Large-scale Urban Cellular Traffic Generation via Knowledge-Enhanced GANs with Multi-Periodic Patterns

Shuodi Hui[1], Huandong Wang[1], Tong Li[1], Xinghao Yang[1],
Xing Wang[2], Junlan Feng[2], Lin Zhu[2], Chao Deng[2], Hui Pan[3] Depeng Jin[1], Yong Li[1]
[1]Beijing National Research Center for Information Science and Technology (BNRist),
Department of Electronic Engineering, Tsinghua University; [2]China Mobile Research, [3] Hong Kong University of Science and Technology

ABSTRACT

With the rapid development of the cellular network, network planning is increasingly important. Generating large-scale urban cellular traffic contributes to network planning via simulating the behaviors of the planned network. Existing methods fail in simulating the long-term temporal behaviors of cellular traffic while cannot model the influences of the urban environment on the cellular networks. We propose a knowledge-enhanced GAN with multi-periodic patterns to generate large-scale cellular traffic based on the urban environment. First, we design a GAN model to simulate the multi-periodic patterns and long-term aperiodic temporal dynamics of cellular traffic via learning the daily patterns, weekly patterns, and residual traffic between long-term traffic and periodic patterns step by step. Then, we leverage urban knowledge to enhance traffic generation via constructing a knowledge graph containing multiple factors affecting cellular traffic in the surrounding urban environment. Finally, we evaluate our model on a real cellular traffic dataset. Our proposed model outperforms three state-of-art generation models by over 32.77%, and the urban knowledge enhancement improves the performance of our model by 4.71%. Moreover, our model achieves good generalization and robustness in generating traffic for urban cellular networks without training data in the surrounding areas.

KEYWORDS

Cellular traffic, generation, knowledge graph, GAN

1 INTRODUCTION

Cellular networks are becoming an indispensable infrastructure of urban lives in recent years [28], which allow mobile users to access extensive network services anytime and anywhere. With the commercialization of the next-generation cellular networks (5G), the network planning problem, which aims to design how to deploy 5G base stations to satisfy the traffic requirements of mobile users, is becoming increasingly important.[1, 14]. Therefore, a critical and challenging problem arises: how would the planned cellular network perform?

Cellular traffic load across base stations plays an important role in assessing the planned cellular network performance in advance. To facilitate the planning of the 5G network, many researchers concentrate on modeling and simulating the urban cellular traffic [11, 21, 22, 27, 42]. For instance, Ma et al. [27] use social characteristics to model the daily traffic of cellular networks, which include traffic fluctuation, entropy, temporal homogeneity, and usage density. Ding et al. [11] generate synthetic base station traffic to model the daily dynamics of cellular traffic in large-scale urban mobile networks based on the measurements of real traffic data. Previous studies have failed to simulate the long-term temporal behaviors of cellular traffic. However, when it comes to the issue of cellular network planning, mobile network operators find the long-term traffic pattern much more valuable. This makes it easier to set up the cellular network and lets them see how well the planned cellular network works. For instance, long-term traffic can assist mobile network operators in designing a cellular network that takes future spatiotemporal traffic dynamics into account [35] and in reducing energy consumption by creating a Hyper Cellular Architecture (HCA) adaptable to traffic fluctuations [46].

For the purpose of cellular network planning, we propose to generate long-term urban cellular traffic in this paper. Given a target urban area and the planned cellular network infrastructure, we aim to generate the prospective traffic by leveraging the urban environment of the target area and the cellular network behavior patterns learned from other areas. Notably, long-term urban cellular traffic generation is challenging for the following reasons:

- **Complicated temporal variations.** As shown in Figure 1, complicated temporal variations exist in cellular traffic generally, for example, multi-scale periodicity [36] and burst phenomenon [8], leading to the coexistence of periodic correlation and aperiodic correlation of traffic in the temporal domain. Existing traffic generation methods [9, 13, 29] focus on generating short-term network traffic in packet- or flow-levels and cannot model such multi-scale temporal patterns and the long-term aperiodic temporal correlations.
- **Complicated urban environment.** The urban environment in which base stations are deployed, such as the regional function and important urban facilities, continues to have a significant impact on cellular traffic. Urban environments logically influence how mobile users behave online and their work rhythm, which results in a variety of cellular traffic dynamics characteristics. Many existing studies have documented this phenomenon [23, 36]. The multi-source urban data can be used to describe the urban environment [45]; however, it is challenging to effectively extract the features of the urban environment that characterize the diverse and cross-correlated influences between multi-source urban data.

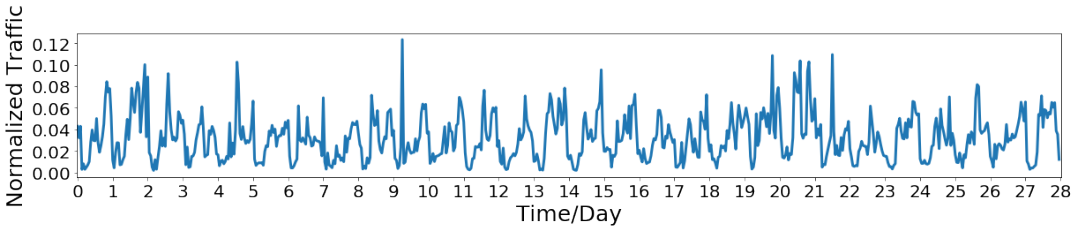


Figure 1: The real traffic of an example base station across one month.

To address the above challenges, the generation model should be capable of learning the multi-periodic patterns and long-term aperiodic temporal correlations while capturing the urban environment influence factors in urban cellular traffic. In this paper, we propose a knowledge-enhanced generative adversarial network (GAN) model that generates large-scale cellular traffic based on the urban environment. **First**, we design a GAN-based model to simulate multi-periodic patterns while capturing long-term aperiodic temporal correlations of cellular traffic. In our model, we design two GANs to learn the fundamental periodic patterns of cellular traffic and create daily and weekly patterns as different combinations of the fundamental patterns. Also, to capture long-term aperiodic temporal correlations of cellular traffic, we develop a GAN based on temporal convolutional networks (TCN), which satisfies temporal causality and retains the information in urban traffic across long time intervals. **Second**, we propose using a knowledge-based paradigm to characterize the cellular network's urban environment through knowledge graph (KG) techniques. A knowledge graph can extract structured knowledge from multi-source data and has shown success in a number of knowledge-based applications, e.g., user profiling [38], recommendation [31], and language understanding [25]. In our case, we construct an urban KG containing multiple factors affecting cellular traffic, including the geographical regions, business areas, point of interests (POIs), and their relations (e.g., attributes and affiliation). This allows us to model spatial dependencies and environmental semantics. We then devise a knowledge graph embedding (KGE) method to extract the urban environment's characteristics for cellular network traffic generation.

In summary, our contributions are summarized as follows:

- We propose a GAN model for long-term urban cellular traffic generation[1]. Our model can simulate cellular traffic while capturing both multi-periodic patterns and long-term aperiodic correlations.
- We build an urban KG containing multiple factors affecting cellular traffic to model the spatial dependencies and content semantics of base stations in the surrounding urban environment. A powerful KGE method is devised to learn urban knowledge to enhance the generation of urban network traffic in the urban environment.
- Extensive experiments on real-world urban traffic datasets demonstrate that our proposed model outperforms state-of-the-art baselines by over 32%, where the performance gain of 4.71% is contributed by the urban knowledge enhancement. For a certain target area, our proposed model that was trained on other areas still does well, which shows that our proposed model is good at generalization and is strong.

[1]Our codes: https://github.com/shirdy/TrafficGeneration/tree/master/Urban/

2 PROBLEM DEFINITION AND SYSTEM OVERVIEW

2.1 Problem Definition

In urban cellular networks, thousands of cellular base stations, which are also known as the cell sites or cellular towers, are providing network services via receiving and transmitting cellular traffic continuously. Hence, we focus on generating fine-grained cellular network traffic of base stations. For each cellular base station, we divide the network traffic in a period into equal intervals, then formally denote the traffic as $V = \{V_t\}_{t=1}^{T}$, where V_t represents the network traffic volume in the t_{th} interval, and T is the amount of these intervals. Concentrated on simulating the traffic variation of base stations, we normalize the traffic V with its two norm, which is represented by $\|V\|_2$. Hence the normalized traffic is denoted as $S = \{S_t\}_{t=1}^{T} = \{V_t/\|V\|_2\}_{t=1}^{T}$, where S_t represents the normalized network traffic in the t_{th} time interval. For the sake of convenience, we divide the traffic into hourly intervals, and T equals to the total number of hours. Accordingly, the urban cellular traffic generation problem can be expressed as follows:

Definition 1 (Urban Cellular Traffic Generation Problem). *Given the normalized traffic $\{S\}$ of the base stations in source areas, generating the prospective normalized traffic $\{\hat{S}\}$ of the base stations in the target area. In addition, urban environment information in the target area can be utilized in this process.*

To deal with the complicated temporal patterns in base station traffic [20, 32, 36, 39, 44], we define the daily pattern and weekly pattern as follows.

Definition 2 (Daily Pattern). *The daily pattern is the average traffic in each day during the total T hours, which is denoted as follows,*

$$S^d = \left\{S_t^d\right\}_{t=1}^{24} = \left\{\left(\sum_{i=0}^{\lfloor \frac{T}{24} \rfloor} S_{t+24i}\right) / \left\lfloor \frac{T}{24} \right\rfloor\right\}_{t=1}^{24}, \quad (1)$$

where S_t^d represents the traffic of the the t_{th} hour in daily pattern, and $\lfloor \frac{T}{24} \rfloor$ is the total number of days.

Definition 3 (Weekly Pattern). *The weekly pattern is the average traffic in each week during the total T hours, which is denoted as follows,*

$$S^w = \{S_t^w\}_{t=1}^{24*7} = \left\{ \left(\sum_{i=0}^{\lfloor \frac{T}{24*7} \rfloor} S_{t+24*7i} \right) / \left\lfloor \frac{T}{24*7} \right\rfloor \right\}_{t=1}^{24*7}, \quad (2)$$

where S_t^w represents the traffic of the the t_{th} hour in weekly pattern, and $\lfloor \frac{T}{24*7} \rfloor$ is the total number of weeks.

We plot the patterns and traffic of an example base station in Figure 1, where the traffic across days and weeks shows similar patterns.

2.2 System Overview

To address the challenges of capturing the complicated temporal patterns and the urban environment influence in cellular traffic, we design a GAN model to learn the long-term traffic behaviors and construct a KG to learn the urban environment for each base station. As shown in Figure 2, we input random noise into our generation model to introduce the randomness of base station traffic; the influence of urban environment is introduced into traffic generation via the KGE of each base station. Specifically, the same as other GAN models, our model can work with the input of noise only. In addition to the noise, KGE introduces prior knowledge as a part of the input, which provides conditions for the generation as a prior distribution. Therefore, using KGE or not is optional in the implementation of our model. In summary, with the given urban environment of base stations integrated by a KG, we design a GAN model to generate the corresponding daily patterns S^d, weekly patterns S^w, and long-term traffic S with the random noise and the urban KGE.

3 METHOD

We design a knowledge-enhanced GAN to generate the daily pattern, weekly pattern, and residual traffic between long-term traffic and periodic patterns step by step, where the noise and KGE are inputted to the GAN model for introducing randomness and urban environment information, as shown in Figure 4. For better understanding, we introduce the urban KG and GAN model as follows.

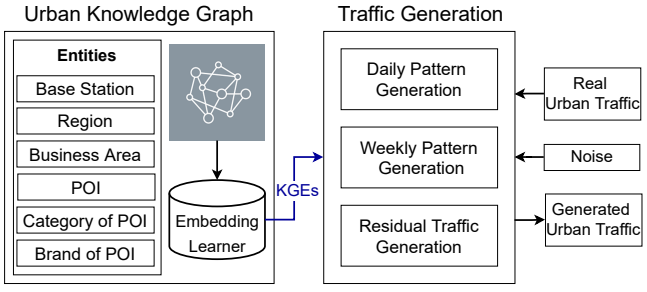


Figure 2: Overview of the urban cellular traffic generation problem.

3.1 Urban Knowledge Graph

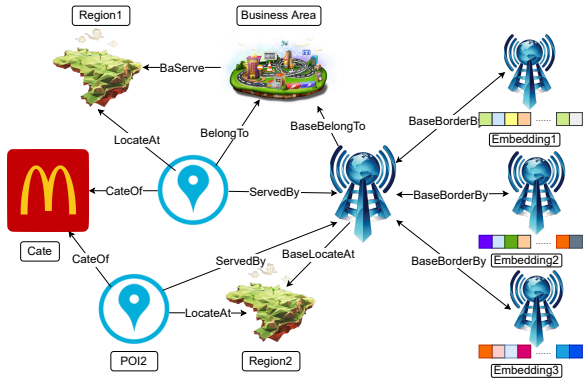


Figure 3: Illustration of the urban knowledge graph.

To model the urban environment, both spatial information and urban contents are significant. Urban contents refer to the meaningful things that belong to a city, for instance, administrative division, business area, bank, stores, residential area, etc. Inspired by the success of applying KGs in many domains, we construct a KG to model the urban contents as entities, and their spatial and semantic correlation can be modeled as relations. The entities fall in six categories, including base stations, POIs, regions, business areas, the categories of POI, and the brands of POI. Specifically, as shown in Figure 3, a base station is linked to other entities via four kinds of relation: 1) a base station is located in a region; 2) a base station belongs to a business area; 3) a POI is served by a base station; 4) a base station border by another base station. Particularly, as a POI can be served by many base stations within their service radius, we select the nearest five base stations to reduce the computational burden. Similarly, the nearest five adjacent base stations are retained for each base station. Moreover, we model the spatial and semantic correlation between other entities as relations, for instance, a POI belongs to a business area, and a business area serves a region. Finally, the KG includes twenty four kinds of relations. Details can be found in Appendix A.1, where Table 5 illustrates the entities and relations in our KG, and shows the amount of entities and triplets. Based on the constructed KG, we learn the embedding of each base station via TuckER [4]. TuckER is a tensor factorization method for KGE learning, outperforming a number of translation-based models (e.g., TransE [5]), bilinear models (e.g., ComplEx [33]), and neural network models (e.g., ConvE [10]) in practice.

3.2 Knowledge-Enhanced GAN

We design a knowledge-enhanced GAN to generate the long-term cellular traffic with multi-periodic patterns. Figure 4 illustrates the three stage of generation, including generating daily pattern, generating weekly pattern, and generating long-term traffic. Specifically, each generation stage is implemented by a delicately designed GAN, we input homologous noise and KGE to them, and they outputs the corresponding daily pattern, weekly pattern, and total traffic.

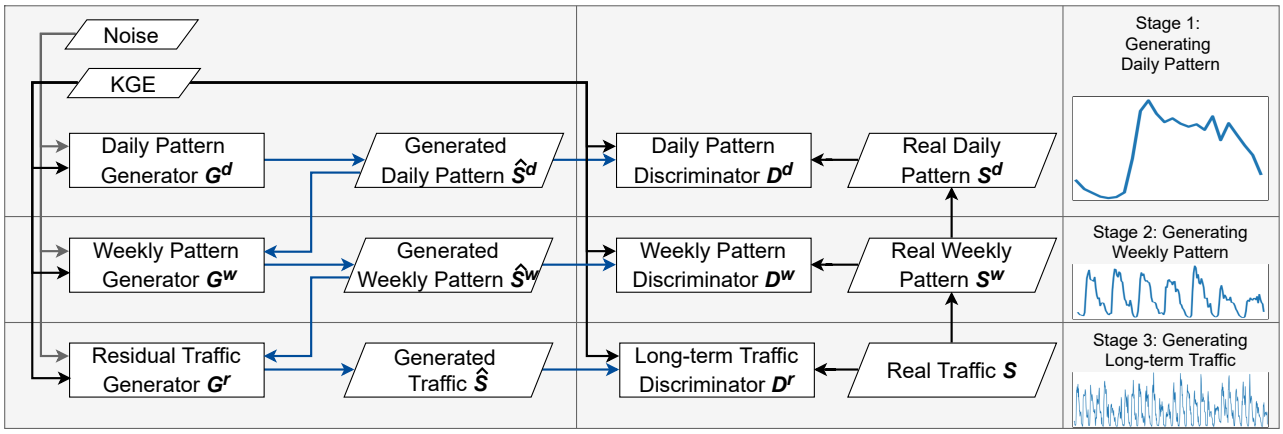


Figure 4: Cellular traffic generation via the knowledge-enhanced GAN.

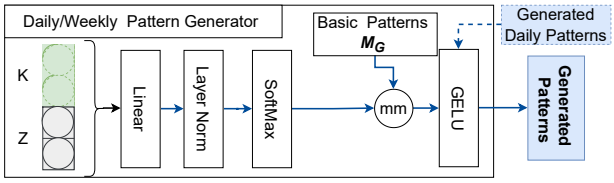


Figure 5: Illustration of the daily/weekly pattern generator.

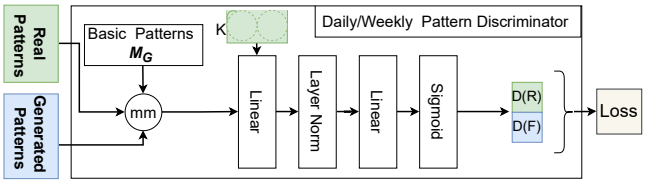


Figure 6: Illustration of the daily/weekly pattern discriminator.

3.2.1 Daily Pattern Generation. In the first stage, we generate the daily patterns in urban cellular traffic via learning the basic daily patterns and the combination methods. Primarily, we use a matrix to learn the basic daily patterns of the traffic in the generator G^d. The matrix can be denoted as M_G^d, in which each vector represents a basic daily pattern. Therefore, the daily pattern can be represented as the weighted sum of these basic pattern vectors. In the generator G^d, as illustrated in Figure 5, we use a multilayer perceptron (MLP) to learn the weight on each basic daily pattern vector from noise Z and KGE K , and use a softmax layer for normalization. Then, M_G^d is multiplied by the learned weights to generate the simulative daily pattern $\hat{S^d}$, which can be denoted as follows,

$$\hat{S}^d = G^d(Z; K^*) = \text{GELU}\left(\text{softmax}(\text{MLP}(Z; K^*))M_G^d\right), \quad (3)$$

where *GELU* is the activate function. Reciprocally, in the discriminator D^d illustrated in Figure 6, we calculate the projection of real daily pattern S^d or generated daily pattern $\hat{S^d}$ on the basic daily pattern matrix for discriminator (i.e., M_D^d), and use MLP to give the discrimination result on condition of K. As the traffic patterns in weekday and weekend show obvious differences, we generate the daily patterns in weekday and weekend respectively. Particularly, the KGE K acts as a condition variable in G^d and D^d, and the green dashed circles in Figure 5, 6 and the * in Equation 3 denotes that KGE K is optional for the input. Therefore, the GAN could operate normally both on condition of urban knowledge and without the knowledge. For weekly pattern and residual traffic generation, KGE K is also optional for the input.

3.2.2 Weekly Pattern Generation. In the second stage, we generate weekly patterns similar to the daily patterns. Differ from the daily pattern generator, the basic weekly pattern matrix M_G^w in the generator G^w for weekly pattern generation represents the weekly patterns with daily patterns removed, which helps to remove the interferences between weekly patterns and daily patterns in the generator. Therefore, in the generator, K and Z are mapped to the weights on the basic weekly patterns with daily patterns removed, then multiplied by M_G^w to generate the abridged weekly pattern with daily pattern removed. Then, as illustrated by the blue dashed box and line in Figure 5, we add the daily patterns in weekday and weekend generated by the daily pattern generator to the corresponding parts of abridged weekly pattern, and output the unabridged weekly pattern, which is denoted as follows,

$$\hat{S}^w = G^w(\hat{S}^d, Z; K^*) = \text{GELU}\left(\text{softmax}(\text{MLP}(Z; K^*))M_G^w + RP(\hat{S}^d)\right), \quad (4)$$

where $RP(\hat{S}^d)$ represents the repetition of generated daily patterns $\hat{S}^d$. In the discriminator D^w, the basic weekly patterns matrix M_D^w contains the unabridged basic weekly patterns, and the discrimination result is calculated similar to the daily pattern discriminator.

3.2.3 Residual Traffic Generation. With weekly pattern generated, our target becomes simulating the long-term aperiodic correlations by generating the residual traffic between long-term traffic and periodic patterns, which refers to the residual components of the total traffic after removing the weekly patterns in each week. We

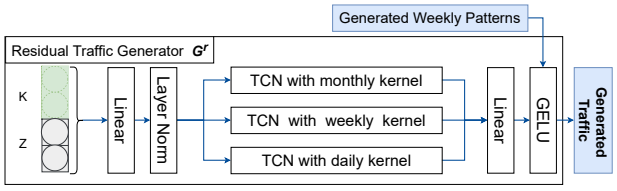


Figure 7: Illustration of the residual traffic generator.

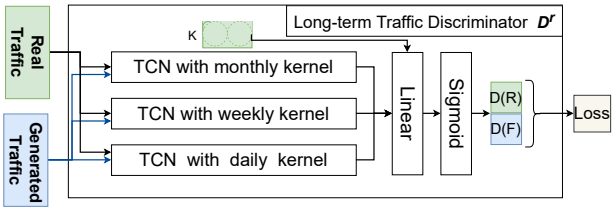


Figure 8: Illustration of the long-term traffic discriminator.

design a TCN-based GAN to capture the long-term aperiodic correlation via residual traffic, the generator G^r and the discriminator D^r are illustrated in Figure 7 and Figure 8 respectively. We adopt TCN [3] with different kernel sizes to capture the daily, weekly, and monthly correlations. TCN is a 1-D convolutional network that uses casual convolutions and dilated convolutions, including input layer, output layer, and several hidden layers. For each layer in the TCN with a daily kernel, the values are calculated based on values during the last day in the previous layer, which helps to capture the influence in previous days across multiple layers. Similarly, the TCNs with weekly and monthly kernels help to capture the influence in previous weeks or months. In the generator G^r, K and Z are mapped to the input layer of TCNs, and we use a linear layer to combine the three output sequences to generate the residual traffic. Then, the generated weekly pattern is added to the residual traffic to generate the total long-term traffic, which can be denoted as follows,

$$\begin{aligned}\hat{S} &= G^r\left(\hat{S}^w, Z; K^*\right) = \mathrm{GELU}\left(\hat{S}^r + RP(\hat{S}^w)\right) \\ &= \mathrm{GELU}\left(\mathrm{MLP}\left(\mathrm{TCNs}(\mathrm{MLP}(Z; K^*))\right) + RP(\hat{S}^w)\right),\end{aligned} \tag{5}$$

where $RP(\hat{S}^w)$ represents the repetition of generated weekly patterns $\hat{S}^w$. In the discriminator D^r, we input the simulative or real traffic to the TCNs with different kernel sizes. Then based on the output of TCNs, we use an MLP to give the discrimination result on condition of K.

3.2.4 Model Training and Loss. Figure 4 illustrates the three stage in our model training. In the first stage, we train the daily pattern generator G^d and daily pattern discriminator D^d to generate the daily pattern $\hat{S^d}$. In the second stage, we train the weekly pattern generator G^w and weekly pattern discriminator D^w to generate the weekly pattern $\hat{S^w}$ based on the output daily pattern $\hat{S^d}$ in the first stage. In the third stage, we train the residual traffic generator G^r and total traffic discriminator D^r to generate the total traffic $\hat{S}$ based on the output weekly pattern $\hat{S^w}$ in the second stage. In each stage, we use Wasserstein distance [2] with gradient penalty [15] in our model, which is demonstrated to be effective in performance improvement for GANs. The loss function is as follows,

$$L = \mathop{\mathbb{E}}_{\hat{S}\sim\mathbb{P}_G}[D(\hat{S})] - \mathop{\mathbb{E}}_{S\sim\mathbb{P}_R}[D(S)] + \lambda \mathop{\mathbb{E}}_{\tilde{S}\sim\mathbb{P}_{\tilde{S}}}\left[(\left\|\nabla_{\tilde{S}} D(\tilde{S})\right\|_2 - 1)^2\right], \tag{6}$$

where $D(S)$ is the discrimination results of real samples S, and $\mathbb{P}_R$ represents the real data distribution, $D(\hat{S})$ is the discrimination results of generated samples $\hat{S}$, and $\mathbb{P}_G$ represents the generator distribution, $\tilde{S}$ are samples uniformly along straight lines between pairs of objects sampled from the real and generated data, $D(\tilde{S})$ is its discrimination results and $\mathbb{P}_{\tilde{S}}$ represents its distribution. In each stage, the generator is trained to minimize the loss, while the discriminator is trained to maximize it.

Particularly, we pre-train the basic daily pattern matrix M^d and the basic weekly pattern matrix M^w to improve the training efficiency. For the basic daily pattern matrix M^d, we calculate the real daily patterns on the training set, cluster them into N clusters via K-means algorithm [19], and take the average of daily patterns in cluster i as the initial value of i_{th} basic daily pattern vectors C_i^d. Hence $M^d = [C_1^d, C_2^d, ..., C_N^d]$, where each C_i^d represents a basic daily pattern, and N is the total number of basic daily patterns. Simultaneously, the initial value of the basic weekly pattern matrix M^w can be calculated based on the clustering results of real weekly patterns.

4 EXPERIMENTS

We conduct experiments on a real cellular dataset to answer the following three research questions:

- **RQ1:** How does our proposed model perform compared with the state-of-the-art(SOTA) models in cellular traffic generation?
- **RQ2:** Can the proposed model leverage urban knowledge to improve the cellular traffic generation?
- **RQ3:** How is the generalization and robustness of our proposed model? Specifically, for urban cellular networks without training data in the surrounding areas, can the proposed model achieve good performance?

We first describe the experiment settings, then answer the three research questions as follows.

4.1 Experiment Setting

4.1.1 Dataset. We acquire a cellular traffic dataset collected by an ISP from Shanghai, a big city in China. The dataset contains the network traffic records of 5,326 base stations all over Shanghai, which are collected between Aug 1st and Aug 28th 2014. As shown in Table 1, we divide the dataset into three sub-datasets according to the location of base stations, including the center area, suburb area, and outer suburb area.

4.1.2 Baselines. We compare our proposed model with the following three baselines. Notably, to answer RQ2, we evaluate each model with two different kinds of inputs: 1) urban KGEs K and noise vectors Z, 2) only noise vectors Z.

TransGAN [18]. TransGAN is a transformer-based GAN framework, which consists of a generator with several transformer blocks that progressively increases feature resolution, and correspondingly a multi-scale discriminator to capture simultaneously semantic contexts and low-level textures. For cellular traffic generation, we adjust the scales of transformer blocks to capture the multi-scale temporal patterns.

RNN-based GAN. Long short term memory (LSTM)[16] is an RNN architecture that is known for memorizing history values over arbitrary intervals. We construct a GAN using two LSTMs as the generator and discriminator. The input for the generator is the KGE K and noise Z, while the output is the generated cellular traffic. Then, the generated and real traffic data are input to the discriminator to give discriminating results.

TCN-based GAN. We construct a GAN using several TCNs[3] as the generator and discriminator, which is similar to the residual traffic generator and long-term traffic discriminator in our framework shown in Figure 7 and Figure 8. The difference is that the KGE K and noise Z are directly mapped to the total traffic of base stations through TCNs, linear layers, and normalization layers in the generator.

4.1.3 Metrics. We evaluate our model via the following four metrics: Traffic Volume, First-order Difference, Daily Frequency Component, and Weekly Frequency Component. Detailed descriptions of metrics can be found in Appendix A.2.

4.1.4 Parameter Setting. To keep the balance between computation complexity and representational capacity, we set the dimensions of KGEs and noise vectors to 32. Then, we train our model and baseline models on each sub-dataset with 300 iterations, and test the trained model on the other two sub-datasets. Specifically, as the inputs of these trained generation models include both KGE K and random noise Z, we apply each trained model on each testing sub-dataset for 20 times with random noise Z initialized from different seeds. Moreover, to evaluate the effectiveness of urban knowledge, we remove KGE K from the inputs of each generation model during both training and testing to apply contrast tests. Finally, we compute the metrics on generated and real data and calculate each metric's mean value and standard deviation.

4.2 Performance Comparison (RQ1)

We compare the performance of our proposed model with the other models via the metrics of traffic volume distribution, first-order difference distribution, daily periodicity, and weekly periodicity. Generally, our model outperforms the other models. Table 2 presents the evaluation results of the cellular traffic generated by different models, where 'Trans' represents the TransGAN model, 'RNN' represents the RNN-based GAN model, 'TCN' represents the TCN-based GAN model, and '+K' represents urban knowledge enhancement. For the JSD of traffic volume, our model gives the best result, which indicates that our model outperforms the other models on capturing the traffic volume distribution of cellular traffic. For the JSD of the first-order differences, our model gives the second-best result while RNN-based GAN model gives the best result, because RNN is good at capturing short-term variation. For daily periodicity, our model outperforms all the other models, which proves that we can successfully learn the daily temporal patterns in cellular traffic. For weekly periodicity, our model reaches a result close to the best. In practice, we find that not all base stations show obvious weekly traffic pattern, indicating that the weekly frequency component can show similar intensity with other frequency components, and can be easily disturbed by noise in frequency domain. Overall, our model outperforms baseline models by 32.77% at least.

Table 1: The three areas and corresponding base station numbers in our dataset.

Area	Center	Suburb	Outer Suburb
District Number	7	4	5
Base Station Number	1665	2480	1181

To present the performance of our model visually, we compare the real and generated temporal patterns of a selected base station in Figure 9. Compared with the best baseline model, i.e., RNN-based GAN, our model generates more realistic daily and weekly patterns. Our generated data are more similar to the real data shown in Figure 1. In contrast, TransGAN and TCN-based model can hardly capture the daily patterns, and RNN-based model generates traffic with only tiny fluctuations. Detailed descriptions can be found in Appendix A.3.

In summary, our proposed model performs better compared with the SOTA baseline models in cellular traffic generation, which prove the capacity of capturing multi-periodic patterns and long-term aperiodic correlations for our model.

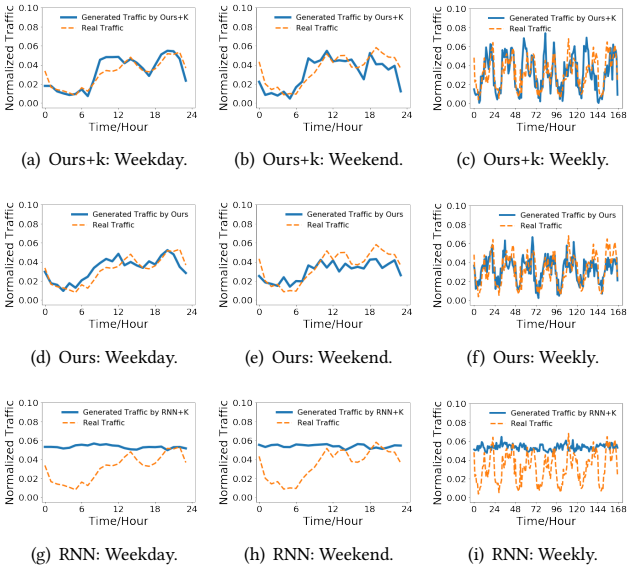
4.3 Urban Knowledge Effect (RQ2)

To evaluate the effect of urban knowledge, we conduct an experiment of predicting the clusters of base stations via their KGEs to verify whether the learned KGEs contain any urban information at first. Considering the surrounding urban functional regions, the base stations can be classified into five clusters: resident, transport, office, entertainment, and comprehensive [36]. We split the real dataset into training and testing set accounting for 50% and 50%, train several classifiers on the training set and apply the trained model to the testing set. To control the impact of classification methods, the experiment includes five common-used classification algorithms: k-nearest neighbors (kNN), logistic regression (LR), random forest (RF), MLP, and multinomial naive Bayes (NB). Table 3 presents that most of the classification results are greater than 65%, indicating that the learned KGEs contain the information of the surrounding urban functional regions, e.g., the type of its urban function. Moreover, the learned KGEs may also contain other kinds of urban information.

We then prove that the learned KGEs can improve the cellular traffic generation in our model. As discussed in Section 2.2 and Section 3.2, the KGEs are inputted to our proposed model and other baseline GAN models as optional condition variables, and the GAN models operate normally both with and without knowledge. We compare the performances of the models on condition of urban knowledge and without the knowledge. Table 2 shows that most of the generation models with knowledge enhanced perform better

Table 2: Evaluation results of cellular traffic generated by different models, where lower results are better. Bold denotes the best(lowest) results and underline denotes the second-best results.

Metrics	Traffic Volume		First-order Difference		Daily Frequency Component		Weekly Frequency Component		$\overline{\Delta}$
	JSD	Δ	JSD	Δ	RMSE	Δ	RMSE	Δ	
Trans	0.5025 ± 0.0070	74.53%	0.1378 ± 0.0029	85.08%	0.0427 ± 0.0048	112.28%	0.0085 ± 0.0009	49.78%	80.42%
Trans+K	0.4948 ± 0.0151	71.87%	0.1075 ± 0.0066	44.36%	0.0384 ± 0.0050	91.06%	**0.0049 ± 0.0006**	−12.51%	48.69%
RNN	0.5748 ± 0.0509	99.65%	0.0813 ± 0.0408	9.19%	0.0477 ± 0.0108	137.04%	0.0063 ± 0.0018	12.52%	64.60%
RNN+K	0.4317 ± 0.0511	49.94%	**0.0631 ± 0.0169**	−15.21%	0.0388 ± 0.0050	92.79%	0.0058 ± 0.0010	3.55%	32.77%
TCN	0.4099 ± 0.1799	42.37%	0.0747 ± 0.0304	0.31%	0.0401 ± 0.0048	99.38%	0.0061 ± 0.0009	8.83%	37.72%
TCN+K	0.4241 ± 0.1137	47.29%	0.1024 ± 0.0360	37.56%	0.0407 ± 0.0049	102.52%	0.0068 ± 0.0010	20.15%	51.88%
Ours	0.3072 ± 0.1323	6.70%	0.0850 ± 0.0551	14.17%	0.0211 ± 0.0075	4.96%	0.0052 ± 0.0009	−6.99%	4.71%
Ours+K	**0.2879 ± 0.0401**	0	0.0744 ± 0.0323	0	**0.0201 ± 0.0073**	0	0.0056 ± 0.0007	0	0



(a) Ours+k: Weekday. (b) Ours+k: Weekend. (c) Ours+k: Weekly.

(d) Ours: Weekday. (e) Ours: Weekend. (f) Ours: Weekly.

(g) RNN: Weekday. (h) RNN: Weekend. (i) RNN: Weekly.

Figure 9: Comparison between the real temporal patterns and the generated temporal patterns, according to the weekday patterns, weekend patterns, and weekly patterns in each column. (a,b,c) Temporal patterns generated by our knowledge-enhanced model, (d,e,f) temporal patterns generated by our model without knowledge, (g,h,i) temporal patterns generated by knowledge-enhanced RNN-based GAN.

Table 3: Results of base station cluster prediction via KGE.

Model	kNN	LR	RF	MLP	NB
Accuracy	0.6432	0.6834	0.6801	0.5777	0.6879

than models without knowledge, which demonstrates the effectiveness of the learned KGEs for cellular traffic generation. For our model, introducing urban KGEs could improve the performance by 4.71%. For all the models in average, the urban knowledge enhancement improves the performances by 13.53%. In addition, Figure 9 plots the traffic generated by our model with urban knowledge removed, which performs weaker than our knowledge-enhanced model on capturing daily and weekly patterns, yet better than the knowledge-enhanced baseline models.

In summary, our proposed model can leverage urban knowledge to improve the cellular traffic generation.

4.4 Generalization and Robustness (RQ3)

To verify the generalization and robustness, we train our model on each area, and apply the trained models to all the areas. Table 4 shows the evaluation results, where 'O-C' represents applying the model trained on the cellular traffic in the outer suburb area to generate cellular traffic in the center area. Models trained on the cellular traffic in the suburb area perform best, which can be explained by the sizes of training sets. As the number of base stations in the suburb area is more than the other areas, it has the largest training set, the models trained on it perform best accordingly. The results show that for the cellular traffic in a certain target area, models trained on the cellular traffic in other areas perform no worse than models trained on itself generally, illustrating that our proposed model can achieve good performance for urban cellular networks without training data in the surrounding areas. Therefore, our proposed model is capable of generalizing the long-term temporal behaviors and urban knowledge effect mechanisms learned from the cellular traffic data in the training areas to the other areas, and operates robustly across different areas.

In summary, our proposed model achieves good generalization and robustness on generating traffic for urban cellular networks without training data in the surrounding areas.

5 RELATED WORK

Our work assists mobile network planning and optimization by simulating cellular traffic via delicately designing a knowledge-enhanced GAN with multi-peridoc patterns. To summarize the related works for our work, we introduce the representative researches on network traffic generation and the applications of KG.

Table 4: Evaluation results of cellular traffic in each area generated by our model trained on different areas, where lower results are better. Bold denotes the best (lowest) results and underline denotes the second-best results.

Metrics	Traffic Volume		First-order Difference		Daily Frequency Component		Weekly Frequency Component		$\overline{\Delta}$
	JSD	Δ	JSD	Δ	RMSE	Δ	RMSE	Δ	
O-C	0.3348 ± 0.0123	37.95%	0.0915 ± 0.0270	82.24%	0.0388 ± 0.00005	171.72%	0.0095 ± 0.00003	84.14%	94.01%
O-S	0.3311 ± 0.0125	36.41%	0.0940 ± 0.0129	87.17%	0.0323 ± 0.00004	126.17%	0.0088 ± 0.00002	70.54%	80.07%
O-O	0.3178 ± 0.0131	30.93%	0.1002 ± 0.0196	99.50%	0.0262 ± 0.00004	83.07%	0.0063 ± 0.00004	23.25%	59.20%
C-C	0.2773 ± 0.0203	14.24%	0.0707 ± 0.0088	40.88%	0.0295 ± 0.00007	106.57%	0.0076 ± 0.00005	46.82%	52.13%
C-S	0.2812 ± 0.0259	15.85%	0.0767 ± 0.0226	52.71%	0.0234 ± 0.00006	64.03%	0.0073 ± 0.00005	42.67%	42.70%
C-O	0.2714 ± 0.0338	11.80%	0.0901 ± 0.0279	79.54%	0.0177 ± 0.00007	24.05%	0.0056 ± 0.00006	9.42%	31.20%
S-C	0.2662 ± 0.0199	9.67%	0.0441 ± 0.0202	-12.19%	0.0201 ± 0.00006	40.32%	0.0074 ± 0.00003	43.78%	20.39%
S-S	**0.2416 ± 0.0318**	-0.46%	**0.0425 ± 0.0261**	-15.40%	0.0164 ± 0.00006	14.93%	0.0071 ± 0.00005	36.91%	9.00%
S-O	0.2427 ± 0.0154	0	0.0502 ± 0.0323	0	**0.0143 ± 0.00007**	0	**0.0051 ± 0.00007**	0	0

5.1 Network Traffic Generation

In the early stage, network traffic generation is solved by network traffic models [34, 40] and applied to test network equipment, network services, and security products [43]. Recently, machine learning methods are applied to network traffic generation, e.g., autoregressive models [6, 41]. As a state-of-art generative model, GAN also becomes popular for network traffic generation [9]. Ring et al. [29] generate network traffic flows via three GAN-based preprocessing approaches, and Dowoo et al. [13] generate pcap files via the PcapGAN model trained on cyber attack data and normal data. However, the above GAN models focus on generating traffic flows or packets of individual entities in the network, requiring detailed configurations and parameters (e.g., network protocols, IP addresses, etc), which is inapplicable for collective network traffic generation tasks like urban cellular traffic generation. Moreover, Lin et al. [24] generate traffic forms data via the DoppelGANger model, which can generate data attributes and feature series simultaneously. Nevertheless, urban cellular traffic data contains only the one-dimensional feature and no attributes, for which their model is inapplicable.

These generation models for network traffic data provide us experience to generate cellular traffic. Based on these models, our proposed model generates both the multi-periodic patterns and the long-term aperiodic correlations of cellular traffic data enhanced by urban knowledge.

5.2 Knowledge Graph Application

KG is widely used in multiple real-world AI applications by injecting rich structured knowledge to improve representation learning, such as natural language understanding (NLU), question answering, and recommendation systems [17]. For example on NLU, Liu et al. [26] infuse domain knowledge into BERT contextual encoder, Sun et al. [30] introduce named entity masking and phrase masking to integrate knowledge into the continual multitask learning language model. For single-fact QA, Chen et al. [7] propose BAMnet to model the two-way interaction between questions and KG with a bidirectional attention mechanism. For multi-hop reasoning, Ding et al. [12] propose CogQA to combine implicit extraction and explicit reasoning and construct a cognitive graph model based on BERT and GNN for multi-hop QA. For recommend system, Wang et al. [37] propose MKR to associate multitask representation and recommendation by sharing latent features and modeling high-order item-entity interaction.

The above works demonstrate the effectiveness of KGs in various application scenarios. Inspired by these applications, we construct the urban KG to describe the urban environment for the base stations, and introduce the urban KGEs into traffic generation.

6 CONCLUSION

In this paper, we propose a knowledge-enhanced GAN to generate urban cellular traffic. First, we learn the multi-periodic patterns and long-term aperiodic correlations via daily patterns, weekly patterns and residual traffic step by step. Then, we utilize urban knowledge to enhance traffic generation by constructing an urban KG containing multiple factors affecting cellular traffic in the surrounding urban environment. We evaluate our proposed model on a real traffic dataset, where our model outperforms the state-of-art generation models at by over 32.77% in terms of key fidelity metrics, and the urban knowledge enhancement improves the performance of our model by 4.71% Moreover, traffic generation performances on urban cellular networks without training data in the surrounding areas demonstrate the generalization and robustness of our proposed model.

We have released our codes, the trained generation models, and the generated urban cellular traffic data on Github to support the reproducibility. Our proposed model can also generate all kinds of traffic data with multi-periodic patterns and long-term aperiodic correlations, for instance, website visiting and urban passenger traffic. We believe this work promotes further studies of the traffic generation problem. One limitation of our work is conducting cross-area experiments in the same city instead of cross-city experiments. Therefore, we plan to collect urban cellular traffic data and construct urban KGs in more cities for cross-city experiments in future work.

REFERENCES

[1] Md Alam et al. 2013. Mobile network planning and kpi improvement.

[2] Martin Arjovsky and Léon Bottou. 2017. Towards principled methods for training generative adversarial networks. *arXiv preprint arXiv:1701.04862* (2017).

[3] Shaojie Bai, J Zico Kolter, and Vladlen Koltun. 2018. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271* (2018).

[4] Ivana Balažević, Carl Allen, and Timothy Hospedales. 2019. TuckER: Tensor Factorization for Knowledge Graph Completion. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 5185–5194.

[5] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems* 26 (2013).

[6] Alisson Assis Cardoso and Flávio Henrique Teles Vieira. 2019. Generation of Synthetic Network Traffic Series Using a Transformed Autoregressive Model Based Adaptive Algorithm. *IEEE Latin America Transactions* 17, 08 (2019), 1268–1275.

[7] Yu Chen, Lingfei Wu, and Mohammed J Zaki. 2019. Bidirectional attentive memory networks for question answering over knowledge bases. *arXiv preprint arXiv:1903.02188* (2019).

[8] Zheng Chen and Ling Qiu. 2014. Analysis and Optimization of Cellular Network with Burst Traffic. *arXiv preprint arXiv:1408.3942 v1 [cs. IT] 18 Aug 2014* (2014).

[9] Adriel Cheng. 2019. Pac-gan: Packet generation of network traffic using generative adversarial networks. In *2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*. IEEE, 0728–0734.

[10] Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2018. Convolutional 2d knowledge graph embeddings. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

[11] Jingtao Ding, Rui Xu, Yong Li, Pan Hui, and Depeng Jin. 2017. Measurement-driven modeling for connection density and traffic distribution in large-scale urban mobile networks. *IEEE Transactions on Mobile Computing* 17, 5 (2017), 1105–1118.

[12] Ming Ding, Chang Zhou, Qibin Chen, Hongxia Yang, and Jie Tang. 2019. Cognitive graph for multi-hop reading comprehension at scale. *arXiv preprint arXiv:1905.05460* (2019).

[13] Baik Dowoo, Yujin Jung, and Changhee Choi. 2019. PcapGAN: Packet Capture File Generator by Style-Based Generative Adversarial Networks. In *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*. IEEE, 1149–1154.

[14] Ali Ehsan and Qiang Yang. 2019. State-of-the-art techniques for modelling of uncertainties in active distribution network planning: A review. *Applied energy* 239 (2019), 1509–1523.

[15] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. 2017. Improved training of wasserstein gans. *arXiv preprint arXiv:1704.00028* (2017).

[16] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[17] Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen, and S Yu Philip. 2021. A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE Transactions on Neural Networks and Learning Systems* (2021).

[18] Yifan Jiang, Shiyu Chang, and Zhangyang Wang. 2021. Transgan: Two transformers can make one strong gan. *arXiv preprint arXiv:2102.07074* (2021).

[19] K Krishna and M Narasimha Murty. 1999. Genetic K-means algorithm. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 29, 3 (1999), 433–439.

[20] Varun Kurri, Vishweshvaran Raja, and P Prakasam. 2021. Cellular traffic prediction on blockchain-based mobile networks using LSTM model in 4G LTE network. *Peer-to-Peer Networking and Applications* 14, 3 (2021), 1088–1105.

[21] Dongheon Lee, Sheng Zhou, Xiaofeng Zhong, Zhisheng Niu, Xuan Zhou, and Honggang Zhang. 2014. Spatial modeling of the traffic density in cellular networks. *IEEE Wireless Communications* 21, 1 (2014), 80–88.

[22] Chao Li, Abbas Yongacoglu, and Claude D'Amours. 2015. Mixed spatial traffic modeling of heterogeneous cellular networks. In *2015 IEEE International Conference on Ubiquitous Wireless Broadband (ICUWB)*. IEEE, 1–5.

[23] Tong Li, Yong Li, Tong Xia, and Pan Hui. 2021. Finding spatiotemporal patterns of mobile application usage. *IEEE Transactions on Network Science and Engineering* (2021).

[24] Zinan Lin, Alankar Jain, Chen Wang, Giulia Fanti, and Vyas Sekar. 2020. Using GANs for Sharing Networked Time Series Data: Challenges, Initial Promise, and Open Questions. *Proceedings of the ACM Internet Measurement Conference*, 464–483.

[25] Weijie Liu, Peng Zhou, Zhe Zhao, Zhiruo Wang, Qi Ju, Haotang Deng, and Ping Wang. 2020. K-bert: Enabling language representation with knowledge graph. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, Vol. 34. 2901–2908.

[26] Weijie Liu, Peng Zhou, Zhe Zhao, Zhiruo Wang, Qi Ju, Haotang Deng, and Ping Wang. 2020. K-bert: Enabling language representation with knowledge graph. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 2901–2908.

[27] Ji Ma, Wei Ni, Jie Yin, Ren Ping Liu, Yuyu Yuan, and Binxing Fang. 2016. Modeling Mobile Cellular Networks Based on Social Characteristics. *INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL* 11, 4 (2016), 480–492.

[28] R Müller et al. 2020. Ericsson mobility report. *Ericsson, Jun* (2020).

[29] Markus Ring, Daniel Schlör, Dieter Landes, and Andreas Hotho. 2019. Flow-based network traffic generation using generative adversarial networks. *Computers & Security* 82 (2019), 156–172.

[30] Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Hao Tian, Hua Wu, and Haifeng Wang. 2020. Ernie 2.0: A continual pre-training framework for language understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 8968–8975.

[31] Zhu Sun, Jie Yang, Jie Zhang, Alessandro Bozzon, Long-Kai Huang, and Chi Xu. 2018. Recurrent knowledge graph embedding for effective recommendation. In *Proceedings of the 12th ACM Conference on Recommender Systems*. 297–305.

[32] Hoang Duy Trinh, Nicola Bui, Joerg Widmer, Lorenza Giupponi, and Paolo Dini. 2017. Analysis and modeling of mobile traffic using real traces. In *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. IEEE, 1–6.

[33] Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2016. Complex embeddings for simple link prediction. In *International conference on machine learning*. PMLR, 2071–2080.

[34] Kashi Venkatesh Vishwanath and Amin Vahdat. 2009. Swing: Realistic and responsive network traffic generation. *IEEE/ACM Transactions on Networking* 17, 3 (2009), 712–725.

[35] Gang Wang, Yi Zhong, Rongpeng Li, Xiaohu Ge, Tony QS Quek, and Guoqiang Mao. 2020. Effect of spatial and temporal traffic statistics on the performance of wireless networks. *IEEE Transactions on Communications* 68, 11 (2020), 7083–7097.

[36] Huandong Wang, Fengli Xu, Yong Li, Pengyu Zhang, and Depeng Jin. 2015. Understanding mobile traffic patterns of large scale cellular towers in urban environment. In *Proceedings of the 2015 Internet Measurement Conference*. 225–238.

[37] Hongwei Wang, Fuzheng Zhang, Miao Zhao, Wenjie Li, Xing Xie, and Minyi Guo. 2019. Multi-task feature learning for knowledge graph enhanced recommendation. In *The World Wide Web Conference*. 2000–2010.

[38] Pengyang Wang, Kunpeng Liu, Lu Jiang, Xiaolin Li, and Yanjie Fu. 2020. Incremental mobile user profiling: Reinforcement learning with spatial knowledge graph for modeling event streams. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 853–861.

[39] Xu Wang, Zimu Zhou, Fu Xiao, Kai Xing, Zheng Yang, Yunhao Liu, and Chunyi Peng. 2018. Spatio-temporal analysis and prediction of cellular traffic in metropolis. *IEEE Transactions on Mobile Computing* 18, 9 (2018), 2190–2202.

[40] Michele C Weigle, Prashanth Adurthi, Félix Hernández-Campos, Kevin Jeffay, and F Donelson Smith. 2006. Tmix: a tool for generating realistic TCP application workloads in ns-2. *ACM SIGCOMM Computer Communication Review* 36, 3 (2006), 65–76.

[41] Shengzhe Xu, Manish Marwah, and Naren Ramakrishnan. 2020. STAN: Synthetic Network Traffic Generation using Autoregressive Neural Models. *arXiv preprint arXiv:2009.12740* (2020).

[42] Chaoyun Zhang, Xi Ouyang, and Paul Patras. 2017. ZipNet-GAN: Inferring fine-grained mobile traffic patterns via a generative adversarial neural network. In *Proceedings of the 13th International Conference on emerging Networking EXperiments and Technologies*. 363–375.

[43] Junhui Zhang, Jiqiang Tang, Xu Zhang, Wen Ouyang, and Dongbin Wang. 2015. A survey of network traffic generation. (2015).

[44] Mingyang Zhang, Haohao Fu, Yong Li, and Sheng Chen. 2017. Understanding urban dynamics from massive mobile traffic data. *IEEE Transactions on Big Data* 5, 2 (2017), 266–278.

[45] Yu Zheng, Licia Capra, Ouri Wolfson, and Hai Yang. 2014. Urban computing: concepts, methodologies, and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)* 5, 3 (2014), 1–55.

[46] Zhengteng Zhu, Xiao Xu, Xi Zheng, Yuxuan Sun, Sheng Zhou, Jie Gong, and Zhisheng Niu. 2015. A simulation study of hyper-cellular architecture with dynamic temporal and spatial traffic. In *2015 21st Asia-Pacific Conference on Communications (APCC)*. IEEE, 126–131.

A APPENDIX

A.1 Urban Knowledge Graph Schema

Urban is an important part of our research which inspire us to build a knowledge graph to introduce the urban information for our task. Table 5 lists the entity types and shows the subject and object of each kind of relation. These entities and relations can describe both semantic knowledge and interaction features.

A.2 Evaluation Metrics

Traffic Volume. We evaluate the distribution of traffic volume for the generated cellular traffic by comparing with the real distribution. Jensen–Shannon divergence (JSD) is a commonly used metric to describe the similarity between two distributions, which is defined as,

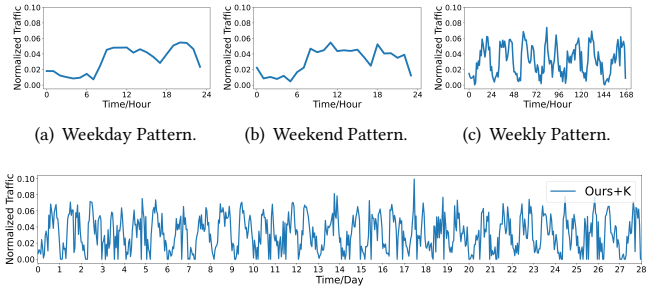
$$\mathrm{JSD}(\mathbf{P}_{\mathrm{G}}, \mathbf{P}_{\mathrm{R}}) = \sqrt{\frac{\mathbf{KL}(\mathbf{P}_{\mathrm{G}} \| \mathbf{P}_{\{\tilde{\mathrm{S}}\}}) + \mathbf{KL}(\mathbf{P}_{\{\tilde{\mathrm{S}}\}} \| \mathbf{P}_{\mathrm{R}})}{2}}, \tag{7}$$

where $\mathbf{P}_R$ represents the real data distribution, $\mathbf{P}_G$ is the generator distribution, $\mathbf{P}_{\{\tilde{S}\}}$ represents the point-wise mean of $\mathbf{P}_R$ and $\mathbf{P}_G$, and **KL** is the Kullback-Leibler divergence. We calculate the JSD between the distribution of traffic volume in each generated dataset $\{\hat{S}\}$ and the corresponding real sub-dataset $\{S\}$, which can be denoted as $\mathrm{JSD}(\mathbf{P}_{\{\hat{S}\}}, \mathbf{P}_{\{S\}})$, and a lower JSD means a closer distribution to the real data, which indicates a better generation model.

First-order Difference. To evaluate the variation in each generated traffic series $\hat{S}$, we compute the first-order difference series for $\hat{S}$, which can be denoted as $\hat{S}' = \{\hat{S}_{t+1} - \hat{S}_t\}_{t=1}^{N-1}$. Then, the first-order differences of generation dataset and real sub-dataset can be denoted as $\{\hat{S}'\}$ and $\{S'\}$ respectively. We calculate the JSD between the first-order differences of each generated dataset and the corresponding real sub-dataset, which can be denoted as $\mathrm{JSD}(\mathbf{P}_{\{\hat{S}'\}}, \mathbf{P}_{\{S'\}})$.

Daily Frequency Component. We evaluate the daily periodicity of the generated traffic via calculating daily frequency component. Firstly, we compute the frequency spectrum of each generation cellular traffic series $\hat{S}$, which is denoted as $\hat{F} = FFT(\hat{S})$. The proportion of daily frequency component can be calculated by $\hat{F}^d = \left\|\hat{F}\left[\left\lfloor\frac{N}{24}\right\rfloor\right]\right\|_2 / \|\hat{F}\|_2$, where $\|\hat{F}\|_2$ is the two norm of total frequency spectrum, and $\left\|\hat{F}\left[\left\lfloor\frac{N}{24}\right\rfloor\right]\right\|_2$ is the two norm of daily frequency component. Then, for each generation cellular traffic series $\hat{S}$, we compute the root-mean-square error (RMSE) to the corresponding real cellular series S on daily frequency component, which can be denoted as $\mathrm{RMSE}(\hat{\mathrm{F}}^{\mathrm{d}}, \mathrm{F}^{\mathrm{d}}) = \sqrt{(\hat{\mathrm{F}}^{\mathrm{d}} - \mathrm{F}^{\mathrm{d}})^2}$.

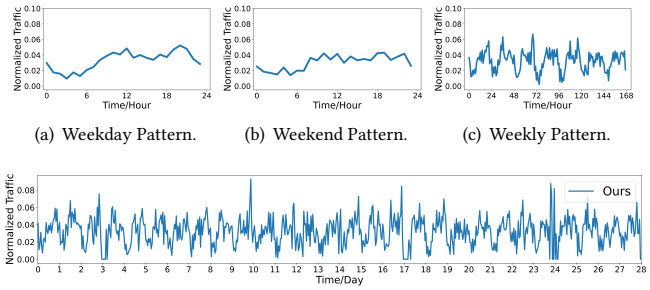
Weekly Frequency Component. We evaluate the weekly periodicity of the generated traffic via calculating weekly frequency component, the proportion of which can be denoted as $\hat{F}^w = \left\|\hat{F}\left[\left\lfloor\frac{N}{24*7}\right\rfloor\right]\right\|_2 / \|\hat{F}\|_2$, where $\left\|\hat{F}\left[\left\lfloor\frac{N}{24*7}\right\rfloor\right]\right\|_2$ is the two norm of weekly frequency component. Similarly, we compare the generated and real weekly frequency component via computing $\mathrm{RMSE}(\hat{\mathrm{F}}^{\mathrm{w}}, \mathrm{F}^{\mathrm{w}}) = \sqrt{(\hat{\mathrm{F}}^{\mathrm{w}} - \mathrm{F}^{\mathrm{w}})^2}$.



(a) Weekday Pattern. (b) Weekend Pattern. (c) Weekly Pattern.

(d) All Generated Traffic.

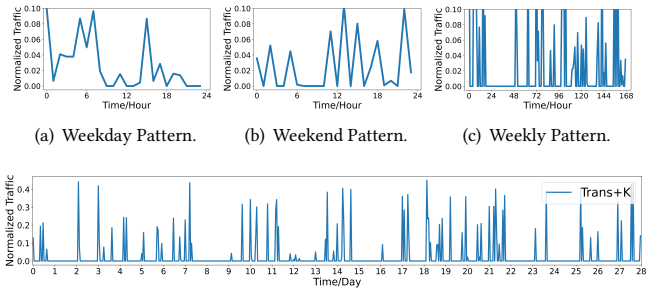
Figure 10: The traffic generated by our model.



(a) Weekday Pattern. (b) Weekend Pattern. (c) Weekly Pattern.

(d) All Generated Traffic.

Figure 11: The traffic generated by our model without KGE.



(a) Weekday Pattern. (b) Weekend Pattern. (c) Weekly Pattern.

(d) All Generated Traffic.

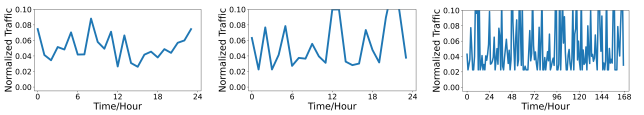
Figure 12: The traffic generated by TransGAN.

A.3 Temporal Patterns Generated by Other Models

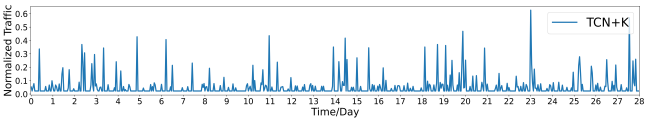
Figures 10, 11, 12, 13, 14 show the temporal patterns generated by our model and all the knowledge-enhanced baseline models. As shown in Figure 12(a) 12(b) 14(a) 14(b), TCN-based model and TransGAN model can't reflect intra-day periodic characteristics, because the two model have peaks late at night. And data generating from RNN-based model illustrated in Figure 13(d) has very little fluctuation, which can't reflect the real cellular traffic variation., Our model shows better on various cycle characteristics than baseline models.

Table 5: Details of Urban Knowledge Graph.

Entity	Amount	Relation	Triplets	Relation	Triplets
Base Station (BS)	12,576	BorderBy(Region,Region)	13,550	CoCheckin(POI,POI)	13,621
POI	1,717,851	NearBy(Region,Region)	39,312	Competitive(POI,POI)	6,848
Region	2,597	LargeOD(Region,Region)	650	RelatedBrand(Brand,Brand)	466
Business Area (BA)	277	SimilarPOIs(Region,Region)	4,170	Brand2Cate1(Brand,Cate1)	1,164
Cate1	14	BaServe(BA,Region)	13,256	Brand2Cate2(Brand,Cate2)	1,136
Cate2	56	LocateAt(POI,Region)	1,717,851	Brand2Cate3(Brand,Cate3)	945
Cate3	480	BelongTo(POI,BA)	1,536,715	SubCateOf_2to1(Cate2,Cate1)	56
Brand	945	Cate1Of(POI,Cate1)	1,717,851	SubCateOf_3to1(Cate3,Cate1)	480
		Cate2Of(POI,Cate2)	1,717,851	SubCateOf_3to2(Cate3,Cate2)	480
		Cate3Of(POI,Cate3)	1,717,851	BrandOf(POI,Brand)	62,607
		ServedBy(POI,Base)	8,511,497	BaseLocatedAt(Base,Region)	12,576
		BaseBelongTo(Base,BA)	10,576	BaseBorderBy(Base,Base)	62,770

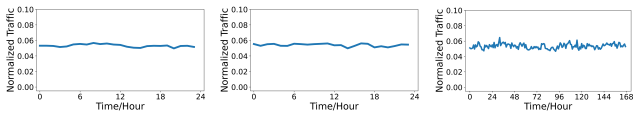


(a) Weekday Pattern. (b) Weekend Pattern. (c) Weekly Pattern.

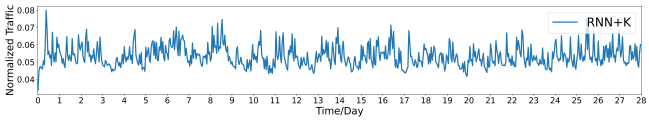


(d) All Generated Traffic.

Figure 14: The traffic generated by TCN-based GAN.



(a) Weekday Pattern. (b) Weekend Pattern. (c) Weekly Pattern.



(d) All Generated Traffic.

Figure 13: The traffic generated by RNN-based GAN.

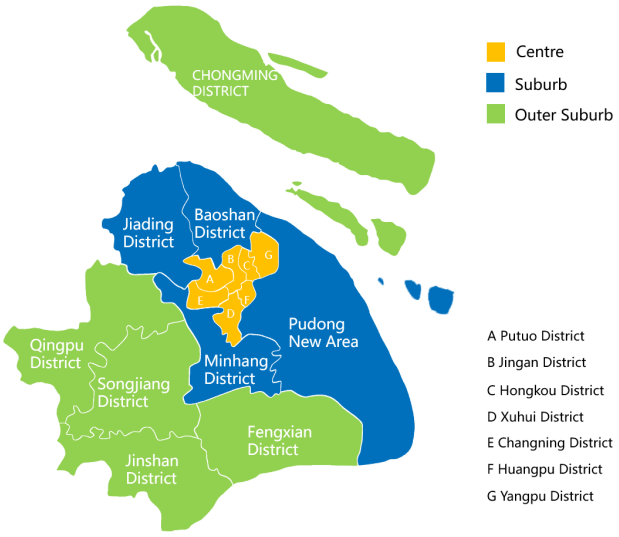


Figure 15: Regionalism in Shanghai.

A.4 Base Station Area Division

In our research, we need a reasonable way to reflect the different cellular traffic temporal patterns between various areas. Generally speaking, different administrative districts may have different industrial, recreational and residential attributes. So we divide base stations into three parts based on administrative districts and geographical location. Figure 15 visually illustrates how we divide regions.