Hindawi
Security and Communication Networks
Volume 2021, Article ID 7751484, 11 pages
https://doi.org/10.1155/2021/7751484

Review Article

AGG: A Novel Intelligent Network Traffic Prediction Method Based on Joint Attention and GCN-GRU

Huaifeng Shi[1,2] **Chengsheng Pan**[1] **Li Yang**[1] **and Xiangxiang Gu**[2]

[1]*School of Automation, Nanjing University of Science and Technology, Nanjing 210094, China*
[2]*School of Information Engineering, Dalian University, Dalian 116622, China*

Correspondence should be addressed to Huaifeng Shi; shihuaifeng314@126.com

Received 2 July 2021; Accepted 14 September 2021; Published 30 September 2021

Academic Editor: Jinwei Wang

Timely and accurate network traffic prediction is a necessary means to realize network intelligent management and control. However, this work is still challenging considering the complex temporal and spatial dependence between network traffic. In terms of spatial dimension, links connect different nodes, and the network traffic flowing through different nodes has a specific correlation. In terms of spatial dimension, not only the network traffic at adjacent time points is correlated, but also the importance of distant time points is not necessarily less than the nearest time point. In this paper, we propose a novel intelligent network traffic prediction method based on joint attention and GCN-GRU (AGG). The AGG model uses GCN to capture the spatial features of traffic, GRU to capture the temporal features of traffic, and attention mechanism to capture the importance of different temporal features, so as to realize the comprehensive consideration of the spatial-temporal correlation of network traffic. The experimental results on an actual dataset show that, compared with other baseline models, the AGG model has the best performance in experimental indicators, such as root mean square error (RMSE), mean absolute error (MAE), accuracy (ACC), determination coefficient (R^2), and explained variance score (EVS), and has the ability of long-term prediction.

1. Introduction

Cisco annual Internet report (2018–2023) notes that device functionality will be combined with higher bandwidth and more intelligent networks by 2023, and the number of devices linked to IP networks will be more than three times the global population [1]. With the increasing number of terminals, the enrichment of multimedia applications, and the continuous expansion of network capabilities, network traffic management has become a critical and challenging task. Real-time and accurate network traffic prediction can greatly improve the control gain of the network.

The existing network traffic prediction methods are divided into model-driven traffic prediction methods and data-driven traffic prediction methods. Model-driven traffic prediction methods are also called parameterization methods, including autoregressive moving average model (ARMA) and autoregressive integrated moving average mode (ARIMA). Laner et al. introduced the ARMA model, which can predict network traffic [2]. Guo et al. introduced the ARIMA model and tested the algorithm with the data collected by a backbone switching node. The experimental results show that compared with other network traffic prediction methods, the model has a better effect in dealing with nonstationary series and higher prediction accuracy [3], so the ARIMA model and its variants are widely used and can well explore the time correlation of network traffic [4–6]. Model-driven traffic prediction methods mostly use a polynomial fitting function to approximate the actual network traffic and then make the fitting effect better through a large number of parameter tuning. However, it is difficult to capture the nonlinear characteristics of network traffic, such as fast fluctuation and time dependence.

The data-driven traffic prediction method can automatically learn statistical rules from a large quantity of historical data to intelligently capture the nonlinear characteristics of network traffic. Specifically, data-driven traffic prediction methods can be divided into machine learning prediction methods and deep learning prediction methods. Among them, machine learning prediction methods include support vector regression (SVR) and k-nearest neighbor algorithm (k-NN). Bermolen et al. applied support vector regression (SVR) to link load prediction [7]. Kremer et al. chose two different machine learning algorithms, SVR and KNN, to explore the balance between complexity and estimation accuracy [8]. However, machine learning methods are not sufficient for processing high-dimensional data and rely on feature engineering. Therefore, the universality of this method is weak.

Compared with machine learning prediction methods, deep learning prediction methods can not only retain the learning characteristics but also ensure the relevance between tasks and effectively address time series problems. Wu et al. proposed a network traffic prediction method based on a deep neural network (DNN), which proves the superiority of the deep learning prediction method in traffic prediction [9]. Lazaris et al. used actual network traffic tracking from ISPs to train long-term short-term memory (LSTM) neural network and generate predictions in a short time. Experiments show that LSTM can predict network traffic with low error [10]. Azzouni et al. proposed an LSTM RNN framework for predicting a large-scale network traffic matrix and proved the fast convergence ability of the LSTM model through actual data from GEANT [11]. Although this kind of deep learning prediction model has achieved good results, the above models all predict the time series of network traffic in a single area but ignore the spatial structure of the network, that is, the spatial correlation of network traffic. To extract the spatial characteristics of network traffic, researchers introduced convolutional neural networks (CNNs) into the task of network traffic prediction. Zhang et al. used a convolutional neural network to capture the temporal and spatial dependence of traffic by processing traffic data to images. The experimental results show that the prediction performance of this method in terms of root mean square error (RMSE) is significantly improved [12]. Li et al. proposed a CNN fusion LSTM model for prediction, used a one-dimensional CNN to obtain the spatial characteristics of network traffic, and used LSTM to obtain the temporal correlation of network traffic. However, the spatial structure of the CNN model is in Euclidean space; that is, the CNN can only deal with Euclidean data, but it cannot effectively deal with non-Euclidean data such as communication network topology.

Therefore, researchers hope to effectively extract spatial features from non-Euclidean data structures such as topological maps [13], so GCNs have become a new research focus. He et al. proposed a spatial-temporal network based on graph attention, which is called GSATN. This model integrates spatial-temporal characteristics, characterizes spatial correlation through geographical relationship graphs, characterizes temporal correlation through recurrent neural networks, and predicts network traffic by combining spatiotemporal characteristics [14]. Yang et al. proposed a network traffic prediction model combining a graph convolution neural network (GCN) and a gate control recursive unit (GRU). The model uses GCN to learn network topology and extract spatial characteristics of traffic and uses GRU to learn the temporal characteristics of network traffic. Thus, the intelligent prediction of network traffic is realized [15]. Although these models have achieved excellent prediction accuracy, most models tend to extract static spatial dependencies in traffic, and such spatial dependencies may evolve over time [16, 17]. Therefore, by introducing an attention mechanism into the GCN-GRU model, this paper proposes a novel intelligent network traffic prediction method based on joint attention and GCN-GRU. This model can not only capture spatial-temporal correlation information but also collect temporal global change information. The main contributions of this paper are as follows:

(1) A network traffic prediction method combining GCN, GRU, and attention mechanism is proposed. The method uses GCN to capture the spatial features of traffic, GRU to capture the temporal features of traffic, and attention mechanism to capture the importance of different temporal features, so as to realize the comprehensive consideration of the spatial-temporal correlation of network traffic.

(2) The attention mechanism is introduced into the GRU, and the weight matrix calculation method in the GRU unit is redesigned. In this mechanism, the state vector is generated by combining the hidden states at different times, a scoring function is designed to calculate the weight of each hidden state, and an attention function is designed to calculate the context vector that can describe the global traffic change information, so as to adjust the importance of different time points and collect the global time information to improve the prediction accuracy.

(3) Considering that the length of the sliding window and the number of hidden units have a significant impact on the timeliness and accuracy of network traffic prediction, an action to determine the experimental parameters is performed, so as to obtain the optimal length of sliding window and optimal number of hidden units, which effectively supports the comparative analysis of the network traffic prediction model AGG proposed in this paper with other baseline models.

(4) The AGG model is trained on the Milan traffic network dataset for many times. The results show that compared with several existing baseline models, the AGG model has the best performance in experimental indicators, such as root mean square error (RMSE), mean absolute error (MAE), accuracy (ACC), determination coefficient (R^2), explained variance score (EVS), and has the ability of long-term prediction.

The rest of this paper is organized as follows. In Section 2, we present the problem formulation of network traffic prediction and design a framework to solve the network traffic prediction problem. Based on the design of the spatial feature extraction model, temporal feature extraction model, and attention mechanism model, a complete intelligent network traffic prediction model is given in Section 3. In Section 4, we introduce the experimental environment and analyze the performance of the proposed traffic prediction model. We conclude this paper in Section 5.

2. The Proposed Prediction Framework

2.1. Problem Formulation. The goal of network traffic prediction is to predict the network traffic information in the future according to the measured historical network traffic information. We can define this process as

$$x_{t-M+1},\ldots,x_{t-1},x_t \xrightarrow{f} (\cdot)\ \widehat{x}_{t+1},\ldots,\widehat{x}_{t+H-1},\widehat{x}_{t+H}, \tag{1}$$

where $x_t \in R^n$ is the observation vector of n observation points at the sampling time t. The purpose of the traffic prediction model is to learn a mapping function $f(\cdot)$ based on the traffic data of the previous M sampling time to predict the network traffic of the H sampling time in the future.

Definition 1 (network topology). The network is composed of nodes and links, which are generally represented by digraphs $G = (V, E)$. V represents the nodes in the network, and $V = \{V_1, V_2, \ldots, V_N\}$, where N is the number of nodes, and E represents the links between nodes. The adjacency matrix A is used to represent the connection relationship of nodes, $A \in R^{N\times N}$. The adjacency matrix only contains the elements 0 and 1. When the element is 0, there is no connection between nodes, and when the element is 1, there is a connection between nodes.

Definition 2 (network traffic prediction). In G, each link is $e_i\,(1 \le i \le n)$, and the time series $x_{t-n},\ldots,x_{t-1},x_t$ represents the network traffic of e_i in the time interval N. The principle of the prediction model proposed in this paper is to learn a mapping function f based on the topological graph structure and network traffic time series to obtain the network traffic data spatial-temporal characteristics and then predict the network traffic information $x_{t+1},\ldots,x_{t+T}$ in the future from the characteristic matrix. The network traffic prediction formula is as follows:

$$\left[x_{t+1},\ldots,x_{t+T}\right] = f\left(G,\left(x_{t-n+1},\ldots,x_{t-1},x_t\right)\right). \tag{2}$$

2.2. Traffic Prediction Framework. For the problem described in Section 2.1, the prediction architecture proposed in this paper is shown in Figure 1. First, the time series data in each region in the dataset at n time sampling points and the adjacency matrix representing the relationship between regions are taken as the input. Then, the GCN model is used to extract the input data spatial features, and the time series with spatial features are used as the input of the GRU model to extract the temporal correlation features between time series. Furthermore, the attention mechanism is introduced into GRU, and the weight matrix calculation method in the original GRU unit is replaced by the attention weight mechanism, which reweights the influence of historical network traffic data to capture the global variation trend of network traffic. Finally, the prediction results of data with spatial-temporal correlation are obtained through the fully connected layer.

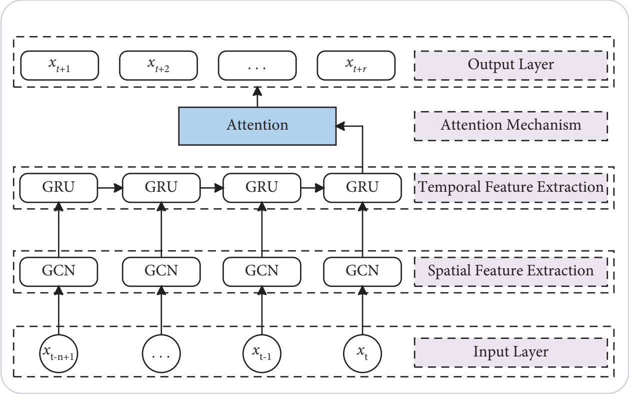

Figure 1: AGG prediction architecture.

3. Prediction Models

3.1. Spatial Feature Extraction Model. Spatial feature extraction is one of the critical problems in network traffic prediction. A regional topological network is a graph structure, and its network traffic data belong to non-Euclidean data. Although traditional convolutional neural networks (CNNs) can obtain spatial features, they can only be used in Euclidean data and cannot effectively extract spatial features from graph data. In this paper, the graph convolution network (GCN) model is used to process the non-Euclidean data represented by graph data, and the spatial features of each region are learned from the network structure.

The principle of GCN is to construct a filter in the Fourier domain and then process the graph nodes and the first-order domain of the nodes with the constructed filter to obtain the spatial features between the nodes in the graph. Finally, the GCN model is established by superposition of multiple convolution layers. In this paper, we designed two convolutional layer processing graph structures, and the formula is as follows:

$$f(X,A) = \sigma\left(\widehat{A}\mathrm{ReLU}\left(\mathrm{AXW}_0\right)W_1\right), \tag{3}$$

where X represents the network traffic characteristic matrix, A represents the adjacency matrix, $\sigma(\cdot)$ and ReLU represent the activation function. $\widehat{A} = \widetilde{D}^{-1/2}\widetilde{A}\widetilde{D}^{-1/2}$ represents the preprocessing step, $\widetilde{D}$ is the degree matrix, $\widetilde{D} = \sum_i \widetilde{A_{ij}}$. $\widetilde{A} = A + I_N$ represents the matrix with a self-connection structure, and W_0 and W_1 represent the weight matrix in the first and second convolution layers, respectively.

3.2. Temporal Feature Extraction Model. Temporal feature extraction is another critical problem in network traffic prediction. At present, the recurrent neural network (RNN) is the most widely used neural network model for processing sequence data. However, due to the defects of gradient disappearance and gradient explosion, the traditional recurrent neural network has limitations in terms of long-term prediction. The LSTM model and GRU model are variants of recurrent neural networks, which can better solve the above defects. As variants of RNN, LSTM, and GRU have the same basic principle, they both use a gate control mechanism to memorize as much long-term information as possible. In this paper, we use the GRU network unit. Compared with the LSTM unit, the GRU unit has fewer parameters. Under the premise of ensuring the prediction accuracy, it can reduce the time of model optimization.

The structure diagram of the GRU unit is shown in Figure 2, in which x_t represents the input data at time t, h_t, h_{t-1}, and h_{t+1} indicate the hidden state at different times, r_t is a reset gate, which controls the degree of information reservation or abandonment at the previous time, u_t is an update gate, which is used to control the extent to which state information of the prior moment enters the current state, c_t is the information stored at time t, and the principle of GRU is to use the hidden state of the prior moment and the input of the current moment together to obtain the network state information of the next moment. The model not only captures the current network information but also retains the change trend of historical network information and has the ability to capture temporal dependence.

3.3. Attention Mechanism Model. When capturing temporal features, we introduce an attention mechanism into GRU in this section and redesigns the weight matrix calculation method in the original GRU unit with the attention weight mechanism.

After replacing the original matrix calculation method in GRU with an attention mechanism, X_t and h_{t-1} are used to obtain the information of the reset gate r_t and update gate u_t at time t. The formulas are as follows:

$$\begin{aligned} r_t &= \sigma\left(W^k\left[X_t, h_{t-1} + b_r\right]\right), \\ u_t &= \sigma\left(W^k\left[X_t, h_{t-1} + b_u\right]\right), \end{aligned} \tag{4}$$

where W^k is the weight matrix information in the attention mechanism, X_t represents the input traffic at the current time, h_{t-1} represents the hidden state passed down from the previous time, andb_r and b_u are deviation parameters.

After obtaining the information of the reset gate r_t and update gate u_t, the reset data $h_{t-1}' = r_t \odot h_{t-1}$ can be obtained first, and then the value range of the data of h_{t-1}' and X_t can be controlled within $[-1, 1]$ through the tanh activation function. That is, the state of memorizing the current moment h' can be obtained. The formula is as follows:

$$h' = \tanh\left(W^k\left[X_t, \left(r_t \odot h_{t-1}\right)\right] + b_n\right), \tag{5}$$

where b_u is a deviation parameter.

After obtaining the current time state of memory, the last step is to update the memory stage, in which the update gate u_t is used. The formula is as follows:

$$h_t = u_t \odot h_{t-1} + \left(1 - u_t\right) \odot h\prime. \tag{6}$$

Through the multilayer GRU with attention mechanism, the temporal features of network traffic can be better captured. The internal structure of the redesigned GRU is shown in Figure 3.

3.4. Traffic Prediction Model. The network traffic prediction model, named AGG model, introduces the attention mechanism based on the GCN-GRU model and reweights the influence of historical network traffic data to capture the global variation trend in network traffic. The model structure is shown in Figure 4.

The AGG model calculation is shown in the following formulas:

$$\begin{aligned} u_t &= \sigma\left(W_u\left[GC\left(A, X_t\right), h_{t-1}\right] + b_u\right), \\ r_t &= \sigma\left(W_r\left[GC\left(A, X_t\right), h_{t-1}\right] + b_r\right), \\ c_t &= \tanh\left(W_c\left[GC\left(A, X_t\right), \left(r_t * h_{t-1}\right)\right] + b_c\right), \\ h_t &= u_t * h_{t-1} + \left(1 - u_t\right) * c_t, \end{aligned} \tag{7}$$

where u_t is the update gate which is used to control the extent to which the state information of the last time enters the state of current time, σ is the activation function of the nonlinear model, W_u, W_u, and W_u are the weight parameters, GC is the graph convolution process, A is the adjacency matrix, X_t is the input of the model at the current time, h_{t-1} and h_t are the hidden state at $t-1$ and t, respectively, b_u, b_u, and b_u are deviation parameters, r_t is the reset gate which controls the level of information retention or abandonment at the previous time, and c_t is the information stored at time t.

The AGG model is constructed by the GCN model combined with the GRU model. The principle is to input n historical time series network traffic data into the AGG model to obtain n hidden states and obtain the vector containing spatial-temporal features: $\left\{h_{t-n+1}, \ldots, h_{t-1}, h_t\right\}$.

Then, the hidden state is inputted into the attention model, and the multilayer perceptron (MLP) is used to calculate the weight of each hidden state h: $\left\{a_{t-n+1}, \ldots, a_{t-1}, a_t\right\}$. The information vector covering the global traffic change is calculated by the sum of the weights. The formulas are as follows:

$$\begin{aligned} a_i &= \frac{\exp\left(e_i\right)}{\sum_{k=1}^{n}\exp\left(e_k\right)}, \\ e_i &= W_{(2)}\left(W_{(1)}H + b_{(1)}\right) + b_{(2)}. \end{aligned} \tag{8}$$

Then, an attention function is used to describe the vector C_t of global traffic change information, and the formula is as follows:

$$C_t = \sum_{i=1}^{n} a_i * h_i. \tag{9}$$

Finally, the final predicted value is obtained through the fully connected layer.

4. Simulation Results and Analysis

In this part, we first introduce the actual traffic dataset of the telephone service provider in the European city of

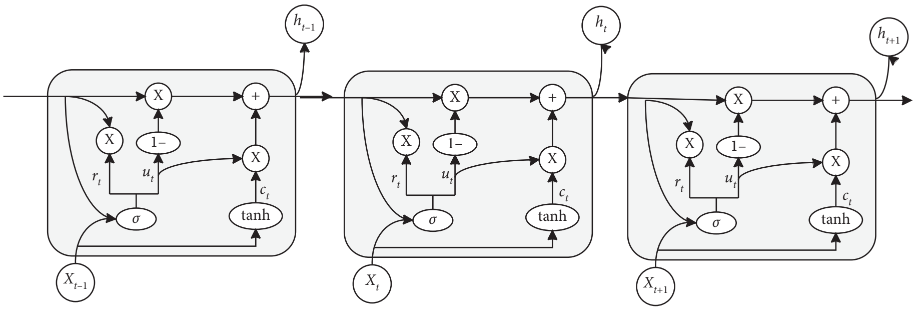

Figure 2: Schematic diagram of the GRU structure.

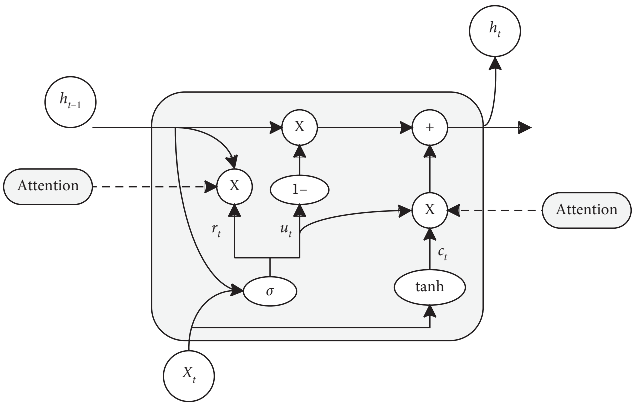

Figure 3: Internal structure of the GRU after redesign.

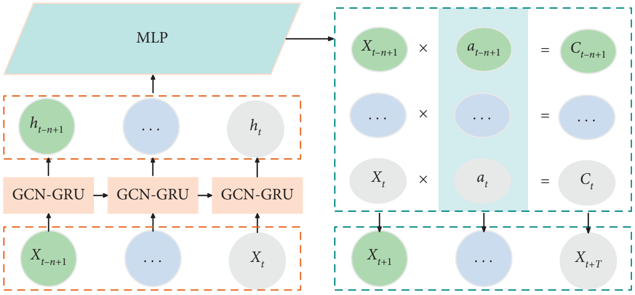

Figure 4: AGG model.

Milan and then analyze comparative experiments based on this dataset to verify the advantages of our proposed model.

4.1. Dataset Description. In this paper, we select an open network traffic dataset which is in https://dataverse.harvard.edu/dataset.xhtml?persistentId = doi:10.7910/DVN/EGZHFV, and the traffic collection time is from 00 : 00 on November 1, 2013, to 00 : 00 on January 1, 2014. Table 1 shows the relevant dataset information. In this experiment, the data of 11/04-11/10 for seven days are selected as the dataset. The time interval of the original data is 10 minutes, and there are 144 data points in each region. In this paper, nine regions are selected, and the data of a week are collected. The grid and map of the area where the dataset is located are shown in Figure 5. Figure 6 shows the network traffic trend of the nine regions within a week.

4.2. Experimental Indicators. In order to thoroughly verify the performance of the model, we set five experimental indicators to judge the flow prediction model proposed in this paper, as follows:

(1) Root mean square error (RMSE) reflects the prediction error of the model. The value range of RMSE is $[0, +\infty)$. The closer the RMSE is to zero, the better the performance of the model is.

$$\text{RMSE} = \sqrt{\frac{1}{T}\sum_{t=1}^{T}\left(Y^{t} - \widehat{Y}^{t}\right)}. \qquad (10)$$

(2) Mean absolute error (MAE) is used to measure the mean absolute error between the predicted value and the true value. The value range of MAE is $[0, +\infty)$. The closer the MAE is to zero, the better the performance of the model is.

$$\text{MAE} = \frac{1}{T}\sum_{t=1}^{T}\left|Y^{t} - \widehat{Y}^{t}\right|. \qquad (11)$$

(3) Accuracy (ACC) reflects the prediction accuracy of the model. The value range of ACC is $[0, 1]$. The closer the ACC is to 1, the better the performance of the model is.

$$\text{Accuracy} = 1 - \frac{\sum_{t=1}^{T}\left|Y^{t} - \widehat{Y}^{t}\right|}{\sum_{t=1}^{T} Y^{t}}. \qquad (12)$$

(4) Determination coefficient (R^2) represents the quality of model fitting. The value range is $[0, 1]$. The closer the R^2 is to 1, the better the model fits the data.

$$R^{2}(Y, \widehat{Y}) = 1 - \frac{\sum_{t=1}^{T}\left(Y^{t} - \widehat{Y}^{t}\right)^{2}}{\sum_{t=1}^{T}\left(\overline{Y^{t}} - \widehat{Y}^{t}\right)^{2}}. \qquad (13)$$

(5) Explained variance score (EVS) is the variance score of the model. The value range is $[0, 1]$. The closer the EVS is to 1, the better the independent variable can explain the variance change of the dependent variable.

Table 1: Dataset.

Dataset	Milan telephone service provider
City	Milan
Time span	2013/11/01–2014/01/01
Time interval	10 minutes
Grid size	(100,100)

$$\mathrm{EVS} = 1 - \frac{\sum_{t=1}^{T} \mathrm{Var}\left\{Y^t - \widehat{Y}^t\right\}}{\sum_{t=1}^{T} \mathrm{Var}\left\{Y^t\right\}}, \tag{14}$$

where Y^t denotes the actual value of traffic data at the time t and $\widehat{Y}^t$ denotes the predicted value of traffic data at the time t. $\overline{Y^t}$ denotes the mean value of traffic data, and T is the number of samples.

4.3. Experimental Parameters. In this experiment, we use a deep learning server to configure the experimental environment, in which the production type of CPU is AMD Ryzen 52600, the production type of GPU is Nvidia GT745 M, the size of Memory is 16 GB. In addition, TensorFlow is used to build the network framework and *Python* is used as the programming environment. Table 2 lists the detailed environment configuration parameters.

Further, we need to determine the model training parameters. In this experiment, Adam is chosen as the optimizer, the learning rate is set to 0.001, and the epoch for model training is 3000. As for the selection of the sliding window length and the number of hidden units, theoretically, on the one hand, the larger the sliding window length is, the larger the perception range will be, and the more features will be predicted, which may cause some interference to the accuracy of prediction. On the other hand, when the number of hidden units increases to a certain extent, the complexity and difficulty of model calculation will also increase, and the accuracy of prediction will also decrease.

Considering that the sliding window length L and the number of hidden units H have a significant impact on the timeliness and accuracy of the traffic prediction, we compared ACC and R^2 under different L and H and obtained the optimal sliding window length and the number of hidden units under the current configuration.

Specifically, the optional range of sliding window length L is set to [4,8,12,16], and by comparing the prediction performance under different L conditions in Figure 7, we obtain the optimal sliding window length, which is 8. That is, we use 8 historical network traffic data $(X_{t-7}, X_{t-6}, X_{t-5}, X_{t-4}, X_{t-3}, X_{t-2}, X_{t-1}, X_t)$ to predict future traffic. Similarly, the optional range of the number of hidden units H is set to [32,64,100,128], and by comparing the prediction performance under different H conditions in Figure 8, we obtain the optimal number of hidden units, which is 100.

In conclusion, when the sliding window length is set to 8 and the number of hidden units is set to 100, the prediction result is optimal. Therefore, the model training parameters containing the above results are listed in detail in Table 3.

4.4. Result Analysis

4.4.1. Comparison Results between AGG Model with Other Baseline Models. To verify the performance of AGG model, 80% of traffic data are selected as the training dataset, and 20% of traffic data are selected as the verification dataset. The comparison indicators are described in Section 4.2. In addition, five baseline models are selected including model-driven methods and data-driven methods to compare with the model proposed in this paper. The comparison results are listed in Table 4; because the sampling interval of the traffic data is 10 minutes, we use 10 minutes (one point) and 20 minutes (two points) to carry out single-step prediction and multistep prediction, respectively.

(1) Historical average model (HA), which models network traffic as a periodic process to predict the time series

(2) An autoregresive moving composite average model (ARIMA), which is used to fit the time series into a parameter model for completing the network traffic prediction

(3) Support vector machine model (SVR), which adopts the machine learning algorithm and uses historical data to fit the relationship between input and output and then predicts future network traffic data

(4) Gated recurrent unit (GRU), which is an efficient solution to the gradients vanishing issue after a long sequence of inputs

(5) GCN-GRU, which is a combination model combining a graph convolution neural network (GCN) and a gate control recursive unit (GRU)

Table 4 shows that the experimental indicators of the AGG model proposed in this paper are significantly better than those of other baseline models. To be specific, we have the following:

(1) At the 10 min prediction span, the AGG model proposed in this paper has optimal values in RMSE, MAE, ACC, R^2, and EVS. For example, the RMSE of the AGG model is 3.7% lower than that of the GCN-GRU model, 4.2% lower than that of the GRU model, 5.5% lower than that of the SVR model, 6.3% lower than that of the ARIMA model, and 14.7% lower than that of the HA model. The ACC of the AGG model is 1.5% higher than that of the GCN-GRU model, 2% higher than that of the GRU model, 2.3% higher than that of the SVR model, 15.9% higher than that of the ARIMA model, and 6.9% higher than that of the HA model. The AGG model proposed in this paper has optimal values in RMSE, MAE, ACC, R2, and EVS. It can be further seen that both AGG and GRU are superior to model-driven traffic prediction methods.

(2) At the 20 min prediction span, the AGG model proposed in this paper still has optimal values in RMSE, MAE, ACC, R^2, and EVS. For example, the RMSE of the AGG model is 1.6% lower than that of

9901	9902	. . .	9999	10000
9801	. . .	. . .	9899	9900
. . .	. . .	. . .	. . .	. . .
101	102	. . .	. . .	200
1	2	3	. . .	100

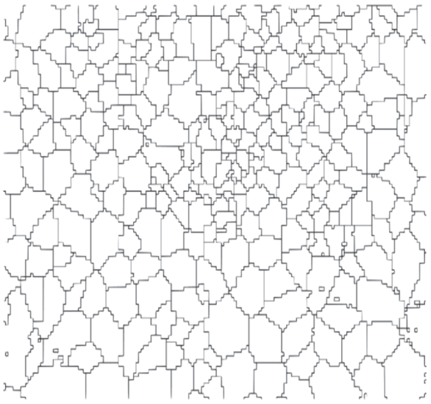

Figure 5: Grid and map of the area.

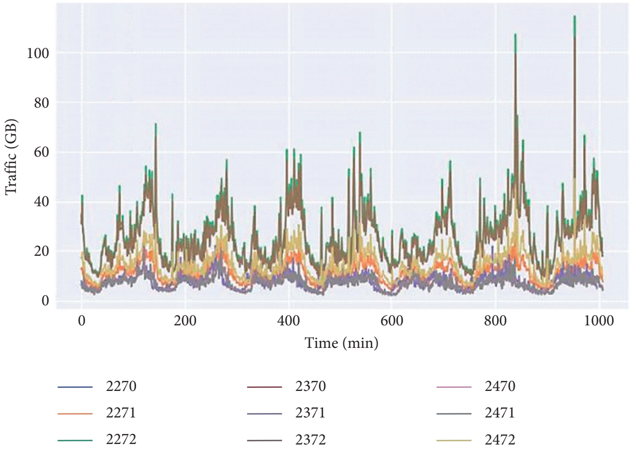

Figure 6: Network traffic trend of the nine regions within a week.

Table 2: Environment configuration parameters.

Environment component	Parameter
CPU production type	AMD Ryzen 52600
GPU production type	Nvidia GT745 M
Memory size	16 GB
TensorFlow version	1.4
Python version	3.7

the GCN-GRU model, 1.7% lower than that of the GRU model, 1.9% lower than that of the SVR model, 2.5% lower than that of the ARIMA model, and 7.4% lower than that of the HA model. The prediction accuracy of the AGG model is 0.7% higher than that of the GCN-GRU model, 1.8% higher than that of the GRU model, 2.5% higher than that of the SVR model, 12.2% higher than that of the ARIMA model, and 3.5% higher than that of the HA model.

(3) It can be further concluded from the prediction results that, in horizontal comparison, the data-driven prediction methods, whether SVR or GRU, are better than other model-driven methods. This result is due to the poor fitting ability of HA and ARIMA for this long series of unstable data, while the neural network models fit the nonlinear data much better. In longitudinal comparison, the performance indicators of the AGG model proposed in this paper decrease with the increase of prediction time, but the decline trend is relatively stable, and it still has long-term prediction ability.

4.4.2. Influence of Spatial-Temporal Correlation and Attention Mechanism on Prediction Performance. In order to further explore the influence of spatial-temporal correlation and attention mechanism on prediction performance, two experimental indicators, RMSE and ACC, are used to compare AGG model with other baseline models at the 10 min prediction scale, and the comparison results are shown in Figures 7 and 8, respectively.

Figure 9 shows the comparison results of RMSE between AGG model and other baseline models. These baseline models include model-driven traffic prediction methods HA and ARIMA, and data-driven traffic prediction methods SVR, GRU, and GCN-GRU. Specifically, RMSE of model-driven traffic prediction method are 6.1774 (HA) and 5.6241 (ARIMA) respectively, and RMSE of data-driven traffic prediction method are 5.5817 (SVR), 5.4932 (GRU), 5.4761 (GCN-GRU), and 5.2721 (AGG), respectively. Therefore, RMSE on the whole presents a downward trend, and the AGG model proposed in this paper has the smallest RMSE, which means that the model of spatial-temporal correlation and the introduction of an attention mechanism are fundamental to reduce the RMSE of network traffic prediction results.

Figure 10 shows the comparison results of ACC between AGG model and other baseline models. These baseline models are consistent with Figure 9. Specifically, ACC of model-driven traffic prediction method are 0.6785 (HA) and 0.6264 (ARIMA), respectively, and ACC of data-driven traffic prediction method are 0.7095 (SVR), 0.7114 (GRU), 0.7150 (GCN-GRU), and 0.7256 (AGG), respectively. Therefore, ACC on the whole presents an upward trend, and the AGG model proposed in this paper has the largest RMSE, which means that the model of spatial-temporal correlation and the introduction of an attention mechanism are significant to improve the ACC of network traffic prediction results.

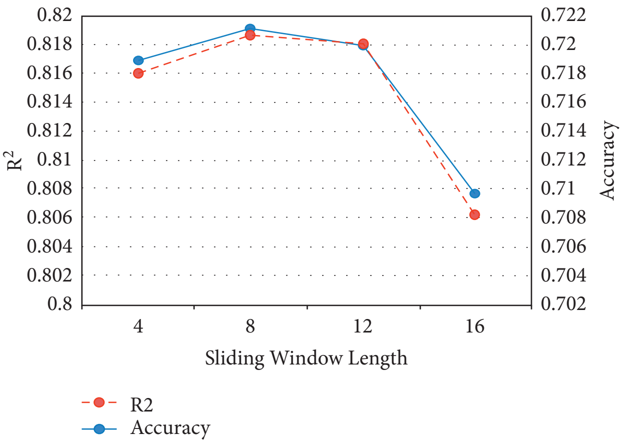

Figure 7: Performance comparison under different sliding window lengths.

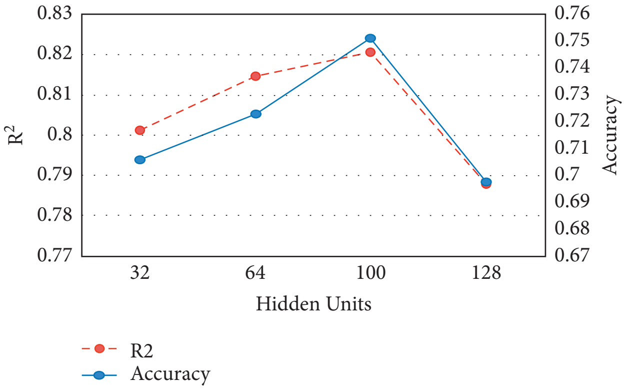

Figure 8: Performance comparison under different number of hidden units.

Table 3: Model training parameters.

Model component	Parameter
Batch size	32
Learning rate	0.001
Training epoch	3000
Sliding window length	8
Hidden units	100

4.4.3. Analysis of Visual Results of Traffic Prediction. In order to more intuitively see the prediction results of the proposed AGG model, Figures 11 and 12, respectively, show the traffic trend comparison diagram between the prediction value and the true value of AGG model in 10 min and 20 min prediction spans of area 2270. In the experiment, the sliding window length is set to 8 and the number of hidden units is set to 100, which has been proved in Section 4.3 that these parameters are optimal.

It can be seen from Figures 11 and 12 that the AGG model proposed in this paper has good prediction performance, but it has the following two flaws. On the one hand, the prediction result of network traffic at the peak is poor. The main reason is that the GCN model defines a smoothing filter in the Fourier domain and captures the spatial characteristics by continuously moving the filter and signal for winding operation. This process leads to smoother prediction of the mutation region. On the other hand, there is a certain error between the true network traffic data and the prediction results. The possible reason is that when there is no communication at a certain time in the region, the value of network traffic may be zero, or the value of network traffic may be very small, and a small difference may cause a large relative error. Further, by comparing Figures 11 and 12, we can also get that, with the increase in the prediction time scale, the fitting level between the prediction value and the actual value also decreases, indicating that the small prediction scale always has a better prediction effect.

Table 4: Comparison results between AGG model with other baseline models.

T (min)	Metric	HA	ARIMA	SVR	GRU	GCN-GRU	AGG
10	RMSE	6.1774	5.6241	5.5817	5.4932	5.4761	5.2721
	MAE	3.2447	3.9729	2.8332	2.7525	2.7009	2.5788
	ACC	0.6785	0.6264	0.7095	0.7114	0.7150	0.7256
	R^2	0.7613	0.0356	0.8051	0.8078	0.8124	0.8261
	EVS	0.7613	0.0008	0.8065	0.8086	0.8125	0.8265
20	RMSE	6.1774	5.8655	5.8321	5.8211	5.8189	5.7209
	MAE	3.2447	3.9767	2.9926	2.9836	2.9732	2.8362
	ACC	0.6785	0.6262	0.6856	0.6898	0.6973	0.7024
	R^2	0.7613	0.0356	0.7749	0.7853	0.7888	0.7959
	EVS	0.7613	0.0009	0.7896	0.7914	0.7931	0.7969

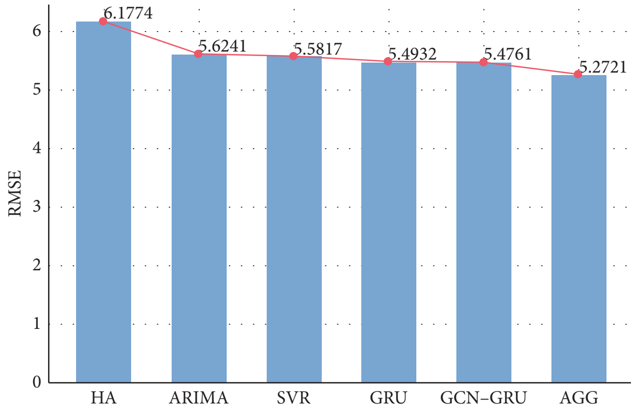

Figure 9: Comparison results of RMSE between AGG model and other baseline models.

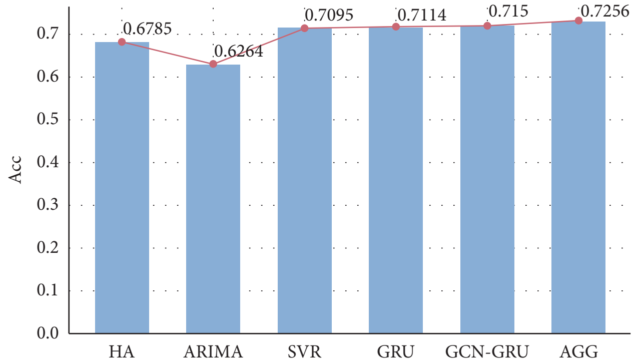

Figure 10: Comparison results of ACC between AGG model and other baseline models.

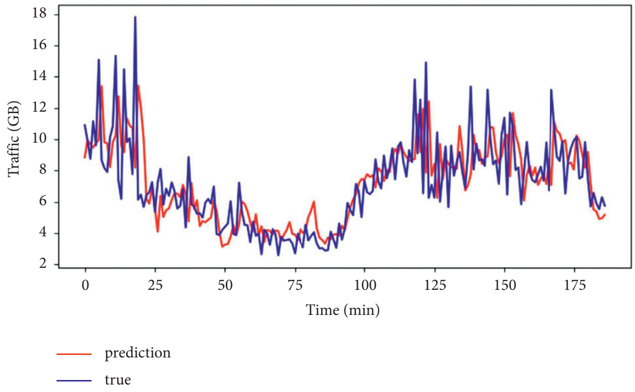

Figure 11: Visual results of traffic prediction by AGG model in 10 min span.

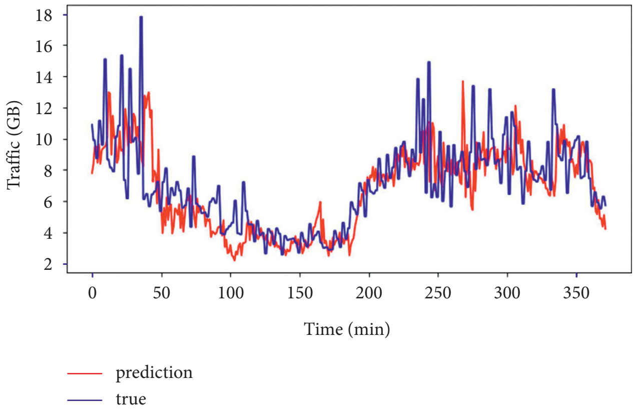

Figure 12: Visual results of traffic prediction by AGG model in 20 min span.

5. Conclusion

In this paper, we propose a network traffic prediction method combining GCN, GRU, and attention mechanism. In this method, GCN is used to capture the network topology to obtain the spatial features of network traffic. GRU model is used to capture the dynamic changes of traffic on nodes, so as to obtain the time features of network traffic. Furthermore, the attention mechanism is used to weight the historical traffic data to dynamically adjust the importance of network traffic information at each sampling time. By using the actual network traffic dataset to carry out the experiment and comparing it with the baseline models such as HA, ARIMA, SVR, GRU, and GCN-GRU, it can be concluded that the AGG model proposed in this paper achieves the best prediction effect under different performance indicators.

Data Availability

This paper selects an open network traffic dataset, the download address is https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/EGZHFV, and the traffic collection time is from 00 : 00 on November 1, 2013, to 00 : 00 on January 1, 2014.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grants 61801073, 61722105, and 61931004.

References

[1] Cisco: Cisco annual internet report (2018–2023) whitepaper [R/OL]. Cisco (2020-03-09) [2020-05-15]. https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internetreport/white-paper-c11-741490.html.

[2] M. Laner, P. Svoboda, and M. Rupp, "Parsimonious fitting of long-range dependent network traffic using ARMA models," *IEEE Communications Letters*, vol. 17, no. 12, pp. 2368–2371, 2013.

[3] J. Guo, Y. Peng, X. Peng, and Q. Chen, "Traffic forecasting for mobile networks with multiplicative seasonal ARIMA models," in *Proceedings of the Electronic Measurement & Instruments, 2009. ICEMI '09. 9th International Conference on*, August 2009.

[4] H.-W. Kim, J.-H. Lee, Y.-H. Choi, Y.-U. Chung, and H. Lee, "Dynamic bandwidth provisioning using ARIMA-based traffic forecasting for Mobile WiMAX," *Computer Communications*, vol. 34, no. 1, pp. 99–106, 2011.

[5] A. Alheraish, S. Alshebeili, and T. Alamri, "Regression video traffic models in broadband networks," *Journal of King Saud University - Engineering Sciences*, vol. 18, no. 1, pp. 19–55, 2005.

[6] Y. Shu, M. Yu, j. Liu, and O. W. W. Yang, "Wireless traffic modeling and prediction using seasonal arima models," *IEICE - Transactions on Communications*, vol. E88B, no. 10, pp. 1675–1679, 2003.

[7] P. Bermolen and D. Rossi, "Support vector regression for link load prediction," *Computer Networks*, vol. 53, no. 2, pp. 191–201, 2009.

[8] G. Kremer, P. Owezarski, and P. Berthou, *Predictive Estimation of Wireless Link Performance from Medium Physical Parameters Using Support Vector Regression and K-Nearest Neighbors*, Springer, Berlin, Heidelberg, 2014.

[9] Y. Wu, H. Tan, L. Qin, B. Ran, and Z. Jiang, "A hybrid deep learning based traffic flow prediction method and its understanding," *Transportation Research Part C: Emerging Technologies*, vol. 90, no. 9, pp. 166–180, 2018.

[10] A. Lazaris and V. K. Prasanna, "An LSTM framework for modeling network traffic," in *Proceedings of the 2019 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, IEEE, Washington DC, USA, April 2019.

[11] A. Azzouni and G. Pujolle, "A long short-term memory recurrent neural network framework for network traffic matrix prediction," arXiv preprint arXiv:1705, 2017.

[12] C. Zhang, H. Zhang, D. Yuan, and M. Zhang, "Citywide cellular traffic prediction based on densely connected convolutional neural networks," *IEEE Communications Letters*, vol. 22, no. 8, pp. 1656–1659, 2018.

[13] L. Zhao, Y. Song, C. Zhang, and Y. Liu, "T-GCN: a temporal graph convolutional network for traffic prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 99, pp. 1–11, 2019.

[14] K. He, Y. Huang, X. Chen, and Z. Zhou, "Graph attention spatial-temporal network for deep learning based mobile traffic prediction," in *Proceedings of the GLOBECOM 2019 - 2019 IEEE Global Communications Conference*, December 2019.

[15] L. Yang, X. Gu, and H. Shi, "A noval satellite network traffic prediction method based on GCN-GRU," in *Proceedings of the 2020 International Conference on Wireless Communications and Signal Processing (WCSP)*, Xi'an, China, October 2020.

[16] J. Zhu, Y. Song, L. Zhao, and H. Li, "A3T-GCN: attention temporal graph convolutional network for traffic forecasting," arXiv:2006.11583, 2020.

[17] H. Zhang, L. Chen, J. Cao, and X. Zhang, "A combined traffic flow forecasting model based on graph convolutional network and attention mechanism," *International Journal of Modern Physics C*, 2021.