Deep Transfer Learning for City-scale Cellular Traffic Generation through Urban Knowledge Graph

Shiyuan Zhang, Tong Li[§], Shuodi Hui
Department of Electronic Engineering,
BNRist, Tsinghua University
China
tongli@mail.tsinghua.edu.cn

Guangyu Li, Yanping Liang, Li Yu
China Mobile Research Institute
China
liguangyu@chinamobile.com
liangyanping@chinamobile.com
yuliyf@chinamobile.com

Depeng Jin, Yong Li
Department of Electronic Engineering,
BNRist, Tsinghua University
China
jindp@tsinghua.edu.cn
liyong07@tsinghua.edu.cn

ABSTRACT

The problem of cellular traffic generation in cities without historical traffic data is critical and urgently needs to be solved to assist 5G base station deployments in mobile networks. In this paper, we propose ADAPTIVE, a deep transfer learning framework for city-scale cellular traffic generation through the urban knowledge graph. ADAPTIVE leverages historical data from other cities that have deployed 5G networks to assist cities that are newly deploying 5G networks through deep transfer learning. Specifically, ADAPTIVE can align the representations of base stations in the target city and source city while considering the environmental factors of cities, spatial and environmental contextual relations between base stations, and traffic temporal patterns at base stations. We next design a feature-enhanced generative adversarial network, which is trained based on the historical traffic data and representations of base stations in the source city. By feeding the aligned target city's base station representations into the trained model, we can then obtain the generated traffic data for the target city. Extensive experiments on real-world cellular traffic datasets show that ADAPTIVE generally outperforms state-of-the-art baselines by more than 40% in terms of Jensen–Shannon divergence and root-mean-square error. Also, ADAPTIVE has strong robustness based on the results of various cross-city experiments. ADAPTIVE has been successfully deployed on the 'Jiutian' Artificial Intelligence Platform of China Mobile to support cellular traffic generation and assist in the construction and operation of mobile networks.

CCS CONCEPTS

• **Networks** → **Network simulations**; • **Computing methodologies** → **Modeling and simulation**; • **Information systems** → **Spatial-temporal systems**.

KEYWORDS

Cellular traffic, urban knowledge graph, transfer learning, GAN

KDD '23, August 6–10, 2023, Long Beach, CA, USA

ACM ISBN 979-8-4007-0103-0/23/08…$15.00
https://doi.org/10.1145/3580305.3599801

ACM Reference Format:
Shiyuan Zhang, Tong Li[§], Shuodi Hui, Guangyu Li, Yanping Liang,, Li Yu, and Depeng Jin, Yong Li. 2023. Deep Transfer Learning for City-scale Cellular Traffic Generation through Urban Knowledge Graph. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '23), August 6–10, 2023, Long Beach, CA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3580305.3599801

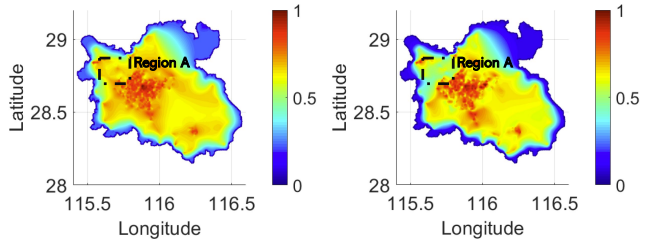
1 INTRODUCTION

Mobile network infrastructures have evolved significantly over the past few decades, moving from First Generation (1G) voice-only networks to the present Fifth Generation (5G) ubiquitous connectivity networks [1, 14]. 5G networks are recognized as providing solutions for all applications, including networked vehicles [22], the internet of things [8], augmented reality [34], virtual reality [30], super-high quality online videos [27], and many more customized services for subscribers. According to a report from the Global Mobile Suppliers Association (GSA) [13], more than 70 countries had been deploying 5G base stations by June 2022. At the same time, a fundamental issue needs to be addressed urgently in deploying 5G base stations: where and how to deploy 5G base stations.

The deployment of base stations traditionally relies on the experiences of experts. Communication engineers manually plan out the sites of base stations. Such a manual approach is limited by expensive labor costs and cannot find optimal solutions for large-scale areas, such as city-scale. Also, merely relying on human experiences can easily result in a high mismatch between human traffic demand and deployed base stations. Figure 1 shows the distribution of normalized base station density and traffic volume of a provincial city in China. We can observe a mismatch in region A, where the traffic demand of users is relatively low, while the number of base stations there remains high. This mismatch will lead to a waste of capital and operating expenses in mobile networks. Hence, deploying base stations according to the estimated traffic load is a more practical approach, which has attracted wide attention from the industry.

Generating or estimating cellular traffic load for newly deployed 5G base stations is challenging due to the lack of historical data. Notably, since 5G networks support a wider range of applications compared to 4G networks, the traffic characteristics of 4G networks differ from 5G networks [17]. As a result, historical data from 4G networks is difficult to be applied to assist in the deployment of 5G networks. One possible and practical solution to address this problem is to leverage the historical data from other cities that have

[§]Corresponding Author.



(a) Normalized base station density. (b) Normalized traffic volume.

Figure 1: Base station density and traffic volume. Relying on human experiences alone results in a high mismatch between traffic demand and deployed base stations in region A.

deployed 5G networks to assist cities that are newly deploying 5G networks through deep transfer learning.

This paper aims to generate cellular traffic for cities without historical data, i.e., the target cities, by leveraging the historical data from other cities, i.e., source cities. Specifically, there are three challenges to achieve this goal.

• **How to build a bridge between the source city and the target city for cellular traffic generation.** We aim to leverage the source city's historical data for the target city's cellular traffic generation. As a result, one big challenge is finding a way to bridge the source and target cities. The urban environment significantly affects base stations' traffic profiles. The urban environment affects how mobile users behave online, such as using mobile apps, and offline, like mobility behavior, which results in various cellular traffic dynamics across base stations. Therefore, the first challenge is how to effectively extract urban environmental factors and transfer environmental knowledge between source and target cities.

• **How to extract and represent relations between base stations.** Apart from environmental factors, the traffic profile of one base station is still affected by nearby base stations. Following Tobler's first law of geography, base stations at a close distance show similar traffic patterns. Meanwhile, base stations with similar environmental contexts, such as surrounding point of interests, also exhibit similar traffic patterns. Hence, we need to model both spatial and environmental contextual relations between base stations and transfer such relations across the source and target cities.

• **How to transfer temporal patterns of base stations' traffic.** The cellular traffic of one base station exhibits specific temporal patterns. For example, base stations located in residential areas have traffic that increases during the day, decreases during the night, and peaks at around 8 p.m. because people generally go out to work in the morning and return home at this time. For generating traffic profiles of base stations, we also need to consider such temporal patterns. Therefore, the last challenge is to transfer temporal traffic patterns of base stations across the source and target cities.

To address the above challenges, we propose **ADAPTIVE**, an urb**A**n knowle**D**ge gr**AP**h **T**ransfer generat**I**ve ad**V**ersarial n**E**twork, which is a deep transfer learning framework for city-scale cellular traffic generation through the urban knowledge graph. Specifically, we first construct an urban knowledge graph to model urban environmental factors for both source and target cities. The urban knowledge graph takes urban content, such as point of interests (POIs), regions, business areas, and base stations, as entities and depicts how these entities relate to one another. Through a knowledge graph embedding model, we can map the entities of both source and target cities into the same latent space while the embeddings of entities retain the environmental knowledge of cities. With the key design of the urban knowledge graph, we can bridge the connection between the source and target cities and transfer environmental knowledge to solve the first challenge. We next leverage graph structure to represent the spatial relations between base stations, where nodes represent base stations and two base stations are connected through an edge if they are spatially close. Moreover, we design a self-supervised task to extract the environmental contextural relations between base stations. By applying a graph neural network (GNN) on the base station graph and taking urban knowledge graph embedding as initial features, we use the representations of base stations to predict their surrounding POIs. As a result, the base stations with similar environmental contexts will have similar representations. By doing so, the second challenge is solved. To solve the third challenge, we conduct a clustering analysis on the source city's base stations according to their temporal traffic profiles and then identify the central representations in each cluster for typical temporal patterns. We then design an attention-driven matching score to regulate the base stations' representations of target cities according to the central representations for transferring temporal patterns of base stations' traffic. As a result, we align the embeddings of base stations in the target city and source city while considering the environmental factors of cities, spatial and environmental contextual relations between base stations, and traffic temporal patterns of base stations. After obtaining the final representations of base stations in the target city, we feed them into a conditional GAN trained with historical data from the source city to generate cellular traffic for the base stations in the target city.

The contributions of our work can be summarized as follows.

• We investigate the problem of cellular traffic generation for cities without historical data to assist 5G base station deployments. To solve this problem, we propose using historical data from other cities that have deployed 5G networks to assist cities that are newly deploying 5G networks through deep transfer learning.

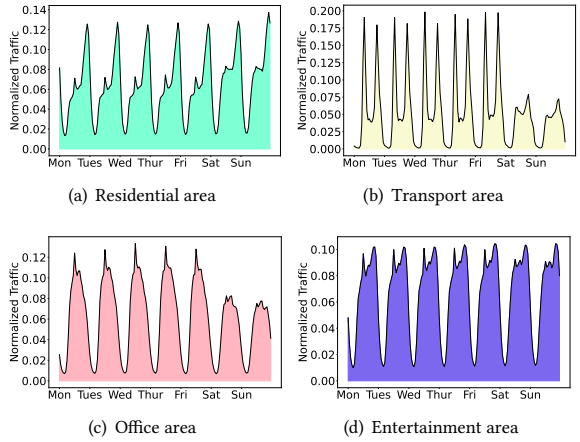
• We propose a new deep transfer learning framework, ADAPTIVE, for cellular traffic generation. ADAPTIVE can align the representations of base stations in the target city and source city while considering the environmental factors of cities, spatial and environmental contextual relations between base stations, and traffic temporal patterns of base stations.

• Extensive experiments on real-world cellular traffic datasets show that ADAPTIVE generally outperforms state-of-the-art baselines by more than 40% in terms of Jensen–Shannon divergence and root-mean-square error. Also, ADAPTIVE has strong robustness based on the results of various cross-city experiments. ADAPTIVE has been successfully deployed on the 'Jiutian' Artificial Intelligence Platform of China Mobile to support cellular traffic generation and assist in the construction and operation of mobile networks.

2 PRELIMINARIES

2.1 Traffic Patterns Across Base Stations

Many previous studies have demonstrated that base stations' traffic shows several patterns related to environmental context [28, 29, 38].



(a) Residential area (b) Transport area

(c) Office area (d) Entertainment area

Figure 2: Cellular traffic patterns under different environmental contexts.

Each traffic pattern can be mapped to a class of geographic locations relevant to regional functions, such as residential, office, transport, and entertainment areas. Base stations in each type of area have their specific traffic temporal patterns. In our paper, we conduct clustering analysis on traffic profiles of Shanghai's base stations and observe four typical traffic patterns. Figure 2 shows the four traffic temporal patterns derived from the Shanghai dataset. The four temporal patterns differ in terms of when peak traffic volume occurs and traffic volume on weekdays and weekends. For instance, as for the base stations located in the residential area, the traffic volume is higher at night than during the day because most people go out to work in the morning and back home in the evening. As for the base stations located in office areas, their traffic peaks during the daytime. Also, the traffic volume on weekends is relatively lower than that on weekdays because weekends are holidays and people do not go to office areas. In addition, as for the base stations near subway and bus stations, their traffic loads peak in the morning and evening rush hour. Therefore, the environmental contexts of base stations are essential factors that impact their traffic profiles. Base stations in similar environmental contexts have similar traffic patterns and vice versa. These findings motivate us to use environmental contexts as a bridge between the source and target cities and to generate traffic profiles of the base stations in the target city.

2.2 Urban Knowledge Graph

To characterize the environmental contexts of base stations, we introduce an urban knowledge graph [33, 45]. A knowledge graph, also known as a semantic network, depicts a network of real-world entities and shows how they relate to one another. Similarly, the urban knowledge graph takes urban content, such as base stations, point of interests (POIs), and regions, as entities where spatial and semantic dependencies are modeled as relations. Specifically, there are six categories of entities in the urban knowledge graph: base stations, POIs, regions, business areas, categories of POIs, and brands of POIs. One base station is connected to other entities with four relations. One base station is **located at** a region. One base station **belongs to** a business area. A POI is **served by** a base station. A base station **borders** another base station. We also model semantic relations among other types of entities. For example,

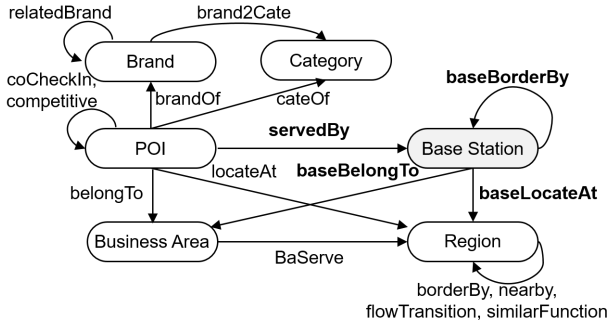


Figure 3: The schema of the urban knowledge graph.

POIs belong to business areas, and business areas serve regions. Figure 3 illustrates the schema of the urban knowledge graph that encompasses the impact of urban environmental contexts on the cellular traffic of base stations.

3 PROBLEM DEFINITION AND FRAMEWORK OVERVIEW

3.1 Problem Definition

We give a formal mathematical definition of the cellular traffic generation problem. Let $\mathbf{X}^{SRC} = \{X_i^{SRC}\}_{i=1}^{N^{SRC}}$ denote the historical traffic data of the source city, where X_i^{SRC} represents the traffic time-series of the i-th base station BS_i^{SRC} in the source city and N^{SRC} represents the number of base stations in the source city. Also, we denote the urban knowledge graph of the source city as $\mathcal{G}^{SRC} = (\mathcal{E}^{SRC}, \mathcal{R}^{SRC}, \mathcal{F}^{SRC})$, where $\mathcal{E}^{SRC}, \mathcal{R}^{SRC}$ and $\mathcal{F}^{SRC}$ represent the sets of entities, relations, and facts, respectively. The fact set includes triplets on factual knowledge, i.e., $\mathcal{F} = \{(e_s, r, e_o)$, with $e_s, e_o \in \mathcal{E}^{SRC}$ denoting subject and object entities respectively and $r \in \mathcal{R}^{SRC}\}$ denoting the relation between them. Similarly, we denote the urban knowledge graph of the target city as $\mathcal{G}^{TGT} = (\mathcal{E}^{TGT}, \mathcal{R}^{TGT}, \mathcal{F}^{TGT})$. Given the historical traffic dataset of the source city $\mathbf{X}^{SRC}$ and the urban knowledge graphs of the source city $\mathcal{G}^{SRC}$ and the target city $\mathcal{G}^{TGT}$, our goal is to generate city-scale traffic $\hat{\mathbf{X}}^{TGT}$ for the base stations located in the target city.

3.2 Framework Overview

As a solution, we propose ADAPTIVE, a deep transfer learning framework for city-scale cellular traffic generation through the urban knowledge graph. We present an overview of our proposed framework in Figure 4. There are four steps in our framework: (1) knowledge graph embedding, (2) learning base station representations, (3) aligning base station representations, (4) cellular traffic generation. The design and details of each step of our proposed framework are given below in the following sections.

4 METHOD

4.1 Knowledge Graph Embedding

We construct urban knowledge graphs for both source and target cities to model and extract the urban environmental factors, where urban contents like base stations, regions, and POIs are modeled as entities, and their spatial and semantic correlations are modeled as

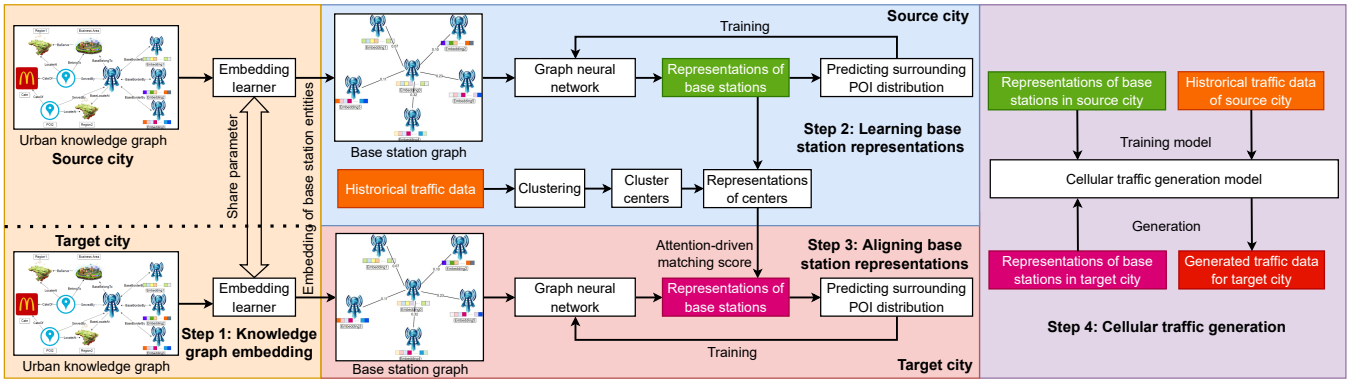


Figure 4: Framework overview. There are four steps in our framework: (1) knowledge graph embedding, (2) learning base station representations, (3) aligning base station representations, (4) cellular traffic generation.

relations. Specifically, as shown in Figure 3, a base station is linked to other entities via four types of relations in the urban knowledge graph. 1) a base station is located at region; 2) a base station belongs to a business area; 3) a base station serves a POI; 4) a base station borders another base station.

To fully exploit environmental features modeled in urban knowledge graphs, we first leverage knowledge graph embedding to learn low-dimensional vectors (embeddings) of entities while preserving their semantic meaning [23]. We adopt a commonly used embedding model, TuckER [4]. TuckER is a state-of-art tensor factorization method for knowledge graph embedding, outperforming many translation-based models (e.g., TransE), bilinear models (e.g., ComplEx), and neural network models (e.g., ConvE) in practice. TuckER model measures the plausibility scores based on the triplets $\mathcal{F} = \{(e_s, r, e_o)\}$ in the urban knowledge graph and uses the cross-entropy loss function for learning embeddings of graph entities. By feeding urban knowledge graphs of the source city $\mathcal{G}^{SRC}$ and target city $\mathcal{G}^{TGT}$ into the TuckER model, we then obtain the embeddings of base station entities E_{BS}^{SRC} and E_{BS}^{TGT} for the source city and target city, respectively.

4.2 Learning Base Station Representations

To characterize the spatial relationships between base stations, we construct base station graphs G_{BS}^{SRC} and G_{BS}^{TGT} for both the source city and the target city, respectively. The base station graph is a weighted attributed graph, denoted by $G_{BS} = (V_{BS}, A_{BS}, H_{BS})$. Vertices V_{BS} represent base stations, adjacency matrix A_{BS} depicts the geographical distance between base stations, and H_{BS} is the set of initial feature vectors of base stations. In our case, we take the embeddings of base station entities learned from the urban knowledge graph as the initial features of vertices in the base station graph. Specifically, a base station is connected with its top M closest base stations. By denoting the geographical distance between base station i and j as $Dist(i, j)$, the weight of the edge connecting base stations i and j, denoted by $A_{BS}(i, j)$, can be computed as,

$$A_{BS}(i,j) = \frac{1/Dist(i,j)}{\max\limits_{i,j}(1/Dist(i,j))}, \tag{1}$$

where the edge weights represents the normalized spatial similarity between base stations and belongs to the interval $[0, 1]$.

We next apply a graph neural network (GNN) on the base station graph to embed graph structure information, i.e., spatial relations of base stations, into the node features. The key idea of graph neural networks is to aggregate features from neighbors of nodes. The message-passing phase and the aggregating and updating phase are the two stages that typically make up a GNN computation [44]. Specifically, a node passes its feature vector to its first-order neighbors via the graph structure during the message-passing phase. In the aggregating and updating phase, a node first aggregates the representation vectors received from neighbors with its representation. The node then updates its representation with the aggregated one. By increasing the network layers, each node can incorporate information from higher-order neighbors and thus integrate graph structure information into node representations. In our case, we use the graph convolutional network (GCN) [26] to learn base station representations. Thus, the representations of base stations of the l-th layer of graph convolutional network can be expressed as,

$$H_{BS}^{(l)} = \sigma\left(D_{BS}^{-\frac{1}{2}} A_{BS} D_{BS}^{-\frac{1}{2}} W^{(l)} H_{BS}^{(l-1)}\right), \tag{2}$$

where $H_{BS}^{(l)}$ denotes the base station representations of the l-th layer of graph convolutional network, D_{BS} denotes the degree matrices for the base station graph, $\sigma(\cdot)$ denotes the sigmoid activation function, and $W^{(l)}$ is the learnable weight matrix for the l-th layer. The initial features of base stations are the embeddings of base station entities learned from the urban knowledge graph, i.e., $H_{BS}^{(0)} = E_{BS}$.

We next design a POI distribution reconstruction task to incorporate environmental contextual relations between base stations into their representations, where GCN is trained to produce base station representations to reconstruct the surrounding POI distributions of corresponding base stations. In practice, we use a two-layer GCN model and concatenate the output of each layer to form the final representations of base stations,

$$Z_{BS} = [H_{BS}^{(1)}, H_{BS}^{(2)}], \tag{3}$$

where Z_{BS} denotes the final representations of base stations and $[\cdot]$ denotes the concatenation operation. We then apply a MLP to

reconstruct the surrounding POI base station distributions,

$$\hat{PD} = \text{MLP}(Z_{BS}), \tag{4}$$

where $\hat{PD}$ denotes the POI distributions predicted by the MLP model. We then use the Kullback-Leibler divergence [25] to measure the distance between the predicted distribution $\hat{PD}$ and the ground truth PD and take this as POI similarity loss $\mathcal{L}_{POI}$,

$$\mathcal{L}_{POI} = KL(\hat{PD}, PD), \tag{5}$$

where $KL(\cdot)$ denotes the Kullback-Leibler divergence. The ground truth of POI distributions of base stations is obtained by counting the number of POIs of each category within the coverage area of each base station. As for the source city, its base station representations Z_{BS}^{SRC} are obtained by training GCN and MLP models with the POI similarity loss, as depicted by the blue part of Figure 4.

4.3 Aligning Base Station Representations

We next conduct a clustering operation on the historical traffic time series of base stations in the source city for explicitly capturing traffic temporal patterns. After clustering, we obtain each base station's cluster labels, denoted by C_{BS}. We then compute the base station representation of each cluster center as,

$$\hat{z}_{BS,c} = \frac{1}{N_{BS,c}} \sum_{c_{BS_i}==c} z_{BS_i}^{SRC}, \tag{6}$$

where $\hat{z}_{BS,c}$ denotes the central base station representation of the cluster c, $N_{BS,c}$ denotes the number of base stations in the cluster c. As a result, $\hat{Z}_{BS} = \{\hat{z}_{BS,1}, \hat{z}_{BS,2}, ...\}$ depicts typical base station representations for traffic temporal patterns.

To transfer temporal patterns of base stations' traffic, we design an attention-driven matching score to align the base station representations of the target city with the typical base station representations of traffic temporal patterns. As for the i-th base station in the target city BS_i^{TGT}, we define its matching score as $M(BS_i^{TGT})$,

$$M(BS_i^{TGT}) = \frac{\exp(sim(z_{BS_i}^{TGT}, \hat{z}_{BS,k}))}{\sum_j^K \exp(sim(z_{BS_i}^{TGT}, \hat{z}_{BS,j})}, \tag{7}$$

where $z_{BS_i}^{TGT}$ denotes the representation vector of the i-th base station in the target city, K denotes the number of clusters, and $sim(\cdot)$ denotes the cosine similarity. $k = \arg\max_j sim(z_{BS_i}^{TGT}, \hat{z}_{BS,j})$ denotes the cluster having the highest cosine similarity to the representations with the base station BS_i^{TGT}. Following the design of focal loss, we define the pattern matching loss $\mathcal{L}_{Pattern}$,

$$\mathcal{L}_{Pattern} = -\sum_i (1 - M(BS_i^{TGT})) \log(M(BS_i^{TGT})). \tag{8}$$

As for the target city, its base station representations Z_{BS}^{TAG} are then obtained by training GCN and MLP models with both the POI similarity loss $\mathcal{L}_{POI}$ and the pattern matching loss $\mathcal{L}_{Pattern}$.

4.4 Cellular Traffic Generation

We propose a feature-enhanced generative adversarial network (GAN) for cellular traffic generation [21]. As shown in Figure 5, there are two parts of the inputs for the generation model: random noise and representations of base stations. Similar to traditional GAN [15], We input random noise to introduce the randomness of base stations for traffic generation. In addition to the noise, representations of base stations also act as a part of inputs, providing conditions for traffic generation. We leverage a multi-generator structure to capture the daily pattern and weekly pattern of the traffic time series. We train the feature-enhanced generative adversarial network with Wasserstein loss [2] based on the historical traffic data and representations of base stations in the source city. By feeding the noise and the target city's representations into the trained model, we can obtain the generated traffic data for the target city.

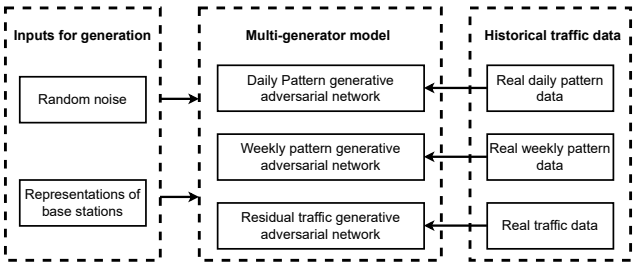


Figure 5: Traffic generation model.

5 EXPERIMENTS

5.1 Experimental Settings

In this section, we conduct extensive experiments on multiple traffic datasets collected from real-world mobile networks and compare the performance of our proposed method, ADAPTIVE, with state-of-the-art baselines. We also run ablation studies to verify the effectiveness of individual modules and test the model's robustness across different cities. The experiments are performed on the Jiutian Artificial Intelligence Platform.

5.1.1 Dataset. We collected three large-scale cellular traffic datasets from three Chinese cities: Shanghai, Nanjing, and Beijing.

- Shanghai The dataset covered the network traffic data of 5,326 base stations in Shanghai and was collected over one month in 2021. The traffic information of each base station is collected every hour.
- Beijing. The dataset covered the network traffic data of 4,351 base stations in Beijing and was collected over one week of October 2021. The records of each base station are collected every hour.
- Nanjing. The dataset covered the network traffic data of 6,890 base stations in Nanjing and was collected over one week of October 2021. The records of each base station are collected every hour.

5.1.2 Baselines. We compare our proposed model with the following three baselines. Notably, we evaluate each model with four different kinds of inputs: 1) noise vectors only, 2) noise vectors and urban knowledge graph embedding E_{BS}, 3) noise vectors and POI distribution vector PD, 4) noise vectors and base station representations learned by our model Z_{BS}.

- **TransGAN.** TransGAN [24] is a transformer-based GAN that combines a multi-scale discriminator to concurrently capture low-level textures and semantic contexts with a generator using transformer blocks that gradually enhance feature resolution. We change the transformer block sizes for cellular traffic generation.

Table 1: Traffic generation performance when Shanghai is the source city and Nanjing and Beijing are the target cities. Bold denotes the best results and underline denotes the second-best results. 'E_{BS}' stands for using urban knowledge embedding E_{BS} for transfer learning. 'PD' stands for using POI distribution vectors of base stations for transfer learning, and 'Z_{BS}' stands for using the base station representations learned by our model for transfer learning.

Cities	Shanghai (Source) → Nanjing (Target)						Shanghai (Source) → Beijing (Target)					
Methods	Traffic Volume		First-order Difference		Daily Periodic Component		Traffic Volume		First-order Difference		Daily Periodic Component	
	JSD	Δ	JSD	Δ	RMSE	Δ	JSD	Δ	JSD	Δ	RMSE	Δ
Trans	0.5378	47.79%	0.1656	160.38%	0.0587	12.67%	0.5869	104.49%	0.1857	166.05%	0.1021	63.1%
Trans+ PD	0.5101	40.18%	0.1666	161.95%	0.0594	14.01%	0.5743	100.1%	0.1851	165.19%	0.1031	64.7%
Trans+ E_{BS}	0.5307	45.84%	0.1564	145.91%	0.0589	13.05%	0.5883	104.98%	0.1749	150.57%	0.1024	63.58%
Trans+ Z_{BS}	0.5140	41.25%	0.1540	142.14%	0.0566	8.64%	0.5759	100.66%	0.1575	125.64%	0.1023	63.42%
RNN	0.7294	100.44%	<u>0.0863</u>	35.69%	0.0567	8.83%	0.7103	147.49%	0.1846	164.47%	0.1054	68.37%
RNN+ PD	0.5914	62.52%	0.1328	108.81%	0.0638	22.46%	0.6613	130.42%	0.0944	42.41%	0.1079	72.36%
RNN+ E_{BS}	0.6226	71.09%	0.0931	46.38%	0.0523	0.38%	0.7026	144.81%	0.0766	9.74%	0.1010	61.34%
RNN+ Z_{BS}	0.5913	62.49%	0.1328	108.81%	0.0638	22.46%	0.6613	130.42%	0.0944	35.24%	0.1078	72.2%
TCN	0.7626	109.56%	0.1426	124.21%	0.1289	147.41%	0.5774	101.18%	0.1853	165.47%	0.1076	71.88%
TCN+ PD	0.5945	63.37%	0.1328	108.81%	0.1036	98.85%	0.4259	48.4%	0.0995	42.55%	0.1016	62.3%
TCN+ E_{BS}	0.7814	68.73%	0.1085	70.6%	0.0927	77.93%	0.8513	196.62%	0.0858	22.92%	0.0965	54.15%
TCN+ Z_{BS}	0.5674	55.92%	0.0963	51.42%	0.0847	62.57%	0.4133	44.01%	0.0844	20.92%	0.0841	34.35%
ADAPTIVE	0.7173	97.11%	0.1047	64.62%	<u>0.0516</u>	-0.96%	0.5703	98.71%	0.1454	108.31%	0.0890	42.17%
ADAPTIVE+ PD	0.5853	60.84%	0.0998	56.92%	0.0540	3.65%	0.5045	75.78%	**0.0679**	-2.72%	<u>0.0698</u>	11.5%
ADAPTIVE+ E_{BS}	<u>0.4985</u>	36.99%	0.0972	52.83%	**0.0470**	-9.79%	<u>0.3444</u>	20.0%	0.1782	155.3%	0.0712	13.74%
ADAPTIVE+ Z_{BS}	**0.3639**	0	**0.0636**	0	0.0521	0	**0.2870**	0	<u>0.0698</u>	0	**0.0626**	0

• **LSTM-based GAN**. Long short-term memory (LSTM) [19] is of the recurrent neural network structure renowned for storing historical values across variable periods. Two LSTMs are used as the generator and discriminator in constructing the GAN.

• **TCN-based GAN**. We construct a GAN using temporal convolutional networks (TCNs) [3] as the generator and discriminator for cellular traffic generation.

5.1.3 Metrics. We evaluate our model via the following 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) [12] is a commonly used metric to describe the similarity between two distributions, which is defined as,

$$JSD(\hat{\mathbf{X}}^{TGT}, \mathbf{X}^{TGT}) = \sqrt{\frac{KL(\mathbf{X}^{TGT} \parallel \hat{\mathbf{X}}^{TGT}) + KL(\hat{\mathbf{X}}^{TGT} \parallel (\mathbf{X}^{TGT})}{2}}, \quad (9)$$

where $\mathbf{X}^{TGT}$ represents the real data, $\hat{\mathbf{X}}^{TGT}$ represents the generated data, and $\mathrm{KL}(\cdot)$ is the Kullback-Leibler divergence. A lower JSD implies a better generation model since the distribution of the generated data is more similar to the real data.

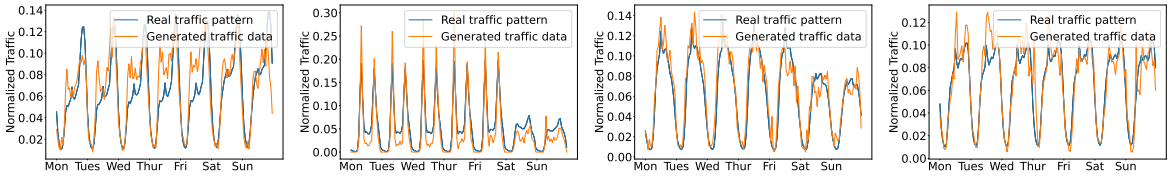
First-order Difference. To evaluate the variation between every two adjacent generated traffic point, we compute the first-order difference of time series, denoted by $\hat{\mathbf{X}}_d^{TGT} = \hat{x}_{t+1}^{TGT} - \hat{x}_t^{TGT}$. We then compute the JSD between the first-order differences of generated datasets and their corresponding real dataset.

Daily Frequency Component. We evaluate the daily periodicity of the generated traffic via calculating daily frequency component. Firstly, we compute the frequency of each generated cellular traffic time series $\hat{\mathbf{X}}^{TGT}$, which is denoted as $\widehat{F} = FFT(\hat{\mathbf{X}}^{TGT})$. $FFT(\cdot)$ denotes the fast Fourier transform operation [18], which extracts frequency components of time series. We then compute the root-mean-square error (*RMSE*) between the generated one and the real cellular traffic series on daily frequency component $F = FFT(\mathbf{X}^{TGT})$, which can be denoted as $RMSE(\widehat{F}, F) = \sqrt{(\widehat{F} - F)^2}$.

5.2 Overall Performance Evaluation

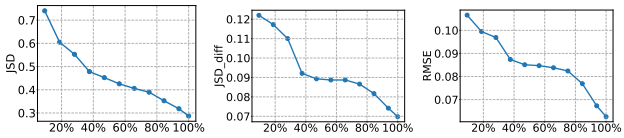
Table 1 shows the experimental results of the proposed model and the baseline model on the task of cellular traffic generation, where Shanghai is the source city, and Nanjing and Beijing are the target cities. 'Trans' represents TransGAN with noise vectors only inputs. 'RNN' represents RNN-based GAN with noise vectors only inputs. 'TCN' represents TCN-based GAN with noise vectors only inputs. 'ADAPTIVE' represents our proposed model. 'E_{BS}' stands for using urban knowledge embedding E_{BS} for transfer learning. 'PD' stands for using POI distribution vectors of base stations for transfer learning, and 'Z_{BS}' stands for using the base station representations learned by our model for transfer learning.

In most cases, the base station representations, learned by our model as a transfer bridge to link the source city and target city, perform the best to improve each model's performance on the traffic generation for the target city. Also, the generation models with knowledge graph embeddings have a relatively good performance, demonstrating the effectiveness of leveraging the urban knowledge graph to extract urban environmental factors and transfer environmental knowledge between source and target cities for cellular traffic generation. Specifically, according to the JSD of traffic volume and the JSD of the first-order differences, our model ADAPTIVE provides the best result. This indicates that our model outperforms the other models in capturing both cellular traffic volume distribution and traffic time series changing trends. RNN-based GAN also



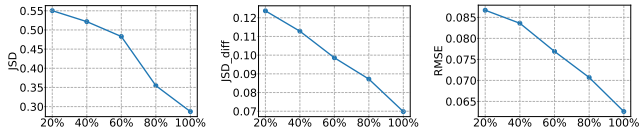
(a) Residential area (b) Transport area (c) Office area (d) Entertainment area

Figure 6: Comparisons of traffic generation results on different functional regions.



(a) Traffic Volume (b) First-order Difference (c) Daily Periodic

Figure 7: Sensitivity to the training data size by changing the proportion of base stations in the training dataset.



(a) Traffic Volume (b) First-order Difference (c) Daily Periodic

Figure 8: Sensitivity to the scale of knowledge graphs by changing the proportion of POIs in the knowledge graph.

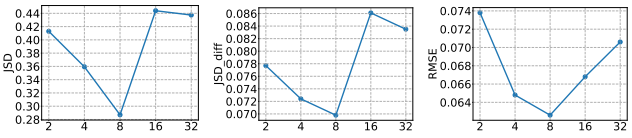
has an acceptable performance in terms of the first-order difference metric since it is good at capturing short-term changes. As for daily periodicity, our model still outperforms the other models, proving that we can successfully learn and transfer the daily traffic temporal patterns across the source and target cities.

We also conduct a case study, which selects four functional regions, residential area, office area, entertainment area, and transport area, in the target city, and compares the generated traffic and real one across different functional regions. As shown in Figure 6, we can observe apparent daily and weekly traffic temporal patterns of generated traffic data. The temporal patterns of the traffic time series generated by our model are consistent with real traffic. This verifies that the traffic temporal patterns are successfully transferred from the source city to the target city across different functional regions, demonstrating the effectiveness of the key designs of the knowledge graph module and attention-driven matching score.

5.3 Ablation Studies

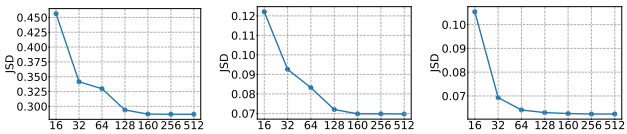
We next conduct ablation studies to test the sensitivity of our method for the training data size, the scale of knowledge graphs, the number of traffic pattern clusters, and the dimension of base station representations.

Training data size: We test how the generation performance of ADAPTIVE changes with the training data size by varying the number of base stations in the training dataset, where Shanghai is the source city and Beijing is the target city. As shown in Figure 7, the model performance improves with the increase in the number of base stations in the training set. Notably, the model performance



(a) Traffic Volume (b) First-order Difference (c) Daily Periodic

Figure 9: Sensitivity to the number of traffic pattern clusters.



(a) Traffic Volume (b) First-order Difference (c) Daily Periodic

Figure 10: Sensitivity to the dimensions of base station representations.

increases dramatically at the initial stage when the proportion of base stations increases from 0 to 40%. After that, there is a plateau stage. However, when the proportion exceeds 80%, the model performance continues to increase. These findings indicate that our model does not have overfitting problems due to unsupervised clustering to classify different temporal patterns. Also, if we cannot collect over 80% of base stations' historical data due to time and expenditure costs, then collecting 40% of base stations' historical data is the most cost-effective option.

Scale of urban knowledge graphs: We next investigate how the generation performance of ADAPTIVE changes with the scale of urban knowledge graphs by randomly removing the POI entities in the target city, where Shanghai is the source city and Beijing is the target city. As shown in Figure 8, our method is highly sensitive to the amount of knowledge introduced into the target city. Only with complete urban environment information, the ADAPTIVE can achieve the best transfer performance. As a result, compared with the scale of historical data, environmental factors play a more critical role in the traffic generation task.

Number of traffic pattern clusters: We set the number of clusters as 2, 4, 8, 16, 32, and 64, respectively. As shown in Figure 9, the performance first increases as the number of cluster categories increases because, in this range, increasing clusters can clearly distinguish the temporal pattern difference between clusters, namely having a low inter-cluster distance while a high intra-cluster distance. However, if we continue to increase the number of clusters, the performance will degenerate. That is because too many clusters will make the difference between clustered patterns smaller and make it difficult for the model to learn the correct pattern.

Table 2: Traffic generation performance when Beijing is the source city and Shanghai and Nanjing are the target cities.

Cities	Beijing (Source) → Shanghai (Target)						Beijing (Source) → Nanjing (Target)					
Methods	Traffic Volume		First-order Difference		Daily Periodic Component		Traffic Volume		First-order Difference		Daily Periodic Component	
	JSD	Δ	JSD	Δ	RMSE	Δ	JSD	Δ	JSD	Δ	RMSE	Δ
Trans	0.4258	131.16%	0.1429	54.65%	0.0731	-0.14%	0.5310	58.89%	0.1656	161.67%	0.0819	94.54%
Trans+ PD	0.4040	119.33%	0.1409	52.49%	0.0740	1.09%	0.5100	52.6%	0.1666	161.83%	0.0594	41.09%
Trans+ E_{BS}	0.4303	133.6%	0.1277	38.2%	0.0730	-0.27%	0.5307	58.8%	0.1564	148.74%	0.0589	39.9%
Trans+ Z_{BS}	0.3979	116.02%	0.0988	6.93%	0.0734	0.27%	0.5134	53.62%	0.1540	154.1%	0.0574	36.34%
RNN	0.6584	257.44%	0.0709	-23.27%	0.0796	8.74%	0.7293	118.22%	0.0862	35.96%	0.0566	34.44%
RNN+ PD	0.5043	173.78%	0.1311	41.88%	0.0786	7.38%	0.6154	84.14%	0.1427	125.08%	0.0737	75.06%
RNN+ E_{BS}	0.5219	183.33%	0.1336	44.59%	0.0848	15.85%	0.7129	113.32%	0.0774	22.08%	0.0587	39.43%
RNN+ Z_{BS}	0.5042	173.72%	0.1311	41.88%	0.0786	7.38%	0.5914	76.96%	0.1322	108.52%	0.062	47.27%
TCN	0.5849	217.54%	0.1344	45.45%	0.0784	7.1%	0.8416	151.83%	0.1019	60.73%	0.0636	51.07%
TCN+ PD	0.4592	149.29%	**0.0669**	-27.6%	0.0784	7.1%	0.6013	79.92%	0.1097	73.03%	0.0636	51.07%
TCN+ E_{BS}	0.4252	130.84%	0.0979	5.95%	0.0780	6.56%	0.8242	146.62%	0.1443	127.6%	0.0537	27.55%
TCN+ Z_{BS}	0.3770	104.67%	0.0734	-20.56%	0.0784	7.1%	0.4739	41.8%	0.1023	61.36%	0.0636	51.07%
ADAPTIVE	0.3247	76.28%	0.1304	41.13%	0.0641	-12.43%	0.5561	66.4%	0.1710	169.72%	0.0517	22.8%
ADAPTIVE+ PD	0.6250	239.31%	0.1119	21.1%	**0.0457**	-37.57%	0.7109	112.72%	0.1444	127.76%	0.0363	-13.78%
ADAPTIVE+ E_{BS}	0.2304	25.08%	0.0729	-21.1%	0.0658	-10.11%	0.4951	48.14%	0.1311	106.78%	**0.0457**	8.55%
ADAPTIVE+Z_{BS}	**0.1842**	0	0.0924	0	0.0732	0	**0.3342**	0	**0.0634**	0	0.0421	0

Table 3: Traffic generation performance when Nanjing is the source city and Beijing and Shanghai are the target cities.

Cities	Nanjing (Source) → Beijing (Target)						Nanjing (Source) → Shanghai (Target)					
Methods	Traffic Volume		First-order Difference		Daily Periodic Component		Traffic Volume		First-order Difference		Daily Periodic Component	
	JSD	Δ	JSD	Δ	RMSE	Δ	JSD	Δ	JSD	Δ	RMSE	Δ
Trans	0.5869	185.46%	0.1857	125.64%	0.1021	89.78%	0.4173	165.46%	0.1326	79.67%	0.0713	-9.06%
Trans+ PD	0.6734	227.53%	0.1894	130.13%	0.1037	92.75%	0.3946	151.02%	0.1266	71.54%	0.0720	-8.16%
Trans+ E_{BS}	0.6814	231.42%	0.1803	119.08%	0.1012	88.1%	0.4319	174.75%	0.1022	38.48%	0.0712	-9.18%
Trans+ Z_{BS}	0.6627	222.32%	0.1653	100.85%	0.0987	83.46%	0.3879	146.76%	0.0924	25.2%	0.0696	-11.22%
RNN	0.7712	275.1%	0.1314	59.66%	0.0999	85.69%	0.5876	273.79%	0.0903	22.36%	0.0710	-9.44%
RNN+ PD	0.7016	241.25%	0.1049	27.46%	0.1088	102.23%	0.4418	181.04%	0.101	36.86%	0.0786	0.26%
RNN+ E_{BS}	0.7185	249.46%	0.0894	8.63%	0.1041	93.49%	0.4697	198.79%	0.1353	83.33%	0.0854	8.93%
RNN+ Z_{BS}	0.6016	192.61%	0.0944	14.7%	0.1078	100.37%	0.4416	180.92%	0.1011	36.99%	0.0717	-8.55%
TCN	0.6037	193.63%	0.1623	97.21%	0.1068	98.51%	0.5926	276.97%	0.1097	48.64%	0.0777	-0.89%
TCN+ PD	0.5012	143.77%	0.0867	5.35%	0.1076	100.0%	0.516	228.24%	0.0965	30.76%	0.0816	4.08%
TCN+ E_{BS}	0.8513	314.06%	0.0858	4.25%	0.0965	79.37%	0.5144	227.23%	0.1079	46.21%	0.0890	13.52%
TCN+ Z_{BS}	0.3855	87.5%	0.1218	48.0%	0.1076	100.0%	0.3682	134.22%	0.0876	18.7%	0.0748	-4.59%
ADAPTIVE	0.3406	65.66%	0.1635	98.66%	0.0711	32.16%	0.3430	118.19%	0.1406	90.51%	0.0633	-19.26%
ADAPTIVE+ PD	0.4311	109.68%	**0.0453**	-44.96%	0.0692	28.62%	0.5881	274.11%	0.0864	17.07%	**0.0454**	-42.09%
ADAPTIVE+ E_{BS}	0.3303	60.65%	0.0584	-29.04%	0.0711	32.16%	0.2218	41.09%	**0.0728**	-1.36%	0.0656	-16.33%
ADAPTIVE+ Z_{BS}	**0.2056**	0	0.0823	0	**0.0538**	0	**0.1572**	0	0.0738	0	0.0784	0

Dimensions of base station representations: We also evaluate the impact of the dimensions of base station representations. As shown in Figure 10, the generation performance increases with the representation dimensions rise. The performance converges when the representation dimensions exceed 128, reflecting that 128-dimensional vectors can represent base stations in a city while containing information on the environmental factors of cities, spatial and environmental contextual relations between base stations, and traffic temporal patterns of base stations.

5.4 Generalization and Robustness

We use Beijing, Shanghai, and Nanjing as the source and destination cities one by one to perform cross-city experiments to verify whether our transfer learning method is robust in different cities. As shown in Tables 1, 2, and 3, ADAPTIVE can always get a good performance according to traffic volume, first-order difference, and daily periodic component. In different groups of experiments, our model improved the performance compared to baselines by at least 41.8%. The extensive cross-city experimental results show that our model

has good generalization and robustness while outperforming the baseline model. Based on the base station representations learned by our method, the generation performance generally improves 20% for ADAPTIVE; it also improves the generation performance for other baseline models. As a result, our method can be effective in traffic generation for transferring cross-city knowledge and has generalization and robustness. Our method also paves the way to use urban knowledge to assist traffic generation.

6 RELATED WORK

6.1 Network Traffic Generation

Network traffic is initially generated by mathematic models [37, 40], e.g., Poisson model, to test network equipment, network services, and security products [43]. With the development of machine learning and artificial intelligence, machine learning tools, like autoregressive models [5, 41], are applied to network traffic generation. The generative adversarial network, as a state-of-the-art generative model, is also becoming popular for generating network traffic. For instance, Ring et al. [35] generate network traffic flows using a three GAN-based pre-processing strategy. Dowoo et al. [11] suggest creating network packet traces using the PcapGAN model trained on network packet-level data. Lin et al. [31] and Lin et al. [42] propose the DoppelGANger model to simultaneously generate packet attributes and feature series. However, the aforementioned GAN models concentrate on producing traffic packets for a single base station, necessitating detailed configurations of network packets, including network protocols, IP addresses, etc. Thus, extending to larger-scale generation tasks is hard, particularly at a city-scale traffic generation. In a nutshell, the aforementioned GAN cannot solve the problem of cellular traffic generation for cities without historical data to assist 5G base station deployment.

6.2 Knowledge Graph Application

By adding rich structured knowledge to assist representation learning, knowledge graphs [6] are often employed in several real-world artificial intelligence applications, including natural language understanding [10], question-answering [20, 23], and recommendation systems [16]. For instance, Sun et al. [36] leveraged a language knowledge graph to integrate knowledge information into the continuous multitask learning language model. Liu et al. [32] used domain knowledge in the BERT contextual encoder. Chen et al. [7] proposed to use the large-scale language knowledge graph to represent the two-way interaction between questions via a bidirectional attention mechanism. Ding et al. [9] suggest multi-hop reasoning to integrate implicit extraction and explicit reasoning and build a cognitive knowledge graph model based on graph neural networks. By exchanging latent characteristics and simulating high-order item-entity interaction, Wang et al. [39] propose a multi-interaction item knowledge graph to link the multitask representation and recommendation for the suggested items. The aforementioned studies demonstrate how well knowledge graphs function in a variety of applications. Therefore, we are motivated to develop urban knowledge graphs to represent the urban environment for various urban applications, including cellular traffic generation.

7 DISCUSSION AND APPLICATION ON SYSTEMS

ADAPTIVE can be used for base station site selection for new regions or cities lacking historical data. Base station deployment directly affects communication services and network performance. Our method verifies that the urban structure information can be used to realize the generation of cross-city transfer learning for cellular traffic generation, which can assist in the deployment of 5G base stations. Notably, ADAPTIVE provides a general 'what-if' service, i.e., what the traffic pattern of the base station would be if it were deployed in a supposed manner. The user can then adjust the deployment plan of base stations based on the generated traffic patterns. The pre-defined base station graph can be obtained with two commonly used approaches. One hands-on approach is to be designed by communication experts. Experts can use our model to adjust and improve their manually designed schemes. The other approach is to generate a base station deployment scheme with a reinforcement learning model. The interaction with our model can then improve the deployment schemes. ADAPTIVE has been deployed on the Jiutian Artificial Intelligence (AI) Platform[1]. Jiutian Artificial Intelligence (AI) Platform is China Mobile's self-developed AI innovation platform, providing intelligent decision-making support for mobile networks. ADAPTIVE can act as a basic model to support downstream applications, including intelligent base station deployment, mobile network simulation, and benchmark dataset generation.

8 CONCLUSION

In this paper, we investigated the problem of cellular traffic generation for cities without historical data to assist 5G base station deployments. To solve this problem, we proposed **ADAPTIVE**, a deep transfer learning framework for city-scale cellular traffic generation through the urban knowledge graph. Extensive experiments on real-world datasets demonstrated the effectiveness, generalization, and robustness of our proposed methods. ADAPTIVE has been successfully deployed on the 'Jiutian' Artificial Intelligence Platform of China Mobile to support cellular traffic generation and assist in intelligent base station deployment, mobile network simulation, and benchmark traffic dataset generation.

ACKNOWLEDGMENTS

This research has been supported in part by the National Key Research and Development Program of China under Grant 2022ZD0116402; in part by the National Natural Science Foundation of China under Grant U1936217, Grant U22B2057, Grant 62272260, Grant 62171260, and Grant 62272262; in part by the International Postdoctoral Exchange Fellowship Program (Talent-Introduction Program) under YJ20210274; in part by the China Postdoctoral Science Foundation under Project 2022M721891; in part by the joint project of China Mobile Research Institute & Tsinghua University; and in part by the grant from the Guoqiang Institute, Tsinghua University under 2021GQG1005.

[1] https://jiutian.10086.cn/portal/

REFERENCES

[1] Jeffrey G Andrews, Stefano Buzzi, Wan Choi, Stephen V Hanly, Angel Lozano, Anthony CK Soong, and Jianzhong Charlie Zhang. 2014. What will 5G be? *IEEE Journal on selected areas in communications* 32, 6 (2014), 1065–1082.

[2] Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein generative adversarial networks. In *International conference on machine learning*. PMLR, 214–223.

[3] Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. 2018. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Workshop Track Proceedings*.

[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] 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.

[6] Xiaojun Chen, Shengbin Jia, and Yang Xiang. 2020. A review: Knowledge reasoning over knowledge graph. *Expert Systems with Applications* 141 (2020), 112948.

[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] Lalit Chettri and Rabindranath Bera. 2019. A comprehensive survey on Internet of Things (IoT) toward 5G wireless systems. *IEEE Internet of Things Journal* 7, 1 (2019), 16–32.

[9] Ming Ding, Chang Zhou, Qibin Chen, Hongxia Yang, and Jie Tang. 2019. Cognitive Graph for Multi-Hop Reading Comprehension at Scale. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 2694–2703.

[10] Jinhua Dou, Jingyan Qin, Zanxia Jin, and Zhuang Li. 2018. Knowledge graph based on domain ontology and natural language processing technology for Chinese intangible cultural heritage. *Journal of Visual Languages & Computing* 48 (2018), 19–28.

[11] 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.

[12] Bent Fuglede and Flemming Topsoe. 2004. Jensen-Shannon divergence and Hilbert space embedding. In *International symposium onInformation theory, 2004. ISIT 2004. Proceedings.* IEEE, 31.

[13] Global mobile suppliers association. 2022. 5G-Market Snapshot June 2022. https://gsacom.com/paper/5g-market-snapshot-june-2022/.

[14] Jiahui Gong, Qiaohong Yu, Tong Li, Haoqiang Liu, Jun Zhang, Hangyu Fan, Depeng Jin, and Yong Li. 2023. Scalable Digital Twin System for Mobile Networks with Generative AI. In *Proceedings of the 21st ACM International Conference on Mobile Systems, Applications, and Services*.

[15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. *Commun. ACM* 63, 11 (2020), 139–144.

[16] Qingyu Guo, Fuzhen Zhuang, Chuan Qin, Hengshu Zhu, Xing Xie, Hui Xiong, and Qing He. 2020. A survey on knowledge graph-based recommender systems. *IEEE Transactions on Knowledge and Data Engineering* 34, 8 (2020), 3549–3568.

[17] Emna Hajlaoui, Aida Zaier, Abdelhakim Khlifi, Jihed Ghodhbane, Mouna Ben Hamed, and Lassâad Sbita. 2020. 4G and 5G technologies: A Comparative Study. In *2020 5th International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)*. IEEE, 1–6.

[18] Paul Heckbert. 1995. Fourier transforms and the fast Fourier transform (FFT) algorithm. *Computer Graphics* 2 (1995), 15–463.

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

[20] Xiao Huang, Jingyuan Zhang, Dingcheng Li, and Ping Li. 2019. Knowledge graph embedding based question answering. In *Proceedings of the twelfth ACM international conference on web search and data mining*. 105–113.

[21] Shuodi Hui, Huandong Wang, Tong Li, Xinghao Yang, Junlan Feng, Lin Zhu, Chao Deng, Pan Hui, Depeng Jin, and Yong Li. 2023. Large-scale Urban Cellular Traffic Generation via Knowledge-Enhanced GANs with Multi-Periodic Patterns. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

[22] Baofeng Ji, Xueru Zhang, Shahid Mumtaz, Congzheng Han, Chunguo Li, Hong Wen, and Dan Wang. 2020. Survey on the internet of vehicles: Network architectures and applications. *IEEE Communications Standards Magazine* 4, 1 (2020), 34–41.

[23] 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* 33, 2 (2021), 494–514.

[24] Yifan Jiang, Shiyu Chang, and Zhangyang Wang. 2021. Transgan: Two pure transformers can make one strong gan, and that can scale up. *Advances in Neural Information Processing Systems* 34 (2021), 14745–14758.

[25] James M Joyce. 2011. Kullback-leibler divergence. In *International encyclopedia of statistical science*. Springer, 720–722.

[26] Thomas N Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *International Conference on Learning Representations*.

[27] Tong Li, Tristan Braud, Yong Li, and Pan Hui. 2021. Lifecycle-Aware Online Video Caching. *IEEE Transactions on Mobile Computing* 20, 8 (2021), 2624–2636.

[28] Tong Li, Yong Li, Mohammad Ashraful Hoque, Tong Xia, Sasu Tarkoma, and Pan Hui. 2022. To What Extent We Repeat Ourselves? Discovering Daily Activity Patterns Across Mobile App Usage. *IEEE Transactions on Mobile Computing* 21, 4 (2022), 1492–1507.

[29] Tong Li, Tong Xia, Huandong Wang, Zhen Tu, Sasu Tarkoma, Zhu Han, and Pan Hui. 2022. Smartphone App Usage Analysis: Datasets, Methods, and Applications. *IEEE Communications Surveys & Tutorials* 24, 2 (2022), 937–966. https://doi.org/10.1109/COMST.2022.3163176

[30] Peng Lin, Qingyang Song, F Richard Yu, Dan Wang, Abbas Jamalipour, and Lei Guo. 2021. Wireless virtual reality in beyond 5G systems with the internet of intelligence. *IEEE Wireless Communications* 28, 2 (2021), 70–77.

[31] 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.

[32] 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.

[33] Yu Liu, Jingtao Ding, Yanjie Fu, and Yong Li. 2023. UrbanKG: An Urban Knowledge Graph System. *ACM Transactions on Intelligent Systems and Technology* 14, 4 (2023), 1–25.

[34] Xiuquan Qiao, Pei Ren, Guoshun Nan, Ling Liu, Schahram Dustdar, and Junliang Chen. 2019. Mobile web augmented reality in 5G and beyond: Challenges, opportunities, and future directions. *China Communications* 16, 9 (2019), 141–154.

[35] 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.

[36] 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.

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

[38] 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.

[39] 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.

[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] Yucheng Yin, Zinan Lin, Minhao Jin, Giulia Fanti, and Vyas Sekar. 2022. Practical gan-based synthetic ip header trace generation using netshare. In *Proceedings of the ACM SIGCOMM 2022 Conference*. 458–472.

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

[44] Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. 2020. Graph neural networks: A review of methods and applications. *AI open* 1 (2020), 57–81.

[45] Zhilun Zhou, Yu Liu, Jingtao Ding, Depeng Jin, and Yong Li. 2023. Hierarchical Knowledge Graph Learning Enabled Socioeconomic Indicator Prediction in Location-Based Social Network. In *Proceedings of the ACM Web Conference 2023*. 122–132.