area Sima Bridge.

Figure 7b shows the demand during 14:00-14:30 after lunch. Locals continue to leave for work or recreational activities. We can see the demands keep going higher in Jianshe Road and Sima Bridge; meanwhile, other places of the city are awaken. Demands in Chunxi Road, the well-known commercial pedestrian street, go higher sharply. And more request orders also appear in the area centered around Southwest Jiaotong University.

According to Fig. 7c, demands during 21:00–21:30 p.m. still remain a high level in Chunxi Road and Jianshe road. It can be inferred that people want to go home or relax themselves in entertainment areas after a long and difficult workday.

5 Conclusion

In this paper, we introduced a deep learning model, DTG-MGCN, to model the spatiotemporal dependencies for taxi demand prediction. Using a fuzzy set method based on adaptive kernel density estimation, we defined the study units as places with specific semantic and humanistic experiences. The proposed model encoded different non-Euclidean correlations with graphs and better utilized the data periodicity with dual temporal gated branches. Experiments on two real-word datasets showed the effectiveness of our model comparing to several baselines. Taxi demand forecasting is crucial to efficient distribution of traffic resources for future smart cities. For future work, we plan to investigate more aspects including (1) evaluate the model with more time-related auxiliary data, such as weather and holiday events; (2) explore more possibilities in the graph correlation types; (3) extend the model for taxi origin–destination prediction problem.

Table 7 Effect of multi-graph construction on Chengdu dataset

Method	RMSE	MAPE(%)
DTG-MGCN (neighborhood)	8.69	18.11
DTG-MGCN (neighborhood and functional similarities)	8.60	17.73
DTG-MGCN (neighborhood and landscape similarities)	8.58	18.06
DTG-MGCN (neighborhood, functional and landscape similarities)	**8.58**	**17.59**

(a) Predicted demand at 7:00-7:30 a.m.

(b) Predicted demand at 14:00-14:30 p.m.

(c) Predicted demand at 21:00-21:30 p.m.
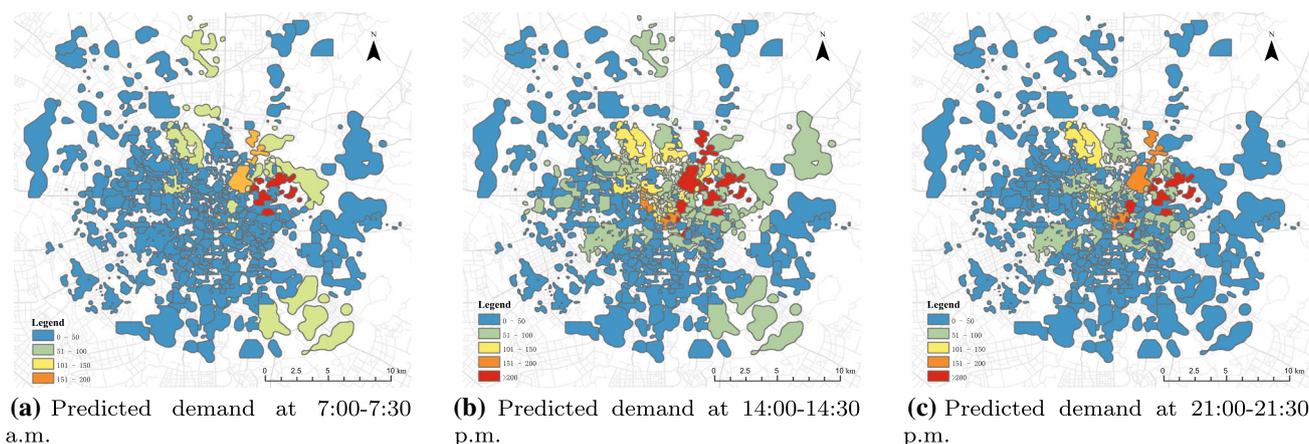
Fig. 7 The visualization of demand prediction

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Wang Y, Wu C, Zhu T (2019) Mobile hailing technology and taxi driving behaviors. Mark Sci 38(5):734–755
2. Chang H, Tai Y, Hsu JY (2010) Context-aware taxi demand hotspots prediction. Int J Bus Intell Data Min 5(1):3–18
3. Davis N, Raina G, Jagannathan K (2016) A multi-level clustering approach for forecasting taxi travel demand. In: 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), IEEE, pp 223–228
4. Li Y, Lu J, Zhang L, Zhao Y (2017) Taxi booking mobile app order demand prediction based on short-term traffic forecasting. Transp Res Rec 2634(1):57–68
5. Kuang L, Yan X, Tan X, Li S, Yang X (2019) Predicting taxi demand based on 3d convolutional neural network and multi-task learning. Remote Sens 11(11):1265
6. Liu Y, Liu Z, Lyu C, Ye J (2019) Attention-based deep ensemble net for large-scale online taxi-hailing demand prediction. IEEE Transactions on Intelligent Transportation Systems PP(99)
7. Zhang J, Zheng Y, Qi D (2017) Deep spatio-temporal residual networks for citywide crowd flows prediction. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol 31
8. Liao S, Zhou L, Di X, Yuan B, Xiong J (2018) Large-scale short-term urban taxi demand forecasting using deep learning. In: 2018 23rd Asia and South Pacific Design Automation Conference (ASP-DAC), IEEE, pp 428–433
9. Li X, Pan G, Wu Z, Qi G, Li S, Zhang D, Zhang W, Wang Z (2012) Prediction of urban human mobility using large-scale taxi traces and its applications. Front Comput Sci 6(1):111–121
10. Moreira-Matias L, Gama J, Ferreira M, Mendes-Moreira J, Damas L (2013) Predicting taxi-passenger demand using streaming data. IEEE Trans Intell Transp Syst 14(3):1393–1402
11. Tobler WR (1970) A computer movie simulating urban growth in the detroit region. Econ Geogr 46(sup1):234–240
12. Tong Y, Chen Y, Zhou Z, Chen L, Wang J, Yang Q, Ye J, Lv W (2017) The simpler the better: a unified approach to predicting original taxi demands based on large-scale online platforms. In: Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining, pp 1653–1662
13. Wang F, Jiang M, Qian C, Yang S, Li C, Zhang H, Wang X, Tang X (2017) Residual attention network for image classification. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3156–3164
14. Shi Q, Liu M, Liu X, Liu P, Li X (2019) Domain adaption for fine-grained urban village extraction from satellite images. IEEE Geosci Remote Sens Lett 17(8):1–5
15. Liu S, Shi Q (2020) Local climate zone mapping as remote sensing scene classification using deep learning: A case study of metropolitan china. ISPRS J Photogramm Remote Sens 164:229–242
16. Firat O, Cho K, Bengio Y (2016) Multi-way, multilingual neural machine translation with a shared attention mechanism. arXiv preprint arXiv:160101073
17. Liu F, Xue S, Wu J, Zhou C, Hu W, Paris C, Nepal S, Yang J, Yu PS (2020) Deep learning for community detection: Progress, challenges and opportunities. arXiv preprint arXiv:200508225
18. Zhu Q, Du B, Yan P (2019) Boundary-weighted domain adaptive neural network for prostate mr image segmentation. IEEE Trans Med Imaging 39(3):753–763
19. Wang Z, Du B, Guo Y (2020) Domain adaptation with neural embedding matching. IEEE Trans Neural Netw Learn Syst 31(7):2387–2397
20. Dong Y, Liang T, Zhang Y, Du B (2020) Spectral-spatial weighted kernel manifold embedded distribution alignment for

remote sensing image classification. IEEE Trans Cybern. https://doi.org/10.1109/TCYB.2020.3004263

21. Elleuch M, Mezghani A, Khemakhem M, Kherallah M (2019) Clothing classification using deep cnn architecture based on transfer learning. In: International Conference on Hybrid Intelligent Systems, Springer, pp 240–248

22. Zhang Y, Wu K, Du B, Hu X (2019) Multitask learning-based reliability analysis for hyperspectral target detection. IEEE J Sel Top Appl Earth Obs Remote Sens 12(7):2135–2147

23. Liu S, Shi Q, Zhang L (2020) Few-shot hyperspectral image classification with unknown classes using multitask deep learning. IEEE Trans Geosci Remote Sens. https://doi.org/10.1109/TGRS.2020.3018879

24. Guo H, Shi Q, Du B, Zhang L, Ding H (2020) Scene-driven multitask parallel attention network for building extraction in high-resolution remote sensing images. IEEE Trans Geosci Remote Sens. https://doi.org/10.1109/TGRS.2020.3014312

25. Zhang Y, Liu K, Dong Y, Wu K, Hu X (2019) Semisupervised classification based on slic segmentation for hyperspectral image. IEEE Geosci Remote Sens Lett. https://doi.org/10.1109/LGRS.2019.2945546

26. Du B, Ru L, Wu C, Zhang L (2019) Unsupervised deep slow feature analysis for change detection in multi-temporal remote sensing images. IEEE Trans Geosci Remote Sens 57(12):9976–9992

27. Li J, Han Z, Cheng H, Su J, Wang P, Zhang J, Pan L (2019) Predicting path failure in time-evolving graphs. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp 1279–1289

28. Yao H, Tang X, Wei H, Zheng G, Yu Y, Li Z (2018) Modeling spatial-temporal dynamics for traffic prediction. arXiv preprint arXiv:180301254

29. Yu R, Li Y, Shahabi C, Demiryurek U, Liu Y (2017) Deep learning: A generic approach for extreme condition traffic forecasting. In: Proceedings of the 2017 SIAM international Conference on Data Mining, SIAM, pp 777–785

30. Xu J, Rahmatizadeh R, Bölöni L, Turgut D (2017) Real-time prediction of taxi demand using recurrent neural networks. IEEE Trans Intell Transp Syst 19(8):2572–2581

31. Vanichrujee U, Horanont T, Pattara-atikom W, Theeramunkong T, Shinozaki T (2018) Taxi demand prediction using ensemble model based on rnns and xgboost. In: 2018 International Conference on Embedded Systems and Intelligent Technology & International Conference on Information and Communication Technology for Embedded Systems (ICESIT-ICICTES), IEEE, pp 1–6

32. Yao H, Wu F, Ke J, Tang X, Jia Y, Lu S, Gong P, Ye J, Li Z (2018) Deep multi-view spatial-temporal network for taxi demand prediction. In: Thirty-Second AAAI Conference on Artificial Intelligence

33. Lai Y, Zhang K, Lin J, Yang F, Fan Y (2019) Taxi demand prediction with lstm-based combination model. In: 2019 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom), IEEE, pp 944–950

34. Geng X, Li Y, Wang L, Zhang L, Yang Q, Ye J, Liu Y (2019) Spatiotemporal multi-graph convolution network for ride-hailing demand forecasting. Proceedings of the AAAI Conference on Artificial Intelligence 33:3656–3663

35. Wu J, Zhu X, Zhang C, Cai Z (2013) Multi-instance multi-graph dual embedding learning. In: 2013 IEEE 13th International Conference on Data Mining, IEEE, pp 827–836

36. Wu J, Hong Z, Pan S, Zhu X, Zhang C, Cai Z (2014) Multi-graph learning with positive and unlabeled bags. In: Proceedings of the 2014 SIAM International Conference on Data Mining, SIAM, pp 217–225

37. Wu J, Pan S, Zhu X, Cai Z (2014) Boosting for multi-graph classification. IEEE Trans Cybern 45(3):416–429

38. Wu J, Zhu X, Zhang C, Philip SY (2014) Bag constrained structure pattern mining for multi-graph classification. IEEE Trans Knowl Data Eng 26(10):2382–2396

39. Agnew JA, Duncan JS (2014) The power of place (RLE Social & Cultural Geography): Bringing together geographical and sociological imaginations. Routledge

40. Goodchild MF (2011) Formalizing place in geographic information systems. In: Communities, neighborhoods, and health, Springer, pp 21–33

41. Zhu D, Zhang F, Wang S, Wang Y, Cheng X, Huang Z, Liu Y (2020) Understanding place characteristics in geographic contexts through graph convolutional neural networks. Annal Am Assoc Geogr 110(2):408–420

42. Wei H, Wang Y, Wo T, Liu Y, Xu J (2016) Zest: a hybrid model on predicting passenger demand for chauffeured car service. In: Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, pp 2203–2208

43. Ke J, Zheng H, Yang H, Chen XM (2017) Short-term forecasting of passenger demand under on-demand ride services: a spatio-temporal deep learning approach. Transp Res Part C Emerg Technol 85:591–608

44. Cui Z, Henrickson K, Ke R, Wang YH (2019) Traffic graph convolutional recurrent neural network: a deep learning framework for network-scale traffic learning and forecasting. IEEE Trans Intell Transp Syst. https://doi.org/10.1109/TITS.2019.2950416

45. Yu B, Yin H, Zhu Z (2017) Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. arXiv preprint arXiv:170904875

46. Wu Z, Pan S, Chen F, Long G, Zhang C, Philip SY (2020) A comprehensive survey on graph neural networks. IEEE Trans Neural Netw Learn Syst. https://doi.org/10.1109/TNNLS.2020.2978386

47. Defferrard M, Bresson X, Vandergheynst P (2016) Convolutional neural networks on graphs with fast localized spectral filtering. In: Advances in neural information processing systems, pp 3844–3852

48. Li X, Cai BY, Ratti C (2018) Using street-level images and deep learning for urban landscape studies. Landscape Archit Front 6(2):20–30

49. Li X, Ratti C, Seiferling I (2017) Mapping urban landscapes along streets using google street view. In: International cartographic conference, Springer, pp 341–356

50. Li X, Santi P, Courtney TK, Verma SK, Ratti C (2018) Investigating the association between streetscapes and human walking activities using google street view and human trajectory data. Trans GIS 22(4):1029–1044

51. Gong FY, Zeng ZC, Zhang F, Li X, Ng E, Norford LK (2018) Mapping sky, tree, and building view factors of street canyons in a high-density urban environment. Build Environ 134:155–167

52. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778

53. Shamai S (1991) Sense of place: an empirical measurement. Geoforum 22(3):347–358

54. Adams B, McKenzie G (2013) Inferring thematic places from spatially referenced natural language descriptions. In: Crowdsourcing geographic knowledge, Springer, pp 201–221

55. Wang S, Liu Y, Chen Z, Shi L, Zhang J (2018) Representing multiple urban places' footprints from dianping. com data. Acta Geod Cartogr Sin 47:1105–1113

56. Silverman BW (1986) Density estimation for statistics and data analysis. CRC Press, Boca Raton

57. Zhao H, Shi J, Qi X, Wang X, Jia J (2017) Pyramid scene parsing network. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2881–2890

58. Cordts M, Omran M, Ramos S, Rehfeld T, Enzweiler M, Benenson R, Franke U, Roth S, Schiele B (2016) The cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3213–3223

59. Girault B, Narayanan SS, Ortega A (2017) Towards a definition of local stationarity for graph signals. 2017 IEEE International Conference on Acoustics. Speech and Signal Processing (ICASSP), IEEE, pp 4139–4143

60. Kingma DP, Ba J (2014) Adam: A method for stochastic optimization. arXiv preprint arXiv:14126980

61. Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, Corrado GS, Davis A, Dean J, Devin M, et al. (2016) Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:160304467

62. Kipf TN, Welling M (2016) Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:160902907

63. Qiu Z, Liu L, Li G, Wang Q, Xiao N, Lin L (2019) Taxi origin-destination demand prediction with contextualized spatial-temporal network. In: 2019 IEEE International Conference on Multimedia and Expo (ICME), IEEE, pp 760–765

64. Melkumova L, Shatskikh SY (2017) Comparing ridge and lasso estimators for data analysis. Procedia Eng 201:746–755

65. Ke G, Meng Q, Finley T, Wang T, Chen W, Ma W, Ye Q, Liu TY (2017) Lightgbm: A highly efficient gradient boosting decision tree. In: Advances in neural information processing systems, pp 3146–3154

66. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780

67. Lai G, Chang WC, Yang Y, Liu H (2018) Modeling long-and short-term temporal patterns with deep neural networks. In: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, pp 95–104

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.