

A Passenger Flow Prediction Method Using SAE-GCN-BiLSTM for Urban Rail Transit

Fan Liu, School of Management, Zhengzhou University of Economics and Business, China*

ABSTRACT

To address the problems of existing passenger flow prediction methods such as low accuracy, inadequate learning of spatial features of station topology, and inability to apply to large networks, a SAE-GCN-BiLSTM-based passenger flow forecasting method for urban rail transit is proposed. First, the external features are extracted layer by layer using stacked autoencoder (SAE). Then, graph convolutional network (GCN) is used to capture the spatial features of station topology, and bi-directional long and short-term memory network (BiLSTM) is used to extract the bi-directional temporal features, realizing the extraction of the spatio-temporal features. Finally, external features and spatio-temporal features are fused for accurate prediction of urban rail transit passenger flow. The experimental results show that the proposed method is higher than several other advanced models in the evaluation indexes under different granularities, indicating that the model effectively develops the accuracy and robustness of urban rail transit passenger flow prediction.

KEYWORDS

Bi-Directional Long and Short-Term Memory Network, External Features, Feature Fusion, Graph Convolutional Network, Passenger Flow Prediction, Rail Transit, Spatial-Temporal Features, Stacked Autoencoder

INTRODUCTION

As the result of the accelerated development of urban rail transit (Z. Cheng et al., 2021), the flow of passenger data is exploding and the data structure is becoming increasingly more complex, so the task of constructing a city rail-transit passenger flow forecast is imminent. Traditional prediction models based on statistical theory show disadvantages, such as low prediction accuracy and difficulty in prediction when dealing with prediction tasks of large-scale complex data (Zhang, Z., et al., 2023). The emergence of prediction approaches utilizing machine learning has well solved the previous problem of traditional prediction methods centered on statistical theory being difficult to deal with when working with large-scale complex data, and machine learning approaches can complete data processing tasks more accurately and efficiently to achieve the expected results.

To learn the feature information of passenger flow data more comprehensively, many scholars have proposed a prediction model that combines multiple models. Zhang, Z., et al. (2020) suggested

DOI: 10.4018/IJSIR.335100

*Corresponding Author

This article published as an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

a multilayer long- and short-term memory network (LSTM)-based passenger flow prediction method, which integrates multiple sources of traffic data and various techniques to refine the expression of passenger flow prediction. In the above studies, forecasting is mainly done for the historical information data of passenger flow, but the urban rail transit passenger flow is additionally impacted by some external factors, including weather conditions, holidays, surrounding stations, and other factors. Therefore, Jing et al. (2020) incorporated the data of external characteristic factors that affect passenger flow into the model's input data, used the method of learning and updating the step rate to predict the number of incoming passengers in a certain time period, and weighted it with the historical data to predict the passenger flow for the subsequent interval.

However, considering the complexity, diversity, and large scale of urban rail transit passenger flow data, traditional prediction models often have some limitations, such as difficulty in fully extracting both temporal and spatial features, and difficulty in fully extracting deep features. To address these issues, a SAE-GCN-BiLSTM-based city rail traffic passenger-flow prediction method that combines a stacked autoencoder (SAE), a graph convolutional network (GCN) and a bi-directional long- and short-term memory network (BiLSTM) is suggested to enhance the reliability of the short-term passenger flow prediction model. The innovations of this article can be summarized in four points:

1. To solve the drawbacks of convolutional neural network in passenger flow prediction, the GCN is introduced, which enhances the extraction ability of spatial feature information of station topological structure map and fits the real passenger flow data better.
2. To fully learn the historical passenger flow time series data, BiLSTM is employed to completely exploit the global features of time series and enhance the extraction ability of temporal features.
3. To enhance the extraction ability of the model in the spatial structure and time series features, the GCN-BiLSTM method is proposed, and the features of GCN and BiLSTM models are fused to achieve efficient extraction of the whole passenger flow data representation.
4. To fully extract the deep features of external factors affecting passenger flow, this paper selects 15 external factors data and proposes to use the stack autoencoder model to extract external features layer-by-layer for external factor information to enhance the precision of passenger flow prediction.

RELATED WORKS

According to model development history, short-time traffic forecasting methods mainly include statistical learning-based forecasting models, machine learning-based forecasting models, and deep learning-based forecasting models (Zhang, J., et al., 2020). These three stages are described in detail in the following paragraphs.

Statistical learning-based forecasting models primarily contain the moving average (MA) model (Abellana, 2021), the Autoregressive Integrated Moving Average (ARIMA) model (Su, C., 2020), the variant ARIMA model (Zhang, J., et al., 2024), the Seasonal Autoregressive Integrated Moving Average (SARIMA) model (Liu, S., et al., 2021), and the Kalman filter model, etc. Although these models are theoretically clear and simple to implement, they have poor prediction accuracy when dealing with large-scale complex data tasks.

The main machine learning-based prediction models are the Bayesian algorithm, support vector machine, random forest, decision tree model, and so on. Wu et al. (2020) suggested a novel scaled stacking-gradient boosting decision tree (SS-GBDT). The gradient boosting decision tree (GBDT) model, with data stacking for passenger flow prediction, improves prediction performance. Zhou and Tang (2020) constructed a support vector machine model-based passenger flow prediction model for city rail transit holidays, and they analyzed the characteristics of station passenger flows during holidays by comparing the features of different prediction models. The hybrid prediction models constructed using machine learning have higher prediction accuracy than the traditional mathematical statistics-

based prediction models. However, most of the prediction models at this stage only make individual predictions for one or more stations, thus they cannot fully learn the more complex spatio-temporal characteristics between the stations, and consequently the accuracy of passenger flow prediction in the whole rail transit is somewhat limited.

Deep learning-based passenger flow forecasting models are broadly classified into recurrent neural network (RNN)-based forecasting models, convolutional neural network (CNN)-based models, SAE, and various hybrid deep learning models (Ganaie, et al., 2022).

RNNs are prone to gradient disappearance and explosion, while LSTM can effectively solve the problems of RNNs and can capture the features of long-time sequences in a more comprehensive way (Ameri, et al., 2023; Chen & Gupta, 2024). Zhao et al. (2021) designed a multi-dimensional passenger flow prediction model, incorporating wavelet transform for the characteristics of subway passenger flow with periodicity, nonlinearity, and burstiness. To overcome the two main obstacles in passenger flow prediction, Jing et al. (2021) suggested a prediction model (LSTM-LGB-DRS) that utilizes LSTM and LightGBM to improve prediction performance. Although the recurrent neural network model has a powerful representation extraction capability in processing data from time series, it is still poor in terms of capturing the spatial features of rail transit station networks, and model training is too time-consuming.

The convolutional neural network-based model possesses an excellent spatial feature extraction capability, which effectively compensates for the problem of incomplete feature extraction in spatial data based on the RNN model. Shen et al. (2020) established a CNN model under two-dimensional spatio-temporal matrix. Although CNN can effectively capture spatial features, it is easy to cause confusion or loss of traffic network topology data in the process of converting non-Euclidean urban rail passenger flow data into Euclidean data. The proposed GCN effectively solves the disadvantage of CNN in handling non-Euclidean traffic data by directly embedding non-Euclidean traffic network topology information into the model construction with the help of an adjacency matrix. It has been widely focused on by research scholars in recent years. GCN-based models (Zhang, S., et al., 2022) can consider the spatio-temporal dependence of traffic networks, especially the topological structure information among stations, roads, and regions, and non-Euclidean traffic data's structural information can be fully utilized, and GCN also has faster training efficiency and fewer hyperparameters compared with RNN and CNN. Therefore, Wang et al. (2022) propose a double-channel graph convolutional network (DCGCN)-based model for extracting the relationship between node features and topology. Two datasets were utilized to test the accuracy and plausibility of the model, and the outcomes showed that DCGCN outperformed the other baselines.

The external influencing factors of urban rail traffic are complex, and the traditional shallow structure algorithm has limited ability to represent them, while the deep learning, multilayer, nonlinear mapping structure is able to complete complex function approximation.

In practice, since a single hidden layer of autoencoder is still inadequate in feature extraction capability, researchers have stacked multiple single-layer sparse self-encoders (Zavrak & İskefiyeli 2020) together to form a deeper neural network and named SAE (Cui, et al., 2020). SAE forms a deeper network of encoders and decoders by code-decoding layer-by-layer transfer, which enables it to extract essential features from more complex and large extracting essential features from larger and more complex datasets. SAE has powerful feature extraction capabilities and advantages in data reconstruction that allow it to perform well in a variety of research contexts. Nayak and Chaubey (2020) used a taboo search algorithm to enhance the initial clustering prime selection, and introduced a SAE based on the Softmax regression (SR) classifier for prediction. On the basis of fully considering external factors such as date, weather, and air index, Hou et al. (2022) proposed a passenger flow prediction method that integrates machine learning, time convolutional networks, and LSTM. However, this method places more emphasis on extracting temporal features and neglects the mining of spatial information. Zhang et al. (2022) proposed a passenger flow prediction method that integrates GCN and an improved Transformer. GCN is used to extract spatial features, while the

improved Transformer is used to extract temporal features, taking into account external information contained in social networks. However, this method only considers the shallow features of external factors while ignoring their deeper features. Wu et al. (2023) proposed the Multi Feature Fusion Graph Convolutional Network (MFGCN), where GCN is used to extract spatial dependencies, while LSTM with attention mechanism is used to extract temporal features. However, this method failed to fully consider external factors such as date, weather, and air index.

In summary, GCN has been proven to effectively extract spatial features from passenger flow data, while LSTM has been proven to effectively extract temporal features from passenger flow data. Considering the large scale, complexity, and diversity of city rail traffic passenger flow data, the use of traditional prediction models cannot adequately extract temporal and spatial features, thus cannot meet the accuracy requirements of city rail traffic passenger flow prediction tasks. In order to better address the limitations of existing passenger flow prediction methods, a passenger flow prediction method based on SAE-GCN-BiLSTM is proposed, where GCN is used to extract high-quality spatial features. Compared to the LSTM used in many methods, BiLSTM can better capture long-term dependencies in input sequences and extract higher quality temporal features. In addition, the proposed SAE-GCN-BiLSTM comprehensively considers external factors and utilizes SAE to effectively extract deep level features of external factors, thereby improving predictive performance.

METHOD

The proposed SAE-GCN-BiLSTM-based passenger flow prediction method is mainly composed of the stack autoencoder SAE and the GCN-BiLSTM model, which integrates GCN and BiLSTM. The whole structure of SAE-GCN-BiLSTM is shown in Figure 1.

Step 1: External deep features extraction. Considering that urban rail transit passenger flow data are influenced by external factors, therefore, 15 external influencing factors, including the station's surrounding environment maturity, time distance index, and interchange index, were selected in this study. Although these external influencing factors data have very important reference significance for urban rail traffic passenger flow, it is difficult to describe them clearly in math language and determine the weighting parameters due to their complex data structure relationship. Therefore, the stack autoencoder SAE to extract deep features from the external influence factor data layer-by-layer, was adopted for this study

Step 2: Spatiotemporal features extraction and fusion. For the existing short time that passenger flow prediction methods cannot effectively extract the space characteristics of a rail station network and other problems, and passenger flow changes have strong correlation in both temporal and spatial aspect. Therefore, GCN was used to extract spatial features, while BiLSTM was used to extract time series features. First, the method divided the historical passenger flow data into three temporal patterns: recent, daily cycle, and weekly cycle. Then, the GCN-BiLSTM model was used to extract the spatial and temporal dependence relationship from the historical spatial-temporal passenger flow sequence, and the features under the three-time modes were fused.

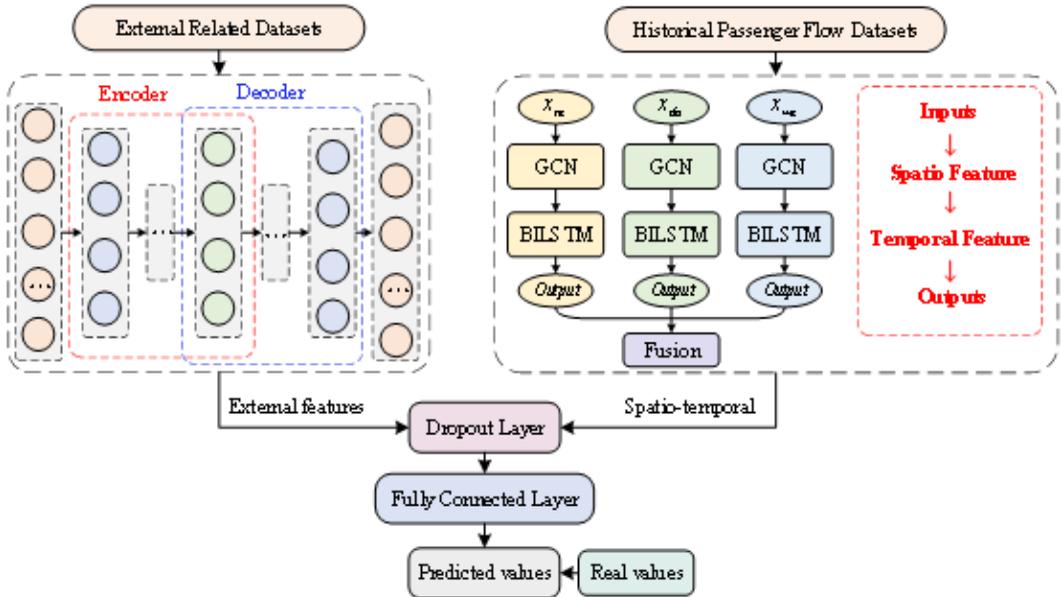
Step 3: Feature concatenation. To obtain a complete feature structure, a feature fusion layer was used to concatenate the external features encoded by SAE with the spatiotemporal features extracted by the GCN-BiLSTM network.

Step 4: Prediction. The final prediction results were outputted using SoftMax through fully connected layers and linear layers. To prevent over fitting, the Dropout mechanism was introduced.

SAE Model Structure

The stack autoencoder is essentially a deep-learning model composed of multiple coding layers and decoding layers (Nayak & Chaubey, 2020). Feature data can be obtained continuously through the learning mechanism of feature extraction layer-by-layer, so as to obtain deeper and more comprehensive feature vectors. The SAE model has excellent feature learning and characterization ability, which can

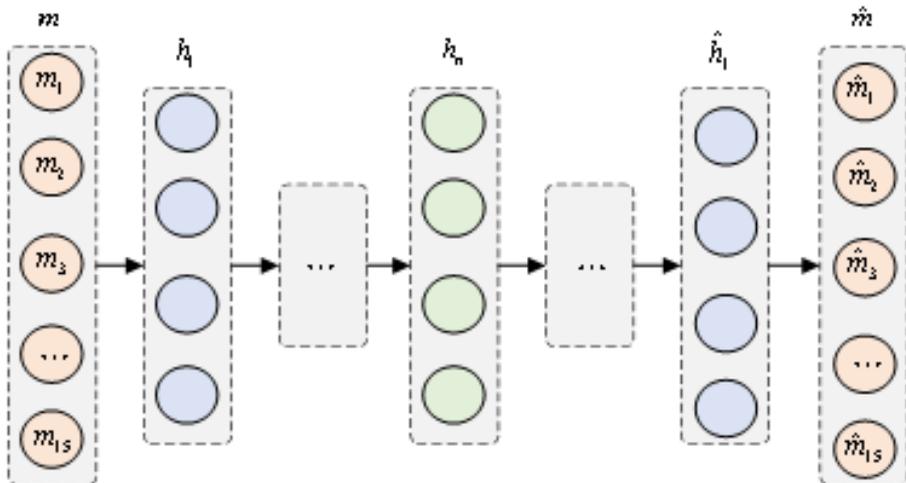
Figure 1. SAE-GCN-BiLSTM-based passenger flow forecasting method for urban rail transit



effectively solve the problem of handling a complex external factor data structure. In this article, the stack autoencoder was adopted to extract features layer-by-layer of external factor data, and finally the deeper external features were obtained. Figure 2 shows the SAE model's structure.

First, the external feature data set m is employed as the input data and reconstructed target data. After the first coding layer, the implicit layer feature h_1 is obtained, then h_1 is input to the next coding layer in the same way, and so on, until h_{n-1} is used as the input data to obtain h_n . Then h_n is input to the next decoding layer each to obtain the output \hat{h}_{n-1} with the same dimension as h_{n-1} . The subsequent decoding layers are trained in the same way, and so on, until \hat{h}_2 is used as input data

Figure 2. Urban rail transit stack autoencoder structure diagram



to obtain \hat{h}_1 with the same dimension as h_1 , where $m \in \{m_1, m_2, \dots, m_{15}\}$; the dimensions of the encoding and decoding layers are symmetric. The encoding function F and the decoding function G are both Sigmoid functions, and the reconstructed target data is made by the decoding function G such that $F(m) \approx m$.

$$F = G = f(m) = 1 / (1 + e^{-m}) \quad (1)$$

The weight matrices and deviations of both the encoding function F and the decoding function G are learned by backpropagation, which ultimately requires learning four sets of parameters, i.e., the encoding layer weight matrix W_h , the decoding layer weight matrix W_m , the encoding layer deviation b_h , and the decoding layer deviation b_m .

Reconstructing the target data \hat{m} is achieved by reducing the reconstruction error, and the loss function, i.e., the reconstruction error needs to be determined before setting update rules for the parameters. SAE model training is performed by adjusting the weights W_h , W_m , and deviations b_h , b_m , and adding constraints on the sparse representation to optimize the objective function as follows:

$$J(W_h, b_h, W_m, b_m) = \frac{1}{d} \sum_{i=1}^d \|\hat{m}(i) - m(i)\|^2 + \lambda \left(\|W_h\|^2 + \|W_m\|^2 \right) + \mu \sum_{j=1}^{d'} KL(\rho | \hat{\rho}_j) \quad (2)$$

$$KL(\rho | \hat{\rho}_j) = \rho \log \frac{\hat{\rho}_j}{\rho} + (1 - \rho) \log \frac{1 - \rho}{1 - \hat{\rho}_j} \quad (3)$$

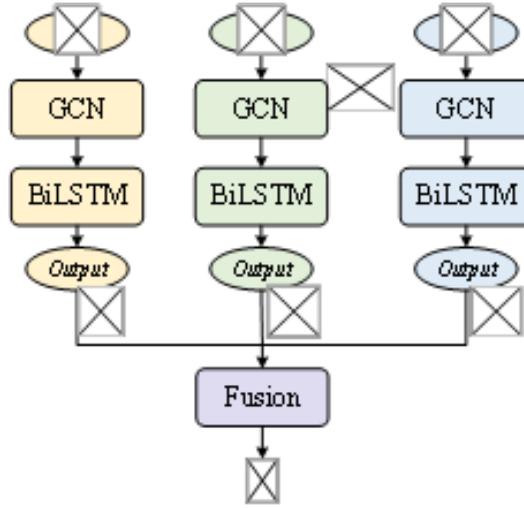
where d is input neurons' number, d' is neurons' number in the hidden layer, $\|W_h\|^2 + \|W_m\|^2$ is the regular term, and the KL scatter is used to calculate the relative entropy $\sum_{j=1}^n KL(\rho | \hat{\rho}_j)$ as the sparse penalty term, ρ the sparsity parameter, $\hat{\rho}_j$ is the average activity of the j -th neuron in the hidden layer, λ and μ are the weight coefficients of the positive regular term and the sparse penalty term, respectively. The hyperparameters are set to $\rho=0.1$, $\lambda=3 \times 10^{-3}$, and $\mu=3$.

GCN-BiLSTM Model Structure

Since there is a strong connection between the time segment to be forecasted and its recent segment, daily cycle segment, and weekly cycle segment, the historical time series is first split into three patterns of recent, daily cycle, and weekly cycle. A model is then built for capturing the spatial correlation features of the track station topology (Sommer, T., et al., 2020) using GCN for each of the three historical time series patterns. The features of each station are considered as the graph signal and processes the graph signal directly onto the graph to capture spatially meaningful patterns and features to obtain the feature vector, including spatial message. It then reads the historical passenger flow information in both directions using BiLSTM (Guo, X., 2020) to obtain the temporal features in the passenger flow data. Lastly, the outputs of these three models are fused through the fully connected layer to get the prediction outcomes. The model structure diagram is depicted in Figure 3.

The subway network, defined on an undirected graph $Q = (K, E, B)$, is built. K denotes the set of metro stations, and the vertices of the graph Q are the rail network's stations. If the station's number is N, the set of track stations can be defined as $K = \{k_1, k_2, k_3, \dots, k_N\}$; each station in K is connected

Figure 3. Structure of GCN-BiLSTM network



by a different edge, and each edge builds connectivity between each station. E denotes the set of edges. Here, two matrices are defined, one of which is used to describe the adjacency relationship between stations and the other is used to describe the station characteristic attributes. The first matrix $B \in R^{N \times N}$ is used to represent the station-to-station adjacency. For example, the adjacency relation from vertex k_i to k_j is denoted as B_{ij} . The second matrix $X \in R^{N \times P}$ is used to represent stations, where rows' number N is stations' number of X and columns' number P is X 's station characteristic attributes. Then the historical passenger flow is all common's characteristic attribute.

In Equation (4), the observed values of passenger flow $x = (x_{t-t_i+1}, x_{t-t_i+2}, \dots, x_t)$ within the historical period of each station are converted to the predicted values of passenger flow $x = (x_{t+1}, x_{t+2}, \dots, x_{t+t_{pt}})$ during the future time period by the function $f(\cdot)$:

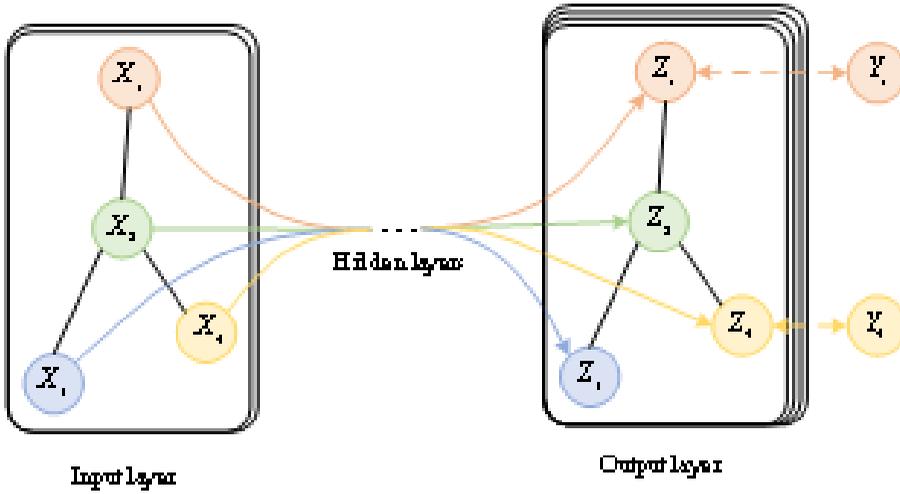
$$\left(x_{t-t_i+1}, x_{t-t_i+2}, \dots, x_t \right) \xrightarrow{f(\cdot)} \left(x_{t+1}, x_{t+2}, \dots, x_{t+t_{pt}} \right) \quad (4)$$

where x_t represents the passenger flow of the station at the t -th time step; t_i represents the selected historical interval's length; and t_{pt} denotes the predicted interval's length.

GCN Model Structure

The city rail transit station network is a typical, irregular, undirected graph (Dalirrooyfard et al., 2022), and each station constitutes an entity node of the graph structure. Compared with using image 2-D convolution to obtain patterns and features of station passenger flow, AFC passenger flow data with graph structure data characteristics, can obtain more original and realistic spatial attributes using GCN. In the proposed GCN-BiLSTM model, the GCN model is applied to the graph structure to extract meaningful patterns and features in the space domain. Traditional CNN models are not applicable to graph structure data, despite their ability to effectively extract the data's local features. Figure 4 depicts the GCN model.

Figure 4. The structure of the GCN model



From Figure 4, the adjacency matrix B and the passenger flow feature matrix X are the input data of the model GCN (Yang et al., 2020), and the propagation between layers in the graph convolution is shown in Equation (5):

$$H^{(p+1)} = f(H^{(p)}, B) = \sigma \left(\hat{D}^{-\frac{1}{2}} \hat{B} \hat{D}^{-\frac{1}{2}} H^{(p)} W^{(p)} \right) \quad (5)$$

Where \hat{B} is the summation of unit matrix I and adjacency matrix B , i.e., $\hat{B} = B + I$; \hat{D} is the degree matrix of \hat{B} ; $H^{(p+1)}$ and $H^{(p)}$ denote the feature matrices of the p -th and $(p+1)$ -th layers, correspondingly. For the input layer, $H^{(0)}$ represents the passenger flow feature matrix X . $W^{(p)}$ denotes the weight matrix of the p -th layer; and σ represents the nonlinear activation function. \hat{B} , \hat{D} , and σ are calculated as Equations (6), (7), and (8):

$$B_{ij} = \begin{cases} 1, & i \neq j \\ 0, & i = j \end{cases} \quad (6)$$

$$\hat{D}_{ij} = \begin{cases} \sum_{j=1}^N \hat{B}_{ij}, & i = j \\ 0, & i \neq j \end{cases} \quad (7)$$

$$\sigma(x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (8)$$

where B_{ij} denotes the connection relationship from station i to station j . In this essay, a 2-layer GCN model is used, so the output of the GCN model is represented in Equation (9):

$$Z = H^{(2)} = f(H^{(1)}, B) = \sigma \left(D^{-\frac{1}{2}} B D^{-\frac{1}{2}} \sigma \left(\hat{D}^{-\frac{1}{2}} \hat{B} \hat{D}^{-\frac{1}{2}} H W^{(1)} \right) W^{(2)} \right) \quad (9)$$

BiLSTM Model Structure

BiLSTM is a bi-directional recurrent neural network structure made by stacking two unidirectional LSTMs in the forward and reverse directions, and Figure 5 shows the structure of the BiLSTM.

In the structure diagram of the BiLSTM, the input $X = \{x_1, x_2, \dots, x_t, \dots, x_n\}$ is entered according to the time series, where t is a certain time scale, and n is the time series' length (Seman, et al., 2023). The model's output contains the results in forward and backward directions. The forward LSTM contains the hidden state h_t^l of the historical passenger flow, while the backward LSTM network reads the same historical passenger flow in reverse, and outputs h_t^r give the future passenger flow. Finally, the two vectors are connected to form the final output h_t of the BiLSTM network. The hidden layer's state at the time t is shown in Equation 10.

$$h_t = [h_t^l \oplus h_t^r] \quad (10)$$

Therefore, after the passenger flow data's spatial feature extraction by the GCN module, the feature information of each node has included the data of the neighboring nodes on the corresponding graph. Then, the BiLSTM model can not only effectively derive the feature information of the time dimension of each node, but can also fuse the feature information of other associated nodes of (Zhao et al., 2023). Since the model stitches the recent segments, daily cycle segments, and weekly cycle segments of the time period to be predicted in chronological order, the time-dimension feature extraction is able to obtain the correlation between different cycle segments. Thus, the temporal dimensional feature extraction unit can obtain the spatio-temporal characteristics of the station traffic. In this essay, a two-layer BiLSTM model is built to extract the passenger flow data of metro stations containing spatial feature information to obtain the spatio-temporal characteristics of the station traffic.

Model Fusion

Using the GCN model for each time mode of the historical observation data separately distills the passenger flow data's spatial features. The features are then inputted into the BiLSTM network to

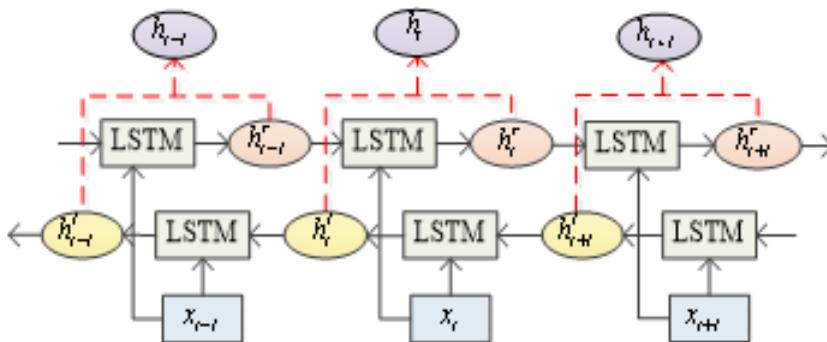


Figure 5. The structure of BiLSTM

collect the temporal features from the passenger flow data containing the spatial feature information, and the outputs of the three time cycles of recent Y_{re} , daily cycle Y_{da} , and weekly cycle Y_{we} are obtained. Finally, the prediction findings are derived by fusing the outputs of the near-term, daily-period, and weekly-period time modes, using the parameter matrix. The specific formula is shown in Equation 11.

$$\hat{Y}_t = \sigma \left(W_{re} \circ \hat{Y}_{re} + W_{da} \circ \hat{Y}_{da} + W_{we} \circ \hat{Y}_{we} \right) \quad (11)$$

Where \hat{Y}_t denotes the prediction result at the t -th time interval; \hat{Y}_{re} , \hat{Y}_{da} , and \hat{Y}_{we} denote the outputs of the three patterns extracted through the BiLSTM network for the recent, daily and weekly cycles, respectively, and W_{re} , W_{da} , and W_{we} correspond to the weights in each mode, reflecting the degree of influence of dependencies in each mode by using the Hadamard product \circ to calculate the average variance of all patterns.

EXPERIMENTAL RESULTS

Experimental Settings and Datasets

In the experiments of this study, Windows 10 was used as the platform, and the main configuration information is shown in Table 1.

The experimental data utilizes each metro station's passenger flow data in Hangzhou from March 2019 to December 2019, including 8 lines and 110 stations. The original data set is the card swipe data collected by AFC ticket vending machines, which mainly includes the time of card swipe occurrence (*time*), metro line ID (*lineID*), metro station ID (*stationID*), card swipe device number ID (*deviceID*), in/out station status (*status*), user identity ID (*userID*), and user card swipe type (*paytype*). Due to differences in the specific operating hours of each station, the daily recording period is uniformly designated from 6:30 to 23:30, and preprocessing operations are carried out on irrelevant, missing, and abnormal records. In order to facilitate the analysis of the predictive ability of the model under different time intervals (10 min, 15 min, and 30 min), the passenger flow information was statistically analyzed at an initial time interval of five minutes during the data collection stage to form an initial sample dataset. To avoid the impact of severe fluctuations in passenger flow during special holidays on the training results of the model, the passenger flow data for seven days of special holidays were excluded. The final total amount of data used for each station was $(30 \times 4 + 31 \times 6-7) \times 12 \times 17 = 60996$ groups; the total data for all stations was $60996 \times 110 = 6709560$ groups. The data set was grouped into training set, validation set, and test set, with the proportions of 6:2:2.

Table 1. Experimental environment configuration information

Environment Configuration	Parameters
IDE Environments	Anaconda3-Windows-x86_64
GPU	NVIDIA GTX 1080Ti(6G)
Hard Disk	1T
CPU	Intel CoreI i7-8750H@2.20GHz
Programming Languages	Python 3.10
Development Framework	TensorFlow 1.14.0

In the task of city rail transit passenger flow prediction, time, stationID, and status are the main data support, so the data with incomplete information of swipe time, station, and in/out status in the original data, are deleted. The abnormal data in non-operating hours are removed. The data with negative values are removed. The passenger flow data of the station in the corresponding time period in a certain cycle and the next time period's passenger flow data are selected, and the small amount and discontinuous abnormal data are repaired using the weighted average of both (Liu, D., et al., 2021).

After cleaning and repairing the passenger flow data, the time field time and the station field stationID are used to count the passenger flow data under different time granularity at a station. Taking a time interval of 10 minutes as an example, Table 2 shows some result cases after data preprocessing.

Evaluation Indicators

To evaluate the experimental results objectively, MAE, RMSE, and WMAPE were used as the experiment's evaluation indicators, and the evaluation indicators were calculated as shown in Equations (12) through (14):

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \tag{12}$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \tag{13}$$

$$WMAPE = \frac{\sum_{i=1}^n |y_i - \hat{y}_i|}{\sum_{i=1}^n y_i} \tag{14}$$

where y_i represents the actual patronage data, \hat{y}_i represents the predicted patronage data, and n represents the sample size.

Implementation Details and Training Strategy

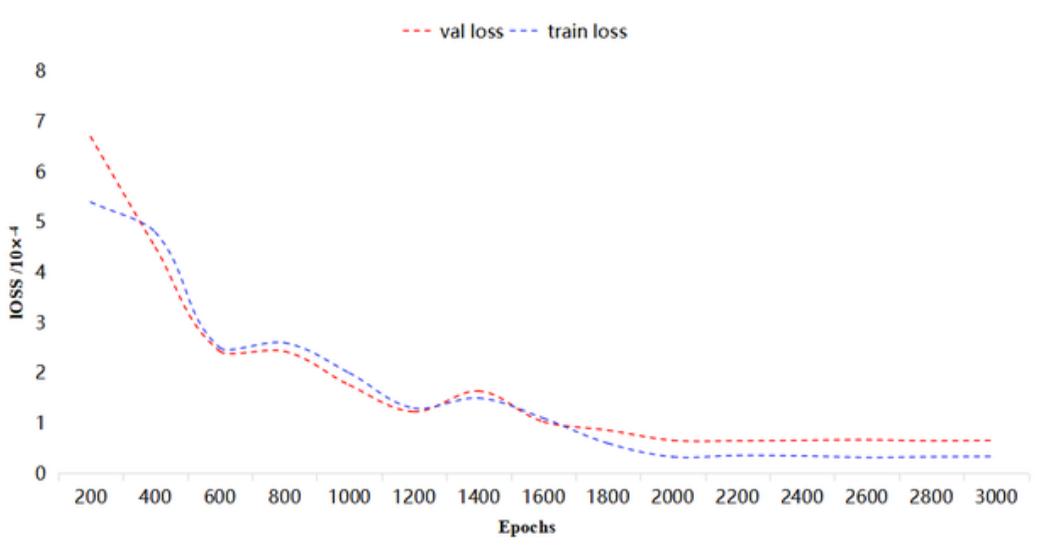
To avoid the influence of random factors, such as random parameter initialization, the SAE-GCN-BiLSTM-based urban rail traffic prediction method is trained several times prior to determining the hyperparameters, and the Model checkpoint and Early stopping (Li, et al., 2020) in the callback function are utilized for saving and terminating the model to keep off-model overfitting during the training process. The training loss and validation loss of the model before terminating the training are shown in Figure 6.

According to Figure 6, it is obvious that the training loss and validation loss fluctuate greatly in the first 19 epochs, and then continue to perform smoothly, showing better robustness of the model.

Table 2. Data pre-processing result cases for a station

ID	Time	StationID	Number of Passenger Flow
1	2019-03-02 17:00:00	18	48
2	2019-03-02 17:10:00	18	95
3	2019-03-02 17:20:00	18	112
4	2019-03-02 17:30:00	18	150
5	2019-03-02 17:40:00	18	137
6	2019-03-02 17:50:00	18	128

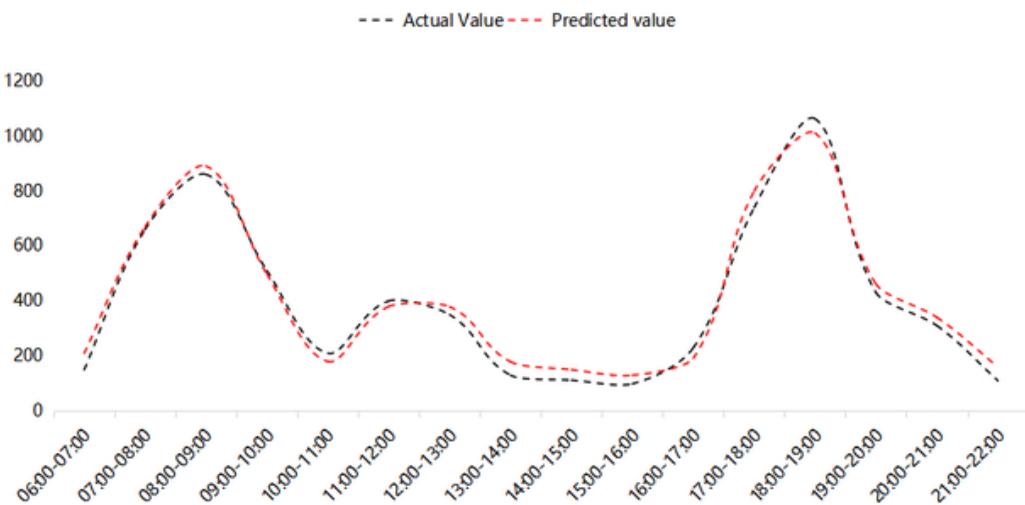
Figure 6. Training loss and validation loss



Experiments were conducted at time granularities of 10 minutes, 15 minutes, and 30 minutes to explore the impact of different learning rates on the predictive performance of the proposed SAE-GCN-BiLSTM model. The results are shown in Figure 7.

In Figure 7, the proposed SAE-GCN-BiLSTM model can achieve the best predictive performance when the learning rate is set to 0.0005. Analysis of the reasons reveals that when the learning rate is too high, the model cannot receive sufficient learning, leading to underfitting. When the learning rate was too low, the parameter update amplitude of the model was too small, making it easy for the model to fall into local optima, and difficult to obtain global optima. Meanwhile, the convergence speed of the model was too slow, requiring more iterations to obtain the best prediction results. In addition,

Figure 7. The impact of learning rate on SAE-GCN-BiLSTM



to prevent the SAE-GCN-BiLSTM model from falling into overfitting, the Dropout mechanism was introduced, and its value was set to 0.5. The relevant training parameter settings of the proposed SAE-GCN-BiLSTM are shown in Table 3.

To clearly observe the effectiveness of passenger flow prediction, taking the data of a certain station on March 13, 2019 as a case study, the operating time was divided into 16 time periods, each spanning one hour. The passenger flow prediction values were obtained using the SAE-GCN-BiLSTM-based model, and the predicted and actual values of passenger flow in each time period were counted. Figure 8 shows this model's prediction effect in each time period.

From Figure 8, the SAE-GCN-BiLSTM model can better forecast the trend of passenger flow of city rail transit, and the result is close to the observed data, which more intuitively reflects the precision of the prediction.

Model Prediction Error Analysis

To research the influence of predicted steps on the prediction results of short-time subway passenger flow, the passenger flow during peak, flat, and full day hours of weekdays are forecasted at intervals of 5, 10, 15, 20, 25, and 30 min, respectively. The peak hours are from 7:30 to 9:30 and 17:00 to 19:30, while the other hours are all flat hours. Using the prediction with SAE-GCN-BiLSTM, Figure 9 shows the comparison effect of prediction error with different step sizes.

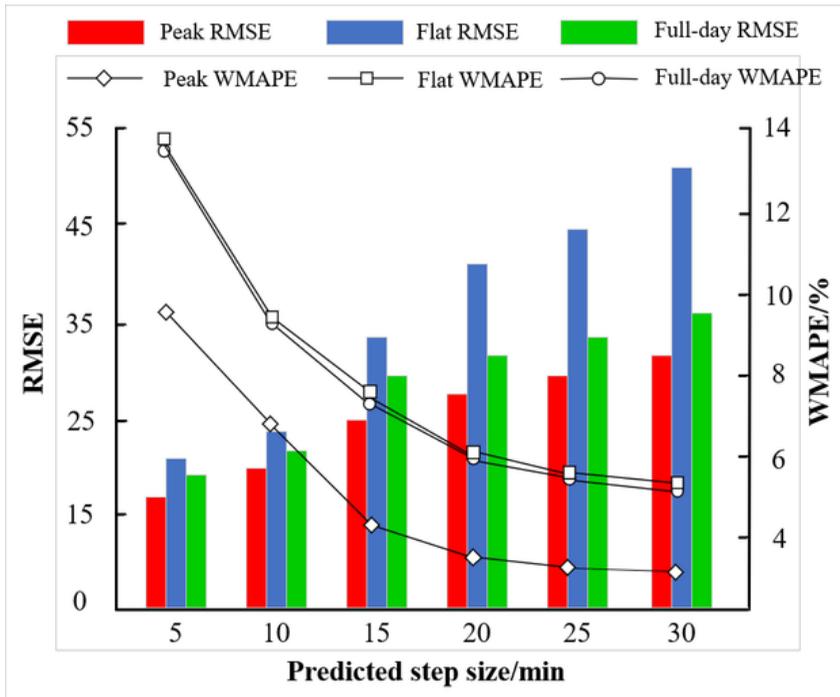
As shown in Figure 9, as the forecast step is gradually increased from 5 min to 30 min, the WMAPE forecast errors of weekday peak, flat, and full-day passenger flows gradually decrease from 9.7%, 13.8%, 13.4% to 2.1%, and 5.43% and 5.36%, i.e., the forecast errors gradually decrease with the increase of the forecast step, which is due to the increase of the time granularity of statistical passenger flows. The granularity increases, the similarity of passenger flow pattern becomes more significant, and the predictability of passenger flow increases. Therefore, in this study, the author chose 30-min intervals for statistical weekday passenger flow to compare the prediction errors of different time periods.

In addition, to verify the capability of the SAE-GCN-BiLSTM-based network model, the ARIMA (Su, C., 2020), LSTM (Zhang, Z., et al., 2020), and ConvLSTM (Baghbani, et al., 2023) models are

Table 3. Model parameter setting

Hyperparameter Name	Values
Epochs	20
Learning Rate	0.0005
Dropout	0.5
Number of hidden layers (SAE)	7
Number of neurons in the hidden layer (SAE)	400
Hidden layer structure (SAE)	[14,12,10,8,10,12,14]
Optimizer (GCN-BiLSTM)	Adam Optimizer
Batch size (GCN-BiLSTM)	32
re	7
da	1
We	1
Number of layers (GCN)	2
Number of layers (BiLSTM)	2

Figure 8. The effect of passenger flow prediction based on the SAE-GCN-BiLSTM model



compared with the prediction errors for peak, flat, and full-day periods, and the prediction errors are shown in Table 4.

From Table 4, at the time level, the prediction indicators of the peak periods are smaller than that of the flat periods, because peak passenger flows are more regular during weekdays, and peak passenger flows are easier to predict. The prediction accuracy of the evening peak passenger flow on weekdays is lower in comparison to that of the morning peak passenger flow.

Through the comparison of several models, it can be found that traditional ARIMA exhibits the weakest performance. ARIMA can achieve feature extraction of time series data, but its fitting performance for non-stationary time series is poor, and its modeling ability for long-term dependencies is weak. Compared to ARIMA, LSTM has stronger modeling ability for long-term dependencies, resulting in better predictive performance. However, LSTM is unable to extract spatial information well, and the length of sequences that can be effectively modeled is usually limited. Compared to LSTM, the ConvLSTM model, with stronger spatial feature extraction ability, can achieve better prediction performance. Furthermore, compared to traditional convolutional neural networks, GCN has stronger spatial feature extraction capabilities. The bidirectional characteristics of BiLSTM make it significantly superior to LSTM in temporal feature extraction. In addition, the introduction of SAE can effectively extract high-dimensional features. Therefore, the proposed SAE-GCN-BiLSTM achieves the minimum MAE, RMSE, and WMAPE, which means, it can achieve the best predictive performance.

Comparison Test With State-of-the-Art Methods

To verify the capability of the models, the proposed SAE-GCN-BiLSTM model was compared with several other advanced models at different time granularities, and the three indicators and their corresponding ranking results are shown in Table 5.

Figure 9. Comparative effect of RMSE and WMAPE with different step sizes

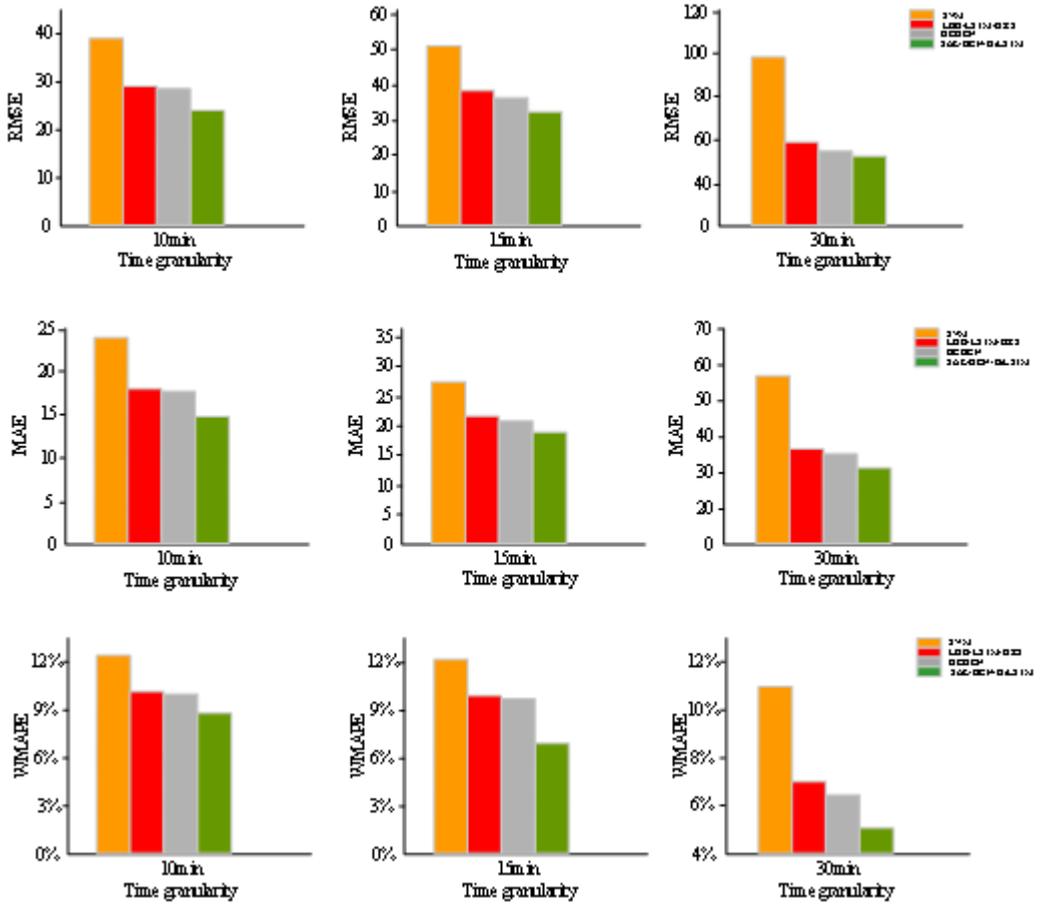


Table 4. Comparison of prediction errors of different models in different time periods

Time Period	Model	RMSE	MAE	WMAPE/%
peak	ARIMA	46.05	21.69	4.79%
	LSTM	39.26	23.58	4.35%
	ConvLSTM	36.30	29.82	3.36%
	SAE-GCN-BiLSTM	29.67	24.71	2.13%
flat	ARIMA	213.05	198.69	47.79%
	LSTM	202.26	191.58	39.43%
	ConvLSTM	189.30	156.82	37.33%
	SAE-GCN-BiLSTM	143.67	108.71	36.36%
full day	ARIMA	65.05	40.69	7.79%
	LSTM	57.26	32.58	6.43%
	ConvLSTM	56.30	32.82	6.33%
	SAE-GCN-BiLSTM	53.67	31.71	5.36%

In Table 5, the experiments show that the SAE-GCN-BiLSTM model achieved better performance than SVM (Zhou & Tang, 2020), LSTM-LGB-DRS (Jing, Y., et al., 2020), and DCGCN (Wang et al., 2022) models, in most cases, demonstrating the good performance capability of this method. Analyzing the reasons for this it can be seen that the traditional machine-learning algorithm SVM can achieve passenger flow prediction, but it is not good at handling multi-classification problems, is sensitive to parameters, and is easily affected by noisy data, leading to a decrease in prediction performance. Compared to SVM, both LSTM-LGB-DRS and DCGCN models, based on deep learning, can achieve better performance. Among them, improvements were made to the LSTM to enhance the modeling ability of the LSTM-LGB-DRS model for long-term dependencies. The lightweight algorithm LightGBM was introduced to achieve accurate fitting of passenger flow data. However, this model has difficulty effectively extract spatial features. The DCGCN model can effectively extract spatial features by using dual-channel GCN, and by integrating attention mechanism and LSTM to effectively extract temporal features, the prediction performance obtained by DCGCN was superior to the LSTM-LGB-DRS model. However, the DCGCN model ignored high-dimensional features in passenger flow data. In the proposed SAE-GCN-BiLSTM, GCN is used to extract spatial features, BiLSTM is used to extract time series features, and SAE is used to extract high-dimensional features, effectively improving the performance of passenger flow prediction.

Table 5. The effect of different models under different time granularity

Time Granularity/ Min	Index	SVM	LSTM-LGB-DRS	DCGCN	SAE-GCN-BiLSTM
10	RMSE	35.29	29.34	28.16	24.57
	rank	4	3	2	1
	MAE	24.15	16.43	17.71	14.39
	rank	4	2	3	1
	WMAPE/%	9.12%	13.33%	9.53%	9.98%
	rank	1	4	2	3
15	RMSE	35.92	38.04	36.39	32.25
	rank	1	4	3	2
	MAE	20.53	21.88	20.73	18.87
	rank	2	4	3	1
	WMAPE/%	12.29%	9.81%	9.96%	7.18%
	rank	4	2	3	1
30	RMSE	54.35	51.96	58.39	53.67
	rank	3	1	4	2
	MAE	38.83	36.58	34.16	31.71
	rank	4	3	2	1
	WMAPE/%	10.98%	6.73%	6.19%	5.36%
	rank	4	3	2	1
Count rank		27	26	24	13
Ave rank		3.00	2.89	2.67	1.44
Total rank		4	3	2	1

Friedman Test

To further verify the significant differences between SAE-GCN-BiLSTM and other models, the Friedman test was used to perform nonparametric tests on several models. The general implementation steps of the Friedman test are: a) Collect experimental data of the algorithm; b) List the best and worst result rankings for each scenario i , which is defined as $r_i^j (1 \leq K \leq j)$; c) Find the average ranking of each algorithm in all problems, and calculate the final ranking $R_j = (1 - n) \sum_{i=1}^n r_i^j$.

The smaller the rank average value, the better the algorithm performance. The Friedman statistical value calculation formula is shown in Equation 15. The smaller the value, the greater the difference between the algorithms. When n and k are large enough (according to experience, $n > 10, k > 5$), it follows a χ^2 distribution of $k - 1$ degrees of freedom:

$$F_f = \frac{12n}{k(k+1)} \left[\sum_j R_j^2 - \frac{k(k-1)^2}{4} \right] \quad (15)$$

The basis for this inspection is from Table 5, and the inspection results are shown in Table 6. Among them, the P-value represents progressive significance. If its value is less than 0.01, it indicates that there is a significant difference between the various data.

In Table 6, the P-value is 1.04E-03, which is much less than 0.01, indicating significant differences between the proposed SAE-GCN-BiLSTM and several other comparative models. From the rank average values of several models, the proposed SAE-GCN-BiLSTM also achieved the minimum results. Overall, in terms of statistical significance, the predictive ability of the proposed SAE-GCN-BiLSTM is significantly improved compared to other advanced comparative models.

Ablation Experiments

The ablation experiment is a decomposition of the method to verify the effect of each part of the method on the results. In this article, the main model's prediction results are compared with GCN, BiLSTM, and SAE+GCN. The model's evaluation metrics results are shown in Table 7.

From Table 7, compared with other experimental models, the improvement of the SAE-GCN-BiLSTM model on RMSE, MAR, and WMAPE under the 10 min prediction step is 0.67~5.57, 0.74~6.96, and 0.34%~2.36%, respectively. Under 15 min prediction step, the evaluation index of the SAE-GCN-BiLSTM model improved 0.9~3.78, 0.97~5.45, and 0.13%~2.63%, respectively. Under 30 min prediction step, the evaluation index of the SAE-GCN-BiLSTM model improved 1.82~11.38, 0.87~8.98, and 0.97%~2.43%, respectively. Independent components, such as GCN and BiLSTM, as well as local combination models, such as GCN-BiLSTM, SAE-BiLSTM, and SAE-GCN, can effectively achieve traffic flow prediction. However, they all struggle to balance temporal, spatial, and deep level features, resulting in weaker prediction performance compared to the proposed model. This verifies that each component has a certain contribution to the performance improvement of the proposed model.

The results of this study's experiments indicate that the prediction effect of the SAE-GCN-BiLSTM method gradually improves with the increase of time granularity. The SAE-GCN-BiLSTM

Table 6. The results of the Friedman test

P-Value	SAE-GCN-BiLSTM	DCGCN	LSTM-LGB-DRS	SVM
1.04E-03	1.61	2.56	3.11	3.39

Table 7. Comparison of Prediction errors of each model module under different time granularity

Model	Evaluation Indicators	Predicted Step Size/Min		
		10	15	30
GCN	RMSE	30.14	36.03	65.05
	MAE	21.35	24.32	40.69
	WMAPE/%	12.25%	9.81%	7.79%
BiLSTM	RMSE	25.34	34.04	57.26
	MAE	16.43	19.89	32.58
	WMAPE/%	11.47%	9.36%	6.43%
SAE-GCN	RMSE	25.78	33.15	56.30
	MAE	15.13	19.84	32.82
	WMAPE/%	10.43%	7.31%	6.33%
SAE-BiLSTM	RMSE	25.37	33.27	56.03
	MAE	15.64	20.04	33.01
	WMAPE/%	10.52%	7.42%	6.54%
GCN-BiLSTM	RMSE	25.32	33.42	55.49
	MAE	15.38	19.98	33.20
	WMAPE/%	10.32%	7.50%	6.81%
SAE-GCN-BiLSTM	RMSE	24.57	32.25	53.67
	MAE	14.39	18.87	31.71
	WMAPE/%	9.98%	7.18%	5.36%

in this article has a superior prediction effect and shows strong robustness, which provides important reference significance for practical operation.

CONCLUSION

To address the problem of the low precision of existing passenger flow prediction models, a passenger flow prediction method for city rail transit with SAE-GCN-BiLSTM is proposed. Experiments confirmed the effectiveness of the proposed method. In terms of the analysis and description, the following conclusions can be reached: 1) Using the stack encoder SAE can take into account the external factors that affect the passenger flow and improve the precision of the forecast; 2) The extraction performance of spatial features of metro station topology can be improved by introducing GCN and; 3) The passenger flow data's temporal features can be accurately obtained by using BiLSTM.

Although the suggested model outperforms other prediction methods in terms of prediction accuracy, the current network structure of the model relies on manual design, so the model's convergence speed needs to be improved, and the proposed SAE-GCN-BiLSTM model lacks generalization performance. In future work, meta heuristic algorithms (Zeng, et al., 2023) will be used to perform spatial automated search optimization on the proposed network, increasing the convergence speed of the network model, thereby enhancing its practicality. In addition, advanced technologies, such as Transformer, will be introduced to deeply improve the structure of the

network model, and it will be applied to more datasets from developed cities, thereby improving its generalization performance.

AUTHOR NOTE

The data used to support the findings of this study are included within the article.

The author declares no conflicts of interest.

This work was supported by the Soft Science Research Program of Henan Province (No. 222400410485).

Correspondence concerning this article should be addressed to Fan Liu, School of Management, Zhengzhou University of Economics and Business, Zhengzhou, Henan, 451191, China. Email: wind091622@163.com

REFERENCES

- Abellana, D. P. M. (2021). Short-term traffic flow forecasting using the autoregressive integrated moving average model in Metro Cebu (Philippines). *International Journal of Applied Decision Sciences*, 14(5), 565–587. doi:10.1504/IJADS.2021.117474
- Ameri, R., Hsu, C. C., Band, S. S., Zamani, M., Shu, C. M., & Khorsandroo, S. (2023). Forecasting PM 2.5 concentration based on integrating of CEEMDAN decomposition method with SVM and LSTM. *Ecotoxicology and Environmental Safety*, 266, 115572. doi:10.1016/j.ecoenv.2023.115572 PMID:37837695
- Baghbani, A., Bouguila, N., & Patterson, Z. (2023). Short-term passenger flow prediction using a bus network graph convolutional long short-term memory neural network model. *Transportation Research Record: Journal of the Transportation Research Board*, 2677(2), 1331–1340. doi:10.1177/03611981221112673
- Chen, X., & Gupta, L. (2024). Training LSTMs with circular-shift epochs for accurate event forecasting in imbalanced time series. *Expert Systems with Applications*, 238(Part C), 121701. doi:10.1016/j.eswa.2023.121701
- Cheng, Z., Trepanier, M., & Sun, L. (2021). Incorporating travel behavior regularity into passenger flow forecasting. *Transportation Research Part C, Emerging Technologies*, 128, 103200. doi:10.1016/j.trc.2021.103200
- Cui, M., Wang, Y., Lin, X., & Zhong, M. (2020). Fault diagnosis of rolling bearings based on an improved stack autoencoder and support vector machine. *IEEE Sensors Journal*, 21(4), 4927–4937. doi:10.1109/JSEN.2020.3030910
- Dalirrooyfard, M., Li, R., & Williams, V. V. (2022). Hardness of approximate diameter: Now for undirected graphs. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)* (pp. 1021–1032). IEEE.
- Ganaie, M. A., Hu, M., Malik, A. K., Tanveer, M., & Suganthan, P. N. (2022). Ensemble deep learning: A review. *Engineering Applications of Artificial Intelligence*, 115, 105151. doi:10.1016/j.engappai.2022.105151
- Guo, X. (2020). Prediction of taxi demand based on CNN-BiLSTM-attention neural network. *Neural Information Processing: 27th International Conference, ICONIP 2020, Bangkok, Thailand, November 23–27, 2020 Proceedings*, 27(Part III), 331–342.
- Hou, Z., Du, Z., Yang, G., & Yang, Z. (2022). Short-term passenger flow prediction of urban rail transit based on a combined deep learning model. *Applied Sciences (Basel, Switzerland)*, 12(15), 7597. doi:10.3390/app12157597
- Jing, Y., Hu, H., Guo, S., Wang, X., & Chen, F. (2020). Short-term prediction of urban rail transit passenger flow in external passenger transport hub based on LSTM-LGB-DRS. *IEEE Transactions on Intelligent Transportation Systems*, 22(7), 4611–4621. doi:10.1109/TITS.2020.3017109
- Jing, Z., & Yin, X. (2020). Neural network-based prediction model for passenger flow in a large passenger station: An exploratory study. *IEEE Access : Practical Innovations, Open Solutions*, 8, 36876–36884. doi:10.1109/ACCESS.2020.2972130
- Li, M., Soltanolkotabi, M., & Oymak, S. (2020, June). Gradient descent with early stopping is provably robust to label noise for overparameterized neural networks. In *International Conference on Artificial Intelligence and Statistics* (pp. 4313–4324). PMLR.
- Liu, D., Zhang, Y., Wang, W., Dev, K., & Khowaja, S. A. (2021). Flexible data integrity checking with original data recovery in IoT-enabled maritime transportation Systems. *IEEE Transactions on Intelligent Transportation Systems*, 24, 2618–2629. doi:10.1109/TITS.2021.3125070
- Liu, S. Y., Liu, S., Tian, Y., Sun, Q. L., & Tang, Y. (2021). Research on forecast of rail traffic flow based on ARIMA model. *Journal of Physics: Conference Series*, 1792(1), 012065. doi:10.1088/1742-6596/1792/1/012065
- Nayak, A. M., & Chaubey, N. (2020, March). Predicting passenger flow in BTS and MTS using hybrid stacked auto-encoder and softmax regression. In *International Conference on Computing Science, Communication and Security* (pp. 29–41). Singapore: Springer Singapore. doi:10.1007/978-981-15-6648-6_3
- Seman, L. O., Stefenon, S. F., Mariani, V. C., & dos Santos Coelho, L. (2023). Ensemble learning methods using the Hodrick–Prescott filter for fault forecasting in insulators of the electrical power grids. *International Journal of Electrical Power & Energy Systems*, 152, 109269. doi:10.1016/j.ijepes.2023.109269

- Shen, C., Zhu, L., Hua, G., Zhou, L., & Zhang, L. (2020, September). A deep convolutional neural network based metro passenger flow forecasting system using a fusion of time and space. In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)* (pp. 1-5). IEEE. doi:10.1109/ITSC45102.2020.9294507
- Sommer, T., Sulzer, M., Wetter, M., Sotnikov, A., Mennel, S., & Stettler, C. (2020). The reservoir network: A new network topology for district heating and cooling. *Energy*, *199*, 117418. doi:10.1016/j.energy.2020.117418
- Su, C. Y. (2020, November). Passenger flow forecast of catering business based on autoregressive integrated moving average and smoothing index prediction model. In *2020 International Signal Processing, Communications and Engineering Management Conference (ISPCEM)* (pp. 53-57). IEEE. doi:10.1109/ISPCEM52197.2020.00016
- Wang, C., Zhang, H., Yao, S., & Liu, M. (2022, August). DCGCN: Double-channel graph convolutional network for passenger flow prediction in urban rail transit. In *2022 8th International Conference on Big Data Computing and Communications (BigCom)* (pp. 304-313). IEEE.
- Wu, J., Li, X., He, D., Li, Q., & Xiang, W. (2023). Learning spatial-temporal dynamics and interactivity for short-term passenger flow prediction in urban rail transit. *Applied Intelligence*, *53*(16), 1–22. doi:10.1007/s10489-023-04508-5
- Wu, W., Xia, Y., & Jin, W. (2020). Predicting bus passenger flow and prioritizing influential factors using multi-source data: Scaled stacking gradient boosting decision trees. *IEEE Transactions on Intelligent Transportation Systems*, *22*(4), 2510–2523. doi:10.1109/TITS.2020.3035647
- Yang, H., Gu, Y., Zhu, J., Hu, K., & Zhang, X. (2020). PGCN-TCA: Pseudo graph convolutional network with temporal and channel-wise attention for skeleton-based action recognition. *IEEE Access : Practical Innovations, Open Solutions*, *8*, 10040–10047. doi:10.1109/ACCESS.2020.2964115
- Zavrak, S., & İskefiyeli, M. (2020). Anomaly-based intrusion detection from network flow features using variational autoencoder. *IEEE Access : Practical Innovations, Open Solutions*, *8*, 108346–108358. doi:10.1109/ACCESS.2020.3001350
- Zeng, J., Yu, X., Yang, G., & Gui, H. (2023). Dynamic robust particle swarm optimization algorithm based on hybrid strategy. *International Journal of Swarm Intelligence Research*, *14*(1), 1–14. doi:10.4018/IJSIR.325006
- Zhang, J., Chen, F., Cui, Z., Guo, Y., & Zhu, Y. (2020). Deep learning architecture for short-term passenger flow forecasting in urban rail transit. *IEEE Transactions on Intelligent Transportation Systems*, *22*(11), 7004–7014. doi:10.1109/TITS.2020.3000761
- Zhang, J., Liu, H., Bai, W., & Li, X. (2024). A hybrid approach of wavelet transform, ARIMA and LSTM model for the share price index futures forecasting. *The North American Journal of Economics and Finance*, *69*, 102022. doi:10.1016/j.najef.2023.102022
- Zhang, Z., Hu, Q., Hou, G., & Zhang, S. (2023). A real-time discovery method for vehicle companion via service collaboration. *International Journal of Web Information Systems*, *19*(5/6), 263–279. doi:10.1108/IJWIS-07-2023-0112
- Zhang, Z., Wang, C., Gao, Y., Chen, Y., & Chen, J. (2020). Passenger flow forecast of rail station based on multi-source data and long short term memory network. *IEEE Access : Practical Innovations, Open Solutions*, *8*, 28475–28483. doi:10.1109/ACCESS.2020.2971771
- Zhao, M., Zhang, X., & Jin, Y. (2021, August). Wavelet embedded attentive Bi-LSTM for short-term passenger flow forecasting. In *2021 IEEE Seventh International Conference on Big Data Computing Service and Applications (BigDataService)* (pp. 177-183). IEEE. doi:10.1109/BigDataService52369.2021.00028
- Zhou, G., & Tang, J. (2020, May). Forecast of urban rail transit passenger flow in holidays based on support vector machine model. In *2020 5th International Conference on Electromechanical Control Technology and Transportation (ICECTT)*. (pp. 585-589). IEEE. doi:10.1109/ICECTT50890.2020.00133

Fan Liu, associate professor, master's degree, graduated from Zhongyuan University of Technology in 2010. Worked in Zhengzhou University of Economics and Business. Her research interests include data analysis, emergency management and information management.