IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 12 June 2024, accepted 4 July 2024, date of publication 10 July 2024, date of current version 18 July 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3425634

RESEARCH ARTICLE

Short-Term Passenger Flow Prediction in Urban Rail Transit Based on Points of Interest

JIE CHENG[1], GUANGJIE LIU[1], SHEN GAO[2], AHMED RAZA[1], JIMING LI[3], AND WU JUAN[4]

[1]School of Electronic and Information Engineering, Nanjing University of Information Science and Technology, Nanjing 210044, China
[2]Nanjing Panda Information Industry Group Company Ltd., Nanjing 210000, China
[3]Nanjing Metro Operation Company Ltd., Nanjing 210028, China
[4]Nanjing Metro Construction Company Ltd., Nanjing, Jiangsu 210017, China

Corresponding author: Guangjie Liu (gjieliu@nuist.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant U21B2003 and in part by the Jiangsu Province Frontier Innovation Project under Grant BE2022075.

ABSTRACT In the rapidly evolving landscape of smart transportation, the passenger volume in urban rail transit consistently demonstrates an upward trajectory. In this context, precise and scientifically grounded short-term passenger flow prediction methods are essential for optimizing operational scheduling and ensuring safety in urban rail transit. Consequently, this paper introduces Temporal Graph Attention Long Short-Term Memory (TGALSTM), a spatiotemporal integrated prediction network model that incorporates the surrounding environment of the station. Initially, the paper enhanced the Temporal Convolutional Network (TCN) model to capture temporal features more accurately. Subsequently, the paper utilizes the Graph Attention Network (GAT) network module specifically to extract the topological structure and surrounding environmental features of the station. Lastly, the prediction task is accomplished by weighted fusion of various features, inputting them into the Attention Long Short-Term Memory (LSTM) network. Experiments were conducted on two authentic datasets, revealing that the TGALSTM model outperforms the baseline model in both single-step and double-step predictions, showcasing the model's exceptional performance and robustness. This research offers a robust method and support to enhance the operational efficiency and passenger flow management of urban rail transit systems.

INDEX TERMS Urban rail transit, short-term passenger flow prediction, surrounding environment, feature fusion.

I. INTRODUCTION

In recent years, with the continuous advancement of smart cities, urban rail transit has emerged as the cornerstone of modern urban transportation. Passenger flow prediction, as a critical component of intelligent transportation systems, holds paramount importance in enhancing transportation efficiency, ensuring station safety, and optimizing operational scheduling. Accurate short-term passenger flow prediction is paramount for adjusting operational strategies, alleviating carriage congestion, and enhancing passenger satisfaction.

The Autoregressive Integrated Moving Average (ARIMA) model is the most commonly used traditional model for time series passenger flow prediction. In recent years, scholars have proposed a series of improvements and optimizations for the ARIMA model. Milenković et al. [1] similarly employed the seasonal autoregressive integrated moving average (SARIMA) model to predict monthly railway passenger flows, taking into account the varying impact of seasons on passenger flows. Wang et al. [2] used wavelet decomposition to eliminate the impact of stochastic fluctuations in passenger flow and then employed the ARIMA model for prediction.

With the rise of deep learning technology, in recent years, there have also emerged passenger flow prediction methods based on deep learning, which can handle passenger flow prediction problems under multiple factors. Short-term passenger flow prediction methods based on deep learning primarily encompass Long Short-Term Memory (LSTM) networks [3], [4], Gated Recurrent Unit (GRU) [5], [6], Convolutional Neural Network (CNN) [7], [8], and Graph Convolutional Network (GCN) [9], [10], [11], [12]. In current research, LSTM and GRU networks are

The associate editor coordinating the review of this manuscript and approving it for publication was Yiqi Liu.

predominantly employed for capturing the temporal dependencies of passenger flow, whereas CNN and GCN networks are primarily utilized for extracting spatial features of stations. Zhang et al. [13] proposed a clustered LSTM model (CB-LSTM) based on the LSTM model. They employed a two-step k-means clustering method to cluster data from nine subway stations for training, thereby considering spatial characteristics locally. Chen et al. [14] proposed a deep learning network that integrates CNN and LSTM end-to-end, enabling simultaneous consideration of temporal and spatial features. Building upon this, Ke et al. [15] introduced a fusion convolutional long short-term memory network (FCL-Net), which integrates spatial and temporal variables into a deep learning architecture. While convolutional neural networks can extract spatial correlations, their effectiveness in extracting spatial characteristics from complex and irregular network topologies is limited, which is also a limitation of CNNs. Therefore, the Temporal Graph Convolutional Networks (T-GCN) model proposed by Zhao et al. [16] utilizes graph convolutional networks (GCN) and GRU to capture spatial and temporal dependencies in traffic data separately. Zhang et al. [17] proposed a fused deep learning architecture that employs GCN and Residual Network (ResNet) to capture spatiotemporal features. Wang et al. [18] proposed a multi-graph data approach, processing spatiotemporal features using multiple graph convolutional networks, and then extracting temporal features using LSTM. With further research, some scholars have begun considering the geographical information of stations.

In summary, researchers have predominantly restricted their consideration of spatial factors to station locations, with little discussion on the influence of the surrounding environment on passenger flow. However, variations in these environmental factors could induce fluctuations in passenger flow, thereby impacting prediction accuracy. To tackle this issue, the paper introduces a spatiotemporal prediction network model that integrates environmental factors surrounding stations to accurately predict short-term passenger flow.

A. RELATED WORKS

In the realm of traffic flow and passenger flow prediction research, numerous scholars have integrated deep learning models to address the shortcomings of traditional methods and enhance prediction accuracy. Hochreiter and Schmidhuber [19] introduced LSTM, resolving the vanishing gradient issue of RNN in long sequence learning, thereby enabling the capture of long-term dependencies despite its high computational complexity. Oord et al.'s [20] WaveNet model showcased the potential of Convolutional Neural Networks (CNN) in processing sequence data, effectively capturing long-term dependencies through causal convolution and dilated convolution. Lea et al. [21] pioneered the application of Time Convolutional Network (TCN) in action segmentation and detection, employing convolutional networks to process sequence data, capturing long-term dependencies, and enhancing processing speed through parallel computation. Yu et al.'s [22] proposal of Spatio-Temporal Graph Convolutional Network (ST-GCN) for traffic prediction amalgamates spatial-temporal convolutional networks to apprehend spatio-temporal dependencies, addressing spatial relationships via graph convolution and temporal relationships through time convolution. Velickovic et al.'s [23] introduction of Graph Attention Network (GAT) enhances the performance of graph neural networks through a self-attention mechanism. Bai et al. [24] conducted a comprehensive evaluation of TCN, demonstrating its superiority over traditional RNN and LSTM in various sequence modeling tasks, underpinned by robust parallel computing capabilities and swift sequence processing speed.

In pursuit of greater accuracy in passenger flow prediction, scholars are increasingly combining multiple models to account for a broader range of factors, catering to the evolving demands of prediction. The hybrid deep learning approach introduced by Wu et al. [25], integrating LSTM and TCN, leverages LSTM to capture long-term dependencies and TCN to boost computational efficiency, resulting in enhanced accuracy in traffic flow prediction. The AST-GCN, introduced by Guo et al. [26], integrates attention mechanisms into a spatiotemporal graph convolutional network, dynamically adjusting weights to enhance model performance and thus improving the accuracy of traffic flow prediction while accommodating complex spatiotemporal data. Zhao et al. [16] introduced the T-GCN, a spatiotemporal prediction model integrating GCN and GRU, leveraging graph convolution to capture spatial dependencies and GRU to capture temporal dependencies, demonstrating robust modeling capabilities in urban traffic flow prediction. Li et al. [27] showcased the robust spatiotemporal modeling capabilities of their hybrid model (GCN-LSTM), which integrates GCN and LSTM. The spatiotemporal synchronous graph convolutional network (STSGCN), introduced by Song et al. [28], enhances model performance and prediction accuracy by concurrently handling spatiotemporal data. Wu et al. [29] presented an approach for multi-variable time series prediction utilizing graph neural networks, which integrates graph neural networks and multi-variable time series, capturing relationships between variables through graph structures, thus handling intricate data relationships. The spatiotemporal graph convolutional network with multi-attention mechanisms, introduced by Hu and Chen [30], demonstrates outstanding performance in short-term traffic flow prediction and adeptly handles complex spatiotemporal data. The CNN-LSTM model for passenger flow prediction based on attention mechanisms, presented by Liu et al. [31], employs CNN for spatial feature extraction, assigns weights to features through attention mechanisms, and ultimately utilizes LSTM to extract time dependencies, thus achieving short-term passenger flow prediction. The TCN-LSTM hybrid prediction model introduced by Hou et al. [32] integrates external factors such as weather and air quality, enhancing overall prediction performance. The ResNet-GCN-AttLSTM prediction model proposed by Liu et al. [33]

employs residual networks and graph convolutional networks for spatial feature extraction, integrating attention-based long short-term memory networks for temporal feature extraction to achieve passenger flow prediction at each station. Zhang et al. [34] utilized Points of Interest (POI) information as graph data and employed CNNs to extract POI features. However, POI information is closely linked to stations, constituting irregular spatial topology, where CNNs may not efficiently extract relevant features.

Based on the aforementioned research background, this paper proposes a combined model that integrates time, space, and station geographic information. In terms of processing POI data, this paper initially computes the similarity of POI data for each station, followed by utilizing the Graph Attention Convolutional Networks (GAT) to comprehensively extract features from the POI information.

B. MAIN CONTRIBUTIONS

The main contributions of the paper include the following:

(1) Introduction of a comprehensive spatiotemporal framework that thoroughly POI information into the prediction model. This framework not only accounts for temporal and spatial variations but also integrates geographical location data. By holistically considering the influence of POIs, it enhances prediction accuracy and reliability.

(2) The proposed prediction model employs graph convolution and graph attention networks to capture the topological structure between stations and the surrounding environmental attributes. It utilizes an enhanced temporal convolutional network for extracting temporal features and ultimately constructs an LSTM model based on attention mechanisms for short-term passenger flow prediction.

(3) This paper evaluates and validates the performance of the model using two real-world datasets. The experimental results indicate that the model outperforms other baseline models, whether in single-step or double-step predictions, across private and public datasets. Furthermore, the model exhibits strong performance in predicting different types of stations.

The remainder of this paper is structured as follows. Section II introduction to the construction of station topology and POI data, laying the groundwork for subsequent modeling, followed by a detailed exposition of the TGALSTM model architecture. Section III introduces the configuration of private datasets and public datasets,as well as the setup of related experiments. Section IV provides an analysis of the experimental results. The main findings and limitations of the current study and their significance are summarized and directions for future research are proposed in Section V.

II. METHODOLOGY

The primary focus of this study is to predict short-term passenger flow for certain stations in the urban rail transit network based on historical passenger flow data, station topology, and POI data around the stations. This section highlights the structure of TGALSTM and how to use this model for short-term passenger flow prediction. The TGALSTM model comprises three main modules: GAT, Improved Temporal Convolutional Module (TCN), and Attention Mechanism Long Short-Term Memory Module (Attention LSTM). Before constructing the model, certain preparatory steps are necessary to ensure the project's smooth progress. The subsequent subsections delineate the preparatory work involved in the research, with the details of the TGALSTM model to be presented in subsequent chapters.

A. PREPARATION

Passenger flow dynamically changes over time, and historical passenger flow data for a time step is defined as $X = [X_{t_1}, X_{t_2}, \cdots X_{t_k}]$. Additionally, this paper involves two main types of graph structure matrices: one is the adjacency matrix representing the topological structure between stations in the rail network, and the other is the similarity matrix based on POI data.

For the construction of station topology relationships, let represent the i-th station out of n stations. The topological structure between stations is represented by the adjacency matrix A.

$$A = \begin{array}{c} \\ S_1 \\ S_2 \\ \vdots \\ S_n \end{array} \begin{array}{c} \begin{array}{cccc} S_1 & S_2 & \cdots & S_n \end{array} \\ \begin{bmatrix} a_{1,1} & a_{2,1} & \cdots & a_{n,1} \\ a_{1,2} & a_{2,2} & \cdots & a_{n,2} \\ \vdots & \vdots & \ddots & \vdots \\ a_{1,n} & a_{2,n} & \cdots & a_{n,n} \end{bmatrix} \end{array} \tag{1}$$

The elements in the matrix are:

$$a_{i,j} = \begin{cases} 1 & \text{Two adjacent stations} \\ 0 & \text{Otherwise} \end{cases} \tag{2}$$

For the construction of the similarity matrix for POI data, assuming there are m types of POIs, the POI information for each station can be represented by a feature vector p.

$$P = [poi_1 poi_2 \cdots poi_{\mathrm{m}}] \tag{3}$$

where poi_c represents the number of the c-th type of POI in the vicinity of the station. Therefore, this paper first calculates the similarity between POI information of stations using cosine similarity, as shown in (4) [35].

$$sim(i,j) = \frac{\sum_{k=1}^{m} P_{i,c} \times P_{\mathrm{j},c}}{\|P_i\| \times \|P_{\mathrm{j}}\|} \tag{4}$$

where m represents the dimension of the feature vector, and p_i represents the feature vector of a certain station. After processing the similarity matrix using cosine similarity, to enhance the capture of non-linear data and the model's expressive capability in prediction, this paper further processes the similarity matrix using a threshold Gaussian kernel function to obtain the final matrix SIM as input. The threshold

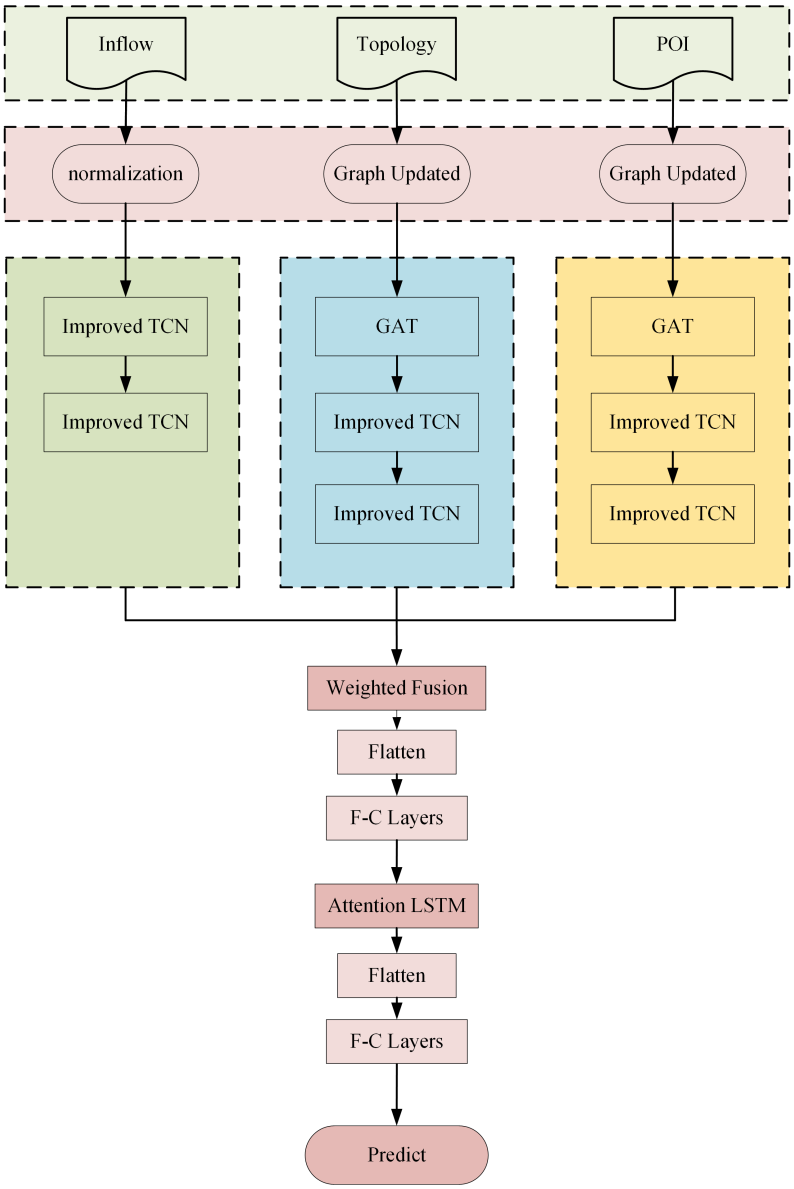


FIGURE 1. Model structure.

Gaussian kernel function is shown in (5) [36].

$$W_{i,j} = \begin{cases} \exp\left(-\dfrac{[\mathrm{sim}(i,j)]^2}{2\theta^2}\right) & \text{if } \mathrm{sim}(i,j) \le k \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

B. THE OVERVIEW OF MODEL FRAMEWORK

To better capture the non-linear characteristics of passenger flow, this paper proposes the TGALSTM passenger flow prediction model, as shown in Fig. 1. The model primarily consists of the improved TCN, GAT, and Attention LSTM. The improved TCN is utilized to extract local temporal correlations from passenger flow data, while Attention LSTM is employed to extract long-term temporal dependencies.

The model comprises three inputs: inbound passenger flow, network topology, and POI similarity matrix. Inbound passenger flow considers three different periodic patterns: weekly, daily, and neighboring. Therefore, the passenger flow prediction problem in this paper can be briefly summarized as follows: Given historical inbound passenger flow data $X = [X_{t_1}, X_{t_2}, \cdots X_{t_k}]$, network topology matrix A, and POI similarity matrix, the TGALSTM algorithm model predicts short-term passenger flow for the target station in future time steps.

C. IMPLEMENTATION STEPS OF THE TGALSTM MODEL FOR SUBWAY PASSENGER FLOW PREDICTION

The main content of the TGALSTM model proposed in this paper for subway passenger flow prediction is as follows, as shown in Fig. 2.

Step 1:Data preprocessing is the initial step in predictive modeling, aiming to ensure data quality and consistency.

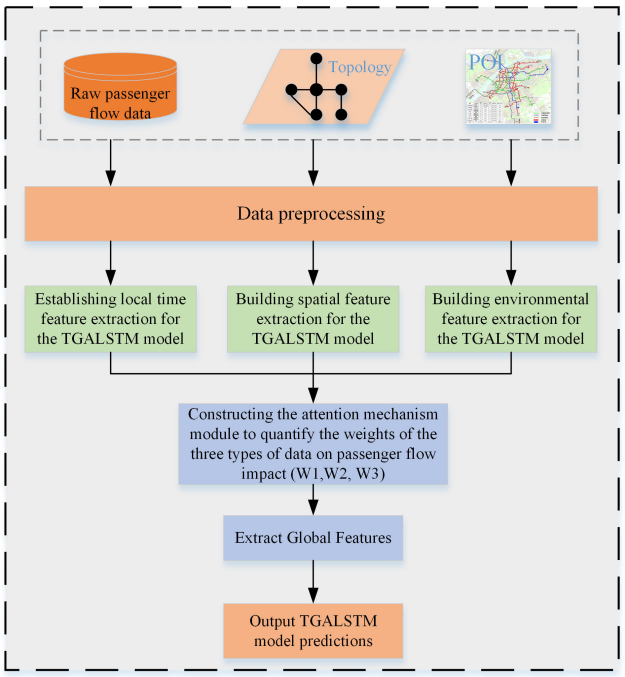


FIGURE 2. Steps to implement the TGALSTM model.

Initially, preprocessing involves cleaning and standardizing inbound passenger flow data to eliminate noise and outliers, ensuring data accuracy and usability. Next, processing the topological structure data of the rail transit network ensures the data accurately reflects real network connections. Lastly, processing POI data involves extracting useful features through aggregation and transformation. This step results in a cleaned and standardized dataset, readying it for subsequent feature extraction.

Step 2:Following data preprocessing, feature extraction becomes a critical step. This stage employs diverse neural network models to extract valuable features from various data types. Initially, enhanced Temporal Convolutional Networks (TCN) are employed to extract features from inbound passenger flow data. Subsequently, Graph Attention Networks (GAT) are utilized to extract features from the topological structure data, which are further processed with an enhanced TCN to derive richer features. Lastly, features are extracted from POI data using GAT and processed with an enhanced TCN to achieve the ultimate feature representation. These procedures facilitate the extraction of rich features from multi-source data, furnishing the model with ample information.

Step 3:Upon completion of feature extraction, it's essential to merge features from diverse sources to construct a unified feature representation. Initially, features extracted from various data sources are weightedly fused to ensure each feature carries an appropriate weight in the final representation. Subsequently, the merged features are flattened to convert them into one-dimensional vectors, facilitating subsequent neural network processing. Lastly, fully connected layers further process the flattened features to extract higher-level feature representations. This step yields a merged feature vector, serving as input for subsequent feature enhancement steps.

Step 4:Following feature fusion, enhancing the features further is vital to enhance the model's predictive capability. Initially, the fused features undergo processing using an Attention LSTM network to capture long dependencies and crucial information among the features. Subsequently, features processed by the Attention LSTM are flattened once more, converting them into one-dimensional vectors. Lastly, fully connected layers further process the flattened features to extract the ultimate high-level feature representation. This step yields an enhanced feature vector, serving as input for the final prediction step.

Step 5:Following the completion of all feature processing, the final prediction step is initiated. Through the preceding steps, a high-quality feature representation has been acquired. Employing these features as input, the prediction model is trained and inferred to produce the ultimate prediction results.

D. ENHANCEMENT OF TCN

Temporal Convolutional Network (TCN) is a network model designed specifically for time series prediction tasks. It mainly consists of causal convolution, dilated convolution, and residual connections [37]. The primary function of this module is to efficiently extract temporal features from input data and enhance the model's focus on significant features through attention mechanisms. As depicted in Fig. 3, the module comprises three layers: dilated causal convolution, channel attention mechanism, and a 1×1 convolutional layer. In comparison to traditional temporal models, this module, with the introduction of dilated causal convolutions, can extract features over a broader time range while bolstering feature expression through attention mechanisms. Specifically, dilated causal convolutions increase the spacing between convolutional kernels (dilation rate), effectively enlarging the receptive field. This allows for the extraction of more abundant temporal features without escalating computational complexity. This paper proposes three dilated causal convolution layers to capture features of three different cycles in passenger inflow simultaneously. Moreover, a local gate control mechanism with a Sigmoid function is introduced after the second dilated causal convolution layer, enabling the network to dynamically adjust attention to input information. Following the extraction of relevant features in the convolutional layers, the ReLU activation function is applied to introduce non-linear features. Subsequently, the channel attention mechanism is employed to amplify the model's focus on significant features. Additionally, we retain the residual structure of the traditional TCN to improve the computational efficiency of the model. To enhance the extraction of temporal features from passenger flow data, this paper utilizes a two-layer improved TCN network, where the

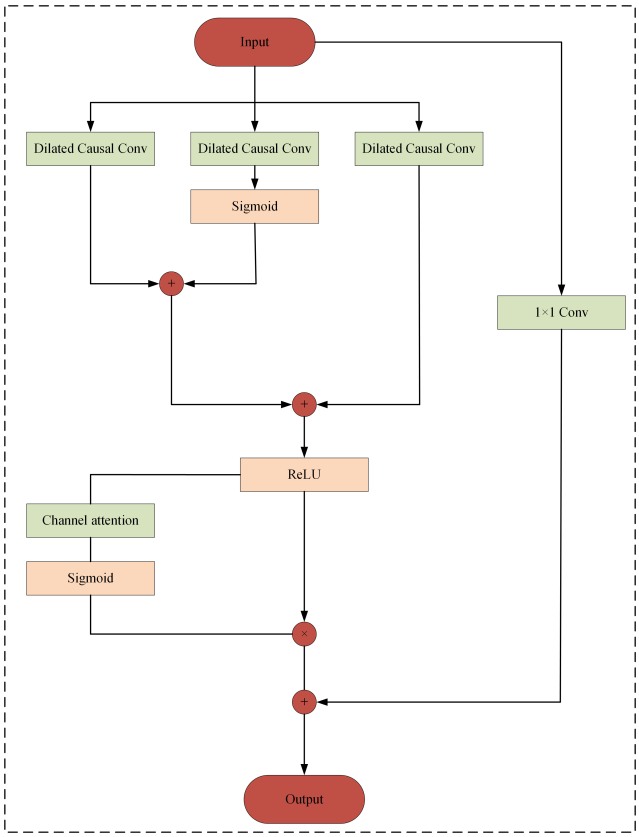


FIGURE 3. Enhanced TCN.

first layer has 32 output channels and the second layer has 64 output channels.

E. CHANNEL ATTENTION MECHANISM

The attention mechanism mimics the resource allocation of human attention. At any given moment, the human brain concentrates attention on areas requiring focus, reducing or even ignoring attention to other areas, thereby acquiring more detailed information pertinent to focus, suppressing irrelevant data, and amplifying relevant information [38].

The channel attention module, illustrated in Fig. 4, in this paper, AdaptiveAvgPool2d and AdaptiveMaxPool2d are employed for global average pooling and global max pooling on the input, respectively, to obtain the global average features and global max features for each channel. Subsequently, two fully connected layers are added. The first fully connected layer reduces the number of channels proportionally. We set the reduction_ratio to 4 to decrease computational load. The second fully connected layer restores the number of channels to the original number. ReLU activation functions are applied in between to introduce non-linearity. Such configuration allows for better attention to the importance of each channel.

F. SPATIAL FEATURE MODULE

Conventional convolutional modules, like CNN modules, represent the transportation network as a grid matrix, which

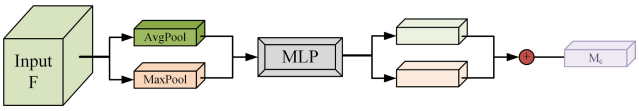


FIGURE 4. Channel Attention Module.

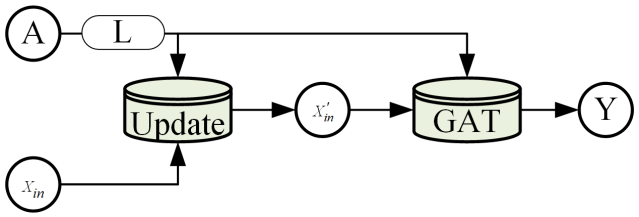


FIGURE 5. Spatial Feature Extraction.

fails to accurately capture the influence of the topological structure between stations on prediction accuracy [26]. Hence, this paper adopts GAT to specifically extract spatial features of the transportation network. Prior to entering the GAT network, the paper initially employs the Laplacian operator to update the graph signal [39], as outlined in (6) - (7).

$$\hat{A} = A + I \tag{6}$$

$$\hat{D}_{ii} = \sum_j A_{ij} \tag{7}$$

where I is the identity matrix with the same dimension as A; D is the degree matrix of A; X_{in} is the input flow matrix; X'_{in} is the updated flow matrix after graph signal updating.

Subsequently, X'_{in} is fed into the GAT network module as input, and the GAT calculates the weight of each node. Finally, the node information is updated again based on these weights to generate the output result Y. The entire process is illustrated in Fig. 5.

The role of the GAT network module is to extract spatial features of the topological structure and the environmental features surrounding the stations. The module comprises multiple Graph Attention Layers, each containing 4 Multi-Head Attention Mechanisms, to more effectively capture the intricate relationships between station nodes.

The selected environmental features in this paper include functional features such as restaurants, hospitals, and residential areas near the stations. Given the varying impact of different features on passenger flow, the attention mechanism in GAT is utilized to determine the weights of each feature. Specifically, the attention coefficients e_{ij} are calculated for each station node i and its adjacent station node j.

$$e_{ij} = Leaky\,\mathrm{Re}\,LU(a^T[Wh_i||Wh_j]) \tag{8}$$

where, a^T represents a learnable attention vector, W stands for a linear transformation matrix, h_i and h_j denote feature vectors of nodes i and j respectively, and || denotes the vector concatenation operation.

Later, the softmax function is applied to normalize the attention coefficients to determine the weight of each

station. Subsequently, the normalized attention coefficients are weighted and summed to update the feature representation of station node i. Through this method, the GAT network can dynamically adjust the weights based on different functional features in the POI information, thus extracting the station's features more accurately.

G. WEIGHTED FEATURE FUSION

Considering the variation in the impact of the three input parts of the model on passenger flow, and given that the output data dimensions of the three branches are identical, this paper employs weighted feature fusion to perform comprehensive prediction [40]. Specifically, the calculation for weighted fusion is depicted in (9).

$$f_{Fusion} = W_1 \cdot f_1 + W_2 \cdot f_2 + W_3 \cdot f_3 \tag{9}$$

where f_1, f_2 and f_3 represent the output features of three distinct branches, with corresponding weights of W_1, W_2 and W_3 learned as parameters by the network.

H. ATTENTION LSTM

LSTM networks have evolved from recurrent neural networks (RNNs) to primarily address the issues of gradient vanishing and explosion that arise when traditional RNNs process long sequential data [41]. To further extract longer temporal information, we utilize an LSTM module with 128 hidden layers to capture the temporal dependencies of 78 stations. In this study, to mitigate the loss of information in long time sequences by LSTM, self-attention mechanism is incorporated. The paper calculates the attention weights of the LSTM output sequence through linear transformation [42]. These weights are utilized to weight the input, thereby emphasizing or suppressing information from different time steps.

$$h'_t = a_t \cdot h_t \tag{10}$$

where a_t represents the attention weight at the current time step; h'_t represents the final output.

III. EXPERIMENT

A. DATA DESCRIPTION

This study employs two datasets. The first dataset[1] is proprietary, obtained from the automatic ticket inspection system of Nanjing Metro Company in China. To mitigate the influence of the sharp increase in passenger flow before New Year's Day and Dragon Boat Festival, it covers passenger flow data from January 4th to January 31st, April 1st to April 24th, and May 1st to May 31st, 2018. The study focuses on the entry passenger flow of 78 stations across Nanjing Metro Lines 1, 2 and 3. It utilizes data collected only from 6:00 to 23:00 on 60 working days within the target period, amounting to 110 million records. Each record comprises entry time, transaction occurrence time, entry station number, transaction route, and transaction station. The specifics are shown in Table 1. In this study, we chose a time interval of 10 minutes,

[1]https://github.com/CJ5202/OD_DATA

TABLE 1. Optimal hyperparameter combinations.

Field Name	Description
Entry Time	Timestamp identifying entry
Transaction Time	Timestamp identifying exit
Transaction Type	Entry or exit
Transaction Device ID	Exit gate device ID
Entry Station ID	Name of the entry station
Transaction Line	Identifying the subway line
Transaction Station	Name of the exit station

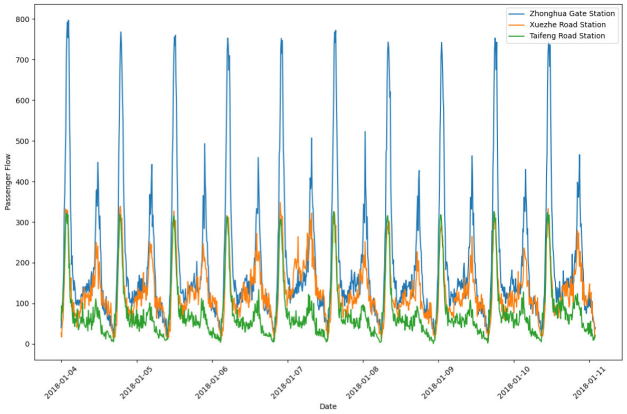


FIGURE 6. The raw passenger flow data.

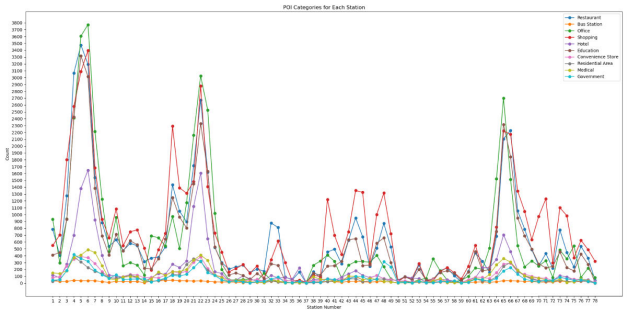


FIGURE 7. POI data.

resulting in a total of 102 time slots per day. To better represent the real data, we randomly selected three stations and conducted visual analysis of their inbound passenger flow every 10 minutes from January 4th to January 11th, 2018. As depicted in Fig. 6, there is significant fluctuation in passenger flow at the three stations, with noticeable differences between them. Therefore, this study adopts a sliding window approach to predict the passenger flow of the next time slot based on the passenger flow of the previous 5 time slots.

Concerning the POI data, this study solely focuses on the time period corresponding to the private dataset, gathering POI-related data within a 1-kilometer radius around the stations from Gaode Maps. The POI data comprises the quantity of 10 categories near the stations, including restaurants, bus stops, companies, shopping venues, accommodation hotels, educational institutions, convenience stores, residential areas, healthcare services, and government facilities. Eventually, processed as inputs to the model using equations 4 and 5

TABLE 2. Dataset comparison.

Attribute	Private Dataset	Public Dataset
Number of Records	110 million	130 million
Review Days	60 days	25 days
Time Period	2018.1.4–5.31	2016.2.29–4.3
TG	10min	10min
Days of the Week	Monday to Friday	Monday to Friday
Number of Stations	78	276

in Section III-A. As illustrated in Fig. 7, the majority of stations have relatively low counts of Points of Interest (POIs), yet a minority of stations demonstrate notable peaks across multiple categories. These stations serve as pivotal transportation hubs or commercial centers within the city, such as Station 6, known as Xinjiekou Station (a commercial hub).

The second dataset is a publicly available one [17], comprising AFC data from Beijing subway, spanning from February 29th, 2016 to April 3rd, 2016. It covers 25 working days between 05:00 and 23:00 over five consecutive weeks. During the forecasting period, the analysis in this study is conducted at 10-minute intervals. To facilitate a comprehensive comparison of the differences and similarities between the two datasets, Table 2 illustrates the distinctions and similarities in terms of time range, time interval, record count, and station count.

B. LOSS FUNCTION AND EVALUATION METRICS

Mean Squared Error (MSE) and Mean Absolute Error (MAE) are the most commonly utilized regression loss functions. MSE is more sensitive to outliers in the training dataset as it assigns greater weight to large errors, whereas MAE is relatively robust to outliers. However, a drawback of using MAE as the loss function for neural network training is its constant gradient, which may lead to being trapped in local minima during training with gradient descent. Huber loss is a loss function that lies between MSE and MAE. It behaves similarly to MSE when the predicted values are close to the true values and similar to MAE when the predicted values are far from the true values [40]. Its mathematical definition is:

$$\text{Loss} = \text{Huber} = \begin{cases} \frac{1}{2}(y-\hat{y})^2 & |y-\hat{y}| \le \delta \\ \delta|y-\hat{y}| - \frac{1}{2}\delta^2 & \text{otherwise} \end{cases} \tag{11}$$

where δ is a hyperparameter of the loss function. When $\delta \geq 0$, the Huber loss approaches MSE, and when $\delta \geq \infty$, it approximates MSE.

Furthermore, this study employs Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Weighted Mean Absolute Percentage Error (WMAPE) as evaluation metrics for prediction.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{12}$$

TABLE 3. Optimal hyperparameter combinations.

Hyperparameters	Determined Values
Batch Size	128
Dropout Rate	0.1
Dilation Rate	1
Hidden Layers	128

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|(y_i - \hat{y}_i)| \tag{13}$$

$$WMAPE = \sum_{i=1}^{n}\left(\frac{y_i}{\sum_{j=1}^{n} y_j}\left|\frac{y_i - \hat{y}_i}{y_i}\right|\right) \tag{14}$$

where y_i represents the true value, $\hat{y}_i$ denotes the predicted value, and n stands for the total number of predicted values.

C. MODEL HYPERPARAMETER SELECTION

The experiments were conducted on a graphics processing unit (GPU) platform with a GeForce RTX 2080Ti model, featuring 10GB of memory (powered by an Intel (R) Xeon (R) CPU E5-2630 v4 2.20GHZ). The model was developed using Python 3.7.10 and PyTorch 1.8. The data was partitioned into training and testing sets at a ratio of 9:1, with a validation split set to 0.1 for tuning model hyperparameters to enhance model performance. To assess the model's performance at different time steps, this study employed the first five time steps to predict the next time step and the following two time steps. During the experiments, Adam optimizer with a learning rate of 0.0003 was employed. The batch size was chosen from [32, 64, 128]. Dropout rates in the GAT network were selected from [0.1, 0.2, 0.3, 0.4], dilation rates in the improved TCN network were chosen from [1, 2, 3], and the number of hidden layers in the LSTM network ranged from [64, 128, 256]. Finally, the optimal hyperparameter combination was determined through multiple experiments, and the best hyperparameter combination is shown in Table 3. To prevent overfitting during training, early stopping technique was employed.

D. BASELINE MODELS

In this experiment, to showcase the model's performance, comparisons will be made with the following baseline model methods.

(1) Temporal Convolutional Networks (TCN) are specifically tailored for sequence modeling tasks, particularly in the context of time-series data. TCN employs one-dimensional

convolutions along the temporal axis to capture patterns and dependencies in sequence data. The dilation rate is set to start at 2, dropout rate at 0.1, kernel size at 3, batch size at 128, learning rate at 0.0003. The number of channels for entry flow is set to [3, 32, 64], and for other inputs, the number of channels is set to [1, 32, 64].

(2) The TCN-LSTM (TCN Convolutional Long Short-Term Memory) composite model integrates their distinct advantages in sequence modeling. TCN is employed to capture short-term local patterns, while LSTM is utilized to capture long-term dependencies. Such combination can improve the model's performance. The number of hidden layers is set to 128, dilation rate to 1, kernel size to 3, batch size to 128, and learning rate to 0.0003.

(3) Conv-LSTM (Convolutional LSTM Network) is a neural network architecture that combines the features of Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) networks, allowing it to handle spatiotemporal data relationships. The number of hidden layers in LSTM is set to 128, kernel size to 3, batch size to 128, and learning rate to 0.0003.

(4) Res-LSTM (Residual LSTM) [17] is a combination model that integrates residual blocks and Long Short-Term Memory (LSTM) networks. The number of filters for the first residual block is set to 32, and for the second residual block is set to 64. The LSTM hidden layer size is 128, batch size is 128, kernel size is 2, and learning rate is 0.001.

(5) SCINet (Sample Convolution and Interaction Network) [43] utilizes multiple fully connected layers to decompose time series for prediction. The convolutional kernel size is set to 3, dropout to 0.2, and the learning rate to 0.0003.

(6) TADSTN (Topology Augmented Dynamic Spatial-Temporal Network) [44] is a model that fully explores spatial-temporal relationships. The convolution kernel is set to 3×5, and the learning rate is $1e^{-3}$.

(7) The TGALSTM-T variant model excludes the input of topological structure and POI branch, while keeping the hyperparameter settings consistent with the TGALSTM model.

(8) The TGALSTM-No POI variant model excludes the input from the POI branch, while keeping the hyperparameters consistent with the TGALSTM model.

(9) The TGALSTM-No TOP variant model excludes consideration of the topology structure, while keeping the hyperparameters consistent with the TGALSTM model.

In the experiment, apart from the Res-LSTM model where data preprocessing adopts Min-Max normalization, data for other baseline models are preprocessed using Z-Score standardization.

IV. PREDICTION RESULTS AND ANALYSIS

A. COMPARATIVE ANALYSIS OF INTEGRATED PREDICTIONS

On the private dataset, the TGALSTM model exhibits the best predictive performance.since the Res-LSTM and TADSTN models do not consider POI information, to ensure fairness in the experiment, the TGALSTM-No POI variant model is used to verify the effectiveness of the model. Similarly, as the SCINet model does not take into account topological structure and POI information, this paper employs the TGALSTM-T variant model for comparison. From Table 4, it is evident that the TGALSTM-No POI variant model performs better than the Res-LSTM and TADSTN models on all evaluation metrics; the evaluation metrics for the TGALSTM-T variant model are also superior to those of the SCINet model. Furthermore, the performance of the TGALSTM-No TOP variant model is only slightly lower than that of the TGALSTM model but surpasses the performance of the TGALSTM-No POI variant model. This suggests that the inclusion of POI information, as considered in this study, can enhance prediction performance and is more critical than the topological structure of the stations. The TGALSTM-T variant model slightly surpasses both the TGALSTM-No POI and TGALSTM-No TOP variant models across all metrics, yet underperforms compared to the TGALSTM model. This suggests that integrating topological structure with POI information can enhance prediction accuracy more effectively. Among the baseline models, the TCN model performs the worst due to its inability to extract spatial features and its limitation to capturing only local patterns and features in the input sequence. Conversely, the TCN-LSTM model significantly improves prediction accuracy by combining the local features of TCN with the long-term dependencies of LSTM. The Conv-LSTM model's performance also exceeds that of TCN because Conv-LSTM can capture more spatial features.

In terms of predictive performance on the public dataset, when no POI information is considered, the TGALSTM model also demonstrates the best performance. The TADSTN model's performance is slightly below that of the TGALSTM model. Additionally, the performance gap between Conv-LSTM and Res-LSTM is relatively small, with the Res-LSTM model slightly outperforming Conv-LSTM in spatiotemporal sequence modeling. Furthermore, the Res-LSTM model outperforms the SCINet model, likely because the SCINet model does not adequately account for spatial dependencies during its predictions. Finally, the performance of the TCN-LSTM model remains superior to that of the TCN model.

In summary, Table 4 and Table 5 present the performance of the TGALSTM model and other baseline methods in short-term passenger flow prediction on both private and public datasets. It is evident that the TGALSTM model achieves the best predictive performance on both datasets, validating the effectiveness and robustness of the model. Additionally, when comparing the performance of TCN-LSTM and Conv-LSTM on both datasets, the TCN-LSTM model outperforms Conv-LSTM on the private dataset, while the reverse is observed on the public dataset. The decrease in sample size may be the reason for this discrepancy.

TABLE 4. Private dataset.

Model	Temporal	Topology	POI	RMSE	MAE	WMAPE(%)
TCN	✓	✓	✓	38.613	22.828	12.31
TCN-LSTM	✓	✓	✓	29.436	17.999	9.71
Conv-LSTM	✓	✓	✓	30.737	18.330	9.88
Res-LSTM	✓	✓	×	30.187	18.612	10.03
SCINet	✓	×	×	31.288	18.351	9.89
TADSTN	✓	✓	×	30.633	18.516	9.98
TGALSTM-T	✓	×	×	29.169	17.589	9.48
TGALSTM-No POI	✓	✓	×	29.292	17.798	9.59
TGALSTM-No TOP	✓	×	✓	29.048	17.672	9.53
TGALSTM	✓	✓	✓	28.823	17.550	9.46

TABLE 5. Public dataset.

Model	Temporal	Topology	POI	RMSE	MAE	WMAPE(%)
TCN	✓	✓	×	50.415	32.119	18.17
TCN-LSTM	✓	✓	×	29.117	18.059	10.12
Conv-LSTM	✓	✓	×	28.794	17.433	9.81
Res-LSTM	✓	✓	×	28.366	16.631	9.35
SCINet	✓	×	×	29.147	16.972	9.59
TADSTN	✓	✓	×	27.851	16.977	9.61
TGALSTM	✓	✓	×	27.119	16.105	9.03

TABLE 6. Private dataset.

Model	Single-step			Two-step		
	RMSE	MAE	WMAPE(%)	RMSE	MAE	WMAPE(%)
TCN	38.613	22.828	12.31	38.899	31.055	14.42
TCN-LSTM	29.436	17.999	9.71	30.272	20.687	11.53
Conv-LSTM	30.737	18.330	9.88	30.868	20.902	11.56
Res-LSTM	30.187	18.612	10.03	33.212	23.160	12.40
SCINet	31.288	18.351	9.89	28.233	19.258	11.35
TADSTN	30.633	18.516	9.98	31.638	22.822	11.62
TGALSTM-T	29.169	17.589	9.48	28.374	20.353	12.12
TGALSTM-No POI	29.292	17.798	9.59	26.193	18.861	11.01
TGALSTM-No TOP	29.048	17.672	9.53	29.595	20.948	11.36
TGALSTM	28.823	17.550	9.46	27.559	19.446	10.72

TABLE 7. Public dataset.

Model	Single-step			Two-step		
	RMSE	MAE	WMAPE(%)	RMSE	MAE	WMAPE(%)
TCN	50.415	32.119	18.17	85.495	67.310	17.28
TCN-LSTM	29.117	18.059	10.12	47.230	29.870	12.21
Conv-LSTM	28.794	18.330	9.81	48.468	29.441	11.44
Res-LSTM	28.366	16.631	9.35	41.986	26.805	11.65
SCINet	29.147	16.972	9.59	39.291	24.526	9.65
TADSTN	27.851	16.977	9.61	39.565	25.241	9.86
TGALSTM	27.119	16.105	9.03	37.580	24.116	9.80

B. ANALYSIS OF PREDICTION AT DIFFERENT TIME STEPS

To comprehensively assess the models' performance in various prediction tasks and gain a better understanding of their applicability and robustness, this study conducts multi-step forecasting on two datasets. The primary evaluation strategy involves dual-step prediction, which enables a thorough examination of the models' predictive capability for multiple future time steps. Taking MAE and WMAPE metrics as examples, as shown in Table 6 and Table 7, it is evident that the predictive accuracy of each model declines to varying degrees in multi-step prediction tasks. Particularly, there is a substantial drop in performance

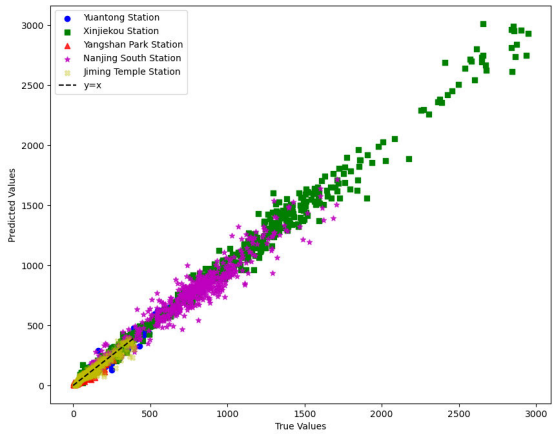


FIGURE 8. Prediction Results for Different Stations.

when conducting multi-step prediction on the public dataset, likely due to the absence of POI information in the public dataset. Considering the RMSE metric, overall, the predictive performance of each model decreases in dual-step prediction tasks. However, on the private dataset,The RMSE metric of the TGALSTM-T variant model improved from 29.169 to 28.374, the TGALSTM model decreases from 28.842 to 26.202, and that of the TGALSTM-No POI variant model decreases from 29.292 to 26.193. This suggests that the model is more sensitive to single-step anomalous changes but more robust to anomalous changes in dual-step time series.

C. VISUALIZATION AND ANALYSIS OF PREDICTION RESULTS FOR DIFFERENT TYPES OF STATIONS

To comprehensively assess the models' performance, this study evaluates the single-step prediction scenario using five types of stations in the private dataset. Specifically, the analysis focuses on stations surrounding commercial areas, hub areas, scenic spots, campuses, and office areas, predicting and analyzing their station's time steps for five days to better understand the models' performance in different scenarios.

As shown in Fig. 8, this paper visualizes the prediction results for five stations. Overall, the predicted values closely match the actual passenger flow values for most time periods at each station, indicating the model's robust predictive performance across different types of stations. Based on POI data, it is evident that Yuantong Station is surrounded by numerous companies and restaurants, indicating its classification as an office area type station. The station notably experiences morning and evening peaks, with a peak value of approximately 500 passengers. Comparing the actual and predicted passenger flows over five days, the model proposed in this paper demonstrates exceptional predictive performance.

Xinjiekou Station is situated in the heart of the commercial district, surrounded by numerous shopping malls and entertainment facilities, with a peak passenger flow reaching up to 3000 people per time step. Nanjing South Station, serving as the city's hub station, exhibits a continuous growth trend in

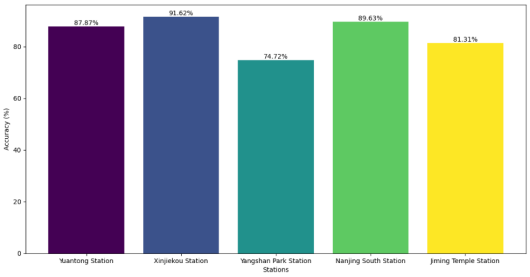


FIGURE 9. Accuracy for different stations.

passenger flow without a distinct morning or evening peak. Its daily peak passenger flow is approximately 1500 people. Both Nanjing South Station and Xinjiekou Station are typical bustling stations, experiencing high passenger flows daily. The model demonstrates high accuracy in predicting large passenger flows based on the forecast results.

Yangshan Park Station is situated in Xianlin University City, surrounded by numerous universities, thus classified as a campus-type station. Generally, the station experiences relatively low passenger flow volume, displaying distinct morning and evening peak features, but with significant fluctuations during non-busy periods. Jiming Temple Station is surrounded by two renowned scenic spots, Xuanwu Lake and Jiming Temple, hence categorized as a scenic area-type station. The overall inbound passenger flow at this station does not exhibit particularly significant regularity, with frequent instances of sudden passenger flow. The model demonstrates its capability to effectively capture changes in passenger flow, indicating its adaptability to the passenger flow characteristics of both campus and scenic area stations.

To further validate the effectiveness of the model, Fig. 9 demonstrates the prediction accuracy of the model at five stations of different types. The results indicate that the model achieves prediction accuracies close to 90% for high passenger flow at Xinjiekou Station and Nanjing South Station. Moreover, the prediction accuracy at Yuantong Station and Jiming Temple Station also exceeds 80%. Lastly, the prediction accuracy at Yangshan Park Station surpasses 70%. These results suggest that the model performs optimally at commercial and hub area stations, while also exhibiting good predictive performance at other station types.

V. CONCLUSION

This paper introduces a spatio-temporal integrated prediction network model (TGALSTM) that incorporates station attributes (POI information). The basic idea is to utilize historical passenger flow data, incorporating station topology and surrounding environmental information into the model input. Through the GAT network module, spatial features are fully explored, and improved TCN is used to extract local temporal features. Finally, weighted feature fusion is applied to the attention mechanism of the LSTM network to complete the passenger flow prediction task. The model is validated on two datasets. The results indicate

that compared to baseline models, whether for single-step or multi-step prediction, the TGALSTM model exhibits higher accuracy. Additionally, through single-step prediction evaluations of five different types of stations, the TGALSTM model demonstrates outstanding performance in commercial areas, hub areas, scenic spots, campuses, and office areas. By comprehensively utilizing POI data, the model accurately captures the passenger flow patterns of various types of stations, particularly excelling in predicting peak periods and sudden surges in passenger flow, reflecting the model's robustness. However, the proposed model has not yet been applied to important holiday scenarios, and future research will further explore predictions during such periods.

REFERENCES

[1] M. Milenković, L. Švadlenka, V. Melichar, N. Bojovic, and Z. Avramovic, "SARIMA modelling approach for railway passenger flow forecasting," *Transport*, vol. 33, no. 5, pp. 1–8, Mar. 2016.

[2] X. Wang, N. Zhang, Y. Chen, and Y. Zhang, "Short-term forecasting of urban rail transit ridership based on ARIMA and wavelet decomposition," in *AIP Conf. Proc.*, 2018, pp. 1–13.

[3] L. Tang, Y. Zhao, J. Cabrera, J. Ma, and K. L. Tsui, "Forecasting short-term passenger flow: An empirical study on Shenzhen metro," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3613–3622, Oct. 2019.

[4] W. Shu, K. Cai, and N. N. Xiong, "A short-term traffic flow prediction model based on an improved gate recurrent unit neural network," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 16654–16665, Sep. 2022.

[5] G. Dai, C. Ma, and X. Xu, "Short-term traffic flow prediction method for urban road sections based on space-time analysis and GRU," *IEEE Access*, vol. 7, pp. 143025–143035, 2019.

[6] Y. Zhang, Y. Chen, Z. Wang, and D. Xin, "TMFO-AGGRU: A graph convolutional gated recurrent network for metro passenger flow forecasting," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 3, pp. 1–15, Jun. 2023.

[7] Y. Qiao, Y. Wang, C. Ma, and J. Yang, "Short-term traffic flow prediction based on 1DCNN-LSTM neural network structure," *Modern Phys. Lett. B*, vol. 35, no. 2, Jan. 2021, Art. no. 2150042.

[8] T. Bogaerts, A. D. Masegosa, J. S. Angarita-Zapata, E. Onieva, and P. Hellinckx, "A graph CNN-LSTM neural network for short and long-term traffic forecasting based on trajectory data," *Transp. Res. Part C, Emerg. Technol.*, vol. 112, pp. 62–77, Mar. 2020.

[9] J. Wang, Y. Zhang, Y. Wei, Y. Hu, X. Piao, and B. Yin, "Metro passenger flow prediction via dynamic hypergraph convolution networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 12, pp. 7891–7903, Dec. 2021.

[10] A. Agafonov, "Traffic flow prediction using graph convolution neural networks," in *Proc. 10th Int. Conf. Inf. Sci. Technol.*, Sep. 2020, pp. 91–95.

[11] S. Fukuda, H. Uchida, H. Fujii, and T. Yamada, "Short-term prediction of traffic flow under incident conditions using graph convolutional recurrent neural network and traffic simulation," *IET Intell. Transp. Syst.*, vol. 14, no. 8, pp. 936–946, Aug. 2020.

[12] X. Geng, Y. Li, L. Wang, L. Zhang, Q. Yang, J. Ye, and Y. Liu, "Spatiotemporal multi-graph convolution network for ride-hailing demand forecasting," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 3656–3663.

[13] J. Zhang, F. Chen, and Q. Shen, "Cluster-based LSTM network for short-term passenger flow forecasting in urban rail transit," *IEEE Access*, vol. 7, pp. 147653–147671, 2019.

[14] W. Chen, Z. Li, C. Liu, and Y. Ai, "A deep learning model with conv-LSTM networks for subway passenger congestion delay prediction," *J. Adv. Transp.*, vol. 2021, pp. 1–10, May 2021.

[15] J. Ke, H. Zheng, H. Yang, and X. Chen, "Short-term forecasting of passenger demand under on-demand ride services: A spatio-temporal deep learning approach," *Transp. Res. Part C, Emerg. Technol.*, vol. 85, pp. 591–608, Dec. 2017.

[16] L. Zhao, Y. Song, C. Zhang, Y. Liu, P. Wang, T. Lin, M. Deng, and H. Li, "T-GCN: A temporal graph convolutional network for traffic prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 9, pp. 3848–3858, Sep. 2020.

[17] 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 Trans. Intell. Transp. Syst.*, vol. 22, no. 11, pp. 7004–7014, Nov. 2021.

[18] Y. Wang, Y. Qin, J. Guo, Z. Cao, and L. Jia, "Multi-point short-term prediction of station passenger flow based on temporal multi-graph convolutional network," *Phys. A, Stat. Mech. Appl.*, vol. 604, Oct. 2022, Art. no. 127959.

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

[20] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "WaveNet: A generative model for raw audio," 2016, *arXiv:1609.03499*.

[21] C. Lea, M. D. Flynn, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks for action segmentation and detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1003–1012.

[22] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," 2017, *arXiv:1709.04875*.

[23] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *Stat*, vol. 1050, no. 20, 2017, Art. no. 1048550.

[24] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," 2018, *arXiv:1803.01271*.

[25] Y. Wu, H. Tan, L. Qin, B. Ran, and Z. Jiang, "A hybrid deep learning based traffic flow prediction method and its understanding," *Transp. Res. Part C, Emerg. Technol.*, vol. 90, pp. 166–180, May 2018.

[26] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan, "Attention based spatial–temporal graph convolutional networks for traffic flow forecasting," in *Proc. AAAI Conf. Artif. Intell.*, 2019, vol. 33, no. 1, pp. 922–929.

[27] Z. Li, G. Xiong, Y. Chen, Y. Lv, B. Hu, F. Zhu, and F.-Y. Wang, "A hybrid deep learning approach with GCN and LSTM for traffic flow prediction," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 1929–1933.

[28] C. Song, Y. Lin, S. Guo, and H. Wan, "Spatial-temporal synchronous graph convolutional networks: A new framework for spatial–temporal network data forecasting," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 914–921.

[29] Z. Wu, S. Pan, G. Long, J. Jiang, X. Chang, and C. Zhang, "Connecting the dots: Multivariate time series forecasting with graph neural networks," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020, pp. 753–763.

[30] J. Hu and L. Chen, "Multi-attention based spatial–temporal graph convolution networks for traffic flow forecasting," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2021, pp. 1–7.

[31] Y. Liu, C. Mu, and P. Zhou, "The short-term passenger flow prediction method of urban rail transit based on CNN-LSTM with attention mechanism," in *Proc. 18th Int. Conf. Mobility, Sens. Netw. (MSN)*, Dec. 2022, pp. 909–914.

[32] Z. Hou, Z. Du, G. Yang, and Z. Yang, "Short-term passenger flow prediction of urban rail transit based on a combined deep learning model," *Appl. Sci.*, vol. 12, no. 15, pp. 75–97, 2022.

[33] L. Liu, M. Wu, R.-C. Chen, S. Zhu, and Y. Wang, "A hybrid deep learning model for multi-station classification and passenger flow prediction," *Appl. Sci.*, vol. 13, no. 5, p. 2899, Feb. 2023.

[34] J. Zhang, Y. Chen, K. Panchamy, G. Jin, and L. Yang, "Attention-based multi-step short-term passenger flow spatial–temporal integrated prediction model in URT systems," *J. Geo-Inf. Sci.*, vol. 25, pp. 698–713, 2023.

[35] P. Xia, L. Zhang, and F. Li, "Learning similarity with cosine similarity ensemble," *Inf. Sci.*, vol. 307, pp. 39–52, Jun. 2015.

[36] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Process. Mag.*, vol. 30, no. 3, pp. 83–98, May 2013.

[37] R. Zhang, F. Sun, Z. Song, X. Wang, Y. Du, and S. Dong, "Short-term traffic flow forecasting model based on GA-TCN," *J. Adv. Transp.*, vol. 2021, pp. 1–13, Dec. 2021.

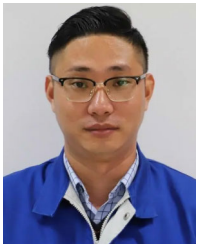
[38] X. Ran, Z. Shan, Y. Fang, and C. Lin, "An LSTM-based method with attention mechanism for travel time prediction," *Sensors*, vol. 19, no. 4, p. 861, Feb. 2019.

[39] S. Kruzick and J. M. F. Moura, "Graph signal processing: Filter design and spectral statistics," in *Proc. IEEE 7th Int. Workshop Comput. Adv. Multi-Sensor Adapt. Process. (CAMSAP)*, Dec. 2017, pp. 1–5.

[40] Y. He, L. Li, X. Zhu, and K. L. Tsui, "Multi-graph convolutional-recurrent neural network (MGC-RNN) for short-term forecasting of transit passenger flow," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 10, pp. 18155–18174, Oct. 2022.
[41] F. Toqué, E. Come, M. K. El Mahrsi, and L. Oukhellou, "Forecasting dynamic public transport origin-destination matrices with long-short term memory recurrent neural networks," in *Proc. IEEE 19th Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2016, pp. 1071–1076.
[42] Q. Tang, M. Yang, and Y. Yang, "ST-LSTM: A deep learning approach combined spatio-temporal features for short-term forecast in rail transit," *J. Adv. Transp.*, vol. 2019, pp. 1–8, Feb. 2019.
[43] M. Liu, A. Zeng, M. Chen, Z. Xu, Q. Lai, L. Ma, and Q. Xu, "SciNet: Time series modeling and forecasting with sample convolution and interaction," in *Proc. 36th Conf. Neural Inf. Process. Syst.*, 2022, pp. 5816–5828.
[44] P. Yi, F. Huang, J. Wang, and J. Peng, "Topology augmented dynamic spatial–temporal network for passenger flow forecasting in urban rail transit," *Int. J. Speech Technol.*, vol. 53, no. 21, pp. 24655–24670, Nov. 2023.

JIE CHENG received the bachelor's degree from Nanjing University of Information Science and Technology, in 2022. He is currently pursuing the master's degree with the School of Electronic and Information Engineering. His research interest includes the prediction of passenger flow in rail transit.

GUANGJIE LIU received the B.Sc. degree in information engineering and the Ph.D. degree in control science and engineering from Nanjing University of Science and Technology, China, in 2002 and 2007, respectively. Since 2006, he has been a Faculty Member with the School of Automation, Nanjing University of Science and Technology. He was with the University of California at Davis, as a Visiting Scholar, from June 2016 to July 2017. He is currently a Professor with the School of Electronics and Information Engineering, Nanjing University of Information Science and Technology, China. His research interests include network and communication security. He has published more than 100 articles in these areas.



SHEN GAO received the master's degree in computer applications from Guilin University of Electronic Technology, in 2010. He is currently a Senior Engineer and an off-campus Supervisor with Nanjing University of Science and Technology, having long been engaged in research and development and management work in modern digital cities and urban rail transit. He is also the Chief Engineer of Nanjing Panda Information Industry Company Ltd., and the General Manager of the Digital City Research and Development Center. He is a member of the Communist Party of China.

AHMED RAZA received the B.E. degree in electrical engineering and automation from Bohai University, China, in 2017, and the M.E. degree in intelligent manufacturing and control engineering from Zhejiang University of Science and Technology, China, in 2020. He is currently pursuing the Ph.D. degree in information and communication engineering with the School of Electronics and Information Engineering, Nanjing University of Information Science and Technology, China. His research interests include passenger flow awareness and utilization technology (passenger flow sensing and application), smart traffic technology, and adversarial machine learning.

JIMING LI received the bachelor's degree in computer science and technology from Sichuan Agricultural University, in 2014. He is currently a Senior Engineer with Nanjing Metro Operation Company Ltd., specializing in the management of subway automatic ticketing and clearing systems, where he is also the Chief of the Technical Equipment Department, Ticketing Department.

WU JUAN received the master's degree in control theory and control engineering from Nanjing University of Science and Technology, in 2004. She has been extensively involved in the project management of construction and development of automatic ticketing systems for rail transit. She is currently the Senior Supervisor of the Station Equipment Department, Nanjing Metro Construction Company Ltd.