Proceedings of the 2nd International Conference on Machine Learning and Automation
DOI: 10.54254/2755-2721/113/2024.18304

Attention-LSTM Based Modelling for Hangzhou City Metro Passenger Flow Forecasting Study

Wenyi Wang

Nanjing University of Information Science and Technology, Nanjing, China.

202283710044@nuist.edu.cn

Abstract. AbstractsWith the expanding scale of urbanisation, urban traffic problems are increasing, and traffic flow prediction is often of great significance as the core of intelligent transport system (ITS) to solve traffic problems. This paper firstly analyses the temporal and spatial characteristics of the urban traffic flow prediction problem, and for the temporal and spatial characteristics, respectively, establishes a long and short-term memory network LSTM urban traffic flow prediction model based on the attention mechanism; finally, based on the kaggle dataset, the Hangzhou metro passenger flow dataset is used for the effect test of the model, and the results show that the model has a good predictive effect for the prediction of the Hangzhou metro passenger flow. The results show that the model has good prediction effect for passenger flow prediction in Hangzhou metro.

Keywords: deep learning, urban traffic flow prediction, temporal-spatial properties, attention mechanisms.

1. Introduction

Intelligent Traffic System (ITS) is an effective means to solve the current problems of urban traffic in a time of increasing traffic problems. Traffic flow prediction, as an important part of ITS, plays an important role in optimising all aspects of urban operations.

As an important part of urban transport, metro and station have the characteristics of large passenger flow, multiple data, and high real-time requirements, and it is important to do accurate prediction of metro passenger flow for metro operation. With the development of China's intelligent transport system and big data technology, urban rail transit system (Automatic Fare Collection System, AFC) has accumulated a considerable scale of passenger flow changes in the original data, with the help of artificial intelligence technology, these data can be mined in depth, so as to accurately predict the future flow.

The metro network in Hangzhou is increasingly expanding, and as a rapidly developing transport hub, the difference in passenger flow between weekdays and holidays in Hangzhou is also increasing year by year. How to use historical data to build accurate prediction models is one of the key issues in the current operation and management of metro rail. In this paper, the passenger flow data set of Hangzhou metro is used as the main analysis data set on the kaggle platform; after data preprocessing, the data is sorted into hourly station passenger flow data, and the flow data is visualised to analyse the temporal and spatial characteristics of Hangzhou metro passenger flow, and to discover the characteristics of the morning and evening peak hours of the passenger flow, and use them to make adjustments to the passenger flow prediction model; a convolutional neural network and long-short time network-based prediction model is established. The prediction model based on convolutional neural network and long-short time network is established to accurately predict the passenger flow.

2. Relevant studies

Traffic flow depends on vehicles, pedestrians and other disturbances, and is closely linked to changes in time and space, and exhibits huge randomness and uncertainty. A lot of research has been done on this by different researchers and scholars. Various prediction models have been developed for short-term traffic flow prediction using methods from various subject areas, which are usually classified into three methods: statistical methods, machine learning methods and deep learning methods.

Zheng Weizhong[1] took the advantages of back propagation neural network and applied radial basis function in describing complex nonlinear systems, they combined the two models to construct a new Bayesian neural network, so that the model prediction is always carried out along the direction of low error, improving the prediction accuracy compared with the two types of BP neural network and RBF neural network models. On the base of Slime Mould Algorithm (SMA) optimization, Yue Xinxin[2] et al. proposed a Support Vector Regression (SVR) model, using SMA to efficiently optimise the penalty parameters and kernel function parameters of the SVR model, which resulted in higher accuracy compared to the SVR model based on Particle Swarm Optimisation Algorithm (PSOA) and Sparrow Search Algorithm (SSA). SVR model with the highest prediction accuracy. A fuzzy neural model (FNM) was developed by Ying[3], the input data is divided into multiple clusters by fuzzy method, and adaptively adjusts the model coefficients according to the real-time traffic conditions to enhance its prediction capability. Feng Ning[4] et al. explored the characterisation and spatio-temporal correlation analysis of spatio-temporal data with graph structure other than the traditional grid data, and proposed the Multi-component Spatio-temporal Graph Convolutional Model (MCSTGCN), which sets up the near-term, daily and weekly cycles 3 components to describe the three-dimensional spatiotemporal characteristics of traffic data, fusing the outputs into a parameter matrix, resulting in that the final prediction obtained are outstanding in the medium- to long-term forecasting. Mingjian Zhang[5] et al., based on temporal information, proposed a new model augmented Transformer, using multi-graph convolutional network to model spatial features from different perspectives, simulating short - and long-term time features separately and adopting a lightweight structure to improve the model’s real-time response speed.

Zhang[6] et al. proposed a k-nearest neighbour (KNN) model, and established a KNN based short-time traffic flow prediction system for urban expressways from old databases, search methods, and parameters, and the accuracy of the method was above 90%. Zeng[7] et al., based on SVM, combining with the autocorrelation of traffic flow, proposed an modified prediction model consisted of ARIMA and SVM, and compared the combined model with a single SVM and ARIMA model, and the results proved that the combined model was the most effective. A system based on k-means and decision trees is proposed by Maria Viorela Muntean[8] for Key traffic flow time slots in traffic data, including two main stages. In the initial stage, K-means clustering divides the data into two different traffic clusters, normal and abnormal, by calculating the distance between instances. In the next stage, a decision tree classifier is applied to predict time slots with respect to the previously found class attributes. It was shown that the accuracy of detection of abnormal instances in the collected records was 99.73% using this model. Kumar[9] proposed a model using Seasonal ARIMA (SARIMA) model, After making the necessary differences to smooth the input time series, The appropriate order of SARIMA model was determined by drawing the autocorrelation function (ACF) and partial autocorrelation function (PACF), and the effect of increasing the sample size of input data on the prediction results was studied. Cheng[11] et al. used an compositive econometrics and mixed deep learning methods to construct a forecasting framework. In the first step, a VAR model is used to check whether the relationships of these variables are predictable and classify three traffic variables: TI, speed and traffic volume. The second step is to construct a hybrid model to perform the attention mechanism fusing the CNN model and the LSTM model into two channels, and the multi-feature velocity prediction of spatial position is carried out.

Finally, a spatio-temporal heat map is used to visualise the predicted speeds, to more intuitively assess the future trend of traffic conditions and to make suggestions for congestion management.

Ma[10] et al. combined the benefits of sequential data and the long-term dependence of LSTM, introducing Bi-LSTM into the prediction model, which effectively overcame the large prediction error. Zheng[12] et al. designed a hybrid deep learning model based on Conv-LSTM network, in which an attention mechanism is designed, which automatically assigns different weights to distinguish the importance of different flow time series. In addition, a Bi-LSTM module is added and connected to catch the tendency of the traffic flow in 2 directions, forward and backward, and the periodic features are extracted by considering the daily and weekly periodicity. Finally, the obtained spatio-temporal and periodic features are connected to input the regression layer for prediction. The performance comparison with other models such as LSTM and DCRNN in different scenarios shows that the prediction results of AT-Conv-LSTM are more similiar to the real situation. Dai[13] et al. combined spatio-temporal analysis and GRU to build a new prediction model, which analyses the space-time correlation of the collected traffic flow data, defines the optimal inputs using feature selection algorithm and converts the traffic flow data into a two-dimensional matrix, and then processes the spatio-temporal feature information within the matrix through the GRU to reach the prediction effect.

In summary, it can be seen that the input data type of artificial intelligence algorithms is more flexible, and they can effectively capture non-linear laws by virtue of their own learning ability and adaptive ability. In particular, due to the deeper structure and emphasis on feature learning, deep learning can represent the complicated association between input and output more accurately. However, a single mathematical and statistical model or an artificial intelligence model has its own limitations; a hybrid model can improve prediction accuracy to a certain extent by combining the advantages of a variety of single prediction models. The hybrid model extracts the advantages of various single prediction models, which can increase the prediction accuracy. Based on the neural network structure of deep learning, this paper introduces detailedly the application of self-attention mechanism and LSTM in the field of short-time traffic flow prediction, and conducts in-depth research and discussion.

3. Methodology of this paper

3.1. Recurrent Neural Network RNN and its variant LSTM

Recurrent Neural Network (RNN)[14] is mainly used to process sequence data. The remarkable feature of RNN is that the model can take the output of the current time step as the input of the next time step, thus forming a serial structure that preserves the dependencies in the data, which is suitable for time series data.

The traditional RNN structure includes an input layer, one or more loop hidden layers, and an output layer. Loop unit is before receiving the current of time step input and hide status, and to generate the output of time step and hidden state, as shown in Figure 1 after unfolding. The unfolded RNN has a repetitive structure with shared parameters, reducing the amount of network parameters which need to be trained. However, in the backpropagation process, the gradient is gradually and recursively updated with time steps, and when the eigenvalue of the weight matrix is less than 1 or more than 1, the gradient will show exponential decay or growth, leading to the problem of disappearing or exploding gradient, which seriously affects the performance of the RNN in long sequence tasks.

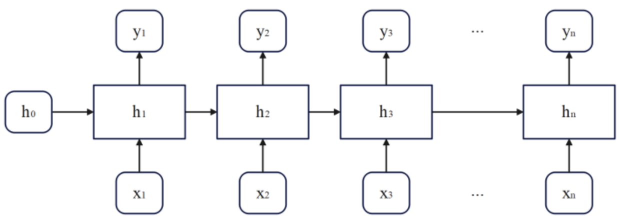

Figure 1. RNN unfolding diagram

To solve the problems of gradient explosion, gradient vanishing and long-term dependency which traditional RNN stuck in, Hochreiter et al. proposed Long-Short Term Memory (LSTM) networks, and LSTM has become one of the most effective sequence models in practical applications. Unlike the hidden unit of RNN, LSTM can better model short-term and long-term dependencies by adding a gating mechanism that allows information to be selectively retained or discarded as it flows through the network. LSTM has hidden unit called the LSTM cell, and its structure is shown in Figure 2.

LSTM[15] cell consists of three types of gating: input gates, oblivion gates, and output gates. These gating are essentially fully connected layers that achieve selective passing of information through sigmoid functions and dot product operations. When the door control output is 0, said information is blocked; When the output is 1, it indicates that the message is completely passed. Input gate: controls the current input data x_t entering into the memory cell, i.e., decides which to store into the current memory cell c_t in the current memory unit. Oblivion Gate: A key component used to selectively retain or unregister historical information, thereby mitigating gradient decay or explosion problems in time back propagation. The forgetting gate acts on the memory cell from the previous moment in timec_{t-1} and determines which historical information affects the current memory unit c_t. Output gate: controls the response of a memory cell c to the current memory cell, it determines the information of the memory cell output at the current time step t.

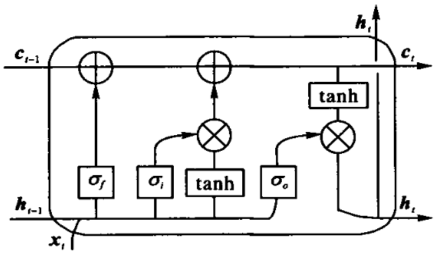

Figure 2. LSTM cell structure

According to its excellent properties, LSTM has been widely used in tasks related to sequence learning, such as speech recognition, language modeling, lexical annotation, and machine translation.

3.2. Gated Recycling Unit GRU

Another widely used LSTM unit variant is the Gated Recurrent Unit (GRU), a simplified structure proposed by Cho et al. The structure is shown in Fig. Later, scholars proposed Local Feature-based GRU (LFGRU) network based on GRU, which is a mixed approach combining manual feature design and automatic feature learning, aiming to achieve machine health monitoring. The steps are as follows: first, features are extracted from a window of input time series; then, design enhanced two-way GRU helped network, and applied to the local characteristics of generating sequence, said learning characteristics; and finally, the machine state is predicted by a supervised learning layer. This LFGRU uses the bidirectional GRU structure to enhance the model representation and validates the validity and robustness of the model in three machine health monitoring tasks.

Instead of using peephole connections and output activation functions, and linearly self-connected memory cells, the GRU operates directly on linearly accumulated hidden states h. The GRU controls the flow of information through two gating gates, which combine the input gate and forgetting gate in the LSTM into an updating gate to control the updating of the hidden states. When ut=0u_t = 0ut=0, the information of the initial time step is maintained even if the sequence is long. The reset gate determines whether the previously hidden state should be ignored, and the update gate and reset gate are denoted by u and r respectively. The forward propagation of GRU [16] is computed as follows:

$$\begin{cases} u_t = \sigma(W_{xu}x_t + W_{hu}h_{t-1} + b_u) \\ r_t = \sigma(W_{xr}x_t + W_{hr}h_{t-1} + b_r) \\ \widetilde{h_t} = \tanh(W_{xh}x_t + W_{hh}(r \cdot h_{t-1}) + b_h) \\ h_t = u_t h_{t-1} + (1 - u_t) \cdot \widetilde{h_t}) \end{cases} \quad (1)$$

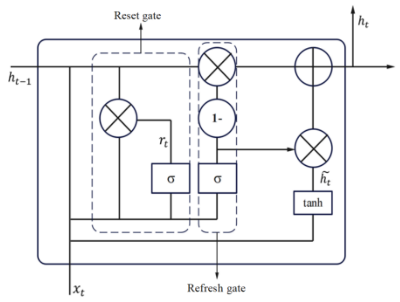

Figure 3. GRU unit structure

3.3. Transformer models and attention mechanisms

The Transformer model uses an Encoder-Decoder (ED) architecture, where the encoder and decoder each consist of a stack of 6 layers, as shown in Fig 4. The input data is first processed through the encoder and the encoded results are passed to each layer of the decoder for attention computation.

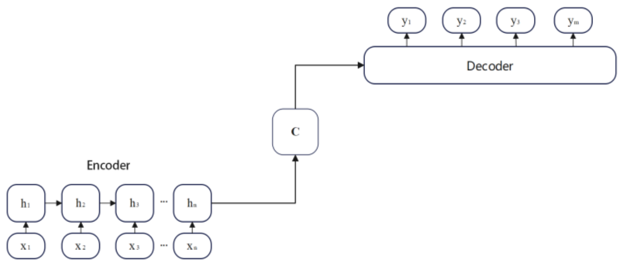

Figure 4. Encoder-decoder architecture

Encoder of each layer is composed of two sub layers, long note layer (multi - head attention) and feed forward connection layer (feed - forward).The decoder includes three sub layers: shielding long note, long notice layer and layer feed forward. After each layer has a remaining links and a layer of standardization. In the decoder, an additional masked multi-head attention layer is used to ensure that data after the current position does not participate in the current attention computation. Since the encoder inputs data of different lengths and the decoder usually uses the maximum length of the data as the unit of computation and only focuses on the effect of previous data on the current one, this layer ignores subsequent unprocessed data through a masking mechanism.

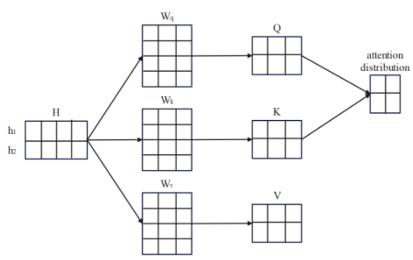

Figure 5. Introduction of the self-attention mechanism

Attention Mechanism[17] Attention Mechanism, proposed by Treisman and Gelade, is a combinatorial function that enhances the effect of key inputs on outputs by calculating the probability distribution of attention. Typically, Attention Mechanism is based on the Encoder-Decoder framework, and is especially widely used in the field of text processing. It is widely used in the fields of machine translation, speech recognition and image processing, and has become an important concept in the field of neural networks.

There are three reasons for the rapid development of Attention Mechanism: firstly, it is an advanced multitasking method; secondly, it increases the interpretability of neural networks; and finally, it helps to overcome some of the problems of RNNs such as performance degradation with increasing input length and computational inefficiency due to input order dependence. Self-attention mechanism is based on improvement of attention mechanism, which is more adaptable at finding the internal correlation of data because of the reduced reliance on external information. Self-Attention (SAN), also known as internal attention, usually focuses only on the relationship between the internal elements of the data, and is suitable for scenarios that do not rely on external information. Traditional attention mechanisms differ in the content of the source and target, establishing a dependency between the source and target words; whereas in Self-Attention mechanisms, attention acts between the internal elements of the source or target.

The Transformer architecture introduces a self-focusing mechanism where global input and output dependencies are captured entirely by self-focusing, avoiding the use of recursive structures. The architecture employs Scaled Dot-Product Attention, which computes similarity by dot product for faster and more space-efficient computation. Specifically, the inputs are linearly transformed to obtain the query matrix Q, the key-value matrix K, and the summation matrix V, which are computed as follows:

$$\boldsymbol{Attention(Q,K,V) = softmax(\frac{QK^T}{\sqrt{d_k}})V} \quad (2)$$

where d_k is the columns number of the matrices Q, K, representing the vector dimension.

3.4. LSTM based on attention mechanism

Existing literature mostly adopts neural network combination models to make the most of the advantages of different models and comprehensively capture the implicit temporal and spatial characteristics in traffic flow, so as to achieve better results in various prediction tasks. In this paper, we propose an LSTM prediction model that combines the self-attention mechanism, and the model structure is shown in Fig 6. The model consists of an input layer, two LSTM layers (each with a hidden unit size of 64), a multi-attention mechanism, and a fully connected layer.

In the model, the preprocessed two-dimensional data matrix of traffic flow is firstly passed to the LSTM hidden layer through the input layer, and the result is input to the attention layer after two layers of bi-directional LSTM unit computation. In the notice, use the LSTM the output of the hidden layer as the input to calculate pay attention to the score, and the attention weights are obtained after normalisation. Subsequently, the weighted inputs are passed to the fully-connected layer to produce the end traffic flow prediction value y. This design helps the model to better extract the temporal dependence in the sequence and use the attention mechanism to concentrate on the important parts of the inputs to improve the prediction performance.

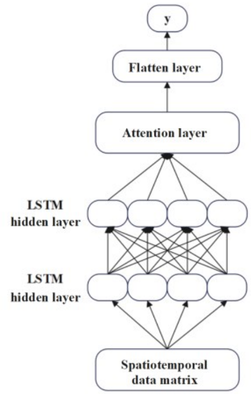

Figure 6. CNN-LSTM prediction model based on attention mechanism

The realization idea of Attention-LSTM model based on attention mechanism is as follows:

Step 1: Data preprocessing, using interpolation to fill in missing values, remove anomalous data, and normalise the data.

Step 2: Feature extraction, the preprocessed traffic flow data is constructed into a data matrix and input into the LSTM model based on the attention mechanism.

Step 3: feature enhancement, the output of the LSTM hidden layer is h_t into the self-attention mechanism layer, weighted and summed by weight assignment, and then fed into the fully connected layer to compute the output at moment t y_t at moment t.

Step 4: Inverse normalisation, where the output values are y_t Inverse normalisation is performed to obtain the actual value.

Step 5: Comparison of results, where the predictions of the model are analysed against other models.

4. Data and evaluation indicators

4.1. Data collection and pre-processing

The research data in this paper is the underground passenger flow data and associated data of Hangzhou in January 2019 provided by Kaggle, and the real IC card history record of Hangzhou underground is selected as the data source of the LSTM short-time passenger flow prediction model for urban rail based on the attention mechanism. Due to the differences in the operating time range of each station, outliers are processed to ensure the integrity of the data. Too short or too long passenger flow statistics lack practical significance, which is not conducive to the actual operation arrangement of the station and the emergency warning arrangement, therefore, this paper chooses 1h time interval as the short-time passenger flow statistics of urban rail, and the daily statistics period is 24; the data of 2019-01-01-2019-01-12 is chosen as the historical data set to train the model, and the data of the last 12 h of 2019-01-12 is used as the test set of the model. h of data as the test set of the model to verify the validity of the model.

Taking the data from 1 January to 13 January as an example, the relationship between the underground passenger flow over time is illustrated in Figure 7. From Figure 7, it can be found that The detector has a certain regularity of the distribution of traffic flow time can be used as the prediction of traffic flow data set.

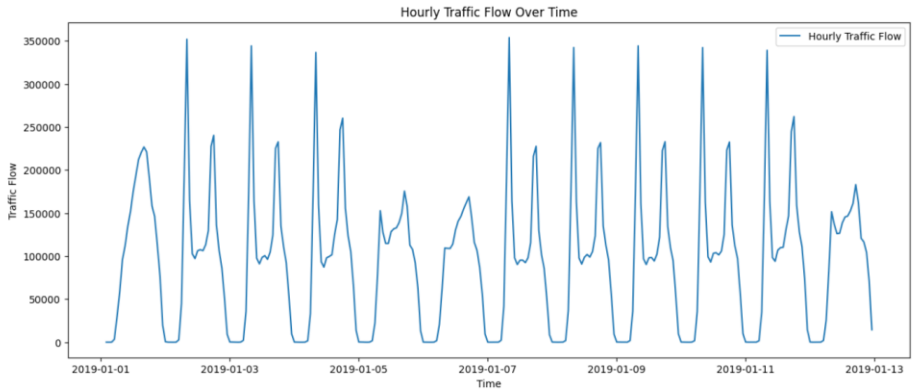

Figure 7. Relationship between metro passenger flow over time

When analysing the single-day metro passenger flow data, it can be observed that its distribution presents typical concentrated and discrete characteristics, as shown in Figure 8. From the figure, it can be seen that most of the passenger flow data is concentrated in the lower flow interval, reflecting the fact that Hangzhou metro has relatively low passenger flow during most of the time, and low flow situations are more frequent. There is a clear aggregation of data in the medium flow range (between 50,000 and 150,000), which is more common and usually corresponds to regular flow fluctuation intervals such as the peaks appearing in the morning and evening. This clustering reflects the "alternating peaks and valleys" nature of the urban rail system, whereby the metro system is significantly affected by the peaks of the commuter peaks during daily operations. The frequency of the data decreases significantly as the number of passengers rises above 200,000, indicating the rarity of high-flow situations. Such peaks tend to correspond to specific events or holiday passenger surges. In addition, a small number of very high traffic anomalies may be caused by traffic anomalies at specific times of the year, such as short bursts of high traffic due to festivals, major events, or other emergencies.

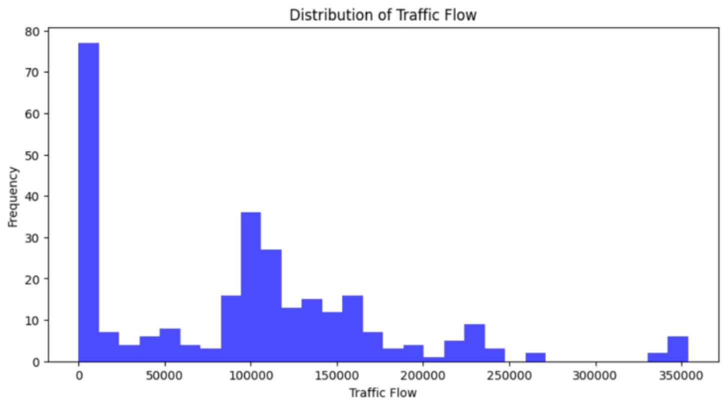

Figure 8. Image of single-day metro passenger flow distribution

According to Figure 9, the metro passenger flow shows a typical "double peak" trend during a 24-hour day, with peaks in the morning peak (around 8 a.m.) and the evening peak (around 18-19 p.m.). This trend may reflect the regularity of urban commuting: the morning peak consists mainly of people travelling to work and school, while the evening peak corresponds to the time when people return home from work and school. Outside of the morning and evening peaks, metro traffic is significantly lower and relatively flat. Particularly during the night and early morning hours, metro traffic is low, indicating that there is less demand for travelling at this time. During the daytime periods outside the peak hours (e.g., midday to afternoon), the flow of passengers fluctuates slightly but is generally stable.

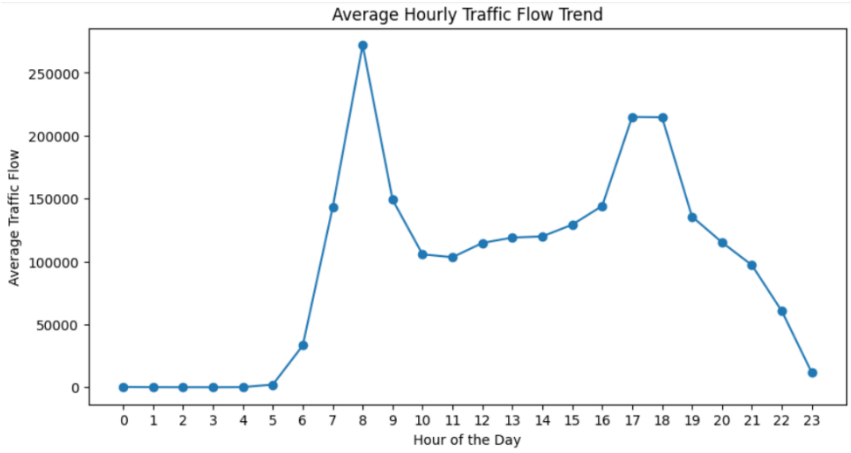

Figure 9. Trends in single-day metro passenger flow

4.2. Model indicators

Time series data prediction belongs to a typical prediction problem, aimed at evaluating the generalization ability and prediction effect of the model, this paper chooses three commonly used evaluation indexes. The details are as follows:

(1) Mean Absolute Error (MAE): MAE can be used to measure model prediction error between the actual value of the average absolute value, and avoids the problem of positive and negative errors cancelling each other out by calculating the average of the absolute value of the difference between the n predicted values and the actual value. The formula is as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \quad (3)$$

(2) Root Mean Squared Error (RMSE): RMSE is the result of squaring the average of the squares of the deviations between the predicted and the actual values of the model, which is used to measure the degree of deviation between the predicted and the real values. The formula is as follows:

$$RMSE = \frac{1}{n}\sqrt{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad (4)$$

(3) Mean Absolute Percentage Error (MAPE): MAPE measures the average of the absolute difference between the actual value and the predicted value, and is expressed in the form of a percentage of the actual value, so as to be able to reflect more intuitively the relative bias of the prediction error in the data of different magnitudes. Its calculation formula is as follows:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100\% \quad (5)$$

5. Results and evaluation

5.1. Summary of the study

In this paper, we use Python to build a passenger flow prediction model based on the deep learning framework PyTorch, and the experimental hardware is NVIDIA 1660Ti GPU.In the training process of the model, the optimiser is chosen to be Adam, the learning rate is set to be 0.001, and the loss function is the Mean Squared Error (MSE), and the activation function of all the layers is used to be the ReLU function. A total of 150 rounds (epochs) of training were performed, and the model was completely updated on the training set once per round. In addition, Batch Gradient Descent (BGD) is used in the

DOI: 10.54254/2755-2721/113/2024.18304

training, and 32 samples (batch size of 32) are used in each update, which not only helps to improve the training speed, but also reduces the variance of the gradient update to a certain extent, so as to take into account the convergence speed and stability.

5.2. Comparison of effects

An analysis of the hourly traffic importance based on the attention weights shows an increasing trend during the first 10 hours, as shown in Figure 4.1. This indicates that as the time delay increases, earlier time points become more important for the current moment prediction; the weights corresponding to delays between 10 and 12 hours reach a peak, indicating that the traffic data at these time points have the greatest impact on the current prediction; after 12 hours, the attention weights drop significantly, indicating that the contribution of data at these time delays to the current prediction weakens; after 16 hours, the attention weights gradually picks up and reaches another peak at 23 h. It is possible that traffic patterns from a day earlier still have a significant impact on the current moment prediction. This attentional weight image reflects the cyclical nature of daily traffic, suggesting that there is a correlation between traffic at different points in the day.

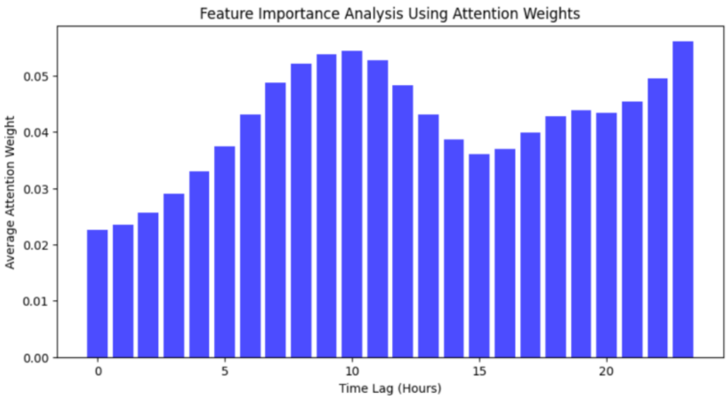

Figure10. Hourly images of attentional weights

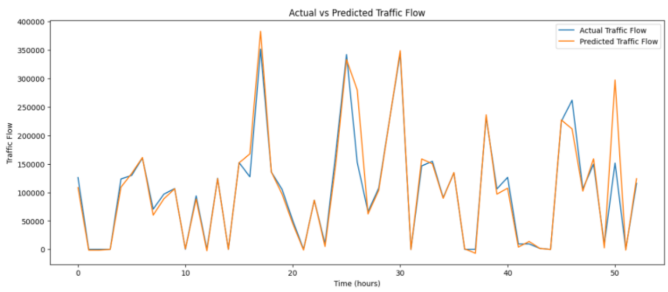

Figure 11. Effectiveness of LSTM for predicting traffic flow based on attention mechanism

6. Summary Outlook

This paper investigates the application of LSTM model based on attention mechanism in urban rail transit passenger flow prediction. By constructing and analysing the passenger flow data of Hangzhou metro, an LSTM prediction model containing multi-head attention mechanism is proposed. The model is able to effectively capture the time-dependent features of short-time and inter-day, thus achieving significant improvement in prediction accuracy. In the experiments, MAE, RMSE and MAPE are chosen to evaluate the model; through the analysis of the attention weights of the model, the differences in the influence of the flows at different time points on the prediction are revealed, which provides a strong support for a better understanding of the spatial and temporal characteristics of urban rail traffic.

DOI: 10.54254/2755-2721/113/2024.18304

In future research, firstly, the data scope can be further expanded to include external influences such as holidays and weather changes in order to improve the generalisation ability and adaptability of the model; secondly, spatial correlation information between stations can be introduced in combination with methods such as graph neural networks to capture the complex spatio-temporal interactions within the metro network so as to achieve a more accurate passenger flow prediction.

References

[1] Zheng Weizhong, Shi Qixin. Short-term traffic forecasting based on Bayesian combinatorial model[J]. China Highway Journal, 2005, 18(1): 85-89.
[2] YUE Xinxin, CHANG Shan, MA Lu, et al. Short-time traffic flow prediction on urban roads based on SMA-SVR model[J]. Journal of Anshun College,2024,26(03):131-136.
[3] Yin H, Wong SC, Xu J, et al. Urban transport flow prediction using a fuzzy-neural approach[J]. Transportation Research Part C: Emerging Technologies, 2002, 10(2): 85-98.
[4] FENG Ning, GUO Shengnan, SONG Chao, et al. Multi-component spatial-temporal graph convolution networks for traffic flow forecasting[J]. Journal of Software, 2019, 30(3): 759-769.
[5] ZHANG Mingjian,YE Baolin,DONG Rui,et al. Short-time traffic flow prediction model based on time-informed augmented Transformer[J]. Software Engineering,2024,27(08):41-45+50.DOI:10.19644/j.cnki.issn2096-1472.2024.008.009.
[6] Zhang L, Liu Q, Yang W, et al. An improved k-nearest neighbor model for short-term traffic flow prediction[J]. Procedia-Social and Behavioral Sciences, 2013, 96: 653-662.
[7] Zeng D, Xu J, Gu J, et al. Short term traffic flow prediction using hybrid ARIMA and ANN models[C]//2008 Workshop on Power Electronics and Intelligent IEEE, 2008: 621-625.
[8] Muntean M V. Identifying critical traffic flow time slots using k-means and decision trees[C]//2020 IEEE 10th International Conference on Intelligent Systems (IS). IEEE, 2020: 364-369.
[9] Kumar S V, Vanajakshi L. Short-term traffic flow prediction using seasonal ARIMA model with limited input data[J]. European Transport Research Review, 2015, 7: 1-9.
[10] Ma C, Dai G, Zhou J. Short-term traffic flow prediction for urban road sections based on time series analysis and LSTM_BILSTM method[J]. IEEE Transactions on Intelligent Transportation Systems, 2021, 23(6): 5615-5624.
[11] Cheng Z, Lu J, Zhou H, et al. Short-term traffic flow prediction: an integrated method of econometrics and hybrid deep learning[J]. IEEE Transactions on Intelligent Transportation Systems, 2021, 23(6): 5231-5244.
[12] Zheng H, Lin F, Feng X, et al. A hybrid deep learning model with attention-based conv-LSTM networks for short-term traffic flow prediction[J]. IEEE Transactions on Intelligent Transportation Systems, 2020, 22(11): 6910-6920.
[13] Dai G, Ma C, Xu X. Short-term traffic flow prediction method for urban road sections based on space-time analysis and GRU[J]. IEEE Access, 2019, 7: 143025-143035.
[14] Sherstinsky A. Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network[J]. Physica D: Nonlinear Phenomena, 2020, 404: 132306.
[15] Yu Y, Si X, Hu C, et al. A review of recurrent neural networks: lstm cells and network architectures[J]. Neural computation, 2019, 31(7): 1235-1270.
[16] Dey R, Salem F M. Gate-variants of gated recurrent unit (GRU) neural networks[C]//2017 IEEE 60th international midwest symposium on circuits and systems (MWSCAS). ieee, 2017: 1597-1600.
[17] Soydaner D. Attention mechanism in neural networks: where it comes and where it goes[J]. Neural Computing and Applications, 2022, 34(16): 13371-13385.