Highlights from the 2020 ITS World Congress

TRR
JOURNAL OF THE TRANSPORTATION RESEARCH BOARD

Transportation Research Record
2021, Vol. 2675(11) 34–43

DOI: 10.1177/03611981211003896
journals.sagepub.com/home/trr
SAGE

Spatiotemporal Demand Prediction Model for E-Scooter Sharing Services with Latent Feature and Deep Learning

Seung Woo Ham[1], Jung-Hoon Cho[1], Sangwoo Park[2], and Dong-Kyu Kim[1,3]

Abstract

The electric scooter (e-scooter) sharing service has attracted significant attention because of its extensive usage and eco-friendliness. Since e-scooters are mostly accessed by foot, the presence of e-scooters within walking distance has a crucial effect on the service quality. Therefore, to maintain appropriate service quality, relocation strategies are often used to properly distribute e-scooters within service areas. There are extensive literatures on demand forecasting for an efficient relocation. However, the study of the relocation of small-scale spatial units within walking distance level is still inadequate because of the sparsity of demand data. This research aims to establish an effective methodology for predicting the demand for e-scooters in high spatial resolution. A new grid-based spatial setting was created with the usage data. The model in the methodology predicts not only the identified demand but also the unmet demand to increase practicality. A convolutional autoencoder is used to obtain the latent feature that can reduce the problem of representing sparse data. An encoder–recurrent neural network–decoder (ERD) framework with a convolutional autoencoder resulted in a huge improvement in predicting spatiotemporal events. This new ERD framework shows enhanced prediction performance, reducing the mean squared error loss to 0.00036 from 0.00679 compared with the baseline long short-term memory model. This methodological strategy has its significance in that it can solve any prediction issue with spatiotemporal data, even those with sparse data problems.

Recently, the number of electric scooter (e-scooter) sharing services has skyrocketed in various cities on a global scale. According to Shaheen and Cohen (*1*), about 25,000 standing e-scooters are being shared in ten cities in the United States. The percentage of people using the e-scooter sharing service is higher than any other mobility services. One of the largest e-scooter sharing companies provided 6 million rides during its first 58 weeks of operation, whereas Lyft, a well-known ride-hailing service, provided only 1 million rides during its first 61 weeks of operation (*2*). Several cities have welcomed e-scooter services because of their potential to resolve traffic congestion and environmental issues. One study showed that one-third of e-scooter users would have used a motor vehicle if the e-scooter sharing service had not been available (*1*). These figures represent the positive effects of e-scooters on solving environmental problems. However, the e-scooter sharing service still faces some additional issues, including safety, operational feasibility, and vandalism. The focus of this paper is on operational feasibility. As most e-scooter sharing services use dockless systems, there can be spatiotemporal imbalances between the supply of e-scooters and their demand. The first key in the rebalancing strategy is anticipating the demand for the usage of e-scooters.

Research on predicting demand using machine learning for personal mobility has been conducted extensively for bike-sharing services (*3*–*8*). Zhou et al. (*6*) used a Markov chain model to predict the daily demand for the Zhongshan bike-sharing service. They stated that the Markov chain model outperformed the neural network, the support vector machine, and the regression model. Lin et al. (*7*) proposed an advanced model based on the graph convolutional network (GCN) to predict the level of hourly demand in specific stations of the New York City bike-sharing service. Kim et al. (*8*) also presented a multilayer GCN to extract the spatiotemporal property

[1]Department of Civil and Environmental Engineering, Seoul National University, Seoul, Republic of Korea
[2]VP Product Deer Corporation Co., Ltd., Seongdong-gu, Seoul, Republic of Korea
[3]Institute of Construction and Environmental Engineering, Seoul National University, Seoul, Republic of Korea

Corresponding Author:
Dong-Kyu Kim, dongkyukim@snu.ac.kr

of bike demand patterns. Although the GCN framework shows high predictive performance, there is also the drawback of immense computational cost.

As long as the personal mobility prediction problem handles spatiotemporal data, the recurrent neural network (RNN) model can provide an advantage in predictive ability (*9*). Several attempts have been made using the RNN model to forecast the demand for bike-sharing services (*10–13*). Ai et al. (*13*) used a convolutional long short-term memory (LSTM) network to anticipate the pattern of dockless bike-sharing service in Chengdu. They partitioned the city into 7×7 grids, measuring 4 km on each side, far longer than ordinary walking distance. As typical e-scooter users access the service on foot, the size of the grid had to be shortened to walking distance to design a model from the user's point of view, maintaining high practicality.

However, the difficulty of the spatiotemporal prediction problem is inversely proportional to the size of the grid. The smaller the size of the grid is, the more cells are in the same area, resulting in an increased number of zero-event cells. When the zero-event cells outnumber the event cells, a zero-inflation problem, known as sparse data representation problem, occurs. The data with consecutive zeros in most cases obstruct the model's learning, inducing close-to-zero results in prediction. This problem can be mitigated by transforming the data into latent feature space representation (*14, 15*). In this research, an autoencoder based on the convolutional neural network was used to efficiently extract the latent feature of the data and alleviate the problem caused by sparse data representation (*16*). The autoencoder has the advantage of being able to extract latent features from data in a non-parametrical way without assuming the distribution in feature space.

This study established a demand prediction model of an e-scooter sharing service optimized for the user. Based on the actual operating data, the prediction model used the latent feature obtained via the autoencoder. This methodological approach is termed as an encoder–RNN–decoder (ERD) structure, which solves the issue of sparse data representation while maintaining the high resolution of the grid.

The key contributions of this article are as follows:

- A user-friendly, grid-based spatial framework was established by analyzing the real usage-based data. Whereas previous studies focused on making macroscale predictions, this study developed microscale predictions for practicality.
- In addition to including the demand that was actually satisfied/used (hereafter referred to as "identified demand"), this study also covered the "unmet demand" data. The mobile app execution record was employed to identify the intention of the user.
- The problem of sparse data representation was solved using a convolutional autoencoder and RNN simultaneously, known as an ERD structure as mentioned earlier. The latent feature achieved by the autoencoder dramatically enhanced the result of the prediction.
- The circulating nature of the e-scooter service—one user's destination converted to another user's origin—was considered in the input. Although combining the demand and destination data increased the heterogeneity of the input, it eventually enhanced the prediction ability.

The LSTM with naïve input was set as a baseline model (*17*). This study invented a new model with a latent feature using LSTM and the gated recurrent unit (GRU) for the RNN cell (*18*). The remainder of the paper provides a detailed methodology, the result of this model, and the conclusions of the study.

Methodology

Data and Site Description

Data from Deer Corporation, a South Korean e-scooter company, were used in this study. The data were collected from September 8, 2019, to October 10, 2019, and the total number of usage and total number of users were 32,136 and 6,398, respectively. This identified demand and the unmet demand, which will be described in detail in the next section, were used as input and output of the model. The service area includes the Konkuk University Station area, a densely populated college district with both a shopping center and a residential complex. Six metro stations are located within walking distance of the service area, one of which is slightly outside the appointed service area but still adjacent. As there is little difference in altitude within the service area, the usage of e-scooters is barely affected by the slope in any location.

Determination of Spatial Units: The Size of the Grid

To improve the practicality of this study, it was crucial to determine the proper spatial unit for the model. In this research, the spatial unit equals the size of the grid. As the size of the grid increases, the difficulty of the prediction problem decreases as the total number of grid cells declines. However, this lengthened size of grid fades the practicality of the model from the user's perspective. Thus, the average distance that the users travel on the e-scooters was identified to determine the size of the grid.

Table 1. Algorithm 1: Decision of the Size of the Grid by Median Access Time

Algorithm 1:	The decision of the size of the grid by median access time
Input:	1. Start time and start location of every e-scooter usage
	2. Time and location log of every (mobile) app execution
Output:	The size of the grid
1:	**for all** e-scooter usage **do**
2:	collect app execution record of the user, which has done before the start time of e-scooter usage
3:	**for all** collected app execution record **do**
4:	$D \leftarrow$ distance between start location of e-scooter usage and location of mobile app execution
	$T \leftarrow$ time difference between the start time of e-scooter usage and time of mobile app execution
5:	$V \leftarrow D/T$
	if $T < 15$ min **and** $V < 5$ m/s, **then**
6:	**append** T **to** time list
7:	initial $T \leftarrow$ min {T **in** time list}
8:	**append** initial T **to** initial time list
9:	median access time $\leftarrow$ median of the elements in the initial time list
10:	size of the grid $\leftarrow$ median access time $\times$ average walking speed
11:	**return** size of the grid

As the e-scooters are used to provide transportation for the first and last mile of a user's movement, it is reasonable to assume that they access the e-scooter by walking. People who have an available e-scooter within walking distance will use them; otherwise, users will reluctant to lend them. Thus, it is reasonable to set the size of the grid within the walking distance, and this walking distance can be obtained from the service database.

First, the median access time was calculated by computing the time difference between the initial execution of the mobile application and the beginning of the e-scooter usage. The database stores the app execution record, which accepts the data every time the user opens the mobile application. To prevent any underestimation or overestimation of access time, two constraints were applied: time and speed. As the distance between the two furthest points in the service area is 3 km, it was estimated that people would not walk more than 15 min to access an e-scooter. Therefore, movements of more than 15 min were considered as outliers and were excluded. Users moving faster than 5 m/s were also excluded. This constraint was set to ensure that access to e-scooters was accomplished solely by walking rather than by metro or bus.

Second, the median access time was multiplied by an average walking speed of 1.47 m/s, given that the average age of users was below 28 years (*19*). Algorithm 1 in Table 1 briefly summarizes the whole process of determining the size of the grid.

The median access time was determined to be 166 s. Using this result, the walking distance to access the service was determined to be about 244 m. Rounding up the number, the size of the grid was set as 250 m. The map of the service area with a grid is presented in the right image of Figure 1. The range of the prediction grid does not perfectly match the service area depending on the actual data in the usage records. This is because e-scooters can still be operated outside the service area at limited speed, although some service areas are hardly passed by.

Estimating Unmet Demands

It is important to anticipate the identified demands, that is, demands that have already been met by using e-scooters. However, identifying the extent of unmet demands is also essential to maximize the user utility. In this research, the term "demand" includes both the identified demands and unmet demands. Here, a model is formulated to predict the demand for e-scooters. The database includes the application execution data and the zoom-in data on the map. The zoom-in data show the zoomed-in location on the map and the user's location at that precise moment. If a user zooms in the map in the vicinity of their present position, it was assumed the user is likely looking for a nearby e-scooter.

In this study, when the map was zoomed in within 250 m from the user, it was assumed the user has the intention to use an e-scooter service. The value of 250 m is the size of the grid defined by Algorithm 1. As a result, another 30,323 unmet demands in addition to the 32,136 identified demands were found. Algorithm 2 in Table 2 explains the procedure of collecting unmet demand data in detail.

Converting Sparse Features to Dense Features using an Autoencoder

In this research, the usage records were collected in real time and its future value was predicted on an hourly basis. There was a total of 62,459 demands, with an average of 1,952 demands per day, which amounts to about

Table 2. Algorithm 2: Collection of Unmet Demands

Algorithm 2:	Collection of unmet demands
Input:	1. Map zoom-in record, containing the data where the user zoomed in and the current location of the user
	2. The start time of every e-scooter usage
Output:	List of unmet demands
1:	**for** each map zoom-in record **do**
2:	collect e-scooter usage record of the user
3:	$T \leftarrow$ time difference between map zoom-in time and nearest e-scooter usage time after the map zoom-in time
4:	$D \leftarrow$ distance between the center of the zoom and the location of the user
5:	**if** $T > 15$ min and $D < 250$ m, **then**
6:	**append** the first zoom-in record from the consecutive zoom-ins of the user **to** the unmet demand list
7:	**return** unmet demand list

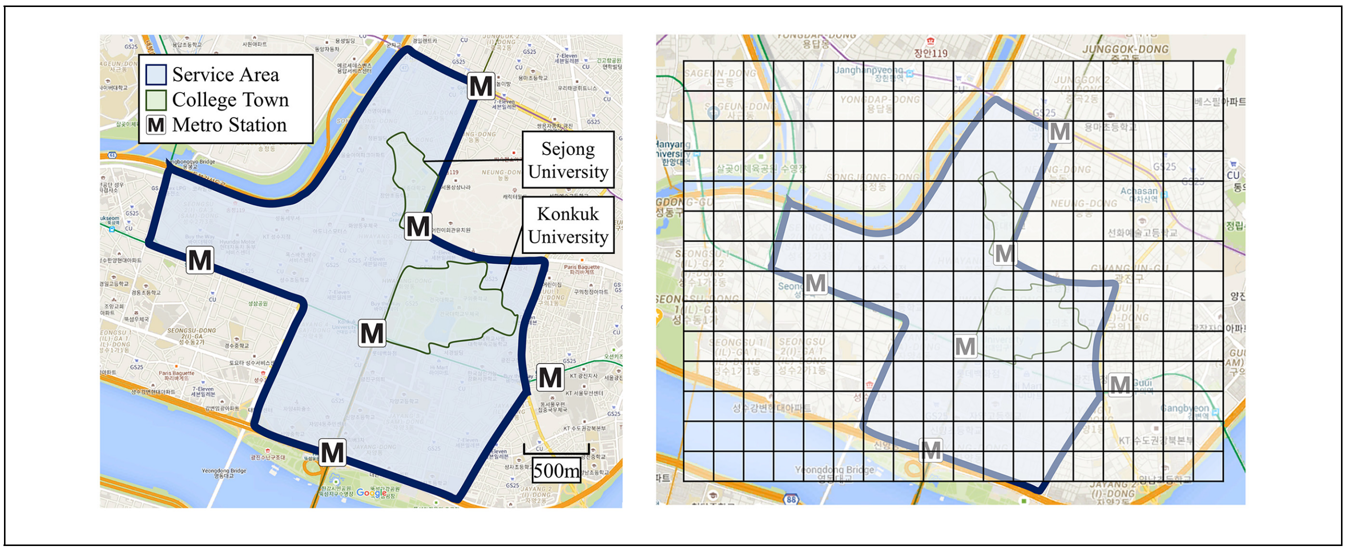

Figure 1. Map of the service area (*left*) and map of the service area with the grid (*right*).

81 demands per hour. The grid used in the study consisted of 14 rows and 18 columns, with a total of 252 cells. Therefore, even if all uses were distributed evenly, more than two-thirds of the grid cells would be filled with zeros. As the actual rides/usages were much more concentrated in limited regions and in limited time spans, 84.5% of the grid cells were filled with zeros. If most cells contain zeros, and even non-zero cells display single-digit demands, the prediction model would reduce the loss by putting a value close to zero in all cells. Figure 2 illustrates a typical outcome of this phenomenon. The prediction result on the right side of Figure 2 shows that most of the values are close to zero. In this case, although the loss decreases as the iteration proceeds, it hardly warrants the model's accuracy. It is unreasonable to state that the model has learned how to predict future demands.

In Figure 2, the previous 5 h of demand data were used as input to forecast demand for the next hour. The five-layer LSTM was used as the model, which has a similar structure as the model described later. (The model described later includes an autoencoder in the front and back of RNN.) With naïve demand input, the outcome values were close to zero in most grid cells. In this model, the mean squared error (MSE) loss converged to 0.00679.

To deal with this problem, an autoencoder was used, which is an artificial neural network-based, unsupervised, representation transforming algorithm. The autoencoder effectively extracts the latent features inside the data, and the latent features can function as compressed data. Figure 3 illustrates the structure of the convolutional autoencoder designed in this study. A convolutional autoencoder with three convolutional layers was used to encode the data, and another with two transposed convolutional layers was used to decode the data. In the prediction model, the latent feature was used as an input to the LSTM cell.

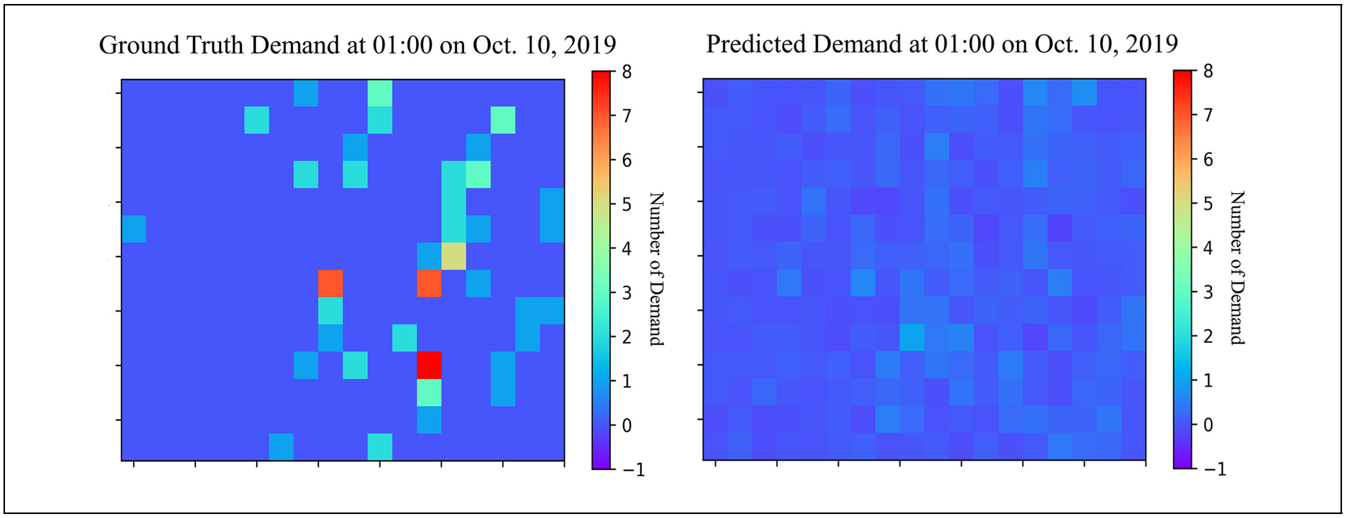

Figure 2. Comparison of ground truth data (*left*) and prediction (*right*) with naïve demand input.

Characteristics of the E-Scooter Service Data

Regarding the e-scooter service, the words "origin" and "demand" have similar meanings. This is because demand arises from a certain starting point, the origin. Therefore, origin and demand are used interchangeably in the literature.

In the dockless e-scooter service, the destination of the previous timestamp often is the origin of the next timestamp. If the e-scooter has never been used, its initial placement can also be a future origin. Assuming there is no relocation of the e-scooters, the initial placement and the last destination of an e-scooter are the only possible origin of the next time step. This rather simple idea is detailed in the formula below:

$$\text{Origin at } n\text{th Time Step} \in \text{Initial Placement} + \sum_{k=1}^{n-1} \text{Movement in } k\text{th Time Step} \tag{1}$$

$$\text{Movement in } k\text{th Time Step} = \text{Destination in } k\text{th Time Step} - \text{Origin in } k\text{th Time Step} \tag{2}$$

The key idea here is that the performance is improved by applying destination data to the model. Even if the heterogeneity of the data is increased with the mixed input of the origin and the destination, it leads to an ameliorated performance. It is a result of the circulating nature of the dockless e-scooter service: one user's destination converted to another user's origin.

Model Construction

Figure 4 provides a visual representation of the transformation in dimension of the input data at each step. The raw origin (demand) and destination matrices are

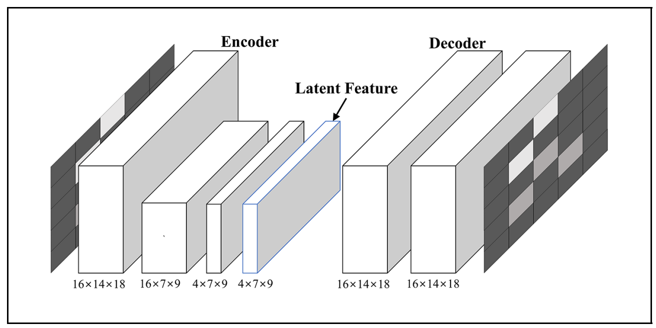

Figure 3. Structure of the convolutional autoencoder.

converted to unfolded latent features via the encoder, unfolding the latent feature vector. Figure 5 presents a schematic structure of the entire model. The prediction model, which is composed of five-layer LSTM or GRU cells, forecasts the next time step in the shape of the unfolded latent feature. Then, the output of the model is folded to a latent feature with the dimensions of $4 \times 7 \times 9$ and works as an input for the decoder of the convolutional autoencoder.

The ERD structure is a linearly coupled form, consisting of three parts: encoder, RNN cell, and decoder. This coupled form can disrupt the performance of the model by accumulating the error of each part. However, as shown in Figure 2, the predictive power of unencoded samples is useless. The high performance obtained using the ERD structure will be introduced later in the results section.

Results

Feature Transformation by Autoencoder and Its Performance

Results confirmed that the loss of the convolutional autoencoder was less when the origin and destination data

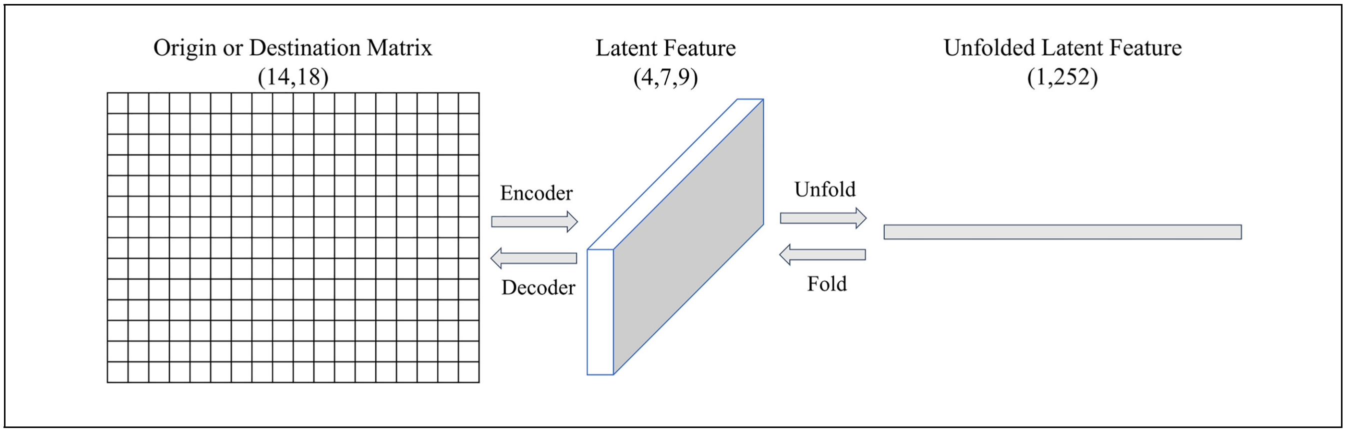

Figure 4. Detailed dimension of input data.

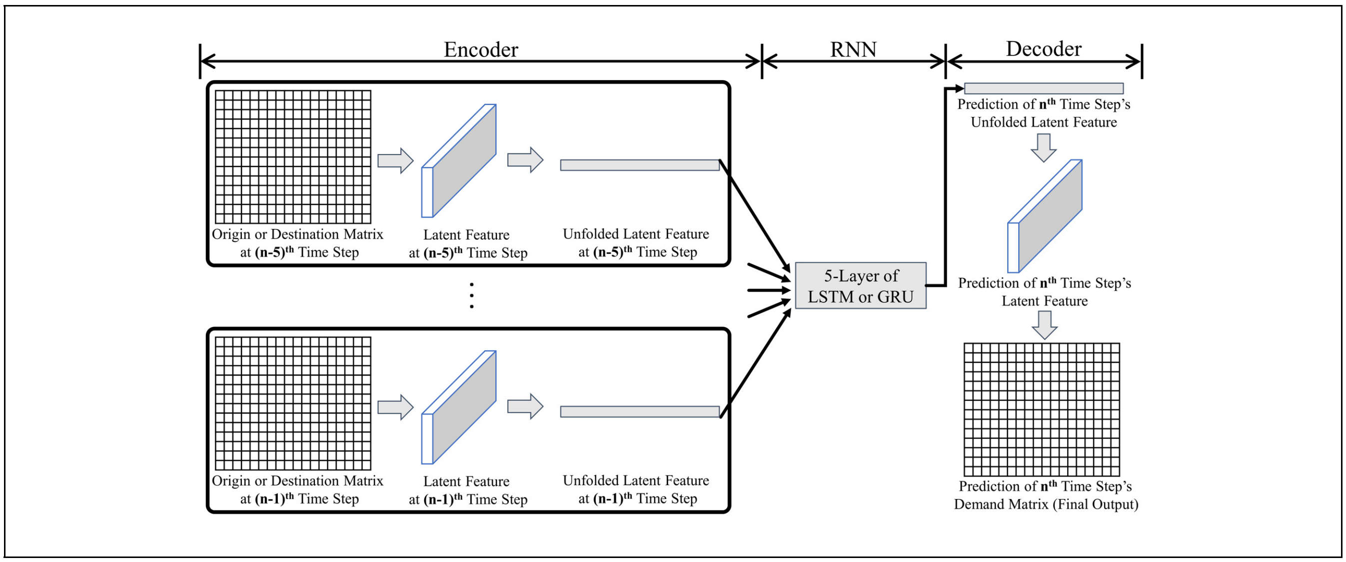

Figure 5. The ERD structure of the prediction model.
Note: ERD = encoder–recurrent neural network–decoder; LSTM = long short-term memory; GRU = gated recurrent unit; RNN = recurrent neural network.

were used together rather than when solely the former were used. Figure 6 shows an example of the output of an autoencoder.

The difference is not evident enough to be visible to the naked eye. This means that both models showed reasonable encoding and decoding capabilities; however, there was a difference in their numerical value of losses. Using the origin data as the only input, MSE was 0.423, but combined the origin and destination data as an input, MSE decreased to 0.253. Also, the standard deviation of the MSE was 0.0349 for the autoencoder that used the origin and the destination simultaneously, proving that this mixed input leads to a more credible result. With only the origin data, the learning process did not proceed at all, the MSE loss diverging to a weird value. Therefore, it is more effective to use the origin and the destination data together to enhance the absolute performance and stability of the model.

Predicting Encoded Feature using Deep Learning Algorithm

The prediction performance of an algorithm was verified repeatedly by four different conditions: (1) the type of activation function, (2) the type of RNN cell, (3) whether the destination data were included in the input, and (4) the learning rate. Ten replicates were conducted for each different combination of conditions.

Table 3 contains the overall result of each model. There was no apparent difference between GRU and LSTM. In some cases, GRU was the predominant algorithm, but the opposite result also appeared frequently.

Table 3. Performance of Prediction Model in Various Conditions

Type of activation function		Without non-linear			Sigmoid			ReLU			ELU		
Type of RNN cell and input	Learning rate	0.0005	0.0010	0.0015	0.0005	0.0010	0.0015	0.0005	0.0010	0.0015	0.0005	0.0010	0.0015
GRU cell with only origin data	Avg. MSE loss	0.00080	0.00066	0.00087	0.00096	0.00127	0.00173	0.00073	0.00056	0.00070	0.00079	0.00108	0.00088
	SD (σ)	0.00006	0.00005	0.00032	0.00005	0.00019	0.00031	0.00003	0.00004	0.00019	0.00006	0.00171	0.00041
LSTM cell with only origin data	Avg. MSE loss	0.00088	0.00070	0.00070	0.00092	0.00096	0.00218	0.00084	0.00070	0.00053	0.00087	0.00068	0.00070
	SD (σ)	0.00003	0.00005	0.00003	0.00002	0.00007	0.00000	0.00003	0.00003	0.00002	0.00002	0.00002	0.00004
GRU cell with O. and D. data	Avg. MSE loss	**0.00047**	**0.00041**	0.00058	0.00061	0.00081	0.00112	**0.00044**	0.00054	0.00065	**0.00040**	0.00060	0.00076
	SD (σ)	0.00002	0.00003	0.00030	0.00002	0.00012	0.00023	0.00002	0.00010	0.00011	0.00003	0.00024	0.00015
LSTM cell with O. and D. data	Avg. MSE loss	**0.00047**	**0.00042**	0.00063	0.00060	0.00137	0.00137	**0.00045**	**0.00036**	0.00065	**0.00046**	0.00058	0.00061
	SD (σ)	0.00004	0.00002	0.00016	0.00004	0.00000	0.00000	0.00001	0.00004	0.00011	0.00002	0.00070	0.00020

Note: Avg. = average; SD = standard deviation; O. = origin; D. = destination; MSE = mean squared error; LSTM = long short-term memory; GRU = gated recurrent unit; RNN = recurrent neural network; ReLU = rectified linear unit; ELU = exponential linear unit. The bold underlined entries indicate the MSE, which was less than 0.00050.

Computing the overall mean of MSE, the values of GRU and LSTM were 0.00077 and 0.00078, respectively, indicating almost an identical performance. The type of data input was found to be a more important factor that determined the performance than the type of RNN cell. When both the origin and destination data were used, the average MSE was reduced to 0.00064, but when only the origin data were used, the average MSE reached 0.00090. An average MSE of less than 0.00050 also occurred when the origin and destination data were used together. The lowest average MSE loss was obtained with rectified linear unit (ReLU) as the activation function, the LSTM RNN cell, and the learning rate of 0.00010.

Results of the Predictions of Demands

The top row of Figure 7 shows the ground truth of demand, and the bottom row shows the results predicted by the model using the ERD structure. It is apparent that there was a noticeable increase in practicality compared with the naïve input in the middle row. The accuracy also increased as the MSE loss decreased by 95%, from 0.00679 to 0.00036.

Figure 8 shows that the enhancement of the model was also pronounced in identifying the largest number of future demand among all grid cells of each time step. In particular, it can be observed that the model identifies peak time and off-peak time of the service. Owing to its performance in predicting the relative demand and the absolute maximum demand, the model is a well-functioning spatiotemporal demand prediction model.

Conclusions

Prior research has focused on the station-based demand prediction problem, and the target was mainly a bike-sharing service. For instance, Lin et al. (*7*) used the GCN to predict a station-level demand for a bike-sharing service in New York City. Ai et al. (*13*) attempted to predict demand for a dockless bike-sharing problem, but avoided dealing with the problem of sparse data by setting the grid size as long as 4 km × 4 km. Besides, these studies did not consider the intrinsic circulating nature of a personal mobility service, and they only treated the identified demand based on the available usage data. The present study suggests using a novel algorithm to establish a methodology for e-scooter research by setting the grid architecture and including the unmet demand properly. The encoded feature was used to extract the spatiotemporal meaning of the data and to make future predictions. Using the demand and destination data as input simultaneously showed a creditable result in both the autoencoder and prediction models. When ReLU was used as the activation function, the LSTM RNN cell,

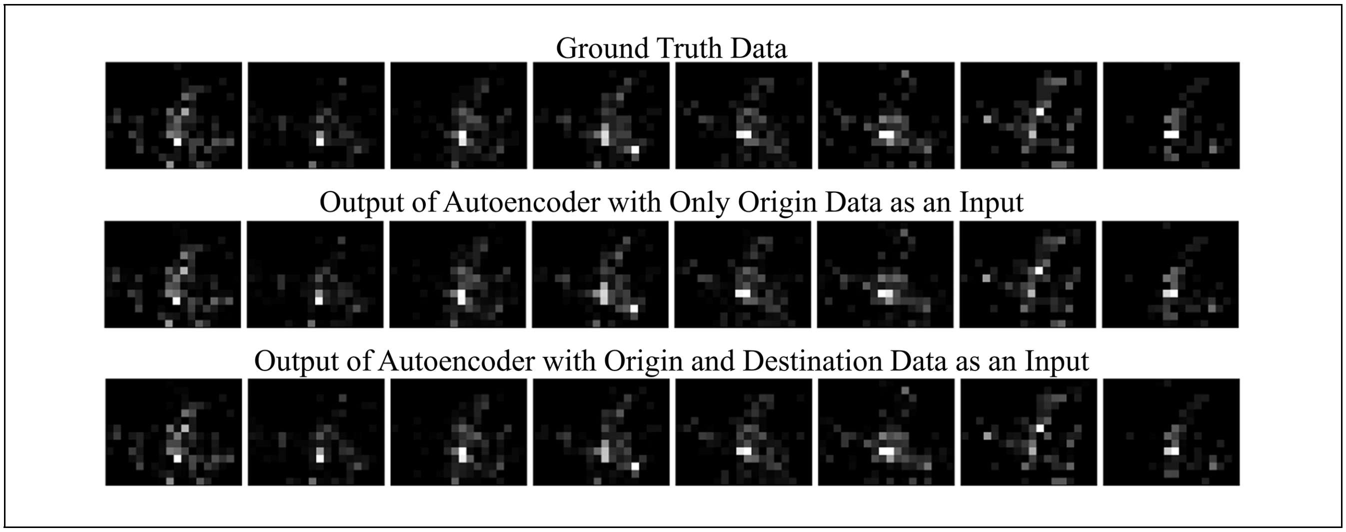

Figure 6. Reconstructed data of the convolutional autoencoder by varying the input.

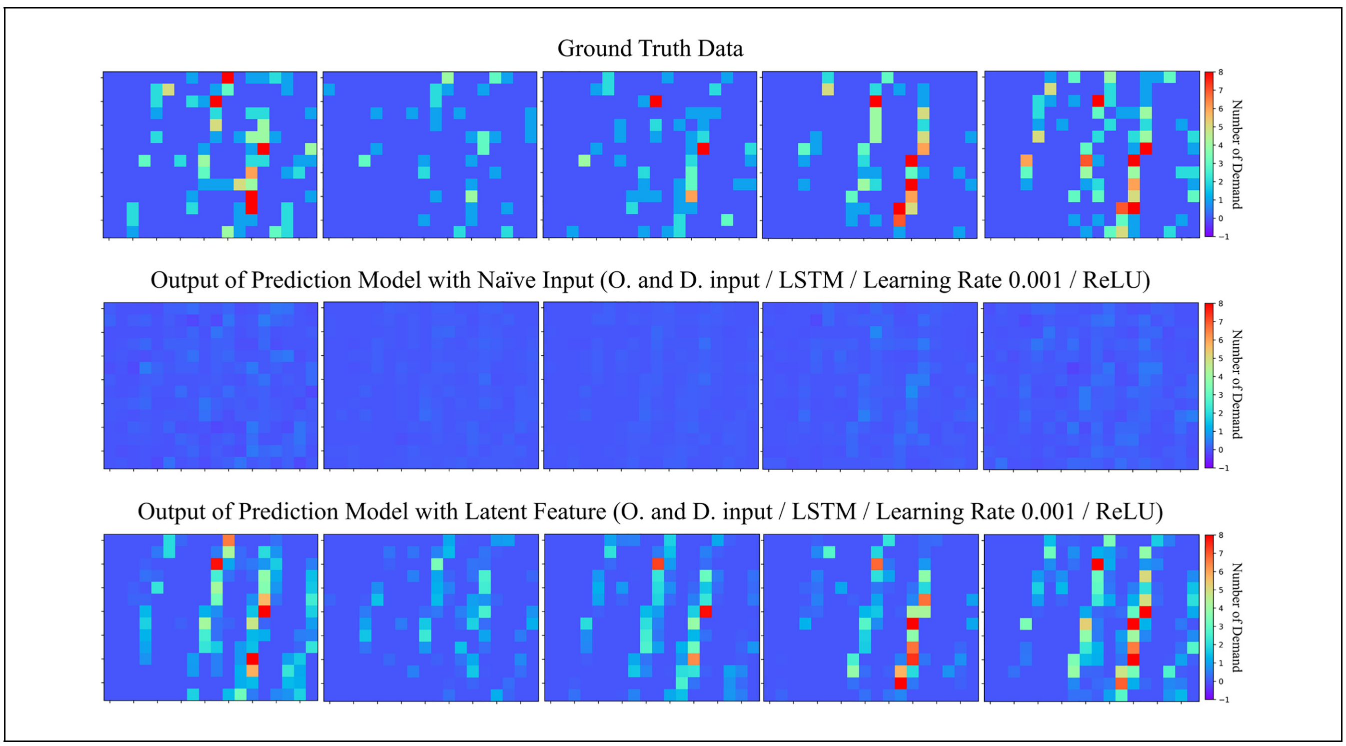

Figure 7. Examples of the ground truth data (*top*), the prediction data with naïve input (*middle*), and the prediction data with latent feature input (*bottom*).
Note: LSTM = long short-term memory; ReLU = rectified linear unit; O = origin; D = destination.

and the learning rate of 0.00010, it led to a dominant performance and stability.

The findings of this research extended the current personal mobility studies to the field of e-scooters, considering the intrinsic characteristic of the service correspondingly. Rather than just extending the research field, this research also proposed the methodological principle in studies regarding e-scooters. To the best of the authors' knowledge, this is the first study that has considered multiple aspects of the e-scooter service to make accurate predictions. The ERD structure used in this research can also be used in other research efforts with sparse data problems.

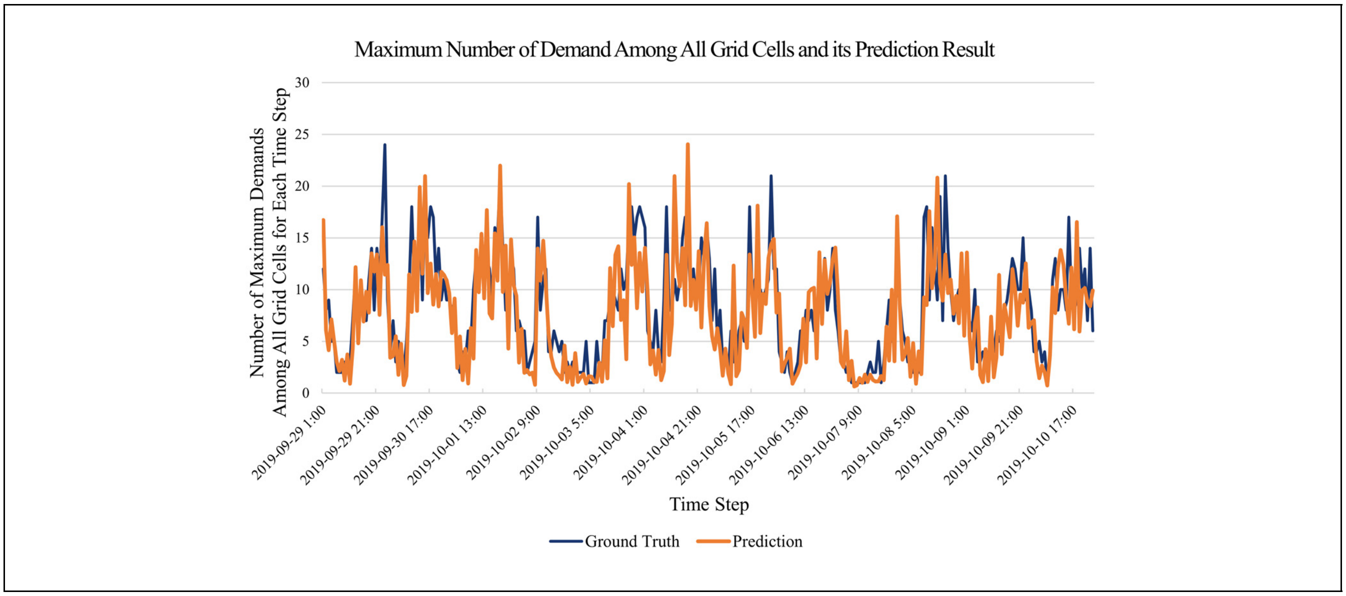

Figure 8. Maximum number of demand among all grid cells in each timestamp and its prediction result.

The direction of future research regarding this study can be divided into three categories. The first is to compare the candidate methodology for demand prediction in detail. In addition to prediction accuracy, which was compared numerically in the current study, other criteria such as execution time and required computational power can be analyzed quantitatively. The second is to replace the component inside the ERD structure with advanced elements. LSTM RNN cell can be modified to a transformer (*20*) and the GCN can be added into the autoencoder to represent the relationship between cells. The third is to cover various features such as the geographic characteristic of each grid cell; for example, the current model did not consider whether the region is in a college town, a residential area, or a commercial area. Future research is expected to reflect the spatial characteristics of a region, thereby providing a more precise prediction model.

Acknowledgments

The physical and technical support of Deer Corporation in the whole process of research is truly appreciated. The computing resource was provided by the "HPC Support" Project, which is supported by the 'Ministry of 2 Science and ICT' and NIPA.

Author Contributions

The authors confirm contribution to the paper as follows: study conception and design: S.-W. Ham, J.-H. Cho, D.-K. Kim; data collection: S.-W. Ham, S. Park; analysis and interpretation of results: S.-W. Ham, S. Park, J.-H. Cho, D.-K. Kim; draft manuscript preparation: S.-W. Ham, J.-H. Cho. All authors reviewed the results and approved the final version of the manuscript.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: Basic Science Research Program through the National Research Foundation of Korea, funded by the Ministry of Science and ICT (2020R1F1A1074395).

References

1. Shaheen, S., and A. Cohen. *Shared Micromoblity Policy Toolkit: Docked and Dockless Bike and Scooter Sharing*. Transportation Sustainability Research Center, UC Berkeley, 2019. http://dx.doi.org/10.7922/G2TH8JW7.
2. Ajao, A. Electric Scooters and Micro-Mobility: Here's Everything You Need to Know. https://www.forbes.com/sites/adeyemiajao/2019/02/01/everything-you-want-to-know-about-scooters-and-micro-mobility/#66bffe855de6. Accessed November 15, 2020.
3. Chen, L., D. Zhang, L. Wang, D. Yang, X. Ma, S. Li, Z. Wu, G. Pan, T. Nguyen, and J. Jakubowicz. Dynamic Cluster-Based Over-Demand Prediction in Bike Sharing Systems. *Proc., 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, Association for Computing Machinery*, Heidelberg, 2016.
4. Chang, P. C., J. L. Wu, Y. Xu, M. Zhang, and X. Y. Lu. Bike Sharing Demand Prediction Using Artificial Immune

System and Artificial Neural Network. *Soft Computing*, Vol. 23, 2019, pp. 613–626.

5. Patil, A., K. Musale, and B. P. Rao. Bike Share Demand Prediction Using Random Forests. *International Journal of Innovative Science, Engineering & Technology*, Vol. 2, 2015, pp. 1218–1223.
6. Zhou, Y., L. Wang, R. Zhong, and Y. Tan. A Markov Chain Based Demand Prediction Model for Stations in Bike Sharing Systems. *Mathematical Problems in Engineering*, Vol. 2018, 2018, pp. 1–8.
7. Lin, L., Z. He, and S. Peeta. Predicting Station-Level Hourly Demand in a Large-Scale Bike-Sharing Network: A Graph Convolutional Neural Network Approach. *Transportation Research Part C: Emerging Technologies*, Vol. 97, 2018, pp. 258–276.
8. Kim, T. S., W. K. Lee, and S. Y. Sohn. Graph Convolutional Network Approach Applied to Predict Hourly Bike-Sharing Demands Considering Spatial, Temporal, and Global Effects. *PLoS One*, Vol. 14, 2019, p. e0220782.
9. Mikolov, T., M. Karafiát, L. Burget, J. Černocký, and S. Khudanpur. Recurrent Neural Network Based Language Model. *Proc., 11th Annual Conference of the International Speech Communication Association*, Makuhari, 2010.
10. Chen, P. C., H. Y. Hsieh, X. K. Sigalingging, Y. R. Chen, and J. S. Leu. Prediction of Station Level Demand in a Bike Sharing System Using Recurrent Neural Networks. *Proc., 2017 IEEE 85th Vehicular Technology Conference (VTC Spring)*, Sydney, IEEE, New York, 2017.
11. Pan, Y., R. C. Zheng, J. Zhang, and X. Yao. Predicting Bike Sharing Demand Using Recurrent Neural Networks. *Procedia Computer Science*, Vol. 147, 2019, pp. 562–566.
12. Xu, C., J. Ji, and P. Liu. The Station-Free Sharing Bike Demand Forecasting with a Deep Learning Approach and Large-Scale Datasets. *Transportation Research Part C: Emerging Technologies*, Vol. 95, 2018, pp. 47–60.
13. Ai, Y., Z. Li, M. Gan, Y. Zhang, D. Yu, W. Chen, and Y. Ju. A Deep Learning Approach on Short-Term Spatiotemporal Distribution Forecasting of Dockless Bike-Sharing System. *Neural Computing and Applications*, Vol. 31, 2019, pp. 1665–1677.
14. Li, Y., Z. Zhu, D. Kong, M. Xu, and Y. Zhao. Learning Heterogeneous Spatial-Temporal Representation for Bike-Sharing Demand Prediction. *Proc., 33rd AAAI Conference on Artificial Intelligence*, Honolulu, Hawaii, 2019, pp. 1004–1011.
15. Xue, A. Y., J. Qi, X. Xie, R. Zhang, J. Huang, and Y. Li. Solving the Data Sparsity Problem in Destination Prediction. *The VLDB Journal*, Vol. 24, No. 2, 2015, pp. 219–243.
16. Vincent, P., H. Larochelle, I. Lajoie, Y. Bengio, and P. A. Manzagol. Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion. *Journal of Machine Learning Research*, Vol. 11, 2010, pp. 3371–3408.
17. Gers, F. A., J. Schmidhuber, and F. Cummins. Learning to Forget: Continual Prediction with LSTM. *Proc., 9th International Conference on Artificial Neural Networks: ICANN '99*, Edinburgh, 1999.
18. Chung, J., C. Gulcehre, K. Cho, and Y. Bengio. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *arXiv Preprint arXiv:1412.3555*, 2014.
19. Bowman, B. L., and R. L. Vecellio. Pedestrian Walking Speeds and Conflicts at Urban Median Locations. *Transportation Research Record: Journal of the Transportation Research Board*, 1994. 1438: 57–73.
20. Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. *arXiv Preprint arXiv:1706.03762*, 2017.