STTF: A Spatiotemporal Transformer Framework for Multi-task Mobile Network Prediction

Jiahui Gong, Yu Liu, Tong Li,
Jingtao Ding, Zhaocheng Wang, and Depeng Jin

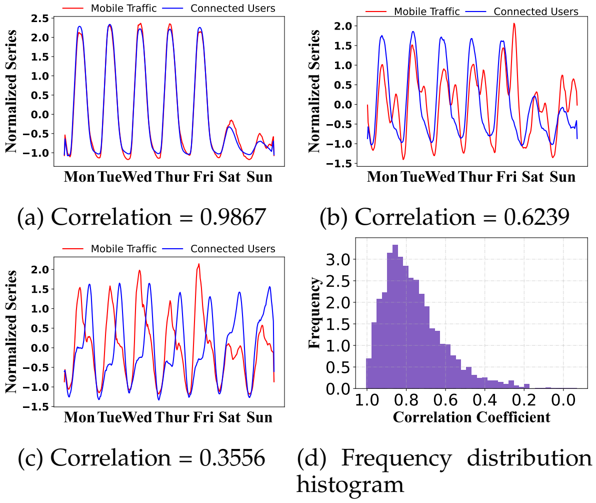
Abstract—Accurately predicting mobile traffic and accessed user amount is of great importance to network resource allocation, energy saving, etc. However, due to the complicated environmental contexts and complex interaction between mobile traffic and connected users, mobile network prediction is still challenging. Besides, the existing works could not be applied to large-scale networks because of the limited hardware resources and unacceptable time cost. In this work, we propose the spatiotemporal transformer framework for the multi-task mobile network prediction. Our proposed model contains three key parts. First, to capture the complex interaction between mobile traffic and connected users, we propose the temporal cross-attention encoder. Then, to identify and extract the most relevant information from various semantic relationships, we propose the hierarchical spatial encoder. This information is then used to create a more comprehensive representation of the network. Finally, the subgraph sampling method could significantly reduce the amount of computing power required and have comparable performance to the methods that input the whole network, enabling the model for real-world applications. Extensive experiments demonstrate that our proposed model significantly outperforms the state-of-the-art models by over 17% in both mobile traffic prediction and connected user prediction.

Index Terms—Large-scale network; transformer; mobile traffic prediction

1 INTRODUCTION

The explosive growth of smartphones and the Internet of Things (IoT) has resulted in a doubling of global mobile network traffic in just two years, reaching a staggering 115 EB in Q4 of 2022 [1]. This trend poses a significant challenge for communication operators, who must effectively allocate network resources and maintain high-quality service to ensure a satisfactory user experience. In the domain of mobile networks, the number of users and mobile traffic are critical factors that impact the performance of a base station [2]. As more users connect to a base station, the generated volume of mobile traffic increases proportionally, leading to slower data transfer rates, increased latency, dropped connections, and, ultimately, network congestion [3].

Mobile network prediction refers to the process of predicting the future number of connected users and the volume of mobile traffic generated in a mobile cellular network to enable communication operators to allocate network resources effectively, maintain high-quality service and detect abnormal user behaviours [4], [5], [6]. Compared to the separate prediction, the joint prediction could better reflect this interdependent relationship and more accurately predict the future base station traffic and user count. Besides, joint prediction only requires a comprehensive model which could reduce the number of models to lower the maintenance burden on communication operators. Moreover, the separate prediction may result in inconsistent trends between the two predictions leading to incorrect network resource allocation. However, currently, there is a lack of joint prediction methods for mobile traffic and the number of connected users.



(a) Correlation = 0.9867 (b) Correlation = 0.6239

(c) Correlation = 0.3556 (d) Frequency distribution histogram

Fig. 1: The correlation coefficient between the mobile traffic and connected users of different base stations varies greatly.

Looking back at existing works and digging into the practical mobile cellular network, we encounter three major challenges for the joint prediction of mobile traffic and connected users.

How to capture the complex interaction between mobile traffic and connected users? Mobile traffic and connected users are two critical elements of the base station, which are closely intertwined. However, Figure 1 highlights the diverse correlations between mobile traffic and connected users in various base stations, underlining the

- *J. Gong, Y. Liu, T. Li, J. Ding, Z. Wang, and D. Jin are with the Department of Electronic Engineering, Beijing National Research Center for Information Science and Technology (BNRist), Tsinghua University, Beijing 100084, China.*
E-mail: tongli@mail.tsinghua.edu.cn.

importance of analyzing these two factors together. However, existing studies on mobile cellular network prediction have mainly focused on modelling mobile traffic and have overlooked the impact of connected users on network performance.

How to identify rich semantic relationships between base stations? The relationships between base stations are diverse and complex, with many factors influencing their performance and the mobile traffic they generate such as the distance relationship and the similar flow relationship. To improve the accuracy of prediction, it is essential to capture the relationships that exist between base stations. However, existing works largely focus on the single spatial relationship, failing to capture further semantics between base stations [7], [8], [9].

How to build a scalable and powerful model for large-scale mobile networks? The size of networks presents a significant challenge to machine computing power. Researchers commonly use graph neural network (GNN) [10] and graph convolutional network (GCN) [11] to capture relationships between base stations, effectively modelling complex dependencies and interactions between nodes [12], [13], [14], [15], [16], [17]. However, due to the $O(mn)$ and $O(n^2)$ computing complexity of GNN and GCN, their computation becomes increasingly difficult and requires significant computing power as the graph size increases. Additionally, GNN and GCN have limited generalization to new graphs or removed nodes, as they are highly dependent on specific graph structures.

To overcome the aforementioned challenges, in this study, we propose a model that is applicable to large-scale networks for joint prediction of mobile traffic and the number of connected users. To address the first challenge, we propose the cross-attention mechanism in the temporal encoder. This mechanism allows us to exchange features between mobile traffic and connected users, enabling the model to effectively capture the complex interactions between these two types of data. To address the second challenge, we propose a hierarchical spatial attention mechanism to capture the various spatial features through different semantic relationships, which allows us to effectively identify and utilize the most relevant information from different levels of the network. We then fuse these features to produce a more comprehensive representation of the network. To address the third challenge, we design a subgraph sampling strategy. By limiting each node in the network to interact with only a specific number of its neighbours in one hop, we can reduce the computational complexity of the network, making it more manageable for real-world applications.

The main contributions can be summarized as follows:

- We address the joint prediction problem of mobile traffic and the number of connected users, uncovering their intricate correlation which provides benefits for joint prediction. Compared to separate predictions, joint prediction considers interdependence, reduces complexity and provides a more comprehensive understanding of the mobile network.
- We propose a spatiotemporal transformer framework (STMP) for the multi-task mobile network prediction in large-scale networks. Our framework includes cross-attention and hierarchical spatial attention mechanisms to capture complex interactions between these data and identify semantic relationships between base stations. Additionally, we design a subgraph sampling strategy to reduce computing power requirements.
- Our model is evaluated on two real-world datasets, demonstrating its accuracy and effectiveness. Extensive experiments demonstrate that our proposed model significantly outperforms the state-of-the-art models by over 17% in both mobile traffic prediction and connected user prediction. We also analyze how different types of base stations prioritize semantic relationships and the influence of the subgraph sampling strategy's number of neighbours and hops.

2 PRELIMINARY

2.1 Semantic Relationships between Base Stations

To better understand the traffic and user prediction tasks, we discover four relationships between base stations to capture their spatial and temporal features [12], [18], [19]. These four relationships model the relationships between base stations from different temporal and spatial perspectives.

Proximity Relationship. The proximity relationship between two base stations is established when their physical locations are within a certain distance of each other. The formulation can be formed as,

$$a_{i,j} = \begin{cases} exp(-\frac{dis_{i,j}}{\sigma^2}), & exp(-\frac{dis_{i,j}}{\sigma^2}) \geq \epsilon \\ 0, & exp(-\frac{dis_{i,j}}{\sigma^2}) < \epsilon \end{cases} \tag{1}$$

where $dis_{i,j}$ denotes the distance between base station i and base station j, and σ and ϵ are thresholds to control the distribution and sparsity of the adjacency matrix of proximity relationship, where we set σ and ϵ are 37 and 0.5. When two base stations are closer, there is a higher likelihood of traffic shifting between them because mobile users are more likely to connect to a nearby base station if they experience low signal strength or disconnect from their current base stations. This behaviour of mobile users can result in traffic shifting between the two base stations, which can affect the overall traffic patterns and usage of the mobile network.

Function Similarity. POI(Point of Interest) is the basic functional unit and place in a city, such as schools, hospitals, shopping malls, etc. It is a fine-grained place where people carry out social production and life in the city. People in POIs with similar functions will generate traffic with similar patterns. For instance, during rush hours, people tend to commute to and from work at similar times, resulting in the generation of large amounts of traffic simultaneously by base stations located near traffic hubs. So, if POI distributions of base stations are similar, their traffic flows are more likely similar. We first calculate the number of each category of POI in the vicinity of the base station and then calculate the cosine similarity of each base station to get the similarity matrix. The formulation can be formed as,

$$b_{i,j} = cos(v_i^{POI}, v_j^{POI}), \tag{2}$$

TABLE 1: Summary of Notations.

Notations	Definition
bs, BS	A base station and the set of base stations
t_i^t, u_i^t	The traffic value and the connected user value of base station bs_i at time step t
I, P	The input length and the patch length
$\mathbf{t}_i^t, \mathbf{u}_i^t$	The traffic and connected user sequence of bs_i at t in the past I steps
N_g, N_n	Number of graph nodes and number of sampling neighbours
E_t, E_u	Input of traffic encoder and user encoder
P_t, P_u	Position embedding of traffic and connected users
T_t, T_u	Temporal embedding of traffic and connected users
S_t, S_u	Spatial embedding of traffic and connected users
F_t, F_u	Fuse embedding of traffic and connected users
D_t, D_u	Input of traffic decoder and user decoder

where $b_{i,j}$ denotes the cosine similarity of POI distribution between base station i and base station j, and the v_i^{POI} represents the POI distribution vector of base station i. For each base station, we select the top 20 base stations with the highest similarity and consider them to have a function similarity relationship.

Pattern Similarity. After an in-depth analysis, we found that each base station has a unique and consistent traffic pattern that reflects its typical usage by mobile users. Although a base station's traffic pattern is relatively stable, there is still some variability in real mobile traffic flow, which fluctuates around the traffic pattern on a weekly scale. Additionally, a base station's location can influence its traffic pattern. To group similar base stations based on their normalized patterns, we use the hierarchical clustering method, an unsupervised machine learning method. If two base stations are in the same cluster and located within a certain distance, we consider them to have a pattern similarity relationship, which can be formed as,

$$c_{i,j} = \begin{cases} 1 \cdot a_{i,j} \ , & v_i^p = v_j^p \\ 0, & otherwise. \end{cases} \tag{3}$$

where v_i^p denotes the results of clustering the pattern series of base station i.

Flow Similarity. Though base stations have a pattern similarity relationship, their traffic flow may still be very different, such as their absolute value and the changing speed of their traffic flows. To further explore the temporal features of base stations, we propose a Similar Series relationship. We use dynamic time warping methods [20] to calculate the similarity between the traffic series of base stations and get the similarity matrix D. The formulation can be formed as,

$$d_{i,j} = DTW(v_i^{flow}, v_j^{flow}), \tag{4}$$

where $d_{i,j}$denotes the traffic series similarity between base station i and base station j, and v_i^{flow} represents the traffic flow of base station i. We select the top 20 base stations with the highest similarity and consider them to have the flow similarity relationship.

3 System Model and Problem Definition

3.1 System Model

This study considers a heterogeneous network scenario, as illustrated in Figure 2, with a network architecture comprising three core components: a central controller module, an edge server module, and base stations. The central controller manages the operational status of each edge server, oversees resource allocation, and coordinates key functions to ensure seamless network performance. Each edge server performs essential tasks such as perception, data collection, analysis, and storage. It also controls the base stations within its vicinity, managing their data uploads and downloads.

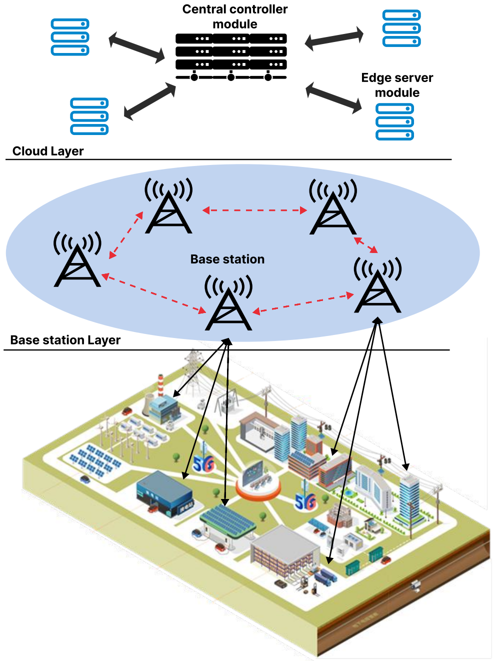


Fig. 2: The diagram of mobile traffic prediction.

The centralized prediction model architecture is built on four key modules: local data collection, local data upload, global model training, and prediction result delivery. Each edge server collects traffic and the connected user data from its associated base stations preprocesses this data, and uploads it to the central controller. The central controller aggregates data from all edge servers to build a unified model generates global predictions, and then distributes these predictions back to each edge server. Based on these predictions, the edge servers manage the power settings and operational status—such as sleep or active mode—of their respective base stations.

Traffic data and the connected user data are critical indicators of base station load. Mobile traffic represents the total volume of data transmitted through a base station over a specific period, encompassing activities such as voice calls, text messages, internet browsing, video streaming, and other services initiated by mobile devices. This traffic, typically measured in units like kilobytes (KB), reflects the intensity of network usage and data demand within a base station's coverage area. Higher traffic volumes indicate greater data demand, placing increased pressure on the base station's resources.

Similarly, the number of connected users indicates how many mobile devices or users are actively utilizing the network at any given time. These users, within a base station's coverage area, engage in activities like voice calls, messaging, or data services, directly impacting the station's load. The interplay between user numbers and traffic levels determines the overall load, influencing network performance, service quality, and resource allocation efficiency. Effectively managing this load enables network operators to optimize base station capacity and maintain consistent service for all users.

Early traffic forecasting approaches often focused solely on the temporal distribution of traffic at individual base stations, using statistical methods or simple machine learning models. However, in urban environments, evaluating the load of each base station in isolation is insufficient. Users frequently move between base stations, bringing their data needs with them. This mobility causes the load to shift dynamically across multiple base stations as traffic volumes and user density change according to movement patterns. Thus, the load on any given base station is influenced not only by its local users and traffic but also by user activities and movements among neighboring stations. Understanding these spatial interactions is crucial for accurate load forecasting and effective network management.

In addition to spatial interactions, historical data plays a vital role in predicting future base station loads. By analyzing past patterns of user connections and mobile traffic, trends and recurring behaviors can be identified, offering valuable insights into future usage patterns.

The centralized forecasting model integrates spatial factors, enabling a comprehensive analysis that considers both temporal and spatial dimensions. By leveraging both historical and current data, this model achieves greater accuracy than traditional single time-series forecasting methods. It can simultaneously predict mobile traffic and the number of connected users by capturing the inherent correlation between these indicators, while also accounting for the spatiotemporal dependencies between base stations. This capability allows for more accurate forecasts, improved resource allocation, and optimized network performance in urban areas.

3.2 Problem Definition

The joint prediction is a multi-task learning problem where both tasks are related but distinct. Modeling them together allows us to improve predictive accuracy by learning a shared representation for both tasks. Specifically, the number of users $\mathbf{u}_t$ affects traffic patterns directly through usage behavior, while traffic $\mathbf{t}_t$ reflects the underlying user activity. However, it is not sufficient to model this relationship within a single base station, as users and traffic between different base stations can affect each other at varying times. This inter-base station interaction introduces a more complex dynamic, where the user count and traffic load at one station may influence those at neighboring stations.

To capture this complexity, it is essential to model both tasks jointly, accounting for the spatial-temporal dependencies between base stations. Predicting mobile traffic and user count together helps us capture the nuanced interactions between user numbers and traffic load across stations.

Mathematically, we refine the joint modeling framework to capture these interdependencies. The updated joint prediction problem can be formulated as estimating a mapping function f that captures the interactions between mobile traffic $\mathbf{t}_t$, the number of users $\mathbf{u}_t$, and the spatial-temporal relationships between base stations:

$$\hat{t}_{t+1}, \hat{u}_{t+1} = f_\theta(\mathbf{t}_t, \mathbf{u}_t, g(\mathbf{t}_t, \mathbf{u}_t)) \tag{5}$$

where θ are the model parameters, $g(\mathbf{t}_t, \mathbf{u}_t)$ is a function that explicitly models the interaction between the number of users and traffic at time t, incorporating dependencies both within and between base stations. This function leverages the mobile traffic prediction task and user count prediction task to improve the model's generalization by capturing shared features. The detailed definition of notations is illustrated in Table 1.

4 Method

We first elaborate on the general framework of the spatiotemporal transformer framework (STMP) as shown in Figure 3. We have adopted the encoder-decoder framework of the transformer [21] in our model. The mobile traffic and user series are first encoded separately and then decoded using the same decoder. To capture the temporal features and complex interactions between these two types of data, we first apply the temporal cross-attention encoder. After that, we apply the hierarchical spatial encoder to capture and integrate diverse spatial features. Then, we apply the same decoder to decode the embeddings of mobile traffic and connected users.

4.1 Temporal Cross-Attention Encoder

Patching. To process the input time series in an efficient manner, we adopt the patch-based approach in our model. The input series is first divided into patches which can be either overlapped or non-overlapped. We denote the patch length as P and non-overlapped, then the patching process will divide the historical series with length I into N_p patches, where $N_p = \lfloor \frac{I}{P} \rfloor + 1$. With the use of patches, the number of input tokens can reduce from I to approximately $\frac{I}{P}$, which implies the memory usage and computational complexity of the attention map are significantly decreased. Moreover, patches can capture the local shapes of the time series by being aware of the local context instead of the point-wise value of the time series. This allows the model to better capture the local patterns and fluctuations in the data.

Temporal cross-attention encoder. Then, to make our model use the sequence, we add the positional embedding [21] to the input embedding before the encoder. The Positional Embedding has the same dimension as the embeddings so that the two can be summed. In this work, we use sine and cosine functions. In the temporal cross-attention encoder, we apply the multi-head cross-attention mechanism. The multi-head mechanism could focus on the different aspects of temporal features, both long and short terms. The cross-attention mechanism could exchange the features of the mobile traffic and users, which is a method used to exchange information between different input sources. By implementing the cross-attention mechanism, the model can learn to exchange relevant features between mobile traffic and connected users, allowing for a more comprehensive understanding of relationships between two inputs and

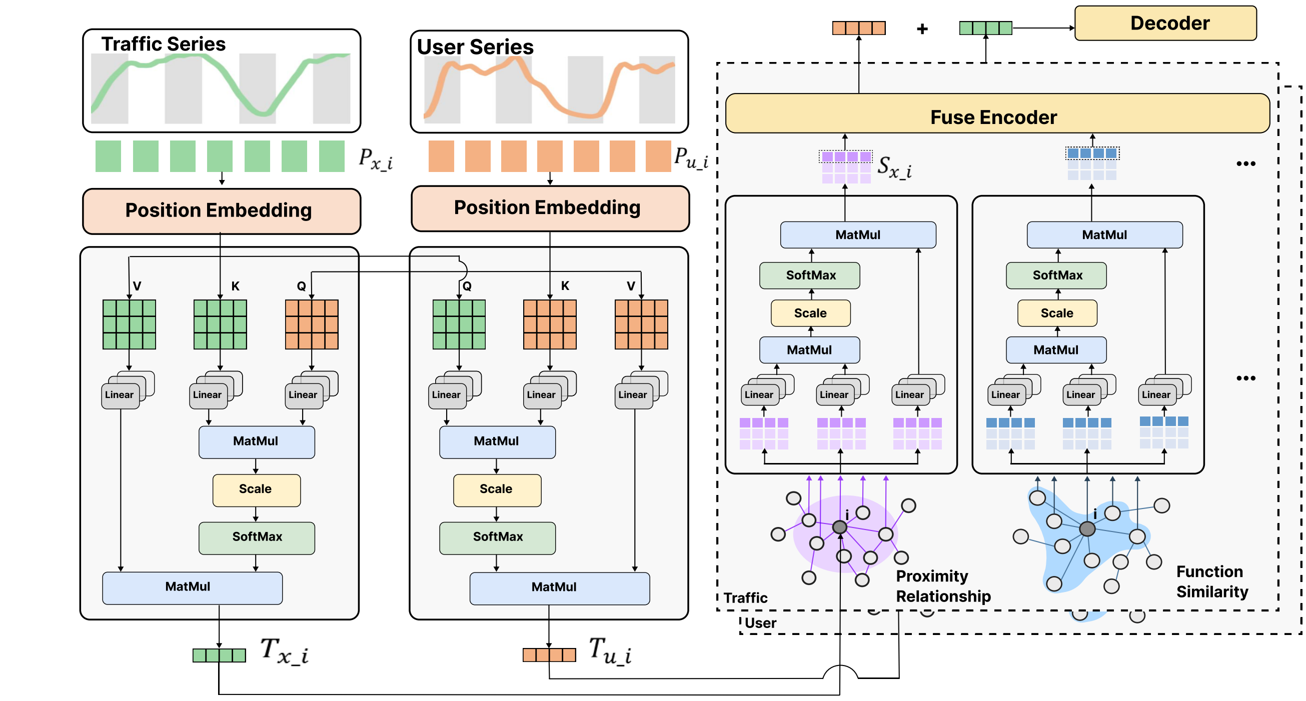


Fig. 3: The Framework Overview of STMP. The Temporal Encoder applies the Cross-Attention mechanism. The Spatial Encoder applies the Hierarchical Spatial Attention mechanism.

better capturing the complex and dynamic nature of mobile networks and improving the accuracy of prediction, which can be formed as,

$$\begin{aligned}\boldsymbol{T_t} =& CrossAttention(\boldsymbol{F_t}, \boldsymbol{F_u}, \boldsymbol{F_u}) \\ =& MultiHead(\boldsymbol{F_t}, \boldsymbol{F_u}, \boldsymbol{F_u}) \\ =& Concat(head_1, ..., head_h)W^0\end{aligned} \tag{6}$$

$$head_i = Attention(\boldsymbol{F_t} W_i^Q, \boldsymbol{F_u} W_i^K, \boldsymbol{F_u} W_i^V) \tag{7}$$

$$Attention(Q, K, V) = softmax(\frac{\mathbf{QK}^T}{\sqrt{d_k}})\mathbf{V}. \tag{8}$$

In the cross-attention mechanism, the queries are modified to the intermediate features of other tasks which introduce cross-interaction between the mobile traffic and users. These interactions are further exploited to obtain the temporal traffic features $\boldsymbol{T_t}$ and user features $\boldsymbol{T_u} \in \mathbb{R}^{N_p \times d}$, where d denotes the embedding dimension.

4.2 Hierarchical Spatial Encoder

Subgraph Sampling. To efficiently apply our model to large-scale mobile networks, we utilize subgraph sampling. This approach breaks down the entire network into smaller, more manageable subgraphs, which are processed independently. For each semantic relationship, we constrain each node to interact with a limited number N_n of its nearest neighbours within one hop. In other words, when processing a subgraph, each node only considers its closest neighbours. This method greatly reduces computational complexity, enhances scalability, and facilitates parallel processing, making the model more efficient and manageable for large networks.

Hierarchical spatial encoder. To create an efficient hierarchical spatial encoder, we introduce a novel hierarchical spatial attention mechanism that integrates diverse spatial features. Our proposed encoder is composed of two distinct Transformer blocks, each serving a specific purpose in feature extraction. The first block leverages a Transformer encoder to capture correlations between the central node and its neighboring nodes for each semantic relationship. The second block, a fusion encoder, is designed to select and combine the most relevant features from various relationships. This hierarchical approach enables more comprehensive spatial feature learning and can be formulated as follows:

$$S_t = SelfAttention(T_t) \tag{9}$$

$$S_c = Split(S_t) \tag{10}$$

$$F_t = SelfAttention(S_c) \tag{11}$$

where the *split* means to extract the embedding of the central nodes. This approach enables the integration of a wide range of environmental features, resulting in a robust and flexible architecture adaptable to various applications. By employing this method, we create an encoder capable of effectively combining diverse environmental features, providing a comprehensive and nuanced understanding of the surrounding space.

To elaborate, the first block of the encoder is designed to perform attention between the central node and its neighboring nodes for each semantic relationship. This attention mechanism extracts the most relevant spatial features for the central node. The resulting embeddings for each central node are then passed into the second block of the spatial encoder, known as the fuse encoder. This block performs attention across embeddings from different semantic relationships, fusing them to produce a unified and refined representation.

4.3 Prediction and Training

The input of the decoder consists of two parts: the latter part of the historical series with length $I-O$ to provide the recent information, and placeholders with length O filled with zero. Then, the patching module and position embedding module are applied to be aware of the local context. Besides, The encoded embedding is obtained by concatenating the output of the mobile traffic and connected users that have passed through the hierarchical spatial encoder. Based on the input and the embedding, the decoder could combine the spatial and temporal dependencies and generate the prediction of mobile traffic and connected users. After the decoder, we utilize two Multi-layer Perceptron (MLP) as the predictor to predict the future mobile traffic $\hat{a}^{t+1}$ and connected users $\hat{b}^{t+1}$ respectively.

The computational complexity of the model is primarily driven by three key components: patch embeddings, temporal encoders, and the hierarchical spatial encoder. The patch embedding process, which converts input sequences into patches, has a complexity of $O(N_P \times d)$, where N_P is the number of patches, and d is the embedding size. The temporal encoder and the hierarchical spatial encoder, which leverages Transformer layers to capture temporal and spatial dependencies, contribute a complexity of $O(N_P^2 \times d)$ each. Since these components function sequentially, their complexities are additive, leading to an overall model complexity of $O(N_P^2 \times d)$.

Algorithm 1 outlines the spatiotemporal transformer framework (STMP) training procedure. STMP is designed to jointly predict mobile traffic $\hat{t}^{t+1}$ and the number of connected users $\hat{u}^{t+1}$ at base stations by effectively leveraging both temporal and spatial relationships. Initially, the mobile traffic and user data are divided into patches and transformed into patch embeddings $\boldsymbol{P_t}, \boldsymbol{P_u}$. These embeddings are processed through temporal encoders to capture time-dependent features and enable feature exchange between traffic $\boldsymbol{T_t}$ and user $\boldsymbol{T_u}$. The hierarchical spatial encoder captures correlations between the central node and its neighboring nodes, accounting for various semantic relationships. Finally, the decoder combines both spatial and temporal dependencies to predict future mobile traffic and the number of connected users. STMP is trained in mini-batches to minimize the difference between predicted values $\hat{t}^{t+1}, \hat{u}^{t+1}$ and their corresponding true values t^{t+1}, u^{t+1}. We use the Mean Squared Error (MSE) loss function to optimize the model's parameters, which can be expressed as follows,

$$\mathcal{L} = \left\|\hat{a}^{t+1} - a^{t+1}\right\|_2 + \left\|\hat{b}^{t+1} - b^{t+1}\right\|_2 . \tag{12}$$

Algorithm 1: Spatiotemporal Transformer Framework (STMP)

```
Input: The input of Encoder
       $\boldsymbol{E_t}, \boldsymbol{E_u} \in \mathbb{R}^{N_g \times (N_n+1) \times I}$, The input of
       Decoder $\boldsymbol{D_t}, \boldsymbol{D_u} \in \mathbb{R}^{I}$, the patch length $P$
Output: Mobile traffic and connected users
        prediction for the next time step $\hat{t}^{t+1}, \hat{u}^{t+1}$
$\boldsymbol{P_t}, \boldsymbol{P_u}$ = Patch Embedding ($\boldsymbol{E_t}, \boldsymbol{E_u}$)
    // $\boldsymbol{P_t}, \boldsymbol{P_u} \in N_g \times (N_n+1) \times \frac{I}{P} \times d$, Segment the input and transform each
    patch into embedding space.
$\boldsymbol{T_t}, \boldsymbol{T_u}$ = Temporal Encoder ($\boldsymbol{P_t}, \boldsymbol{P_u}$) // Capture
    temporal dependencies and
    interactions, according to Equ 7.
$\boldsymbol{S_t}, \boldsymbol{S_u}$ = Spatial Encoder ($\boldsymbol{T_t}, \boldsymbol{T_u}$) // Capture
    spatial correlations among different
    regions and their neighbors.
$\boldsymbol{F_t}, \boldsymbol{F_u}$ = Fuse Encoder ($\boldsymbol{S_t}, \boldsymbol{S_u}$)
    // $\boldsymbol{F_t}, \boldsymbol{F_u} \in \mathbb{R}^{N_g \times \frac{I}{P} \times d}$, Combine the
    spatially- and temporally-aware
    representations of traffic and user
    data.
$\boldsymbol{M_t}$ = Mean($\boldsymbol{F_t}$), $\boldsymbol{M_u}$ = Mean($\boldsymbol{F_u}$)
$\boldsymbol{F}_{en}$ = Concat($\boldsymbol{F_t}, \boldsymbol{F_u}$), $\boldsymbol{F}_{de}$ = Concat($\boldsymbol{D_t}, \boldsymbol{D_u}$) ;
$\boldsymbol{P}_{de}$ = Patch Embedding ($\boldsymbol{F}_{de}$) ;
$\boldsymbol{H}_{de}$ = Decoder ( $\boldsymbol{P}_{de}, \boldsymbol{F}_{en}$) // $\boldsymbol{H}_{de} \in \mathbb{R}^{2\frac{I}{P} \times d}$
$\hat{t}^{t+1}, \hat{u}^{t+1}$ = Predictor ($\boldsymbol{H}_{de}$) // Predict the
    future traffic and connected users.
```

TABLE 2: Statistics of the datasets used in our experiments.

Dataset	Shanghai	Nanjing
Collection Duration	Aug. 1st - 31st, 2014	Feb. 2nd - Mar. 31st, 2021
Time Interval	30 minutes	
Covered Users	≥150,000	≥1,250,000
Covered BSs	4505	14724
Covered Area	6340 km^2	6587 km^2
Flow Records	8.65× 10^8	8.18× 10^8
Number of logs	6,703,440	40,991,616

5 Experiment and Results

5.1 Experimental Settings

5.1.1 Datasets.

Shanghai Dataset. The Shanghai Datasets are anonymous base station mobile traffic collected from August 1st to August 31st, 2014 in Shanghai [22]. The datasets contain 4505 base stations and more than 150,000 users. Each dataset entry includes the anonymous ID of the device, the start and the end time, the anonymous ID of the base station the device is connected to, and the amount of data transmitted in the connection. We contribute 1.96 billion tuples of entries to 4505 base stations every 30 minutes, according to the tracing logs.

Nanjing Dataset. The Nanjing Datasets are also anonymous base station mobile traffic collected from February 2nd to March 31st, 2021 [23]. The datasets contain 14724 base stations in Nanjing, 3.2 times larger than Shanghai Datasets, and more than 1,250,000 users. Each time step is 30 minutes.

Table 2 shows the statistics of the Shanghai dataset and Nanjing dataset. The large-scale and fine-grained datasets can ensure the validity of mobile traffic reality and the model test.

TABLE 3: Overall prediction performance of our model in comparison with compared algorithms on Shanghai and Nanjing datasets.

	Shanghai Dataset						Nanjing Dataset					
	Mobile Traffic			Number of Connected Users			Mobile Traffic			Number of Connected Users		
Model	**MAE**	**RMSE**	R^2	**MAE**	**RMSE**	R^2	**MAE**	**RMSE**	R^2	**MAE**	**RMSE**	R^2
SVR[24]	0.2092	0.3018	0.7479	36.07	47.48	0.9271	0.3130	0.4370	0.7276	1007	1358	0.8260
ARIMA[25]	0.2058	0.3041	0.7499	23.41	34.62	0.9353	0.2944	0.4115	0.7585	842	1147	0.8513
LSTM [34]	0.2036	0.1927	0.7505	25.63	33.96	0.9373	0.2912	0.4071	0.7639	754	1062	0.8619
GAT [27]	0.1984	0.2650	0.5547	43.60	57.86	0.8933	0.3861	0.5345	0.5926	1220	1642	0.8004
GraphSAGE [28]	0.2138	0.2979	0.7418	31.40	41.23	0.9244	0.3056	0.4207	0.7483	670	972	0.9095
STGCN [12]	0.1996	0.2785	0.7767	18.99	24.46	0.9601	0.2593	0.3665	0.8084	351	854	0.8926
T-GCN [13]	0.1908	0.2694	0.7990	16.80	25.59	0.9691	0.2535	0.3595	0.8157	301	777	0.9119
DeepTP [8]	0.1869	0.2610	0.7991	16.13	25.47	0.9695	0.2250	0.3305	0.8191	335	830	0.8998
GMAN [26]	0.1807	0.2554	0.8078	15.68	24.32	0.9704	0.2209	0.3237	0.8237	208	590	0.9511
MC-STGCN [29]	0.1852	0.2600	0.8003	15.97	25.07	0.9699	0.2231	0.3294	0.8195	306	790	0.9120
GinAR[30]	0.1836	0.2579	0.8027	0.16.01	26.19	0.9659	0.2254	0.3317	0.8160	257	659	0.9296
Autoformer [31]	0.2004	0.2798	0.7661	27.89	35.16	0.9606	0.2798	0.3918	0.7810	244	669	0.9371
PatchTST [32]	0.1963	0.2708	0.7791	26.11	33.98	0.9659	0.2676	0.3769	0.7974	231	639	0.9426
FR-Net[33]	0.1951	0.2679	0.7803	24.56	32.03	0.9666	0.2601	0.3638	0.7952	225	621	0.9454
our model	**0.1498**	**0.2253**	**0.8534**	**12.66**	**18.3**	**0.9845**	**0.2149**	**0.3197**	**0.8548**	**141**	**324**	**0.9827**
Improv.	17.10%	11.78%	5.64%	19.26%	23.39%	1.45%	2.71%	1.23%	3.77%	32.21%	45.08%	3.32%

5.1.2 Metrics.

To handle the large absolute value of mobile traffic and focus on the magnitude, we apply log-normalization to mobile traffic. In evaluating the performance of mobile traffic prediction, we carefully select three metrics: Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Coefficient of Determination (R^2).

Each of these metrics provides unique insights into the regression model's performance. Respectively, MAE and RMSE measure accuracy and variability, while R^2 assesses how well the model fits the data. By considering all three metrics, we can gain a more comprehensive understanding of the model's performance and its potential limitations.

5.1.3 Baselines.

We elaborately select the following ten representatives to be compared with our proposed model, which covers representative classical machine learning models [24], [25], state-of-the-art spatial-temporal models [12], [8], [26], [27], [28], [13], multi-task prediction methods [29], [30] and representative Transformer-based models [31], [32], [33].

Support Vector Regression (SVR) [24]. SVR is an extension of SVM for regression tasks that predicts continuous output based on input features. It captures complex and non-linear relationships in the data.

Auto-Regression Integrated Moving Average (ARIMA) [25]. ARIMA is a statistical model for analyzing and predicting time series data that uses three parameters to represent auto-regressive terms, moving average terms, and differences or orders.

Graph Attention Network (GAT) [27]. GAT is a dynamic graph neural network that learns node weights, capturing varying connection importance. It handles changing graphs or contextual variables. We use mask graph attention with GAT.

Graph Sample and Aggregate (GraphSAGE) [28]. GraphSAGE is a graph representation learning method that enhances the scalability and performance of GNN. It maximizes the ratio of sampling the current neighbour node to sampling the entire graph.

Deep Traffic Predictor [8]. DeepTP is an end-to-end deep learning model that predicts spatial-temporally dependent cellular traffic over a lengthy period. It handles complex and dynamic traffic patterns influenced by spatial and temporal factors, using a sequential module and a broad feature extractor.

Spatial-Temporal Graph Convolutional Network (STGCN)[12]. STGCN combines GCN and gated CNN architectures to capture spatial-temporal patterns in graph-structured data. It uses GCN to mine the graph's topology and gated CNN to explore dynamic mobile traffic features. The model predicts future mobile traffic and its code is publicly available.

Temporal-Graph Convolutional Networks(T-GCN) [13]. T-GCN combines GCN and GRU to model time series and capture the dynamic mobile traffic change of node attribution.

MC-STGCN [29] MC-STGCN contains a cross-scale GCN for learning the multi-scale spatial features a cross-scale temporal network for capturing intra- and inter-scale temporal correlations and a feature correlation learning component for capturing the feature correlations.

GinAR [30] GinAR introduces interpolation attention and adaptive graph convolution to effectively model spatial-temporal dependencies and recover missing variables in limited datasets, replacing the fully connected layers of traditional recursive units for accurate multivariate time series forecasting.

Graph Multi-Attention Network (GMAN) [26]. To describe the effect of spatial-temporal variables on traffic conditions, GMAN employs an encoder-decoder architecture with spatial-temporal attention blocks. Input traffic characteristics are encoded by the encoder, and the decoder forecasts the output time step sequence.

Autoformer [31]. Autoformer is a deep learning architecture that handles complex time series data using the Auto-Correlation mechanism to progressively decompose temporal patterns. Based on time series data periodicity, it captures sub-series dependencies to capture complex temporal relationships.

PatchTST[32]. PatchTST utilizes two key components, segmentation of time series into subseries-level patches which are served as input tokens to Transformer, and channel-independence where each channel contains a single univariate time series that shares the same embedding and Transformer weights.

FR-Net[33]. FR-Net explores dynamic period features by decomposing time series into period and trend components using frequency domain rotations, employing a period frequency rotation module for predicting the period component and a patch frequency rotation module for predicting the trend component.

5.1.4 Parameter Settings.

Our deep learning model is implemented using the Adam optimizer [35]. We utilize the Mean Squared Error (MSE) loss function and set the learning rate to 0.0005. The length of the historical traffic series I is set to 12, while the patch length p is set to 3 for predicting the next O = 1 step, which provides both higher accuracy and faster speed. To capture the semantic relationships in the data, we choose N_g = 4 relationships and N_n = 20 neighbours for each relationship. The dimension of the embedding vector in both the encoder and decoder d is set to 64 and the number of layers is set to 2. To evaluate the performance of our model, we divide the datasets into three parts: training, validation, and testing with a ratio of 0.7:0.15:0.15. We train it on a Linux server with eight GPUs (NVIDIA RTX 2080 Ti * 8). The code and data are available at https://github.com/tsinghua-fib-lab/STTF .

5.2 Overall Performance

In Table 3, we display the overall performance of our model, temporal models (SVR, ARIMA), spatial models (GAT, GraphSAGE), spatial-temporal models (DeepTP, STGCN, T-GCN, GMAN), the multi-task prediction methods (MC-STGCN, GinAR), and Transformer-based models (Autoformer, PatchTST) to predict the next time stamp of Shanghai and Nanjing. From the results, we have the following findings:

Our framework steadily achieves the best performance. Our model achieves superior results on both datasets and outperforms other compared baselines. The R^2 improvement of STMP compared with the second-best performance model (GMAN), is around 3.7% to 5.6% in mobile traffic prediction. In connected user prediction, the R^2 improvement is about 1.4% to 3.3%, and the RMSE reduction is around 23% to 45%.

Spatial models perform poorly in the mobile traffic prediction task. Spatial models are commonly used to analyze spatial data such as geographic patterns and location-based information. However, these models may not have the necessary modules to model time series data or capture temporal features. As a result, their performance may be inferior to models that incorporate temporal information. By incorporating temporal components into spatial models, we can achieve more accurate predictions and better performance in real-world applications.

It is essential to model both spatial information and temporal information. The spatial-temporal models could not only capture the spatial features but also capture the environment information, which has enhancements for spatial models and temporal models. Besides, the Transformer-based models also lack the module to capture the spatial information resulting in the poor performance of mobile traffic prediction. Meanwhile, compared with STGCN and T-GCN, we can conclude that our spatial encoder, consisting of four semantic relationships, can capture more spatial and temporal features than only the distance relationship and improve about 6.8% to 9.8%.

Cross-attention mechanism enables better feature exchange. Compared with MC-STGCN and GinAR, we could find that the cross-attention mechanism enables more effective exchange of features between mobile traffic and the number of connected users, leading to better integration of information from both sources. This mechanism allows the model to focus on relevant aspects of each feature set, improving the ability to capture complex relationships between them. In contrast, multi-task prediction methods typically rely on parameter sharing to model multiple variables simultaneously. While this approach can be efficient, it often struggles to capture the unique characteristics of each variable fully.

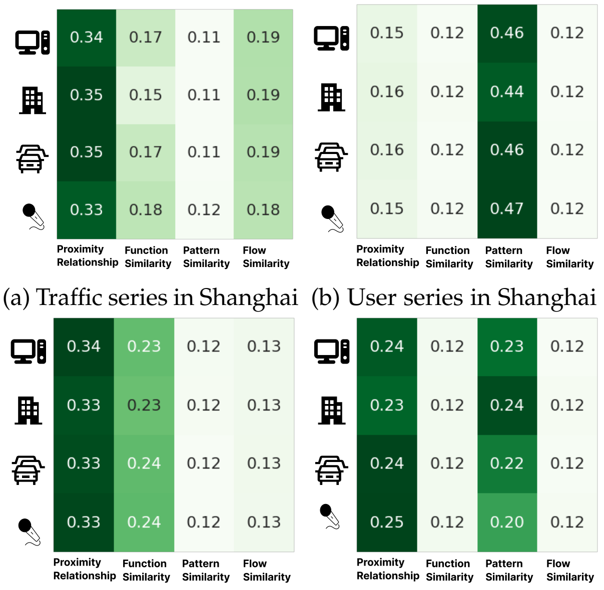
5.3 Ablation Study

To gain a better understanding of the performance of our proposed model, we conducted an ablation study that evaluated the effectiveness of four different variants of the model. Specifically, we predicted the traffic and connected users separately, removed the decoder, the temporal encoder, the spatial encoder, and the patch-based approach, and changed the cross-attention mechanism to the self-attention mechanism separately.

The results of the ablation study are presented in Table 4, which clearly shows that our model outperforms the six variants. The joint prediction could uncover the intricate correlation and improve the model performance. The temporal encoder in our model is a critical component of our model as it enables the capture of the complex interactions between mobile traffic and connected users, as well as the mining of the temporal characteristics of the data. Without the temporal encoder, the model would not be able to effectively capture these temporal dependencies, around 7%. Similarly, the spatial encoder is essential for extracting various environmental information related to base stations. Without the spatial encoder, the model would not be able to effectively obtain this information through its neighbours. The decoder component of our model plays a critical role in decoding the encoded embeddings and predicting future values accurately, which is essential for mobile traffic prediction. The patch-based approach can aid the model in learning patterns and long-term dependencies more effectively, while also reducing memory requirements, as shown in Figure 6a. With the use of patches, the model could capture the local shapes of the time series by being aware of the local context instead of the point-wise value of the time series, which allows the model to better capture the local patterns and long-term fluctuations.

TABLE 4: Prediction results of different variants. Training: Validation: Test = 0.7:0.15:0.15.

	Shanghai Dataset						Nanjing Dataset					
	Mobile Traffic			Number of Connected Users			Mobile Traffic			Number of Connected Users		
Model	MAE	RMSE	R^2	MAE	RMSE	R^2	MAE	RMSE	R^2	MAE	RMSE	R^2
our model	**0.1498**	**0.2253**	**0.8534**	**12.66**	**18.3**	**0.9845**	**0.2149**	**0.3197**	**0.8548**	**141**	**324**	**0.9827**
w/o joint prediction	0.1574	0.2356	0.8525	12.21	19.01	0.9821	0.2259	0.3341	0.8423	176	356	0.9742
w/o decoder	0.1581	0.2375	0.8471	13.82	20.76	0.9789	0.2213	0.3237	0.8457	173	356	0.9792
w/o temporal	0.1976	0.2834	0.7857	18.95	34.41	0.9426	0.2798	0.3918	0.7810	220	618	0.9463
w/o spatial	0.1601	0.2390	0.8482	13.36	20.43	0.9793	0.2248	0.3278	0.8336	179	377	0.9717
w/o patch	0.1633	0.2430	0.8431	15.03	22.71	0.9745	0.2401	0.3297	0.8312	185	376	0.9696
self-attention	0.1625	0.2401	0.8469	13.85	20.89	0.9784	0.2335	0.3226	0.8362	173	368	0.9724



(a) Traffic series in Shanghai (b) User series in Shanghai

(c) Traffic series in Nanjing (d) User series in Nanjing

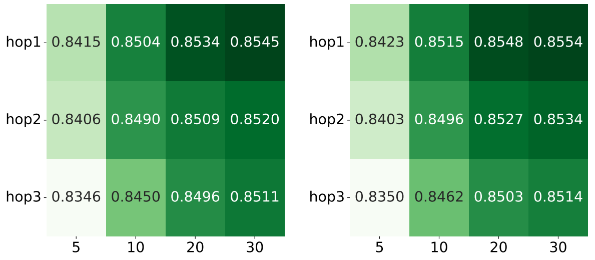
Fig. 4: Semantic Relationship Prioritization in Different Types of Base Stations.

5.4 Case Study

• **Semantic Relationship Prioritization.** To assess how different types of base stations prioritize various semantic relationships, we carried out a series of experiments. According to the distribution of POI around the base stations, we categorized the base stations into four distinct types, each of which demonstrated its own unique mobile traffic and user connection patterns. The model outputs the attention map of the fuse encoder. Besides, we also test the

Figure 4 displays the resulting attention maps, revealing how each type of base station prioritizes different semantic relationships. Our findings indicate that traffic patterns are more heavily influenced by the proximity relationship, while user patterns are more strongly affected by the pattern similarity relationship. The findings of this experiment can help researchers develop new network mapping techniques that take into account the unique characteristics and behaviours of different types of base stations. By understanding how different types of base stations prioritize semantic relationships, researchers can create more accurate and detailed maps of mobile networks.

• **Influence of the number of neighbours and hops.** We conduct a series of experiments to investigate the impact of varying the number of neighbours and neighbour hops in the subgraph. Specifically, we randomly selected different numbers of neighbours from the neighbours of different hops and evaluated the performance of our proposed model on the Shanghai and Nanjing datasets.



(a) Shanghai Datasets (b) Nanjing Datasets

Fig. 5: Influence of the number of neighbours and the hops in a subgraph.

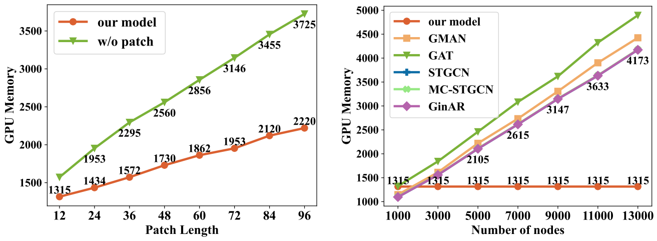
The results of these experiments are presented in Figure 5, where we report the R^2 results. We observed that when the same number of neighbours were used, the performance of one-hop neighbours was better than that of two-hop and three-hop neighbours. This suggests that the immediate neighbours of a base station contain more valuable information for prediction tasks than those further away. Furthermore, as the number of neighbours increased, the model was able to obtain more environmental information and achieved better performance. These results highlight the importance of carefully selecting the number of neighbours and neighbour hops when constructing subgraphs for our model. While including more neighbours can provide additional information, balancing this with the increased computational cost is essential.

• **Effectiveness of the spatial encoder.** In order to demonstrate the effectiveness of the spatial encoder in our proposed model and the benefits of sampling subgraphs, we compared the performance of our proposed model using subgraphs as input to the spatial encoder with the performance of GCN and RGCN using the entire base station network as input.

The results of these experiments are presented in Table 5. We observe that GCN could only input one type of base station network and was not able to effectively incorporate various environmental information, resulting in poor performance compared to our proposed model. On the other hand, RGCN could take the entire base station network as input, but this approach required significant computing power. Despite their capability to capture global features, these methods encounter significant scalability challenges

TABLE 5: Prediction results of different spatial models. Training: Validation: Test = 0.7:0.15:0.15.

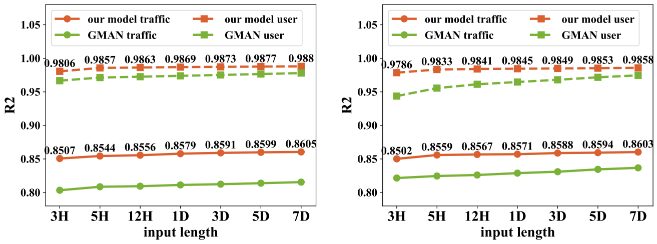
	Shanghai Dataset						Nanjing Dataset					
	Mobile Traffic			Number of Connected Users			Mobile Traffic			Number of Connected Users		
Model	**MAE**	**RMSE**	R^2	**MAE**	**RMSE**	R^2	**MAE**	**RMSE**	R^2	**MAE**	**RMSE**	R^2
our model	**0.1498**	**0.2253**	**0.8534**	**12.66**	**18.3**	**0.9845**	**0.2149**	**0.3197**	**0.8548**	**141**	**324**	**0.9827**
GCN	0.1503	0.2268	0.8521	12.83	19.54	0.9829	0.2185	0.3227	0.8521	147	356	0.9792
RGCN	0.1494	0.2229	0.8566	12.39	18.42	0.9847	0.2082	0.3037	0.8582	135	311	0.9833



(a) Comparison of GPU usage with and without patch method

(b) Comparison of GPU usage with our model and baselines

Fig. 6: GPU Usage.



(a) Shanghai Datasets

(b) Nanjing Datasets

Fig. 7: Influence of the input length.

when applied to large-scale cellular networks. In contrast, our approach of sampling subgraphs and inputting them into the spatial encoder achieved comparable performance to RGCN, while significantly reducing the amount of computing power required. These results demonstrate the effectiveness of our proposed approach for incorporating environmental information into our model using subgraphs.

Additionally, we conducted a comparison of GPU memory usage between our method and baselines. As illustrated in Figure 6b, the results reveal that the GPU memory requirement for baselines escalates with the increasing number of nodes, while the GPU memory usage of our method remains consistently stable. Our method employs a subgraph strategy that significantly reduces computational complexity. As the graph size increases, our method maintains a stable computational complexity, while other approaches experience a substantial rise in complexity. This rapid increase in other methods demonstrates the superior scalability and practicality of our approach.

• **Influence of the input length.** In order to investigate the influence of the input length, we conducted experiments by changing the input length of the historical series. And compare the performance with GMAN.

The outcomes of this experiment are depicted in Figure 7. It is observed that with the increase in input length, there is a slight improvement in the performance across all models. This trend underscores the efficacy of the multi-head cross-attention mechanism in capturing long-term dependencies. However, it is also noted that a longer input length significantly escalates the demand for computing power. Therefore, to strike a balance between model performance and computational efficiency, we have chosen to set the input length at 12.

• **Performance of Semantic Relationships between Base Stations.**

In order to demonstrate the effectiveness of different semantic relations between base stations, we conducted experiments by selecting one relationship at a time. The result of this experiment is presented in Table 6. Our findings indicate that incorporating all four semantic relationships simultaneously leads to better performance than using a single relationship, This suggests that each relation captures different aspects of the relationships between base stations, and utilizing all of them leads to a more comprehensive understanding of the behaviour of base stations in the urban environment. Of the four individual relationships, the proximity relationship performs best in traffic prediction, while the pattern similarity relationship performs best in the connected user prediction, consistent with the result of the attention map in Figure 4.

• **Transfer experiment.** To test the transferability of our model, we conduct the transfer experiment, where the model trained on the Nanjing dataset was tested on the Shanghai dataset, indicating Shanghai-T, and similarly indicating Nanjing-T. The Table 7 shows the results of the transfer experiment. According to the results, we can find that transferring between cities does not have a significant impact on the performance of our model, which proves the transferability and robustness of our model.

6 RELATED WORKS

6.1 Mobile Traffic Prediction

Mobile traffic Prediction is considered a general time series prediction task, and considerable efforts and models have been devoted to improving its performance. Hong et al. [36] employ the support vector regression (SVR), and Shu et al. [37] employ the seasonal auto-regression integrated moving average (SARIMA) into mobile traffic prediction to model the short-period mobile traffic series. However, since SARIMA and SVR rely on the average volume of past traffic series to anticipate, they fail to capture the fast change in traffic flow and they could not model the long-term and non-linear relationship. Li et al. [38] propose a software-defined cellular radio access network (SDCRAN) architec-

TABLE 6: Prediction results of different semantic relationships between base stations. Training: Validation: Test = 0.7:0.15:0.15.

	Shanghai Dataset						Nanjing Dataset					
	Mobile Traffic			Number of Connected Users			Mobile Traffic			Number of Connected Users		
Model	**MAE**	**RMSE**	R^2	**MAE**	**RMSE**	R^2	**MAE**	**RMSE**	R^2	**MAE**	**RMSE**	R^2
our model	**0.1498**	**0.2253**	**0.8534**	**12.66**	**18.3**	**0.9845**	**0.2149**	**0.3197**	**0.8548**	**141**	**324**	**0.9827**
Proximity relationship	0.1578	0.2366	0.8512	12.68	19.62	0.9808	0.2204	0.3315	0.8502	150	341	0.9794
pattern similarity	0.1585	0.2367	0.8511	12.97	19.04	0.9820	0.2213	0.3341	0.8483	148	336	0.9798
Function similarity	0.1631	0.2405	0.8464	12.75	19.20	0.9817	0.2316	0.3447	0.8428	168	355	0.9761
Flow similarity	0.1596	0.2374	0.8503	12.84	19.68	0.9801	0.2248	0.3386	0.8498	157	352	0.9778

TABLE 7: Prediction results of transfer experiment.

	Mobile Traffic			Number of Connected Users		
Dataset	**MAE**	**RMSE**	R^2	**MAE**	**RMSE**	R^2
Shanghai	**0.1498**	**0.2253**	**0.8534**	**12.66**	**18.3**	**0.9845**
Shanghai-T	0.1514	0.2347	0.8509	13.89	19.9	0.9807
Nanjing	**0.2149**	**0.3197**	**0.8548**	**141**	**324**	**0.9827**
Nanjing-T	0.201	0.3224	0.8518	150	332	0.9781

ture, and Xu et al. [39] propose a Gaussian Process (GP) method. These two methods concentrate on the single base station in cellular networks to predict short-term mobile traffic. However, due to the computing complexity of these methods being $O(n^2)$, these models could not be applied to a large-scale cellular network mobile traffic prediction task.

Owing to the flourishment of deep learning, various neural network models have been proposed for cellular traffic prediction recently. Fu et al. [40] employ Long-Short Term Memory (LSTM) [34] and Gated Regression Unit (GRU) [41] for mobile traffic prediction. However, the aforementioned models disregard geographical information in favour of solely taking into account temporal data. Besides, researchers employ the convolutional neural network (CNN) in their model to characterize spatial dependence. Zhang et al. [42] propose the STN model for precise network-wide mobile traffic prediction. Furthermore, several works also apply graph convolutional network (GCN) [11] for mobile network prediction. Fang et al. [43] use GCN to model geographic dependency, where the edges represent the spatial relationships between nodes. Feng et al. [8] propose an end-to-end model for acquiring spatially dependent and long-term cellular traffic, which utilizes a sequential module to model complex temporal changes and a broad feature extractor to model spatial relationships and encode external data. Wang et al. [7] propose an LSTM unit and a unique autoencoder-based deep model for spatial modelling, as well as spatial-temporal modelling and prediction, which were implemented in cellular networks. Wang et al. [9] present a unique breakdown of in-cell and inter-cell data flow, and apply a graph-based deep learning technique for large-scale cellular traffic prediction. Hu et al. [44] propose a spatial-temporal down-sampling neural network, which is adept at dynamically and simultaneously capturing the temporal, local, and global spatial dependencies in mobile traffic.

Overall, mobile traffic prediction has transformed from a general time series prediction task to a spatial-temporal series prediction task, and many efforts and models have been devoted to improving its performance. Recent works have focused on deep learning models, including LSTM, GRU, GNN, and GCN, which aim to capture both the temporal and spatial dependencies of mobile traffic and have been applied in various scenarios in cellular networks. However, some of these models have high computational complexity, limiting their applicability to large-scale mobile traffic prediction tasks. Besides, Mobile traffic and connected users are two critical elements of the base station, which are closely intertwined. The existing works lack the modelling of connected users in mobile cellular network prediction.

6.2 Road Traffic Prediction

Since road traffic prediction shares the same mathematical formulation as mobile traffic prediction, here we review related works. For example, Yu et al. [12] propose STGCN, merging GCN and gated CNN to model dynamic mobile traffic characteristics and capture the topological structure of the graph using the distance-based adjacency matrix. Zhao et al. [13] introduce T-GCN, a model that combines GCN and GRU to capture the topology similarity of the graph using the distance-based adjacency matrix and to model dynamic mobile traffic changes of node attribution. Guo et al. [14] propose a new attention-based spatial-temporal graph convolutional network (ASTGCN) model, which contains three independent components to model three temporal properties of mobile traffic, and the three temporal patterns are weighted fused to be the final output. Wu et al. [45] propose GraphWaveNet to model the spatial-temporal dependency, which develops a novel and learnable adaptive dependency matrix through node embedding and a stacked dilated convolution is applied to expand the receptive field. In order to predict traffic conditions for time steps in the future at various locations, Zheng et al. [26] propose a graph multi-attention network that adapts an encoder-decoder architecture, where both the encoder and the decoder consist of multiple spatial-temporal attention blocks to model the impact of the spatial-temporal factors. Diao et al. [46] propose DGCNN to track dynamic spatial dependencies by a dynamic Laplacian matrix estimator which could capture the stable global long-term spatial-temporal traffic relationships and the local traffic functions. Feng et al. [47] propose DeepSTN+, a deep learning-based convolutional model, which employs the convolution structure to model the long-range spatial dependence and a temporal attention-based fusion mechanism to capture the temporal features.

However, road traffic prediction is more similar to predicting the connected users in the mobile network since both involve a simple addition or subtraction relationship. However, the amount of traffic carried by each user is different, leading to a more complex relationship between mobile traffic and connected users.

6.3 Transformer-based Time Series Prediction Models

The success of the Transformer also motivates the development of time series prediction, and various Transformer-based models have been proposed in recent years. While the vanilla Transformer model is a popular choice for time-series prediction, it has some limitations due to its quadratic time and memory complexity caused by the self-attention mechanism.

LogTrans [48] utilizes convolutional self-attention layers with Log-Sparse design to collect the local information and lessen the space complexity. Although the LogSparse avoids the point-wise dot product of the key and query, its result is still dependent on a single time step, ignoring the time series' local structure. Informer [49] uses the ProbSparse self-attention with distillation approaches to effectively extract the most crucial keys. Autoformer [31] utilizes the concepts of decomposition and auto-correlation from conventional time series analysis techniques. The auto-correlation may link at the patch level, but because it was handmade, it does not take into account all of the semantic data included in a patch. FEDformer [50] employs a Fourier-enhanced structure to attain linear complexity, and mixture-of-experts techniques are used to combine the trend components obtained by moving average kernels of varying kernel sizes. Pyraformer [51] employs the pyramidal attention module with connections between and across scales, as well as a linear complexity. PatchTST [32], which uses channel-independence and divides time series into subseries-level patches, with each channel containing a single univariate time series that uses the same Transformer weights and embeddings as the other series. FR-Net [33] explores dynamic period features by decomposing time series into period and trend components using frequency domain rotations, employing a period frequency rotation module for predicting the period component and a patch frequency rotation module for predicting the trend component.

The above Transformer-base models are designed to excel in long-term time series prediction tasks with their ability to capture complex temporal patterns and dependencies. However, they may not be as effective in short-term prediction tasks, which require models to capture rapid changes and fluctuations in the data. Besides, these models do not fully incorporate spatial information, which is crucial in accurately predicting mobile traffic. Meanwhile, due to the above reasons, there is a lack of work using the Transformer framework in spatial-temporal sequence prediction tasks.

6.4 Multi-task Prediction

Compared with single-task prediction, multi-task prediction shares a common feature representation or parts of the model. This allows the model to learn a more generalizable and robust representation of the data, potentially improving performance across all tasks, and reducing the computational resources.

Li et al. [52] propose a multi-task graph Synchronous neural network (MTSGNN) to synchronously predict the spatial-temporal data at the regions and transitions between regions. Wang et al. [53] propose a multi-task adversarial spatial-temporal network model to predict the crowd flow and flow OD simultaneously. Wang et al. [29] propose feature correlation-aware spatiotemporal graph convolutional networks to predict the traffic flow and traffic speed. Yu et al. [30] propose GinAR which introduces interpolation attention and adaptive graph convolution to effectively model spatial-temporal dependencies and recover missing variables in limited datasets, replacing the fully connected layers of traditional recursive units for accurate multivariate time series forecasting. Currently, there are few works on multi-task prediction of mobile networks, and we are the first to jointly predict traffic and the number of connected users.

6.5 Summary and Discussion

Table 8 provides a summary of the advantages and disadvantages of the related works. Most mobile and other traffic prediction methods, such as DeepTP [8], STGCN [12], and TGCN [13], primarily focus on the proximity relationships between nodes, neglecting other semantic relationships and the use of subgraph techniques, which increases computational complexity. Transformer-based time series prediction methods, such as Autoformer [31], FR-Net [33], while effective for temporal patterns, fail to capture spatial features, resulting in suboptimal performance for mobile traffic prediction. Additionally, current multi-task prediction methods, such as MTSGNN [52], MCSGCN[29], and GinAR [30] do not account for the interdependencies between base stations, making them difficult to adapt for mobile traffic prediction scenarios.

TABLE 8: Comparison of related work with our model

Method	Patch	long-term Modeling	Proximity modeling	Semantic modeling	Graph Sampling	Joint Prediction
STN[42]	X	X	✓	X	X	X
GCLSTM [43]	X	X	✓	X	X	X
DeepTP [8]	X	X	✓	X	X	X
GSAE [7]	X	X	✓	X	X	X
GNN-D [9]	X	X	✓	X	X	X
STD-Net [44]	X	X	✓	X	✓	X
STGCN [12]	X	X	✓	X	X	X
TGCN [13]	X	X	✓	X	X	X
ASTGCN [14]	X	✓	✓	X	X	X
GMAN [26]	X	X	✓	✓	✓	X
DeepSTN+ [47]	X	✓	✓	X	X	X
PatchTST [32]	✓	✓	X	X	X	X
Autoformer [31]	✓	✓	X	X	X	X
FR-Net [33]	✓	✓	X	X	X	X
MTSGNN [52]	X	X	✓	X	X	✓
MCSGCN [29]	X	X	✓	X	X	✓
GinAR [30]	X	X	✓	X	X	✓
Our	✓	✓	✓	✓	✓	✓

7 CONCLUSION

Our study aimed to investigate the relationship between mobile traffic and the Number of Connected Users within a base station network and develop a predictive model for large-scale prediction. We introduced the spatiotemporal transformer framework, a novel model that leverages Cross-Attention and hierarchical spatial attention mechanisms to capture the complex interactions between the two variables. The Cross-Attention mechanism captures the interdependence between mobile traffic and connected users, while the hierarchical spatial attention mechanism identifies and uses the most relevant information from various semantic relationships. Our subgraph-picking method enables us to apply our model to real-world applications without excessive computing power. This approach has significant implications for the practical implementation of our model, allowing us to scale up to larger networks while maintaining high prediction accuracy.

In our future work, we plan to extend our proposed approach to evaluate its effectiveness in diverse settings by applying it to other cities and regions. This will help us assess the generalizability of our model and identify any limitations or challenges in applying it to various mobile network environments. Moreover, we aim to investigate the transferability of our model to different types of mobile networks, such as 5G networks, which have unique characteristics and requirements. We will explore how our model can be adapted to these networks and applied in network optimization and management. Finally, we will continue collaborating with industry partners to integrate our model into existing mobile network management systems and improve network performance and user experience.

Acknowledgment

This research has been supported in part by the National Natural Science Foundation of China under Grant U22B2057 and 62471277, in part by the National Key Research and Development Program of China under Grant 2023YFB2904804, and in part by the Institute for Guo Qiang, Tsinghua University.

References

[1] Ericsson, "Ericsson mobility report," Ericsson, Tech. Rep., 11 2022, technical Report. [Online]. Available: https://www.ericsson.com/en/reports-and-papers/mobility-report

[2] X. Wang, J. Zhao, L. Zhu, X. Zhou, Z. Li, J. Feng, C. Deng, and Y. Zhang, "Adaptive multi-receptive field spatial-temporal graph convolutional network for traffic forecasting," in *2021 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2021, pp. 1–7.

[3] F. Sun, P. Wang, J. Zhao, N. Xu, J. Zeng, J. Tao, K. Song, C. Deng, J. C. Lui, and X. Guan, "Mobile data traffic prediction by exploiting time-evolving user mobility patterns," *IEEE Transactions on Mobile Computing*, vol. 21, no. 12, pp. 4456–4470, 2022.

[4] X. Wang, K. Yang, Z. Wang, J. Feng, L. Zhu, J. Zhao, and C. Deng, "Adaptive hybrid spatial-temporal graph neural network for cellular traffic prediction," *arXiv preprint arXiv:2303.00498*, 2023.

[5] F. Yang, Y. Jiang, T. Pan, and X. E., "Traffic anomaly detection and prediction based on sdn-enabled icn," in *2018 IEEE International Conference on Communications Workshops (ICC Workshops)*, 2018, pp. 1–5.

[6] Y. Sun, M. Peng, Y. Zhou, Y. Huang, and S. Mao, "Application of machine learning in wireless networks: Key techniques and open issues," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3072–3108, 2019.

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

[8] J. Feng, X. Chen, R. Gao, M. Zeng, and Y. Li, "Deeptp: An end-to-end neural network for mobile cellular traffic prediction," *IEEE Network*, vol. 32, no. 6, pp. 108–115, 2018.

[9] X. Wang, Z. Zhou, Z. Yang, Y. Liu, and C. Peng, "Spatio-temporal analysis and prediction of cellular traffic in metropolis," in *2017 IEEE 25th International Conference on Network Protocols (ICNP)*, 2017, pp. 1–10.

[10] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 61–80, 2009.

[11] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[12] T. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," in *International Joint Conference on Artificial Intelligence*, 2017.

[13] L. Zhao, Y. Song, C. Zhang, Y. Liu, P. Wang, T. Lin, M. Deng, and H. Li, "T-gcn: A temporal graph convolutional network for traffic prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 9, pp. 3848–3858, 2020.

[14] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan, "Attention based spatial-temporal graph convolutional networks for traffic flow forecasting," in *AAAI Conference on Artificial Intelligence*, 2019.

[15] Y. Jin, N. Duffield, J. Erman, P. Haffner, S. Sen, and Z.-L. Zhang, "A modular machine learning system for flow-level traffic classification in large networks," *ACM Trans. Knowl. Discov. Data*, vol. 6, no. 1, mar 2012. [Online]. Available: https://doi.org/10.1145/2133360.2133364

[16] C. Chen, K. Li, S. G. Teo, X. Zou, K. Li, and Z. Zeng, "Citywide traffic flow prediction based on multiple gated spatio-temporal convolutional neural networks," *ACM Trans. Knowl. Discov. Data*, vol. 14, no. 4, may 2020. [Online]. Available: https://doi.org/10.1145/3385414

[17] F. A. Silva, A. C. S. A. Domingues, and T. R. M. B. Silva, "Discovering mobile application usage patterns from a large-scale dataset," *ACM Trans. Knowl. Discov. Data*, vol. 12, no. 5, jun 2018. [Online]. Available: https://doi.org/10.1145/3209669

[18] S. Wang, J. Zhang, J. Li, H. Miao, and J. Cao, "Traffic accident risk prediction via multi-view multi-task spatio-temporal networks," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–1, 2021.

[19] H. Li, D. Jin, X. Li, J. Huang, and J. Yoo, "Multi-task synchronous graph neural networks for traffic spatial-temporal prediction," in *Proceedings of the 29th International Conference on Advances in Geographic Information Systems*, ser. SIGSPATIAL '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 137–140. [Online]. Available: https://doi.org/10.1145/3474717.3483921

[20] F. Itakura, "Minimum prediction residual principle applied to speech recognition," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 23, no. 1, pp. 67–72, 1975.

[21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[22] H. Wang, F. Xu, Y. Li, P. Zhang, and D. Jin, "Understanding mobile traffic patterns of large scale cellular towers in urban environment," in *Proceedings of the 2015 Internet Measurement Conference*, 2015, pp. 225–238.

[23] J. Gong, T. Li, H. Wang, Y. Liu, X. Wang, Z. Wang, C. Deng, J. Feng, D. Jin, and Y. Li, "Kgda: A knowledge graph driven decomposition approach for cellular traffic prediction," *ACM Transactions on Intelligent Systems and Technology*, vol. 15, no. 6, pp. 1–22, 2024.

[24] H. Drucker, C. Burges, L. Kaufman, A. Smola, and V. Vapnik, "Support vector regression machines," *Adv Neural Inform Process Syst*, vol. 28, pp. 779–784, 01 1997.

[25] B. Williams and L. Hoel, "Modeling and forecasting vehicular traffic flow as a seasonal arima process: Theoretical basis and empirical results," *Journal of Transportation Engineering*, vol. 129, pp. 664–672, 11 2003.

[26] C. Zheng, X. Fan, C. Wang, and J. Qi, "Gman: A graph multi-attention network for traffic prediction," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 01, pp. 1234–1241, Apr. 2020. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/5477

[27] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio', and Y. Bengio, "Graph attention networks," *ArXiv*, vol. abs/1710.10903, 2017.

[28] W. Hamilton, R. Ying, and J. Leskovec, "Inductive representation learning on large graphs," 06 2017.

[29] S. Wang, M. Zhang, H. Miao, Z. Peng, and P. S. Yu, "Multivariate correlation-aware spatio-temporal graph convolutional networks for multi-scale traffic prediction," vol. 13, no. 3, jan 2022. [Online]. Available: https://doi.org/10.1145/3469087

[30] C. Yu, F. Wang, Z. Shao, T. Qian, Z. Zhang, W. Wei, and Y. Xu, "Ginar: An end-to-end multivariate time series forecasting model suitable for variable missing," in *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, ser. KDD '24. New York, NY, USA: Association for Computing Machinery, 2024, p. 3989–4000. [Online]. Available: https://doi.org/10.1145/3637528.3672055

[31] H. Wu, J. Xu, J. Wang, and M. Long, "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting," *Advances in Neural Information Processing Systems*, vol. 34, pp. 22 419–22 430, 2021.

[32] Y. Nie, N. H. Nguyen, P. Sinthong, and J. Kalagnanam, "A time series is worth 64 words: Long-term forecasting with transformers," *arXiv preprint arXiv:2211.14730*, 2022.

[33] X. Zhang, S. Feng, J. Ma, H. Lin, X. Li, Y. Ye, F. Li, and Y. S. Ong, "Frnet: Frequency-based rotation network for long-term time series forecasting," in *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, ser. KDD '24. New York, NY, USA: Association for Computing Machinery, 2024, p. 3586–3597. [Online]. Available: https://doi.org/10.1145/3637528.3671713

[34] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, pp. 1735–80, 12 1997.

[35] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014.

[36] W. Hong, "Application of seasonal svr with chaotic immune algorithm in traffic flow forecasting," *Neural Computing and Applications*, vol. 21, pp. 583 – 593, 2010.

[37] Y. Shu, M. Yu, J. Liu, and O. Yang, "Wireless traffic modeling and prediction using seasonal arima models," in *IEEE International Conference on Communications, 2003. ICC '03.*, vol. 3, 2003, pp. 1675–1679 vol.3.

[38] R. Li, Z. Zhao, X. Zhou, J. Palicot, and H. Zhang, "The prediction analysis of cellular radio access network traffic: From entropy theory to networking practice," *IEEE Communications Magazine*, vol. 52, no. 6, pp. 234–240, 2014.

[39] Y. Xu, W. Xu, F. Yin, J. Lin, and S. Cui, "High-accuracy wireless traffic prediction: A gp-based machine learning approach," in *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, 2017, pp. 1–6.

[40] R. Fu, Z. Zhang, and L. Li, "Using lstm and gru neural network methods for traffic flow prediction," in *2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, 2016, pp. 324–328.

[41] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint arXiv:1409.1259*, 2014.

[42] C. Zhang and P. Patras, "Long-term mobile traffic forecasting using deep spatio-temporal neural networks," in *Proceedings of the Eighteenth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2018, pp. 231–240.

[43] L. Fang, X. Cheng, H. Wang, and L. Yang, "Mobile demand forecasting via deep graph-sequence spatiotemporal modeling in cellular networks," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 3091–3101, 2018.

[44] Y. Hu, Y. Zhou, J. Song, L. Xu, and X. Zhou, "Citywide mobile traffic forecasting using spatial-temporal downsampling transformer neural networks," *IEEE Transactions on Network and Service Management*, vol. 20, no. 1, pp. 152–165, 2023.

[45] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang, "Graph wavenet for deep spatial-temporal graph modeling," in *International Joint Conference on Artificial Intelligence*, 2019.

[46] Z. Diao, G. Wang, D. Zhang, Y. Liu, K. Xie, and S. He, "Dynamic spatial-temporal graph convolutional neural networks for traffic forecasting," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 890–897, 07 2019.

[47] J. Feng, Y. Li, Z. Lin, C. Rong, F. Sun, D. Guo, and D. Jin, "Context-aware spatial-temporal neural network for citywide crowd flow prediction via modeling long-range spatial dependency," vol. 16, no. 3, oct 2021. [Online]. Available: https://doi.org/10.1145/3477577

[48] S. Li, X. Jin, Y. Xuan, X. Zhou, W. Chen, Y.-X. Wang, and X. Yan, "Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting," *Advances in neural information processing systems*, vol. 32, 2019.

[49] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang, "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 12, 2021, pp. 11 106–11 115.

[50] T. Zhou, Z. Ma, Q. Wen, X. Wang, L. Sun, and R. Jin, "Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting," in *International Conference on Machine Learning*. PMLR, 2022, pp. 27 268–27 286.

[51] S. Liu, H. Yu, C. Liao, J. Li, W. Lin, A. X. Liu, and S. Dustdar, "Pyraformer: Low-complexity pyramidal attention for long-range time series modeling and forecasting," in *International conference on learning representations*, 2021.

[52] H. Li, D. Jin, X. Li, J. Huang, and J. Yoo, "Multi-task synchronous graph neural networks for traffic spatial-temporal prediction," ser. SIGSPATIAL '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 137–140. [Online]. Available: https://doi.org/10.1145/3474717.3483921

[53] S. Wang, H. Miao, H. Chen, and Z. Huang, "Multi-task adversarial spatial-temporal networks for crowd flow prediction," in *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, ser. CIKM '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 1555–1564. [Online]. Available: https://doi.org/10.1145/3340531.3412054