Data Augmentation based Cellular Traffic Prediction in Edge Computing Enabled Smart City

Zi Wang, Jia Hu*, Geyong Min*, Zhiwei Zhao, and Jin Wang

Abstract—**With the massive deployment of 5G cellular infrastructures, traffic prediction has become an indispensable part of the cellular resource management system in order to provide reliable and fast communication services that can meet the increasing Quality-of-Service (QoS) requirements of smart city. A promising approach for handling this problem is to introduce intelligent methods to implement a highly effective and efficient cellular traffic prediction model. Meanwhile, integrating the multi-access edge computing framework in 5G cellular networks facilitates the application of intelligent traffic prediction models by enabling their implementation at the network edge. However, the data shortage and privacy issues may still be obstacles for training a robust and accurate prediction model at the edge. To address these issues, we propose a data augmentation based cellular traffic prediction model (ctGAN-S2S) where an effective data augmentation sub-model based on generative adversarial networks is proposed to improve the prediction performance while protecting data privacy, and a long short-term memory based sequence-to-sequence sub-model is used to achieve the flexible multi-step cellular traffic prediction. The experimental results on a real-world city-scale cellular traffic dataset reveal that our ctGAN-S2S model achieves up to 48.49% improvement of the prediction accuracy compared to four typical reference models.**

Index Terms—**Data Augmentation, Time-Series Prediction, Neural Networks, Cellular Networks, Smart City.**

I. Introduction

THE tremendous advance in modern technologies, many cities have recently introduced emerging infrastructures and applications for smart city development. For example, smart transportation systems, unmanned aerial vehicle delivery, augmented and virtual reality and industry 4.0 applications [1] bring great benefits for making our cities more convenient and efficient, which significantly improve the quality of life for city residents.

To speed up the development of smart city, smart technologies should be introduced in our daily life and also need to be constantly upgraded to meet the increasing requirements on Quality-of-Services (QoS). Therefore, a reliable communication system is essential to meet the demanding requirements from smart city applications on high speed transmission and short delay. The fifth generation of wireless communications for digital cellular networks (5G) enters a fast development period in recent years and becomes an ideal and practical solution. 5G cellular networks are assumed to be the key enabler and infrastructure provider of smart city by offering the enhanced mobile broadband (eMBB), ultra-reliable low latency service (URLLC) and massive machine-type communications (mMTC) [2]. 5G is also the first cellular standard that explicitly targets industrial use cases [3].

All these visions highly depend on the proactive resource allocation from the 5G cellular management systems and the prediction model plays a critical role therein. The cellular traffic usage data not only can be used in billing but also be regarded as an important and rich data source [4] for developing other utilities such as traffic prediction models. A traffic prediction model monitors real-time traffic data and provides future state predictions. By referring the short term or long term prediction results, the cellular management system can be more efficient to allocate resources to meet different requirements and improve the reliability and QoS of smart city applications.

However, there are many challenges on implementing a reliable traffic prediction model for cellular networks. First, the prediction model needs to have capabilities to handle dynamics in terms of different scenarios and time periods. Second, not all models can own a sufficient amount of data to train and build an effective prediction model. Third, the privacy concern is raised when the cooperation and data sharing among base stations (BSs) is needed for improving the accuracy of prediction models. The conventional statistical prediction models with the assistance of expert knowledge were usually applied for cellular resource allocation under a specific scenario. These models have limited capabilities to efficiently adapt to the dynamic changes and handle increasing complex network conditions. Deep learning has experienced tremendous success in recent years. For example, a sequence-to-sequence (S2S) [5] model can give a reliable and satisfying prediction result for time-series data and a generative adversarial network (GAN) [6] achieves great progress in generating new data with the similar distribution as the targeting dataset. Thus, deep learning not only can help to build a reliable prediction module but provide a way to alleviate the data shortage problem by using the GAN based data augmentation method.

Implementing these intelligent methods requires many computing resources for the model training. Conventional hardware on cellular BSs does not have enough capability to efficiently handle such requirements. To mitigate this problem,

Z. Wang is with the School of Computer Science and Technology, University of Electronic Science and Technology of China, and the College of Engineering, Mathematics and Physical Sciences, University of Exeter, UK. (Email: zw300@exeter.ac.uk)

J. Hu, G. Min and J. Wang are with the College of Engineering, Mathematics and Physical Sciences, University of Exeter, UK. (Email: {j.hu, g.min, jw855}@exeter.ac.uk)

Z. Zhao is with the School of Computer Science and Technology, University of Electronic Science and Technology of China, China. (Email: zzw@uestc.edu.cn)

*Corresponding Author

multi-access edge computing (MEC) has been proposed as a key part of 5G networks. Different from cloud computing that provides scalable and high-performance computing resources at a centralised data centre, MEC equips cellular BSs with edge servers that have communicating and computing capabilities to provide services in close proximity to users to cut down the service delay and network traffic. Thus, all the model training and data augmentation work can be handed over to edge servers.

In this work, we propose a data augmentation based cellular traffic prediction model (ctGAN-S2S) that can be deployed on the edge servers in 5G cellular networks. This model consists of two sub-models: 1) the data augmentation model (ctGAN) and 2) the cellular traffic prediction model. The main contributions of this paper are:

- We propose the ctGAN-S2S model to effectively achieve data augmentation and improve the cellular traffic prediction performance. This new model can protect privacy by using the data augmentation model to generate close-to-real cellular traffic data, which is used to reduce direct data sharing between base stations for training the prediction model.
- We propose a cellular traffic data augmentation sub-model ctGAN, where the potential time-series samples of target cellular traffic can be generated for data augmentation through the generative adversarial process.
- We design a long short-term memory (LSTM) based S2S prediction sub-model, which can achieve flexible multistep prediction on time-series cellular traffic data.
- We evaluate the proposed ctGAN-S2S model on real-world cellular traffic dataset, and the results show that this new model achieves up to 48.49% improvement of prediction accuracy compared with four reference models.

The rest of this paper is organised as follows. Section II presents the preliminaries of our work. Section III summarises the related works. Section IV introduces the system framework. Section V presents the details of the data augmentation based cellular traffic prediction model. Section VI evaluates the proposed work. Section VII concludes this work.

II. Preliminaries

A. Recurrent Neural Network

A recurrent neural network (RNN) consists of the input, hidden and output layers. The RNN provides an internal memory mechanism which can jointly consider the current input and previous information in hidden states before producing its output. The RNN can well support in describing the dynamic sequential data with strong temporal features. Let x_t denote the input traffic information vector at time step t. The hidden state vector h_t and output vector y_t can be calculated as

$$h_t = \sigma_h(W_h x_t + U_h h_{t-1} + b_h) \tag{1}$$
$$y_t = \sigma_y(W_y h_t + b_y) \tag{2}$$

where W, U and b are the connecting parameters and bias term; $\sigma(\cdot)$ is the activation function.

B. Long Short-term Memory

The long short-term memory network (LSTM) is an improved RNN. LSTM was proposed to solve the gradient vanishing and explosion problem during training for long-term dependencies in RNN [7]. LSTM cell receives its current input x_t as well as its previous cell state c_{t-1} and hidden state h_{t-1} and then uses three controllers called the input gate i_t, the forget gate f_t and the output gate o_t to determine what extent of information to be added, removed and presented. Formally, the current cell state c_t, hidden state h_t and three gates can be calculated as

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{3}$$
$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{4}$$
$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{5}$$
$$g_t = \phi(W_g x_t + U_g h_{t-1} + b_g) \tag{6}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \tag{7}$$
$$h_t = o_t \odot \phi(c_t) \tag{8}$$

where g_t is the intermediate state, and W, U and b are the connecting parameters and bias term; $\sigma(\cdot)$ and $\phi(\cdot)$ are the activation functions such as sigmoid and tanh.

III. Related Work

In this section, we investigate the state-of-the-art research in 5G cellular networks, the MEC framework and learning-based augmentation and prediction models.

Caragliu *et al.* [8] proposed a comprehensive survey and investigation on the impact of smart city policies through information and communication technologies. A detailed discussion about smart city applications enabled by 5G communication technologies with unprecedented reliability, very low latency and massive connections was given by Rao *et al.* [9]. Multi-access edge computing, also named as mobile edge computing, is an emerging framework in recent years. In the ETSI white paper, the MEC is expected to play an important role and is specifically designed to be deployed and integrated into the 5G cellular system. Various applications will benefit from the MEC-based 5G cellular services [10]. Recent research proved that MEC has capabilities to facilitate smart city applications such as IoT networks [11] [12], VANETs [13] and healthcare systems [14] [15].

Li *et al.* [2] proposed the 5G cellular network architecture enabled by artificial intelligence. In their work, they described that resource management systems can be proactively notified to save cost by taking advantage of the embedded prediction module. Several works (e.g. [4] [16]) made efforts for promoting intelligent methods in 5G. Their work proved that the knowledge behind the cellular usage data is sufficient and useful, and worth exploiting. It also has the potential to make great contributions in developing smart technologies for both in smart city development and 5G communications.

The data is critical to train a robust and reliable learning model. However, the amount of data may be a bottleneck for those small and micro businesses. Therefore, data augmentation is an important method to alleviate the negative impact of data shortage. GAN [6] is the first and the most successful neural network model that is designed for data generation via an adversarial process. Recently, Zhang *et al.* [17] proposed GAN-based model to solve the insufficiency of training data and infer the fine-grained mobile traffic patterns. They regarded the mobile traffic patterns as images or videos and were inspired by the resolution enhance in image process that used GAN to achieve their goals. In their work, the generative model of GAN can generate the high-resolution approximations of the real traffic distribution. Their work proved the effectiveness of using GAN for data augmentation. Moreover, applications of GAN have been scaled from the image processing to the sequential data such as music generation [18], medical ICU data generation [19] and time-series anomaly detection [20]. However, to the best of our knowledge, there has not been any work reported that adopted GAN-based model in data augmentation for cellular traffic usage data in 5G.

For the prediction model, RNN is a very effective and popular network for time-series data processing, and LSTM is an evolution of RNN. Yuan *et al.* [21] proposed a LSTM-based model in the soft sensor industrial application to learn the dynamic hidden states which makes contributions for quality prediction. Recently, sequence-to-sequence (S2S) model [22] was investigated and applied to process time-series data. The S2S has capabilities of reading and inferring sequences of arbitrary length, which provides flexible temporal sequential data processing ability for applications such as the trajectory prediction [23] and voice conversion [24].

IV. System Framework

In the various application scenarios of the smart city such as smart transportation systems, IoT devices and industrial applications, many related management works not only need a high-speed and low-latency network connection but also require extra computing assistance to cope with complex models. With the emerging 5G technologies, these requirements can be easily fulfilled by directly connecting to cellular networks. MEC is an important framework of the 5G cellular networks to enhance 5G advantages. MEC provides computation and communication services in close proximity to subscribers to meet the high-workload and low-latency requirements.

As described in Fig. 1, the multi-access edge cellular BS is considered as the basic local communication and processing unit that provides adequate communicating and computing resources for edge users and service providers to assist their work. Various devices and applications directly connect to the BS to request and receive MEC services. The local BS communicates with other MEC BSs and connects to the Internet with the reliable backhaul link. Meanwhile, BS records the device connectivity and traffic usage data when it provides services. These records naturally can be exploited to help BS management system to improve the utilisation of cellular resources. Therefore, the data augmentation based cellular traffic prediction model is deployed at the BS in our system framework. This model collects cellular network traffic data from its connected ends, and shares the collected records with

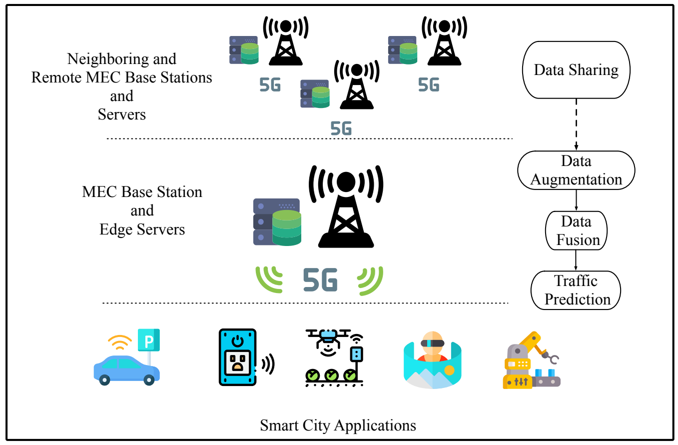


Fig. 1. The Framework for a 5G Multi-access Edge Computing System in Smart City

other BSs only for the purpose of the augmentation model training. By using this shared information, the data augmentation model at the local BS can train its data generation module and generate time-series data which is close to the real cellular traffic. Then, the model can fuse the real collected cellular traffic and the generative data together to train the prediction module. After finishing the training process, this model can provide a reliable short term or long term traffic prediction for cellular management.

The benefits of this framework can be explained in three aspects. 1) MEC deployed in the 5G cellular network can provide ultra-reliable low latency communication and computation services which explicitly target at the use cases of smart city. It not only fulfils the increased QoS requirements for smart city applications but also reduces the traffic workload in the backbone networks. 2) For some service providers that do not have sufficient data to train an accurate prediction model, this data augmentation based cellular traffic prediction model can help them to acquire more accurate prediction results. 3) For privacy concerns, the cellular network traffic data will only be kept within the cellular edge servers and will not be shared to other service providers. Using the generative data instead of shared real data for the prediction model training can reduce privacy issues for involved users.

V. Data Augmentation based Cellular Traffic Prediction Model

In this section, the details of the data augmentation based cellular traffic prediction model are presented. This model consists of two sub-models, the data augmentation model and the prediction model.

A. Data Augmentation Model

The **c**ellular **t**raffic data augmentation model is the **g**enerative **a**dversarial **n**etworks with LSTM core (ctGAN). As illustrated in Fig. 2, ctGAN consists of two networks: a generator (G) and a discriminator (D), which are two different deep LSTM networks. During the training process, the discriminator receives cellular traffic time-series data as the input to learn the experience from real data and then to identify the generative data. The generator aims to produce time-series data which is closed to the real one and makes the discriminator cannot tell the difference between the generative time-series and the real time-series traffic data. Given a random sequence z and cellular traffic time-series data x as inputs, for a batch of m samples the loss functions of the generator L_G and discriminator L_D can be calculated as

$$L_G = \frac{1}{m}\sum_{i=1}^{m} log(1 - D(G(z))) \tag{9}$$

$$L_D = \frac{1}{m}\sum_{i=1}^{m} -(logD(x) + (log(1 - D(G(z))))) \tag{10}$$

The iterative training process of these two networks becomes a zero-sum game and stop until their loss functions converge. After the training of ctGAN finished, the generator is able to produce the generative cellular traffic usage data which is closed to the one the exists in the real situation. Compared to the data generated by specific distributions, our proposed data augmentation model can generate complex but strong likeness to the real cellular traffic usage data which brings better augmentation for training the prediction model.

B. Prediction Model

According to the requirement for the flexible configuration from service users, we use the S2S model to achieve the multi-step prediction. As shown in Fig. 2, our S2S prediction model is implemented by two LSTM networks. One of them works as an encoder and the other works as a decoder. Benefiting from this separate architecture, the S2S model in our proposed model has the capability of learning from the cellular traffic usage history data with an arbitrary length time-window denoted by T while outputting the prediction for the future usage with another length time-window denoted by T'. When $T' > 1$, the S2S prediction model can easily achieve the multi-step prediction.

Given the cellular traffic history time-series data x_T with the length of T, at each time step $t \in T$ the encoder receives the inputs of x_t as well as the previous network output the hidden state h^e_{t-1} and cell state c^e_{t-1}. After T steps recursively updates, the encoder produces the final hidden and cell states h^e_T and c^e_T through Eqs (8) and (7) respectively. Then, the decoder receives h^e_T and c^e_T as the initial states from the encoder. The decoder starts its calculation by using a trigger input (e.g. 0). During the training process, the decoder receives the sample target $y'_{t'-1}$ as the input $x_{t'}$ thus assuming that it makes a perfect prediction at the previous time step. The decoder recursively produces the values until the output time-series reaches the required length of T'. The loss function is defined as

$$L(y', y) = \frac{1}{T'}\sum_{t=T+1}^{T'} (y' - y)^2 \tag{11}$$

where y' is the target time-series observations and y is the corresponding predicted value. The objective is to minimise the loss and get corresponding parameters Θ of S2S model, $\Theta = \arg\min_{\Theta} L(y', y)$, which can be trained by a stochastic gradient-based optimiser. During the prediction process, the decoder does not know the actual value for near-future time instances when proceeding the multi-step prediction, so it feeds back the previous output $y_{t'-1}$ as the input $x_{t'}$ for the next time step prediction.

To enhance the capability of a better representation of traffic history, we use a bidirectional LSTM network for the encoder. The Bi-LSTM consists of forward and backward LSTMs. The forward LSTM reads the input x_t from x_0 to x_T and calculates the forward hidden states as $(\overrightarrow{h}_1, ..., \overrightarrow{h}_T)$. The backward LSTM reads the input x_t from x_T to x_0 and calculates the backward hidden states as $(\overleftarrow{h}_1, ..., \overleftarrow{h}_T)$. Then, the final hidden state for the input x_t can be obtained by concatenating the forward and backward hidden states as $h_t = [\overrightarrow{h}_t; \overleftarrow{h}_t]$. In this way, the final hidden state h_t contains the summaries of both the preceding usage state and the following usage state.

For the prediction model, mean square error (MSE) and mean absolute error (MAE) are two popular metrics for performance judgement, so we adopt them to evaluate prediction results in this work. Let $\hat{x}_t$ represent the target time-series observations and x_t denote the prediction result of our model. These two metrics can be described as

$$MSE = \frac{1}{N}\frac{1}{W}\frac{1}{T}\sum_{i=1}^{N}\sum_{w=1}^{W}\sum_{t=1}^{T}(\hat{x}^{w,i}_t - x^{w,i}_t)^2 \tag{12}$$

$$MAE = \frac{1}{N}\frac{1}{W}\frac{1}{T}\sum_{i=1}^{N}\sum_{w=1}^{W}\sum_{t=1}^{T}|\hat{x}^{w,i}_t - x^{w,i}_t| \tag{13}$$

where N, W and T denote the number of BSs, time window and time steps respectively. They are used to measure the accuracy of time-series traffic between the prediction result and true value while linear metric MAE treats all differences with equal weight.

C. Overall Procedure

Fig. 2 illustrates the overall procedures for the data augmentation based cellular traffic prediction model and Algorithm 1 shows the pseudocode. First, two steps are executed at the same time ①: the ctGAN generator receives the random vectors as input and tries to generate the time-series traffic data that the discriminator cannot differentiate from the real. The ctGAN discriminator works as a binary classifier and receives the data from both the generator and the real data samples as fake and real instances respectively to train itself. ② After training, the generator enables to produce the time-series data that looks very close to the real traffic usage data, which means the same traffic usage may exist in the future. Therefore, the data augmentation for the training set of the prediction model is achieved by fusing the generative data with the real traffic data from local users. In this work, the real traffic dataset and generative dataset are concatenated to finish this fusion process. ③ Then, the S2S prediction model takes the fusion data to train the encoder and decoder module. The S2S can achieve flexible multi-step prediction by slicing different lengths of time-window. ④ When a S2S model is

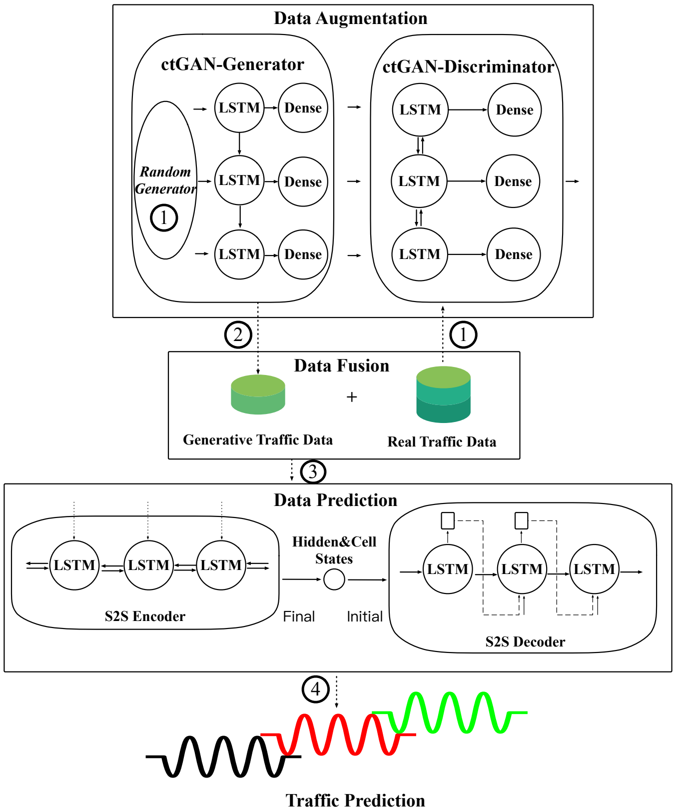


Fig. 2. Data Augmentation based Cellular Traffic Prediction Model. ①-④ show the overall procedures of the model.

Algorithm 1 ctGAN-S2S

Input:

z, x, T, T';

Output:

The predicted time-series, y;

Initialise parameters;
while Not converged **do**
    The generator G produces the generative sequences $G(z)$;
    The discriminator D discriminates the generative sequences $D(G(z))$ and real data x;
    Compute losses of L_G and L_D;
    Update parameters via Stochastic Gradient Descent (SGD)-based optimiser;
end while
The generator produces the generative sequences;
Concatenate the generative sequences and real local time-series as an augmentative fusion dataset;
Sample $(x_T, y_{T'})$ from the augmentative fusion dataset;
while Not converged **do**
    $(\overrightarrow{h_t^e}, \overrightarrow{c_t^e})$ = encoder(x_T);
    $(\overleftarrow{h_t^e}, \overleftarrow{c_t^e})$ = encoder(x_T);
    y = decoder($x_{t'}, (\overrightarrow{h_t^e}, \overleftarrow{h_t^e}), (\overrightarrow{c_t^e}, \overleftarrow{c_t^e})$);
    Compute losses $L(y', y)$;
    Update parameters Θ via SGD-based optimiser;
end while
Input target cellular traffic to well-trained S2S;
S2S predicts the results of future T' steps.

trained, it can provide a more accurate prediction within the local community that the MEC services cover.

VI. Performance Evaluation

In this section, we conduct extensive experiments to evaluate the effectiveness of the proposed model.

A. *Dataset Description*

Our proposed model is evaluated based on the city-scale cellular traffic dataset [25]. This traffic dataset consists of five dimensions information including BS identity, hourly time stamp, the number of active users, transferred packets and transferred bytes. The time span of this dataset is a continuous 8 days from 19 August to 26 August 2012 and each individual is detected by the hashed International Mobile Subscriber Identity (IMSI).

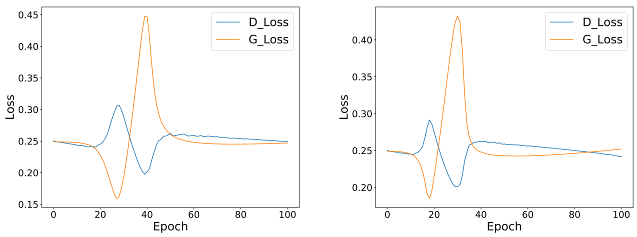
Due to the system issues during the recording process, not all BSs hold a full record for 192 continuous hours in the dataset. Therefore, in our experiments, we used the package information of a subset of nearly 2K BSs which hold the complete record within this period. This effective dataset contains 1983 BSs information out of 13269 BSs, which is nearly 15% of the total BSs. This effective rate also supports our previous viewpoint that not all businesses can acquire sufficient data and the data augmentation is an important method to help improve the training of the prediction model.

The cellular traffic data from the first 96 hours are used as the training set and the rest is used as the test set. The length of the time window for these two sub-datasets is set as 30 hours which includes the 24-hour data for the encoder and 6-hour data for the decoder in the S2S network. All of our proposed model and comparing methods will be evaluated on these two sub-datasets. We use the min-max normalisation method to scale the traffic usage data into [0, 1] for the training process.

B. *Experiment Settings*

1) Reference Models: We compare our model with three reference models, which are described below:

- Autoregressive Integrated Moving Average (ARIMA): ARIMA is a widely adopted model to predict the future value of a time-series, which converts the relations among the original time-series to reflect the relations among the difference between the time-series. ARIMA is implemented by using the default out of box tools provided in the statsmodels [26].
- Nonlinear autoregressive neural network (NARNN): The NARNN is a basic fully connected artificial neural network that can be used in cellular traffic prediction. NARNN regards the time steps within a time window as a whole part and learns from this time window. [27]
- LSTM: The standard LSTM is used to directly learn and predict future time-series from the cellular traffic usage data. The standard LSTM uses the memory cell to store useful long-term status information and it is widely used in single-step sequence prediction. [28]



((a)) The losses with lr=10^{-4} for both G and D

((b)) The losses with lr=$2*10^{-4}$ for G and lr=10^{-4} for D

Fig. 3. The losses with the same learning rate and different learning rates for the generator and discriminator

TABLE I
COMPARISONS OF PREDICTION PERFORMANCE ON DIFFERENT EVALUATION METRICS

		Metrics	
		MSE	MAE
Models	ARIMA	0.0076	0.053
	NARNN	0.0066	0.054
	LSTM	0.0064	0.046
	S2S	0.0057	0.043
	ctGAN-S2S	0.0042	0.043

- Sequence to sequence model (S2S): The standard S2S model without data augmentation model ctGAN.

2) Model Training: We implement the ARIMA model using the ‘statsmodels’ python package, and empirically set the lag value to 10 for autoregression, use a difference order of 1 to make the time series stationary and use a moving average model of 0. We implement the NARNN model and set 256 hidden units for first two fully connected layers and 6 neurons for the output layer with the rectified linear unit (ReLU) activation function. We implement the two layers LSTM model and choose 256 hidden units for each layer with a ReLU activation function. In the training settings of the ctGAN, a bidirectional LSTM is used for the discriminator and a unidirectional LSTM is used for the generator. Each LSTM cell has 256 hidden units. For the discriminator and generator, the outputs are followed by a fully-connected layer with sigmoid and tanh activations respectively. All parameters are learned and updated through stochastic gradient-based optimiser, Adam [29], with the mini-batch size 128 and back-propagation through time. In the training of ctGAN, the separate learning rates are set as 10^{-4} for the discriminator and $2*10^{-4}$ for the generator. The training of the ctGAN is an adversarial process, so it is hard to find an optimal learning rate. Therefore, the learning rate that we chose is the best based on the empirical results. Normally, the learning rates of generator and discriminator are the same. We set the learning rate of generator to be larger than discriminator in order to make the generator receive more training than the discriminator, which can avoid the discriminator becoming too strong and effectively help to improve the generator. The results of losses with the same learning rate and our rates shown in Fig. 3 also indicate that our setting can help the ctGAN converge faster than the conventional one.

In the training settings of the S2S, a bidirectional LSTM is used for the encoder and a unidirectional LSTM is used for the decoder. 256 hidden units are set in each LSTM cell. The optimiser for S2S is the same as ctGAN but the mini-batch size is set to 256. More hidden units can provide more details and better representations of time-series data but also bring higher computation and time consumption and the potential over-fitting problem during the training phase. These models are implemented by using the Keras with TensorFlow backend and run on a platform with 16GB of memory, 4-core Intel CPU i5@3.40GHz and GTX1070 GPU.

For the prediction performance evaluation, two metrics MSE and MAE that we mentioned in Section V are employed. Recall that MSE denotes the average squared difference between the predicted values and the actual values and MAE denotes the arithmetic average between the predicted values and the actual values as a scale-dependent accuracy measure. A smaller value of MSE and MAE indicates a better performance of the model.

C. Performance Evaluation

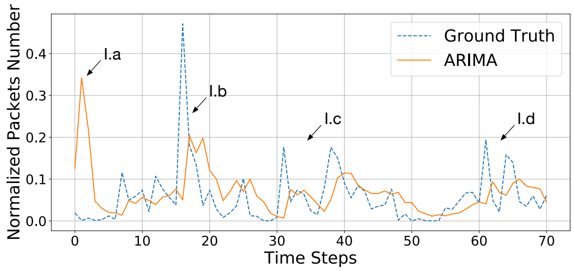
The numeric results of evaluation metrics on different methods are presented in Table I. As can be seen from the results, our proposed augmentation based prediction model for the cellular traffics (ctGAN-S2S) outperforms other models on the MSE and MAE metrics. It achieves up to 44.74%, 36.36%, 34.38% and 26.32% performance improvement compared to the reference models, respectively. The reasons can be attributed to the following aspects.

First, as the cellular traffic data is considered to be composed of the linear autocorrelation and nonlinear time-series, ARIMA is mainly used to model the linear relations in time-series so it fails to capture the nonlinear component, which affects the prediction performance on our real cellular traffic dataset. This shortage can be solved in the artificial neural networks which have capabilities of handling nonlinear time-series. Therefore, the results in Table I reveal that all neural network based model outperforms the ARIMA model.

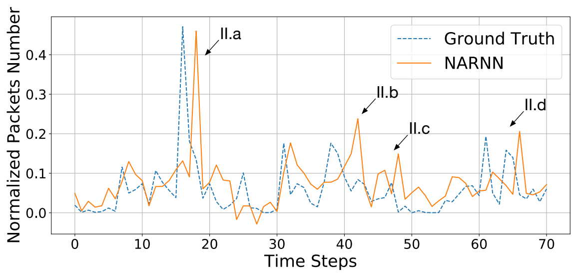
Second, benefiting from the internal memory of the recurrent mechanism, RNN-based models have natural advantages in learning from sequential data. For time-series of the cellular traffic, the LSTM can learn more from it and show better performance than the NARNN, which regards each time-window-length fragment of time-series as a whole block and fails to capture the temporal within it.

Third, S2S model can train its encoder and decoder separately, which provides better performance on flexible multi-step prediction capability.

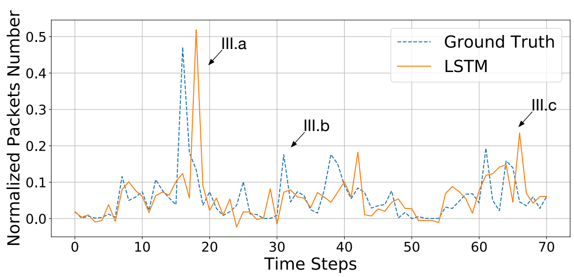
Our proposed ctGAN-S2S model integrates all the advantages mentioned above. Besides, with the close-to-real generative data generated by a well-trained ctGAN model, the S2S model can even learn from those time-series data that may reflect the future cellular traffic usage. The numeric



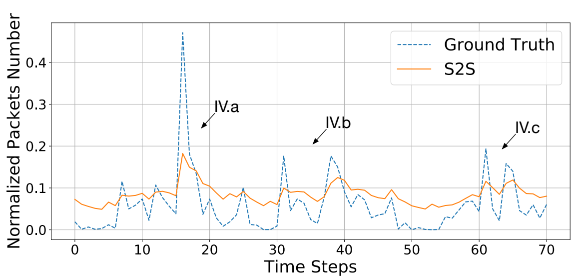
((a)) Case I:ARIMA



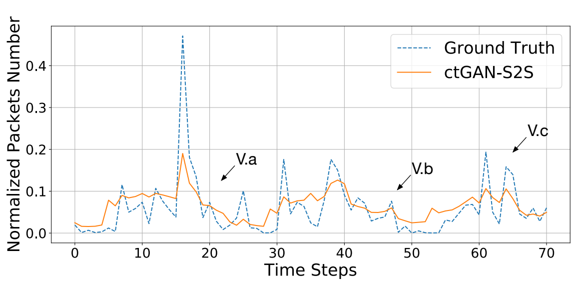
((b)) Case II: NARNN



((c)) Case III: LSTM



((d)) Case IV: S2S



((e)) Case V: ctGAN-S2S

Fig. 4. Comparisons of the prediction performance on base station No.4324 for different models

TABLE II
COMPARISONS OF PREDICTION PERFORMANCE ON DIFFERENT AUGMENTATION RATE

		Metrics	
		MSE	MAE
Data Augmentation Rate	5%	0.004245	0.043295
	7.5%	0.004244	0.040744
	10%	0.003915	0.038342
	15%	0.004146	0.041435
	20%	0.004305	0.043507

result proves the effectiveness of the ctGAN model in data augmentation.

To further illustrate the prediction performance of our proposed model, we randomly select one BS (No. 4324) to show the comparison between the prediction results and ground truth for different models. As we can see from the case in Fig. 4(a), the prediction results of ARIMA exist several obvious faults (e.g. Arrow I.a-d) and miss nearly all peaks. These obviously inaccurate predictions result in the worst performances among other methods in both MSE and MAE metrics. The results of NARNN and LSTM presented in Fig. 4(b) and Fig. 4(c) are better than ARIMA's in terms of the trend and seasonality, but non-negligible number of data points present inaccuracy such as delayed phenomenon (e.g. Arrow II.a-d and Arrow III.a-c) compared to the ground truth. The S2S model captures the trend and seasonality changes much better than the previous models. This ability is proved by the results shown in Fig. 4(d). It indicates that S2S model remains the capabilities in trend and seasonality learning and then gives correct predictions of them. Most peaks (e.g. Arrow IV.a-c) are correctly predicted without delayed phenomena, which are major contributions for the improvement of performance metrics. Furthermore, these capabilities of S2S model are enhanced by fusing the generative data from ctGAN. Fig. 4(e) shows the best results from our proposed model among all reference models. It indicates that ctGAN-S2S well captures the trend and seasonality without obvious fault points and missed peaks and provides a more compact reflection (e.g. Arrow V.a-c) for the ground truth among others.

Similarly, in the above evaluation experiment, we still fuse 5% of the original training amount generative data into the training set. To further explore the performance improvement of our proposed model, we evaluate the ctGAN-S2S with different data augmentation rates.

Table II depicts the prediction performance against the generative rate of data fused into the training set. When we raise the augmentation rate from 5% to 10%, it is shown that the performance increases on both MSE and MAE metrics. However, when we further increase the amount of generative data the effect of performance improvement halts and it becomes worse when fusing 15% and 20% generative data into the training set. The performance keeps decreasing when more than 20% generative data is fused into the real cellular dataset.

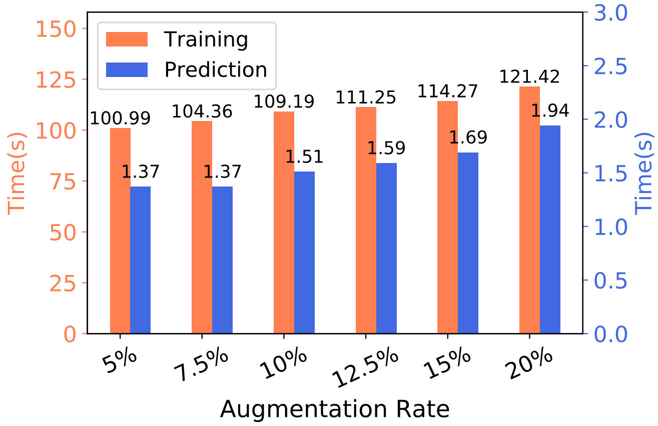


Fig. 5. Training and Prediction Time Consumption

The reason is that the prediction model will learn more from them and the impact of real data on model weights will be reduced when the proportion of the generative data increases. Then the prediction model will focus more on the generative data rather than the real one and the prediction results will lose the accuracy.

Another important issue that needs to be analysed is the trade-off between prediction accuracy and computation resources. Fig. 5 shows the time consumption in the training and prediction process based on different augmentation rates. The results indicate that the time consumption of both training and prediction increases with the augmentation rate. Specifically, the average incremental rates of time consumption in training and prediction process are only 3.1% and 5.78% with 2.5% incremental augmentation rate, respectively. Combined with the prediction performance improvement we analysed above, the extra time consumption is an affordable cost that MEC servers can provide.

VII. Conclusion

In this paper, we first introduced how 5G technologies can bring benefits for smart city development and identified the related challenges. In order to solve these challenges, we investigated the state-of-the-art research and discussed the feasibility of using 5G cellular traffic data, promising neural network-based models and MEC framework. We made efforts on exploring the problem of jointly considering predicting accurate cellular traffic with limited real data and protecting data privacy. To this end, we proposed a data augmentation based cellular traffic prediction model called ctGAN-S2S which consists of a cellular traffic generative adversarial network and a sequence-to-sequence neural network to provide an improved traffic prediction. We then evaluated our proposed model using a real city-scale cellular traffic dataset. The numerical results showed that our proposed model achieves the best prediction accuracy which was improved by 26.32% $\sim$ up to 48.49% among all reference models. The proposed method can be used to provide accurate cellular traffic prediction and alleviate the negative impact of data insufficiency for smart city development. For our future work, we will consider the geographic information of the cellular base stations and exploit the spatial-temporal dependencies of corresponding cellular traffic to further improve the accuracy of traffic prediction.

Acknowledgement

This work is partially supported by the National Key R&D Program of China (No. G072017YFB1400102), the National Natural Science Foundation of China (No. 61972075 and No. 61972074), and the China Scholarship Council (No. 201806070140).

References

[1] M. Wollschlaeger, T. Sauter, and J. Jasperneite, "The future of industrial communication: Automation networks in the era of the internet of things and industry 4.0," *IEEE industrial electronics magazine*, vol. 11, no. 1, pp. 17–27, 2017.

[2] R. Li, Z. Zhao, X. Zhou, G. Ding, Y. Chen, Z. Wang, and H. Zhang, "Intelligent 5g: When cellular networks meet artificial intelligence," *IEEE Wireless Communications*, vol. 24, no. 5, pp. 175–183, 2017.

[3] S. Vitturi, C. Zunino, and T. Sauter, "Industrial communication systems and their future challenges: next-generation ethernet, iiot, and 5g," *Proceedings of the IEEE*, vol. 107, no. 6, pp. 944–961, 2019.

[4] D. Jiang, Y. Wang, Z. Lv, S. Qi, and S. Singh, "Big data analysis-based network behavior insight of cellular networks for industry 4.0 applications," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 2, pp. 1310–1320, 2020.

[5] Z. Mariet and V. Kuznetsov, "Foundations of sequence-to-sequence modeling for time series," in *Proc. PMLR*, vol. 89, 2019, pp. 408–417.

[6] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. NIPS*, 2014, pp. 2672–2680.

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

[8] A. Caragliu and C. F. Del Bo, "Smart innovative cities: The impact of smart city policies on urban innovation," *Technological Forecasting and Social Change*, vol. 142, pp. 373–383, 2019.

[9] S. K. Rao and R. Prasad, "Impact of 5g technologies on smart city implementation," *Wireless Personal Communications*, vol. 100, no. 1, pp. 161–176, 2018.

[10] S. Kekki, W. Featherstone, Y. Fang, P. Kuure, A. Li, A. Ranjan, D. Purkayastha, F. Jiangping, D. Frydman, G. Verin *et al.*, "Mec in 5g networks," *ETSI white paper*, vol. 28, pp. 1–28, 2018.

[11] W. Gao, Z. Zhao, Z. Yu, G. Min, M. Yang, and W. Huang, "Edge computing based channel allocation for deadline-driven iot networks," *IEEE Transactions on Industrial Informatics*, 2020.

[12] K. Kaur, S. Garg, G. S. Aujla, N. Kumar, J. J. Rodrigues, and M. Guizani, "Edge computing in the industrial internet of things environment: Software-defined-networks-based edge-cloud interplay," *IEEE communications magazine*, vol. 56, no. 2, pp. 44–51, 2018.

[13] S. Garg, A. Singh, K. Kaur, G. S. Aujla, S. Batra, N. Kumar, and M. S. Obaidat, "Edge computing-based security framework for big data analytics in vanets," *IEEE Network*, vol. 33, no. 2, pp. 72–81, 2019.

[14] C. Shu, Z. Zhao, G. Min, and S. Chen, "Mobile edge aided data dissemination for wireless healthcare systems," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 5, pp. 898–906, 2019.

[15] G. S. Aujla, R. Chaudhary, K. Kaur, S. Garg, N. Kumar, and R. Ranjan, "Safe: Sdn-assisted framework for edge–cloud interplay in secure healthcare ecosystem," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 1, pp. 469–480, 2018.

[16] Q. Zhang, L. T. Yang, Z. Yan, Z. Chen, and P. Li, "An efficient deep learning model to predict cloud workload for industry informatics," *IEEE transactions on industrial informatics*, vol. 14, no. 7, pp. 3170–3178, 2018.

[17] C. Zhang, X. Ouyang, and P. Patras, "Zipnet-gan: Inferring fine-grained mobile traffic patterns via a generative adversarial neural network," in *Proc. CoNEXT*, 2017, pp. 363–375.

[18] O. Mogren, "C-rnn-gan: Continuous recurrent neural networks with adversarial training," *arXiv preprint arXiv:1611.09904*, 2016.

[19] C. Esteban, S. L. Hyland, and G. Rätsch, "Real-valued (medical) time series generation with recurrent conditional gans," *arXiv preprint arXiv:1706.02633*, 2017.

[20] D. Li, D. Chen, B. Jin, L. Shi, J. Goh, and S.-K. Ng, "Mad-gan: Multivariate anomaly detection for time series data with generative adversarial networks," in *Proc. ICANN*, 2019, pp. 703–716.

[21] X. Yuan, L. Li, and Y. Wang, "Nonlinear dynamic soft sensor modeling with supervised long short-term memory network," *IEEE Transactions on Industrial Informatics*, 2019.

[22] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proc. NIPS*, 2014, pp. 3104–3112.

[23] S. H. Park, B. Kim, C. M. Kang, C. C. Chung, and J. W. Choi, "Sequence-to-sequence prediction of vehicle trajectory via lstm encoder-decoder architecture," in *Proc. IEEE IV*, 2018, pp. 1672–1678.

[24] J.-X. Zhang, Z.-H. Ling, L.-J. Liu, Y. Jiang, and L.-R. Dai, "Sequence-to-sequence acoustic modeling for voice conversion," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 3, pp. 631–644, 2019.

[25] X. Chen, Y. Jin, S. Qiang, W. Hu, and K. Jiang, "Analyzing and modeling spatio-temporal dependence of cellular traffic at city scale," in *Proc. IEEE ICC*, 2015, pp. 3585–3591.

[26] S. Seabold and J. Perktold, "Statsmodels: Econometric and statistical modeling with python," in *Proc. SciPy*, vol. 57, 2010, p. 61.

[27] M. Balanici and S. Pachnicke, "Machine learning-based traffic prediction for optical switching resource allocation in hybrid intra-data center networks," in *Proc. OFC*, 2019.

[28] A. Azari, P. Papapetrou, S. Denic, and G. Peters, "Cellular traffic prediction and classification: a comparative evaluation of lstm and arima," in *Proc. DS*, 2019, pp. 129–144.

[29] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

Zi Wang is currently a computer science Ph.D. student in the College of Engineering, Mathematics and Physical Science at the University of Exeter, UK. He received his M.Sc. and B.E. in computer science from the University of Electronic Science and Technology of China (UESTC) in 2018 and 2015, respectively. His research interests focus on deep learning, applied machine learning, cloud and edge computing and computer networks.

Jia Hu is a Lecturer in Computer Science at the University of Exeter. He received his Ph.D. degree in Computer Science from the University of Bradford, UK, in 2010, and M.Eng. and B.Eng. degrees in Electronic Engineering from Huazhong University of Science and Technology, China, in 2006 and 2004, respectively. His research interests include mobile and ubiquitous computing, resource optimization, applied machine learning, and network security.

Geyong Min is a Professor of High Performance Computing and Networking in the Department of Computer Science within the College of Engineering, Mathematics and Physical Sciences at the University of Exeter, United Kingdom. He received the PhD degree in Computing Science from the University of Glasgow, United Kingdom, in 2003, and the B.Sc. degree in Computer Science from Huazhong University of Science and Technology, China, in 1995. His research interests include Computer Networks, Wireless Communications, Parallel and Distributed Computing, Ubiquitous Computing, Multimedia Systems, Modelling and Performance Engineering.

Zhiwei Zhao is currently an associate professor at the College of Computer Science and Engineering in University of Electronic Science and Technology of China. He received his PhD degree at the College of Computer Science, Zhejiang University in 2015. His research interests include edge computing, IoT systems, heterogeneous wireless networks, and protocol design. He is a member of IEEE, ACM and CCF.

Jin Wang is a Ph.D. student in Computer Science at the University of Exeter. He received both B.Eng. and M.Eng. degree in Computer Science from the University of Electronic Science and Technology of China (UESTC), Chengdu, China in 2014 and 2017, respectively. His research interests include deep reinforcement learning, applied machine learning, cloud and edge computing, and computer system optimization.