Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

A distributed EEMDN-SABiGRU model on Spark for passenger hotspot prediction*#

Dawen XIA†‡[1], Jian GENG[1], Ruixi HUANG[1], Bingqi SHEN[1], Yang HU[2], Yantao LI[3], Huaqing LI†‡[4]

[1]*College of Data Science and Information Engineering, Guizhou Minzu University, Guiyang 550025, China*

[2]*Department of Automotive Engineering, Guizhou Traffic Technician and Transportation College, Guiyang 550008, China*

[3]*College of Computer Science, Chongqing University, Chongqing 400044, China*

[4]*College of Electronic and Information Engineering, Southwest University, Chongqing 400715, China*

†E-mail: dwxia@gzmu.edu.cn; huaqingli@swu.edu.cn

Received Dec. 5, 2022; Revision accepted Apr. 11, 2023; Crosschecked Aug. 3, 2023

Abstract: To address the imbalance problem between supply and demand for taxis and passengers, this paper proposes a distributed ensemble empirical mode decomposition with normalization of spatial attention mechanism based bi-directional gated recurrent unit (EEMDN-SABiGRU) model on Spark for accurate passenger hotspot prediction. It focuses on reducing blind cruising costs, improving carrying efficiency, and maximizing incomes. Specifically, the EEMDN method is put forward to process the passenger hotspot data in the grid to solve the problems of non-smooth sequences and the degradation of prediction accuracy caused by excessive numerical differences, while dealing with the eigenmodal EMD. Next, a spatial attention mechanism is constructed to capture the characteristics of passenger hotspots in each grid, taking passenger boarding and alighting hotspots as weights and emphasizing the spatial regularity of passengers in the grid. Furthermore, the bi-directional GRU algorithm is merged to deal with the problem that GRU can obtain only the forward information but ignores the backward information, to improve the accuracy of feature extraction. Finally, the accurate prediction of passenger hotspots is achieved based on the EEMDN-SABiGRU model using real-world taxi GPS trajectory data in the Spark parallel computing framework. The experimental results demonstrate that based on the four datasets in the 00-grid, compared with LSTM, EMD-LSTM, EEMD-LSTM, GRU, EMD-GRU, EEMD-GRU, EMDN-GRU, CNN, and BP, the mean absolute percentage error, mean absolute error, root mean square error, and maximum error values of EEMDN-SABiGRU decrease by at least 43.18%, 44.91%, 55.04%, and 39.33%, respectively.

Key words: Passenger hotspot prediction; Ensemble empirical mode decomposition (EEMD); Spatial attention mechanism; Bi-directional gated recurrent unit (BiGRU); GPS trajectory; Spark

https://doi.org/10.1631/FITEE.2200621 **CLC number:** TP39

1 Introduction

With the rapid development of data technology, mobile trajectory big data analytics has become a research hotspot in urban computing and smart cities (Batty et al., 2012; Zheng Y et al., 2014; Zheng Y, 2017). In urban transportation networks, Global Positioning System (GPS) equipped taxis play an

‡ Corresponding authors

* Project supported by the National Natural Science Foundation of China (Nos. 62162012, 62173278, and 62072061), the Science and Technology Support Program of Guizhou Province, China (No. QKHZC2021YB531), the Natural Science Research Project of Department of Education of Guizhou Province, China (Nos. QJJ2022015 and QJJ2022047), the Science and Technology Foundation of Guizhou Province, China (Nos. QKHJCZK2022YB195, QKHJCZK2022YB197, and QKHJCZK2023YB143), the Scientific Research Platform Project of Guizhou Minzu University, China (No. GZMUSYS202104), and the 7th Batch High-Level Innovative Talent Project of Guizhou Province, China

Electronic supplementary materials: The online version of this article (https://doi.org/10.1631/FITEE.2200621) contains supplementary materials, which are available to authorized users

ORCID: Dawen XIA, https://orcid.org/0000-0002-0151-9643; Huaqing LI, https://orcid.org/0000-0001-6310-8965

essential role in our daily life, and the data extracted from the massive taxi GPS trajectories can effectively reflect valuable information, such as passenger hotspot distribution (Bi et al., 2021), passenger travel pattern (Gong et al., 2016), taxi cruising pattern (Xu et al., 2017; Xia et al., 2021a; Zhang WY et al., 2022), and traffic flow distribution (Ali et al., 2021; Seng et al., 2021; Xia et al., 2021b). The practical information mined from taxi mobile trajectory data can provide valuable decisions for passengers, taxi drivers, and traffic managers. Simultaneously, in terms of taxis carrying passengers, some passengers have difficulty in finding taxis in certain places, and some taxi drivers have difficulty in searching for passengers. Moreover, this imbalance between supply and demand may cause severe traffic congestion, wasted resources, decreased profits, and reduced passenger satisfaction (Zheng LJ et al., 2018). Therefore, data interpretation, data manipulation, and data value extraction with big data techniques have become critical issues for intelligent transport systems (ITSs) (Engelbrecht et al., 2015; Zhu et al., 2018). Given this, it is necessary to predict potential passenger hotspots to reduce fuel consumption and cruising time (Dong et al., 2017).

Passenger hotspot prediction has been a hot research direction in smart cities, and researchers have conducted many studies in recent years, which can be divided into two types. Traditional time-series methods are used to predict passenger hotspots (Li XL et al., 2012; Jamil and Akbar, 2017; Qu et al., 2019), but they fail to consider non-stationary series and the problem of reduced prediction accuracy caused by excessive differences in values, as taxi GPS trajectory data are non-stationary spatiotemporal data with variability between values. Therefore, it is vital to consider non-stationary data, and their variability is essential. With the rapid development of neural networks, many researchers have used neural networks such as long short-term memory (LSTM) and gated recurrent unit (GRU) to predict passenger hotspots. However, these models are not applied in passenger hotspot prediction (Kim et al., 2020; Li XF et al., 2020; Luo et al., 2021; Yang et al., 2021), and do not consider the backward and forward contextual information either. Furthermore, researchers have used attention mechanisms in neural network models to enhance prediction accuracy. For example, the self-attention mechanism and soft attention mechanism focus only on the correlation of the data without considering the spatial correlation between the map road network and passenger hotspots. In addition, several researchers have employed ensemble empirical mode decomposition (EEMD) combined with traditional models for short-term metro passenger flow and ship movement prediction (Nie et al., 2020; Liu XP et al., 2022), as well as combining neural networks for traffic flow (Gao et al., 2020), metro passenger flow (Cao et al., 2022), and waiting time (Xia et al., 2022a) prediction. However, they combined two methods but did not implement these methods in the Spark distributed framework.

To address the above problems, we propose a distributed ensemble empirical mode decomposition with normalization of spatial attention mechanism based bi-directional gated recurrent unit (EEMDN-SABiGRU) model on Spark for passenger hotspot prediction. Specifically, the non-smooth data are smoothed using the EEMDN method. Then, the spatial attention mechanism is used to capture the correlation of passenger hotspots between grids. Therefore, the BiGRU algorithm is fused to predict passenger hotspots, and the prediction results are inversely normalized and superimposed. Finally, we evaluate EEMDN-SABiGRU based on GPS trajectory data of taxis in Beijing, China. The results indicate that the prediction accuracy of EEMDN-SABiGRU is superior to those of the comparable models.

The main contributions of this work are summarized as follows:

1. An EEMDN method is proposed to reduce the influence of non-stationary time series on the prediction performance and to solve the intrinsic mode function (IMF) confusion problem of the EMD algorithm.

2. A spatial attention mechanism is constructed to capture spatial correlation, extract the number of passengers getting on and off in the grid, form the grid's spatial weights, and improve the performance of passenger hotspot prediction.

3. A BiGRU model is incorporated to deal with the problem that GRU can obtain only forward contextual information but ignores backward contextual information, which improves the accuracy of feature extraction.

4. With the proposed EEMDN-SABiGRU model and the big taxi GPS trajectory data in Beijing, China, passenger hotspots are successfully

predicted in the Spark framework. The experimental results demonstrate that the prediction accuracy of EEMDN-SABiGRU is significantly higher than those of LSTM, EMD-LSTM, EEMD-LSTM, GRU, EMD-GRU, EEMD-GRU, EMDN-GRU, convolutional neural network (CNN), and backpropagation (BP).

2 Related works

In this section, we briefly introduce the works related to passenger hotspots and then analyze the problems. Existing works consist mainly of the traditional time-series methods and neural network methods.

2.1 Time-series methods

The time-series analysis method is widely applied to passenger hotspot prediction. Jamil and Akbar (2017) conducted time-series analysis using an automated ARIMA model to forecast hotspot areas for passengers based on historical spatiotemporal data provided by taxi companies. Li XL et al. (2012) proposed a model to delineate urban hotspots and a method based on an improved ARIMA model to predict the number of passengers in urban hotspots in time and space. Qu et al. (2019) investigated a profitable and graphical taxi route recommendation method called adaptive shortest expected cruising route (ASECR), which uses Kalman filter prediction to obtain probability and capacity locality. Xia et al. (2022b) developed a parallel GS-SVM algorithm based on the Spark framework to predict taxi passenger hotspots. The EMD method has also been widely used because of its significant advantage in processing non-stationary data (Huang et al., 2019). Yao et al. (2016) employed the EMD-PSO-SVM algorithm to predict safety conditions. Nie et al. (2020) improved the EMD and SVR algorithms for short-term ship motion prediction. In addition, the EEMD method responds to the EMD method's modal confounding problem. Liu XP et al. (2022) put forward three hybrid forecasting models, EEMD-ARIMA, EEMD-BP, and EEMD-SVM, for predicting short-term urban metro demand changes. Wang et al. (2022) employed an improved EEMD method for active power filter (APF) detection. Qin et al. (2020) proposed an EEMD-LPP model for carbon price prediction. Cheng et al. (2021) presented an EEMD-SVD-LWT denoising algorithm for atmospheric LiDAR. Jiang et al. (2014) used an EEMD-GSVM model for short-term prediction to address the high-speed rail passenger flow prediction problem.

2.2 Neural network methods

In recent years, neural network methods have attracted significant attention in passenger hotspot prediction. Yang et al. (2021) proposed a new wave-LSTM model based on LSTM and wavelets to predict passenger flow. Li XF et al. (2020) investigated a fast correlation filter and an LSTM based on wavelet transform to predict passenger demand in different regions at different time intervals. Kim et al. (2020) designed an interpretable deep learning model to evaluate a quota system that balanced two demanded modes. A two-stage interpretable machine learning modeling framework LSTM was developed through a linear regression (LR) model combined with a long short-term hierarchical neural network. Luo et al. (2021) proposed a multi-task deep learning (MTDL) model to predict short-term taxi demand at multiple regional levels to reduce hybrid bus emission (HBE) while improving efficiency. Li ML et al. (2021) designed a new predictive energy management strategy based on passenger forecasting and exhaust emission optimization. Ou et al. (2020) presented a new deep learning framework, STP-TrellisNets, which augments the emerging temporal convolution framework (TrellisNet) to predict subway station passenger flows accurately by spatiotemporal forecasting. Saadallah et al. (2020) proposed BRIGHT for forecasting demand using a supervised learning framework for perceivable demand. BRIGHT aims to provide accurate forecasts for demand in short term through an innovative time-series analysis method to deal with different types of conceptual drift. Zhou et al. (2020) put forward the ST-attention model, which uses a multi-output strategy, but does not employ recurrent neural network (RNN) units of learning, to determine ridership demand and periods in key projected city areas during special periods using a spatiotemporal attention mechanism. In addition, the EEMD method was often combined with neural network methods such as the EEMD-LSTM algorithm, which was proposed to predict surface temperature (Zhang XK et al., 2018). Rezaei et al. (2021) constructed the

CEEMD-CNN-LSTM and EMD-CNN-LSTM models to predict financial time series. Yu et al. (2021) constructed an EEMD-Conv3d method for soil temperature prediction. Niu et al. (2021) used an EEMD combined with RNN for landslide prediction problems. Liu J et al. (2020) developed an EEMD-DBN model for urban short-term traffic flow prediction.

In the above studies, researchers used time-series methods to predict passenger hotspots. However, traditional time-series methods do not consider the impact of non-stationary series on prediction accuracy. Furthermore, although the EMD method can reduce the non-stationarity of time series, it still has the problems of end effects and modal confounding. When using neural networks for passenger hotspot prediction, researchers employed the self-attention mechanism to focus on the correlation of the data without considering the spatial correlation between the map road network and passenger hotspots. Although GRU can solve the problems of gradient disappearance and gradient explosion in RNN, the information-dependent GRU method ignores the information context in the road network, with high complexity and a long prediction time. In addition, few researchers used the EEMD method combined with neural network models for passenger hotspot prediction. To this end, in this paper we present a distributed EEMDN-SABiGRU model on Spark to accurately predict passenger hotspots.

3 EEMDN-SABiGRU model

In this section, we describe the EEMDN-SABiGRU model in detail.

3.1 Model overview

The prediction framework based on a distributed EEMDN-SABiGRU model includes data preprocessing, model construction, and model implementation as shown in Fig. 1. In data preprocessing, the taxi GPS trajectory data are processed by data extracting, data sorting, grid mapping, and data counting. Then, in model construction, the EEMD algorithm with the normalization method is employed to obtain a finite number of IMFs and a residual (Res) sequence, and the BiGRU algorithm with a spatial attention mechanism is used to construct the EEMDN-SABiGRU model. The prediction results are superimposed by inverse normalization. Finally, in model implementation, the EEMDN-SABiGRU model is implemented on the Spark parallel computing framework.

3.2 Data preprocessing

When collecting taxi GPS data, there are problems such as equipment failure and signal delay, which can cause the collected data to be wrong or missing. For example, some taxi drivers do not update the passenger status in time after picking up a passenger. The data are not collected when the

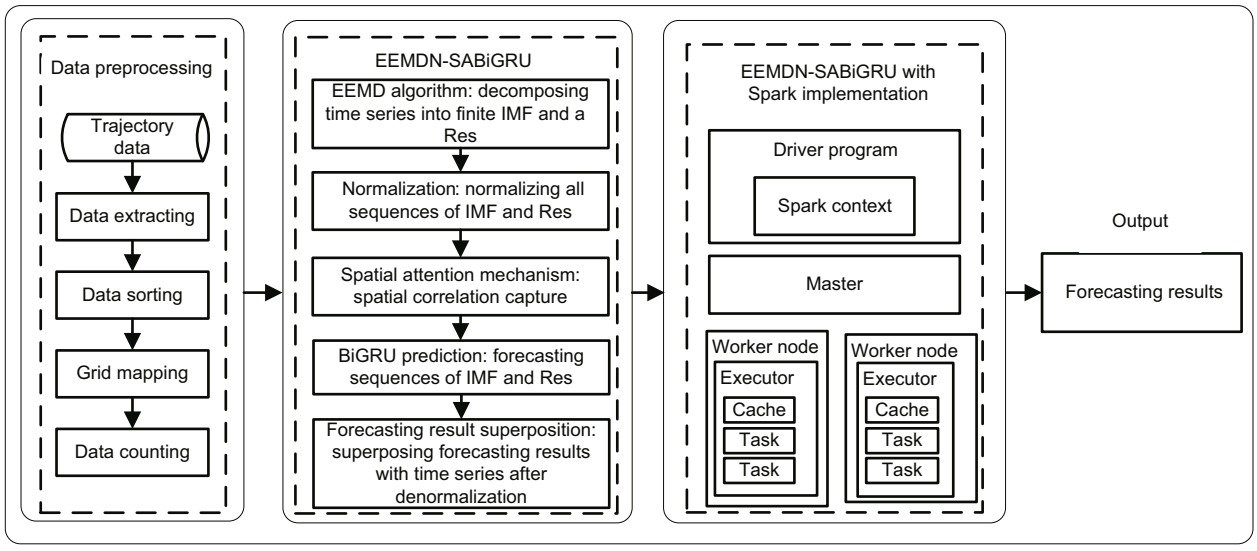

Fig. 1 A distributed EEMDN-SABiGRU model on Spark

signal is mid-range, and the vehicle passes through a long tunnel. Therefore, it is necessary to remove errors and fill in missing data when processing data to improve the accuracy and reliability of prediction. The process of data preprocessing is illustrated in Fig. 2.

Step 1: data extracting. We first store the data in the Hadoop distributed file system (HDFS) and convert the data into a resilient distributed dataset (RDD) on Spark. We then split the RDD, eliminate the blank and wrong data, and finally extract the required data (taxi identity document (ID), operation status, time, longitude, and latitude). Details are specified in Algorithm 1.

Step 2: data sorting. From the data obtained at step 1, the duplicate IDs are filtered, and the complete 011 passenger-carrying events in the operation status (0 means empty and 1 means passenger-carrying) are extracted and sorted in chronological order. Details are illustrated in Algorithm 2.

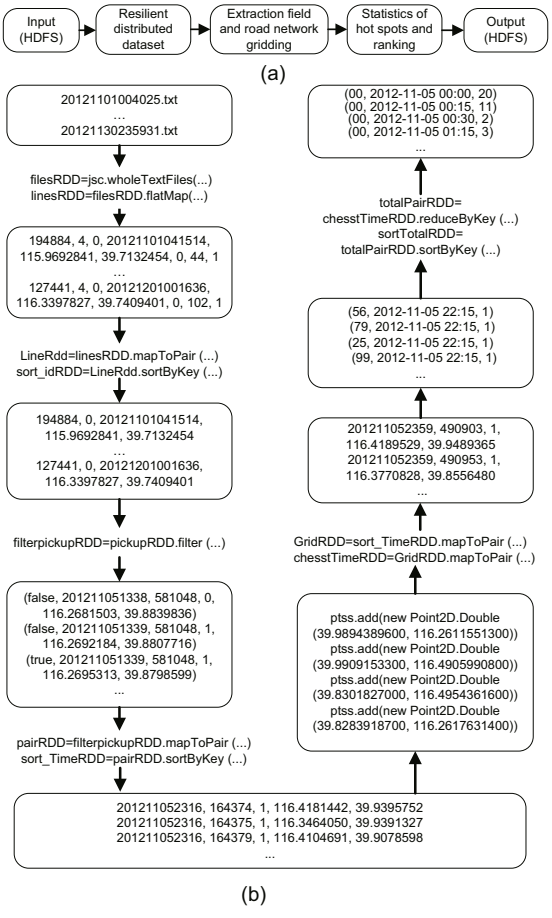

Fig. 2 Process of data preprocessing: (a) data preprocessing; (b) data flow

Step 3: grid mapping. In this work, the latitude and longitude ranges of 39.828 391 87–39.990 915 33 and 116.261 155 13–116.495 436 16 are selected, respectively. The sorted data are mapped or matched into these latitude and longitude ranges using the sorted data, and the latitude and longitude of the sorted data gridded as 10×10 are illustrated in Fig. 3. Details are described in Algorithm 3.

Step 4: data counting. We use the data after mapping the grids, divide them into intervals of 15 min, and count the taxi boarding hotspot data within the same grid at 15-min intervals. Details are given in Algorithm 4.

3.3 Model construction

The process of the EEMDN-SABiGRU model includes three steps: design of the EEMDN

Algorithm 1 Data extracting

Input: GPS trajectory data of taxis
1: **if** GPS status=1 and direction<250 and speed<250 **then**
2: **if** operation status=0 || 1 **then**
3: **Put in** linesRDD
4: **end if**
5: **end if**
6: **for** each linesRDD **do**
7: Extract key: taxi ID, time, value, required field
8: Sort by ascending taxi ID
9: **end for**
Output: sort_idRDD

Algorithm 2 Data sorting

Input: sort_idRDD
1: **if** the ID is fixed **then**
2: Find a sequence with operating condition 011
3: **if** three conditions = 011 **then**
4: 011 is considered a pick-up point
5: **end if**
6: **end if**
7: **Put in** pick-up RDD
8: Data with operating condition 1
9: Sort the data in ascending order
Output: sort_TimeRDD

Algorithm 3 Grid matching

Input: sort_TimeRDD
1: Determine the latitude and longitude intervals
2: Determine the rectangular box by latitude and latitude intervals
3: Divide the rectangular frame into 10×10 grids
4: **for** each sort_TimeRDD **do**
5: Determine which grid the latitude and longitude are in and mark it (GridRDD)
6: **end for**
Output: GridRDD

Fig. 3 Road network grid: (a) 10×10 grid; (b) road network with 10×10 grid

Algorithm 4 Data counting

Input: Grid RDD
1: Divide the time into intervals of 15 min
2: Mark the number of hotspots in each grid
3: **Put in** chesstTimeRDD
4: **for** each chesstTimeRDD **do**
5: Count the number of hotspots with 15-min intervals on the same grid
6: Sort the same grid in chronological order
7: **end for**
Output: pick-up hotspotsRDD

algorithm, integration of the BiGRU model, and construction of the spatial attention mechanism.

Step 1: design of the EEMDN algorithm. EEMD, an upgraded algorithm of EMD, smooths the abrupt changes on the time scale by adding white noise to the original signal sequence and adaptively maps the signals at different scales to a suitable reference scale using the uniform distribution of the white noise spectrum. Then, the white noise signal is inputted into EMD for multiple decompositions to obtain the average result, eliminating the noise effect. Finally, the IMF and Res sequences containing a single time scale are obtained, and the IMF and Res are mapped in [0, 1] and inputted to the BiGRU model for prediction. The decomposition process is composed of four sub-steps:

(1) Set the total average number of times as M, and add the white noise amplitude signal $n_i(t)$ with a standard normal distribution to the original signal $x(t)$ to obtain an additional noise signal, which is defined as

$$x_i(t) = x(t) + n_i(t), \quad i = 1, 2, \cdots, M, \tag{1}$$

where $x(t)$ is the original signal, $n_i(t)$ is the i^{th} white noise sequence, and $x_i(t)$ is the i^{th} additional noise signal.

(2) The noisy signal $x_i(t)$ is decomposed by EMD, and the sum of IMFs is obtained, defined as

$$x_i(t) = \sum_{j=1}^{J} c_{i,j}(t) + r_{i,j}(t), \tag{2}$$

where $c_{i,j}(t)$ is the j^{th} IMF decomposed after the i^{th} white noise, $r_{i,j}(t)$ is the residual function, and J is the number of IMF sequences.

(3) Sub-steps 1 and 2 are repeated M times, and white noise signals with different amplitudes are added for each decomposition to obtain the IMF set, which is defined as

$$c_{1,j}(t), c_{2,j}(t), \cdots, c_{M,j}(t). \tag{3}$$

(4) The above IMFs are averaged to obtain the final IMF result after EEMD, as shown in Eq. (4), and the IMF and Res obtained by EEMD decomposition are mapped in [0, 1].

$$c_j(t) = \frac{1}{M} \sum_{i=1}^{M} c_{i,j}(t), \tag{4}$$

where $c_j(t)$ is the j^{th} decomposed IMF.

Step 2: integration of the BiGRU model. The BiGRU model is an improvement of the GRU model, consisting of two unidirectional and opposite GRUs, and the GRU model is defined in Eqs. (5)–(8):

$$\boldsymbol{z}_t = \sigma(W_{\boldsymbol{z}}[\boldsymbol{h}_{t-1}, \boldsymbol{x}_t]), \tag{5}$$

$$\boldsymbol{r}_t = \sigma(W_{\boldsymbol{r}}[\boldsymbol{h}_{t-1}, \boldsymbol{x}_t]), \tag{6}$$

$$\tilde{\boldsymbol{h}}_t = \tan \boldsymbol{h}\left(W[\boldsymbol{r}_t \boldsymbol{h}_{t-1}, \boldsymbol{x}_t]\right), \tag{7}$$

$$\boldsymbol{h}_t = 1 - \boldsymbol{z}_t \boldsymbol{h}_{t-1} + \boldsymbol{z}_t \tilde{\boldsymbol{h}}_t, \tag{8}$$

where $\boldsymbol{z}_t$ is the updated gate, $\boldsymbol{r}_t$ represents the reset gate, $\boldsymbol{h}_{t-1}$ denotes the output value at time $t-1$, $\boldsymbol{x}_t$ is the input value at time t, σ and tanh represent the activation functions, $W_{\boldsymbol{z}}$ is the updated gate weight, $W_{\boldsymbol{r}}$ denotes the reset gate weight, $\tilde{\boldsymbol{h}}_t$ represents the output value of tanh, and $\boldsymbol{h}_t$ is the output of the results.

The structure of the BiGRU model (Fig. 4) is composed of two GRUs facing in opposite directions. The hidden layer state of BiGRU at time t is the weighted sum of $\overrightarrow{\boldsymbol{h}}_{t-1}$ and $\overleftarrow{\boldsymbol{h}}_{t-1}$, defined in Eqs. (9)–(11):

$$\overrightarrow{\boldsymbol{h}}_t = \text{GRU}(\boldsymbol{x}_t, \overrightarrow{\boldsymbol{h}}_{t-1}), \tag{9}$$

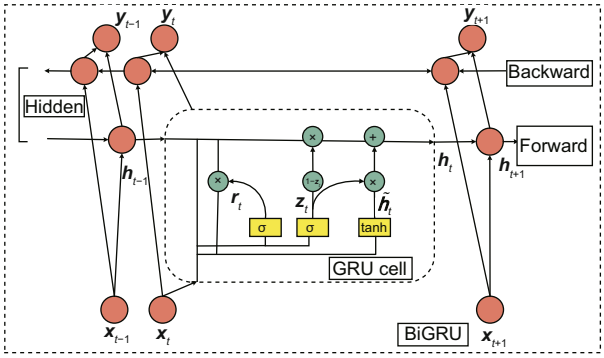

Fig. 4 Structure of the BiGRU model

$$\overleftarrow{\boldsymbol{h}}_t = \text{GRU}(\boldsymbol{x}_t, \overleftarrow{\boldsymbol{h}}_{t-1}), \tag{10}$$

$$\boldsymbol{h}_t = \boldsymbol{w}_t \overrightarrow{\boldsymbol{h}}_t + \boldsymbol{v}_t \overleftarrow{\boldsymbol{h}}_t + \boldsymbol{b}_t, \tag{11}$$

where GRU(·) represents the nonlinear transformation of the input word embeddings, $\overrightarrow{\boldsymbol{h}}_t$ is the forward output result, $\overleftarrow{\boldsymbol{h}}_t$ is the reverse output result, $\overrightarrow{\boldsymbol{h}}_{t-1}$ and $\overleftarrow{\boldsymbol{h}}_{t-1}$ are the positive and negative output values at time $t-1$, respectively, and the word embedding is encoded in the corresponding GRU hidden layer states. $\boldsymbol{b}_t$ represents the deviation corresponding to the hidden layer state at time t, and $\boldsymbol{w}_t$ and $\boldsymbol{v}_t$ are the weights of the forward hidden layer state $\overrightarrow{\boldsymbol{h}}_t$ and the reverse hidden layer state $\overleftarrow{\boldsymbol{h}}_t$ corresponding to BiGRU at time t, respectively.

Step 3: construction of the spatial attention mechanism. The spatial attention mechanism is an adaptive spatial region selection mechanism, through which the BiGRU network is guided to pay more attention to the significant spatial regions on the grid graph. We take the number of passengers getting on and off the grid as the weight. The input features are processed by MaxPool and AvgPool, then convoluted, and finally inputted to the Sigmoid function for activation, as defined in Eqs. (12) and (13). The process of building the spatial mechanism module is plotted in Fig. 5.

$$\boldsymbol{M}(\boldsymbol{F}) = [\text{AvgPool}(\boldsymbol{F}), \text{MaxPool}(\boldsymbol{F})], \tag{12}$$

$$\boldsymbol{M}_{\text{S}}\boldsymbol{F} = \sigma(f(\boldsymbol{M}(\boldsymbol{F}))), \tag{13}$$

where $\boldsymbol{F}$ is a characteristic graph, AvgPool represents average pooling, MaxPool denotes maximum pooling, f is the convolution operation, σ represents the Sigmoid activation function, and $\boldsymbol{M}_{\text{S}}\boldsymbol{F}$ denotes the spatial attention parameter matrix.

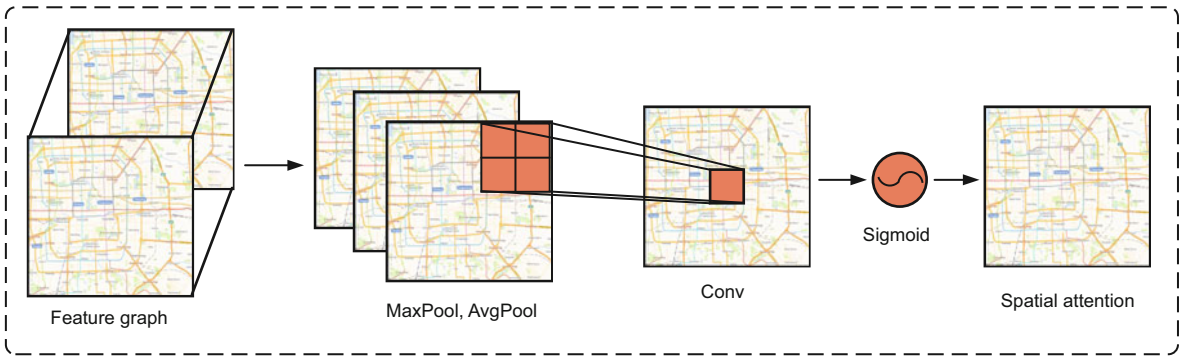

Fig. 5 Spatial attention mechanism

3.4 Model implementation

In this work, the batch-size and epoch in the BiGRU structure are set as 4 and 180 respectively, and the numbers of neural network layers and neurons are 2 and 16 respectively. The process of EEMDN-SABiGRU with Spark implementation is shown in Fig. 6.

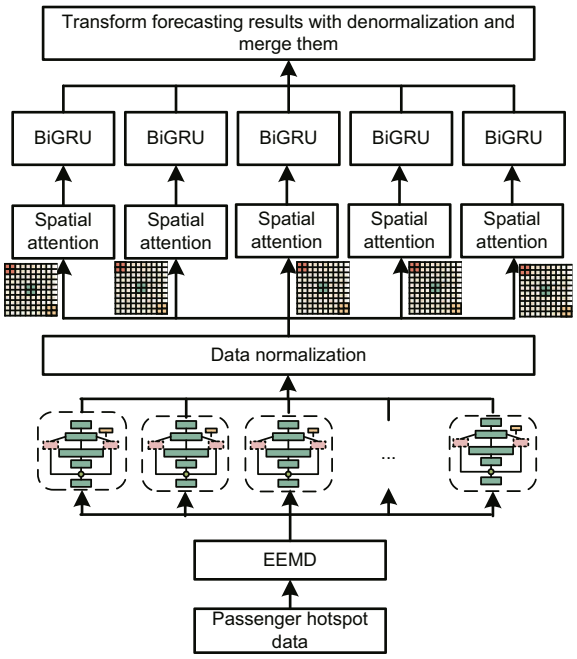

Fig. 6 Implementation process of the EEMDN-SABiGRU model

Step 1: data decomposition. The EEMD algorithm decomposes the 15-min interval passenger hotspot data in the grid to obtain a finite number of IMFs and a Res sequence.

Step 2: data normalization. Normalize the IMF and Res, and map them to a range of [0, 1].

Step 3: model prediction. The normalized IMF and Res are inputted into the BiGRU model with spatial attention for prediction.

Step 4: result superposition. The prediction values are denormalized, and the values are summed to obtain the final prediction results.

4 Experiments

In this section, we validate the performance of the proposed EEMDN-SABiGRU model for passenger hotspot prediction with real-world taxi trajectory data from a case study. Specifically, the fit test is performed after EEMD and the original sequence. Then, the normalization test is executed on the sequence prediction results obtained from EEMD, and the effects before and after normalization are analyzed. Next, the prediction performances of 1-, 10-, 20-, and 30-day datasets in the 00-grid are compared using different models, and the results are analyzed in detail. Finally, the robustness of the EEMDN-SABiGRU model is evaluated with the 30-day dataset under different grids.

4.1 Experimental setup

The extensive experiments are performed on a Hadoop distributed platform with a Spark parallel computing framework. The experimental platform is configured with Hadoop 3.1.1 + Spark 2.4.3 + Java + DL4J on Ubuntu 18.04 OS, AMD Ryzen7 4800H, and 8 GB ECC DDR3.

Moreover, the EEMDN-SABiGRU model is compared with LSTM, EMD-LSTM, EEMD-LSTM, GRU, EMD-GRU, EEMD-GRU, EMDN-GRU, CNN, and BP.

4.2 Experimental data

The experimental data come from the GPS trajectory data of 12 000 taxis in Beijing, China, in 2012, which have more than 9.0×10^8 GPS trajectory records (about 50 GB). An example of the dataset is plotted in Fig. 7.

Furthermore, to compare the effectiveness of the EEMDN-SABiGRU model, we divide the dataset into four groups (1-day: November 1; 10-day: November 1–November 10; 20-day: November 1–November 20; 30-day: November 1–November 30), and the time interval of each group is 15 min. In addition, 70% of the data are chosen as the training set, and 30% are used as the test set in all experiments.

4.3 Evaluation metrics

To validate the measures of effectiveness (MOEs) of the EEMDN-SABiGRU model, four metrics, mean absolute percentage error (MAPE), root mean square error (RMSE), mean absolute error (MAE), and maximum error (ME), are employed for evaluation:

$$\mathrm{MAPE}=\frac{1}{n}\sum_{t=1}^{n}\frac{\left|X_t-\hat{X}_t\right|}{X_t}\times100\%, \qquad (14)$$

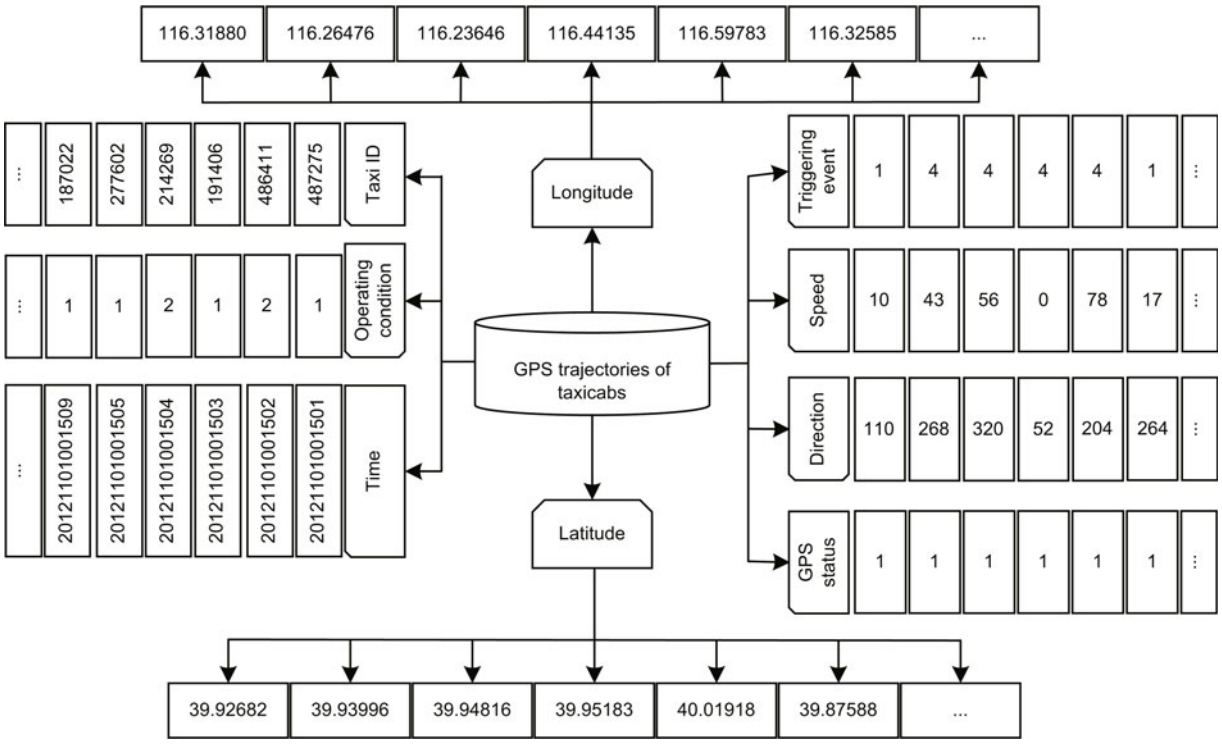

Fig. 7 Taxi GPS trajectory data

$$\mathrm{RMSE}=\sqrt{\frac{1}{n}\sum_{t=1}^{n}\left(X_t-\hat{X}_t\right)^2}, \tag{15}$$

$$\mathrm{MAE}=\frac{1}{n}\sum_{t=1}^{n}\left|X_t-\hat{X}_t\right|, \tag{16}$$

$$\mathrm{ME}=\max_{t=1,2,\ldots,n}\left|X_t-\hat{X}_t\right|, \tag{17}$$

where X_t is the real value of passenger hotspots, $\hat{X}_t$ is the prediction value of passenger hotspots, and n is the total number of samples in the provided time. The MAPE value is used to compare the accuracy of each model. The lower the MAPE value, the higher the prediction accuracy.

4.4 Result analysis

4.4.1 Sequence prediction

EEMD is carried out with the 1- and 10-day datasets, and the decomposed series are tested for goodness-of-fit. The overall trend of the IMF1 series is steeper, because the original series are non-stationary series with significant differences. Next, the EEMD algorithm is repeated by subtracting the IMF1 series from the original series until no IMF series are generated. The 1-day dataset is shown in Fig. 8a, and the results can be fitted perfectly with the original data. To further verify how well the sequences obtained by the EEMD algorithm fit the initial data, the amount of data is increased to 10-day and then the sequences are fitted to the original sequences, as plotted in Fig. 8b. Obviously, the sum of the sequences fits the original dataset well, indicating that the EEMD algorithm does not produce missing data cases with the increased dataset.

4.4.2 Normalization test

Although the EEMD algorithm is suitable for dealing with non-smooth sequences and also solves the modal mixing problem of the EMD algorithm, there will be too many differences in values between IMF sequences during the decomposition process, which will cause BiGRU to fluctuate in prediction and lead to unsatisfactory overall prediction. We employ the normalization method to limit the preprocessed data to a specific range (e.g., [0, 1] or [−1, 1]) and eliminate the adverse effects caused by singular sample data. Therefore, it is an excellent choice to normalize the EEMD sequences. The prediction results of the sequences before and after normalization are illustrated in Tables 1 and 2, respectively.

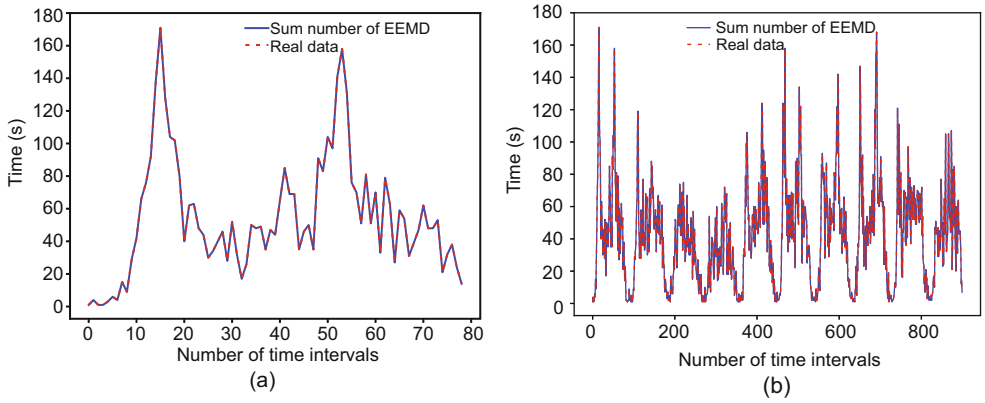

Fig. 8 Data fit test of EEMD: (a) data test on the 1-day dataset; (b) data test on the 10-day dataset

Table 1 Measures of effectiveness values of forecasting results before normalization

IMF	MAPE (%)	MAE	RMSE	ME
IMF1	7.300	1.618	3.102	9.534
IMF2	8.200	0.491	0.701	2.410
IMF3	31.300	0.927	0.946	1.213
IMF4	90.300	6.087	8.479	14.675
IMF5	12.500	8.297	8.351	9.609
IMF6	2.800	1.674	1.808	2.702
IMF7	32.600	12.425	13.760	22.124

Table 2 Measures of effectiveness values of forecasting results after normalization

IMF	MAPE (%)	MAE	RMSE	ME
IMF1	3.600	0.861	1.102	2.177
IMF2	1.300	0.333	0.369	0.657
IMF3	4.000	0.237	0.265	0.412
IMF4	3.900	1.036	1.314	2.696
IMF5	0.025	0.030	0.032	0.047
IMF6	0.020	0.007	0.009	0.027
IMF7	3.400	1.269	1.507	2.889

Tables 1 and 2 show that the normalized sequence prediction results are significantly superior to the pre-normalized sequence prediction results. Moreover, the prediction results before normalization compared with the sequence prediction results after normalization show a minimum reduction of 50.68% and a maximum reduction of 99.80% in MAPE, a minimum reduction of 32.18% and a maximum reduction of 99.64% in MAE, a minimum reduction of 47.36% and a maximum reduction of 99.62% in RMSE, and a minimum reduction of 66.03% and a maximum reduction of 99.51% in ME. Finally, combined with the above analysis, it is concluded that the results of sequence prediction are greatly improved after using the normalization method, so the EEMD algorithm is chosen to decompose and normalize the sequence for prediction.

4.4.3 Prediction results of different days in the 00-grid

We use a 00-grid with 1-, 5-, 10-, 15-, 20-, 25-, and 30-day datasets from the road network for model validation, and compare EEMDN-SABiGRU with LSTM, EMD-LSTM, EEMD-LSTM, GRU, EMD-GRU, EEMD-GRU, EMDN-GRU, CNN, and BP.

The MOE values of different models with seven datasets are shown in Table 3. We conduct fine-grained analysis on four datasets: 1-, 10-, 20-, and 30-day. On the 1-day dataset, it is evident that the fit of the data with large fluctuations is greatly improved after using the EMD algorithm for data decomposition compared to the LSTM model, and the MOE values are greatly reduced. After using the EEMD algorithm, the MAE, RMSE, and ME values are increased, except for the reduced MAPE values, and all of them are lower than those of the LSTM model. Compared with the GRU model, the MOE values decrease significantly after using the EMD algorithm with the EEMD algorithm. Moreover, the prediction accuracy of the EEMD-GRU model is better than that of the EMD-GRU model. It can be concluded that the GRU model combined with the EEMD algorithm is significantly superior to the LSTM algorithm. Therefore, this work employs the GRU model combined with the EEMD algorithm for passenger hotspot prediction. As illustrated in Fig. S1j in the supplementary materials, the prediction values fit the real values well, and the evaluation metrics of EEMDN-SABiGRU are significantly

Table 3 Comparisons of models in different datasets using the 00-grid

Dataset	MOE	LSTM	EMD-LSTM	EEMD-LSTM	GRU	EMD-GRU	EEMD-GRU	EMDN-GRU	CNN	BP	EEMDN-SABiGRU
1-day	MAPE (%)	28.900	12.000	5.700	25.400	10.900	8.100	3.600	31.300	9.500	1.500
	MAE	18.099	2.197	2.453	15.607	3.970	2.657	1.336	14.972	4.571	0.736
	RMSE	23.397	2.508	4.017	20.512	4.756	3.257	1.637	16.028	4.901	0.736
	ME	48.642	6.477	7.943	44.560	9.842	6.029	3.488	25.524	7.834	2.116
5-day	MAPE (%)	12.300	11.500	12.000	10.800	29.300	26.900	6.800	20.800	9.500	3.100
	MAE	3.529	2.197	0.966	3.489	3.626	3.599	1.808	8.843	3.584	0.449
	RMSE	6.503	2.508	1.291	7.002	4.057	3.976	2.518	10.598	4.420	0.563
	ME	38.176	6.477	6.020	42.404	10.926	22.635	9.080	26.963	11.868	1.983
10-day	MAPE (%)	5.000	18.500	8.100	9.500	22.800	21.900	4.400	21.400	10.700	2.500
	MAE	2.378	2.695	1.395	4.395	3.957	2.973	1.990	10.154	3.495	0.705
	RMSE	5.397	4.398	2.477	7.340	6.211	4.213	2.984	12.440	4.437	0.978
	ME	50.297	28.234	18.028	61.684	37.755	21.053	14.305	38.353	15.135	3.392
15-day	MAPE (%)	6.300	18.800	17.500	25.600	31.500	22.300	20.000	16.600	15.700	2.800
	MAE	2.407	2.199	2.009	3.256	2.972	2.156	2.015	8.735	3.652	0.265
	RMSE	4.777	3.756	3.100	6.996	5.109	3.776	3.614	10.633	4.595	0.361
	ME	79.900	19.632	27.407	94.321	31.845	30.907	12.538	39.290	19.754	2.193
20-day	MAPE (%)	5.800	18.300	8.800	16.600	42.500	11.700	16.300	16.100	16.700	1.500
	MAE	1.275	2.729	0.986	3.421	4.528	1.419	1.828	7.950	3.025	0.208
	RMSE	2.137	3.821	1.440	4.891	5.840	2.115	2.559	9.821	3.864	0.294
	ME	21.752	19.704	15.155	33.022	26.693	18.093	12.392	28.836	13.480	1.779
25-day	MAPE (%)	16.100	16.000	12.100	17.100	29.000	56.700	40.800	15.500	17.300	2.000
	MAE	2.166	2.307	1.735	3.560	5.115	4.482	3.520	8.223	3.119	0.362
	RMSE	4.893	4.987	3.698	7.443	6.942	6.636	5.009	10.278	4.080	0.673
	ME	112.576	77.116	63.480	129.499	96.555	84.950	56.937	45.614	23.005	9.526
30-day	MAPE (%)	7.600	24.700	40.400	17.300	68.200	36.000	8.400	16.500	20.600	2.800
	MAE	2.693	4.071	3.850	2.552	9.013	4.464	2.006	9.385	3.290	0.396
	RMSE	8.194	6.094	6.002	10.144	9.989	8.294	4.209	12.105	4.407	0.670
	ME	118.708	64.821	62.146	133.717	82.860	80.284	29.887	51.783	23.545	9.091

lower than those of the comparable models. Among them, based on the 1-day dataset, compared with LSTM, EMD-LSTM, EEMD-LSTM, GRU, EMD-GRU, EEMD-GRU, EMDN-GRU, CNN, and BP, the MAPE value of the EEMDN-SABiGRU model is reduced by 94.81%, 87.50%, 73.68%, 94.09%, 86.24%, 81.48%, 58.33%, 95.21%, and 84.21%, respectively; MAE is reduced by 95.93%, 66.50%, 70.00%, 95.28%, 81.46%, 72.30%, 44.91%, 95.08%, and 83.90%, respectively; RMSE is decreased by 96.85%, 70.65%, 81.68%, 96.41%, 84.52%, 77.40%, 55.04%, 95.41%, and 84.98%, respectively; ME is decreased by 95.65%, 67.33%, 73.36%, 95.25%, 78.50%, 64.90%, 39.33%, 91.71%, and 72.99%, respectively. Therefore, it can be concluded that the EEMDN-SABiGRU model has better prediction performance on the 1-day dataset.

On the 10-day dataset, from Fig. S2 in the supplementary materials, the LSTM, GRU, CNN, and BP models, although more effective in predicting data with relatively minor fluctuations, tend to decrease in accuracy once the fluctuations are large. With the addition of the EMD and EEMD algorithms, the MAPE values of the LSTM and GRU models increase, while the values of MAE, RMSE, and ME decrease. Although the predictions in high and low peaks tend to follow roughly the same trend as the test set, there are some numerical differences between the prediction values and the test set with the EMD-LSTM, EEMD-LSTM, EMD-GRU, and EEMD-GRU models. Moreover, the EMDN-GRU and EEMDN-SABiGRU models fit the test set well after normalization. Therefore, in Table 3, the MOE values of EMDN-GRU and EEMD-GRU are lower than those of the other models. Finally, from Fig. S2j in the supplementary materials, it can be concluded that the EEMDN-SABiGRU model generates accurate predictions for both higher and lower volatility data compared to the LSTM, EMD-LSTM, EEMD-LSTM, GRU, EMD-GRU, EEMD-GRU, EMDN-GRU, CNN, and BP models. From Table 3, based on the 10-day dataset, compared with LSTM, EMD-LSTM, EEMD-LSTM, GRU, EMD-GRU, EEMD-GRU, EMDN-GRU, CNN, and BP,

MAPE is reduced by 50.00%, 86.49%, 69.14%, 73.68%, 89.04%, 88.58%, 43.18%, 88.32%, and 76.64%, respectively; MAE is reduced by 70.35%, 73.84%, 49.46%, 83.96%, 82.18%, 76.29%, 64.57%, 93.06%, and 79.83%, respectively; RMSE is decreased by 81.88%, 77.76%, 60.52%, 86.68%, 84.25%, 76.79%, 67.22%, 92.14%, and 77.96%, respectively; ME is decreased by 93.26%, 87.99%, 81.18%, 94.50%, 91.02%, 83.89%, 76.29%, 91.16%, and 77.59%, respectively. For passenger hotspot prediction on the 10-day dataset, the EEMDN-SABiGRU model can still obtain accurate prediction results.

As observed from Table 3, with the addition of the EMD and EEMD algorithms on the 20-day dataset, the MAPE, MAE, and RMSE values of the LSTM and GRU models increase, while the ME values decrease. When the EMD algorithm is replaced by the EEMD algorithm, all the MOE values are reduced, proving that the EEMD algorithm can compensate for the EMD algorithm's modal mixing problem. Meanwhile, the MAPE values of the EEMD-LSTM and EEMD-GRU algorithms are lower than those of the traditional LSTM and GRU, so we use the normalized EMDN-GRU algorithm to solve this problem. Finally, as shown in Table 3 and Fig. S3j in the supplementary materials, the EEMDN-SABiGRU model outperforms LSTM, EMD-LSTM, EEMD-LSTM, GRU, EMD-GRU, EEMD-GRU, EMDN-GRU, CNN, and BP. From Table 3, based on the 20-day dataset, compared with LSTM, EMD-LSTM, EEMD-LSTM, GRU, EMD-GRU, EEMD-GRU, EMDN-GRU, CNN, and BP, MAPE is reduced by 74.14%, 91.80%, 82.95%, 90.96%, 96.47%, 87.18%, 90.80%, 90.68%, and 91.02%, respectively; MAE is reduced by 83.69%, 92.38%, 78.90%, 93.92%, 95.41%, 85.34%, 88.62%, 97.38%, and 93.12%, respectively; RMSE is decreased by 86.24%, 92.30%, 79.58%, 93.99%, 94.96%, 86.10%, 88.51%, 97.00%, and 92.39%, respectively; ME is decreased by 91.82%, 90.97%, 88.26%, 94.61%, 93.34%, 90.17%, 85.64%, 93.83%, and 86.80%, respectively. The EEMDN-SABiGRU model is nevertheless able to obtain accurate passenger hotspot prediction results with the 20-day dataset.

With the 30-day dataset, it can be seen from Fig. S4 in the supplementary materials that the greater the fluctuation of the data, the worse the fit of the model, particularly the worst at the summit. The CNN, BP, LSTM, and GRU models have a worse fitting effect with real values when the data fluctuate significantly. The EMDN-GRU and EEMDN-SABiGRU models also have excellent fitting results when the data fluctuate dramatically. Meanwhile, from Table 3, it is concluded that after the EMD and EEMD algorithms are used for LSTM and GRU, the RMSE and ME values gradually decrease. However, MAPE and MAE appear to increase because the increase in data leads to the difference between the values, resulting in unsatisfactory prediction results. In addition, the MAE, RMSE, and ME values of the normalized EMDN-GRU model decrease significantly, but the MAPE values are higher than those of the LSTM model due to the model's lower ability to capture distinct features during training and prediction. However, the bi-directional gating mechanism of EEMDN-SABiGRU with the addition of spatial attention can solve these problems. Finally, from Fig. S4j in the supplementary materials, it can be concluded that the EEMDN-SABiGRU model outperforms LSTM, EMD-LSTM, EEMD-LSTM, GRU, EMD-GRU, EEMD-GRU, EMDN-GRU, CNN, and BP. From Table 3, based on the the 30-day dataset, compared with LSTM, EMD-LSTM, EEMD-LSTM, GRU, EMD-GRU, EEMD-GRU, EMDN-GRU, CNN, and BP, MAPE is reduced by 63.16%, 88.66%, 93.07%, 83.82%, 95.89%, 92.22%, 66.67%, 83.03%, and 86.41%, respectively; MAE is decreased by 85.30%, 90.27%, 89.71%, 84.48%, 95.61%, 91.13%, 80.26%, 95.78%, and 87.96%, respectively; RMSE is decreased by 91.82%, 89.00%, 88.84%, 93.40%, 93.29%, 91.92%, 84.08%, 94.46%, and 84.80%, respectively; ME is decreased by 92.34%, 85.98%, 85.37%, 93.20%, 89.03%, 88.68%, 69.58%, 82.44%, and 61.39%, respectively. The EEMDN-SABiGRU model obtains accurate prediction results for passenger hotspots on the 30-day dataset.

4.4.4 Prediction results of different grids

To further evaluate the scalability of the EEMDN-SABiGRU model, the surrounding grids, such as the 00-, 55-, and 99-grid, are selected for prediction with the 30-day dataset, and the results are shown in Tables 4–6.

Table 4 shows the MOE values of the 00-, 01-, 10-, and 11-grid with the 30-day dataset. Table 5 illustrates the MOE values of the 55-grid peripheral

grids 44, 45, 54, and 55 with the 30-day dataset. Table 6 shows the MOE values of the 99-grid peripheral grids 88, 89, 98, and 99 with the 30-day dataset. According to Tables 4–6, the average MAPE values of the EEMDN-SABiGRU model in the Spark framework are 3.150%, 2.175%, and 2.275%, respectively, which indicates that the EEMDN-SABiGRU model is strongly reliable and scalable in predicting passenger hotspots between different grids in different areas, as illustrated in Fig. 9.

4.4.5 Time complexity analysis

In this work, we compare the execution time of each model with 1- and 10-day datasets, as described in Table 7.

Table 4 Measures of effectiveness for 00-grid with the same dataset

Dataset	MAPE (%)	MAE	RMSE	ME
00-30-day	2.800	0.396	0.670	9.091
01-30-day	2.200	0.191	0.274	2.113
10-30-day	5.000	0.447	0.571	3.077
11-30-day	2.600	0.288	0.319	0.915

Table 5 Measures of effectiveness for 55-grid with the same dataset

Dataset	MAPE (%)	MAE	RMSE	ME
44-30-day	1.500	0.981	1.325	8.490
45-30-day	3.700	0.746	0.929	4.167
54-30-day	2.500	1.010	1.211	6.014
55-30-day	1.000	0.160	0.210	1.048

Table 6 Measures of effectiveness for 99-grid with the same dataset

Dataset	MAPE (%)	MAE	RMSE	ME
88-30-day	2.200	0.881	1.055	6.988
89-30-day	2.000	0.461	0.616	3.265
98-30-day	2.800	0.572	0.698	2.873
99-30-day	2.100	0.947	1.651	6.453

From Table 7, the time complexity of the proposed EEMDN-SABiGRU model is in the same level as those of the other comparable models. Although our EEMDN-SABiGRU model does not improve the time complexity of passenger hotspot prediction, it has the same level of execution efficiency as other models.

In summary, the aforementioned prediction results demonstrate that the EEMDN-SABiGRU model shows excellent prediction performance. However, with the increased dataset, the value of ME increases, which shows that the prediction error of the model also increases. As shown in Fig. 10, the prediction performances of the LSTM and GRU models after using EMD, EEMD, and EMDN are significantly improved with the 1-day dataset. With the 10-, 20-, and 30-day datasets, after adding EMD and EEMD to the LSTM and GRU models, the RMSE and ME values decrease and the MAPE and MSE values increase, indicating that the neural network model incorporating the EMD and EEMD methods produces a decreasing trend of prediction performance when the dataset increases. The MOE values of the EEMDN-SABiGRU model fluctuate slightly with the 30-day dataset, which proves that the EEMDN-SABiGRU model generates excellent prediction stability when the prediction performance is satisfactory. In particular, the time complexity of the EEMDN-SABiGRU model is in the same level as those of the comparable models.

5 Conclusions

This paper proposed a distributed EEMDN-SABiGRU model on Spark to predict passenger

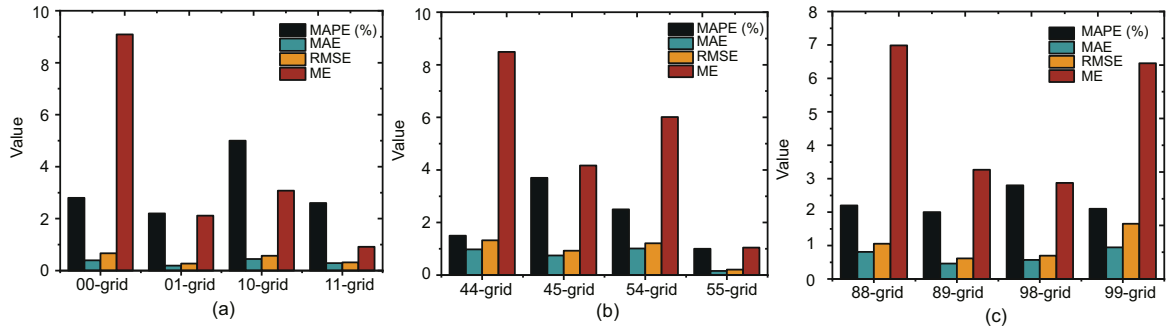

Fig. 9 Comparisons of MOE values for EEMDN-SABiGRU under different grids with the 30-day dataset: (a) 00-grid; (b) 55-grid; (c) 99-grid

Table 7 Execution time of different models

Dataset	Execution time (s)									
	LSTM	EMD-LSTM	EEMD-LSTM	GRU	EMD-GRU	EEMD-GRU	EMDN-GRU	CNN	BP	EEMDN-SABiGRU
1-day	7.18	46.49	43.04	7.97	34.11	48.97	27.83	15.45	13.58	68.97
10-day	147.57	403.29	546.62	66.79	408.25	462.67	411.75	276.53	164.32	748.58

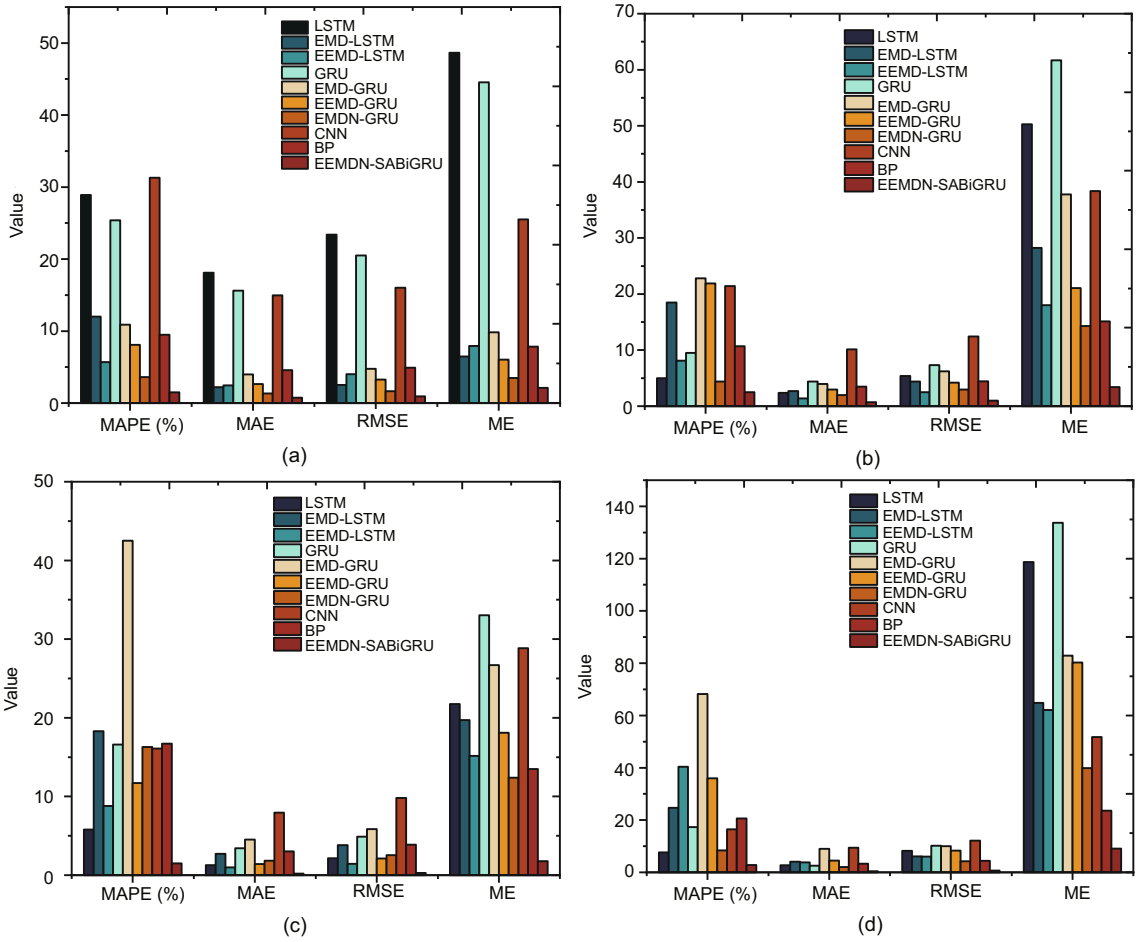

Fig. 10 Comparisons of MOE values for different models with different datasets in the 00-grid: (a) 1-day; (b) 10-day; (c) 20-day; (d) 30-day

hotspots. Specifically, the urban road network was rasterized under the Spark framework. Then, to improve the prediction accuracy of the model and to solve the non-smooth sequences and numerical differences, the EEMD algorithm and normalization method were introduced to process the rasterized road network data. Next, the fusion BiGRU model dealt with the deficiency that the GRU model cannot extract contextual information, and the spatial attention module was constructed to focus on the travel hotspot areas on the map and obtain the prediction results. Finally, the prediction results were merged by inverse normalization to obtain the final prediction results. In particular, the prediction results of the EEMDN-SABiGRU model were compared with those of the LSTM, GRU, EMD-LSTM, EMD-GRU, EEMD-LSTM, EEMD-GRU, EMDN-GRU, CNN, and BP models. The experimental results demonstrated that EEMDN-SABiGRU is significantly superior to LSTM, GRU, EMD-LSTM, EMD-GRU, EEMD-LSTM, EEMD-GRU, EMDN-GRU, CNN, and BP. From the experiment results, it

can be concluded that the EEMDN-SABiGRU model can predict the passenger hotspots more accurately, and the prediction performance was still satisfactory with the increased dataset.

In future work, we will consider the effects of weather, traffic conditions, and passenger mobility on passenger hotspots, and validate the EEMDN-SABiGRU model using GPS data from taxis in different cities.

Contributors

Dawen XIA and Jian GENG designed the research. Dawen XIA, Jian GENG, and Huaqing LI proposed the approaches and performed the experiments. Ruixi HUANG, Bingqi SHEN, and Yang HU processed the data. Dawen XIA, Jian GENG, and Huaqing LI drafted the paper. Dawen XIA, Jian GENG, Yang HU, Yantao LI, and Huaqing LI revised and finalized the paper.

Compliance with ethics guidelines

Dawen XIA, Jian GENG, Ruixi HUANG, Bingqi SHEN, Yang HU, Yantao LI, and Huaqing LI declare that they have no conflict of interest.

Data availability

The data that support the findings of this study are available from the corresponding authors upon reasonable request.

References

Ali A, Zhu YM, Zakarya M, 2021. A data aggregation based approach to exploit dynamic spatio-temporal correlations for citywide crowd flows prediction in fog computing. *Multim Tool Appl*, 80(20):31401-31433. https://doi.org/10.1007/s11042-020-10486-4

Batty M, Axhausen KW, Giannotti F, et al., 2012. Smart cities of the future. *Eur Phys J Spec Top*, 214(1):481-518. https://doi.org/10.1140/epjst/e2012-01703-3

Bi SB, Xu RZ, Liu AL, et al., 2021. Mining taxi pick-up hotspots based on grid information entropy clustering algorithm. *J Adv Transp*, 2021:5814879. https://doi.org/10.1155/2021/5814879

Cao Y, Hou XL, Chen N, 2022. Short-term forecast of OD passenger flow based on ensemble empirical mode decomposition. *Sustainability*, 14(14):8562. https://doi.org/10.3390/su14148562

Cheng X, Mao JD, Li J, et al., 2021. An EEMD-SVD-LWT algorithm for denoising a lidar signal. *Measurement*, 168:108405. https://doi.org/10.1016/j.measurement.2020.108405

Dong YH, Qian SY, Zhang K, et al., 2017. A novel passenger hotspots searching algorithm for taxis in urban area. Proc 18th IEEE/ACIS Int Conf on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, p.175-180. https://doi.org/10.1109/SNPD.2017.8022719

Engelbrecht J, Booysen MJ, van Rooyen GJ, et al., 2015. Survey of smartphone-based sensing in vehicles for intelligent transportation system applications. *IET Intell Transp Syst*, 9(10):924-935. https://doi.org/10.1049/iet-its.2014.0248

Gao HH, Liu C, Li YHZ, et al., 2020. V2VR: reliable hybrid-network-oriented V2V data transmission and routing considering RSUs and connectivity probability. *IEEE Trans Intell Transp Syst*, 22(6):3533-3546. https://doi.org/10.1109/tits.2020.2983835

Gong L, Liu X, Wu L, et al., 2016. Inferring trip purposes and uncovering travel patterns from taxi trajectory data. *Cartogr Geogr Inform Sci*, 43(2):103-114. https://doi.org/10.1080/15230406.2015.1014424

Huang ZH, Tang JY, Shan GX, et al., 2019. An efficient passenger-hunting recommendation framework with multitask deep learning. *IEEE Int Things J*, 6(5):7713-7721. https://doi.org/10.1109/JIOT.2019.2901759

Jamil MS, Akbar S, 2017. Taxi passenger hotspot prediction using automatic ARIMA model. Proc 3rd Int Conf on Science in Information Technology, p.23-28. https://doi.org/10.1109/ICSITech.2017.8257080

Jiang XS, Zhang L, Chen XQ, 2014. Short-term forecasting of high-speed rail demand: a hybrid approach combining ensemble empirical mode decomposition and gray support vector machine with real-world applications in China. *Transp Res Part C Emerg Technol*, 44:110-127. https://doi.org/10.1016/j.trc.2014.03.016

Kim T, Sharda S, Zhou XS, et al., 2020. A stepwise interpretable machine learning framework using linear regression (LR) and long short-term memory (LSTM): city-wide demand-side prediction of yellow taxi and for-hire vehicle (FHV) service. *Transp Res Part C Emerg Technol*, 120:102786. https://doi.org/10.1016/j.trc.2020.102786

Li ML, Yan M, He HW, et al., 2021. Data-driven predictive energy management and emission optimization for hybrid electric buses considering speed and passengers prediction. *J Clean Prod*, 304:127139. https://doi.org/10.1016/j.jclepro.2021.127139

Li XF, Zhang Y, Du MY, et al., 2020. The forecasting of passenger demand under hybrid ridesharing service modes: a combined model based on WT-FCBF-LSTM. *Sustain Cities Soc*, 62:102419. https://doi.org/10.1016/j.scs.2020.102419

Li XL, Pan G, Wu ZH, et al., 2012. Prediction of urban human mobility using large-scale taxi traces and its applications. *Front Comput Sci*, 6(1):111-121. https://doi.org/10.1007/s11704-011-1192-6

Liu J, Wu NQ, Qiao Y, et al., 2020. Short-term traffic flow forecasting using ensemble approach based on deep belief networks. *IEEE Trans Intell Transp Syst*, 23(1):404-417. https://doi.org/10.1109/TITS.2020.3011700

Liu XP, Zhang YQ, Zhang QC, 2022. Comparison of EEMD-ARIMA, EEMD-BP and EEMD-SVM algorithms for predicting the hourly urban water consumption. *J Hydroinf*, 24(3):535-558. https://doi.org/10.2166/hydro.2022.146

Luo HM, Cai JM, Zhang KP, et al., 2021. A multi-task deep learning model for short-term taxi demand forecasting considering spatiotemporal dependences. *J Traffic Transp Eng Engl Ed*, 8(1):83-94. https://doi.org/10.1016/j.jtte.2019.07.002

Nie ZH, Shen F, Xu DJ, et al., 2020. An EMD-SVR model for short-term prediction of ship motion using mirror symmetry and SVR algorithms to eliminate EMD boundary effect. *Ocean Eng*, 217:107927. https://doi.org/10.1016/j.oceaneng.2020.107927

Niu XX, Ma JW, Wang YK, et al., 2021. A novel decomposition-ensemble learning model based on ensemble empirical mode decomposition and recurrent neural network for landslide displacement prediction. *Appl Sci*, 11(10):4684. https://doi.org/10.3390/app11104684

Ou JJ, Sun JH, Zhu YC, et al., 2020. STP-TrellisNets: spatial-temporal parallel trellisnets for metro station passenger flow prediction. Proc 29th ACM Int Conf on Information & Knowledge Management, p.1185-1194. https://doi.org/10.1145/3340531.3411874

Qin QD, He HD, Li L, et al., 2020. A novel decomposition-ensemble based carbon price forecasting model integrated with local polynomial prediction. *Comput Econ*, 55(4):1249-1273. https://doi.org/10.1007/s10614-018-9862-1

Qu BT, Yang WX, Cui G, et al., 2019. Profitable taxi travel route recommendation based on big taxi trajectory data. *IEEE Trans Intell Transp Syst*, 21(2):653-668. https://doi.org/10.1109/TITS.2019.2897776

Rezaei H, Faaljou H, Mansourfar G, 2021. Stock price prediction using deep learning and frequency decomposition. *Exp Syst Appl*, 169:114332. https://doi.org/10.1016/j.eswa.2020.114332

Saadallah A, Moreira-Matias L, Sousa R, et al., 2020. BRIGHT—drift-aware demand predictions for taxi networks. *IEEE Trans Knowl Data Eng*, 32(2):234-245. https://doi.org/10.1109/TKDE.2018.2883616

Seng DW, Lv FS, Liang ZY, et al., 2021. Forecasting traffic flows in irregular regions with multi-graph convolutional network and gated recurrent unit. *Front Inform Technol Electron Eng*, 22(9):1179-1193. https://doi.org/10.1631/FITEE.2000243

Wang RK, Huang WJ, Hu BT, et al., 2022. Harmonic detection for active power filter based on two-step improved EEMD. *IEEE Trans Instrum Meas*, 71:9001510. https://doi.org/10.1109/TIM.2022.3146913

Xia DW, Jiang SY, Yang N, et al., 2021a. Discovering spatiotemporal characteristics of passenger travel with mobile trajectory big data. *Phys A Stat Mech Appl*, 578:126056. https://doi.org/10.1016/j.physa.2021.126056

Xia DW, Zhang MT, Yan XB, et al., 2021b. A distributed WND-LSTM model on MapReduce for short-term traffic flow prediction. *Neur Comput Appl*, 33(7):2393-2410. https://doi.org/10.1007/s00521-020-05076-2

Xia DW, Bai Y, Geng J, et al., 2022a. A distributed EMDN-GRU model on Spark for passenger waiting time forecasting. *Neur Comput Appl*, 34(21):19035-19050. https://doi.org/10.1007/s00521-022-07482-0

Xia DW, Zheng YL, Bai Y, et al., 2022b. A parallel grid-search-based SVM optimization algorithm on Spark for passenger hotspot prediction. *Multim Tool Appl*, 81(19):27523-27549. https://doi.org/10.1007/s11042-022-12077-x

Xu DW, Wang YD, Jia LM, et al., 2017. Real-time road traffic state prediction based on ARIMA and Kalman filter. *Front Inform Technol Electron Eng*, 18(2):287-302. https://doi.org/10.1631/FITEE.1500381

Yang X, Xue QC, Yang XX, et al., 2021. A novel prediction model for the inbound passenger flow of urban rail transit. *Inform Sci*, 566:347-363. https://doi.org/10.1016/j.ins.2021.02.036

Yao XW, Wang FG, Zhang Y, 2016. A prediction model of security situation based on EMD-PSO-SVM. Proc Int Conf on Electrical and Information Technologies for Rail Transportation, p.355-363. https://doi.org/10.1007/978-3-662-49370-0_37

Yu FH, Hao HBW, Li QL, 2021. An ensemble 3D convolutional neural network for spatiotemporal soil temperature forecasting. *Sustainability*, 13(16):9174. https://doi.org/10.3390/su13169174

Zhang WY, Xia DW, Chang GY, et al., 2022. APFD: an effective approach to taxi route recommendation with mobile trajectory big data. *Front Inform Technol Electron Eng*, 23(10):1494-1510. https://doi.org/10.1631/FITEE.2100530

Zhang XK, Zhang QW, Zhang G, et al., 2018. A novel hybrid data-driven model for daily land surface temperature forecasting using long short-term memory neural network based on ensemble empirical mode decomposition. *Int J Environ Res Publ Health*, 15(5):1032. https://doi.org/10.3390/ijerph15051032

Zheng LJ, Xia D, Zhao X, et al., 2018. Spatial-temporal travel pattern mining using massive taxi trajectory data. *Phys A Stat Mech Appl*, 501:24-41. https://doi.org/10.1016/j.physa.2018.02.064

Zheng Y, 2017. Urban computing: enabling urban intelligence with big data. *Front Comput Sci*, 11(1):1-3. https://doi.org/10.1007/s11704-016-6907-2

Zheng Y, Capra L, Wolfson O, et al., 2014. Urban computing: concepts, methodologies, and applications. *ACM Trans Intell Syst Technol*, 5(3):38. https://doi.org/10.1145/2629592

Zhou YR, Li J, Chen H, et al., 2020. A spatiotemporal attention mechanism-based model for multi-step citywide passenger demand prediction. *Inform Sci*, 513:372-385. https://doi.org/10.1016/j.ins.2019.10.071

Zhu L, Yu FR, Wang YG, et al., 2018. Big data analytics in intelligent transportation systems: a survey. *IEEE Trans Intell Transp Syst*, 20(1):383-398. https://doi.org/10.1109/TITS.2018.2815678

List of electronic supplementary materials

Fig. S1 Comparisons of models using the 1-day dataset
Fig. S2 Comparisons of models using the 10-day dataset
Fig. S3 Comparisons of models using the 20-day dataset
Fig. S4 Comparisons of models using the 30-day dataset