WILEY | Hindawi

Research Article

A Dual-View Approach for Multistation Short-Term Passenger Flow Prediction in Bus Transit Systems

Gang Luo (ID)**, Haoxuan Kuang** (ID)**, Dongran Zhang** (ID)**, Kunxiang Deng** (ID)**, and Jun Li** (ID)

Guangdong Provincial Key Laboratory of Intelligent Transportation System, School of Intelligent Systems Engineering, Sun Yat-sen University, Shenzhen 518107, China

Correspondence should be addressed to Jun Li; stslijun@mail.sysu.edu.cn

Timely and accurate prediction of bus passenger flow plays a crucial role in uncovering real-time traffic demand, presenting an essential and formidable challenge in the realm of bus scheduling and management. The extensive application of deep learning methods in transit passenger flow prediction can be attributed to their exceptional ability to effectively capture spatiotemporal features, resulting in superior performance. However, prevailing deep learning models in transit passenger flow prediction tend to ignore the data enhancement. Additionally, the predominant focus on a single station in the prediction task presents challenges in effectively capturing the spatiotemporal features of the entire network. A model named TSD-ST is proposed to better accomplish the task of predicting short-term transit passenger flow at multistation. The TSD-ST model leverages time series decomposition for data enhancement. Simultaneously, in addition to considering the adjacency graph, the similarity of all the stations of the entire transit network is also considered and uses multigraph convolution and graph fusion modules. This approach enables the TSD-ST model to effectively capture spatiotemporal dependencies. Experiments based on real-world bus transit datasets confirm that the TSD-ST model shows better performance in prediction tasks at 30-min, 60-min, and 90-min time scales, with an average improvement of 21.87%. The effectiveness of each component has been verified through ablation experiments.

1. Introduction

Accurate and timely multistation short-term passenger flow prediction is important for bus transit management and scheduling. Fluctuations in travel demand or the transportation system's short-term changes can result in undesirable waiting times and congestion for passengers [1], thereby diminishing the appeal of the bus transit system to passengers. Short-term prediction of passenger flow enables bus transit managers to optimize schedules, improve station passenger flow regulation planning, facilitate vehicle operation scheduling, and achieve efficient resource allocation. This approach helps meet the demand for passenger flow while enhancing service quality.

Currently, studies on passenger flow prediction mostly focus on a single station, but due to the complex spatial structure and time-varying features of transit networks, the bus passenger flow at a single station will be simultaneously affected by the spatial and temporal features of the historical passenger flow at stations that are directly or indirectly connected to the entire network [2]. Although some methods have achieved better results in predicting passenger flow at specific stations, complex modeling for each station is inappropriate if one wants to understand the state of the entire network, especially not conducive to future network adjustment and expansion. Therefore, prediction models for a single station cannot dynamically and effectively predict the spatial and temporal distribution and congestion of the entire network, limiting the real-time passenger flow organization, formulation, and adjustment of management strategies. The task of predicting passenger flow at multistation is challenging because transit passenger flow can be affected by many complex factors [3], such as (1) temporal dependencies: bus passenger flow is affected by temporal features, and current passenger flow is correlated with historical passenger flow [4]. Moreover, passenger flow on the same workday or on the

same nonworkday will show similar trends. (2) Spatial dependencies of bus passenger flow exist not only between neighboring stations under the influence of network topology [5] but also between stations that are far away from each other but located within similar urban functional areas. In conclusion, there exists a demand for a multistation short-term bus transit passenger flow prediction approach that adequately incorporates both temporal and spatial dependencies.

On the other hand, factors such as weather conditions and random events can exert an impact on bus passenger flow [6]. It is worth noting that automated fare collection (AFC) data mainly serve as the source of bus passenger flow data, and most of it becomes available only after passengers have boarded the bus. Consequently, the arrival time of the bus at the station to obtain passenger data introduces significant uncertainty in the data collection process [7]. Thus, it is necessary to develop a data enhancement methodology that accounts for both cyclic changes in passenger flow and deviations from cyclic patterns resulting from temporary factors.

In recent years, deep learning has been widely applied to short-term passenger flow prediction tasks [8]. Typical statistical models, such as autoregressive integrated moving average (ARIMA) and its variants [9, 10], are usually used for single time series prediction, which can easily compute and capture the linear features of the data, but ignore potential dependencies between multiple time series under relatively complex traffic conditions. Deep learning is much more complex in terms of structure, with a strong fitting ability for complex function processing, and is more suitable for solving short-term passenger flow prediction problems.

Recurrent neural networks (RNN) and their variants, such as long short-term memory (LSTM) and gated recurrent units (GRU), perform well in time series prediction and are widely used in mainstream research on traffic prediction [11, 12]. However, RNN-based models only focus on capturing local temporal features and ignore spatial features [13]. In order to consider the spatial dependencies of the network, some researchers have constructed prediction models using a convolutional neural network (CNN) after converting the traffic network into images [5, 14]. However, CNN-based models only consider the absolute distance relationship between stations in a two-dimensional Euclidean space and are not conducive to the expansion of network structures in the real world [15]. Graph convolutional network (GCN) provides a more suitable approach for modeling spatial similarity of transit networks compared to CNN, which maintains the real topology of the network and captures the spatial dependencies between stations [16]. Several researchers have made attempts to enhance the effectiveness of GCN through various approaches [17–21]. However, it is important to note that GCN-based models solely focus on spatial features. While some researchers have attempted to integrate GCN and RNN models to create combined prediction models [19, 22], most of these combined models ignore data enhancement and lack comprehensiveness in capturing temporal dependencies, spatial dependencies, and other influential factors. As a result, these models fail to effectively address the challenge of multistation short-term transit passenger flow prediction.

A novel model, named TSD-ST, is proposed for accurately predicting multistation short-term bus passenger flow. The proposed model combines time series decomposition (TSD) for data enhancement, capturing both cyclic variations in bus passenger flow patterns and the impact of temporary random events. Additionally, a dual-view multigraph convolution and graph fusion module is employed to consider global spatial dependencies. Consequently, this approach integrates temporal dependencies, spatial dependencies, and other influencing factors. The main contributions of this paper are summarized as follows:

(1) A spatiotemporal model (TSD-ST) is proposed to solve the problem of predicting multistation short-term bus passenger flow.

(2) The temporal dependence of bus transit passenger flow is effectively harnessed through the utilization of a time series decomposition method, and enhancing the quality of the data.

(3) Constructing two types of graphs, namely, adjacency graph and similarity graph. Through the utilization of multigraph convolution and graph fusion, a dual-view approach is employed to comprehensively consider the spatial dependencies of passenger flows between different stations

(4) Experiments were conducted using a real-world bus transit dataset to evaluate the performance of the TSD-ST model in predicting multistation short-term bus passenger flow. The experimental results demonstrate that the TSD-ST model exhibits excellent predictive capability.

The remaining sections of this paper are structured as follows: Section 2 presents a comprehensive review of pertinent research in the domain of passenger flow prediction. In Section 3, a systematic overview of the components comprising the TSD-ST model is provided. To validate the performance of the proposed model, Section 4 conducts experiments using real-world bus transit datasets. Finally, Section 5 summarizes the findings of this study and discusses potential directions for future research.

2. Related Literature

2.1. Statistical Methods. Initial attempts at passenger flow prediction have focused on linear models. Among the classical prediction models, linear regression models and Kalman filter-based methods are more commonly used in the passenger flow prediction literature. In 2009, a general linear regression prediction of passenger flow was conducted using public transportation smart card data [23], and in 2011, a Kalman filter-based method was developed to predict passenger flow by using the transaction records of automated fare collection (AFC) system and the video surveillance systems equipped in buses and stations equipped with video surveillance systems to predict the short-term passenger flow at each bus station [24].

Due to the spatiotemporal nature of most of the data used for passenger flow prediction, among different linear models, autoregressive moving average (ARMA) and

autoregressive integrated moving average (ARIMA) models have received attention from researchers. One study proposed an interactive multiple model- (IMM-) based approach combined with time series methods to predict short-term passenger flow on bus lines [25]. There are several studies that address the problem of passenger flow prediction through ARIMA [9, 10, 26].

Statistical methods can easily compute and capture the linear characteristics of the data. However, such methods rely on static assumptions and are largely affected by fluctuating traffic data [27]. In addition, they are difficult to reflect the nonlinear and complex features of passenger flow.

2.2. Machine-Learning Methods. Traditional machine-learning methods are better at fitting complex data, and machine-learning techniques have been widely used by scholars in the field of short-term passenger flow prediction. Some scholars have developed a bus transit passenger flow prediction model based on least squares and support vector machines (LS-SVM) for predicting passenger flow on bus transit lines [28]. A Gaussian process-based approach is proposed to model and predict bus passenger flow [29]. Some scholars developed prediction models using random forests and regression trees and used automatic vehicle location (AVL) and automatic passenger counting (APC) data from bus fleets to predict passenger demand [30]. One scholar developed a hybrid model combining wavelet and SVM models [31]. Some scholars have used random forest regression to predict short-term passenger flow in railroad transportation systems [32].

Among different machine-learning models, artificial neural network (ANN) has gained the attention of scholars, and in 2014, some scholars applied two neural network models, Elman and BP, to solve the problem of passenger flow prediction, and the comparison of the prediction results of the two methods found that the Elman method can achieve higher prediction accuracy [33]. In 2019, some scholars used the radial basis function (RBF) network method to solve the problem of short-term passenger flow prediction at stations, and the research results show that the prediction accuracy evaluation indexes are all less than 1.5%, and the model prediction performance is better [34]. In addition, scholars have proposed various network models, such as multitask convolutional neural network (MTUNN) and parallel integrated neural network [35], and some scholars have combined neural networks with other linear and nonlinear methods simultaneously in their research [36, 37].

Machine-learning methods are difficult to achieve good results in complex networks with many nodes [38]. This is because most of them rely on complex artificial feature engineering, which leads to insufficient robustness in modeling massive data and cannot handle raw spatiotemporal data.

2.3. Deep Learning Methods. The emergence of real-time data collection and information dissemination systems and the increasing complexity of transportation networks have accelerated the popularity of deep learning models. In 2017, some scholars proposed the deep neural network (DNN),

applying the SAE-DNN model for passenger flow prediction, and the results of the study found that the model can get better prediction results for stations with different passenger flow features [4]. In recent years, several scholars have proposed the LSTM model to solve the problem of passenger flow prediction, and the research results found that the LSTM model has a higher average prediction accuracy of its algorithm compared with other neural network models [4, 39–41]. The GRU model has the advantage of fewer parameters and faster training compared to the LSTM model [42].

The emergence of GCN provides a new idea for traffic network modeling. GCN views the whole network as a graph, which preserves the real topology of the network and extracts the spatial features more effectively [16, 18]. In addition, GCN greatly maintains the global nature of the network by convolving the entire structured graph, which is theoretically superior to CNNs that can only capture adjacent spatial patterns due to the limited size of the kernel window.

There are many researchers based on the GCN model and fused it with temporal features to achieve good results in traffic prediction problems. For example, a diffusion convolutional recurrent neural network (DCRNN) [17] has been proposed to model traffic flow as a diffusion process on a directed graph and capture spatial dependencies using bidirectional random wandering on the graph. A new architecture called Graph WaveNet has been developed which uses an adaptive adjacency matrix to capture hidden spatial dependencies [43]. A novel enhanced dynamic graph convolutional network model (RDGCNI) [44] has been proposed, which generates graph adjacency matrices representing dynamic spatiotemporal dependencies between sites. A parallel-structured deep learning model consisting of a graph convolutional network and a stacked bidirectional unidirectional long-term and short-term memory network (GCN-SBULSTM) has been proposed [45], which treats the subway network as a structured graph and introduces a K-hop matrix that combines the travel distance, the population flow rate, and the adjacency to capture the dynamic spatial correlation between subway stations. A deep learning architecture combining residual networks (ResNet), GCN, and LSTM (called "ResLSTM") has been proposed [46], and a novel metro passenger flow prediction based on the optimization of the parameter estimation of graph convolutional gated recurrent neural network model (TMFO-AGGRU) [47], and a T-GCN model combining GCN and GRU was proposed [18]. The graph convolutional network is used to capture the topology of the road network and model the spatial dependencies, and the GRU is used to capture temporal variations of traffic data on the road.

3. Methodology

3.1. Problem Statement. The transit passenger flow prediction problem addressed in this study can be classified as a spatiotemporal prediction problem. As shown in Figure 1, the bus passenger flow data for each time interval form a spatiotemporal graph. First of all, the bus network structure can be represented as a graph structure $\mathbf{G} = (\mathbf{D}, \mathbf{L}, \mathbf{W})$, where $\mathbf{D} =$
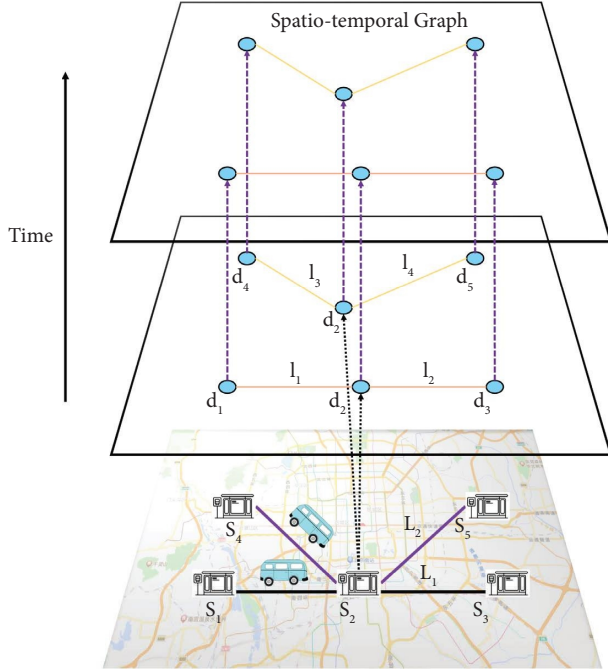
Figure 1: Spatial and temporal graph of bus passenger flow.

$\{d_1, d_2, \cdots, d_N\}$ denotes the set of nodes of all stations in the bus network, corresponding to the real-world station $\mathbf{S}$, and it is important to note that the same station belonging to different lines is distinguished into two stations, and N denotes the number of stations. $\mathbf{L}$ denotes the connecting edges between the bus stations, including the physical connecting edges and virtual connecting edges. $\mathbf{W}$ is the weight of an edge. In addition, the bus passenger flow is not only affected by the spatial structure of the bus network but also by the passenger flow in the historical period. At moment t, the passenger flow of each station in the network is $X(t) \in \mathbb{R}^{N \times T}$. To summarize, the forecasting problem can be expressed as follows:

$$\mathbf{M} = F(\mathbf{X}, \mathbf{G}, \mathbf{W}), \tag{1}$$

where $\mathbf{M} \in \mathbb{R}^N$ is the predicted bus passenger flow at N stations at time T', $\mathbf{X} \in \mathbb{R}^{N \times T'}$ is the historical passenger flow at station N, F is the deep learning model, $\mathbf{G}$ is the spatial graph structure of the bus network, and $\mathbf{W}$ is the learnable parameters.

The general framework of the TSD-ST model is shown in Figure 2. First, in feature engineering module, the dataset is decomposed into three items, trend, cycle, and effect, using time series decomposition (TSD), and the Trend and Cycle are nested and then trained by GRU. Next, two graph structures are constructed based on the spatial dependence of neighboring stations and the spatial dependence of functionally similar stations, and the output values of the GRUs are passed through the multilayer GCN, respectively, and then the outputs of the two graphs are summed up and

put into the temporal attention, and finally, they are nested with the effect into the MLP decoder to complete the task of multistation passenger flow prediction.

3.2. Feature Engineering Module. Passenger flow at bus stations has obvious cyclical features [48], and unlike other modes of transportation, bus passenger flow is subject to greater uncertainty and can be affected by multiple contingencies. Therefore, in order to accurately represent the time series information, a time series decomposition (TSD) model is used to extract the temporal features [49, 50].

TSD is a common and flexible conception for mode decomposition, and its benefits are reflected in our approaches to modeling the time series features, namely, trend, cycle, and effect. By using the backtracking mechanism of RNNs to define the trend vector, the trend term can model near-future changes, e.g., upward, downward, turning, and leveling. By fitting a Fourier series, which consists of a set of triangle signals, the cycle term can model the multiple periodic patterns in passenger flow changes, including peaks and off-peaks, day and night, and workdays and nonworkdays. By telling how much the current passenger flow deviates from the "cycle," the effect term can model arranged or casual events, e.g., holidays, extreme weather, and special events.

As shown in Figure 3, the temporal features are analyzed through trend, cycle, and effect, and equation (2) represents the relationship between the three features, where $Y(t + \lambda')$ is the predicted value of passenger flow at the moment of $t + \lambda'$, λ' is the predicted time step, $\mathbf{A}(t_b)$ is the trend term obtained from the change of the historical passenger flow data, the cyclical change is included in $\mathbf{C}(t_f)$, and $E(t)$ represents the moment t of the random effect.

$$Y(t + \lambda') = \mathbf{A}(t_b) + \mathbf{C}(t_f) + E(t). \tag{2}$$

"Trend" feature is used to predict recent changes in the model (e.g., upward, downward, steering, and flat). Future trends can be estimated from historical data with respect to neighboring times, so a backtracking mechanism is used to define the trend term, as shown in equation (3), where the prediction interval λ denotes the size of the step.

$$\mathbf{A}(t_b) = [y(t - \lambda + 1), y(t - \lambda + 2), \cdots, y(t)]. \tag{3}$$

"Cycle" feature is that the standard Fourier series can be used to model periodic variations, as shown in equation (4). Theoretically, the Fourier series can fit any function as long as there are an infinite number of triangular signals; however, in practice, it is too costly to build such a model. Therefore, the number of sines and cosines is critical to the fit. Given the time, the number of triangular signals N, and the period length T, equation (4) can be converted to (5). In addition, in equation (5), let $\omega = [a_1, b_1, \cdots, a_n, b_n]^T$, $\nu(t) = [\cos 2\pi t/T, \sin 2\pi t/T, \cdots, \cos 2\pi nt/T, \sin 2\pi nt/T]^T$, which can be obtained as equation (6). Thus, the periodicity fitting can be solved as a multivariate linear regression task. Therefore, the periodicity characteristic can be expressed by equation (7).
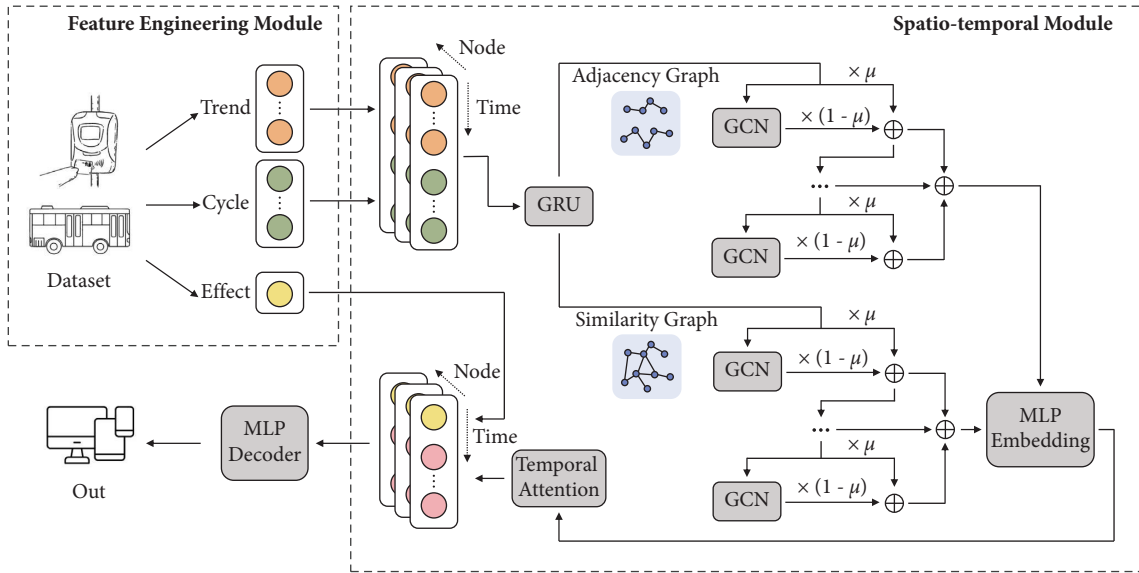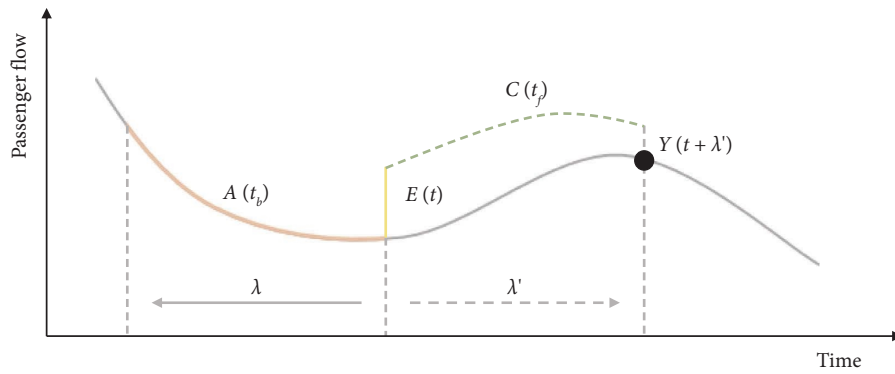
FIGURE 2: TSD-ST model.

FIGURE 3: Time series decomposition.

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx), \tag{4}$$

$$c(t) = \frac{a_0}{2} + \sum_{n=1}^{N} \left(a_n \cos \frac{2\pi nt}{T} + b_n \sin \frac{2\pi nt}{T} \right), \tag{5}$$

$$c(t) = \frac{a_0}{2} + v(t) \times \omega, \tag{6}$$

$$\mathbf{C}(t_f) = \left[c(t+1), c(t+2), \cdots, c\left(t+\lambda'\right) \right]. \tag{7}$$

"Effect" feature is used to consider that, in addition to trends and cycles, bus transit passenger flow may be affected by scheduled or fortuitous events, such as holidays, extreme weather, and special events. Assuming that the passenger flow at the current time is larger than that of the past cycle, indicating that the passenger flow at subsequent moments may also be larger, and the effect feature $E(t)$ can be expressed by equation (8). It should be noted in particular that the historical data of passenger flow come from the data

of passengers boarding the bus and swiping the card, and if there is no bus arriving at the station during the sampling time interval, there will be a situation that the passenger flow is 0. In this case, when calculating the $E(t)$, $y(t)$ is taken as a relatively small value α for calculation, and at the same time, in order to prevent $o(t)$ from being too large, a parameter value β is chosen to constrain the upper limit of $o(t)$.

$$E(t) = o(t) \times \delta(t),$$

s.t.

$$o(t) = \begin{cases} \dfrac{c(t) - y(t)}{y(t)}, & y(t) \neq 0, \\[4mm] \min\left(\dfrac{c(t) - \alpha}{\alpha}, \beta \right), & y(t) = 0, \end{cases} \tag{8}$$

$$\delta(t) = \begin{cases} 1, & |k(t)| > \rho, \\[2mm] 0, & \text{otherwise}. \end{cases}$$

3.3. Spatiotemporal Module

3.3.1. Temporal Features

(1) GRU-Based Temporal Feature Mining. Aiming at the features of large, cyclical, and complex bus transit passenger flow prediction data, a gated recurrent unit (GRU) model is established to deal with temporal features. As an alternative to long short-term memory (LSTM), GRU is widely used for temporal modeling, which is usually implemented with standard convolution or full connectivity. GRU has a strong memory capability, which enables selective memory of the extracted time series features of passenger flow and backward propagation through hidden nodes. GRU possesses several advantages over LSTM, including a smaller parameter count and faster training speed. This makes GRU better equipped to effectively capture the intrinsic patterns within time series data. The specific calculation process is shown as follows:

$$\mathbf{u}_t = \sigma\left(\mathbf{W}_u\left[\mathbf{I}_t, \mathbf{h}_{t-1}\right] + b_u\right), \tag{9}$$

$$\mathbf{q}_t = \sigma\left(\mathbf{W}_r\left[\mathbf{I}_t, \mathbf{h}_{t-1}\right] + b_q\right), \tag{10}$$

$$\mathbf{z}_t = \tanh\left(\mathbf{W}_z\left[\mathbf{I}_t, \left(\mathbf{q}_t \times \mathbf{h}_{t-1}\right)\right] + b_z\right), \tag{11}$$

$$\mathbf{h}_t = \mathbf{u}_t \times \mathbf{h}_{t-1} + \left(1 - \mathbf{u}_t\right) \times \mathbf{z}_t. \tag{12}$$

In spatiotemporal module, $\mathbf{A}(t_b)$ and $\mathbf{C}(t_f)$ embeddings, which have gone through the feature engineering module, are nested as $\mathbf{I}_t$ as input to the GRU, and $\mathbf{W}$ and b represent the weights and biases in the training process. The final output $\mathbf{h}_t$ is obtained by resetting and updating the gating control, which realizes the selective forgetting and remembering of the passenger flow data and achieves the effective mining of temporal features of passenger flow.

(2) Scaled Dot-Product Attention. Scaled dot-product attention is one part of the attention mechanism proposed in the transformer model [51], which allows the model to focus on the relevant part of the input sequence during self-attention. It is computed from the three input data: query, key, and value. Scaled-dot attention in this study can pay more attention to the historical time points that contribute most to the prediction and better capture the temporal relationships. The calculation formula is shown in equation (13). Specifically, the output hidden layer of MLP graph embedding is processed by three linear functions. $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$ are obtained, respectively, as shown in Figure 4.

$$\text{Attention}\left(\mathbf{Q}, \mathbf{K}, \mathbf{V}\right) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V}, \tag{13}$$

where $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$ are the matrix of query, key, and value, respectively. d_k is the dimension of query and keys.

3.3.2. Spatial Features.

To comprehensively encode the relationships between stations, two graphs containing the physical neighborhood of sites and the correlation between
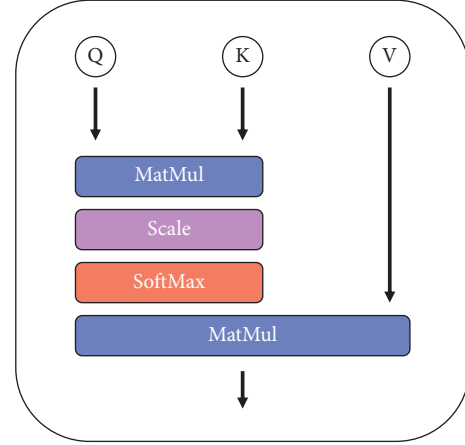

FIGURE 4: Temporal attention.

stations were generated. Among them, the adjacency graph reflects the physical connectivity between stations, while the similarity graph represents the semantic or functional correlation between stations. To better utilize this graph information, multigraph convolution and graph fusion are used in the neural network model. These parallel graph convolution operators are able to process different types of graph data simultaneously and fuse the obtained results to capture the spatial relationships and features between stations more comprehensively.

(1) Adjacency Graph and Similarity Graph. This subsection describes how adjacency graph and similarity graph are constructed. By definition, a graph consists of nodes, edges, and the weights of the edges, so the physical and similar graphs are denoted as $\mathbf{G}_p = (\mathbf{D}, \mathbf{L}_p, \mathbf{W}_p)$ and $\mathbf{G}_s = (\mathbf{D}, \mathbf{L}_s, \mathbf{W}_s)$, respectively. It should be noted that, unlike ordinary road networks, here stations with the same geographic coordinates but belonging to different bus routes are distinguished as different nodes. Physical and similar graphs share the same nodes but have different edges and edge weights. $\mathbf{L}_p$ and $\mathbf{L}_s$ are the sets of edges of different graphs, and for a particular graph $\mathbf{G}_\varphi$ ($\varphi = p, s$), $\mathbf{W}_\varphi \in \mathbb{R}^{N \times N}$ denotes the weights of all edges. Specifically, $\mathbf{W}_\varphi(i, j)$ is the weight of the edges from node i to node j.

Adjacency graph ($\mathbf{G}_p$) is constructed directly from the physical topology of the bus transit system under study. An edge connecting nodes i and j is formed if nodes i and j in E_p connect stations i and j in the real world. In order to compute the weights of these edges, we first construct a physical connectivity matrix $P \in \mathbb{R}^{N \times N}$. As shown in Figure 5, if there exists an edge between nodes i and j, $P(i, j) = 1$, otherwise $P(i, j) = 0$. Finally, we obtain the edge weights $\mathbf{W}_p$. Specifically, $\mathbf{W}_p(i, j)$ is computed by the following equation:

$$\mathbf{W}_p(i, j) = \frac{P(i, j)}{\sum_{k=1}^{N} P(i, k)}. \tag{14}$$

Similarity graph ($\mathbf{G}_s$) is constructed based on similarities between stations. Typically, stations located within similar urban functional areas will have similar passenger flow trends. For example, the boarding passenger flow of bus

1	1	0	0	0	0
1	1	1	0	0	0
0	1	1	1	1	1
0	0	1	1	0	0
0	0	1	0	1	0
0	0	1	0	0	1

Adjacency Graph · Adjacency Matrix · Graph Embedding

1.00	0.65	0.22	0.18	0.28	0.71
0.65	1.00	0.43	0.37	0.74	0.88
0.22	0.43	1.00	0.45	0.69	0.74
0.18	0.37	0.45	1.00	0.72	0.41
0.28	0.74	0.69	0.72	1.00	0.86
0.71	0.88	0.74	0.41	0.86	1.00

1	1	0	0	0	1
1	1	0	0	1	1
0	0	1	0	1	1
0	0	0	1	1	0
0	1	1	1	1	1
1	1	1	0	1	1

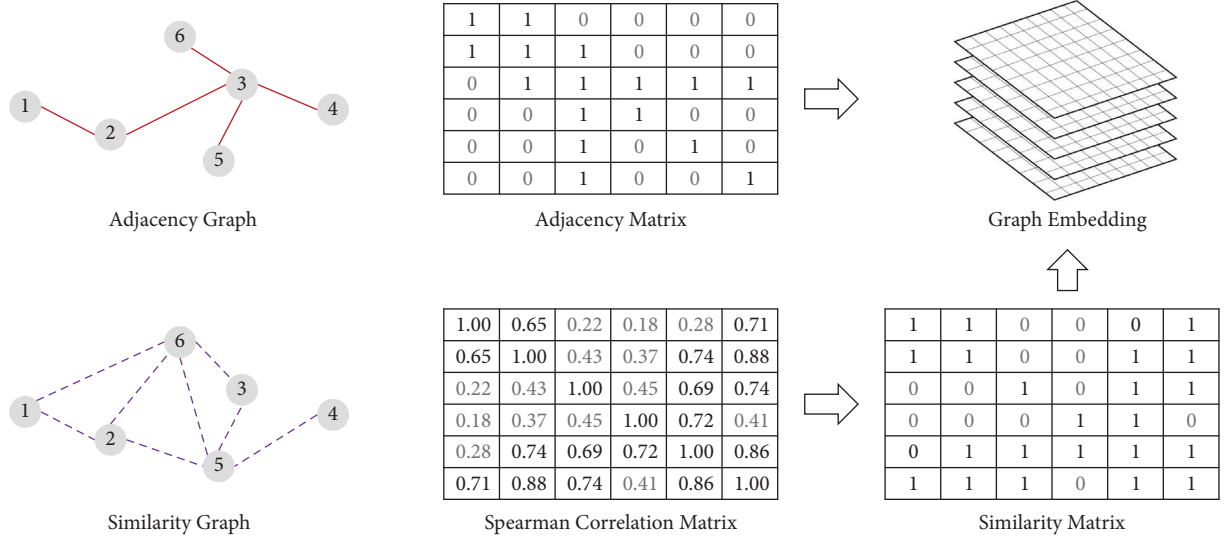Similarity Graph · Spearman Correlation Matrix · Similarity Matrix

Figure 5: Adjacency graph and similarity graph.

stations in office areas will have a significant evening peak on weekdays, while stations in residential areas will have a significant morning peak on weekdays. To quantify this similarity, this paper uses the Spearman rank correlation coefficient to measure this relationship.

The Spearman correlation coefficient is used to deal with the monotonic relationship between two variables and has a value between −1 and 1. The closer the absolute value of the coefficient is to 1, the stronger the correlation between the two variables. Bus transit passenger flow datasets often exhibit a discrete nature and may not adhere to the assumptions of a normal distribution. In contrast to the Pearson correlation coefficient, the Spearman correlation coefficient offers a nonparametric approach that does not rely on assumptions regarding the distribution pattern of the data and is also able to deal with nonlinear relationships. Therefore, when analyzing such datasets, the Spearman correlation coefficient is deemed more appropriate. For a sample size, the historical passenger flow data (i_T, j_T) of the two stations at T moments are first assigned a rank (i_T, j_T) according to their average descending position in the overall data. The correlation coefficient ρ can be expressed as follows:

$$\mathbf{r}(i, j) = \frac{\sum_T (i_T - \overline{i_T})(j_T - \overline{j_T})}{\sqrt{\sum_T (i_T - \overline{i_T})^2 (j_T - \overline{j_T})^2}}. \quad (15)$$

The Spearman correlation matrix is defined as follows:

$$\mathbf{S}(i, j) = |\mathbf{r}(i, j)|. \quad (16)$$

On the basis of the Spearman correlation matrix, the construction of the Spearman matrix is flexible. For example, these virtual edges can be determined by a predetermined similarity threshold ε or weights can be assigned based on the similarity values, as shown in Figure 5, which is a simple example of the Spearman matrix.

$$\mathbf{W}_s(i, j) = \begin{cases} 1, & \mathbf{S}(i, j) \geq \varepsilon, \\ 0, & \mathbf{S}(i, j) < \varepsilon. \end{cases} \quad (17)$$

(2) GCN for Multigraph Convolution and Graph Fusion. Conventional approaches are unable to process graph-structured data efficiently, and it is difficult to perform deep mining from graph-structured data for prediction tasks. A widely used graph neural network, graph convolutional network (GCN), has emerged in recent years, which is capable of extracting useful feature information by performing graph convolution operations on graph-structured data. A major dependency of GCN is the adjacency matrix information; however, a single adjacency matrix may not be able to adequately account for the bus stops' complex relationships. In order to consider these relationships more comprehensively, TSD-ST extends GCN using multigraph convolution and graph fusion.

First, two relationship graphs were constructed to represent the physical adjacencies of bus stops and the correlation relationships between stops, respectively. Together, these two graphs describe the multiple relationships between stops. Next, on each relational graph, multigraph convolutional operations and fusion are performed separately to capture richer feature information. Finally, the two graph relation structures are fused together to obtain multiple spatial relations in the transit network. Through multigraph convolution and graph fusion, TSD-ST is able to better utilize the graph structure data to fully exploit the spatial information between bus stops. The process of multigraph convolution and graph fusion is shown in Figure 2. The input of convolution is $\mathbf{h}_t = \{\mathbf{h}_t^1, \mathbf{h}_t^2, \cdots, \mathbf{h}_t^N\}$, $\odot$ is Hadamard product, Θ denotes the parameters corresponding to the two graph convolutions, k is the number of layers of graph convolution, $\mathcal{N}(i)$ is the set of neighbor node indices of node i, μ is a custom parameter, and $\mathbf{g}_\varphi^{(k)}$ denotes the output of the k layer graph convolution. The specific process is as follows:
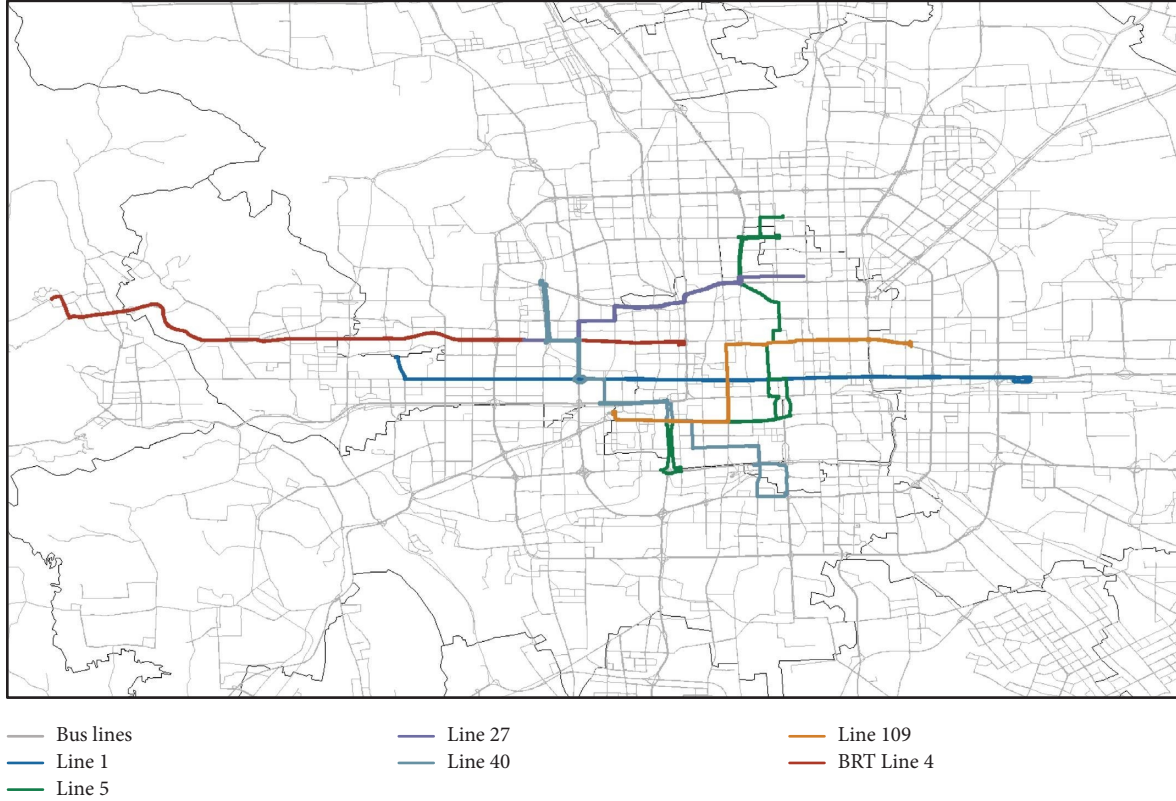
—— Bus lines	—— Line 27	—— Line 109
—— Line 1	—— Line 40	—— BRT Line 4
—— Line 5		

FIGURE 6: Selected experimental lines from the dataset.

$$\mathbf{g}_\varphi^{(1)} = \mu\mathbf{h} + (1 - \mu) \sum_{j \in \mathcal{N}_\varphi(i)} \mathbf{W}_\varphi(i, j) \odot \Theta_\varphi \mathbf{h},$$

$$\mathbf{g}_\varphi^{(2)} = \mu\mathbf{g}_\varphi^{(1)} + (1 - \mu) \sum_{j \in \mathcal{N}_\varphi(i)} \mathbf{W}_\varphi(i, j) \odot \Theta_\varphi \mathbf{g}_\varphi^{(1)},$$

$$\cdots \tag{18}$$

$$\mathbf{g}_\varphi^{(3)} = \mu\mathbf{g}_\varphi^{(k-1)} + (1 - \mu) \sum_{j \in \mathcal{N}_\varphi(i)} \mathbf{W}_\varphi(i, j) \odot \Theta_\varphi \mathbf{g}_\varphi^{(k-1)},$$

$$\mathbf{g}_\varphi = \mathbf{g}_\varphi^{(1)} + \mathbf{g}_\varphi^{(2)} + \cdots + \mathbf{g}_\varphi^{(k)}.$$

Both MLP embedding and MLP decoder use multilayer perceptron (MLP), which is capable of learning complex nonlinear relationships, and the specific process is as follows:

$$\mathbf{h}^{(k)} = \mathrm{ReLU}\left(\mathbf{W}_M^{(k)}\mathbf{h}^{(k-1)} + b^{(k)}\right). \tag{19}$$

4. Experiments and Analyses

4.1. Dataset and Settings. The TSD-ST model was validated using the April 2018 Beijing bus transit dataset, obtained in real time through the automated fare collection (AFC) system. This dataset includes various information, such as bus route numbers, passenger boarding times, boarding stop numbers, and alighting stop numbers. Prior to analysis, the dataset underwent preprocessing to eliminate erroneous data, including instances where passengers boarded and alighted at the same stop, empty data records, and cases

where alighting occurred before boarding. This preprocessing step was crucial to ensure data validity. Furthermore, the dynamic information, such as bus line numbers, boarding station numbers, and alighting station numbers, was matched with static line information, encompassing details such as line names and boarding and alighting station names, and the corresponding latitude and longitude coordinates. By combining these datasets, comprehensive bus passenger travel data are obtained. Finally, different running directions of buses on the same line are distinguished according to the order of boarding and alighting stations, and these traffic data are summarized into raw data every 15 minutes. The time series decomposition (TSD) method is utilized to decompose the data in advance. The training set encompasses data from the initial 24 days, while the test set comprises data from the final 6 days. The ratio of the training set to the test set is maintained at 8 : 2.

Six different types of lines were selected for the experiment as shown in Figure 6, and the basic information of these six experimental lines is described in Table 1. According to these lines, the passenger flow at each of their stations for three consecutive days and the Spearman rank correlation coefficients between the stations were statistically analyzed as shown in Figure 7, and it is evident that there is a more pronounced temporal dependence and spatial dependencies in bus transit passenger flow.

To evaluate the TSD-ST approach, all the experiments were conducted on a Windows workstation equipped with an NVIDIA Quadro RTX 4000 GPU, an Intel(R) Core(TM)

TABLE 1: Information on the experimental lines.

Lines	Type	Departure interval (min)	Average station distance: up* (km)	Average daily passenger flow in April: up	Average station distance: down** (km)	Average daily passenger flow in April: down
Line 1	Urban trunk line	5	1.00	25182	0.97	25833
Line 5	Business/tourist line	10	0.59	9708	0.65	9883
Line 27	Commuter line	10	0.73	7651	0.64	7930
Line 40	Commuter line	7	1.00	5968	1.00	6224
Line 109	Business/tourist line	10	0.46	6088	0.51	5600
BRT line 4	Suburban trunk line	10	1.34	21586	1.34	23997

*The upward direction of the bus and **the downward direction of the bus.
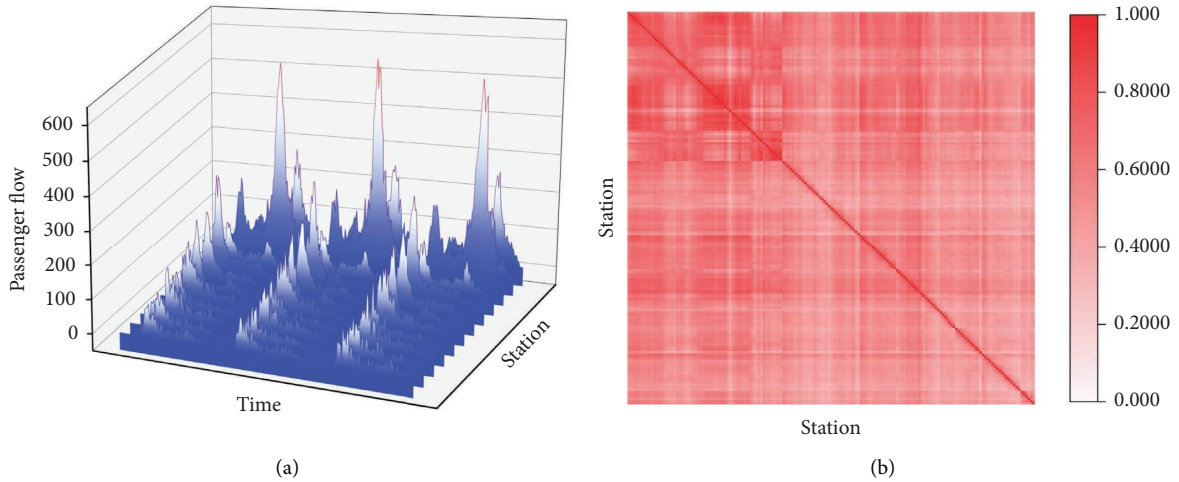

(a)

(b)

FIGURE 7: Spatiotemporal features of the experimental data. (a) Passenger flow for three consecutive days. (b) Spearman rank correlation coefficient between stations.

i9-10900K CPU, and 64 G RAM. According to several experiments, the parameters are set as follows: the number of epoch is 2000, the learning rate is 0.003, the model is optimized using the Adam optimizer, and the loss function uses the mean squared error (MSE), and in order to better train and measure the model effect, this paper chooses the root mean squared error (RMSE) and the mean absolute error (MAE) as the evaluation indexes. When calculating $E(t)$, set $\alpha = 0.01$ and $\beta = 50$. The Spearman matrix was constructed by setting the threshold $\varepsilon = 0.6$, and the multigraph convolution and graph fusion process was performed by taking $k = 2$, using a 2-layer graph convolution for the computation, and taking $\mu = 0.5$.

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (\widehat{y}_i - y_i)^2,$$

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |\widehat{y}_i - y_i|, \quad (20)$$

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (\widehat{y}_i - y_i)^2},$$

where y_i and $\widehat{y}_i$ are the true value and predicted value, and n is the number of all predicted values.

4.2. Experimental Performance. Prediction tasks with different time scales, including 30-min prediction, 60-min prediction, and 90-min prediction, were conducted to evaluate the performance of the TSD-ST model. Some commonly used and latest prediction models were selected for comparison:

(1) Historical Average (HA): a method of averaging historical observations as a prediction for the future

(2) Least Absolute Shrinkage and Selection Operator (LASSO): a linear regression method for feature selection and sparse modeling

(3) Fully Connected Neural Network (FCNN): a traditional artificial neural network (ANN) architecture

(4) Graph Convolutional Network (GCN) [52]: a deep learning model for processing graph data

(5) Long Short-Term Memory (LSTM) [53]: a variant of recurrent neural networks (RNN), and LSTM is widely adopted in the literature for time series prediction

TABLE 2: The list of evaluation metrics of compared models.

Metrics (×10−2)	30 min		60 min		90 min		Average	
Model	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
HA	12.52	7.61	13.64	8.38	14.65	9.12	13.60	8.37
LASSO	11.89	7.78	12.68	8.48	13.33	9.08	12.63	8.45
FCNN	11.86	7.63	12.65	8.35	13.30	8.95	12.60	8.31
GCN	11.84	7.62	12.60	8.33	13.27	8.95	12.57	8.30
LSTM	11.75	7.44	12.51	8.12	13.16	8.72	12.47	8.09
GRU	11.76	7.44	12.50	8.11	13.17	8.74	12.47	8.10
GGRU	11.85	7.58	12.60	8.26	13.24	8.86	12.56	8.23
T-GCN	11.67	7.37	12.22	7.94	12.66	8.36	12.18	7.89
TSD-ST	**10.13**	**5.90**	**10.34**	**6.07**	**10.58**	**6.37**	**10.35**	**6.11**

Note. Bolding denotes that the metrics perform best in all models.
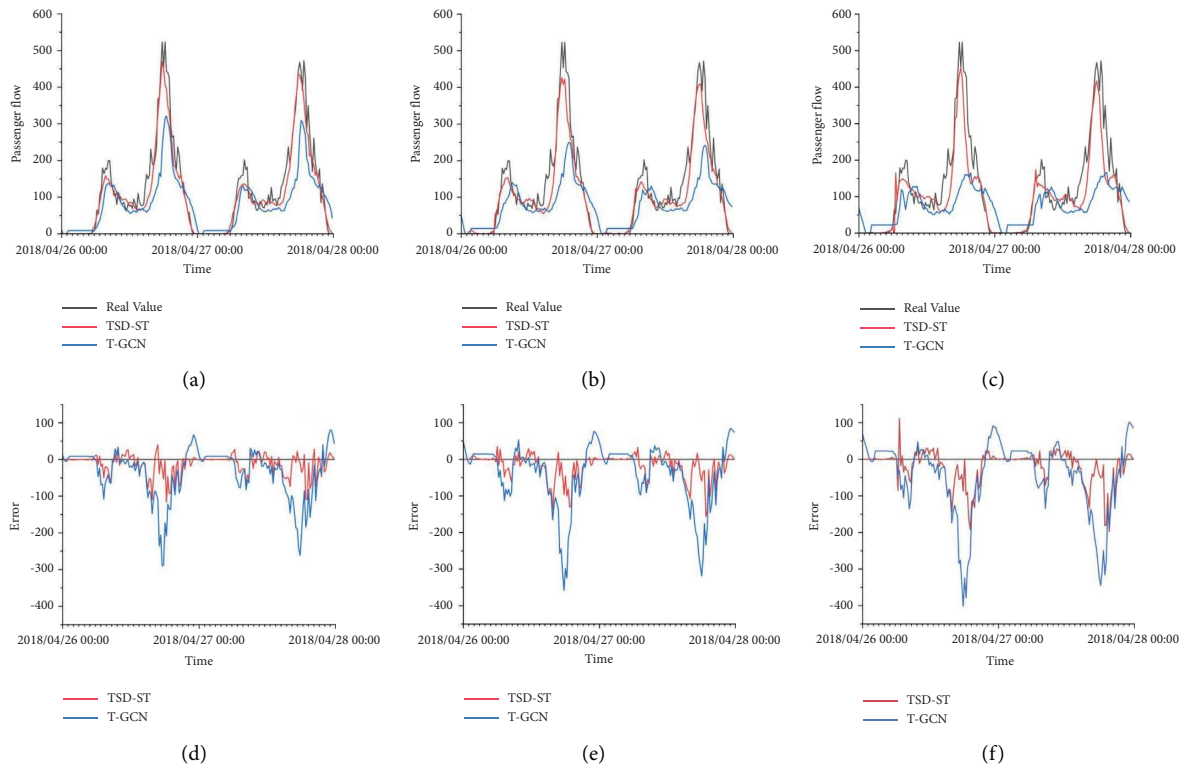

FIGURE 8: Prediction results and errors for TSD-ST and T-GCN from April 26, 2018, to April 27, 2018: (a) 30-min prediction results, (b) 60-min prediction results, (c) 90-min prediction results, (d) 30-min prediction errors, (e) 60-min prediction errors, and (f) 90-min prediction errors.

(6) Gated Recurrent Unit (GRU) [54]: a special recurrent neural network (RNN) model

(7) Graph Gated Recurrent Unit (GGRU) [47]: the linear operation in the gated recurrent neural network is replaced by the graph convolution operation, and GGRU uses attention mechanisms in graph convolution

(8) Temporal Graph Convolutional Network (T-GCN) [18]: a spatiotemporal graph convolution model for traffic prediction

In the experiments, the simplest structure of the proposed model was first established, and then, the structures of other comparative models were determined accordingly, ensuring that each model had the same number of learnable parameters and input information. In addition, for the model-agnostic parameters, we used the most common settings and deployed the same configurations fairly to each test, such as the learning rate, epoch number, and optimizer. The experimental results are shown in Table 2.

The experimental results show that traditional time series prediction methods such as HA are not suitable for nonlinear data with complex features such as bus passenger flow, and deep learning models such as GRU and GCN can learn the trend in the temporal and spatial dimensions to a certain extent, but due to the large stochastic of the bus passenger flow, learning only from a single temporal dimension or spatial dimension does not effectively capture the features. GGRU

TABLE 3: The list of evaluation metrics of ablation experiments.

Metrics (×10⁻²)	30 min		60 min		90 min		Average	
Ablation components	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
TSD-ST	**10.13**	**5.90**	**10.34**	**6.07**	**10.58**	**6.37**	**10.35**	**6.11**
–Adjacency graph	10.62 (↓0.49)	6.34 (↓0.44)	10.89 (↓0.55)	6.44 (↓0.37)	11.09 (↓0.51)	6.55 (↓0.18)	10.87 (↓0.52)	6.44 (↓0.33)
–Similarity graph	10.90 (↓0.77)	6.53 (↓0.63)	11.12 (↓0.78)	6.66 (↓0.59)	11.47 (↓0.89)	6.82 (↓0.45)	11.16 (↓0.81)	6.67 (↓0.56)
–Cycle	10.85 (↓0.72)	6.55 (↓0.65)	11.26 (↓0.92)	6.61 (↓0.54)	11.60 (↓1.02)	7.25 (↓0.88)	11.24 (↓0.89)	6.80 (↓0.69)
–Effect	10.16 (↓0.03)	6.04 (↓0.14)	10.36 (↓0.02)	6.22 (↓0.15)	10.73 (↓0.15)	6.57 (↓0.20)	10.42 (↓0.07)	6.28 (↓0.17)
–TSD	11.28 (↓1.15)	6.90 (↓1.00)	11.34 (↓1.00)	7.17 (↓1.10)	11.67 (↓1.09)	7.27 (↓0.90)	11.43 (↓1.08)	7.11 (↓1.00)
–AT & MLP decoder	10.23 (↓0.10)	6.14 (↓0.24)	10.45 (↓0.11)	6.30 (↓0.23)	10.61 (↓0.03)	6.63 (↓0.26)	10.43 (↓0.08)	6.36 (↓0.25)

Note. Bolding denotes that the metrics perform best in all models. "–" denotes "without" and "↓" denotes the worsening of metrics.

and T-GCN, as combined models, synthesize the temporal and spatial features, but they did not achieve satisfactory results in experiments. The TSD-ST model outperformed all other comparison models across all prediction tasks, exhibiting an average improvement of 18.08% in root mean square error (RMSE) and 25.65% in mean absolute error (MAE) on the three time scales. Specifically, the RMSE prediction metrics for 30-min, 60-min, and 90-min predictions were improved by 14.82%, 18.42%, and 20.73%, while the MAE was improved by 21.94%, 26.39%, and 28.00%. In comparison with the T-GCN model, the RMSE was improved by 15.02% and the MAE was improved by 22.56% on average in the prediction task for the three time scales. Specifically, the RMSE prediction metrics on 30-min prediction, 60-min prediction, and 90-min prediction were improved by 13.20%, 15.38%, and 16.43%, and the MAE was improved by 19.95%, 23.55%, and 23.80%.

As shown in Figure 8, the TSD-ST model excels in effectively handling the presence of unavoidable zero values within the bus transit dataset, demonstrating swift responsiveness to changes in transit passenger flow, and accurately capturing the peak periods. Conversely, as the prediction task's time scale expands, the performance of the T-GCN model notably diminishes, particularly when forecasting peak passenger flows. In contrast, the TSD-ST model consistently maintains its superior predictive capability.

In summary, real-world prediction tasks can demonstrate that TSD-ST is able to learn multiple spatial feature relationships between nodes of a bus transit network and capture the fluctuating trend of passenger flow changes in the time dimension and is more applicable to bus transit datasets. The model can excellently fulfill the task of multistop passenger flow prediction.

4.3. Ablation Experiments. To evaluate the effects of different components/modules in the TSD-ST, a series of ablation experiments were conducted to observe the change in overall performance by gradually removing certain design components/modules.

The results of these ablation experiments are presented in Table 3. It can be seen that deleting each component has an impact on the overall performance. In the 30-min prediction task, when the components of adjacency graph, similarity graph, cycle, effect, TSD, and AT & MLP decoder were reduced, RMSE values increased from 10.13 to 10.62, 10.90, 10.85, 10.16, 11.28, and 10.23, and MAE values increased from 5.90 to 6.34, 6.53, 6.55, 6.04, 6.90, and 6.14. In particular, when the similarity graph and time series decomposition (TSD) components were removed, there was a greater impact on the results. This observation further validates the conclusion of the previous analysis that bus passenger flow data have obvious temporal dependencies and spatial dependencies; the similarity graph captures the similarity between different stations to better utilize the spatial information, while the TSD model accurately predicts the temporal evolution trend of the bus passenger flow to make full use of the temporal dependencies.

5. Conclusions and Future Work

Accurate and effective passenger flow prediction can provide data support for bus transit operation planning and assist decision-making. In this study, a new deep learning framework (TSD-ST) is proposed for multistation short-term bus passenger flow prediction. The network can effectively incorporate temporal dependencies, spatial dependencies, and internal and external influences on bus passenger flow. Since bus passenger flow will show different trends between weekdays and nonweekdays, time series decomposition (TSD) can fully consider the temporal variation of passenger flow to understand the fluctuation and trend patterns of passenger flow in different periods, and it also takes into account the influence of random events. The similarity graph is introduced, the spatial similarity between stations is fully considered by the Spearman rank correlation coefficient, and multigraph convolution and graph fusion are performed to make up for the lack of traditional spatial learning limited to topological relations. Local and global features are modeled more accurately using GRU, temporal attention, and MLP.

Experimental results on real-world bus transit datasets show that the accuracy of both the TSD-ST proposed in this paper outperforms the current state-of-the-art models. The ablation experiments validate the effectiveness of the TSD module as well as the similarity graph in improving the overall accuracy, suggesting that considering the global spatial correlation of the network rather than the simple physical adjacency has greater advantages in the task of bus transit passenger flow prediction. In addition, the relatively low accuracies of GGRU and T-GCN demonstrate that bus transit passenger flow prediction is significantly different from other traffic prediction tasks, that bus transit passenger flow datasets have unique features, and that the TSD-ST model maintains the integrity of temporal and spatial dependencies to a large extent and is more applicable to bus transit passenger flow prediction.

Improvements can be made in future work. One issue is the expansion and refinement of the "effect," such as incorporating more temporary factors, such as weather conditions, temporary events, and large meetings, to enhance the ability of the prediction model. In addition, the TSD-ST model can be further improved by using more bus transit passenger flow data from other cities to validate and optimize the model.

Data Availability

The bus transit data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

References

[1] Y. Montero-Lamas, M. Novales, A. Orro, and G. Currie, "A new big data approach to understanding general traffic impacts on bus passenger delays," *Journal of Advanced Transportation*, vol. 2023, Article ID 4082587, 15 pages, 2023.

[2] X. C. Li, Y. Z. Peng, Z. X. Wu, and Z. W. Chen, "Short-term forecast of metro station passenger flow based on deep spatial-temporal network," *Traffic Transport*, vol. 33, pp. 55–61, 2020.

[3] H. Zhang, J. He, J. Bao, Q. Hong, and X. Shi, "A hybrid spatiotemporal deep learning model for short-term metro passenger flow prediction," *Journal of Advanced Transportation*, vol. 2020, Article ID 4656435, 12 pages, 2020.

[4] L. Liu and R.-C. Chen, "A novel passenger flow prediction model using deep learning methods," *Transportation Research Part C: Emerging Technologies*, vol. 84, pp. 74–91, 2017.

[5] J. Zhang, Y. Zheng, and D. Qi, "Deep spatio-temporal residual networks for citywide crowd flows prediction," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31, no. 1, 2017.

[6] W. Wu, Y. Xia, and W. Jin, "Predicting bus passenger flow and prioritizing influential factors using multi-source data: Scaled stacking gradient boosting decision trees," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 4, pp. 2510–2523, 2021.

[7] X. Yang, Y. Zou, and L. Chen, "Operation analysis of freeway mixed traffic flow based on catch-up coordination platoon," *Accident Analysis & Prevention*, vol. 175, Article ID 106780, 2022.

[8] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[9] S. Shekhar and B. M. Williams, "Adaptive seasonal time series models for forecasting short-term traffic flow," *Transportation Research Record*, vol. 2024, no. 1, pp. 116–125, 2007.

[10] L. Moreira-Matias, J. Gama, M. Ferreira, J. Mendes-Moreira, and L. Damas, "Predicting taxi– passenger demand using streaming data," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 3, pp. 1393–1402, 2013.

[11] R. Yu, Y. Li, C. Shahabi, U. Demiryurek, and Y. Liu, "Deep learning: a generic approach for extreme condition traffic forecasting," in *Proceedings of the 2017 SIAM international Conference on Data Mining*, pp. 777–785, SIAM, Philadelphia, PA, USA, June 2017.

[12] Z. G. Cui, R. Ke, Z. Pu, and Y. Wang, "Stacked bidirectional and unidirectional lstm recur- rent neural network for forecasting network-wide traffic state with missing values," *Transportation Research Part C: Emerging Technologies*, vol. 118, Article ID 102674, 2020.

[13] X. Zhang, S. Wen, L. Yan, J. Feng, and Y. Xia, "A hybrid-convolution spatial–temporal recurrent network for traffic flow prediction," *The Computer Journal*, p. 171, 2022, https://academic.oup.com/comjnl/advance-article-pdf/doi/10.1093/comjnl/bxac171/47465236/bxac171.pdf.

[14] 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.

[15] Z. Xie, W. Lv, S. Huang, Z. Lu, B. Du, and R. Huang, "Sequential graph neural network for urban road traffic speed prediction," *IEEE Access*, vol. 8, pp. 63349–63358, 2020.

[16] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A comprehensive survey on graph neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 1, pp. 4–24, 2021.

[17] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: data- driven traffic forecasting," in *Proceedings of the International Conference on Learning Representations*, Sacramento, CA, USA, July 2018.

[18] L. Zhao, Y. Song, C. Zhang et al., "T-GCN: a temporal graph convolutional network for traffic prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 9, pp. 3848–3858, 2020.

[19] G. Jin, Y. Cui, L. Zeng, H. Tang, Y. Feng, and J. Huang, "Urban ride-hailing demand prediction with multiple spatio- temporal information fusion network," *Transportation Research Part C: Emerging Technologies*, vol. 117, Article ID 102665, 2020.

[20] B. Lu, X. Gan, H. Jin, L. Fu, and H. Zhang, "Spatiotemporal adaptive gated graph con- volution network for urban traffic flow forecasting," in *Proceedings of the 29th ACM International conference on information & knowledge management*, pp. 1025–1034, New York, NY, USA, October 2020.

[21] L. Liu, J. Chen, H. Wu, J. Zhen, G. Li, and L. Lin, "Physical-virtual collaboration modeling for intra-and inter- station metro ridership prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 4, pp. 3377–3391, 2022.

[22] Z. Cui, K. Henrickson, R Ke, and Y. Wang, "Traffic graph convolutional recurrent neural network: a deep learning framework for network-scale traffic learning and forecasting," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 11, pp. 4883–4894, 2020.

[23] Z. Yang, Q. Zhao, and S. Zhao, "Passenger flow volume forecasting method based on public transit intelligent card (ic) survey data," *Transport Standardization*, vol. 9, pp. 115–119, 2009.

[24] Z. Chun-Hui and S.-Y. Song-Rui, "Kalman filter-based short-term passenger flow forecasting on bus stop," *Journal of transportation systems engineering and information technology*, vol. 11, no. 4, p. 154, 2011.

[25] Z. Ma, J. Xing, M. Mesbah, and L. Ferreira, "Predicting short-term bus passenger demand using a pattern hybrid approach," *Transportation Research Part C: Emerging Technologies*, vol. 39, pp. 148–163, 2014.

[26] Y. Ye, L. Chen, and F. Xue, "Passenger flow prediction in bus transportation system using arima models with big data," in *Proceedings of the 2019 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC)*, pp. 436–443, IEEE, Guilin, China, October 2019.

[27] S. Wu, Y. Zou, L. O. Wu, and Y. Zhang, "Application of bayesian model averaging for modeling time headway distribution," *Physica A: Statistical Mechanics and its Applications*, vol. 620, Article ID 128747, 2023.

[28] Q. Chen, W. Li, and J. Zhao, "The Use of Ls-SVM for Short-Term Passenger Flow Prediction/Mažiausių Kvadratų Atraminių Vektorių Metodo Taikymas Trumpalaikiam Keleivių Srautui Prognozuoti/Application of Support Vector Machines with Quadratic Loss Function for Forecasting Short-Term," *Transport*, vol. 26, no. 1, pp. 5–10, 2011.

[29] S. Bhattacharya, S. Phithakkitnukoon, P. Nurmi, A. Klami, and M. Veloso, "Gaussian process-based predictive modeling for bus ridership," in *Proceedings of the 2013 ACM conference*

on Pervasive and ubiquitous computing adjunct publication, pp. 1189–1198, New York, NY, USA, September 2013.

[30] P. Samaras, A. Fachantidis, G. Tsoumakas, and I. Vlahavas, "A prediction model of passenger demand using avl and apc data from a bus fleet," in *Proceedings of the 19th Panhellenic Conference on Informatics*, pp. 129–134, New York, NY, USA, October 2015.

[31] Y. Sun, B. Leng, and W. Guan, "A novel wavelet-svm short-time passenger flow prediction in Beijing subway system," *Neurocomputing*, vol. 166, pp. 109–121, 2015.

[32] L.-H. Li, J.-S. Zhu, X.-H. Shan, and X. Zhang, "Prediction modeling of railway short-term passenger flow based on random forest regression," in *Green Intelligent Transportation Systems: Proceedings of the 8th International Conference on Green Intelligent Transportation Systems and Safety*, pp. 867–875, Springer, Singapore, September 2019.

[33] Q. Li, Y. Qin, Z. Y. Wang, Z. X. Zhao, M. H. Zhan, and Y. Liu, "Prediction of urban rail transit sectional passenger flow based on elman neural network," *Applied Mechanics and Materials*, vol. 505-506, pp. 1023–1027, 2014.

[34] H. Li, Y. Wang, X. Xu, L. Qin, and H. Zhang, "Short-term passenger flow prediction under passenger flow control using a dynamic radial basis function network," *Applied Soft Computing*, vol. 83, Article ID 105620, 2019.

[35] T.-H. Tsai, C.-K. Lee, and C.-H. Wei, "Neural network based temporal feature models for short-term railway passenger demand forecasting," *Expert Systems with Applications*, vol. 36, no. 2, pp. 3728–3736, 2009.

[36] S.-Z. Zhao, T.-H. Ni, Y. Wang, and X.-T. Gao, "A new approach to the prediction of passenger flow in a transit system," *Computers & Mathematics with Applications*, vol. 61, no. 8, pp. 1968–1974, 2011.

[37] J. Teng and S. Chen, "Modified bus passenger flow forecasting model based on integrating arima with neural network," in *Proceesings of the CICTP 2015. Efficient, Safe and Green Multimodal Transportation. 15th COTA International Conference of Transportation Professionals*, pp. 1300–1310, Beijing, China, July 2015.

[38] Y. He, L. Li, X. Zhu, and K. L. Tsui, "Multi-graph convolutional-recurrent neural net- work (mgc-rnn) for short-term forecasting of transit passenger flow," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 10, pp. 18155–18174, 2022.

[39] S. Hao, D.-H. Lee, and D. Zhao, "Sequence to sequence learning with attention mechanism for short- term passenger flow prediction in large-scale metro system," *Transportation Research Part C: Emerging Technologies*, vol. 107, pp. 287–300, 2019.

[40] F. Toqué, E. Côme, M. Khalil El Mahrsi, and L. Oukhellou, "Forecasting dynamic public transport origin-destination matrices with long-short term memory recurrent neural networks," in *Proceedings of the 2016 IEEE 19th international conference on intelligent transportation systems (ITSC)*, pp. 1071–1076, IEEE, Rio de Janeiro, Brazil, November 2016.

[41] X. Yang, Q. Xue, M. Ding, J. Wu, and Z. Gao, "Short-term prediction of passenger volume for urban rail systems: a deep learning approach based on smart-card data," *International Journal of Production Economics*, vol. 231, Article ID 107920, 2021.

[42] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, and W.-K. Wong, "Convolutional lstm network: a machine learning approach for precipitation nowcasting," in *Proceedings of the 28th International Conference on Neural Information Processing Systems*, vol. 28, pp. 802–810, Cambridge, MA, USA, December 2015.

[43] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang, "Graph wavenet for deep spatial- temporal graph modeling," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence. International Joint Conferences on Artificial Intelligence Organization*, AAAI Press, Washington, DC, USA, August 2019.

[44] Y. Chen and X. M. Chen, "A novel reinforced dynamic graph convolutional network model with data imputation for network-wide traffic flow prediction," *Transportation Research Part C: Emerging Technologies*, vol. 143, Article ID 103820, 2022.

[45] P. Chen, X. Fu, and X. Wang, "A graph convolutional stacked bidirectional unidirectional-lstm neural network for metro ridership prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 6950–6962, 2022.

[46] J. Zhang, F. Chen, Z. Cui, Y. Guo, and Y. Zhu, "Deep learning architecture for short-term passenger flow forecasting in urban rail transit," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 11, pp. 7004–7014, 2021.

[47] Y. Zhang, Y. Chen, Z. Wang, and D. Xin, "Tmfo-aggru: a graph convolutional gated recurrent network for metro passenger flow forecasting," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–15, 2023.

[48] D. Luo, D. Zhao, Q. Ke et al., "Fine-grained service-level passenger flow prediction for bus transit systems based on multitask deep learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 11, pp. 7184–7199, 2021.

[49] S. J. Taylor and B. Letham, "Forecasting at scale," *The American Statistician*, vol. 72, no. 1, pp. 37–45, 2018.

[50] J. Li, H. Qu, and L. You, "An integrated approach for the near real-time parking occupancy prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 4, pp. 3769–3778, 2023.

[51] A. Vaswani, N. Shazeer, N. Parmar et al., "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[52] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, vol. 29, pp. 3844–3852, Red Hook, NY, USA, December 2016.

[53] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[54] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," in *Proceedings of the NIPS 2014 Workshop on Deep Learning*, vol. 42, Montreal, Canada, December 2014.