


Article

# Predicting User Activity Intensity Using Geographic Interactions Based on Social Media Check-In Data

Jing Li <sup>1</sup>, Wenye Guo <sup>2</sup>, Haiyan Liu <sup>1,\*</sup>, Xin Chen <sup>2</sup>, Anzhu Yu <sup>2</sup>  and Jia Li <sup>1</sup>

<sup>1</sup> Institute of Data and Target Engineering, PLA Strategic Support Force Information Engineering University, Zhengzhou 450052, China; brandy12367@163.com (J.L.); lijia\_kk@163.com (J.L.)

<sup>2</sup> Institute of Geospatial Information, PLA Strategic Support Force Information Engineering University, Zhengzhou 450052, China; guowyer@163.com (W.G.); xinchen\_cosmos@126.com (X.C.); anzhu\_yu@126.com (A.Y.)

\* Correspondence: liuharry2020@163.com

**Abstract:** Predicting user activity intensity is crucial for various applications. However, existing studies have two main problems. First, as user activity intensity is nonstationary and nonlinear, traditional methods can hardly fit the nonlinear spatio-temporal relationships that characterize user mobility. Second, user movements between different areas are valuable, but have not been utilized for the construction of spatial relationships. Therefore, we propose a deep learning model, the geographical interactions-weighted graph convolutional network-gated recurrent unit (GGCN-GRU), which is good at fitting nonlinear spatio-temporal relationships and incorporates users' geographic interactions to construct spatial relationships in the form of graphs as the input. The model consists of a graph convolutional network (GCN) and a gated recurrent unit (GRU). The GCN, which is efficient at processing graphs, extracts spatial features. These features are then input into the GRU, which extracts their temporal features. Finally, the GRU output is passed through a fully connected layer to obtain the predictions. We validated this model using a social media check-in dataset and found that the geographical interactions graph construction method performs better than the baselines. This indicates that our model is appropriate for fitting the complex nonlinear spatio-temporal relationships that characterize user mobility and helps improve prediction accuracy when considering geographic flows.

**Keywords:** geographical interactions; graph convolutional network; gated recurrent unit; spatial adjacency; temporal granularity



**Citation:** Li, J.; Guo, W.; Liu, H.; Chen, X.; Yu, A.; Li, J. Predicting User Activity Intensity Using Geographic Interactions Based on Social Media Check-In Data. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 555. <https://doi.org/10.3390/ijgi10080555>

Academic Editor: Wolfgang Kainz

Received: 30 June 2021

Accepted: 15 August 2021

Published: 17 August 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Research on user movement is critical for various applications, including point-of-interest (POI) recommendations and location-based advertising [1]. It can aid the analysis of the traffic in a city, the functional area, and population activity distribution, which have important applications in traffic management [2,3], disasters and emergencies [4,5], tourism recommendations [6,7], and urban planning [8,9], among others. Recently, with the development of intelligent sensor equipment—based on GPS and other sensors—mobile devices have can determine human positions. On social platforms, such as Foursquare, people post texts and pictures and record their locations, leaving a large amount of spatio-temporal data related to daily life [10]. Large-scale spatio-temporal data record the moving processes of people across space and, thus, contain a variety of personal preferences and human life patterns, enabling researchers to examine user movement [11]. However, most studies have focused on predicting a user's next location through open-source geotagged data [2,12,13]. Attempting to precisely predict a location will result in low accuracy [2]. In several cases, it is not necessary to acquire an accurate location for individuals; regional predictions for users are also crucial. Dynamically predicting the changes in user population

across space could be helpful for various applications, such as spatio-temporal modeling of disease transmission [14] and advanced prevention of trampling [15].

User movements show structural patterns due to geographical and social constraints [16]. Several studies have extensively examined user mobility prediction. Li et al. [17] used taxi-tracking data to find the spatial temporal patterns of taxi passengers as well as predict the change in the activity intensity with an improved ARIMA model. Cho et al. [16] proposed a model considering social networks to predict user mobility based on social media data. Liang et al. [18] proposed a recurrent neural network to predict the density of people in metro stations based on streaming CDR data. These studies have attempted to predict the intensity and change in spatial activities. They focused on predictions of the changes in the user's activity intensity and periodicity [19]. However, the dynamic interactions of humans between regions is also an important factor [20,21]. Some researchers have begun to consider the interactions associated with the predictions of user mobility. Crivellari and Beinat [22] developed an embedding method for locations, traces, and visitors to calculate the relationships among them. Li et al. [23] used parallel convolution to capture spatial neighborhood interactions. Chen et al. [24] constructed an artificial neural network-based model using mobile phone location data to predict the urban population. Wang et al. [25] grouped adjacent regions in a partitioned study area (a city) based on similarities between their origin–destination (OD) flow patterns to obtain the functional zones of the city. Thereafter, they used kernel density estimation to forecast the origin–destination flows between all region pairs. These methods usually consider the spatial proximity relationship between the target area and the surrounding area as input into the prediction model; this, however, renders it difficult to choose an appropriate distance threshold from different distance ranges in the spatial interactions when learning the spatial patterns of the variations in user activity [26]. Consequently, the association between one location and another no longer decreases with distance in a simple manner [27,28]; in other words, distance has ceased to be an all-encompassing measure of spatial association [29]. The regional user activity intensity, for instance, can be affected by factors such as the traffic and venue function; a person may choose to travel to areas that are conveniently connected by transportation networks instead of more proximal venues. Therefore, the prediction results of these methods are not satisfactory and require further improvements.

Meanwhile, predicting the mobility of users is one of the core problems associated with spatio-temporal prediction [30]. These spatio-temporal relationships in user activities are complex and nonlinear. Therefore, fitting models to these complex spatio-temporal relationships poses a central challenge in this field. The spatio-temporal prediction method can be divided into two categories: statistical parameter models and machine learning-based nonparametric models [31]. Statistical parameter models, such as the autoregressive integrated moving average model (ARIMA), as used by Deveaud et al. [32] to construct time-series models of venue attendance, are based on parameters, but are limited by the stability of the data [32]. The effect of a prediction is not particularly good [24]. With technological advancement, especially the rapid development of deep learning, the prediction accuracy has significantly improved. Machine learning methods, such as the space-time support vector machine model (SVR) [33], and deep learning methods have been used in spatial-temporal predictions. However, the traditional machine learning method is restricted by the hypothesis of independent and distributed samples, which renders it difficult to describe the non-stationarity of spatio-temporal data. The ability of deep learning methods to fit discontinuous and nonlinear data can fully exploit the nonlinear features in spatio-temporal data. A convolutional neural network (CNN) can learn the spatial characteristics of a time-space series. In contrast, a recurrent neural network (RNN) and its variants, i.e., the long short-term memory (LSTM) and gated recurrent unit (GRU), can learn the time characteristics of a time-space series. For example, Zhang et al. [34] proposed a space-time residual network to predict the inflow and outflow of people in a certain region at a specific time. Ren et al. [35] introduced the LSTM to predict the volumes of citywide movement. However, deep learning methods may encounter two

major problems when dealing with this type of research. First, the space-time prediction method based on deep learning is mainly used in transportation, but less often in research on people's movement. There are essential differences between traffic prediction and user movements; therefore, a deep learning model should be devised to predict user behavior according to the actual research objectives. Second, predictions of user mobility are usually performed by dividing a geographic region into its basic elements, followed by constructing spatio-temporal relationships between these elements. Limited by the input of a deep learning model, the geographic divisions usually adopt regular grid forms [24,34,36]. Most spatial vector data are intrinsically irregular (e.g., partitioning based on road networks). However, irregular partitions have different numbers of neighbors, rendering the input data length unfixed and not applicable in machine learning models, which usually require fixed-length data as the input.

To address the aforementioned problems, we propose the geographical interactions weighted graph convolutional network-gated recurrent unit (GGCN-GRU) model, which comprises deep learning methods on graphs to dynamically predict user activity intensity. Owing to the large amount of social media data, with more detailed personal trajectory and attribute information, which can better reflect the purpose of people's movement, we used social media check-in data to test the effectiveness of the model. The main contributions of this paper are as follows:

- We represent the spatial relationship of user movement in the form of graphs, which can be directly input into the prediction model. Nodes represent regions, while edges represent adjacency. In addition, we used regional interactions extracted from historical activity data to construct the edges of the graphs. In this manner, the interactions of people in a physical space are considered.
- We used a deep learning model, which has been shown to perform well for predictions in discontinuous nonlinear problems. The model, which recasts the regression problem for predicting the spatial-temporal variation of users as a judgement model, uses a combination of the graph convolutional network (GCN) and gated recurrent unit (GRU). GCN, which is efficient at processing graph data, extracts spatial features [2,33,34]. These features are then input into the GRU, which extracts their temporal features. Finally, the GRU output is passed through a fully connected layer to obtain the predictions.

## 2. Methodology

### 2.1. Problem Description

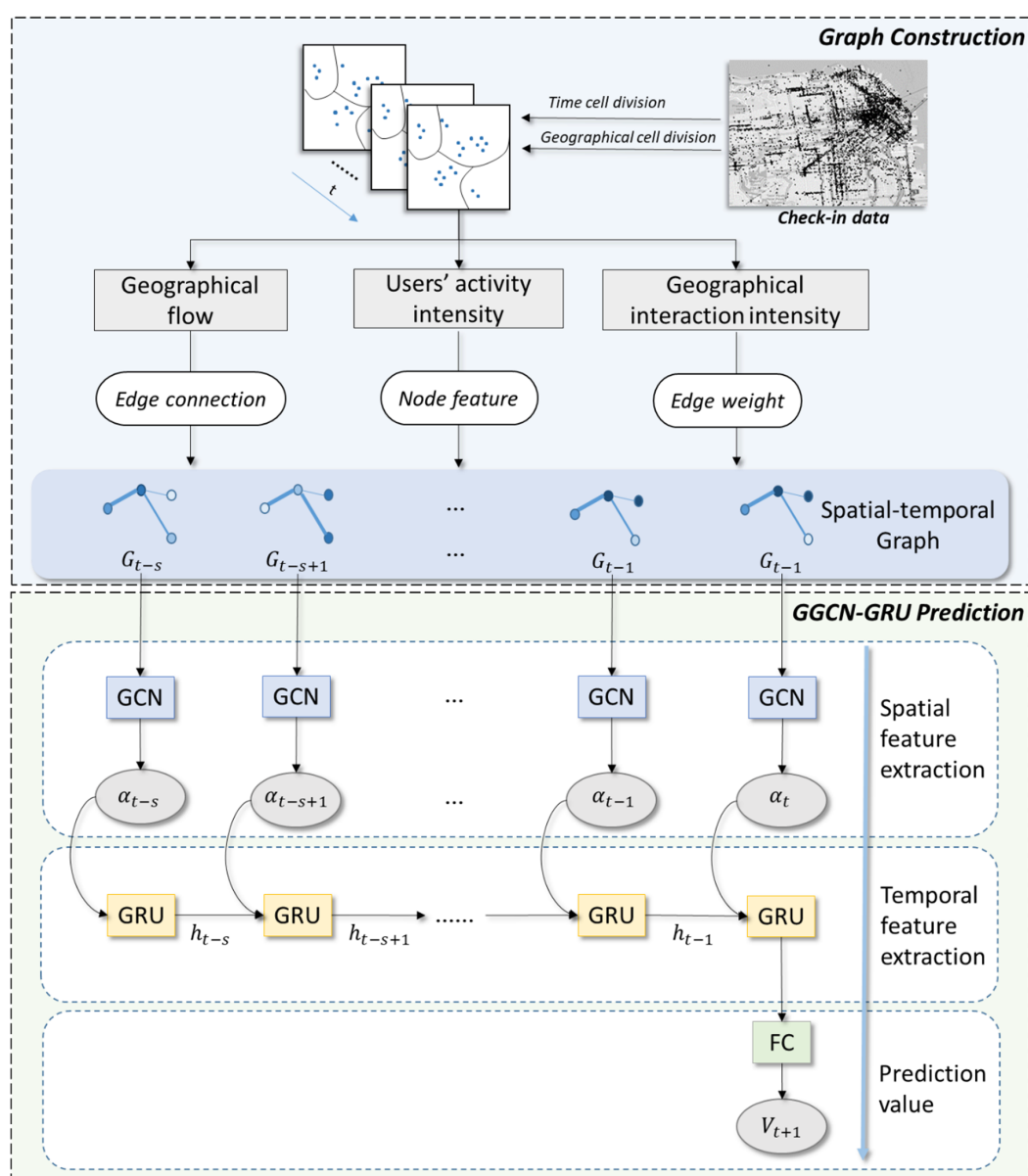
Suppose that the study area,  $R$ , can be divided into  $n$  geographic cells, such that  $R = \{r_1, r_2, \dots, r_n\}$ , while the period  $T$ , can be divided into  $m$  equal time cells, such that  $T = \{t_1, t_2, \dots, t_m\}$ . The user activity intensity of a geographic cell,  $r_i$ , at time cell  $t_j$  can be expressed as  $V_{t_j}^{r_i}$ . The time sequence of the user activity intensity in region  $r_i$  can be described as  $V^{r_i} = \{V_{t_1}^{r_i}, V_{t_2}^{r_i}, \dots, V_{t_m}^{r_i}\}$ . The prediction of the activity intensity is based on the historical activity intensity; therefore, we aimed to better establish the mapping between the historical and predicted values. The problem can be described as follows:

$$V_{t_{m+1}}^{r_i} = f\{V_{t_1}^{r_i}, V_{t_2}^{r_i}, \dots, V_{t_m}^{r_i}\} \quad (1)$$

where  $V_{t_{m+1}}^{r_i}$  is the user activity intensity in the next time cell (i.e., the predicted value) and  $f$  represents the mapping between the predicted and true values. The number of users can reflect the active degree of users in a region [34]; therefore, the number of users was selected to reflect the active degree in this study. Simultaneously, for convenient applications, we used max-min normalization to map the user's activity intensity to a  $[0, 1]$  scale.

## 2.2. GGCN-GRU Model

User activity intensity predictions are the combined result of temporal and spatial analyses. The GGCN-GRU model consists of three components: (1) graph generation via the geographic interactions (GIF) method; (2) spatial feature extraction via GCNs; and (3) capturing user activity intensity dynamics via GRUs. Figure 1 shows the GGCN-GRU model architecture. First, the raw data are partitioned into geographic and time cells. Each geographic cell is treated as a node in the graph; the graph's edges are defined by the geographical interactions of the users, which are in turn weighted by the intensity of these interactions. A spatio-temporal graph is then constructed by assigning values to each node according to the spatio-temporal matrix. This graph is passed to the GCN, which extracts its spatial features. The spatial feature vectors are then input into the GRU module to extract their temporal features. Finally, the temporal feature vectors are input into the fully connected layer, which performs regression computations with an activation function to obtain the predictions.

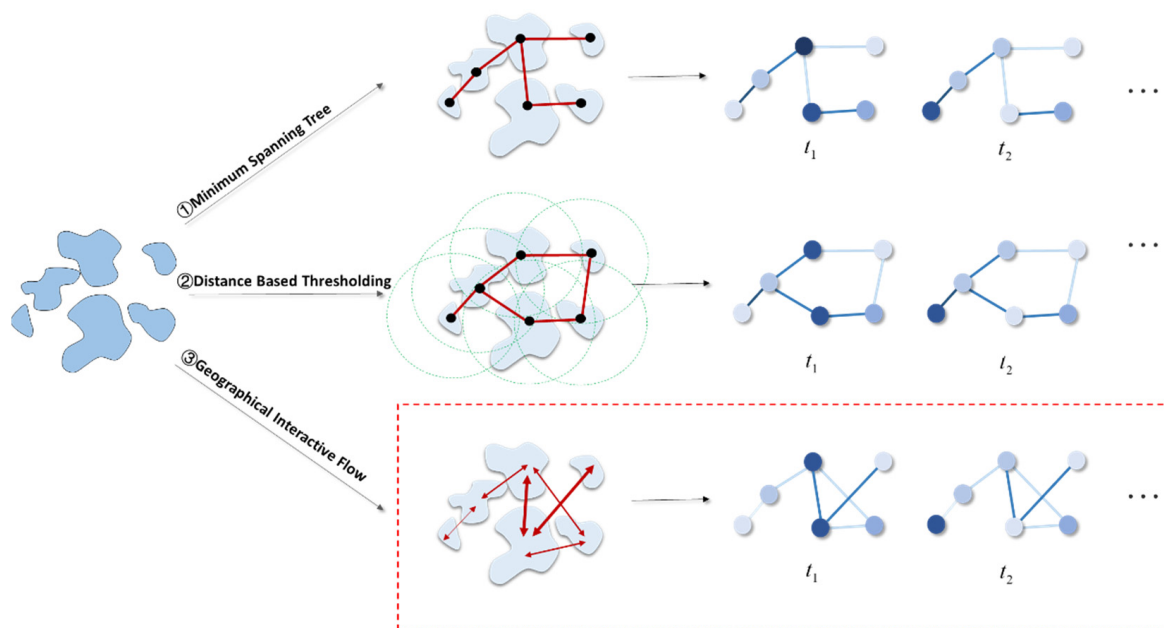


**Figure 1.** Architecture of the geographical interactions weighted graph convolutional network-gated recurrent unit (GGCN-GRU) approach. A spatial temporal graph based on geographical interactions is constructed and then input into the GCN and GRU to extract features.

### 2.3. Construction of the Spatio-Temporal Graph

As the GGCN-GRU model introduces graphs as the direct input, the predictive accuracy of this model will depend on the graph construction method. The graphs can be generated by various approaches; for example, using distance thresholds and connectivity relationships between roads [37], the k-nearest neighbor graph algorithm, the Gabriel graph algorithm, the minimum spanning tree algorithm, and the Delaunay triangulation [38]. Each graph construction method will lead to different graph connectivities. Most current graph construction approaches consider only spatial factors such as spatial relationships and adjacencies while overlooking human factors, such as transportation networks and venue functions.

The three graph construction approaches are illustrated; each method results in different node connectivities (Figure 2). The first method, i.e., the minimum spanning tree (MST), generates minimum connectivity graphs with the minimum possible total edge weight. This method has short training times because it produces only a small number of edges. However, the nodes of the resulting graph may not have valid connections, resulting in a low predictive accuracy. The second method, i.e., distance-based thresholding (DBT), is subject to an over-reliance on the spatial distance to determine the node attributes. The third method, i.e., the GIF method, is a semantics-based approach that uses historical interregional interactions to construct graphs, thus transcending factors, such as distance. As the GIF method is better suited for describing user mobility patterns than the MST or DBT methods, we chose it to construct the graph as input for the GGCN-GRU model.



**Figure 2.** Graph construction methods considered in this study were minimum spanning tree (MST), distance-based thresholding (DBT), and geographical interactions (GIF). MST keeps all nodes connected while minimizing the number of connected edges. DBT connects nodes according to distance; the connected edges are different with the size of the distance threshold. GIF, which uses the geographical interaction flows of historical user movement records, is the method used in this study.

#### 2.3.1. Spatio-Temporal Graph

The user activity intensity depends on both temporal and spatial features. As conventional graphs cannot adequately describe temporal attributes, we used spatio-temporal graphs to characterize the user activity intensity. Suppose that an undirected graph,  $G$ , which possesses time-series attributes, is the composite of multiple spatio-temporal



sub-graphs,  $G_t = (N, E, W, V_t)$ , such that  $G = (G_{t_1}, G_{t_2}, \dots, G_{t_m})$ . Here,  $N$ ,  $E$ ,  $W$ , and  $V_t$  represent the node, edge, edge weight, and time-dependent node attribute (i.e., user movement records at time  $t$ ), respectively. This representation shows that spatio-temporal graphs describe global structures; the time-dependent attributes of their nodes allow these graphs to simultaneously characterize the spatial and temporal features of the user activity intensity.

### 2.3.2. Node Representation

The node structure of the graph depends on geographic cells. In many cases, centroids of geographic cells are used as nodes. When edges are constructed according to spatial adjacency, the distance between each centroid are used to construct the edges, as in the MST and DBT methods (Figure 2). Because the GIF method does not involve distance calculations, geographic cells can directly be abstracted as points (Figure 2). The values of each node depend on the user activity intensity in their corresponding geographic cell. If the user activity intensity in a geographic cell,  $r_i$ , at time cell  $t_j$  can be expressed as  $V_{t_j}^{r_i}$ , the intensity of  $n$  geographic cells in  $m$  time cells can then be described by an  $m \times n$  spatio-temporal matrix,  $V$ , which participates in subsequent computations and is expressed as follows:

$$V = \begin{bmatrix} V_{t_1}^{r_1} & \dots & V_{t_1}^{r_n} \\ \vdots & \ddots & \vdots \\ V_{t_m}^{r_1} & \dots & V_{t_m}^{r_n} \end{bmatrix}. \quad (2)$$

Geographic cells may be partitioned regularly [24,34,36] or irregularly, e.g., based on road networks [36] or clustering areas [2]. As a theoretical analysis of geographic-cell partitioning methods is beyond the scope of this study, we designed three sets of experiments (see Section 3.2) to probe how the shape and size of geographic cells affect the accuracy of the GGCN-GRU.

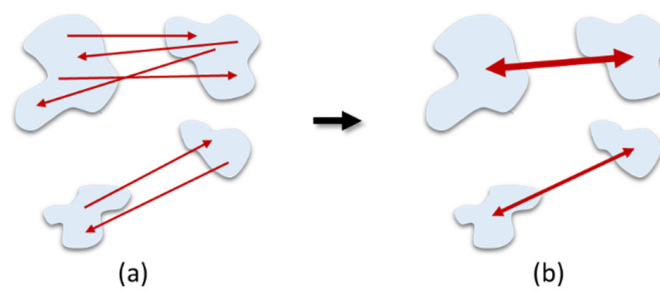
### 2.3.3. Edge Representation

User activity generally exhibits correlations in space. As such, the user activity intensity in a specific geographic cell during a future period depends not only on its historical activity intensity, but also on the movements between other geographic cells [38,39]. If a person moves from one geographic cell to another between two instances of time, an interaction occurs between these geographic cells. This is the underlying assumption of edge construction via the GIF method. Here, we defined the interaction intensity as the number of people that moved between two geographic cells during a period,  $P$ . The weight of each edge was determined by their interaction intensity (Figure 3). The geographic interaction intensity formula can be expressed as follows:

$$I_{(r_i \leftrightarrow r_j)}^P = I_{(r_i \leftarrow r_j)}^P + I_{(r_i \rightarrow r_j)}^P \quad (3)$$

where  $I_{(r_i \leftrightarrow r_j)}^P$  is the interaction intensity between geographic cells  $r_j$  and  $r_i$  during period  $P$ ;  $I_{(r_i \leftarrow r_j)}^P$  is the number of people who moved from  $r_j$  to  $r_i$  during  $P$ , while  $I_{(r_i \rightarrow r_j)}^P$  is the number of people who moved from  $r_i$  to  $r_j$  during period  $P$ . The interaction intensities of  $n$  geographic cells can be used to construct an  $n \times n$  adjacency matrix,  $A$ , which participates in the graph convolution computations described in Section 2.3.1. As a geographic cell cannot interact with itself,  $A$  is a symmetric matrix with a diagonal of 0:

$$A = \begin{bmatrix} 0 & \dots & I_{(r_1, r_n)}^P \\ \vdots & \ddots & \vdots \\ I_{(r_n, r_1)}^P & \dots & 0 \end{bmatrix} \quad (4)$$



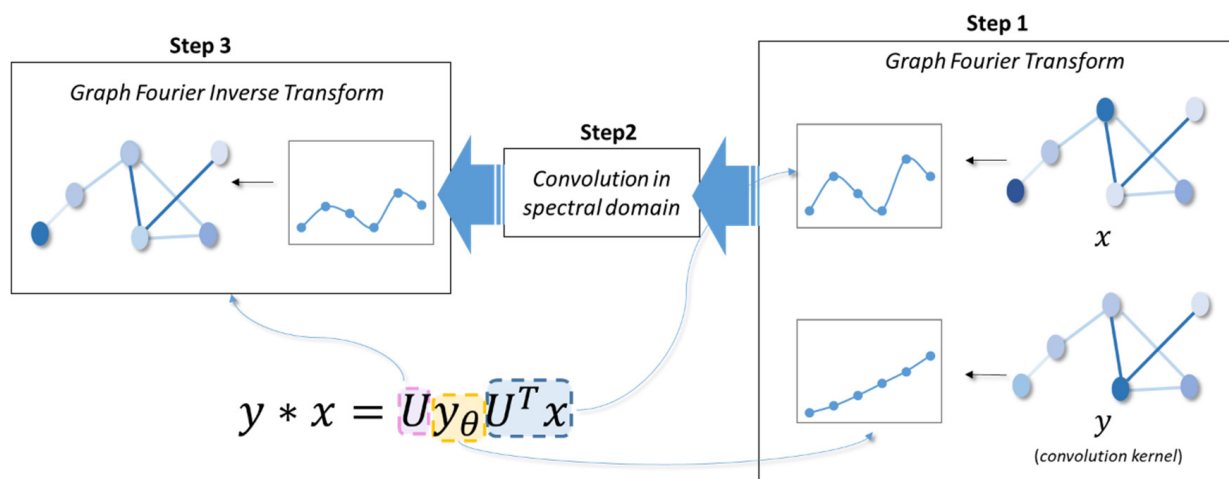
**Figure 3.** Edge weighting based on the geographic interactions between users: (a) edges from user movement flows between different geographic cells and (b) edges with weights from geographic interaction intensities.

#### 2.4. Spatial Feature Extraction by GCN

Although CNNs perform well in feature extraction from regular data, applying CNNs directly to irregular spatio-temporal graphs is difficult. Therefore, we used GCNs, which operate directly on graph data and execute convolutional computations, to extract the spatial features of user activity. The end-to-end graph-based GCN learning process can be adapted to various problems.

##### 2.4.1. Spectral Domain Graph Convolution Operations

The core purpose of a GCN is to extract features by performing convolutional operations on graph data. As irregular graph data are not translationally invariant, it is impossible to perform convolution operations in the spatial domain. Bruna [40] proposed a graph convolution for the spectral domain, which uses a Fourier transform to convert graph data from the spatial domain into the spectral domain for convolution operations. An inverse Fourier transform is then performed to convert the data back to the spatial domain (Figure 4).



**Figure 4.** Graph convolution in the spectral domain.

Fourier transforms are a useful tool in digital signal processing, as they can convert complex convolution operations in the spatial domain into much simpler dot-product operations in the spectral domain. Orthogonalizing the Laplacian matrix representation of the graph,  $L$ , yields an eigenvector,  $U$ , which is usually used as the Fourier basis vector. Given a Laplacian matrix,  $L \in R^{n \times n}$ ,  $L$  can then be calculated from the adjacency matrix,  $A$ , and degree matrix,  $D$ , i.e.,  $L = D - A$ . The Fourier transform of a graph signal,  $x$ , on a Fourier basis,  $U = [u_1, u_2, \dots, u_n]$ , can be expressed as follows:

$$\tilde{x} = U^T x \quad (5)$$

The inverse Fourier transform of the graph signal  $x$  is:

$$x = U\tilde{x} \quad (6)$$

Based on the definition of convolution operations, a convolution in the spatial domain is equivalent to a dot-product operation in the spectral domain. Hence, the convolution of graph signals  $y$  and  $x$  can be expressed as:

$$\begin{aligned} y * x &= U(U^T y \cdot U^T x) \\ &= U(\text{diag}(\tilde{y})(U^T x)) \end{aligned} \quad (7)$$

where  $\text{diag}(\tilde{y})$  is a convolution core characterized by a set of free parameters, i.e.,  $\theta = [\theta_1, \theta_2, \dots, \theta_n]$ ; if  $y_\theta$  is the to-be-learned parameterized function that must be activated by the activation function, the graph neural network layer then has the following expression:

$$x' = F(U y_\theta U^T x) \quad (8)$$

#### 2.4.2. Layer-Wise GCN

In the GCN described in Section 2.3.1, the number of parameters that must be learned is equal to the number of graph nodes. This can lead to high computational complexity and a strong tendency towards overfitting. To avoid these issues, we used the fast approximate graph convolution approach proposed by Kipf and Welling [41], which removes the need to learn all node parameters. Instead, it considers only the first-order neighborhood of the nodes, and increases the size of the spatial domain's receptive field by stacking multiple graph convolutional layers. Figure 5 illustrates the spatial-domain receptive field obtained by stacking two graph convolutional layers. The parameterized function of this simplified multilayer GCN has the following expression:

$$H^{(k+1)} = F(\widetilde{L_{sym}} H^{(k)} W_k) \quad (9)$$

where  $H^{(k+1)}$  is the output of the  $k$ -th layer, with  $H^{(0)}$  being the spatio-temporal matrix  $V$ ;  $F$  is the activation function; and  $\widetilde{L_{sym}} = \widetilde{D}^{-\frac{1}{2}} \widetilde{A} \widetilde{D}^{-\frac{1}{2}}$  is a renormalized Laplacian matrix, where  $\widetilde{A} = A + I_N$  is the self-connected adjacency matrix,  $I_N$  is an identity matrix of size  $N$ ,  $\widetilde{D}_{ii} = \sum_j \widetilde{A}_{ij}$  is the degree of each node, and  $W_k$  is the trainable weight matrix.

As oversmoothing will occur and dramatically reduce the training efficacy if the graph convolutional layers are stacked deeply [42], we chose to use two graph convolutional layers for spatial feature extraction, i.e.,  $k = 2$  (Figure 6). The derivation of the expression that represents spatial-feature extraction by a  $k = 2$  GCN is as follows:

$$k = 0, H^{(0)} = V \quad (10)$$

$$k = 1, H^{(1)} = \text{Relu}(\widetilde{L_{sym}} H^{(0)} W_0) = \text{Relu}(\widetilde{L_{sym}} V W_0), \text{ and} \quad (11)$$

$$k = 2, \alpha = H^{(2)} = \sigma(\widetilde{L_{sym}} H^{(1)} W_1) = \sigma(\widetilde{L_{sym}} \text{Relu}(\widetilde{L_{sym}} V W_0) W_1) \quad (12)$$

where  $\text{Relu}$  is the rectified linear activation function,  $\sigma$  is the sigmoid activation function, and  $\alpha$  is the spatial features obtained from the two GCN layers.



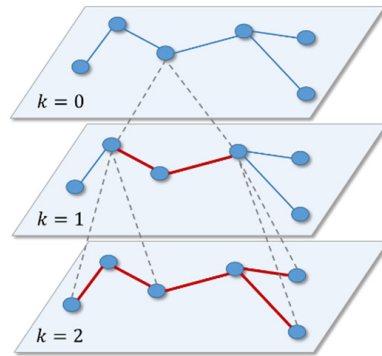


Figure 5. Receptive field of the two stacked graph convolutional layers.

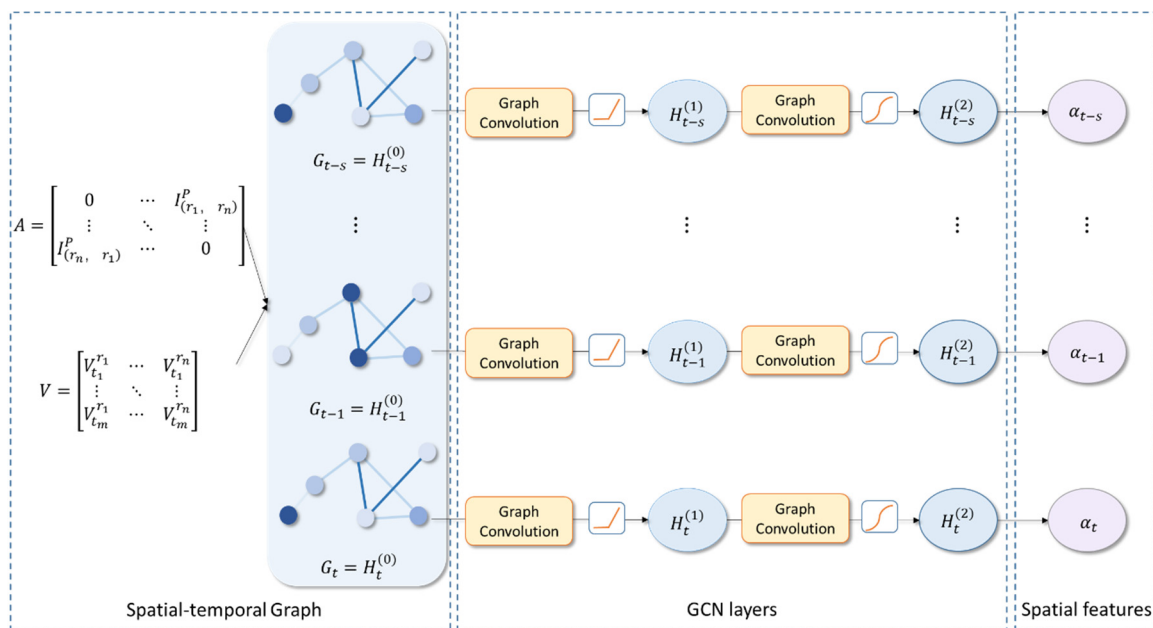


Figure 6. Graph convolutional network (GCN) learning process. Two GCN layers have been used in GGCN-GRU.

### 2.5. Extraction of Temporal Features by GRUs

A GRU is a type of gated RNN; it is one of the most effective sequence modelers available [43]. In principle, GRUs are similar to LSTMs, as they both use gates to control their input and memory in solving the vanishing gradient in conventional RNNs. However, an LSTM has three gates, whereas a GRU only has two (the reset and update gates), which reduces the number of parameters, thus improving the learning efficiency. Hence, we used a GRU to capture the time-dependence of the spatio-temporal series (Figure 7). The predictions of the model are partially affected by the length of the input time steps. The prediction of some information in the next instant, based on the information in the  $s$  preceding time steps, can be expressed as follows:

$$y_{t+1} = \text{GRU}(\alpha_{t-s}, \alpha_t) \quad (13)$$

GRUs control the input of information using reset and update gates. The GRU update gate  $u^t$  controls how much information is carried from the previous GRU to the next GRU, whereas the reset gate  $r^t$  controls how much information from the previous GRU is ignored. This can be expressed as follows:

$$u^t = \sigma(W_u[\alpha^t, h^{t-1}] + b_u), \quad (14)$$

$$r^t = \sigma(W_r[\alpha^t, h^{t-1}] + b_r), \quad (15)$$

$$c^t = \tanh(W_c[\alpha^t, (r^t \otimes h^{t-1})] + b_c), \text{ and} \quad (16)$$

$$h^t = u^t \otimes h^{t-1} + (1 - u^t) \otimes c^t \quad (17)$$

where  $u^t$  is the update gate;  $r^t$  is the reset gate;  $c^t$  is the candidate hidden state of the current time;  $h^{t-1}$  is the hidden state of the previous time;  $h^t$  is the hidden state that is sent to the next time;  $\alpha^t$  is the spatial eigenvector computed by Equation (11); “ $\otimes$ ” indicates a tensor product;  $\sigma$  and  $\tanh$  are activation functions in the neural network layer; and  $W$  and  $b$  are the trainable weight and bias terms, respectively.

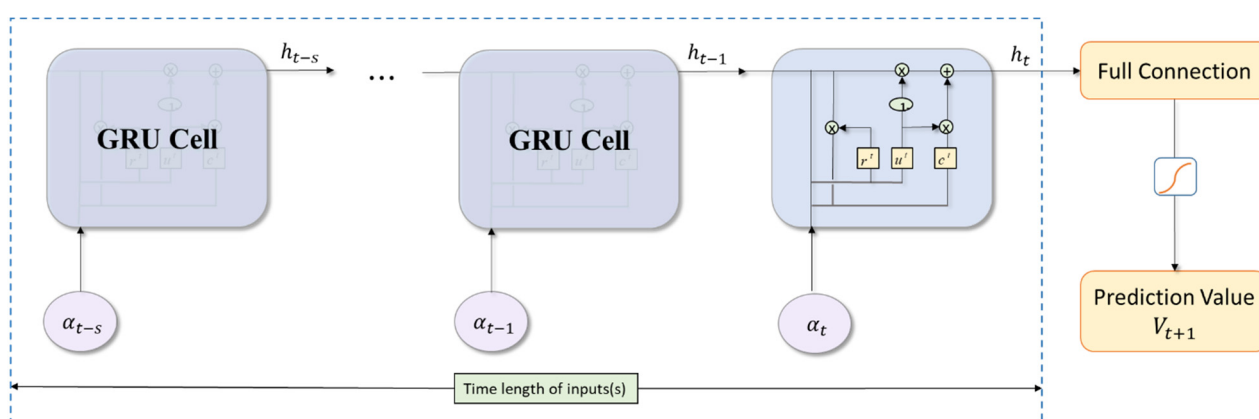


Figure 7. Gated recurrent unit (GRU) structure.

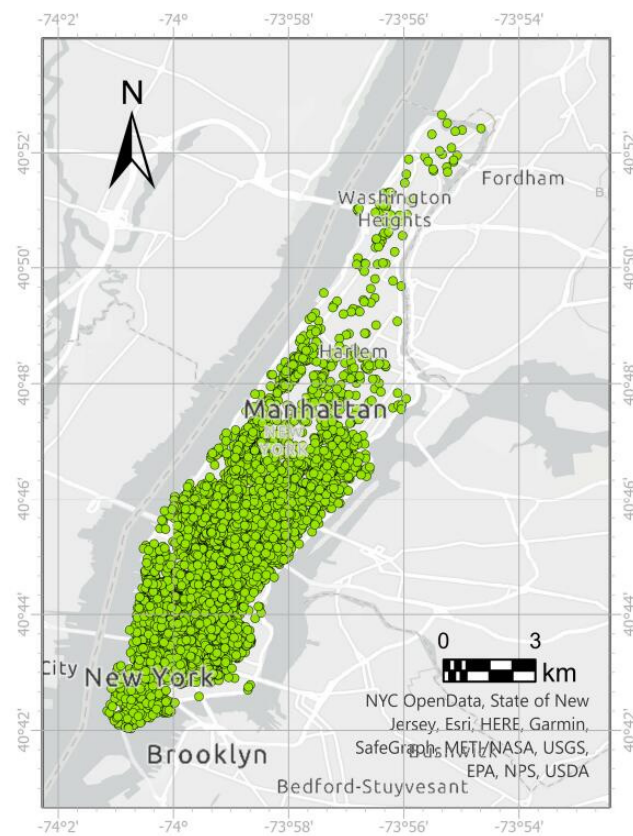
### 3. Experiments

#### 3.1. Data Description

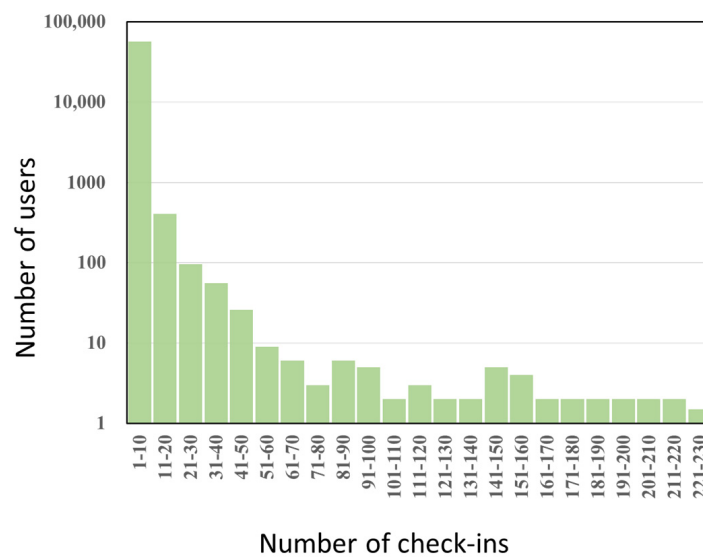
Social media check-ins record the location of a person dynamically. From this data, user preferences and habits can be extracted, and predictions of the spatio-temporal nature of the user activity intensity can be performed [44]. Here, we validated our method using a check-in dataset from the Manhattan borough of New York City (NY). The check-in dataset was collected from the Foursquare social media platform. The experimental dataset comprised 57,297 check-in records over 280 d, dated from 1 January, 2012 to 4 October, 2012. The raw data contained eight attributes, including check-in time, latitude, and longitude (Table 1). Figure 8 shows distribution and statistical information of the social media check-ins. Significant spatial features are present in the core check-in areas (Figure 8a). The number of users corresponding to the number of check-ins had a long-tailed distribution (Figure 8b). Low-frequency users dominated the check-in dataset. Hence, the check-in data used in this experiment adequately reflected the classic check-in behaviors of users [45].

Table 1. Details of social media check-in data.

Attributes	Samples
USER_ID	13,299
LATITUDE	40.782
LONGITUDE	−73.958
DATE	625
TIME	15:30
POI_TYPE	Museum
POI_TYPENU	12,348
CITY	New York



(a) Geographical distribution of check-ins



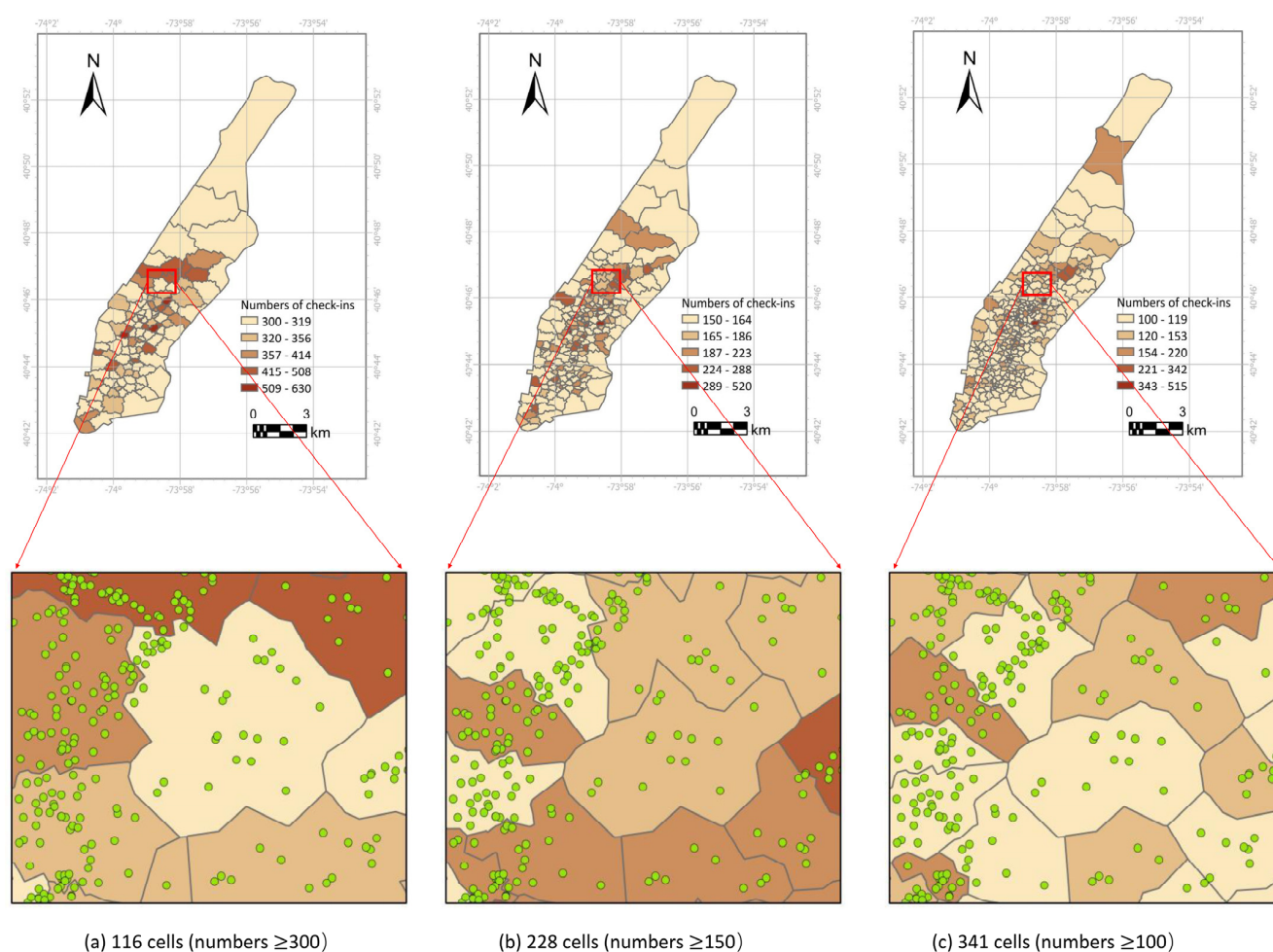
(b) Statistical information of check-ins

**Figure 8.** Overview of the social media check-ins for New York City: (a) Geographical distribution of check-ins. Most of the check-ins are concentrated in the middle and south, while there are few in the north; (b) Statistical information of check-ins. The statistical information of different users demonstrates a long tail effect, with the number of check-ins per user ranging from 1 to 230.

### 3.2. Data Processing

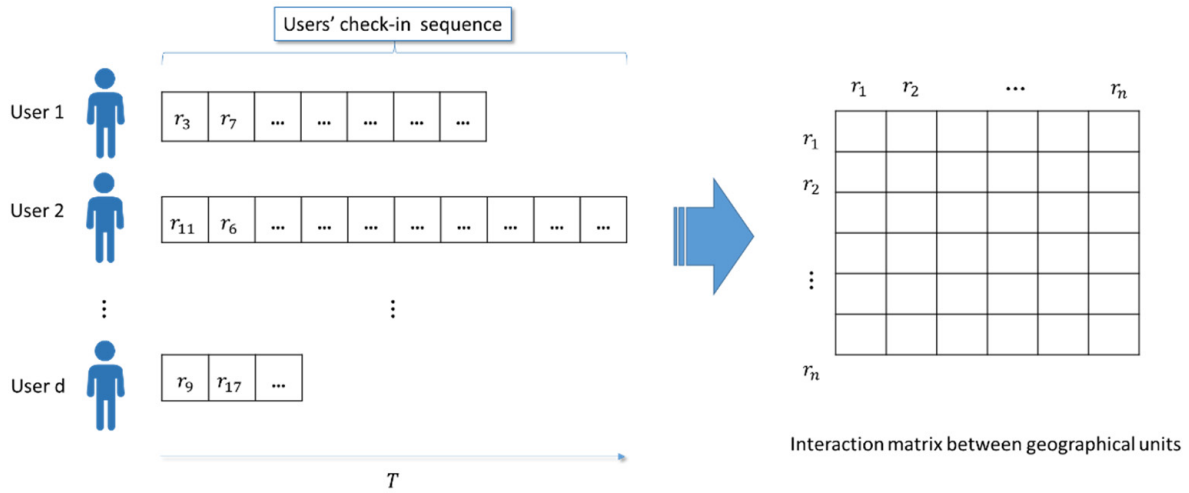
The training dataset comprised 45,838 check-in records from the first 224 days (1 January 2012 to 11 August 2012), and the testing dataset comprised 11,459 records from the remaining 56 days (12 August 2012 to 4 October 2012). As check-in data are sparse and sampled over an extended period, a short time interval renders it difficult to extract significant features from the data, whereas an excessively long interval will mask periodic trends in the data. To avoid producing an overly sparse dataset as well as to account for the semantic meanings of each time-of-day period, each day was divided into four time intervals: dawn (00:00–06:00), morning (06:00–12:00), afternoon (12:00–18:00), and night (18:00–24:00). The datasets were partitioned into 1120 time intervals.

The “Generate Subset Polygons” tool in ArcGIS Pro (<https://pro.arcgis.com>, accessed on 16 August 2021) was used to generate irregular polygons to group the check-ins into compact non-overlapping subsets. Geographic cells were then constructed around the subsets. We limited the number of subset polygons by setting a minimum number of check-ins for each cell, thus also preventing empty-cell generation. Setting the minimum number of check-ins to 100, we obtained 341 geographic cells; at 150 check-ins, we obtained 228 cells; and at 300 check-ins, there were 116 cells. These three datasets were used to validate the effectiveness of our method as well as to test how the number of geographic cells affects the GGCN-GRU model. Figure 9 shows the partitioning of the study area in each of these cases.



**Figure 9.** Three geographical unit partitioning schemes, depending on the minimum specified number of check-ins: (a) 116, (b) 228, and (c) 341 cells were obtained when the minimum number of check-ins was set to 300, 150, and 100, respectively.

To ensure the independence of the testing data, the first 224-day (1 January 2012 to 11 August 2012) check-in training dataset was used to calculate the interaction intensities of each geographic cell. The users were numbered to track their location at each instant. The interactions that occurred between the geographic cells in each time interval were identified by tracing the movement of all users during said time interval (Figure 10).



**Figure 10.** Computation of the geographic interactions of user movements. The user check-in sequences are constructed from their location records at different time intervals.

### 3.3. Assessment Metrics

The root mean square error (RMSE), mean absolute error (MAE), and coefficient of determination ( $R^2$ ) were used to evaluate the model's predictive accuracy. The RMSE and MAE are proportional to the difference between the true and predicted values; lower values indicate greater accuracy.  $R^2$  refers to the goodness of fit, which measures the ability of the predictions to represent the truth.  $R^2$  values closer to 1 indicate a better fitting degree of the regression line to the true value. In contrast, lower values indicate a poorer fitting effect. The RMSE, MAE, and  $R^2$  were calculated as follows:

$$\text{RMSE} = \sqrt{\frac{1}{\xi} \sum (\hat{y}_{t+1}^i - y_{t+1}^i)^2} \quad (18)$$

$$\text{MAE} = \frac{1}{\xi} \sum |\hat{y}_{t+1}^i - y_{t+1}^i|, \text{ and} \quad (19)$$

$$R^2 = 1 - \frac{\sum (\hat{y}_{t+1}^i - y_{t+1}^i)^2}{\sum (\bar{Y} - y_{t+1}^i)^2} \quad (20)$$

where  $y_{t+1}^i$  and  $\hat{y}_{t+1}^i$  are, respectively, the true and predicted user activity intensities of region  $i$  at time  $t + 1$ ,  $\xi$  is the total number of samples, and  $\bar{Y}$  is the mean of the set of  $y_{t+1}^i$ .

### 3.4. Baselines

We selected the following prediction methods to compare with the GGCN-GRU model to reflect the effectiveness of this method. The comparison results are provided in Section 4.3.

HA [46]: the historical average model is a simple and classic prediction method that uses the average information in the historical period for predictions.



ARIMA [17]: the autoregressive integrated moving average model is a parameter-based model that predicts the user activity intensity by fitting historical time-series. This method depends on the stationarity of historical data.

SVR [33]: the support vector regression model employs historical user activity data to train and obtain the relationship between the input and the output data. The trained model is finally used for predictions.

GRU [43]: the simplified RNN with less parameters and faster operation (see Section 2.5).

T-GCN [37]: the temporal graph convolutional network is a short-term traffic prediction model that uses GCN to extract the spatial features of traffic flows by only considering the proximity between regions.

### 3.5. Model Parameter Settings

The experiments were carried out in Windows 10/64 bit/i7 processor and an 8G-memory hardware environment. The proposed GCN-GRU model was implemented with Python in TensorFlow. We set the ratio between the training and testing datasets to 4:1. The learning rate was set to 0.001 (as is conventional), and the batch size was set to 256, according to the bitrate of the graphics card in the experimental environment. The number of training epochs was set to 50. The loss of the model's predictions decreased as the number of epochs increased (Figure 11), indicating normal convergence during training. As the number of hidden units in the GRU affects the performance and accuracy of the GGCN-GRU model, we calculated the RMSE and MAE for 16, 32, 64, 100, and 128 hidden units: they took the minimal values for 128 units in all three datasets (Figure 12). Therefore, we set the number of hidden units in the GRU to 128. Based on Section 2.4, the length of the GRU's input time step affects predictions in the next time. Here, we investigated how the length of the time step affects the predictions by setting the input time step to 3, 6, 12, 15, and 18, and calculating the RMSEs (Figure 13). This revealed that the effects of the length of the time step on the RMSE differ among datasets. A short time step reduces the model's ability to learn time-series data, whereas an excessively long time step leads to overlearning, thus reducing accuracy. For each dataset, we used the time step that yielded the minimum RMSE value (9, 6, and 6 for the datasets with 116, 228, and 341 geographic cells, respectively).

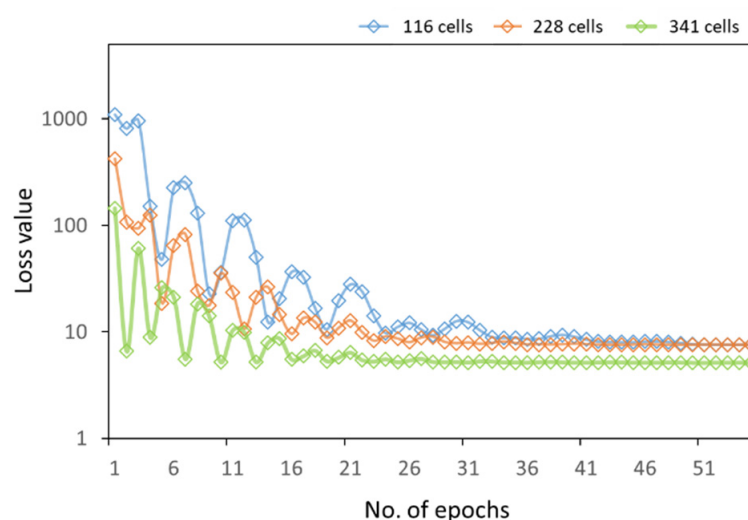
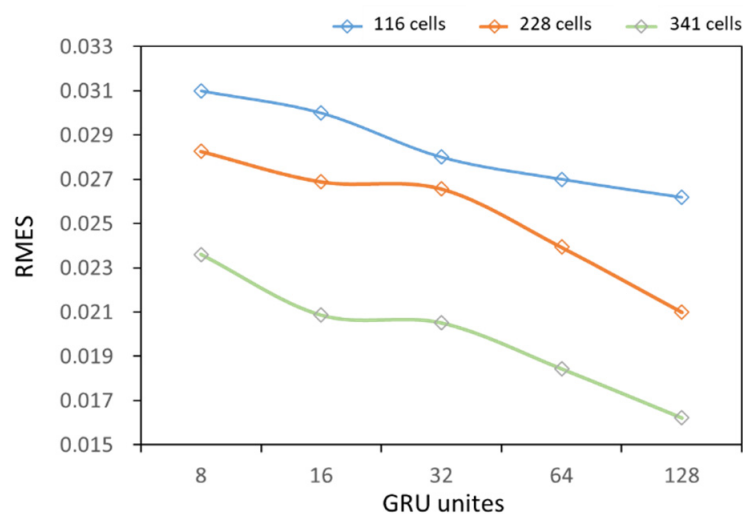
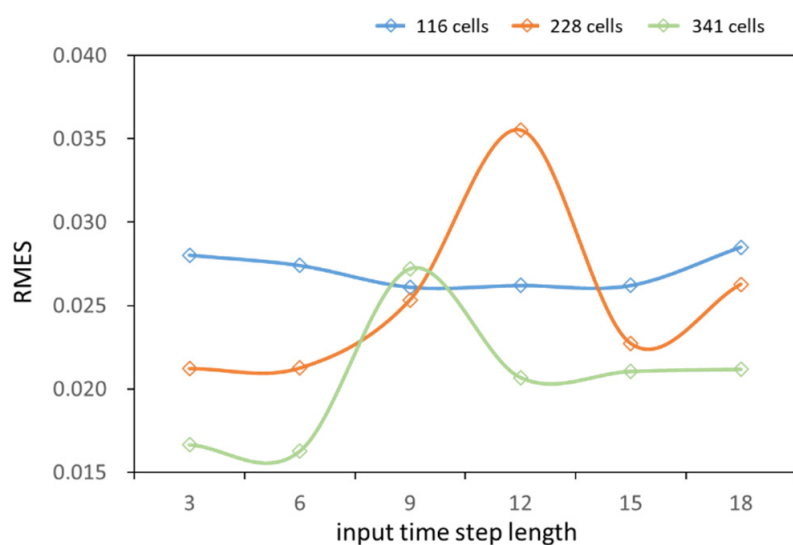


Figure 11. Loss of predictive value versus the number of epochs.





**Figure 12.** Predictive accuracy versus the number of hidden units.



**Figure 13.** Predictive accuracy, in terms of the root mean square error (RMSE) versus the input time step length.

## 4. Results and Analyses

### 4.1. Comparing Accuracies of the Three Graph Construction Methods

We compared the predictive accuracies of the models constructed using the three graph construction methods, i.e., DBT (with distance thresholds of 500, 1000, and 2000 m), MST, and GIF, using three geographic partitioning schemes (Section 3.2). Table 2 lists the number of connected nodes and edges. Based on the RMSE and MAE values, GIF is significantly more accurate than the other two methods. This proves that the GIF method is effective for the problem.

As each graph construction method results in different numbers of connected nodes and edges, the connectivities of their graphs also differs. As the DBT depends on distance, the number of connected nodes in the DBT is smaller than the number of geographic cells at small distance thresholds (such as at distances  $\leq 500$  m). At the small distance threshold, DBT is poorly connected, and there are a number of nodes without connecting edges. The connectivity of the graph improves as the distance threshold increases. For the dataset with 116 geographic cells, the connectivity of DBT improved as the distance threshold increased, which subsequently improved the model's accuracy. MST, by definition, creates minimally connected graphs, but this also results in poor accuracy. Hence, even if the graph is fully

connected, the model accuracy still depends on the number of edge connections in the graph. For all three methods, by using 116 geographic cells, the RMSE and MAE values decreased with an increase in the number of edges for the same number of connected nodes. However, using 228 or 341 geographic cells, GIF still had the lowest RMSE and MAE values, despite having a significantly lower number of edges than DBT for the 2000 m distance threshold. Therefore, although increasing the number of edges increases the node-to-node connectivity, it also increases the graph density. This increases the training time of the model. Furthermore, an excessive number of edges increases the number of invalid connections, thus reducing the efficacy of the spatial feature extraction. As the edges of GIF are based on real historical activities, they reflect the real-world connectivity of the geographic cells. Hence, GIF creates realistic and valid node connections.

For all three datasets, the RMSE and MAE both decreased with an increase in the number of geographic cells. All three methods achieved their highest accuracies with 341 geographic cells. This indicates that the spatial scale affects the model's accuracy. The positive association between the accuracy and number of geographic cells may be attributed to the receptive field becoming smaller with the use of more geographic cells. This increases the granularity of the spatial feature extraction and strengthens the inputs of the next neural network, which then enhances the accuracy of the final output.

To measure the ability of the predicted results to represent the truth, we calculated the  $R^2$  of each method. Our method achieved the highest value of  $R^2$  in all three datasets. This indicates that the prediction result of this method was more representative of the real value than those of the other methods. However, the maximum value of  $R^2$  did not exceed 0.8, which is not particularly ideal for the prediction problem. This shows that the prediction is good, but not sufficiently good. This may be due to neglecting the time periodicity in the model design.

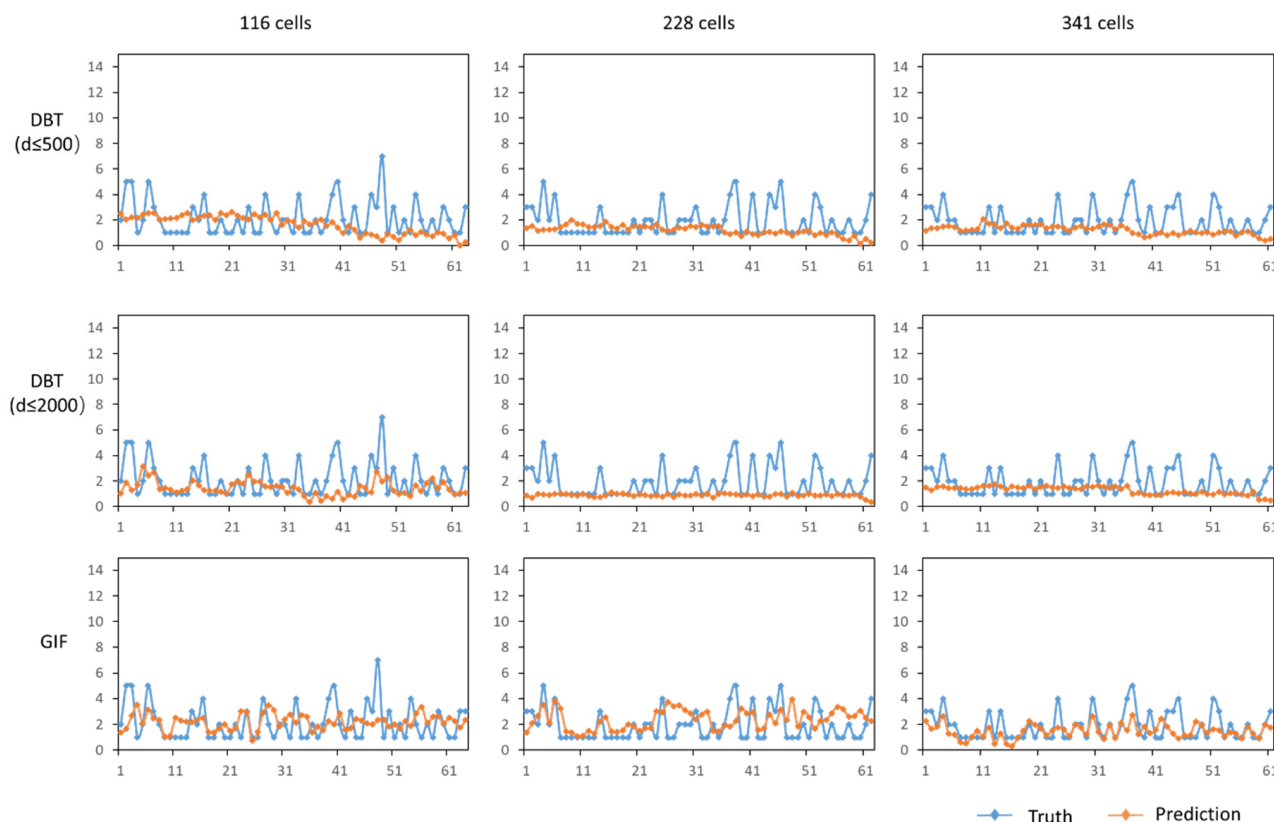
**Table 2.** Predictive accuracy of the graph construction methods at different scales, as indicated by distance (“d”).

Geographic Cells	Graph Type	Connected Nodes	Edges	RMSE	MAE	$R^2$
116	DBT( $d \leq 500$ m)	77	171	0.037	0.022	*
	DBT( $d \leq 1000$ m)	112	685	0.029	0.016	0.518
	DBT( $d \leq 2000$ m)	116	1779	0.028	0.015	0.579
	MST	116	115	0.036	0.022	*
	<b>GIF</b>	116	2704	<b>0.026</b>	<b>0.014</b>	<b>0.695</b>
228	DBT( $d \leq 500$ m)	204	817	0.032	0.023	*
	DBT( $d \leq 1000$ m)	223	3119	0.029	0.019	*
	DBT( $d \leq 2000$ m)	227	9062	0.032	0.021	*
	MST	228	227	0.025	0.014	*
	<b>GIF</b>	228	5212	<b>0.021</b>	<b>0.011</b>	<b>0.733</b>
341	DBT( $d \leq 500$ m)	315	1936	0.029	0.019	*
	DBT( $d \leq 1000$ m)	337	7056	0.020	0.009	*
	DBT( $d \leq 2000$ m)	340	20,080	0.030	0.010	*
	MST	341	340	0.021	0.011	*
	<b>GIF</b>	341	6981	<b>0.016</b>	<b>0.008</b>	<b>0.793</b>

The symbol “\*” indicates that  $R^2$  is  $< 0.5$ , the boldface indicates best results.

Figure 14 illustrates the predicted and true values obtained using DBT (at distances  $\leq 500$ , 1000, and 2000 m) and GIF for certain geographic cells at different times. DBT, at the 500 m distance threshold, caused oversmoothing using all three datasets. The 500-m distance threshold resulted in limited connections to adequately describe the spatial features of the dataset. The accuracy of DBT, at the 2000 m threshold, decreased as the number of geographic cells increased. This is likely due to the large distance threshold, which caused the graph to have excessive connections, thus hindering the extraction of the valid spatial features. We can also conclude that the predictive accuracy of spatial adjacency methods strongly depends on the selection of an optimal distance threshold, which must be determined empirically using a large number of trials. As the fit of GIF is superior to

that of the other methods in all three datasets, we can conclude that GIF can characterize spatial features while also being generalizable to different datasets. However, the peaks of the true values were poorly fitted, even when using the GIF method. This problem may be intrinsic to GCNs and the input data. In the frequency domain, the check-in data consists of low-frequency (low values) and high-frequency (high values) features. As the GCN acts like a low-pass filter, it excludes high-frequency information (high values) in the data and focuses on learning low-frequency information (low values), for which there are several valid features. This indicates that the filtering of high-frequency information by the GCN resulted in poor predictions of peaks and a “smoothened” range of predictions.



**Figure 14.** True and predicted values for certain geographic cells at different times. X-axis: time cells (one cell = 6 h). Y-axis: user activity intensity.

#### 4.2. Model Accuracies Using Different Time Granularities

To test the effects of the length of the time interval on the predictive accuracy, we conducted control experiments with a 12 h time interval. The RMSE and MAE decreased with an increasing number of geographic cells, at both 6 and 12 h time intervals (Table 3). This provides further support for the conclusions reported in Section 4.1, i.e., the number of geographic cells is positively correlated with the predictive accuracy. The 6 h interval provided more accurate predictions than the 12 h interval. Therefore, the time interval configuration partially affects the GGCN-GRU model’s accuracy. As using a longer time interval will increase the check-in records of users within each time cell, this reduces the granularity of temporal feature extraction, thus reducing the accuracy.

**Table 3.** Root mean square error (RMSE) and mean absolute error (MAE) at different time interval lengths.

Units	Time Interval = 6 h		Time Interval = 12 h	
	RMSE	MAE	RMSE	MAE
116	0.026	0.014	0.045	0.027
228	0.021	0.011	0.024	0.016
341	0.016	0.008	<b>0.021</b>	<b>0.012</b>

#### 4.3. Comparing GGCN-GRU to Other Common Spatio-Temporal Prediction Methods

We compared the GGCN-GRU model to five other methods: historical average [46], ARIMA [17], SVR [33], GRU [43], and T-GCN [37] (Table 4). GGCN-GRU outperformed the other models in terms of the RMSE (by up to 1.314) and MAE (by up to 0.683), indicating that it is effective for predictions of the user activity intensity. The two conventional time-series modelers (historical average and ARIMA) and the regression-based method (SVR) performed poorly with respect to the predictions because these methods rely entirely on historical data, without accounting for spatial factors. Furthermore, these methods generally perform poorly when fitting non-stationary time-series with trends and periodic behaviors. The GRU and temporal GCN methods performed reasonably well at this task, indicating that neural network methods are suitable for fitting complex nonlinear spatio-temporal data. However, the GRU method extracted only temporal features and neglected spatial features. Although temporal GCNs do account for spatial features, they do not consider the strength of spatio-temporal associations. The GGCN-GRU method, in contrast, considers not only the spatial and temporal features of user activity, but also the strength of the geographic cell interactions. Owing to these characteristics, the GGCN-GRU method fits complex nonlinear spatio-temporal relationships with relatively higher accuracy than the other methods.

**Table 4.** Comparison of the GGCN-GRU method and other spatio-temporal prediction methods.

Model	116 Cells		228 Cells		341 Cells	
	RMSE	MAE	RMSE	MAE	RMSE	MAE
HA	1.340	0.697	0.844	0.411	0.677	0.310
ARIMA	1.331	0.554	0.820	0.406	0.674	0.165
SVR	1.317	0.528	0.817	0.326	0.657	0.266
GRU	0.041	0.024	0.037	0.022	0.035	0.022
T-GCN	0.031	0.019	0.027	0.019	0.018	0.011
<b>GGCN-GRU</b>	0.026	0.014	0.021	0.011	<b>0.016</b>	<b>0.008</b>

HA: historical average; ARIMA: autoregressive integrated moving average; SVR: support vector regression; GRU: gated recurrent unit; T-GCN: temporal graph convolutional network; and GGCN-GRU: geographical interactions weighted graph convolutional network-GRU.

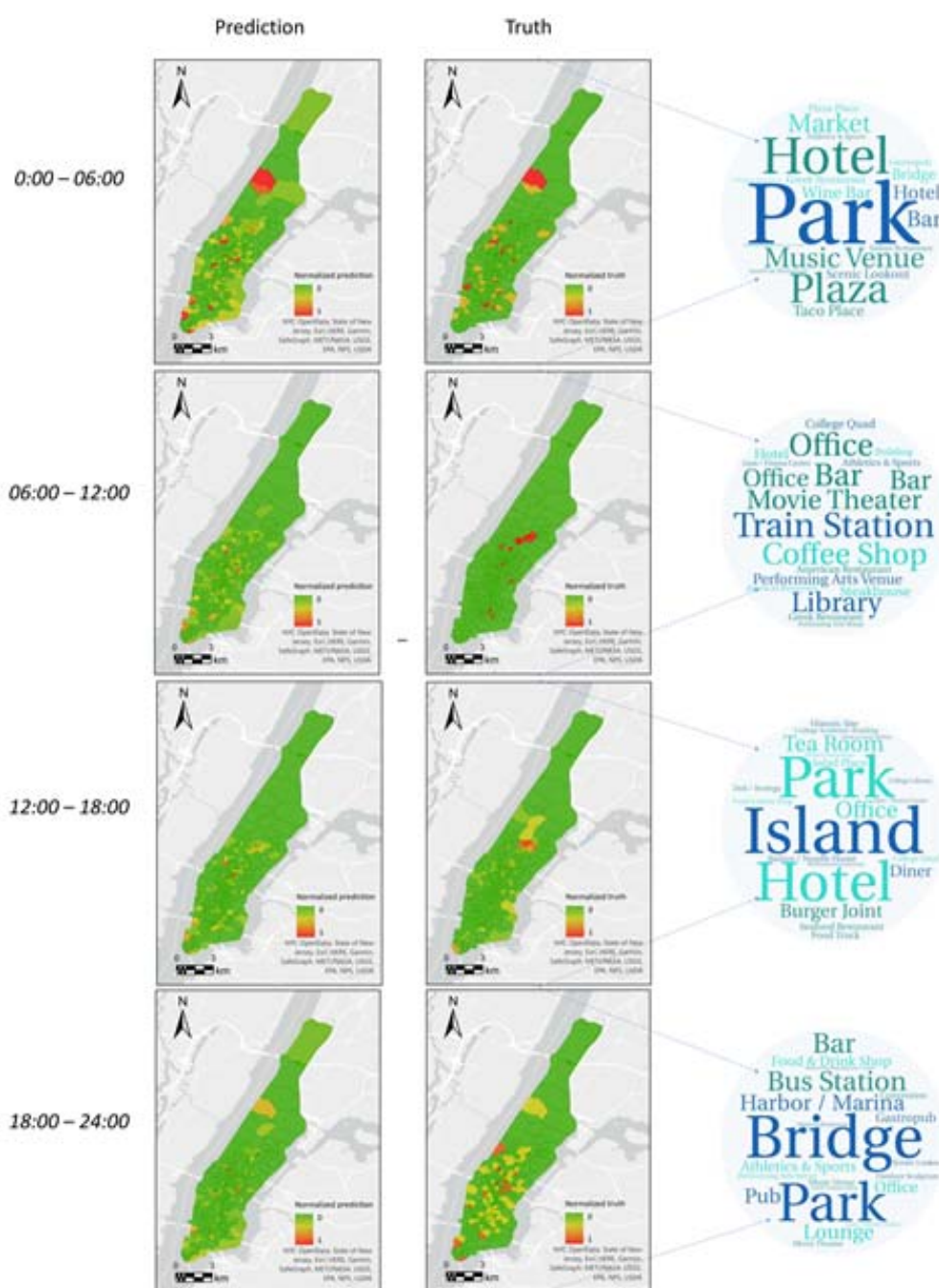
#### 4.4. Visualization

Figure 15 shows the true and predicted activity intensity of 341 geographic cells at four time intervals (dawn, morning, afternoon, and night). To provide a semantic explanation for user movements, we used the “frequency-inverse document frequency (TF-IDF) algorithm” [47] to determine the significance (i.e., the importance, rather than statistical significance) of the local point-of-interest (POI) types in each geographic cell. The POI types of the check-in data can reflect the types of popular places in a region, which can explain why people like to visit a specific place. The significance of the POI types in a geographical unit reflects the preferences for the functional place-types. Although the number of check-ins for a given POI type can reflect place-preferences, it does not reflect the relationship between the place’s function and the local area. The TF-IDF algorithm is used to evaluate the importance of a word (or phrase) in a set of files (e.g., a set of articles). Words or phrases that appear frequently in one article and rarely in others are considered

to have high utility for distinguishing between categories. In this study, we regarded the local area as a document and the entire research area as a document set when calculating the significance of the local POI types, as follows:

$$S_{r_i(w)} = \left( \frac{f_{r_i(w)}}{f_{r_i}} \right) \times \lg \frac{N}{n_w} \quad (21)$$

where  $S_{r_i(w)}$  is the significance of the POI type $_{(w)}$ ,  $f_{r_i(w)}$  is the check-in frequency of the POI type $_{(w)}$  in unit  $r_i$ ,  $f_{r_i}$  is the total number of check-ins in unit  $r_i$ ,  $N$  is the number of geographical units divided by the entire region, and  $n_w$  is the number of units where the POI type $_{(w)}$  occurs.



**Figure 15.** User activity intensity heat map. The values were normalized by the min-max method. The font size in the tag cloud corresponds to the significance of the POI type. The more significant the POI type, the larger the corresponding font.



Figure 15 reveals three findings. First, the GGCN-GRU predictions approximated the real activity intensity of users reasonably well, although less accurately at times with a high intensity (e.g., 06:00 to 12:00). Second, the user activity intensity showed temporal differences, e.g., it was significantly greater from 18:00–24:00 than from 06:00–12:00. Third, the user activity intensity reflects different preferences at different times. For example, certain venues, such as bus stations, coffee shops, and offices, are preferred destinations during the day, whereas parks and hotels are preferred destinations at night. Therefore, the influence that a venue has on user movements will depend on its function.

## 5. Conclusions

User activity intensity prediction is an important aspect of spatio-temporal human mobility studies. Owing to the rapid development of transportation systems and road networks, spatial distance is no longer the sole constraint in human mobility. As a result, conventional spatio-temporal prediction approaches based solely on spatial adjacency are no longer suitable for the prediction of the user activity intensity. Meanwhile, different geographic cells may have various numbers of neighbors, but some machine learning methods require a fixed length data form as input. To address these issues, we constructed spatial relationships between cells in the form of graphs to address the conflict of the fixed-length input format and the various number of neighbors. We determined adjacency based on user movements between geographic cells (i.e., the “geographical interactions” (GIF) method) to improve the prediction accuracy. The predicting model was created by combining the GIF graph construction method with the GCN and GRU approaches. It was designed to fit the nonlinear spatio-temporal patterns of user activities; this provided more accurate predictions than earlier methods. We validated the GGCN-GRU model on real check-in data, demonstrating that it could be good at fitting nonlinear spatio-temporal relationships, thus predicting the intensity.

We used three datasets with different numbers and geographic cell shapes to validate this model’s effectiveness; it performed well for three datasets. Furthermore, its accuracy was positively associated with the number of geographic cells. We compared this proposed GIF graph construction method to the DBT and MST methods, evaluating how well our model performed. The GIF method yielded the highest accuracies. These findings reveal that user movements are well characterized by the spatial adjacencies provided by this method. Two types of time partitions were used to test the effects of the length of the time cell on the model: the model was more accurate when using shorter time intervals. Therefore, the time cell configuration affects the GGCN-GRU model’s efficacy. The GGCN-GRU model had a better accuracy than the other commonly used prediction methods. Hence, this model could fit the complex nonlinear spatio-temporal relationships that characterize user movements. Finally, we visualized the true and GGCN-GRU-predicted user activity intensity for a geographic region and calculated the significance of the POI types reflected in the check-in data, to provide a semantic explanation for the intensity in each time interval. The time intervals varied in terms of the intensities and user preferences. This result will support future studies of human spatio-temporal behavior.

Nevertheless, our method has certain limitations:

(1) The peaks of the true values were poorly fitted. As the GCN acts like a low-pass filter [42], it excludes high-frequency information (high values) in the data and focuses on the learning of low-frequency information (low values), for which there are many valid features. This indicates that the filtering of high-frequency information by the GCN resulted in poor peak predictions and a “smoothed” range of predicted values.

(2) The model only considers physical interactions and ignores social interactions, which also have an important role in user mobility. In addition, it ignores the impact of context. Taking social interactions and the semantic context, such as POIs, into consideration is crucial as well.



(3) For construction of graphs, historical data are used; therefore, when applying new data, the graph may need to be reconstructed. The boundary and generalization ability of the model require further testing.

(4) Although the prediction accuracy of the model was superior to that of other spatio-temporal prediction models, the ability of the model to represent the real value was not sufficiently good. This may be due to the fact that the periodicity and seasonality of crowd activities were not considered in the design of the model. In our future studies, we will attempt to introduce the deep-learning attention mechanism [48] to autonomously learn the temporal periodicity and spatial relationships of user movements, thus assigning weights to the model and thereby further improving its predictive accuracy.

(5) Although social media check-in datasets have been shown to reflect common human activity patterns, similar to other geotagged datasets [16], there are still some problems in these data types, such as sparsity and non-representativeness. For example, people may perform false check-ins due to the reward mechanisms featured on social media platforms. One effective method to deal with this bias and the sparsity of data is to integrate various types of human tracking data, such as mobile phone and GPS location data.

**Author Contributions:** Jing Li and Haiyan Liu conceived this study. Jing Li designed the methodology and Wenyue Guo implemented the main model. Anzhu Yu aided in the interpretation of the results. Jia Li and Xin Chen illustrated the main figures. All authors have read and agreed to the published version of the manuscript.

**Funding:** This study was funded by the National Natural Science Foundation of China under grant number 41801388, as well as the Natural Science Foundation of Henan Province under grant number 182300410005.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data presented in this study are available from the author upon reasonable request.

**Acknowledgments:** The authors thank the editors and anonymous reviewers for their insightful comments and constructive suggestions.

**Conflicts of Interest:** The authors declare no potential conflict of interest.

## References

1. Bao, Y.; Huang, Z.; Li, L. A BiLSTM-CNN model for predicting users' next locations based on geotagged social media. *Int. J. Geogr. Inf. Sci.* **2021**, *35*, 639–660. [[CrossRef](#)]
2. Yuan, N.J.; Zheng, Y.; Zhang, L. T-finder: A recommender system for finding passengers and vacant taxis. *IEEE Trans. Knowl. Data Eng.* **2013**, *25*, 2390–2403. [[CrossRef](#)]
3. Ma, X.; Tao, Z.; Wang, Y.; Yu, H.; Wang, Y. Long short-term memory neural network for traffic speed prediction using remote microwave sensor data. *Transp. Res. Part C Emerg. Technol.* **2015**, *54*, 187–197. [[CrossRef](#)]
4. Wang, Y.; Zhou, X.; Noulas, A. Predicting the Spatio-Temporal Evolution of Chronic Diseases in Population with Human Mobility Data. In Proceedings of the 27th International Joint Conference on Artificial Intelligence, Stockholm, Sweden, 13 July 2018; pp. 3578–3584.
5. Balcan, D.; Colizza, V.; Gonçalves, B. Multiscale mobility networks and the large scale spreading of infectious diseases. In Proceedings of the APS March Meeting, Portland, OR, USA, 15–19 March 2010.
6. Leskovec, J.; Horvitz, E. Planetary-scale views on a large instant-messaging network. In Proceedings of the 17th International Conference on World Wide Web, Beijing, China, 21–25 April 2008.
7. Vaccari, A.; Liu, L.; Biderman, A. A holistic framework for the study of urban traces and the profiling of urban processes and dynamics. In Proceedings of the 2009 12th International IEEE Conference on Intelligent Transportation Systems, St. Louis, MO, USA, 4–7 October 2009; pp. 1–6.
8. Gang, P.; Qi, G.; Wu, Z. Land-Use Classification Using Taxi GPS Traces. *IEEE Trans. Intell. Transp. Syst.* **2013**, *14*, 113–123.
9. Yuan, N.J.; Zheng, Y.; Xie, X.; Wang, Y.; Zheng, K.; Xiong, H. Discovering Urban Functional Zones Using Latent Activity Trajectories. *IEEE Trans. Knowl. Data Eng.* **2015**, *27*, 712–725. [[CrossRef](#)]

10. Feng, L.U.; Kang, L.; Jie, C. Research on Human Mobility in Big Data Era. *J. Geo-Inf. Sci.* **2014**, *16*, 665–672.
11. Ding, M.; Toshihiro, O.; Takuya, O. Exploring the heterogeneity of human urban movements using geo-tagged tweets. *Int. J. Geogr. Inf. Sci.* **2020**, *34*, 2475–2496.
12. Scellato, S.; Musolesi, M.; Mascolo, C. *NextPlace: A Spatio-Temporal Prediction Framework for Pervasive Systems*; Springer: Berlin/Heidelberg, Germany, 2011.
13. Liao, D.; Liu, W.; Zhong, Y. Predicting Activity and Location with Multi-task Context Aware Recurrent Neural Network. In Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, Stockholm, Sweden, 13 July 2018.
14. Li, M.; Shi, X.; Li, X. Integration of spatialization and individualization: The future of epidemic modelling for communicable diseases. *Ann. GIS* **2020**, *26*, 219–226. [\[CrossRef\]](#)
15. Bin, C.; Yi, M. Real-Time Estimation of Population Exposure to PM2.5 Using Mobile- and Station-Based Big Data. *Int. J. Environ. Res. Public Health* **2018**, *15*, 573.
16. Cho, E.; Myers, S.A.; Leskovec, J. Friendship and mobility: User movement in location-based social networks. In Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, CA, USA, 21–24 August 2011.
17. Li, X. Prediction of urban human mobility using large-scale taxi traces and its applications. *Front. Comput. Sci.* **2012**, *6*, 111–121.
18. Liang, V.C. Mercury: Metro density prediction with recurrent neural network on streaming CDR data. In Proceedings of the 2016 IEEE 32nd International Conference on Data Engineering (ICDE), Helsinki, Finland, 16–20 May 2016.
19. Hoang, M.X.; Zheng, Y.; Singh, A.K. FCCF: Forecasting citywide crowd flows based on big data. In Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Burlingame, CA, USA, 31 October–3 November 2016.
20. Andris, C. Integrating social network data into GISystems. *Int. J. Geogr. Inf. Sci.* **2016**, *30*, 2009–2031. [\[CrossRef\]](#)
21. Shaw, S.L.; Yu, H. A GIS-based time-geographic approach of studying individual activities and interactions in a hybrid physical–virtual space. *J. Transp. Geogr.* **2009**, *17*, 2. [\[CrossRef\]](#)
22. Crivellari, A.; Beinat, E. From motion activity to geo-embeddings: Generating and exploring vector representations of locations, traces and visitors through large-scale mobility data. *ISPRS Int. J. Geo-Inf.* **2019**, *8*, 134. [\[CrossRef\]](#)
23. Li, J.; Liu, H.; Guo, W.; Chen, X. A spatio-temporal network for human activity prediction based on deep learning. *Acta Geod. Cartogr. Sin.* **2021**, *50*, 522–531.
24. Chen, J. Fine-grained prediction of urban population using mobile phone location data. *Int. J. Geogr. Inf. Sci.* **2018**, *32*, 1770–1786. [\[CrossRef\]](#)
25. Wang, S.; Yao, Z.; Yang, S. Discovering Urban Travel Demands Through Dynamic Zone Correlation in Location-Based Social Networks. In *Joint European Conference on Machine Learning & Knowledge Discovery in Databases*; Springer: Cham, Switzerland, 2018.
26. Zhu, D. Understanding place characteristics in geographic contexts through graph convolutional neural networks. *Ann. Am. Assoc. Geogr.* **2020**, *110*, 408–420. [\[CrossRef\]](#)
27. Castells, M. *Rise of the Network Society: The Information Age: Economy, Society and Culture*; Blackwell Publishers, Inc.: Cambridge, MA, USA, 1996.
28. Xiu, C.L.; Wei, Y. *City and Regional Structure from the View of “Space of Flows”*; Science Press: Beijing, China, 2015.
29. Pei, T.; Shu, H.; Guo, S.H. The concept and classification of spatial patterns of geographical flow. *J. Geo-Inf. Sci.* **2020**, *22*, 30–40.
30. Zheng, Y. Trajectory data mining: An overview. *ACM Trans. Intell. Syst. Technol. (TIST)* **2015**, *6*, 29. [\[CrossRef\]](#)
31. Deng, D.; Shahabi, C.; Demiryurek, U. Latent Space Model for Road Networks to Predict Time-Varying Traffic. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 1525–1534.
32. Deveaud, R.; Albakour, M.D.; Macdonald, C. Experiments with a Venue-Centric Model for Personalised and Time-Aware Venue Suggestion. In Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, Melbourne, Australia, 19–23 October 2015.
33. Smola, A.J.; Schölkopf, B. A tutorial on support vector regression. *Stats Comput.* **2014**, *14*, 199–222. [\[CrossRef\]](#)
34. Zhang, J.; Zheng, Y.; Qi, D. Deep Spatio-Temporal Residual Networks for Citywide Crowd Flows Prediction. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2016.
35. Ren, Y. A hybrid integrated deep learning model for the prediction of citywide spatio-temporal flow volumes. *Int. J. Geogr. Inf. Sci.* **2020**, *34*, 4. [\[CrossRef\]](#)
36. Yao, H.; Wu, F.; Ke, J. Deep multi-view spatial-temporal network for taxi demand prediction. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32, pp. 2588–2595.
37. Zhao, L.; Song, Y.; Zhang, C.; Liu, Y.; Wang, P.; Lin, T.; Deng, M.; Li, H. T-GCN: A Temporal Graph Convolutional Network for Traffic Prediction. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 3848–3858. [\[CrossRef\]](#)
38. Yan, X.; Ai, T.; Yang, M.; Yin, H. A graph convolutional neural network for classification of building patterns using spatial vector data. *ISPRS J. Photogramm. Remote Sens.* **2019**, *150*, 259–273. [\[CrossRef\]](#)
39. Zhu, D.; Huang, Z.; Shi, L.; Wu, L.; Liu, Y. Inferring spatial interaction patterns from sequential snapshots of spatial distributions. *Int. J. Geogr. Inf. Sci.* **2017**, *32*, 783–805. [\[CrossRef\]](#)

- 
40. Bruna, J.; Zaremba, W.; Szlam, A.; Lecun, Y. Spectral networks and locally connected networks on graphs. In Proceedings of the International Conference on Learning Representations (ICLR), Banff, AB, Canada, 14–16 April 2014.
  41. Kipf, T.N.; Welling, M. Semi-supervised classification with graph convolutional networks. In Proceedings of the International Conference on Learning Representations (ICLR), Toulon, France, 24–26 April 2017.
  42. Hoang, N.T. Takanori Maehara, Revisiting Graph Neural Networks: Graph Filtering Perspective. In Proceedings of the 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 10–15 January 2021.
  43. Chung, J.; Gulcehre, C.; Cho, K. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv* **2014**, arXiv:1412.3555.
  44. Huang, Q.; Wong, D.W.S. Modeling and visualizing regular human mobility patterns with uncertainty: An example using Twitter data. *Ann. Assoc. Am. Geogr.* **2015**, *105*, 1179–1197. [[CrossRef](#)]
  45. Cheng, Z.; Caverlee, J.; Lee, K.; Sui, D. Exploring millions of footprints in location sharing services. In Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media, Barcelona, Spain, 17–21 July 2011; Volume 5.
  46. Liu, J.; Guan, W. A summary of traffic flow forecasting methods. *Highway Transp. Res. Dev.* **2004**, *21*, 82–85.
  47. Salton, G.; Buckley, C. Term-weighting approaches in automatic text retrieval. *Inf. Process. Manag.* **1988**, *24*, 513–523. [[CrossRef](#)]
  48. Bahdanau, D.; Cho, K.; Bengio, Y. Neural Machine Translation by Jointly Learning to Align and Translate. In Proceedings of the International Conference on Learning Representations, San Diego, CA, USA, 7–9 May 2015.