Short-Term Forecast of Bicycle Usage in Bike Sharing Systems: A Spatial-Temporal Memory Network

Xinyu Li, Yang Xu, Qi Chen, Lei Wang, Xiaohu Zhang, and Wenzhong Shi

***Abstract*— Bike-sharing systems have made notable contributions to cities by providing green and sustainable mobility service to users. Over the years, many studies have been conducted to understand or anticipate the usage of these systems, with the hope to inform their future developments. One important task is to accurately predict usage patterns of the systems. Although many deep learning algorithms have been developed in recent years to support travel demand forecast, they have mainly been used to predict traffic volume or speed on roadways. Few studies have applied them to bike-sharing systems. Moreover, these studies usually focus on one single dataset or study area. The effectiveness and robustness of the prediction algorithms are not systematically evaluated. In this study, we propose a Spatial-Temporal Memory Network (STMN) to predict short-term usage of bicycles in bike-sharing systems. The framework employs Convolutional Long Short-Term Memory models and a feature engineering technique to capture the spatial-temporal dependencies in historical data for the prediction task. Four testing sites are used to evaluate the model. These four sites include two station-based systems (Chicago and New York) and two dockless bike-sharing systems (Singapore and New Taipei City). By assessing STMN with several baseline models, we find that STMN achieves the best overall performance in all the four cities. The model also achieves superior performance in urban areas with varying levels of bicycle usage and during peak periods when demand is high. The findings suggest the reliability of STMN in predicting bicycle usage for different types of bike-sharing systems.**

***Index Terms*— Bike sharing, deep learning, travel demand, prediction, shared mobility.**

Manuscript received 2 December 2020; revised 10 May 2021; accepted 30 June 2021. Date of publication 27 July 2021; date of current version 9 August 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 41801372 and in part by the Hong Kong Polytechnic University Start-Up under Grant 1-BE0J. The Associate Editor for this article was H. G. Jung. *(Corresponding author: Yang Xu.)*

Xinyu Li is with the Department of Land Surveying and Geo-Informatics, The Hong Kong Polytechnic University, Hong Kong (e-mail: joeylee.li@connect.polyu.hk).

Yang Xu is with the Department of Land Surveying and Geo-Informatics, The Hong Kong Polytechnic University, Hong Kong, and also with The Hong Kong Polytechnic University Shenzhen Research Institute, Shenzhen 518057, China (e-mail: yang.ls.xu@polyu.edu.hk).

Qi Chen is with the School of Geography and Information Engineering, China University of Geosciences, Wuhan, Hubei 430074, China.

Lei Wang is with the University of Toronto Institute for Aerospace Studies, University of Toronto, Toronto, ON M5S 1A1, Canada.

Xiaohu Zhang is with the Department of Urban Planning and Design, Faculty of Architecture, The University of Hong Kong, Hong Kong.

Wenzhong Shi is with the Department of Land Surveying and Geo-Informatics, The Hong Kong Polytechnic University, Hong Kong, and also with the Smart Cities Research Institute, The Hong Kong Polytechnic University, Hong Kong.

Digital Object Identifier 10.1109/TITS.2021.3097240

I. Introduction

In THE past few decades, with increasing concerns over global warming and energy consumption, cities around the world have made notable efforts to promote bike-sharing systems as a green mobility strategy. Bike-sharing systems have been widely used by citizens, primarily for short-distance travels (e.g., facilitating first- and last-mile trips in cities). Meanwhile, the systems also provide alternatives to other urban issues, such as greenhouse emissions, traffic congestion, and human health deterioration [1]–[3]. There are many issues that hinder efficient operations of bike-sharing systems. For instance, a spatial mismatch between demand and supply in bike-sharing systems could lead to bicycles' unavailability in certain areas, thus affecting users' experiences. Hence, accurately predicting short-term bicycle usage could benefit operation of the systems (e.g., reblancing of bicycles) [4]–[7].

In recent years, short-term traffic prediction using deep learning frameworks (Artificial Intelligence) [8] has drawn much attention with the development of Intelligent Transportation System [9]–[11]. However, the majority of research focuses on forecasting short-term freeway traffic volumes or predicting traffic speed on urban roadways. Few studies have focused on predicting short-term travel demand for bike-sharing systems [12]–[16]. Moreover, most studies predict station-based bicycle usage, while limited effort has been paid to the dockless bike-sharing system (also known as free-floating bike-sharing system) [14], [15]. With the development of Internet of Things (IoT), dockless sharing bikes can be parked in any proper places by users, which not only improves the availability of bicycles but also increases service coverage [17]–[19]. There is no systematic research, however, that evaluates the performance of prediction algorithms on both station-based and free-floating bike-sharing systems.

Capturing spatial-temporal dependency is a critical task for predicting roadway traffic or bicycle usage. Existing deep learning models have different strategies in modeling spatial-temporal dependency in the datasets [20]–[29]. These models have their own strengths and limitations. For example, Convolution Neural Network (CNN) is widely adopted to predict traffic speed on urban roadways. Despite its superior performance in capturing the spatial dependency of roadway traffic, the framework cannot capture the temporal characteristics of traffic information [28]. Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) are also

employed to predict traffic volume and speed in transport systems [25], [29]. However, these architectures do not capture two-dimensional neighbor characteristics. As a result, some hybrid deep learning frameworks are developed to integrate convolution and RNN based methods to better capture the spatial-temporal dependency [23], [24]. There are a limited number of studies using such hybrid frameworks for predicting bicycle usage in bike-sharing systems. Moreover, the performance of these hybrid frameworks is usually assessed over one single dataset or study area [14], [16]. The effectiveness of these models has not been systematically evaluated across different types of bike-sharing systems.

In this study, we propose a Spatial-Temporal Memory Network (STMN) to predict bicycle usage in both station-based and free-floating bike-sharing systems. The framework incorporates Convolutional Long Short-Term Memory module (Conv-LSTM) [30] to capture spatial-temporal dependency in bicycle usage across urban locations. The Conv-LSTM architecture captures two-dimensional neighbor information through convolution operators and further encodes such information into a recurrent neural network. STMN adopts multiple Conv-LSTM modules to capture spatial-temporal dependencies from short-term and long-term historical records, and fuses such information through a feature engineering technique. We assess the performance of STMN with other baseline models across four bike-sharing systems. These testing sites include two free-floating systems (Singapore and New Taipei) and two station-based systems (Chicago and New York). According to the results, STMN achieves a higher level of overall accuracy than other baselines models in all four cities. The proposed framework also shows best performance in predicting bicycle usage in high-demand urban areas and during peak periods.

The remaining of this article is organized as follows. In Section II, we review and discuss related works. We then give the problem formulation in Section III, and formally introduce the STMN framework in Section IV. Section V provides a systematic evaluation of STMN over other baseline models. Finally, we discuss the implications of the study and propose future works.

II. Related Work

In this section, we discuss existing methodologies for short-term traffic forecast. The prediction methodologies can be classified into three categories: naive models, parametric models, and non-parametric models [31], [32]. Naive models are based on statistical assumptions to forecast future traffic status. For example, Historical Average (HA) [33], [34] adopts the average value of historical data as prediction results. Parametric models adopt finite parameters to describe historical data distribution and then predict future traffic demand. Autoregressive Integrated Moving Average (ARIMA) with its variants and Kalman Filtering are typical parametric models. ARIMA with its variants usually are applied to time-series data for predicting the future state through regression operations, including predicting traffic states (e.g., volume, speed, and travel time), traffic accident, or traffic noise [34]–[44]. The mechanism of Kalman Filtering is different from that of ARIMA. Kalman Filtering is an optimization algorithm to minimize the residual error between estimations and observations. Besides predicting traffic states, Kalman Filtering is also applied in traffic management and control [45]–[50]. All parametric prediction models assume that traffic data are linear and stationary. However, due to constraints of model assumptions, amounts of information is filtered by the models. For example, ARIMA adopts a difference method to create a stationary sequence. As a result, the prediction performance of parametric models cannot meet requirements for short-term predictions.

The third category, the non-parametric model, adopts data-driven approaches to capture non-linear and non-stationary process from traffic data. Thus, more training data are required to obtain sufficient information for accurate predictions. Recent research has been applied machine-learning and deep learning approaches in traffic predictions. For instance, support vector machine (SVM) [51], random forest [52], Bayesian network [53]–[56], Markov model [12]–[15], K-nearest neighbors method (KNN) [57], [58], neural network model, and hybrid deep learning approaches.

Among these non-parametric models, deep learning methods have received increasing attentions in recent years. Several existing deep learning methods originally used for other tasks are transferred to predict traffic flow/demand through the reconstruction of traffic data structures, such as CNN, RNN, and Stacked Autoencoder (SAE). In [28], the vehicle trajectory data are transferred into time-space matrix whose row represents locations of stable sensors ordered by road directions, and whose column denotes the time order. Meanwhile, adopting CNN is for extracting near and distant space-time features. RNN and its variants are employed for traffic prediction, including Gated Recurrent Unit (GRU) and Long Short Term Memory (LSTM). GRU is a variant of LSTM, and it reduces the time complexity of LSTM. Both of them are usually used to forecast traffic flow, traffic congestion, and traffic speed [25], [29], [59]. They have also been used to predict human and vehicle movements [60]–[62]. The drawbacks of above deep learning models are that the prediction approaches cannot consider both spatial and temporal dependencies in an explicit way.

Scholars realize the limitations of traditional deep learning approaches. Therefore, several hybrid approaches are proposed. In [23], Spatial-Temporal Residual Network (STRN) is proposed. In this framework, a deep residual network is adopted to capture spatial and temporal features from three historical periods, and then these features are merged by weighted element-wise addition approach to do the final prediction. Although the residual network regards a time slice as a channel, temporal correlation is hard to be captured among channels. Moreover, the weighted element-wise addition can easily fuse the different features, yet it also mixes information from different historical periods. Based on the STRN, Ren *et al.* adds an LSTM block before the residual network to capture temporal features. This network uses such a hybrid structure to capture temporal-spatial features from historical data [24]. However, the temporal correlation of traffic flow

is based on spatial information changes. In other words, the extraction of temporal features should be based on the spatial information, which is able to capture spatial-temporal dependency well.

To overcome such drawbacks, Liu *et al.* employs Convolutional LSTM (Conv-LSTM) to extract each period's spatial features by the convolution operator, and then to capture the dynamic temporal information [63]. Bi-directional LSTM technology (bi-LSTM) is also adopted to improve prediction accuracy. Bi-LSTM updates hidden layers from both past and future states simultaneously, and it is mainly applied in the field of text forecast based on the mutual relationship between contexts. However, traffic change is one-way development. In other words, the future traffic flow only changes according to the laws of historical traffic flow. As a result, although bi-LSTM can improve model prediction performance, it is hard to be explained in traffic management. Another framework named FCL-Net adopts two Conv-LSTM modules for capturing spatial-temporal dependency to predict taxi passenger demand in Hangzhou, China [64]. These modules capture high-dimensional spatial-temporal dependencies from two attributes, respectively, including travel time rate and demand intensity. The dependencies are merged after dimension reduction by CNN with external factors for final prediction. In summary, existing deep learning methods for traffic prediction have resulted into a notable improvement in prediction accuracy compared to parametric models or naive models. Meanwhile, these deep learning frameworks can be further improved to better capture the spatial-temporal dependency in the datasets to achieve better performance.

III. Problem Formulation

In this section, we introduce terminologies used in the paper and formulate the prediction problem.

Definition 1 (Grid-Based Data Structure): We divide a city into a regular $w \times h$ grid map based on a particular spatial resolution. The value of a cell (i, j) in the grid map represents the cell's bicycle demand, namely, the number of bicycle pickups. At the k^{th} time interval, the bicycle demand in the entire grid map are defined as:

$$X_k(w,h) = \begin{bmatrix} x_k(1,1) & x_k(1,2) & \cdots & x_k(1,h) \\ x_k(2,1) & x_k(2,2) & \cdots & x_k(2,h) \\ \vdots & \vdots & \ddots & \vdots \\ x_k(w,1) & x_k(w,2) & \cdots & x_k(w,h) \end{bmatrix} \quad (1)$$

Definition 2 (Measurements): We use Original-Destination (OD) trips as measurements in this case. Typically, OD trips can be derived from a collection of bike trajectories $\mathbb{P}$. At the k^{th} time interval, for a cell (i, j) that lies at the i^{th} row and the j^{th} column in the grid map, the sharing bicycle demand of this cell can be defined as:

$$x_k(i,j) = \sum_{ori_k \in \mathbb{P}} |\{g_{ori} \mid g_{ori} \in (i,j) \wedge g_{dest} \notin (i,j)\}| \quad (2)$$

where $ori_k : g_{ori} \rightarrow g_{dest}$ denotes the original-destination locations at the k^{th} time slot; g_{ori} and g_{dest} denote the spatial coordinates of starting and ending locations for a bicycle OD trip, respectively; $g_{ori} \in (i,j) \wedge g_{dest} \notin (i,j)$ denotes that the OD trip starts from the cell (i, j) but does not end in the same cell; $|\cdot|$ denotes the cardinality of a set. Particularly, the number of bicycle pick-ups during one hour in four study areas are shown in Fig. 1.

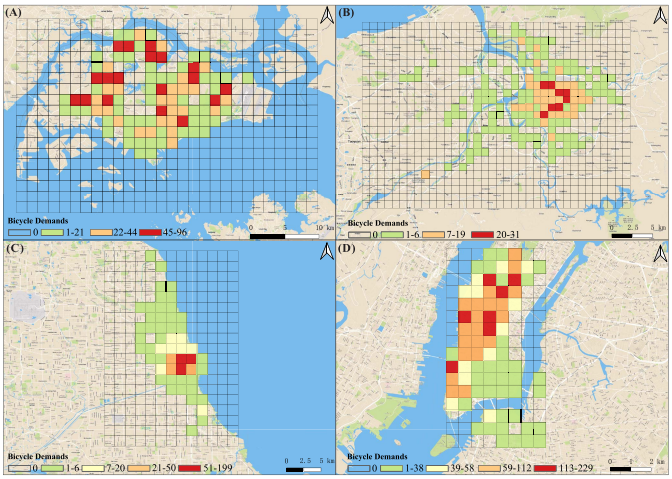

Fig. 1. The number of bicycle pick-ups in four study sites. (*A*) Singapore during 8-9 AM on June 25^{th}, 2017; (*B*) New Taipei City during 6-7 PM on August 1^{st}, 2017; (*C*) Chicago during 4-5 PM on August 26^{th}, 2019; (*D*) New York during 5-6 PM on July 9^{th}, 2014.

Problem 1: Given the historical observations $\mathcal{X}_{t-1}(w,h) = \{X_k(w,h) | k = 1, 2, \ldots, t-1\}$, predict $X_t(w,h)$.

IV. Spatial-Temporal Memory Network

A. Overview of the Proposed Model

In this study, we propose a deep learning model, Spatial-Temporal Memory Network (STMN), for the bike-sharing demand forecast. Fig. 2 illustrates the architecture of STMN. STMN consists of three individual Conv-LSTM modules and a feature fusion module. Each Conv-LSTM module is composed of convolution operators and an LSTM module. Conv-LSTM is used to extract the temporal dependency based on spatial relationships from a historical sequence. Next, all spatial-temporal features are fused by a fusion strategy. Three strategies are evaluated in this research, namely, weighted element-wise addition (STMN-WADD), simple concatenation (STMN-CAT), and weighted concatenation (STMN-WCAT). The weights of features are learnable parameters during training iterations. Finally, a two-dimensional convolution module is adopted to reduce this fusion feature's dimension such that the dimension is consistent with that of the input.

B. Convolutional LSTM

Conv-LSTM is the main component of STMN, and it can capture spatial-temporal dependencies from historical data. Conv-LSTM is composed of convolution operators and an LSTM module. The structure of Conv-LSTM is shown in Fig. 3. To aggregate characteristics of neighbor cells for

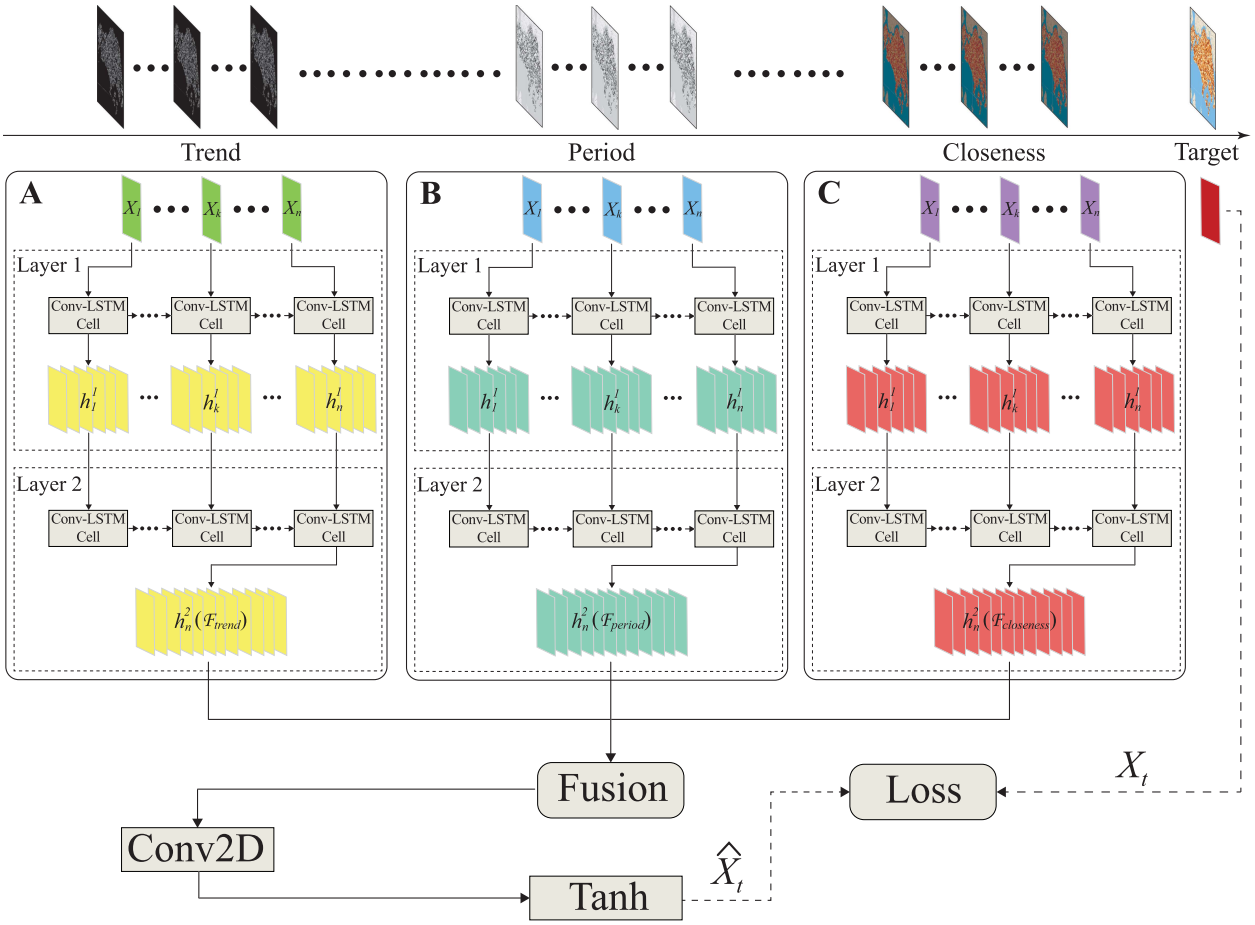

Fig. 2. The architecture of Spatial-Temporal Memory Network. Three independent Conv-LSTM modules, A, B, and C are utilized to capture spatial-temporal features from corresponding historical periods: $Trend$, $Period$ and $Closeness$. Each Conv-LSTM module contains two stacked layers for extracting deep-level spatial-temporal correlations. The fusion module is to merge three spatial-temporal features for final forecasting. Then a $Conv2D$ is to reduce the dimension, and a function $Tanh$ is used to activate a non-linear process. $\hat{X}_t$ and X_t denote the prediction result and the ground truth, respectively. The backpropagation is implemented in loss module to adjust the weights.

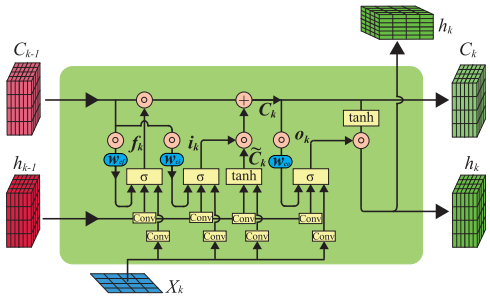

Fig. 3. The structure of Conv-LSTM cell.

the central target, Conv-LSTM adopts convolution operations for replacing vector calculations used in the traditional LSTM. Moreover, LSTM has several advantages in extracting temporal dependencies compared to RNN. For instance, the method is able to capture long-term dependency from historical sequential data based on the gate theory. It can also address gradient vanishing issue during backpropagation process.

The input of Conv-LSTM is a spatial-temporal tensor $\mathcal{X}_n = \{X_k(w,h)\}_n$, with n denoting the length of historical sequence. The matrix $X_k(w,h)$ expresses the spatial information during the time interval k. w and h represent the width and height of the study area, respectively. The computations of the Conv-LSTM can be represented through Eq. (3) to Eq. (8):

$$f_k = \sigma(W_{xf} * X_k + W_{hf} * h_{k-1} + W_{cf} \circ C_{k-1} + b_f) \quad (3)$$

$$i_k = \sigma(W_{xi} * X_k + W_{hi} * h_{k-1} + W_{ci} \circ C_{k-1} + b_i) \quad (4)$$

$$\widetilde{C}_k = tanh(W_{xc} * X_k + W_{hc} * h_{k-1} + b_c) \quad (5)$$

$$C_k = f_k \circ C_{k-1} + i_k \circ \widetilde{C}_k \quad (6)$$

$$o_k = \sigma(W_{xo} * X_k + W_{ho} * h_{k-1} + W_{co} \circ C_k + b_o) \quad (7)$$

$$h_k = o_k \circ tanh(C_k) \quad (8)$$

where X_k denotes the input observation at k^{th} time interval of the Conv-LSTM; f_k, i_k and o_k represent the outputs of forget, input and output gates; C_k represents the updated cell state; $\circ$ represents the element-wise product and $*$ denotes the convolution operation. h_k denotes the current hidden state that incorporates the hidden states of the previous layer and the outputs of the forget, input, and output gate. Moreover, the hidden state aggregates the spatial-temporal information of all past time slots and passes the information to the next time slot. Thus, the hidden state plays an essential role in

controlling long-term information flow in Conv-LSTM. The convolution operation adopts multiple fixed-size filter kernels to extract neighbor spatial information. Each convolution kernel is a square matrix to detect a grid's or a matrix's characteristics from different aspects. For example, if the shape of the input tensor for convolution module is $\mathcal{X}(C_{in}, w, h)$, the output $\mathcal{Y}(C_{out}, w, h)$ can be represented in Eq. (9). C_{in} and C_{out} represent the number of channels of input and output tensor, respectively.

$$\mathcal{Y} = b + \sum_{c=0}^{C_{in}-1} W_c \star X_c \tag{9}$$

where W_c denotes weights in the convolution kernel, $\star$ represents the valid 2D cross-correlation operator, b denotes the learnable bias.

Similar to the traditional LSTM, Conv-LSTM contains three gates: input gate (i_k), forget gate (f_k) and output gate (o_k). In addition to the actual input X_k, the previous cell state C_{k-1} and the output of the previous cell h_{k-1} (hidden state) are other inputs to the current cell. The forget gate f_k filters the information from former cell states to remember important knowledge for the current cell state. The input gate i_k as a filter on the input data can remember important knowledge for the current state. The outputs from both forget and input gates are merged as the current state C_k connecting to the next cell. The output gate o_k comprehensively considers the current state, the current input, and the previous cell state to determine what information is used as the output h_k from the current unit.

Note that zero-padding is implemented before the convolution operation to ensure that the output has the same number of rows and columns as the input. Moreover, unlike traditional CNN, the pooling layer is not used in Conv-LSTM since the resampling process will reduce the output size.

C. Temporal Dependencies

There are three historical periods for providing temporal dependency instead of all training data or a quite long subset of the data. The length of a training sequence can affect the model's perception of the temporal features, the extraction of hidden temporal features, and the model's training time complexity. An appropriate training sequence does not mean too lengthy time series. It may only require certain historical periods to provide crucial temporal information. For example, the spatial-temporal patterns of bike-sharing demand during rush hours may be similar on weekdays, and bicycle usage in the previous week might be correlated to the usage at the same time in this week.

Therefore, in this study, the temporal dependency is captured from three aspects: *closeness*, *period*, and *trend*. This way of capturing temporal features has been used in previous studies [23], [24]. Given the time slot of the prediction task, *closeness* captures the temporal dependency of bicycle usage in the past one day; *period* captures the bicycle usage in the past a couple of days; *trend* aims to capture the long-term dependency on a weekly scale. Their formal definitions are provided in Eq.(10) to Eq.(12):

$$\mathcal{X}_t^{closeness} = \{X_{t-l_c}, X_{t-(l_c-1)}, \ldots, X_{t-1}\} \tag{10}$$

$$\mathcal{X}_t^{period} = \{X_{t-24*l_p}, X_{t-24*(l_p-1)}, \ldots, X_{t-24}\} \tag{11}$$

$$\mathcal{X}_t^{trend} = \{X_{t-168*l_q}, X_{t-168*(l_q-1)}, \ldots, X_{t-168}\} \tag{12}$$

In this study, l_c is selected as 24 to capture the historical observations in the past 24 time slots. l_p is chosen as 7 to reflect the bicycle usage at the same time in the past seven days. l_q is selected as 2 to capture the bicycle usage at the same time in the past week and the week before the past. The training time complexity of the above setting will be lower than training with a lengthy period of data. The strategy might also reduce the interference of data noise. The proposed model will be compared with a simple Conv-LSTM model in this study that performs the training using a long period of historical data (i.e., in the past 336 hours).

D. Feature Fusion

As shown in Fig.2, after extracting temporal features, the outputs of three Conv-LSTM modules ($\mathcal{F}_{closeness}$, $\mathcal{F}_{period}$ and $\mathcal{F}_{trend}$) need to be fused to do the final prediction. There are three strategies of fusion evaluated in this study, including weighted element-wise addition (STMN-WADD), concatenation (STMN-CAT), and weighted concatenation (STMN-WCAT). The equations of them are shown as follows:

$$\mathcal{F}_{CAT} = \mathcal{F}_{closeness} \oplus \mathcal{F}_{period} \oplus \mathcal{F}_{trend} \tag{13}$$

$$\mathcal{F}_{WADD} = \mathcal{W}_c \circ \mathcal{F}_{closeness} + \mathcal{W}_p \circ \mathcal{F}_{period} + \mathcal{W}_t \circ \mathcal{F}_{trend} \tag{14}$$

$$\mathcal{F}_{WCAT} = \mathcal{W}_f \circ (\mathcal{F}_{closeness} \oplus \mathcal{F}_{period} \oplus \mathcal{F}_{trend}) \tag{15}$$

where $+$ and $\circ$ represent element-wise addition and Hadamard product [24], [65], respectively. $\oplus$ represents the concatenate operator. $\mathcal{W}_c, \mathcal{W}_p, \mathcal{W}_t$, and $\mathcal{W}_f$ represent the parametric tensor of the corresponding spatial-temporal feature, respectively.

The weights are learnable parameters during model training for adjusting the contribution of each Conv-LSTM module in STMN. Weighted addition and weighted concatenation have apparent distinctions when fusing features. The shape of weights in $\mathcal{F}_{WADD}$ keeps the same as the shape of spatial-temporal features in all dimensions, and the weight in $\mathcal{F}_{WCAT}$ maintains the same shape as the fused features at least two dimensions. Moreover, the concatenation expands one of dimensions to guarantee the diversity of spatial-temporal features from different historical periods. However, the addition mixes all historical information together to predict bike usage. After assessing the prediction accuracy of STMN using above mentioned feature fusion strategies, the performance of STMN-WCAT is much better than the other two variants (the results will be introduced in next section).

V. Experiments and Results

A. Settings

1) Research Areas: In order to assess the prediction accuracy of STMN and baseline models, we adopt bike-sharing

TABLE I
PARAMETERS ABOUT RESEARCH AREAS

City / Para.	Singapore	New Taipei City	Chicago	New York City
Cell Size	2km × 2km	1km × 1km	2km × 2km	0.8km × 0.8km
Temporal Granularity	1 hour	1 hour	1 hour	1 hour
Training Period	16/06-02/08 2017	21/06-29/10, 2017	01/06-30/09, 2019	01/04-31/08, 2014
Validation Period	03/08-31/08 2017	30/10-30/11, 2017	01/10-25/10, 2019	01/09-30/09, 2014

origin-detestation datasets from four cities, including Singapore, New York, New Taipei City, and Chicago. For station-based systems in New York and Chicago, the datasets are collected from the records of travel starting and ending stations; for dockless systems in Singapore and New Taipei City, the datasets are collected from raw GPS records documenting coordinates when a user starts or ends his/her travel. GPS trajectories have been pre-processed to remove outliers. In particular, we adopt the approach from an existing research [2]. Firstly, oscillation sequences caused by GPS drifts are detected and removed. We also remove short-range location switches that are possibly caused by imprecision of GPS positioning (e.g., within 150 meters in Singapore). Finally, the OD pairs are generated after removing trips with speed exceeding a threshold (e.g., 30 km/h) since they could be caused by the redistribution of bikes. Four research areas are shown in Fig. 1. The value of each cell represents the number of travels that start from this cell during one hour, excluding travels ending in the same cell. Since bicycle trips starting and ending in the same cell do not affect the overall availability of bicycles in the cell. We do not consider such trips in this study. These within-cell trips can be easily incorporated into the model input if demanded. Besides, eighty percent of bicycle demand data are used to train models, and the remaining twenty percent are used to validate the performance of models.

Table I shows spatial resolution (cell size), data aggregation interval (temporal granularity), training period, and validation period of datasets. Spatial resolution ranges from 800 meters to 2 kilometers in four cities. The New York dataset adopted in this study is the same as one of the datasets used in [23]. The dataset is provided in an aggregate form, and the spatial resolution is predefined. The grid cell size is roughly 800 meters. For the three remaining cities, to ensure that each city will have an adequate proportion of cells (i.e., over 40%) with bicycle usage, we adopt 2 km as the spatial resolution in Singapore and Chicago, and 1 km in New Taipei City.

2) Index of Performance: In order to evaluate the model performance, we select three indices which are commonly used to assess prediction accuracy, namely, Mean Absolute Percentage Error (MAPE), Root Mean Square Error (RMSE) and Mean Absolute Error (MAE). All of them are defined as following equations from Eq. 16 to Eq. 18, respectively [23], [24], [66].

$$MAPE = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{\hat{y}_i - y_i}{y_i}\right| \tag{16}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2} \tag{17}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|\hat{y}_i - y_i\right| \tag{18}$$

where $\hat{y}_i$ denotes the prediction result and y_i denotes the true value of bicycle usage.

3) Baselines: We compare all variants of STMN with the following baselines:

A) **ARIMA:** Auto-Regressive Integrated Moving Average model is a well-known time-series prediction model. According to the autocorrelation and partial autocorrelation analysis, it can select an autoregressive model (AR), a moving average (MA) model, or both of them (ARMA) to predict future states.

B) **LSTM:** Long Short-Term Memory is a typical deep learning model commonly used to forecast sequential information, especially applied in natural language processing and machine translation. The performance of LSTM is better than Recurrent Neural Network since it adopts the gate theory to keep long-term dependency in its hidden states.

C) **Simple Conv-LSTM:** The simple Conv-LSTM model is an effective prediction model to forecast spatial-temporal information. This model is able to consider spatial and temporal features together to predict feature states. It has multiple application scenarios, such as precipitation nowcasting and traffic flow prediction [30], [63]. Simple Conv-LSTM adopted in this study is a single independent framework to predict future bike usage training with the historical usage data in the past 336 hours (past two weeks).

D) **STRN:** Spatial-Temporal Residual Network is a hybrid deep learning model, and it represents good performance on crowd flows prediction. This model is satisfactory for extracting spatial-temporal features using deep residual units. Many studies have adopted the architecture similar to STRN to improve the prediction accuracy [23], [24].

4) Hyperparameter Settings: The hyperparameters of each Conv-LSTM module are the same in three historical periods. There are two stacked layers in each Conv-LSTM module, the first layer Conv1 uses 64 filters and second layer Conv2 adopts 32 filters. Thus, the dimensions of such two hidden states layers are $(64, w, h)$ and $(32, w, h)$, respectively. Here, w and h refer to the width and height of a study area, respectively. The size of each convolution kernel is 3×3 to capture short-distance correlations. Also, in order to maintain the same size as the research area, we adopt zero-padding technology that uses zero to fill the outside states, which assumes no prior knowledge for outside. The optimization algorithm adopts Root Mean Square Prop (RMSProp) that was initially used in recurrent neural network [67]. The MSELoss function is adopted in this study as usual practice, and it

TABLE II
OVERALL ACCURACY ABOUT ALL DATASETS

City	Index	ARIMA	LSTM	CONV-LSTM	STRN	STMN-WADD	STMN-CAT	STMN-WCAT
Singapore	RMSE	9.7105	6.2404	7.8266	5.8792	5.3777	4.7112	4.6776*
	MAPE	1.8185	0.9971	1.2699	0.8709	0.6079	0.628	0.5959*
	MAE	2.8405	1.8358	5.2248	2.0594	1.4817	1.4739	1.3873*
New Taipei	RMSE	2.8082	2.2408	1.6946	1.9723	1.7531	1.7148	1.6363*
	MAPE	1.2721	0.8728	0.7185	0.8146	0.7148	0.6936*	0.6954
	MAE	0.7859	0.5604	0.5343	0.5445	0.5261	0.5911	0.4646*
Chicago	RMSE	8.9316	6.6926	3.1632	2.9758	2.9071	2.7323*	2.8974
	MAPE	1.8405	1.0773	0.7758	0.6991	0.6892	0.696	0.6761*
	MAE	1.6441	1.2186	1.1304	0.7996	0.8005	0.871	0.7935*
New York	RMSE	16.2427	6.3901	7.306	6.3900	5.6627	5.5004	5.4703*
	MAPE	1.9607	0.4713	0.6208	0.5933	0.4485*	0.4914	0.4511
	MAE	7.8756	2.8793	3.7701	3.0917	2.5659	2.7129	2.5119*

utilizes mean squared error (MSE) to measure the loss between the prediction and the ground truth [68]. The definition of MSE is shown in Eq. 19.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2 \tag{19}$$

where $\hat{y}_i$ denotes the prediction value, and y_i denotes the ground truth.

For two weighted fusion strategies, $\mathcal{W}_f$ in STMN-WCAT represents the parametric tensor with shape $\mathbb{R}^{F\times w\times h}$. F denotes the number of channels of the concatenated feature. $\mathcal{W}_c, \mathcal{W}_p$ and $\mathcal{W}_t$ in STMN-WADD represent three parametric tensors having the same shape as $\mathcal{F}_{closeness}$, $\mathcal{F}_{period}$ and $\mathcal{F}_{trend}$, respectively. The shape of parametric tensors in STMN-WADD is $\mathbb{R}^{\zeta\times w\times h}$. ζ represents the number of channels of the corresponding spatial-temporal features.

For STRN, the hyperparameters are selected the same as in its paper, including four residual layers for each historical periods [23]. For LSTM and ARIMA, the hyperparameters of them have been calibrated to achieve optimal prediction results. All experiments are implemented by Pytorch framework [69]. And they are conducted on a workstation with Intel Core i7-8700 CPU and one Nvidia GeForce RTX 2070 Super Graphics Card.

B. Results

1) Overall Accuracy of Models: We use average MAPE, MAE, and RMSE as overall measurements based on all cells containing bicycle demand on entire validation periods. Table II shows overall measurements of three types of STMN variants comparing with other baselines. Given the same indicator, the best performing model is marked with * in Table II.

According to prediction results, STMN performs better than other baselines across all datasets. In general, the time-series model (ARIMA) owns the lowest prediction accuracy since it is hard to handle the non-stationary sequence, and it does not leverage any spatial information. Similarly, LSTM captures temporal patterns without considering the spatial relationship among the cells. Therefore, the performance of LSTM is still worse than Simple Conv-LSTM, STRN, and all STMN variants. Simple Conv-LSTM trains with the historical data during the past 336 hours using neighbor spatial information. However, we find that the prediction accuracy of this model is lower than the accuracy of STRN and all STMN variants. Such long-term historical data contain much unnecessary information that interferes with the model when capturing spatial-temporal dependency. Although STRN acquires good prediction results, STMN achieves better performance in all indicators. Particularly, STMN-WCAT produces an improvement of MAPE from 3% (in Chicago) to 27% (in Singapore) compared with STRN.

Regarding the impacts of different feature fusion strategies, we hereby discuss how the fusion strategy influences the prediction accuracy of STMN. We find that the performance of STMN-WCAT is the best most of time, but the other two variants have a few indicators better than STMN-WCAT. Specifically, in New Taipei City, STMN-CAT achieves a slightly better result than STMN-WCAT (MAPE: 69.36% vs. 69.54%). STMN-CAT performs better than STMN-WCAT (2.7323 vs. 2.8974) in Chicago from the perspective of MAPE. Also, in New York, STMN-WADD approaches a better result in MAPE (44.85% vs. 45.11%). However, we find that the accuracy difference between STMN-WCAT and the other two variants is pretty small. Besides, the computing costs of all three STMN variants are quite similar, such as time complexity when training models. The most important fact is that STMN-WCAT performs better than the other variants in most indicators.

We think that two main reasons lead to such difference between STMN-WCAT and its variants. Firstly, the weighted concatenation can maintain the diversity of spatial-temporal information generated from three key historical periods. Secondly, during the backpropagation step, the weights of the fused feature can be updated based on the loss information calculated by the loss function. Therefore, STMN-WADD mixes the spatial-temporal information extracted from different historical periods, and STMN-CAT is not sensitive to responding to the loss information. In sum, for predicting sharing-bike usage, STMN achieves the best performance compared with baselines, and the weighted concatenation is an effective strategy to fuse spatial-temporal features.

2) Performance of Models in Areas With Varying Levels of Bicycle Usage: We further evaluate the models' performance

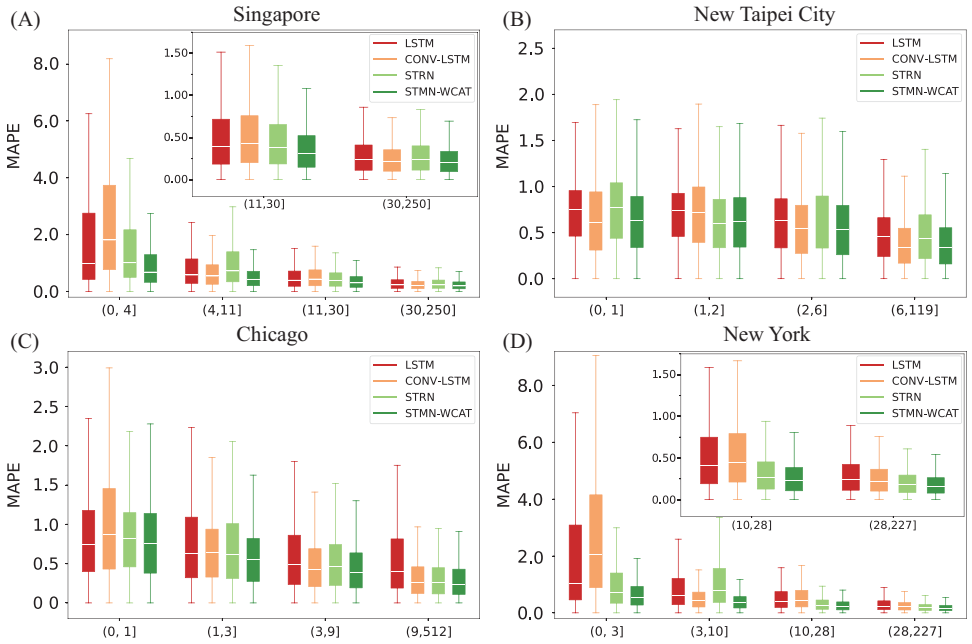

Fig. 4. Performance of four deep learning models in areas with varying levels of bicycle usage.

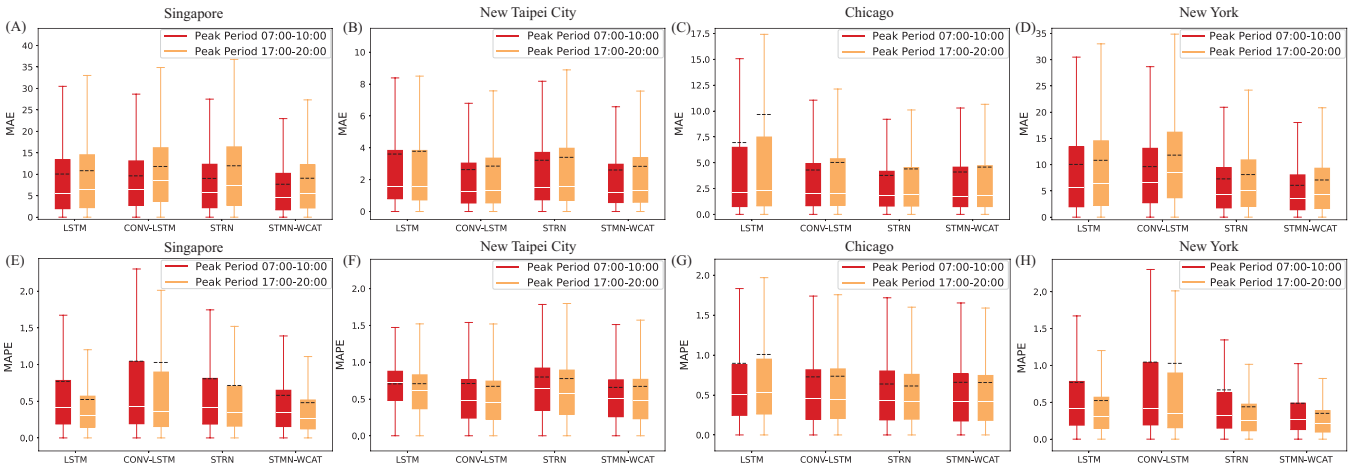

Fig. 5. Performance of four deep learning models during peak periods.

in areas with different levels of bicycle usage. We separate cells with different bicycle usage into four quantiles in each city and then assess the prediction accuracy of cells in each quantile. Based on the models' performance in overall accuracy, in the following performance analysis, we only evaluate STMN-WCAT and other deep learning baseline models. (We also discuss the relationship between the length of historical training periods and the prediction accuracy for STMN-WCAT in Appendix Section.)

We find that STMN-WCAT outperforms other deep learning models at all four quantiles of bicycle usage on most datasets. Fig. 4 shows the distributions of MAPE of four models under different levels of bicycle usage. Specifically, the performance of models in high-demand cells is important since precise prediction results in such cells help to satisfy users' needs when operating bike-sharing systems. For these high-demand cells, the prediction accuracy of STMN-WCAT is much higher than the accuracy of other models. Particularly, in Singapore (Fig. 4A) and New York (Fig. 4D), for cells in the fourth quantile, the median MAPE of STMN-WCAT is smaller than 25%, and it is the lowest compared with other models. However, in New Taipei City (Fig. 4B) and Chicago (Fig. 4C), the performance of STMN-WCAT is similar to other deep learning models for cells with large usage. This fact illustrates that STMN-WCAT has a similar prediction ability to other models on the two datasets for cells with large usage. However, STMN-WCAT performs much better in Singapore and New York. Furthermore, the advantages of STMN are also shown on the other quantiles of bicycle usage.

Note that the prediction accuracy of low-demand cells (in the first quantile) is commonly lower than the accuracy of cells in other usage levels on all datasets. These low-demand

cells are indispensable because the proportion of them is relatively large. Furthermore, the spatial distribution of such cells is random, and the emergence of low-demand cells is irregular. It is hard for prediction models to perceive the historical pattern of such low-demand cells. However, even for such cells, the performance of STMN-WCAT is still better than other models. To sum up, STMN has achieved better performance in areas with varying levels of bicycle usage.

3) Performance of Models During Peak Periods: We further assess the prediction accuracy of bike usage during peak hours. Meeting the users' needs during peak periods is an essential task for bike-sharing systems. We select cells containing bike usage during the morning peak (07:00-10:00) and the evening peak (17:00-20:00) to assess the accuracy of STMN-WCAT and other deep learning baseline algorithms. Fig. 5 shows the models' performance during peak hours in all cities (indicators including MAPE and MAE).

Generally, STMN-WCAT outperforms other models during both morning peak and evening peak. During the morning peak, in Singapore, STMN-WCAT performs better than other models. Specifically, the average MAE of STMN-WCAT is the lowest, and the median MAE is smaller than five. Also, during the evening peak, the maximum MAE of STMN-WCAT is less than twenty-seven, but the maximum MAE of STRN is more than thirty-five. In New York, STMN-WCAT outperforms other models during both peak periods, and the prediction error of STMN-WCAT is much smaller than errors of other models. Although the prediction accuracy of STMN-WCAT is quite close to the accuracy of several baselines in New Taipei City and Chicago, the performance of STMN-WCAT is stable across all datasets.

In sum, the performance of STMN is better than other baseline models across four datasets from three perspectives. The results suggest the robustness of STMN for short-term forecast of bicycle usage.

VI. Discussions and Conclusion

This paper proposes a hybrid deep learning model to predict bicycle usage across both docked and dockless bike-sharing systems. In particular, Spatial-Temporal Memory Network (STMN) is proposed to predict future bicycle usage by capturing dynamic spatial-temporal dependency. We evaluate several existing prediction models and STMN across datasets from four cities, including Singapore, Singapore, New Taipei City, Chicago, and New York. According to the results, STMN outperforms other baseline models on all datasets from aspects of overall accuracy, accuracy in areas with varying levels of usage, and accuracy during peak periods. The experimental results illustrate that STMN is reliable and robust in predicting bicycle usage for two different types of bike-sharing systems. Moreover, based on more accurate forecasts of bicycles' demand in cities, this algorithm is helpful for meeting users' needs more effectively when operating bike-sharing systems.

Currently, we only use historical observations of bicycle usage as the input to train the STMN models. According to existing research [23], [24], [27], other factors related to bicycle usage could be considered, such as weather conditions, topography, public transportation accessibility, and other built environment characteristics. For example, it is found that usage of bicycles could decrease during bad weather or in areas with steep slopes [70]. On the other hand, demand for shared bicycles around the transit stations is relatively high in some cities [2], [71]. Therefore, incorporating such factors could possibly improve the performance of the prediction model.

In the future, we plan to incorporate such factors into STMN and evaluate their impact on the model performance. For factors that vary with time (e.g., precipitation), one possible strategy is to introduce a convolution layer to map such dynamic information to a high-dimension tensor that have the same shape with the output of each Conv-LSTM, and fuse this tensor with the output using feature engineering. For long-term stable factors (e.g., topography), one possible approach is to embed such static information into a high-dimension tensor, which can then be merged with the output of the feature fusion layer (from three individual Conv-LSTM modules) to support the prediction task. Note that the availability of bicycles in grids also affects the bicycle usage. Therefore, one possible future work is to introduce the number of available bicycles or available docks in each cell as an additional constraint, and further evaluate its impact on the model performance.

TABLE III

Overall Accuracy About All Datasets Under Different Temporal Resolutions

		Reference Group l_c=24,l_p=7,l_q=2	Experimental Group I l_c=12, l_p=5, l_q=1	Experimental Group II l_c=48, l_p=14, l_q=2	Experimental Group III l_c=36, l_p=7, l_q=2
City	**Index**	**STMN-WCAT**	**STMN-WCAT**	**STMN-WCAT**	**STMN-WCAT**
Singapore	RMSE	4.6776	4.731	5.1297	4.6270*
	MAPE	0.5959*	0.6388	0.7971	0.6029
	MAE	1.3873	1.6021	1.6083	1.3441*
New Taipei	RMSE	1.6363*	1.7684	1.6864	1.6631
	MAPE	0.6954*	0.7386	0.7461	0.7176
	MAE	0.4646*	0.5035	0.4840	0.5170
Chicago	RMSE	2.8974	3.0235	2.7941*	2.8562
	MAPE	0.6761	0.7065	0.6403	0.6388*
	MAE	0.7935	0.8274	0.6641	0.6586*
New York	RMSE	5.4703	5.0015*	5.0207	5.2889
	MAPE	0.4511*	0.4725	0.4592	0.5223
	MAE	2.5119	2.4371*	2.3773	2.5918

Appendix

In this appendix, we report how the definition of the three periods (i.e., *Trend*, *Period*, *Closeness*) affects the performance of the prediction model. Note that the temporal dependency captured by the STMN-WCAT is provided by the observations from the three historical periods shown from Eq.(10) to Eq.(12), where l_c, l_p and l_q control the length of the corresponding historical period, respectively. Here, we define the model with the same parameter settings in the main body as the Reference Group. We incorporate three sets of experiments with reducing such parameters to 12, 5 and 1 as Experimental Group I, increasing l_c and l_p to 48 and 14 as Experimental Group II, and only increase l_c to 36 as Experimental Group III, respectively. The parameter settings of each group and the prediction accuracy of STMN-WCAT show in Appendix Table III, and the best performing model is marked with $*$ under given the same indicator.

In general, the Reference Group achieves the best performance. Compared to all Experimental Groups, the Reference Group is better at MAPE in Singapore, all three indicators in New Taipei City, and MAPE in New York. Experimental Group I achieves better at RMSE and MAE in New York. Experimental Group II only outperforms at RMSE in Chicago. Moreover, Experimental Group III performs better at MAE and RMSE in Singapore, MAE and MAPE in Chicago. These results indicate that the STMN-WCAT captures temporal dependency from the three historical periods selected in the Reference Group, making the models more robust.

Note that all historical training periods are shortened in Experiment Group I, the accuracy of Experiment Group I is generally lower than the Reference Group except RMSE and MAE in New York. The prediction model cannot capture sufficient temporal dependency from such short periods, and therefore, the accuracy in this group is lower. Additionally, the accuracy of Experiment Group II is also not higher than that of Reference Group, which indicates that several redundant information or noise affects the model's performance. The performance of Experimental Group III is similar to that of the Reference Group, but the prediction accuracy in several cities is still lower than the Reference Group. Some redundant information is captured from the extending *closeness*, which denotes that the past 24-hour bicycle usage data is enough for the model to extract sufficient temporal dependency. These results suggest that shortening the training periods leads to worse performance, while extending the periods does not necessarily improve the prediction accuracy of the model. Additionally, choosing appropriate training periods has an impact on the prediction accuracy when training the prediction model.

REFERENCES

[1] P. DeMaio, "Bike-sharing: History, impacts, models of provision, and future," *J. Public Transp.*, vol. 12, no. 4, pp. 41–56, Dec. 2009.

[2] Y. Xu *et al.*, "Unravel the landscape and pulses of cycling activities from a dockless bike-sharing system," *Comput., Environ. Urban Syst.*, vol. 75, pp. 184–203, May 2019.

[3] S. Jäppinen, T. Toivonen, and M. Salonen, "Modelling the potential effect of shared bicycles on public transport travel times in greater helsinki: An open data approach," *Appl. Geography*, vol. 43, pp. 13–24, Sep. 2013.

[4] T. Qin, T. Liu, H. Wu, W. Tong, and S. Zhao, "RESGCN: RESidual graph convolutional network based free dock prediction in bike sharing system," in *Proc. 21st IEEE Int. Conf. Mobile Data Manage. (MDM)*, Jun. 2020, pp. 210–217.

[5] X. Zhang, H. Yang, R. Zheng, Z. Jin, and B. Zhou, "A dynamic shared bikes rebalancing method based on demand prediction," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 238–244.

[6] L. Zhang, J. Zhang, Z.-Y. Duan, and D. Bryde, "Sustainable bike-sharing systems: Characteristics and commonalities across cases in urban China," *J. Cleaner Prod.*, vol. 97, pp. 124–133, Jun. 2015.

[7] B. Beroud, R. Clavel, and S. Le Vine, "Perspectives on the growing market for public bicycles focus on France and the United Kingdom," in *Proc. Eur. Transp. Conf.*, Glasgow, Scotland, Oct. 2010, pp. 1–15.

[8] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.

[9] H. Chang, Y. Lee, B. Yoon, and S. Baek, "Dynamic near-term traffic flow prediction: System-oriented approach based on past experiences," *IET Intell. Transp. Syst.*, vol. 6, no. 3, pp. 292–305, 2012.

[10] B. L. Smith and M. J. Demetsky, "Short-term traffic flow prediction models-a comparison of neural network and nonparametric regression approaches," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, vol. 2, Oct. 1994, pp. 1706–1709.

[11] H. Zheng, F. Lin, X. Feng, and Y. Chen, "A hybrid deep learning model with attention-based Conv-LSTM networks for short-term traffic flow prediction," *IEEE Trans. Intell. Transp. Syst.*, early access, Jun. 9, 2020, doi: 10.1109/TITS.2020.2997352.

[12] Y. Li, Y. Zheng, H. Zhang, and L. Chen, "Traffic prediction in a bike-sharing system," in *Proc. 23rd SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.*, Nov. 2015, pp. 1–10.

[13] Z. Yang, J. Hu, Y. Shu, P. Cheng, J. Chen, and T. Moscibroda, "Mobility modeling and prediction in bike-sharing systems," in *Proc. 14th Annu. Int. Conf. Mobile Syst., Appl., Services*, 2016, pp. 165–178.

[14] D. Singhvi *et al.*, "Predicting bike usage for new york city's bike sharing system," in *Proc. AAAI Workshop Comput. Sustainability*, 2015, pp. 110–114.

[15] Y. Li and Y. Zheng, "Citywide bike usage prediction in a bike-sharing system," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 6, pp. 1079–1091, Jun. 2020.

[16] Y. Ai *et al.*, "A deep learning approach on short-term spatiotemporal distribution forecasting of dockless bike-sharing system," *Neural Comput. Appl.*, vol. 31, no. 5, pp. 1665–1677, May 2019.

[17] C. Xu, J. Ji, and P. Liu, "The station-free sharing bike demand forecasting with a deep learning approach and large-scale datasets," *Transp. Res. C, Emerg. Technol.*, vol. 95, pp. 47–60, Oct. 2018.

[18] J. Lazarus, J. C. Pourquier, F. Feng, H. Hammel, and S. Shaheen, "Micromobility evolution and expansion: Understanding how docked and dockless bikesharing models complement and compete—A case study of San Francisco," *J. Transp. Geography*, vol. 84, Apr. 2020, Art. no. 102620.

[19] S. Shaheen and A. Cohen, "Shared micromoblity policy toolkit: Docked and dockless bike and scooter sharing," Transp. Sustainability Res. Center, UC Berkeley, Berkeley, CA, USA, Tech. Rep., 2019, doi: 10.7922/G2TH8JW7.

[20] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: A deep learning approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 865–873, Apr. 2015.

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

[22] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," 2017, *arXiv:1707.01926*. [Online]. Available: http://arxiv.org/abs/1707.01926

[23] J. Zhang, Y. Zheng, and D. Qi, "Deep spatio-temporal residual networks for citywide crowd flows prediction," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 1655–1661.

[24] Y. Ren, H. Chen, Y. Han, T. Cheng, Y. Zhang, and G. Chen, "A hybrid integrated deep learning model for the prediction of citywide spatio-temporal flow volumes," *Int. J. Geographical Inf. Sci.*, vol. 34, no. 4, pp. 802–823, Apr. 2020.

[25] R. Fu, Z. Zhang, and L. Li, "Using LSTM and GRU neural network methods for traffic flow prediction," in *Proc. 31st Youth Academic Annu. Conf. Chin. Assoc. Autom. (YAC)*, Nov. 2016, pp. 324–328.

[26] Y. Ren, T. Cheng, and Y. Zhang, "Deep spatio-temporal residual neural networks for road-network-based data modeling," *Int. J. Geographical Inf. Sci.*, vol. 33, no. 9, pp. 1894–1912, Sep. 2019.

[27] J. Jiang, F. Lin, J. Fan, H. Lv, and J. Wu, "A destination prediction network based on spatiotemporal data for bike-sharing," *Complexity*, vol. 2019, Jan. 2019, Art. no. 7643905.

[28] X. Ma, Z. Dai, Z. He, J. Ma, Y. Wang, and Y. Wang, "Learning traffic as images: A deep convolutional neural network for large-scale transportation network speed prediction," *Sensors*, vol. 17, no. 4, p. 818, 2017.

[29] Z. Zhao *et al.*, "LSTM network: A deep learning approach for short-term traffic forecast," *IET Intell. Transp. Syst.*, vol. 11, no. 2, pp. 68–75, 2017.

[30] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 802–810.

[31] A. M. Nagy and V. Simon, "Survey on traffic prediction in smart cities," *Pervas. Mobile Comput.*, vol. 50, pp. 148–163, Oct. 2018.

[32] Z. Wang, X. Su, and Z. Ding, "Long-term traffic prediction based on LSTM encoder-decoder architecture," *IEEE Trans. Intell. Transp. Syst.*, early access, Jun. 3, 2020, doi: 10.1109/TITS.2020.2995546.

[33] B. L. Smith, B. M. Williams, and R. K. Oswald, "Comparison of parametric and nonparametric models for traffic flow forecasting," *Transp. Res. C, Emerg. Technol.*, vol. 10, no. 4, pp. 303–321, Aug. 2002.

[34] Y. Kamarianakis and P. Prastacos, "Forecasting traffic flow conditions in an urban network: Comparison of multivariate and univariate approaches," *Transp. Res. Rec.*, vol. 1857, pp. 74–84, Jan. 2003.
[35] A. M. Khoei, A. Bhaskar, and E. Chung, "Travel time prediction on signalised urban arterials by applying SARIMA modelling on Bluetooth data," in *Proc. 36th Australas. Transp. Res. (ATRF)*, 2013, pp. 1–18.
[36] K. Kumar and V. K. Jain, "Autoregressive integrated moving averages (ARIMA) modelling of a traffic noise time series," *Appl. Acoust.*, vol. 58, no. 3, pp. 283–294, Nov. 1999.
[37] D. W. Xu, Y. D. Wang, L. M. Jia, Y. Qin, and H. H. Dong, "Real-time road traffic state prediction based on ARIMA and Kalman filter," *Frontiers Inf. Technol. Electron. Eng.*, vol. 18, no. 2, pp. 287–302, 2017.
[38] R. Avuglah, K. Adu-Poku, and E. Harris, "Application of ARIMA models to road traffic accident cases in ghana," *Int. J. Statist. Appl.*, vol. 4, no. 5, pp. 233–239, 2014.
[39] N. L. Nihan and K. O. Holmesland, "Use of the Box and Jenkins time series technique in traffic forecasting," *Transportation*, vol. 9, no. 2, pp. 125–143, 1980.
[40] D. Billings and J.-S. Yang, "Application of the ARIMA models to urban roadway travel time prediction—A case study," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, vol. 3, Oct. 2006, pp. 2529–2534.
[41] S. Lee and D. B. Fambro, "Application of subset autoregressive integrated moving average model for short-term freeway traffic volume forecasting," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1678, no. 1, pp. 179–188, Jan. 1999.
[42] B. M. Williams and L. A. Hoel, "Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: Theoretical basis and empirical results," *J. Transp. Eng.*, vol. 129, no. 6, pp. 664–672, Nov. 2003.
[43] M. Van Der Voort, M. Dougherty, and S. Watson, "Combining Kohonen maps with ARIMA time series models to forecast traffic flow," *Transp. Res. C, Emerg. Technol.*, vol. 4, no. 5, pp. 307–318, 1996.
[44] W. Min and L. Wynter, "Real-time road traffic prediction with spatio-temporal correlations," *Transp. Res. C Emerg. Technol.*, vol. 19, no. 4, pp. 606–616, 2011.
[45] J. Guo, W. Huang, and B. M. Williams, "Adaptive Kalman filter approach for stochastic short-term traffic flow rate prediction and uncertainty quantification," *Transp. Res. C, Emerg. Technol.*, vol. 43, pp. 50–64, Jun. 2014.
[46] A. Anand, G. Ramadurai, and L. Vanajakshi, "Data fusion-based traffic density estimation and prediction," *J. Intell. Transp. Syst.*, vol. 18, no. 4, pp. 367–378, 2014.
[47] F. Yang, Z. Yin, H. Liu, and B. Ran, "Online recursive algorithm for short-term traffic prediction," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1879, no. 1, pp. 1–8, Jan. 2004.
[48] C. Antoniou, M. Ben-Akiva, and H. N. Koutsopoulos, "Nonlinear Kalman filtering algorithms for on-line calibration of dynamic traffic assignment models," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 4, pp. 661–670, Dec. 2007.
[49] M. W. Szeto and D. C. Gazis, "Application of Kalman filtering to the surveillance and control of traffic systems," *Transp. Sci.*, vol. 6, no. 4, pp. 419–439, Nov. 1972.
[50] H. van Lint and T. Djukic, "Applications of Kalman filtering in traffic management and control," in *New Directions in Informatics, Optimization, Logistics, and Production*. Catonsville, MD, USA: INFORMS, 2012, pp. 59–91.
[51] Y. Zhang and Y. Liu, "Traffic forecasting using least squares support vector machines," *Transportmetrica*, vol. 5, no. 3, pp. 193–213, 2009.
[52] Y. Hou, P. Edara, and Y. Chang, "Road network state estimation using random forest ensemble learning," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2017, pp. 1–6.
[53] S. Sun, C. Zhang, and G. Yu, "A Bayesian network approach to traffic flow forecasting," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 1, pp. 124–132, Mar. 2006.
[54] A. Pascale and M. Nicoli, "Adaptive Bayesian network for traffic flow prediction," in *Proc. IEEE Stat. Signal Process. Workshop (SSP)*, Jun. 2011, pp. 177–180.
[55] S. Sun, C. Zhang, and Y. Zhang, "Traffic flow forecasting using a spatio-temporal Bayesian network predictor," in *Proc. Int. Conf. Artif. Neural Netw.* Heidelberg, Germany: Springer, 2005, pp. 273–278.
[56] B. Ghosh, B. Basu, and M. O'Mahony, "Bayesian time-series model for short-term traffic flow forecasting," *J. Transp. Eng.*, vol. 133, no. 3, pp. 180–189, 2007.
[57] D. Xia, H. Li, B. Wang, Y. Li, and Z. Zhang, "A map reduce-based nearest neighbor approach for big-data-driven traffic flow prediction," *IEEE Access*, vol. 4, pp. 2920–2934, 2016.
[58] B. Sun, W. Cheng, P. Goswami, and G. Bai, "Flow-aware WPT K-nearest neighbours regression for short-term traffic prediction," in *Proc. IEEE Symp. Comput. Commun. (ISCC)*, Jul. 2017, pp. 48–53.
[59] D.-H. Shin, K. Chung, and R. C. Park, "Prediction of traffic congestion based on LSTM through correction of missing temporal and spatial data," *IEEE Access*, vol. 8, pp. 150784–150796, 2020.
[60] B. Zhao and X. Zhang, "A parallel-res GRU architecture and its application to road network traffic flow forecasting," in *Proc. Int. Conf. Big Data Technol.*, 2018, pp. 79–83.
[61] F. Altche and A. de La Fortelle, "An LSTM network for highway trajectory prediction," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2017, pp. 353–359.
[62] D. Kong and F. Wu, "HST-LSTM: A hierarchical spatial-temporal long-short term memory network for location prediction," in *Proc. IJCAI*, 2018, pp. 2341–2347, vol. 18, no. 7.
[63] Y. Liu, H. Zheng, X. Feng, and Z. Chen, "Short-term traffic flow prediction with Conv-LSTM," in *Proc. 9th Int. Conf. Wireless Commun. Signal Process. (WCSP)*, Oct. 2017, pp. 1–6.
[64] J. Ke, H. Zheng, H. Yang, and X. M. Chen, "Short-term forecasting of passenger demand under on-demand ride services: A spatio-temporal deep learning approach," *J. Transp. Res. C, Emerg. Technol.*, vol. 85, pp. 591–608, Dec. 2017.
[65] R. A. Horn, "The Hadamard product," in *Matrices: Theory and Applications*, vol. 40. Providence, RI, USA: AMS, 1990, pp. 87–169.
[66] Y. Zhang, T. Cheng, Y. Ren, and K. Xie, "A novel residual graph convolution deep learning model for short-term network-based traffic forecasting," *Int. J. Geographical Inf. Sci.*, vol. 34, no. 5, pp. 969–995, 2019.
[67] A. Graves, "Generating sequences with recurrent neural networks," 2013, *arXiv:1308.0850*. [Online]. Available: http://arxiv.org/abs/1308.0850
[68] S. Wang, W. Liu, J. Wu, L. Cao, Q. Meng, and P. J. Kennedy, "Training deep neural networks on imbalanced data sets," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, 2016, pp. 4368–4374.
[69] A. Paszke *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d Alché-Buc, E. Fox, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, 2019, pp. 8024–8035. [Online]. Available: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf
[70] P. Midgley, "Bicycle-sharing schemes: Enhancing sustainable mobility in urban areas," *United Nations, Dept. Econ. Social Affairs*, vol. 8, pp. 1–12, May 2011.
[71] Z. Hong, A. Mittal, and H. S. Mahmassani, "Effect of bicycle-sharing on public transport accessibility: Application to Chicago divvy bicycle-sharing system," Transp. Res. Board 95th Annu. Meeting, Washington, DC, USA, Tech. Rep. 16-6930, 2016.

Xinyu Li received the B.E. degree from the School of Geographic and Environmental Science, Tianjin Normal University, Tianjin, China, in 2015, and the M.S. degree from the Institute of Space and Earth Information Science (ISEIS), The Chinese University of Hong Kong, Hong Kong, in 2017. He is currently pursuing the Ph.D. degree with the Department of Land Surveying and Geo-Informatics, The Hong Kong Polytechnic University, Hong Kong. His research interests include spatial–temporal data mining, deep learning, and geospatial artificial intelligence.

Yang Xu received the B.S. degree in remote sensing and photogrammetry and the M.S. degree in geographic information science from Wuhan University in 2009 and 2011, respectively, and the Ph.D. degree in geography from The University of Tennessee, Knoxville, in 2015. He is currently an Assistant Professor with the Department of Land Surveying and Geo-Informatics, The Hong Kong Polytechnic University. His research interests include GIScience, human mobility, and urban informatics.

Qi Chen received the B.S., M.S., and Ph.D. degrees in photogrammetry and remote sensing from Wuhan University in 2009, 2011, and 2015, respectively. He is currently an Associate Professor with the School of Geography and Information Engineering, China University of Geosciences, Wuhan. His research interests include space/aerial photogrammetry and pattern recognition from remote sensing data.

Lei Wang received the B.S. and M.S. degrees in photogrammetry and remote sensing from Wuhan University in 2009 and 2011, respectively, and the Ph.D. degree in system design from the University of Waterloo, Waterloo, Canada, in 2016. His research interests include computer vision, machine learning, and large-scale mapping.

Xiaohu Zhang is currently an Assistant Professor with the Department of Urban Planning and Design, Faculty of Architecture, The University of Hong Kong. Prior to this, he worked with Singapore-MIT Alliance for Research and Technology, the MIT Senseable City Laboratory, and Sun Yat-Sen University. His scholarship bridges the information gap in sustainable urban and transportation policy-making with stochastic simulation and big data analytics. Broadly interested in urban data science, his recent work explores the sustainability of new shared mobility services, such as scooter sharing, carsharing, and ridesharing. His research uses multi-source datasets to advance understanding of pressing urban and transportation issues, e.g., urban expansion, emerging mobility services, and the interactions between land use and transportation.

Wenzhong Shi is currently an Otto Poon C. F. Professor of urban informatics, the Chair Professor of GISci and remote sensing, and the Director of the Smart Cities Research Institute, The Hong Kong Polytechnic University. His current research interests include urban informatics and smart cities, GISci and remote sensing, intelligent analytics and quality control for spatial data, artificial-intelligence-based object extraction and change detection from satellite imagery, and mobile mapping and 3-D modeling based on LiDAR and remote sensing imagery. He has published over 250 academic papers that are indexed by SCI and 15 books.