Gated Recurrent Unit Network-based Cellular Traffic Prediction

Edmund Pok Leng Shiang
Department of Electrical Engineering
National Dong Hwa University
Hualien, Taiwan, R.O.C
a0979632623@gmail.com

Wei-Che Chien
Department of Engineering Science
National Cheng Kung University
Tainan, Taiwan, R. O. C.
b9944006@gmail.com

Chin-Feng Lai
Department of Engineering Science
National Cheng Kung University
Tainan, Taiwan, R. O. C.
cinfon@ieee.org

Han-Chieh Chao*
Department of Electrical Engineering
National Dong Hwa University
Hualien, Taiwan, R.O.C
hcc@niu.edu.tw

***Abstract*—With the development of 5G and big data, network traffic is growing exponentially year by year. Effective computing resource allocation and network traffic control become an increasingly important issue. If mobile network operators still use traditional traffic control strategies and methods, it will not be able to process heavy and flexible traffic and will easily lead to high packet loss and poor Quality of Service(QoS). With the improvement of computing power of hardware devices, neural networks have been widely used in many fields. A long short-term memory (LSTM) neural network is one of the famous methods for solving time series data. However, LSTM usually needs to spend a lot of training time and computing resources because of its internal structure. In mobile networks, traffic forecasting is a real-time problem. Therefore, we proposed a Gated Recurrent Unit (GRU) based model to predict the traffic of the base station. GRU replaces the forget gate and input gate in the LSTM with an update gate and combines the cell state and the hidden state to reduce the complexity of the architecture. The simulation results show that the proposed GRU-based model has better traffic prediction performance than LSTM and can greatly reduce training time.**

Keywords— LSTM, GRU, Traffic Forecast, 5G

I. Introduction

The 5th Generation Mobile Networks (5G) is expected to be available in 2020. The number and variety of terminal devices will increase year by year. In the 5G environment, there are various network devices, including User Equipment (UE), driverless cars, Internet of Things (IoT), etc. Based on the high reliability and low latency of 5G, it is accompanied by the development of many network applications, such as 8K image transmission, remote surgery, smart factory, smart city, etc. These applications have caused a dramatic increase in network traffic. If mobile network operators continue to use traditional traffic control policies and methods, they may not be able to process with heavy and diversified traffic, which may result in high packet loss and poor Quality of Experience (QoE).

Big data prediction is the foundation for intelligent management functions and is very valuable for a variety of different network applications [1]. For changes in the 5G environment, it is necessary to actively solve the problem of network traffic instead of responding passively as traditional methods [2]. If the network traffic changes can be predicted, resource allocation and network planning can be better prepared and controlled in advance. Therefore, predicting network traffic is very important in the 5G environment. Machine learning has been widely used for time series analysis problems. [3] uses artificial neural networks to predict base station traffic on different wireless network settings. [4] uses an artificial neural network to predict network traffic with time-series features. The simulation results show that the prediction performance of the Recurrent Neural Network (RNN) is better than Stacked Autoencoder (SA).

The traffic data of the network has long-term time series dependence. For example, in a general science park, most users use the mobile network from 9:00 am to 5:00 pm. For some areas, activities will be held at some fixed months, resulting in a significant increase in network traffic. LSTM has better performance in terms of long-term time series prediction [2][5]. LSTM will review historical data and find out the hidden rules in the data. It has the mechanism of selecting memory and forgetting memory to process long-term and regular big data. However, there are some problems with LSTM. Since the model structure of LSTM is complexity, it will lead to long training time when there are many parameters to be processed. Based on the need for immediacy of network traffic prediction, we use GRU (Gated Recurrent Unit) to predict base station traffic.

Both GRU and LSTM are a new model based on the structure of RNN. Comparing GRU and LSTM, the parameters of GRU are relatively less. GRU replaces the Forget Gate and Input Gate in the LSTM with an Update Gate and combines the Cell State and the hidden state to reduce the complexity of the architecture. In order to analysis the features of historical traffic data so as to accurately and fast predict future traffic, in this study, we first set up a GRU model and try to optimize the important parameters through training data. Then, we compare the results with LSTM.

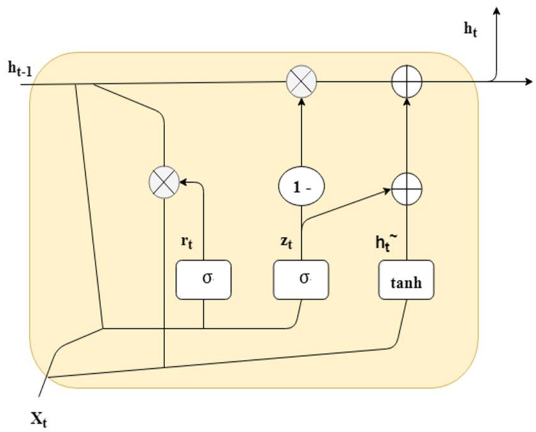

Fig. 1: Structure of the GRU

The remainder of this paper is organized as follows: Section II introduces some related researches on the neural network to solve time series problems. Section III introduces the problem definition. The proposed training model is shown in section IV. Section V presented the simulation results. Some discussions are given in section VI. Finally, the conclusion and future work are shown in section VII.

II. Related work

With the technology of 5G and IoT, there will be a large number of devices connected to the Internet in the future. The monitoring of traffic has become a very important issue. In recent years, due to the improvement of hardware devices, neural networks has begun to develop in a breakthrough. Many scholars use neural networks to deal with various optimization problems, such as image processing, natural language processing, and time series prediction problems. [6] proposed a machine learning technique based on low computational complexity, using neural networks. It was optimized to build a modeling framework for 5G service provision. The neural network was trained to use offline integer linear programming models to make the best decisions. Many scholars use machine learning methods to find out the feature of the data and make predictions.

At present, the machine learning methods for traffic prediction can be divided into two ways. The first is to use time series data as feature data. [2] pointed out that for the problem of traffic congestion, the traditional solution can only be passive and cannot be resolved actively. Therefore, authors adopt LSTM to predict the traffic load of the base station in the future 5G Ultra Dense Network environment. Through predicting traffic, we can know in advance when network congestion may occur and then solve the problem ahead of time.

The second way of network traffic prediction is to use image processing to find out the spatial features that may exist, and then combine time series data for prediction. [7] simulated the dependence of spatial and temporal characteristics on traffic flow. The results show that both these features do affect traffic dependence. Some scholars use Convolution Neural Network (CNN) to obtain spatial characteristics between the base stations. [8] proposed a method based on big data, using deep learning technology for space-time modeling and prediction in cellular networks. The CNN is used to process spatial features of pictures and LSTM is adopted to predict time series data. However, the use of spatial features needs a lot of spatial features and computing resources.

At present, LSTM is mostly used for time series problems with long-term dependence. Although the prediction of LSTM for base station traffic does have very good performance, it requires a lot of computing time and processing resources. In order to meet the immediate forecast of traffic, this study cites the GRU to predict the traffic of the base station because the structure of GRU is relatively simple to LSTM. Therefore, it can reduce a lot of processing time and computing resources.

III. Problem Definition

In order to accurately predict the network traffic of the base station, this study uses GRU for traffic prediction. The GRU architecture are described below.

A. Gated Recurrent Unit（GRU）

Gated Recurrent Unit（GRU）is an improved version of the Recurrent Neural Network (RNN) . GRU was proposed in 2014 by Junyoung Chung, Caglar Gulcehre, Kyung Hyun Cho and Yoshua Bengio [9]. GRU can process long-term dependencies problems because it remember the information gained in the previous steps of the learning process. The structure diagram of the GRU is shown in Fig. 1, r represents reset gate, z represents update gate, X represents input , ht-1 is output of previous step , ht represents current output. There are some equations for GRU as follows:

$$r_t = \sigma(W_r * [h_{t-1}, x_t]) \ , \tag{5}$$

Equation (5) is the output equation for the reset gate. It can be used to judge the influence of traffic results in the last period on traffic at this time. It controls the output of previous step h_{t-1} to affect the current input data (x_t) . If the output (r_t) is equal to one, it means that h_{t-1} has a great influence on x_t , otherwise it has no effect.

The historical flow output and current input at this time may have an effect on the output. Equation (6) is the output equation for the update gate that is used to determine the updated content or degree of activation of unit. When the output is equal to one, the output of the previous step has no effect on the current output, otherwise, it will have a great impact.

$$z_t = \sigma(W_z * [h_{t-1}, x_t]), \tag{6}$$

Equation (7) is to activate the implicit output but not the final output. The implicit output at this time will be affected by the output of the reset gate.

$$\tilde{h}_t = \tanh(W * [r_t * h_{t-1}, x_t]) \ , \tag{7}$$

The current traffic prediction output will be affected by the current implicit output and the previous output. Equation (8) is used to calculate the final output, which is determined by the previous output and the current implicit output. When the value of the update gate is close to zero, it indicates that the data characteristics of the previous step are very important, and vice versa.

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t \ \ , \tag{8}$$

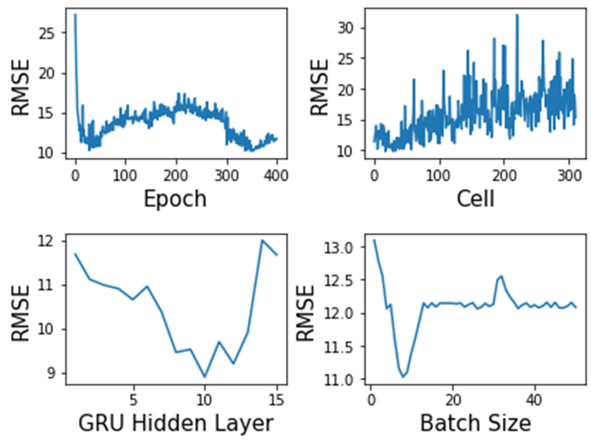

Fig. 3 Optimized parameter curve

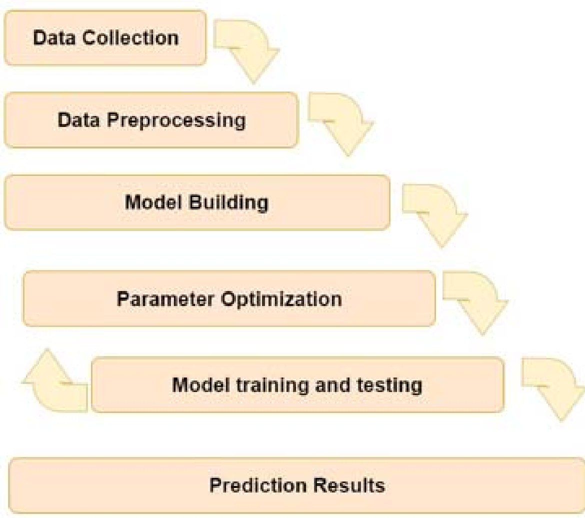

Fig. 2 System flow chart

There are some important parameters as follows:
r_t is result of reset gate, z_t is result of update gate, σ is sigmoid activation function , W_r, W_z, W are the weight value, $\tilde{h}_t$ is the current implied state, h_{t-1} is the previous output.

IV. MODEL TRAINING

Fig. 2 is a system flow chart of this study, including data collection, data preprocessing, model building, parameter optimization, model training and testing, and prediction results.

The first step is to preprocess the data. We normalize the data between 0 and 1 by using feature scaling, which has the advantage of speeding up the gradient to find the best solution speed and improving accuracy.

The GRU input vector corresponds to information of any time step t and the GRU output vector corresponds to information of time step t+1. For GRU training, we divide the data set into three parts, including the training set, the validation set, and the test set. The validation set is used to optimize various parameters of the GRU model; The test set is used to verify the performance of the designed GRU model. First, we will optimize the GRU model and then predict future traffic loads based on historical traffic data.

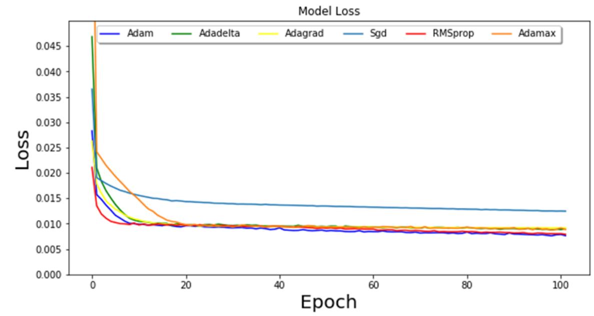

Fig. 4 Comparison of optimization algorithms

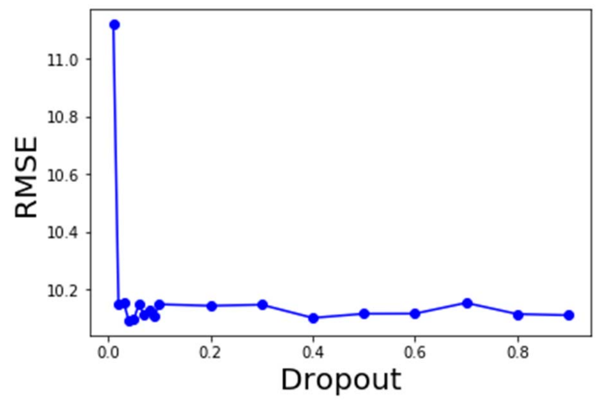

Fig. 5 Dropout adjustment chart

In order to optimize the GRU model, we performed a series of parameter tests on the GRU model, including number of hidden layers, cells, batches, and epochs to improve accuracy. Our goal is to find the best combination of parameters to minimize the loss value.

The epoch means that data set is trained by the neural network once. The size of epoch will affect the number of updates to the weights. Too few epoch may not be optimized for weighting; too many epoch may cause overfitting problems.

The size of batch determines how much time is spent after running an epoch. If the size of batch is larger, it can reduce the training time. If the batch is too small, the epoch will be too small, resulting in insufficient weight updates and insufficient machine training.

Cell is the time step of GRU, which will affect the accuracy and training time of the output. The number of hidden layers also has an impact on the accuracy and speed of the output. The most suitable parameters may be different for different problems.

The GRU process as follows:

Step 1: We optimize the batch size. The batch size is the number of training instances used in each iteration. The weight is updated after each batch propagation.

Step 2: We first optimize the number of epochs and fix other parameters. The number of epoch determines the maximum number of passes of the training data set.

Step 3: We optimize the number of neurons to determine the optimal number of neurons and achieve optimal traffic prediction.

Table 1 Parameters setting

	Number of Epoch	Size of Batch	Number of Cell	Number of Hidden layer
GRU	352	9	20	10

Step 4 : Finally, since too many hidden layers will lead to a lot of training time, we find a suitable number of hidden layers. The best parameters of each parameter are shown in Fig. 3.

The optimizer is used to adjust the weight and offset of the model so that loss can be minimized. It is very important to choose the appropriate optimizer, which can make our model closer to the actual situation Different optimizers have different strengths and weaknesses. Therefore, it needs to be selected based on the model and data. We compared six optimizers, including Adam, Sgd, RMSprop, Adamax, Adadelta, Adagrad. Finally, we found that Adam is the best optimizer for our problem, as shown in Fig. 4

Deep learning neural networks usually contain multiple nonlinear hidden layers. When the model structure is too large and too complex, the training data may be limited, resulting in overfitting. To avoid overfitting, we set the data drop ratio (Dropout) to solve this problem. The neural network node will return the output to zero randomly according to the Dropout. Different dropouts can also lead to different results, so we test the appropriate dropout by experiment, as shown in Fig. 5. Based on the experimental results, the dropout for this study was set to 0.04.

V. RESULTS AND COMPARISON

We evaluate the performance of our proposed GRU-based traffic prediction method by using Python as a simulation tool. This model is based on frameworks of Keras and TensorFlow. The hardware we use includes an Intel Core i7, 3.4 GHz CPU with 12 GB of RAM.

This paper use the cellular traffic dataset [10] as test data which is from the European telephone service provider, Telecom Italia. The data set was collected in Milan between November 10, 2013, and January 1, 2014. The data collection interval is 10 minutes. We split the data into 80% training data and 20% test data and adjust the parameters setting of GRU model according to the experiment results (Fig. 3). The detail parameters setting is shown in Table 1.

We use three commonly used performance metrics to evaluate the performance of the proposed method, including the Mean Absolute Error (MAE), the Mean Absolute Percentage Error (MAPE), and the Root Mean Square Error (RMSE).

$$\mathrm{MAE} = \frac{1}{n}\sum_{t=1}^{n} |y_i - y_i'| \quad , \tag{9}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (|y_i - y_i'|)^2} \quad , \tag{10}$$

$$\mathrm{MAPE} = \frac{1}{n}\sum_{i=1}^{n} \left|\frac{y_i - y_i'}{y_i}\right| \quad , \tag{11}$$

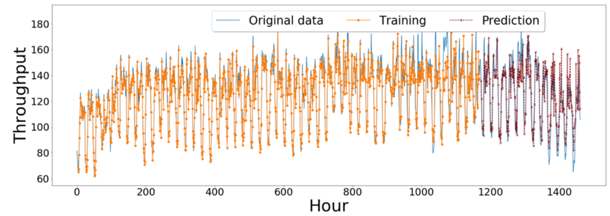

Fig. 6 GRU prediction results

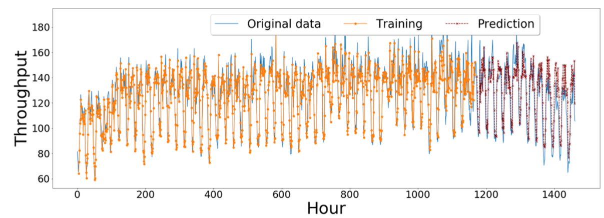

Fig. 7 LSTM prediction results

Table 2 Comparison of experimental results

	MAE	RMSE	MA（%）	TIME（s）
GRU	5.79	8.89	95.28%	717
LSTM	7.82	10.14	93.69%	846

Mean accuracy (MA), which is derived from MAPE, is used for accurate measurement.

$$\mathrm{MA} = (1 - MAPE) \times 100\% \, , \tag{12}$$

We use the best parameters to be adjusted before and train the model. Fig. 6 shows the experimental results of the GRU prediction. Blue line is the original traffic data, the orange line is the training data, and the red line is the prediction. The experimental results show that GRU can accurately predict the trend of traffic because the update gate of GRU can solve the problem of long-term dependence. Therefore, it can get very good results for this regular time series problem.

Since the process of users using the network is continuous, we also use the LSTM method to predict cellular network traffic, as shown in Fig. 7. LSTM aims to solve long-term dependency problems. Therefore, it also has great performance in this research.

The detailed comparison of experimental results between LSTM and GRU is shown in Table 2. The prediction results of GRU and LSTM are similar but training time is quite different. We conducted a comprehensive comparison of GRU and LSTM to evaluate the performance of the two models. We fixed all the parameters and compared the training results of the hidden layers from the first layer to the fifteenth layer. Fig. 8 is a comparison of MAE, Fig. 9 is a comparison of RMSE, Fig. 10 is a comparison of MA, and Fig. 11 is a comparison of training time.

According to the experimental results, it can be found that the performance between GRU and LSTM is similar, and their

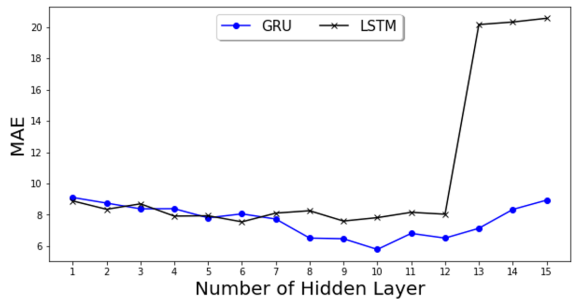

Fig. 8. Comparison of MAE

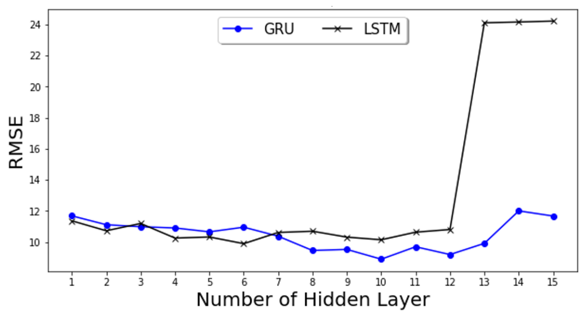

Fig. 9 Comparison of RMSE

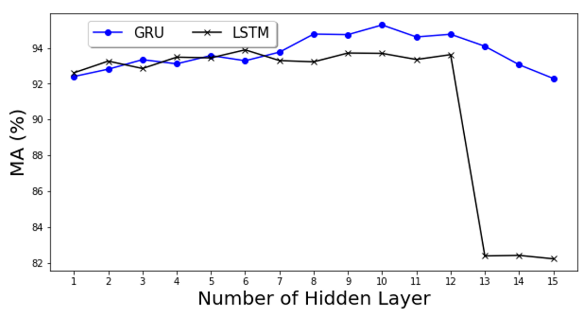

Fig. 10 Comparison of MA

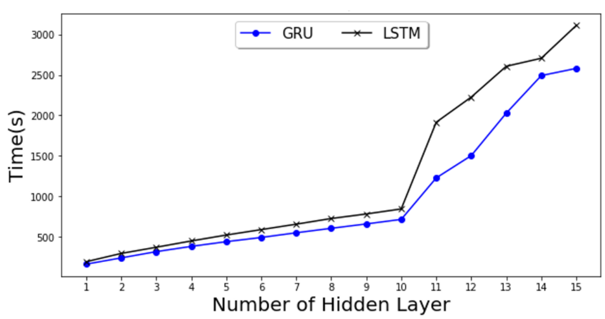

Fig. 11 Comparison of training time.

prediction of base station traffic can be very accurate. The GRU also inherits the short-term dependencies and long-term dependency capabilities of LSTM. The reset gate helps GRU to capture short-term dependencies in time series data and the update gate can capture long-term dependencies in time series data.

However, LSTM requires more training time than GRU, as shown in Fig. 11 because the architecture of the LSTM model is complex. Conversely, GRU replaces the Forget Gate and

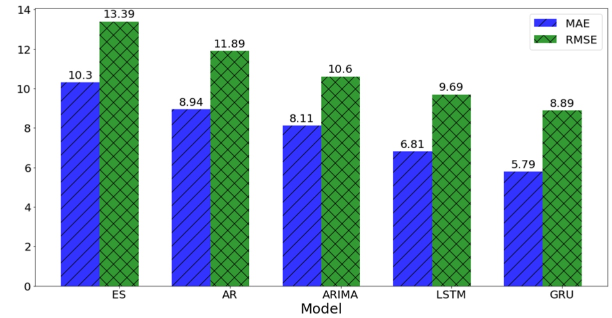

Fig. 12. Comparison of traffic prediction results

Input Gate in the LSTM with an Update Gate that has reduced the complexity of the architecture. However, if data is very large and complex, LSTM may have better training results.

Fig. 12 shows the comparison of traffic prediction results with Exponential Smoothing(ES) [11], Autoregressive(AR) [12], Autoregressive Integrated Moving Average Model (ARIMA)[13], Long Short Term Memory Network (LSTM) [2], and proposed GRU-based model. Since the traditional methods (ES, AR, ARIMA) use statistical methods to predict traffic, time-series data is required to be stationary after differencing. These methods can only capture linear relations, but cannot capture non-linear relationships. Therefore, prediction results are worse than neural networks. Each step of the neural network is a non-linear transformation, which has better adaptability in the real environment.

VI. Discussion

Although GRU is widely used for many optimization problems, it needs to optimize parameters according to different problems, such as hidden layers, activators, etc. The optimization results obtained by parameter adjustment for different data will be different. Therefore, the adjustment of parameters is very important for the learning of neural networks.

In the training models of GRU and LSTM, we recommend using the data drop ratio (Dropout). The network structure of GRU and LSTM is more complicated than the general neural network, and it is easy to produce overfitting problems. Therefore, it is necessary to adjust the optimal dropout. After the experimental parameter test, the Optimizer uses Adam， number of Epoch is 352, the Batch size is 9, and the number of Hidden layers is 10 layers. The best results of GRU has 95.28% accurate and it takes less training time than LSTM.

Although training time for both LSTM and GRU are limited by hardware, the experiment results show that GRU has better performance than LSTM for the real-time problem.

VII. CONCLUSIONS

For the rapid growth of network traffic in the future, we should actively solve the network congestion and QoS problems caused by the increase in traffic. Therefore, using predictive methods to understand network traffic trends in advance is a potential solution. This study adopts GRU to predict network traffic and compare it to LSTM. The simulation results show that both proposed models have good prediction accuracy. However, since the structure of the GRU is simpler than the LSTM, the GRU can reduce a lot of training time and computational resources in the case of similar average accuracy. Therefore, GRU is more suitable for the immediacy problem of traffic prediction. In future works, we will consider the impact of traffic on the position of the base station to improve accuracy.

ACKNOWLEDGMENTS

This work was supported by the Ministry of Science and Technology Program (Project No. MOST 107-2221-E-259-005-MY3 for funding).

REFERENCES

[1] F. Xu, Y. Lin, J. Huang, D. Wu, H. Shi, J. Song, and Y. Li,"Big Data Driven Mobile Traffic Understanding and Forecasting: A Time Series Approach," IEEE Transactions on Services Computing, vol. 9, no. 5, pp. 796–805, Sep. 2016.

[2] Y. Zhou, Z. M. Fadlullah, B. Mao, and N. Kato," A deep-learning-based radio resource assignment technique for 5G ultra dense networks," IEEE Network, vol. 32, no.6, 28-34, 2018.

[3] I. Loumiotis, E. Adamopoulou, K. Demestichas, P. Kosmides, and M. Theologou, "Artificial neural networks for traffic prediction in 4G networks," in Proc. 8th Int. Wireless Internet Conf., Lisbon, Portugal,Nov. 2014, pp. 141–146.

[4] T. P. Oliveira, J. S. Barbar, and A. S. Soares, "Computer network traffic prediction: a comparison between traditional and deep learning neural networks," International Journal of Big Data Intelligence, vol. 3, no. 1, p. 28, 2016.

[5] C.-W. Huang, C.-T. Chiang, and L. Qiuhui, "A study of deep learning networks on mobile traffic forecasting," 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC). IEEE, 2017.

[6] A. Pelekanou, M. Anastasopoulos, A. Tzanakaki, and D. Simeonidou, "Provisioning of 5G services employing machine learning techniques," In 2018 International Conference on Optical Network Design and Modeling (ONDM), pp. 200-205. IEEE, May, 2018.

[7] X. Chen, Y. Jin, S. Qiang, W. Hu, and K. Jiang, "Analyzing and modeling spatio-temporal dependence of cellular traffic at city scale," In 2015 IEEE International Conference on Communications (ICC), pp. 3585-3591, June 2015.

[8] J. Wang, J. Tang, Z. Xu, Y. Wang, G. Xue, X. Zhang, and D. Yang, "Spatiotemporal modeling and prediction in cellular networks: A big data enabled deep learning approach," In IEEE INFOCOM 2017-IEEE Conference on Computer Communications, pp. 1-9, May 2017.

[9] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," arXiv preprint arXiv:1412.3555.

[10] G. Barlacchi et al., "A multi-source dataset of urban life in the city of Milan and the Province of Trentino," Sci. Data, vol. 2, Oct. 2015,Art. no. 150055.

[11] P. S. Kalekar, "Time series forecasting using holt-winters exponential smoothing," Kanwal Rekhi School of Information Technology, vol.4329008, no.13, 2004.

[12] W. Wang and A. K. Wong, "Autoregressive model-based gear fault diagnosis," J. Vib. Acoust. ,vol.124,no.2 ,pp.172-179,2002.

[13] B. Zhou, D. He, and Z. Sun, "Traffic modeling and prediction using ARIMA/GARCH model," Modeling and Simulation Tools for Emerging Telecommunication Networks, Springer, Boston, MA, pp.101-121, 2006.