

Article

Short-Term Rental Forecast of Urban Public Bicycle Based on the HOSVD-LSTM Model in Smart City

Dazhou Li ¹, Chuan Lin ² , Wei Gao ^{1,*}, Zihui Meng ¹ and Qi Song ³

¹ College of Computer Science and Technology, Shenyang University of Chemical Technology, Shenyang 110016, China; lidazhou@syuct.edu.cn (D.L.); centaureacyanus@foxmail.com (Z.M.)

² Key Laboratory for Ubiquitous Network and Service Software of Liaoning province, Dalian University of Technology, Dalian 116024, China; chuanlin1988@gmail.com

³ School of Computer Science and Engineering, Northeastern University, Shenyang 110819, China; songqi1002@163.com

* Correspondence: gwe_mail@syuct.edu.cn

Received: 24 March 2020; Accepted: 25 May 2020; Published: 29 May 2020



Abstract: As a kind of transportation in a smart city, urban public bicycles have been adopted by major cities and bear the heavy responsibility of the “last mile” of urban public transportation. At present, the main problem of the urban public bicycle system is that it is difficult for users to rent a bike during peak h, and real-time monitoring cannot be solved adequately. Therefore, predicting the demand for bicycles in a certain period and performing redistribution in advance is of great significance for solving the lag of bicycle system scheduling with the help of IoT. Based on the HOSVD-LSTM prediction model, a prediction model of urban public bicycles based on the hybrid model is proposed by transforming the source data (multiple time series) into a high-order tensor time series. Furthermore, it uses the tensor decomposition technology (HOSVD decomposition) to extract new features (kernel tensor) from higher-order tensors. At the same time, these kernel tensors are directly used to train tensor LSTM models to obtain new kernel tensors. The inverse tensor decomposition and high-dimensional, multidimensional, and tensor dimensionality reduction were introduced. The new kernel tensor obtains the predicted value of the source sequence. Then the bicycle rental amount is predicted.

Keywords: smart city; IoTs; meteorological; blocking; long short-term memory (LSTM); high order singular value decomposition (HOSVD); public bicycle system

1. Introduction

In November 2018 at the Foreign Relations Council in New York, Samuel Palmisano, CEO of IBM, put forward the concept of a “Smart Planet”. Intelligent technology is applied in many places in life. Transportation, electricity, food, currency, retail, infrastructure, and cities are becoming more and more intelligent, which has also made the earth continue to move towards the intelligent field. “Smart Earth” has stimulated the enthusiasm of countries to create smart cities.

Smart cities use information technology and other high-end technologies as the cornerstone, supported by the Internet of Things (IoT) [1–3] and cloud computing [4–6]. Taking intelligence, transparency, and networking as an essential means, on the one hand, it can reproduce the digital form of the material city. On the other hand, it can combine with the material city and derive a mutually beneficial urban system. Smart cities have significant ecological connotations and social responsibilities, embodying the duality of the integration of virtual and reality and the coexistence of advantages and disadvantages.

The smart city is the product of the new information technology revolution and the rapid development of society, culture, and economy, and it is a profound combination of the “three modernizations”. A variety of networks such as IoTs, telecommunication networks, Wi-Fi, and the internet are used as communication support layers. It is a new idea for urban development that takes the rapid development of smart industries, the deep integration of smart technologies, and the quick and convenient use of smart services as a benefit to the people.

The urban public bicycle system in the smart city powered by the IoTs is mainly composed of bicycle stations, parking piles, public bicycles, smart rental terminals, smart rental cards (generally bound to urban bus cards), control centers, vehicle dispatch centers, dispatch vehicles, and bicycle repair and maintenance bases. In the above links, a large amount of new intelligent computing algorithms will be needed in each link. For these enormous amounts of technical needs, the method of artificial intelligence (AI) must be used to realize the value of its application promptly [7–9].

In this paper, the LSTM prediction model is improved and a city public bicycle prediction model based on the hybrid model is proposed. Furthermore, a comprehensive forecasting model is established by comprehensively considering the effects of time and weather on public bicycle rental across the entire city. In the proposed model, we combine the low-rank tensor decomposition and the tensor prediction model into a unified framework, which is not only faster but also captures the internal relationship between multiple time series. It can improve prediction results, especially for short data. The proposed model is used to predict the number of rental public bicycles in the entire city. Finally, experimental prediction and analysis are performed to obtain the prediction results of the model.

2. Related Works

Many scholars have studied the relevant theories of urban public bicycle systems. Singhvi et al. predicted pairwise bike demand for New York City’s Citi Bike system [10]. Kou et al. analyzed the distributions of trip distance and trip duration for bike-sharing trips for commuting and touristic purposes [11]. Raviv and Kolka introduced an inventory model suited for the management of bike rental stations and a numerical solution method used to solve it. Moreover, a structural result about the convexity of the model has been proved [12]. Mauro et al. addressed the Bike-Sharing Rebalancing Problem (BRP), in which a fleet of capacitated vehicles is employed to redistribute the bikes to minimize total cost [13]. Qiu and He comprehensively analyzed the domestic urban bicycle system and studied the relationship between economic development and transportation development [14].

Kloimüller and Raidl investigated a simplified problem model in which only full vehicle loads had been considered for movement among the rental stations. It appears to have only a minor impact on the achieved quality of the rebalancing in practice but eases the modeling substantially. More specifically, the rebalancing problem as a selective unit-capacity pickup and delivery problem with time budgets on a bipartite graph has been formulated, and a compact mixed-integer linear programming model, a logic-based Benders decomposition and a variant thereof, namely branch-and-check for it, has been given [15]. Caggiani et al. presented a multi-objective model based on a fuzzy inference system to be embedded in a mobile application that could assist cyclists in the selection of the smartest route to follow to reach their destination in terms of travel costs (distance or time), level of air pollution and road safety [16]. Pal and Zhang presented a Novel Mixed Integer Linear Program for solving the Static Complete Rebalancing Problem. The proposed formulation can not only handle single as well as multiple vehicles but also allows for multiple visits to a node by the same vehicle [17]. Zhao et al. studied the dispatching and management of no pile bicycle-sharing and constructed a dispatching model based on semi-open multi-vehicles with fuzzy time windows, with the number of distribution vehicles and dispatching cost as the objective function [18]. Ho and Szeto proposed a hybrid massive neighborhood search for solving the problem. Several removals and insertion operators have been proposed to diversify and intensify the search [19]. Goh et al. proposed a method for estimating the primary demand using a rank-based demand model which accounts for choice substitutions by treating each observed trip as the best available option in a latent ranking over origin-destination

(OD) pairs [20]. Yang et al. proposed a spatiotemporal bicycle mobility model based on historical bike-sharing data and devised a traffic prediction mechanism on a per-station basis with sub-hour granularity [8]. Liu et al. proposed an inter-station bike transition (ISBT) model to predict the station drop-off demand. Moreover, a mixed-integer nonlinear programming (MINLP) formulation of multiple capacitated bike routing problems to minimize total travel distance has been provided [21].

LSTM model, also known as the long- and short-time memory model, is a kind of RNN, but also, by the input layer, hidden layer, and output layer, is a new kind of deep machine learning neural network. LSTM and traditional RNN neural networks are similar. The difference is that LSTM with a memory module instead of an RNN neural network has an implicit layer node so that it has memory ability. Ai et al. employed a deep learning approach, named the convolutional long short-term memory network (Conv-LSTM), to address the spatial dependencies and temporal dependences [22]. Xu et al. developed the LSTM model to predict the bike-sharing trip production and attraction for different time intervals, including for 10-min, 15-min, 20-min, and 30-min intervals [23]. Zhang et al. considered both historical usage and real-time passengers of public transport and used the LSTM model to establish the connection among them [24]. Pan et al. proposed a real-time LSTM model for predicting bike demands in different areas of a city during a future period based on historical data from Citi Bike System Data and meteorology data [25]. Wang et al. focused on the short-term forecasting for docking station usage in the case of Suzhou, China [26]. In [26], two latest and highly efficient models, LSTM and GRU, are adopted to predict the short-term available number of bikes in docking stations with one-month historical data.

3. Method

Each station in the New York City public bicycle trip data published on the Public Bicycle System website corresponds to a unique ID number. As the station name is relatively long, this section will use the corresponding ID number to indicate the station. Figure 1 shows the stations in the geographic coordinates. Different colors represent different blocks. There are 30 blocks in total. Among them, the abscissa is the latitude of the bicycle station and the ordinate is the longitude of the bicycle station.

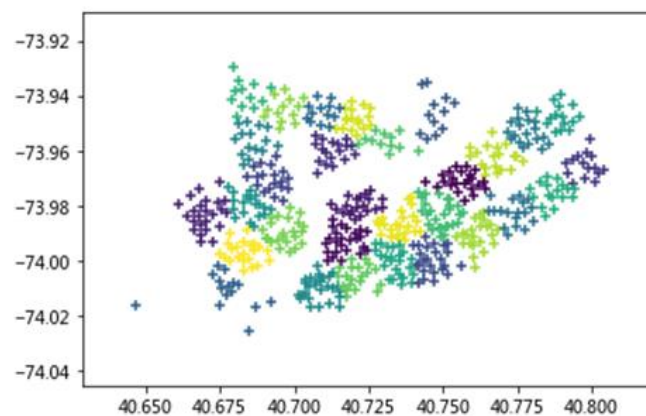


Figure 1. The stations in the geographic coordinates.

As shown in Table 1, Block 1 contains nine bicycle stations. Block 2 contains 12 bicycle stations. Block 3 contains ten bicycle stations. Block 4 contains two bicycle stations, which is also the block with the least number of stations because it is relatively remote and there are only two isolated stations around it. Block 8 contains 15 bicycle stations, which is also the block with the most stations since it is located in the center of New York and the site is more concentrated.

The traditional algorithm uses only the trip data of the predicted bike-sharing station to predict the number of rentals and returns to that bike-sharing station in the future. The bike-sharing station as the predicted object is viewed separately and in isolation from other bike-sharing stations during the prediction process. However, there is a strong correlation between one bike-sharing station and other

bike-sharing stations in the same time frame. If this strong correlation is overlooked in the prediction process, then the accuracy of the calculation results is bound to suffer. Besides, in large cities with many bike-sharing stations, the number of rentals and returns for each bike-sharing station is predicted individually, which also reduces computational efficiency and extends the computational time for the whole city.

Table 1. Block number and site ID number.

Block Number	Site ID Number
1	3430 3094 3089 3086 3084 3083 3081 3076 3074
2	3429 3417 3415 3413 467 416 353 298 274 243 241 83
3	3224 434 405 358 346 284 247 238 225 212
4	3182 3036
5	3427 3263 3260 3244 383 382 380 369 368 357 348
6	336 335 303 280 254 253 252 229 161 151 128
7	3423 3310 3306 3300
8	3254 2008 534 427 415 376 360 351 337 315 304 264 260 259 195

To overcome the above problems of the traditional algorithm, we used the whole public bicycle trip data of all bike-sharing stations in the city as the prediction object. The trip data for all bike-sharing stations throughout the city can be introduced in one prediction process. The method not only introduces the relevant effects between all bike-sharing stations at the same time into the calculation but also avoids the inefficient calculations that result from predicting each bike-sharing station individually. We summarize some notations used in our task definition in Table 2.

Table 2. The symbols used in the proposed method.

Symbol	Meaning of Symbolic Representation
M	whole public bicycle trip data of all bike-sharing stations in the city, M is a tensor, the dimension is $I_1 \times I_2 \times I_3$
I_1	number of the bike-sharing stations
I_2	the period of the public bicycle trip data
I_3	range of the public bicycle trip data
G	kernel tensor $G \in \mathbb{R}^{R_1 \times R_2 \times R_3}$
U_1, U_2, U_3	a series of factor submatrixes from the HOSVD decomposition method, $U_i \in \mathbb{R}^{I_i \times R_i}, i = 1, 2, 3$
X	a low-rank tensor $X \in \mathbb{R}^{I_1 \times I_2 \times I_3}$
$X_{(i)}$	$X_{(i)}, i = 1, 2, 3$ is the module- i expansion matrix of X
λ, μ, τ	$\lambda > 0, \mu > 0, \tau > 0$, regularization parameters
E_{R_i}	R_i -dimensional unit matrix, $i = 1, 2, 3$
$G_{(i)}$	the module- i expansion matrix of $G, i = 1, 2, 3$
Y_i	a multiplier variable
k	k -th iteration
\hat{U}_n, \hat{V}_n	results of singular value decomposition

As described in Table 1, there are 74 bike-sharing stations in the area. Each bike-sharing station generates two sets of public bicycle trip data every hour. These two records respectively record the number of public bicycles rented and the number of public bicycles returned within 1 h. If the recording time is set to 5000 h, then the total number of public bicycle rentals and repayments of all 74 bike-sharing stations within 5000 h will constitute a high-level tensor. The dimension of this tensor is $74 \times 5000 \times 2$, and we use the symbol M to represent this tensor. The dimension of M is equal to $I_1 \times I_2 \times I_3$ in the data set scenario described above, where $I_1 = 74, I_2 = 500$, and $I_3 = 2$.

We applied HOSVD's decomposition of a higher-order tensor $\mathbf{M} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ into a kernel tensor $\mathbf{G} \in \mathbb{R}^{R_1 \times R_2 \times R_3}$ and a series of factor submatrixes $\mathbf{U}_1, \mathbf{U}_2, \mathbf{U}_3$ of the modular-3 product.

$$\mathbf{M} = \mathbf{G} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \times_3 \mathbf{U}_3 \quad (1)$$

Among them, $\mathbf{U}_i \in \mathbb{R}^{I_i \times R_i}$, $i = 1, 2, 3$ is orthogonal. Generally, R_i , $i = 1, 2, 3$, the scale of the kernel tensor \mathbf{G} obtained by decomposition, is smaller than I_3 , $i = 1, 2, 3$. When we set $I_3 = 2$ to present the range of the public bicycle trip data as the rentals and returns, we can obtain $R_3 = 1$ for $R_3 < I_3$. The kernel tensor $\mathbf{G} \in \mathbb{R}^{R_1 \times R_2 \times 1}$ is a matrix. In this sense, the storage in HOSVD's decomposition is much smaller than the storage of the original tensor. Therefore, we are particularly interested in extending the HOSVD decomposition for tensor analysis.

For a given tensor \mathbf{M} , the goal is to solve a low-rank tensor $\mathbf{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ to get the following low-rank tensor problem:

$$\min_{\mathbf{X}} \frac{1}{2} \|\mathbf{X} - \mathbf{M}\|_F^2 \quad (2)$$

where $\|\cdot\|_F^2$ denotes the second order Frobenius norm.

Unlike the matrix case, the low-rank tensor estimation problem defined by Equation (2) is usually difficult to solve [27]. Inspired by the sparse representation, it is transformed and replaced by a (weighted) trace norm minimization problem:

$$\min_{\mathbf{X}} \sum_{i=1}^3 \|\mathbf{X}_{(i)}\|_{tr} + \frac{\lambda}{2} \|\mathbf{X} - \mathbf{M}\|_F^2 \quad (3)$$

where $\mathbf{X}_{(i)}$, $i = 1, 2, 3$ is the module- i expansion matrix of \mathbf{X} , $\|\mathbf{X}_{(i)}\|_{tr}$ is the trace norm of the matrix $\mathbf{X}_{(i)}$ which is the sum of singular values of $\mathbf{X}_{(i)}$, and $\lambda > 0$ is the regularization parameter.

Some scholars [28,29] have provided some matrix rank estimation methods to calculate some values of the three ranks of the tensor involved. Therefore, we consider some integers R_i , $i = 1, 2, 3$.

In [30], let tensor $\mathbf{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ and $\mathbf{G} \in \mathbb{R}^{R_1 \times R_2 \times R_3}$ satisfy $\mathbf{X} = \mathbf{G} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \times_3 \mathbf{U}_3$, and $\mathbf{U}_i^T \mathbf{U}_i = \mathbf{E}_{R_i}$, $i = 1, 2, 3$, T indicate matrix transposition. \mathbf{E}_{R_i} is a R_i -dimensional unit matrix, and then

$$\|\mathbf{X}_{(i)}\|_{tr} = \|\mathbf{G}_{(i)}\|_{tr}, \quad i = 1, 2, 3 \quad (4)$$

The module- i expansion of matrix \mathbf{X} and \mathbf{G} are $\mathbf{X}_{(i)}$ and $\mathbf{G}_{(i)}$, respectively. Based on Equation (4), Equation (3) can be transformed into

$$\begin{aligned} \min_{\mathbf{G}, \{\mathbf{U}_i\}} \sum_{i=1}^3 \|\mathbf{G}_{(i)}\|_{tr} + \frac{\lambda}{2} \|\mathbf{M} - \mathbf{G} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \times_3 \mathbf{U}_3\|_F^2 \\ \text{s.t. } \mathbf{U}_i^T \mathbf{U}_i = \mathbf{E}_{R_i}, \quad i = 1, 2, 3 \end{aligned} \quad (5)$$

The kernel tensor trace norm is Equation (5), which can alleviate the large expansion matrix involved in the convex decomposition problem and SVD computational burden. Besides, the trace norm regularization term in Equation (5) is used to improve the robustness of rank selection. The original Higher-Order Singular Value Decomposition (HOSVD) method is usually for a given rank is very sensitive [31]. Due to the interdependent matrix trace norm term, auxiliary variable $\mathbf{G}_i \in \mathbb{R}^{R_i \times \prod_{j \neq i} R_j}$ is introduced into Equation (5), and Equation (5) is re-represented as the following equivalence:

$$\begin{aligned} \min_{\mathbf{G}, \{\mathbf{U}_i\}} \sum_{i=1}^3 \|\mathbf{G}_{(i)}\|_{tr} + \frac{\lambda}{2} \|\mathbf{M} - \mathbf{G} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \times_3 \mathbf{U}_3\|_F^2 \\ \text{s.t. } \mathbf{G}_i = \mathbf{G}_{(i)}, \quad \mathbf{U}_i^T \mathbf{U}_i = \mathbf{E}_{R_i}, \quad i = 1, 2, 3 \end{aligned} \quad (6)$$

The Lagrangian function of Equation (6) is

$$L = (\mathbf{G}, \mathbf{U}_1, \mathbf{U}_2, \mathbf{U}_3, \mathbf{G}_1, \mathbf{G}_2, \mathbf{G}_3, \mathbf{Y}_1, \mathbf{Y}_2, \mathbf{Y}_3) \\ = \sum_{i=1}^3 \|\mathbf{G}_i\|_{tr} + \frac{\lambda}{2} \|\mathbf{M} - \mathbf{G} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \times_3 \mathbf{U}_3\|_F^2 + \sum_{i=1}^3 \frac{\mu}{2} \|\mathbf{G}_{(i)} - \mathbf{G}_i + \frac{\mathbf{Y}_i}{\mu}\|_F^2 - \sum_{i=1}^3 \frac{\mu}{2} \|\frac{\mathbf{Y}_i}{\mu}\|_F^2 \tag{7}$$

where $\|\cdot\|$ is the norm, $\mu > 0$ is a regularization parameter, and the multiplier variable is \mathbf{Y}_i .

Next, we use an iteration algorithm to solve Equation (7). With the calculation process in its k -th iteration, the calculation of $\mathbf{G}^{k+1}, \mathbf{U}_1^{k+1}, \mathbf{U}_2^{k+1}, \mathbf{U}_3^{k+1}$ is the object. For \mathbf{G}^{k+1} , we add neighbor operator $\frac{\tau}{2} \|\mathbf{G} - \mathbf{G}^k\|_F^2$, $\tau > 0$ as a regularization parameter, and we obtain

$$\mathbf{G}^{k+1} = \min_{\mathbf{G}} \frac{\lambda}{2} \|\mathbf{M} - \mathbf{G} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \times_3 \mathbf{U}_3\|_F^2 + \sum_{i=1}^3 \frac{\mu}{2} \|\mathbf{G}_{(i)} - \mathbf{G}_i^k + \frac{\mathbf{Y}_i^k}{\mu}\|_F^2 + \frac{\tau}{2} \|\mathbf{G} - \mathbf{G}^k\|_F^2 \tag{8}$$

For $\mathbf{U}_1^{k+1}, \mathbf{U}_2^{k+1}, \mathbf{U}_3^{k+1}, \mathbf{U}_i, i = 1, 2, 3$ can be solved by fixing other variables $\mathbf{U}_j, j \neq i, j = 1, 2, 3$. Equation (5) is transmitted to the following problems:

$$\min_{\mathbf{U}_i} \|\mathbf{M} - \mathbf{G}^{k+1} \times_1 \mathbf{U}_1^k \cdots \times_{i-1} \mathbf{U}_{i-1}^k \times_i \mathbf{U}_i \times_{i+1} \mathbf{U}_{i+1}^k \cdots \times_3 \mathbf{U}_3^k\|_F^2 \\ \text{s.t. } \mathbf{U}_i^T \mathbf{U}_i = \mathbf{E}_{R_i}, i = 1, 2, 3 \tag{9}$$

Considering that the matrix \mathbf{U}_i is column orthogonal, we can get

$$\mathbf{U}_i^{k+1} = \hat{\mathbf{U}}_n \hat{\mathbf{V}}_n \tag{10}$$

In Equation (10), $\hat{\mathbf{U}}_n$ and $\hat{\mathbf{V}}_n$ are given by the singular value decomposition of Equation (9). For $\mathbf{G}_i^{k+1}, i = 1, 2, 3$, we add neighbor operator $\frac{\tau_i}{2} \|\mathbf{G}_i - \mathbf{G}_i^k\|_F^2$ and we obtain

$$\mathbf{G}_i^{k+1} = \min_{\mathbf{G}_i} \|\mathbf{G}_{(i)}\|_{tr} + \frac{\mu}{2} \|\mathbf{G}_{(i)}^k - \mathbf{G}_{(i)} + \frac{\mathbf{Y}_i^k}{\mu}\|_F^2 + \frac{\tau_i}{2} \|\mathbf{G}_i - \mathbf{G}_i^k\|_F^2 \tag{11}$$

According to [32], \mathbf{G}_i^{k+1} can be obtained from Equation (11)

$$\mathbf{G}_i^{k+1} = D_{\frac{1}{\mu + \tau_i}} \left(\frac{\mu \mathbf{G}_{(i)}^k + \mathbf{Y}_i^k + \tau_i \mathbf{G}_i^k}{\mu + \tau_i} \right) \tag{12}$$

For the $\mathbf{Y}_i^{k+1}, i = 1, 2, 3$, multiplier variable, it can be calculated as follows

$$\mathbf{Y}_i^{k+1} = \mathbf{Y}_i^k + \mu \gamma \left(\mathbf{G}_{(i)}^{k+1} - \mathbf{G}_i^{k+1} \right) \tag{13}$$

Taking the data set scenario described in Table 1 as an example, all 74 bike-sharing stations within 5000 h will constitute a high-level tensor $\mathbf{M} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, $I_1 = 74, I_2 = 5000, I_3 = 2$, which is the input of the method. After processing through the HOSVD decomposition algorithm, the kernel tensor $\mathbf{G} \in \mathbb{R}^{R_1 \times R_2 \times R_3}$ can be obtained since $R_3 < I_3$ and $I_3 = 2, R_3 = 1$ is the only result. In Table 3, we can set R_1 or R_2 to one at the start of the HOSVD decomposition algorithm. Finally, the kernel tensor \mathbf{G} can be obtained as a vector, because two of the three-dimensionality equal one.

Table 3. Higher-Order Singular Value Decomposition (HOSVD) algorithm detailed process description.

Input: Given the Tensor M, the Scale of The Kernel Tensor R_1, R_2, R_3, Parameter $\lambda = 100$.
Output: Kernel Tensor G, Matrixes U_1, U_2, U_3
Initialization: $Y_i^0 = 0, G_i^0 = 0, U_i^0 = rand(I_i, R_i), \mu^0 = 10^{-4}, \mu_{max} = 10^{10}$
Step 1: Calculate G^{k+1} the tensor according to Equation (8).
Step 2: For $i = 1, 2, 3$, calculate the tensor U_i^{k+1}, G_i^{k+1} and Y_i^{k+1} according to Equations (11)–(13).
Step 3: Calculation parameter $\mu^{k+1} = \min(\lambda \mu^k, \mu_{max})$
Step 4: Determine the convergence condition $\max_{i=1,2,3} (G_i^{k+1} - G_i^k) < 0.001$, if it meets, stop; otherwise, go to Step 1.

When the kernel tensor G is a vector, we can forecast the new G' with the LSTM model with the historical G . The new G' can be used to multiply the matrixes U_1, U_2, U_3 , which are also the result of the HOSVD decomposition algorithm, as listed in Table 3. According to Equation (1), the prediction of high-order tensor M' equals the result of $G' \times_1 U_1 \times_2 U_2 \times_3 U_3$. Because the prediction of M' is the whole public bicycle trip data of all bike-sharing stations in the city in the next hour, we only have to use the LSTM method once.

LSTM in the sequence of data prediction has a good effect at present in the stock price prediction, and other fields have been widely used. Existing algorithms mostly use LSTM to predict the number of public bicycles to be leased and returned at the next moment of a bike-sharing station. The algorithm in this paper then uses LSTM to predict the kernel tensor G' , which is also an eigenvector of the whole public bicycle trip data of all bike-sharing stations in the city in the next moment.

We built an LSTM model with one input layer, three hidden layers, and one output layer to predict the new G' vector at the next moment, where G is the input vector of the current moment, and G' is the output value obtained after training. The model propagation process is trained according to the backpropagation through the time algorithm. The predicted value of the final moment is obtained when the error is minimal, or the maximum number of iterations is reached. The theoretical content and detail of the LSTM are available in [23].

We built the model based on the deep learning framework TensorFlow. The activation function is set to tanh, the number of cycles is 30, the number of time steps is 24, the number of training samples in each batch is 16, the number of neurons in the hidden layers is 20, and the learning rate is 0.001. In TensorFlow, the processing steps of the LSTM model are shown as follows.

- Step 1: Prepare the vector G as a sequence number, which is the output of the HOSVD decomposition algorithm listed in Table 3;
- Step 2: Determine the structure of the LSTM, such as the number of hidden layers and the neurons in each layer;
- Step 3: Choose the right activation function and the right optimization algorithm;
- Step 4: Use the training set data to train and optimize the neural network to obtain the prediction model;
- Step 5: Use the validation data set to validate the model prediction effect. If the effect is good, continue to predict, then go to Step 6, otherwise return to Step 2;
- Step 6: Use the model with good prediction results and test data to predict G' in the future.

When we have the vector G' through the LSTM model, the prediction of the whole public bicycle trip data of the all bike-sharing stations in the city in the next moment, M' , equals the result of $G' \times_1 U_1 \times_2 U_2 \times_3 U_3$, U_1, U_2, U_3 and are also the output of the HOSVD decomposition algorithm, as listed in Table 3.

4. Experiment and Result Analysis

4.1. Dataset

- (1) The first step was to sort the prediction block data. Block 8 included 15 sites with site IDs 3254, 2008, 534, 427, 415, 376, 360, 351, 337, 315, 304, 264, 260, 259, and 195. During the period from

10:00 to 11:00 on 11 September 2016, the number of bicycles rented out at the above eight stations was 4, 11, 20, 9, 17, 6, 4, 1, 0, 6, 13, 9, 9, 4 and 0, respectively. The number of bicycles rented out by Block 8 during this period is the sum of the rental amounts of the above stations, which is 113 bikes. Taking the data interval as 1 h; the bicycle rental amounts of Block 7 and Block 8 in each period were calculated as the following experimental data.

- (2) The second step was to select data for training the model. Based on the experimental data compiled in the first step, the bicycle rental data of bicycle stations from 9 August 2016, to 20 September 2016 were used as training data for this experiment. Since the format of the time record in the original data was 9/11/2016 00:00:06, to facilitate training we used spaces instead of '/', and then entered it in the table.
- (3) The third step was to select real data for comparison with predicted data. Based on the experimental data compiled in the first step, the bicycle rental data from September 21 to 30 September 2016 were used as test data for this experiment. The model input data processing is the same as in the second step.

4.2. Result Analysis

Figures 2a and 3a are the prediction results of Block 7 and Block 8, where the solid blue line represents the real value and the red dotted line represents the prediction. By comparison, it can be found that the prediction effect of Block 7 is better than that of Block 8 because Block 7 contains fewer sites, has a small number of users per day, is relatively stable, and is easier to predict.

Figures 2b and 3b are the absolute values of error between the predicted value and actual value in the station Block 7 and Block 8, respectively. The error data also contains certain components of periodicity. There are also differences in the distribution of errors in different urban areas.

Figure 4a is a comparison chart of the public bicycle rental predicted by the proposed model with the real value in the whole city. Figure 4b is the absolute value of error between the predicted value and actual value in the whole city. In the error data, periodicity still exists but the scope of review increases.

The indicators used to measure the results between predicted values and reality are Root Mean Squared Logarithmic Error (RMLSE) and error rate (ER).

- (1) As shown in Table 4, the mean, data amount N , standard deviation, and standard error of the mean are given. The standard deviations of the true and predicted values are 68.753, 67.755, 25.520, and 26.789, respectively. The true and predicted values of public bicycle rental differ greatly from their respective averages, indicating the instability of public bicycle rental.
- (2) As shown in Table 5, the correlation coefficient and significance of the number of cases between the true value and the predicted value are given. The correlation coefficient indicates the strength of the correlation between the true value and the predicted value. The closer the absolute value is to 1, the stronger the correlation. Generally, when the absolute value is high and 0.75, the two sets of data have a strong correlation. From Table 5, it can be seen that the correlation coefficient distribution of the two groups of data is 0.956 and 0.929, which is much higher than 0.75, so the true value has a high correlation with the predicted value. When the significance is less than 0.05, the two sets of data are statistically significant, and the annotation shows that the parameter is 0.000 so the true and predicted values are statistically significant.
- (3) Table 6 shows a 95% confidence interval of the difference between the real and predicted values.
- (4) Root Mean Squared Logarithmic Error (RMLSE) and Error Rate (ER).

$$RMLSE = \frac{1}{T} \sum_{t=1}^T \sqrt{\frac{1}{m} \sum_{i=1}^m (\log(X\hat{C}_i, t+1) - \log(XC_i, t+1))^2} \quad (14)$$

$$ER = \frac{1}{T} \sum_{t=1}^T \frac{\sum_{i=1}^m |X\hat{C}_i, t - XC_i, t|}{\sum_{i=1}^m XC_i, t} \quad (15)$$

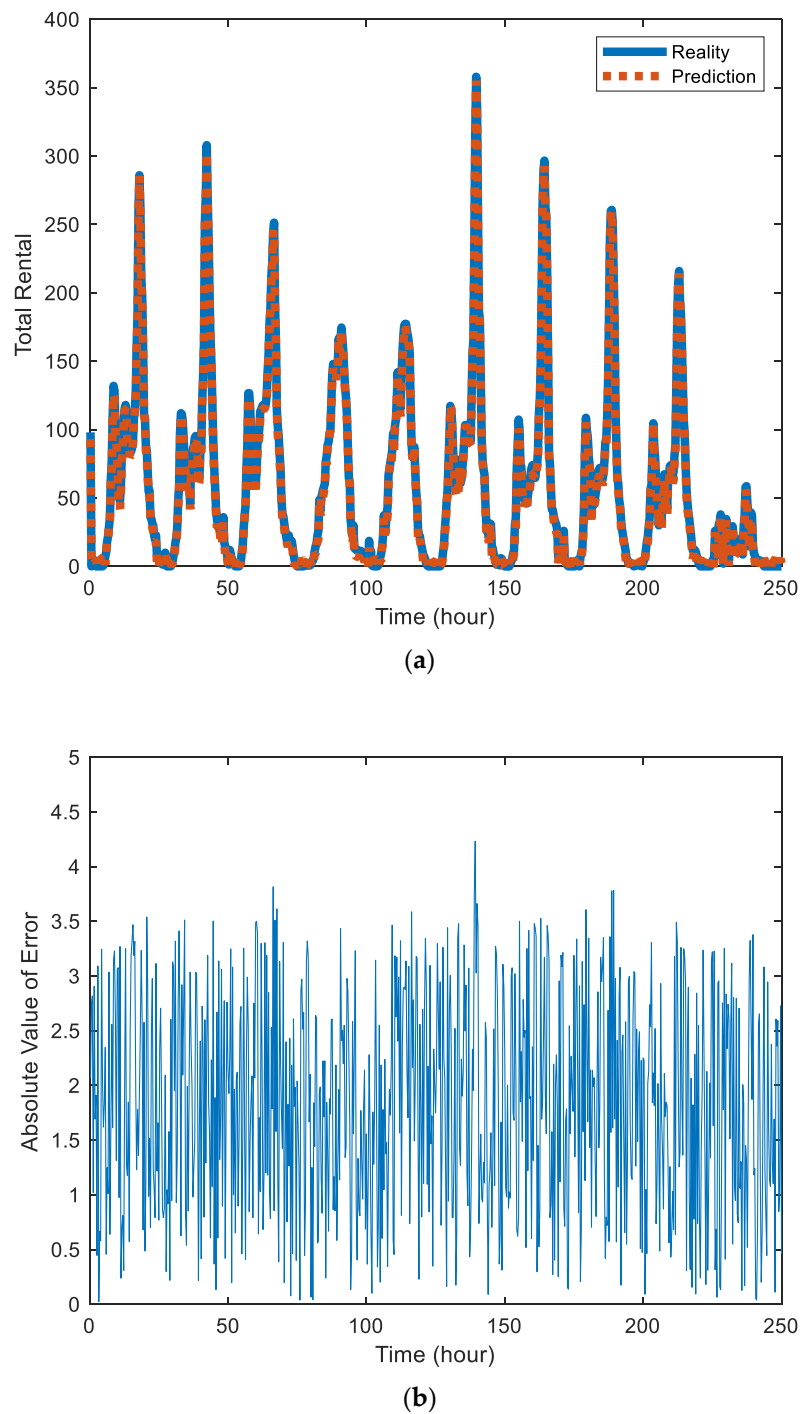
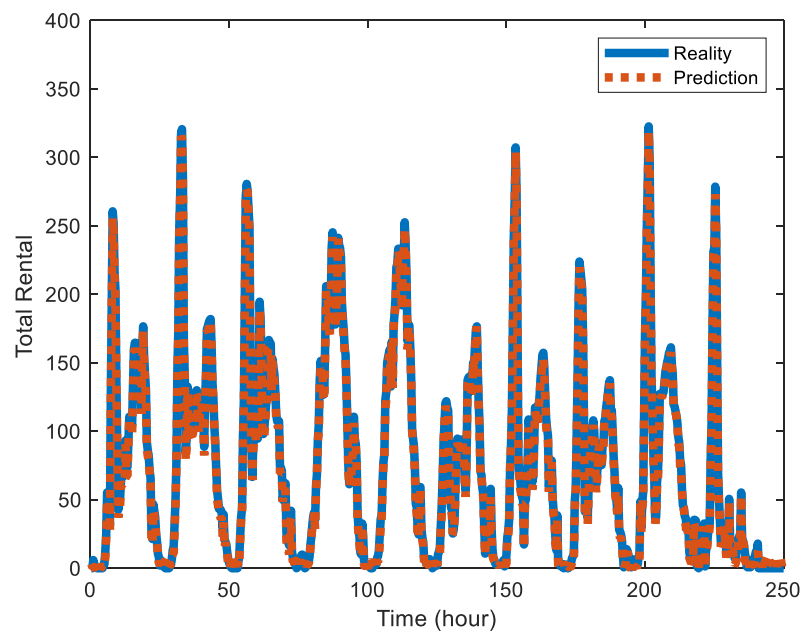
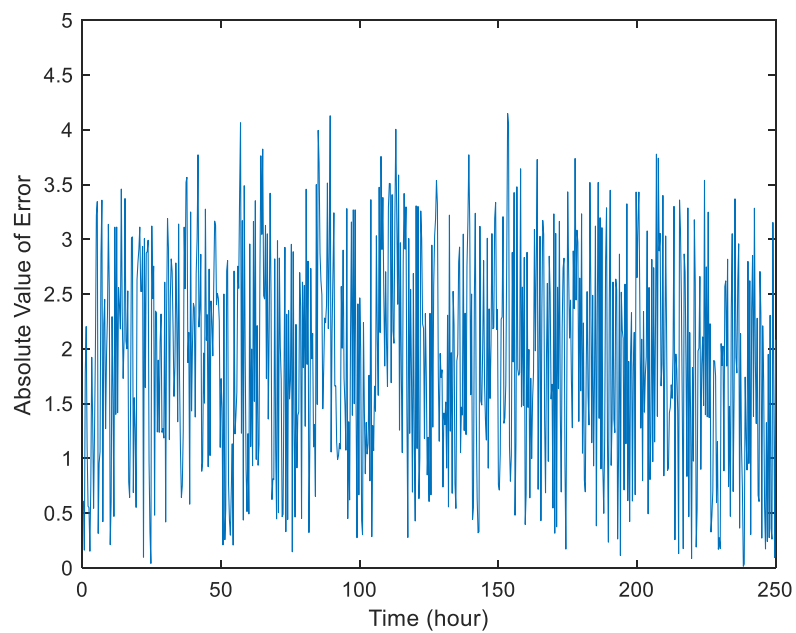


Figure 2. (a) Comparison chart of the predicted value and actual value of public bicycle rental in station Block 7. (b) Absolute value of error between the predicted value and actual value in the station Block 7.

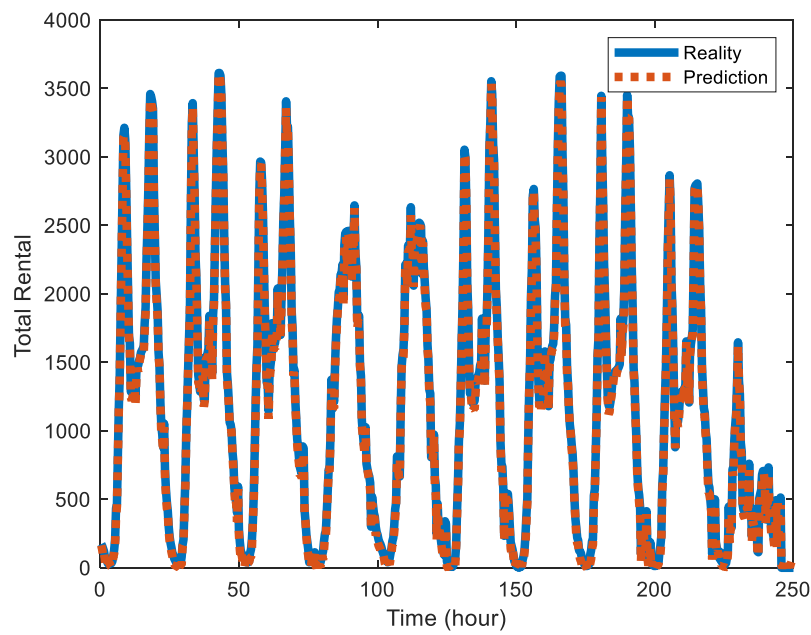


(a)

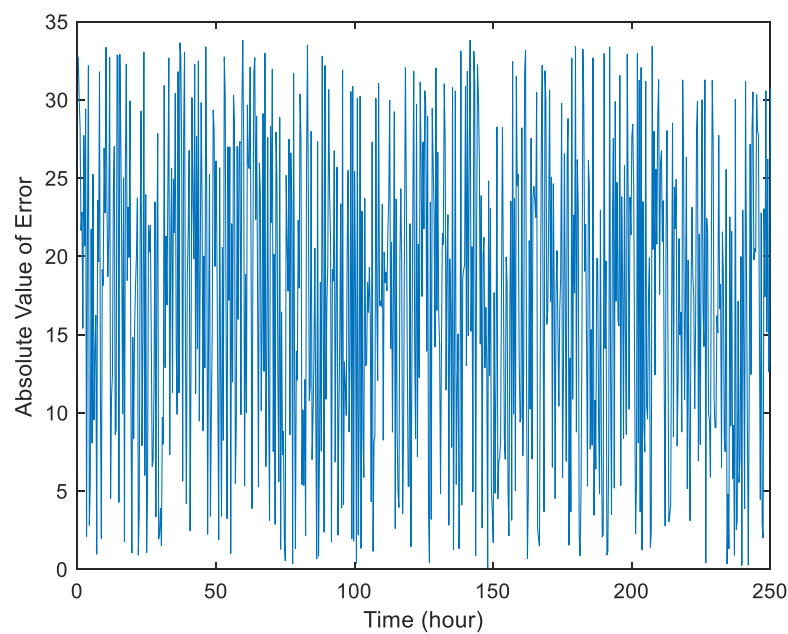


(b)

Figure 3. (a) Comparison chart of the predicted value and actual value of public bicycle rental in station Block 8. (b) Absolute value of error between the predicted value and actual value in the station Block 8.



(a)



(b)

Figure 4. (a) Comparison of predicted value and actual value of public bicycle rental in the whole city. (b) The absolute value of error between the predicted value and actual value in the whole city.

Here $X_{Ci,t}$ is the real value of the car rental in the Block C_i during the period t and $\hat{X}_{Ci,t}$ is the corresponding predicted value. Table 7 shows the root mean square error and error rate of the Block rental.

Table 4. Paired sample statics.

	Mean	N	Standard Deviation	the Standard Error of the Mean
Block 7 reality	60.72	240	68.753	4.447
Block 7 prediction	62.52	240	67.755	4.383
Block 8 reality	25.65	240	25.520	1.647
Block 8 prediction	27.55	240	26.789	1.729

Table 5. Correlation coefficients of the paired sample.

	N	Correlation Coefficient	Sig
Block 7 reality vs prediction	240	0.956	0.000
Block 8 reality vs prediction	240	0.929	0.000

Table 6. Test of the paired sample.

	Block Seven Reality vs. Prediction	Block Eight Reality vs. Prediction
95% confidence upper	−3.695	−2.980
95% confidence lower	0.097	−0.829
t	−1.869	−3.488
degrees of freedom	239	239
Sig (both sides)	0.063	0.001

Table 7. The error of rental.

	RMLSE	ER
Block 7	0.276	0.270
Block 8	0.320	0.322

Table 8 shows RMLSE and ER of the proposed HOSVD-LSTM method, the classic LSTM in [23–26], and the CNN-LSTM in [22] in the same experimental setting. The HOSVD-LSTM has a relatively 3.551 and 1.459 lower RMLSE and 2.623 and 0.851 lower ER than the classic LSTM in [23–26] and the CNN-LSTM in [22], demonstrating that HOSVD-LSTM performs better in time series prediction.

Table 8. Comparison with existing methods.

	RMLSE	ER
HOSVD-LSTM	0.395	0.401
classic LSTM	3.946	3.024
CNN-LSTM	1.854	1.252

5. Discussion

5.1. The Applied Method in a Real Scenario with Real Data

Figure 5 shows the flow of prediction in a real scenario with real data. The input of the HOSVD-LSTM model is whole public bicycle trip data of all bike-sharing stations in the city, M , which is a tensor with the dimension is $I_1 \times I_2 \times I_3$. I_1 is the number of bike-sharing stations. I_2 is the period of the public bicycle trip data. I_3 is the range of public bicycle trip data. The 3-order tensor M can be decomposed by the iterative refinement in the Lagrangian function of HOSVD method to the kernel tensor G and a series of factor submatrixes $U_i \in \mathbb{R}^{I_i \times R_i}$, $i = 1, 2, 3$. When the kernel tensor G is a vector, we can forecast the new G' in the LSTM prediction step in Figure 5 with the historical G . In the HOSVD's revivification step in Figure 5, G' , the result of LSTM prediction, and $U_i \in \mathbb{R}^{I_i \times R_i}$, $i = 1, 2, 3$,

the result of HOSVD's decomposition, can be used to obtain the prediction of 3-order tensor M' according to Equation (1) in which $M' = G' \times_1 U_1 \times_2 U_2 \times_3 U_3$. The distribution of public bicycle trip data within one hour of the all bike-sharing stations in the area can be extracted from the M' , which is the result of the HOSVD-LSTM model, as demonstrated in Figure 5.

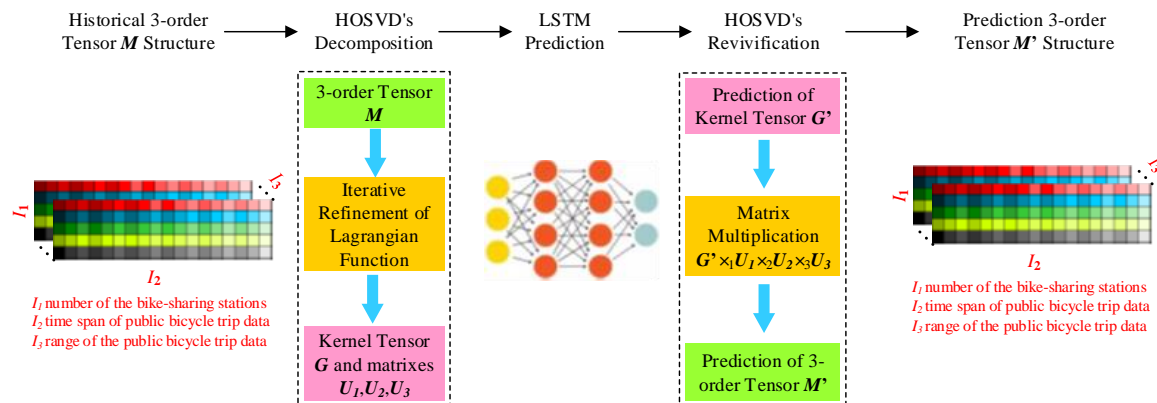


Figure 5. The processing flow of prediction in a real scenario with real data.

5.2. Comparison with Existing Methods

In [22–26], the LSTM model was used in forecasting the urban public bicycle. In this section, we can compare the proposed HOSVD-LSTM model with [22–26] in the algorithm structure and the real data experiment.

In [23–26], the LSTM models were classic models used to predict the bike-sharing trip production. The predicting procedure begins with standardizing the historical data [23]. The observations of the selected variables were then used as inputs of the developed LSTM [23]. After the predicted future production, the mean absolute percentage error was used to compare the predicted results with the original production and attraction data [23].

However, the classic LSTM model cannot capture the spatial and temporal dependences simultaneously. Because it fails to capture the spatial dependencies, the classic LSTM is not an ideal model for the spatiotemporal distribution of sharing bikes. To overcome the above problem, we applied the spatiotemporal variables as spatiotemporal 3D tensors within the LSTM model in [22]. The CNN model was used to combine the 3D tensors in [22].

The proposed method in this paper has the same features as in [22]. The proposed method and [22] both take into account the spatiotemporal correlation of the data distribution. The proposed method used the HOSVD to simultaneously predict the public bicycle trip data for all bike-sharing stations in the entire city as a whole. Thus, the proposed method in this paper can consider the spatiotemporal dependencies, which have been neglected by the classic LSTM model.

Moreover, the result of HOSVD is also an eigenvector of the whole public bicycle trip data of all bike-sharing stations in the city in the next moment. The result of HOSVD has actual spatiotemporal characteristic information because HOSVD is an effective method of tensor decomposition and dimensionality reduction. The HOSVD is a widely used algorithm in machine learning, not only for feature decomposition in downscaled algorithms, but also for recommendation systems and natural language processing, among others. The HOSVD can not only decompose high-dimensional tensor but also reconstruct the decomposed high-dimensional tensor. Compared with the CNN used in [22], the HOSVD has more interpretability and more reliable reduction. The key difference is that the CNN result is an artificial approximation of the optimal spatiotemporal feature, whereas the HOSVD result is essential and mathematically optimal. As shown in Table 8, the proposed HOSVD-LSTM has better experimental results than the classic LSTM in [23–26] and the CNN-LSTM in [22] under the same experimental scenario.

6. Conclusions

We used the publicly available dataset of New York Bike for AI modeling, and the model outputs the predicted value of bicycle rental in the next hour and compares it with the actual value. Based on the extension of the HOSVD-LSTM model, a hybrid model was established to predict the rental of urban public bicycle station stations. This article combines the advantages of both low-rank tensor decomposition technology and the tensor prediction model into a unified framework. It is not only faster but also captures the internal relationship between multiple time series, which can improve prediction results, especially for short data. Next, we verified the actual effect of HOSVD-LSTM through a large number of experiments.

Author Contributions: Conceptualization, methodology, writing—original draft preparation and funding acquisition, D.L.; software and writing—review and editing, C.L.; validation, visualization, and supervision, W.G.; formal analysis and project administration, Z.M.; resources and data curation, Q.S. All authors have read and agreed to the published version of the manuscript.

Funding: Project supported by The Science and Technology Funds from Liaoning Education Department (No. LQ2017008); Doctoral Research Startup Fund Project of Liaoning Province (No. 2016011968); China Postdoctoral Science Foundation (No. 2019M661096).

Acknowledgments: We would like to take this opportunity to acknowledge the hard work of all our editors and also the efforts of all the anonymous reviewers who have provided very valuable and helpful comments. We thank them all for their invaluable contributions, and for helping to our research.

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

References

1. Sánchez, L.; Muñoz, L.; Galache, J.A.; Sotres, P.; Santana, J.R.; Gutiérrez, V.; Ramdhany, R.; Gluhak, A.; Krčo, S.; Theodoridis, E.; et al. SmartSantander: IoT experimentation over a smart city testbed. *Comput. Netw.* **2014**, *61*, 217–238. [\[CrossRef\]](#)
2. Arasteh, H.; Hosseinneshad, V.; Loia, V.; Tommasetti, A.; Troisi, O.; Shafie-Khah, M.; Siano, P. Iot-based smart cities: A survey. In Proceedings of the 2016 IEEE 16th International Conference on Environment and Electrical Engineering (EEEIC), Florence, Italy, 7–10 June 2016; pp. 1–6.
3. Gaur, A.; Scotney, B.; Parr, G.; McClean, S. Smart City Architecture and its Applications Based on IoT. *Procedia Comput. Sci.* **2015**, *52*, 1089–1094. [\[CrossRef\]](#)
4. Clohessy, T.; Acton, T.; Morgan, L. Smart City as a Service (SCaaS): A Future Roadmap for E-Government Smart City Cloud Computing Initiatives. In Proceedings of the 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing, London, UK, 8–11 December 2014; pp. 836–841.
5. Nowicka, K. Smart City Logistics on Cloud Computing Model. *Procedia Soc. Behav. Sci.* **2014**, *151*, 266–281. [\[CrossRef\]](#)
6. Mohamed, N.; Al-Jaroodi, J.; Jawhar, I.; Lazarova-Molnar, S.; Mahmoud, S. SmartCityWare: A Service-Oriented Middleware for Cloud and Fog Enabled Smart City Services. *IEEE Access* **2017**, *5*, 17576–17588. [\[CrossRef\]](#)
7. Bao, J.; He, T.; Ruan, S.; Li, Y.; Zheng, Y. Planning Bike Lanes based on Sharing-Bikes' Trajectories. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining—KDD'17, Halifax, NS, Canada, 22–27 August 2017; pp. 1377–1386.
8. Yang, Z.; Hu, J.; Shu, Y.; Cheng, P.; Chen, J.; Moscibroda, T. Epub Mobility Modeling and Prediction in Bike-Sharing Systems. In Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services Companion—MobiSys'16 Companion, Singapore, 26–30 June 2016; pp. 165–178.
9. Chen, L.; Zhang, D.; Wang, L.; Yang, D.; Ma, X.; Li, S.; Wu, Z.; Pan, G.; Nguyen, T.-M.-T.; Jakubowicz, J. Dynamic cluster-based over-demand prediction in bike sharing systems. In Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing—UbiComp'16, New York, NY, USA, 12–16 September 2016; pp. 841–852.
10. Singhvi, D.; Singhvi, S.; Frazier, P.I.; Henderson, S.G.; O'Mahony, E.; Shmoys, D.B.; Woodard, D.B. Predicting bike usage for new york city's bike sharing system. In Proceedings of the Workshops at the twenty-ninth AAAI conference on artificial intelligence, Austin, TX, USA, 25–30 January 2016; pp. 110–114.

11. Kou, Z.; Cai, H. Understanding bike sharing travel patterns: An analysis of trip data from eight cities. *Phys. A Stat. Mech. Appl.* **2019**, *515*, 785–797. [[CrossRef](#)]
12. Raviv, T.; Kolka, O. Optimal inventory management of a bike-sharing station. *IIE Trans.* **2013**, *45*, 1077–1093. [[CrossRef](#)]
13. Dell’Amico, M.; Hadjicostantinou, E.; Iori, M.; Novellani, S. The bike sharing rebalancing problem: Mathematical formulations and benchmark instances. *Omega* **2014**, *45*, 7–19. [[CrossRef](#)]
14. Qiu, L.-Y.; He, L.-Y. Bike Sharing and the Economy, the Environment, and Health-Related Externalities. *Sustainability* **2018**, *10*, 1145. [[CrossRef](#)]
15. Kloimüller, C.; Raidl, G.R. Full-load route planning for balancing bike sharing systems by logic-based benders decomposition. *Networks* **2017**, *69*, 270–289. [[CrossRef](#)]
16. Caggiani, L.; Camporeale, R.; Ottomanelli, M. A real time multi-objective cyclists route choice model for a bike-sharing mobile application. In Proceedings of the 2017 5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS), Naples, Italy, 26–28 June 2017; pp. 645–650.
17. Pal, A.; Zhang, Y. Free-floating bike sharing: Solving real-life large-scale static rebalancing problems. *Transp. Res. Part C Emerg. Technol.* **2017**, *80*, 92–116. [[CrossRef](#)]
18. Zhao, J.; Li, Y.; Jia, H.; Jian, H.; Cai, J. Study on Allocation Scheme of Bicycle Sharing without Piles. In Proceedings of the CICTP 2019, Nanjing, China, 6–8 July 2019; pp. 1603–1614.
19. Ho, S.C.; Szeto, W. A hybrid large neighborhood search for the static multi-vehicle bike-repositioning problem. *Transp. Res. Part B Methodol.* **2017**, *95*, 340–363. [[CrossRef](#)]
20. Goh, C.Y.; Yan, C.; Jaillet, P. Estimating Primary Demand in Bike-sharing Systems. *SSRN Electron. J.* **2019**. [[CrossRef](#)]
21. Liu, J.; Sun, L.; Chen, W.; Xiong, H. Rebalancing Bike Sharing Systems. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining—KDD’16, San Francisco, CA, USA, 13–17 August 2016; pp. 1005–1014.
22. Ai, Y.; Li, Z.; Gan, M.; Zhang, Y.; Yu, D.; Chen, W.; Ju, Y. A deep learning approach on short-term spatiotemporal distribution forecasting of dockless bike-sharing system. *Neural Comput. Appl.* **2018**, *31*, 1665–1677. [[CrossRef](#)]
23. Xu, C.; Ji, J.; Liu, Z. The station-free sharing bike demand forecasting with a deep learning approach and large-scale datasets. *Transp. Res. Part C Emerg. Technol.* **2018**, *95*, 47–60. [[CrossRef](#)]
24. Zhang, C.; Zhang, L.; Liu, Y.; Yang, X. Short-term Prediction of Bike-sharing Usage Considering Public Transport: A LSTM Approach. In Proceedings of the 2018 21st International Conference on Intelligent Transportation Systems (ITSC), Maui, HI, USA, 4–7 November 2018; pp. 1564–1571.
25. Pan, Y.; Zheng, R.C.; Zhang, J.; Yao, X. Predicting bike sharing demand using recurrent neural networks. *Procedia Comput. Sci.* **2019**, *147*, 562–566. [[CrossRef](#)]
26. Wang, B.; Kim, I. Short-term prediction for bike-sharing service using machine learning. *Transp. Res. Procedia* **2018**, *34*, 171–178. [[CrossRef](#)]
27. Narita, A.; Hayashi, K.; Tomioka, R.; Kashima, H. Tensor factorization using auxiliary information. *Data Min. Knowl. Discov.* **2012**, *25*, 298–324. [[CrossRef](#)]
28. Keshavan, R.H.; Montanari, A.; Oh, S. Matrix Completion From a Few Entries. *IEEE Trans. Inf. Theory* **2010**, *56*, 2980–2998. [[CrossRef](#)]
29. Wen, Z.; Yin, W.; Zhang, Y. Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm. *Math. Program. Comput.* **2012**, *4*, 333–361. [[CrossRef](#)]
30. Liu, Y.; Shang, F.; Fan, W.; Cheng, J.; Cheng, H. Generalized higher-order orthogonal iteration for tensor decomposition and completion. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Montreal, QC, Canada, 2014; pp. 1763–1771.
31. Liu, J.; Musialski, P.; Wonka, P.; Ye, J. Tensor Completion for Estimating Missing Values in Visual Data. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *35*, 208–220. [[CrossRef](#)] [[PubMed](#)]
32. Cai, J.-F.; Candès, E.J.; Shen, Z. A Singular Value Thresholding Algorithm for Matrix Completion. *SIAM J. Optim.* **2010**, *20*, 1956–1982. [[CrossRef](#)]

