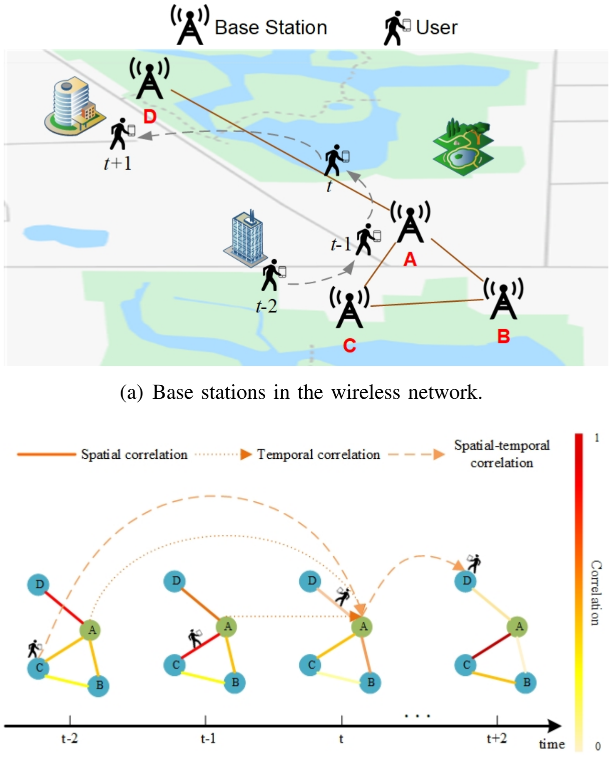
GLOBECOM 2021 - 2021 IEEE Global Communications Conference | 978-1-7281-8104-2/21/$31.00 ©2021 IEEE | DOI: 10.1109/GLOBECOM46510.2021.9685054

Adaptive Multi-receptive Field Spatial-Temporal Graph Convolutional Network for Traffic Forecasting

Xing Wang[1], Juan Zhao[1], Lin Zhu[1], Xu Zhou[2], Zhao Li[2], Junlan Feng[1], Chao Deng[1], Yong Zhang[2]
[1] China Mobile Research Institute, Beijing, China
[2] Electronic Engineering, Beijing University of Posts and Telecommunications, Beijing, China
[1]{wangxing, zhaojuan, zhulinyj, fengjunlan, dengchao}@chinamobile.com, [2]{zhou_xu, L_zhao, yongzhang}@bupt.edu.cn

***Abstract*—Mobile network traffic forecasting is one of the key functions in daily network operation. A commercial mobile network is large, heterogeneous, complex and dynamic. These intrinsic features make mobile network traffic forecasting far from being solved even with recent advanced algorithms such as graph convolutional network-based prediction approaches and various attention mechanisms, which have been proved successful in vehicle traffic forecasting. In this paper, we cast the problem as a spatial-temporal sequence prediction task. We propose a novel deep learning network architecture, Adaptive Multi-receptive Field Spatial-Temporal Graph Convolutional Networks (AMF-STGCN), to model the traffic dynamics of mobile base stations. AMF-STGCN extends GCN by (1) jointly modeling the complex spatial-temporal dependencies in mobile networks, (2) applying attention mechanisms to capture various Receptive Fields of heterogeneous base stations, and (3) introducing an extra decoder based on a fully connected deep network to conquer the error propagation challenge with multi-step forecasting. Experiments on four real-world datasets from two different domains consistently show AMF-STGCN outperforms the state-of-the-art methods.**

***Index Terms*—traffic forecasting, spatial-temporal data, Graph Convolutional Network, mobile traffic, deep learning**



(a) Base stations in the wireless network.

(b) Complex spatial and temporal correlations in traffic network.

Fig. 1. Spatial-temporal correlations in the mobile traffic network.

I. Introduction

Total global mobile data traffic reached 58EB per month at the end of 2020. It is projected to grow monthly by 5 percent in coming years. Data traffic prediction is becoming of fundamental importance for 4G/5G telecom network operation. Accurate traffic forecast enables key parameters of base stations to be automatically adapted for optimal user experience and energy consumption. With large-scale commercial deployment of 5G network, traffic prediction has become one of the key enabling technologies for autonomous network, which is promoted by the entire telecom industry. In addition, traffic forecasting provides important support for many transportation services, such as traffic control, route planning, and navigation, etc [1].

A typical telecom operator operates millions of base stations. Each base station connects with users within a certain radius range and transfers data for them. The generated data traffic shares similar patterns with the road traffic data. First, traffic data of mobile network or road network is inherent with complex spatial-temporal dependencies. Fig. 1(a) shows the user migration between base stations, resulting in explicit or implicit spatial-temporal correlations among the base stations. Due to the migration of users, the data traffic of a base station A at time step t, represented as A_t, is influenced by B_t, C_t and D_t in the spatial domain, and the impacts change over time as the bold lines shown in Fig. 1(b). Meanwhile, A_t correlates with its own past traffic $A_{0,\ldots,t-1}$ in the time axis. In addition, the information can propagate along the spatial and temporal dimensions simultaneously. And the propagation process can be discontinuous due to the uncertainty of wireless

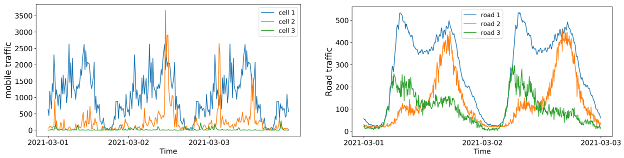
978-1-7281-8104-2/21/$31.00 ©2021 IEEE

signal, user behavior, environment change, etc. For example, as shown in Fig. 1(b), a user can consume a large volume of data via a given base station C at time step $t-2$. At the next moment, this user can stop the connection or migrates to a different base station A without consuming any data traffic until time step t. Then the traffic volume of C_{t-2} actually affects the traffic volume of A_t directly. The discontinuous nature of spatial-temporal data generation results in the inherent non-local spatial-temporal synchronization dependencies along with the current and history moments. And we believe that if we capture the complex spatial-temporal synchronization correlations directly, the traffic prediction performance can be improved significantly.

In this paper, we frame the problem as a Spatial-Temporal series prediction task, where both input and output are spatial-temporal data series. Base stations form a graph in space based on distances between them and business reliances, where each node corresponds to a base station and each edge represents their relationship.

Traditional time series modeling algorithms such as the autoregressive integrated moving average (ARIMA) [2] and its variants are hard to be extended to represent spatial correlations. Researchers have proposed various deep learning models to capture the complex reliances hidden in traffic data of the transportation domain, which has similarity to mobile traffic, though mobile traffic data are relatively more dynamic and unstable. Most of these models use a few building blocks to respectively capture spatial dependencies, temporal dependencies and then fuse embeddings from both spatial and temporal domains. Graph convolutional networks (GCN) is one of the most consolidated approaches to capture the spatial correlations among nodes, while CNN, recurrent neural networks including Gate Recurrent Units (GRU) and long-short term memory (LSTM) have been used to model temporal dynamics [1] [3]. [4] introduced attention mechanisms including spatial attention and temporal attention to enhance GCN and GNN to better model the nonlinearities. These models have advanced the technology and shown significant improvement on prediction performance.

However, two remaining challenges are not addressed fairly. One, most aforementioned models capture spatial correlations and temporal correlations separately, and then fuse them together. We argue the complex spatial-temporal correlations should be synchronously modeled. STSGCN [7] used multiple local spatial-temporal graphs to model the spatial-temporal synchronous correlations of the local adjacent time steps. But STSGCN only focused on the localized spatial-temporal correlations without considering the discontinuous nature of spatial-temporal data generation. Hence, in this paper we propose a novel spatial-temporal joint graph convolutional network to directly capture the intrinsic non-local spatial-temporal reliances.



(a) Mobile traffic data.

(b) Road traffic data.

Fig. 2. The heterogeneity of traffic data

Second, heterogeneity of nodes is not well considered. Especially in 5G, a base station could be a wide-area station, a medium range station, or a local area station. Their reliances with nearby stations are quite different. Heterogeneity is also an evident concern to model traffic network. As shown in Fig. 2, the traffic data sequence of different nodes can behave quite variously in both the mobile network and road network. The traffic in an urban area is denser, which means the roads of such regions can be affected by a wider range of neighborhoods. On the contrary, the traffic of suburban area is relatively sparse. To address this challenge, we propose an Adaptive Multi-receptive Field Spatial-Temporal Graph Convolutional Block (AMF-STConv Block), which contains the multi-scale spatial-temporal feature extraction and a graph attention module to adaptively model the graph. Following the multiple AMF-STConv Blocks, we design a fusion module with two decoders named Fully-Connected decoder and Time-Step decoder along with the pre-training mechanism to establish the relationships between the historical and future time steps to alleviate multi-step error accumulation.

In this paper, we propose a novel Adaptive Multi-receptive Field Spatial-Temporal Graph Convolutional Networks (AMF-STGCN) to address the above mentioned issues of traffic forecasting. To the best of our knowledge, we are the first to apply GCN to mobile network traffic prediction. Our main technical contributions are summarized as follows:

- We propose a novel deep learning network architecture, Adaptive Multi-receptive Field Spatial-Temporal Graph Convolutional Networks (AMF- STGCN) to model mobile network traffic of base stations.
- Our proposed model, AMF-STGCN, is able to effectively model the complex spatial-temporal dependencies in mobile networks, capture the heterogeneity of base stations adaptively with an attention mechanism, and tackle the error propagation challenge with multi-step forecasting.
- We evaluate AMF-STGCN on four real-world datasets from two fields and the experimental results prove AMF-STGCN achieves the best overall prediction performance comparing to current state-of-the-art prediction models.

II. Methodology

A. Preliminaries

In this paper, we define a static undirected graph $\mathcal{G} = (V, E, A)$ as the traffic network. V represents the set of nodes, $|V| = N(N$ indicates the number of nodes). E is the set of edges representing the connectivity between nodes. $A \in \mathbb{R}^{N \times N}$ is the adjacency matrix of $\mathcal{G}$, where $A_{v_i,v_j} \in \{0, 1\}$ represents the connection between nodes v_i and v_j. The graph signal matrix is expressed as $X_t \in \mathbb{R}^{N \times C}$, where t denotes the timestep and C indicates the feature dimension. The graph signal matrix represents the observations of graph network $\mathcal{G}$ at time step t.

Problem Defined

Given the graph signal matrix of historical T time steps $X = (X_{t_1}, X_{t_2}, \ldots, X_{t_T}) \in \mathbb{R}^{T \times N \times C}$, our goal is to predict the graph signal matrix of the next M time steps $\hat{Y} = \left(\hat{X}_{t_{T+1}}, \hat{X}_{t_{T+2}}, ..., \hat{X}_{t_{T+M}}\right) \in \mathbb{R}^{M \times N \times C}$. Given an undirected graph $\mathcal{G}$, we need to learn a mapping function $\mathcal{F}$ to map the graph signal matrix of historical time steps to the future time steps:

$$\left(\hat{X}_{t_{T+1}}, \hat{X}_{t_{T+2}}, \cdots, \hat{X}_{t_{T+M}}\right) = \mathcal{F}_\theta \left(X_{t_1}, X_{t_2}, \cdots, X_{t_T}; \mathcal{G}\right) \tag{1}$$

where θ represents learnable parameters of our model.

B. Architecture

The architecture of the AMF-STGCN is shown in Fig. 3(a), which consists of multiple Adaptive Multi-receptive Field Spatial-Temporal Graph Convolutional Blocks (AMF-STConv block), a Fully-Connected decoder, a Time-Step decoder, and a Fusion module. Each AMF-STConv block is composed of the proposed spatial-temporal joint graph convolution with adaptive receptive field graph attention mechanism, to adaptively model the graph heterogeneity. We also incorporate two decoders and the pre-training mechanism into the model to achieve accurate multi-step traffic forecast.

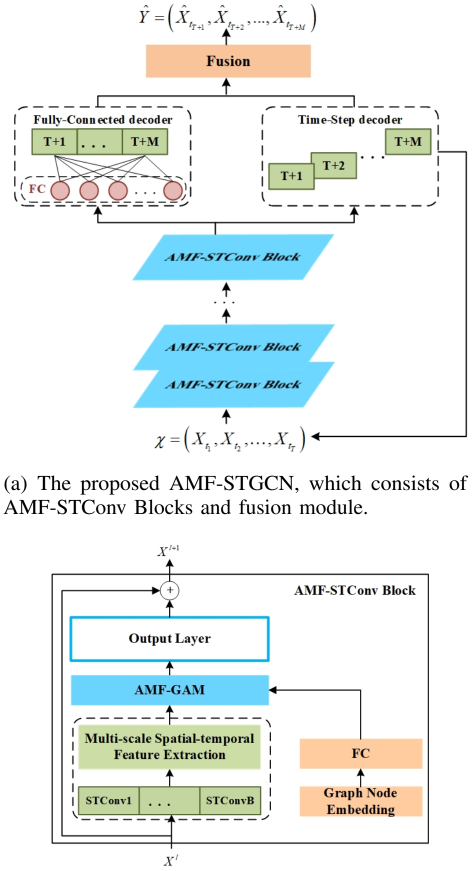
C. Adaptive Multi-Receptive Field Spatial-Temporal Graph Convolutional Block

The AMF-STConv block, as shown in Fig. 3(b), first implements multiple spatial-temporal joint convolution kernels, such as B branches (kernels), to achieve multi-receptive field spatial-temporal features extraction. Then the extracted features go through an Adaptive Multi-receptive Field Graph attention module (AMF-GAM), which combine graph node embedding with the attention to achieve the heterogeneity modeling. At last, we use a two-layer fully connected network (the output layer) to produce the output. We denote the input of the l^{th} AMF-STConv block as X^l, and the output as X^{l+1}.

Spatial-temporal joint graph convolution

Previous models captured spatial and temporal features with separate modules respectively, which ignored the synchronous correlation [1], [3]–[6]. In order to extract spatial-temporal correlations simultaneously, we propose spatial-temporal joint Graph Convolution (STConv). In this paper, we construct STConv based on graph convolution in the spectral domain, which is implemented by using the graph Fourier transform basis from eigenvalue decomposition of the Laplacian matrix. Then it transforms the graph signals from spatial into the spectral domain. To reduce the computation complexity, Chebyshev polynomial $T_k(x)$ is used for approximation. The spectral graph convolution can be written as [1], [9]:

$$\Theta *_{\mathcal{G}} x = \Theta (L) x \approx \sum_{k=0}^{K-1} \theta_k T_k \left(\tilde{L}\right) x \tag{2}$$


$\hat{Y} = \left(\hat{X}_{t_{T+1}}, \hat{X}_{t_{T+2}}, ..., \hat{X}_{t_{T+M}}\right)$
Fusion
Fully-Connected decoder
T+1
. . .
T+M
FC
. . .
Time-Step decoder
T+M
T+2
T+1
. . .
AMF-STConv Block
. . .
AMF-STConv Block
AMF-STConv Block
$\chi = \left(X_{t_1}, X_{t_2}, ..., X_{t_T}\right)$

(a) The proposed AMF-STGCN, which consists of AMF-STConv Blocks and fusion module.

(b) The architecture of AMF-STConv Block.

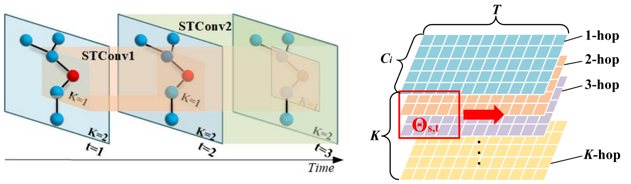
Fig. 3. Overall architecture of AMF-STGCN.

where $*_{\mathcal{G}}$ is graph convolutional operator, Θ is graph convolution kernel, $x \in \mathbb{R}^N$ is the graph signal, $T_k\left(\tilde{L}\right) \in \mathbb{R}^{N \times N}$ is the Chebyshev polynomial of order k with the scaled Laplacian $\tilde{L} = \frac{2}{\lambda_{max}} L - I_N$ (L is the normalized graph Laplacian, λ_{max} is the largest eigenvalue of L, I_N is identity matrix). θ_k is the coefficient of the k-th order polynomial.

Based on spectral domain graph convolution, as shown in Fig. 4 (b), we concatenate the K-hop $T_k\left(\tilde{L}\right)$ to form the graph convolution result with multi-scale spatial receptive fields. K indicates the furthest receptive field in the spatial dimension. Different from [17] which only calculated the $T_K\left(\tilde{L}\right)$ graph convolution without the concatenation, all nodes in the graph share the same Kth-hop only spatial receptive field without multi-scale. Then we construct the spatial-temporal joint graph convolution kernel $\Theta_{s,t} \in \mathbb{R}^{K_t \times K_s \times C_i \times C_o}$, where K_t and K_s represents the kernel size in the temporal and spatial dimension respectively, C_i is the number of input channels, C_o is the number of filters. And the spatial-temporal joint graph convolution can be formulated as:

$$\mathbf{T_K}\left(\tilde{L}\right) = Concat(T_0\left(\tilde{L}\right), T_1\left(\tilde{L}\right), \ldots, T_{K-1}\left(\tilde{L}\right)) \tag{3}$$

$$\Theta_{s,t} * X^l = \Theta_{s,t} * \mathbf{T_K}\left(\tilde{L}\right) X^l \tag{4}$$



(a) Two STConv kernel examples.

(b) A schematic diagram of STConv kernel.

Fig. 4. Spatial-temporal joint graph convolution.

where $\mathbf{T_K}\left(\tilde{L}\right) \in \mathbb{R}^{K\times N\times N}$ is the concatenation of all Chebyshev polynomials in (K-1) hop, $*$ is the convolution operation between $\Theta_{s,t}$ and X^l, $X^l \in \mathbb{R}^{N\times T\times C_i}$ is the input of the l_{th} AMF-STConv block, T is the input time steps. The concatenated graph convolution result can be $\mathbf{T_K}\left(\tilde{L}\right) X^l \in \mathbb{R}^{N\times T\times K\times C_i}$. After the spatial-temporal joint graph convolution without padding, the output can be written as $F \in \mathbb{R}^{N\times(T-K_t+1)\times(K-K_s+1)\times C_o}$.

So the STConv kernel $\Theta_{s,t}$ has the local spatial-temporal receptive field of $K_t \times K_s$, and K_s should be smaller than K because of the defined largest graph convolution of K-hop. For example, as the STConv1, STConv2 shown in Fig. 4(a), STConv1 represents the STConv kernel with size of 3×2 ($K_t \times K_s$), which can be represented as $\theta_{s,t} \in \mathbb{R}^{3\times 2\times C_i\times C_o}$. This means it can extract the spatial-temporal features of the node itself and its 2-hop spatial neighbors in the three adjacent time steps by one layer, which is needed by STSGCM with above two layers in STSGCN [7]. And deep Graph Convolutional Network (GCN) is prone to cause the over smoothing problem.

Besides, since the spatial neighbors have various influences on the central node, so we implement a learnable spatial mask matrix $W_{mask} \in \mathbb{R}^{N\times N}$ [7] to adjust the graph adjacency relationship for assigning weights to different neighbors. Like [7], we do the element-wise product between adjacency matrix A and W_{mask} to build a weight adjusted adjacency matrix: $A' = W_{mask} \otimes A \in \mathbb{R}^{N\times N}$. A' is used to compute all graph convolutions in AMF-STGCN.

Adaptive Multi-Receptive Field Graph Attention Module

Different from the images whose data distribution is consistent, each node of the graph usually represents a road segment or a base station etc. Affected by external factors such as geographic location and surrounding environment, the traffic data of the graph nodes are various, namely the heterogeneity, as shown in Fig. 2. To address this problem, an intuitive method is to learn the individual model for each node, but this method could cause extensive parameters and is hard to generalize. As a result, we aim to build a unified model to achieve traffic forecasting with heterogeneity modeling. Due to the nodes' various properties and local spatial structures, we understand that a key manifestation of heterogeneity is the different local spatial-temporal receptive fields of each graph node. This means that different graph nodes have different perceptions of multiple spatial-temporal receptive fields. Inspired by [6], [8], we apply a learnable graph node embedding $S_e \in \mathbb{R}^{N\times d}$ to represent the properties of each node in high-dimensional space, where d represents the node embedding dimension. Meanwhile, we introduce inception [10] to build multi-scale spatial-temporal receptive fields. Then, the graph node-level attention combined with multi-scale spatial-temporal receptive fields is proposed to adaptively model the graph heterogeneity.

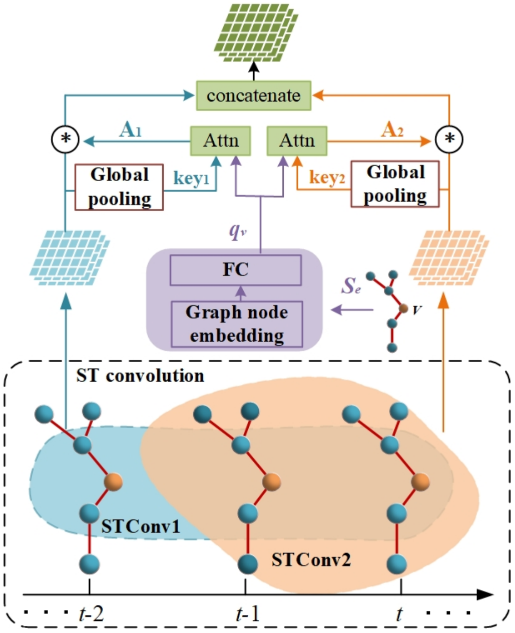


Fig. 5. Adaptive Multi-Receptive Field Graph Attention Module.

The entire architecture of Adaptive Multi-receptive Field Graph Attention Module (AMF-GAM) is shown in Fig. 5 as an example. Firstly, we implement spatial-temporal joint graph convolution for the input $X^l \in \mathbb{R}^{N\times T\times C_i}$ with B STConv kernels to extract multiple spatial-temporal features. Specifically the inception is implemented with padding, so the output of each branch b can be $F_b \in \mathbb{R}^{N\times T\times K\times C_o}$. Then extracted features of B branches are concatenated to form the output $B_{out} \in \mathbb{R}^{N\times T\times K\times (C_o\times B)}$, where we set the number of output channels C_o of each branch to be the same.

Here we introduce the graph node attention, which has being widely used [11]. To solve the heterogeneity, we use the $Q = S_e W_q$ as graph node query, where $Q \in \mathbb{R}^{N\times C_o}$, $W_q \in \mathbb{R}^{d\times C_o}$. For each branch output F_b, we apply global pooling [16] $Key_b = \sum\limits_{i=1}^{T}\sum\limits_{j=1}^{K} F_b$, where $Key_b \in \mathbb{R}^{N\times C_o}$ as the corresponding key of branch b, then $Key \in \mathbb{R}^{N\times(C_o\times B)}$ for B branches (one can also use $W_k \in \mathbb{R}^{C_o\times C_o}$ to do transform, here we omit it for simplicity). Then we compute the attention score by $A = QKey^T$, where $A \in \mathbb{R}^{N\times B}$. Take a graph node v_i as an example, $A_{b,v_i} = \frac{Q_{v_i}\cdot Key_{b,v_i}}{\sqrt{C_o}}$, where $Q_{v_i} \in \mathbb{R}^{1\times C_o}$, $Key_{b,v_i} \in \mathbb{R}^{1\times C_o}$, denotes the attention of v_i with each branch extracted spatial-temporal features. Then we concatenate results of each branch F_b adjusted by attention score As to obtain the final AMF-GAM output Ao. The calculation can be formulated as follows:

$$As_{v_i,b} = \frac{\exp(A_{b,v_i})}{\sum\limits_{r=1}^{B}\exp(A_{r,v_i})} \tag{5}$$

$$Ao_{v_i} = ||_{b=1}^{B}\left\{A_{b,v_i} \cdot F_{b,v_i}\right\} \tag{6}$$

where $A_{b,v_i} \in \mathbb{R}$, $F_{b,v_i} \in \mathbb{R}^{T \times K \times C_o}$, $Ao_{v_i} \in \mathbb{R}^{T \times K \times C_o \times B}$ represent the output with node v_i of the AMF-GAM. Therefore, the final output of the AMF-GAM is $Ao \in \mathbb{R}^{N \times T \times K \times C_o \times B}$.

AMF-STConv block output Layer

Then, we use a two-layer fully connected neural network to generate the output of AMF-STConv block. We first reshape Ao into $Ao \in \mathbb{R}^{N \times T \times K \times (C_o \times B)}$. Next, the learnable weight matrix $W_s \in \mathbb{R}^{K \times (C_o \times B) \times (C_o \times B)}$ is used to convert Ao to $Ao \in \mathbb{R}^{N \times T \times (C_o \times B)}$. We also implement SE-net [12] to model the channel attention. Finally, the output is converted to $X^{l+1} \in \mathbb{R}^{N \times T \times C_o}$ using the second full connection layer with $W_o \in \mathbb{R}^{(C_o \times B) \times C_o}$. And the process can be formulated as $X^{l+1} = AoW_sW_o$. In addition, we implement residual networks [13] and layer normalization to improve the network performance.

D. Fusion Output Module

After pre-trained in the AMF-STConv block, we propose two decoder modules to fuse features: Time-Step decoder and Fully-Connected decoder, as shown in Fig. 3. In Time-Step decoder, each forecasted time step is concatenated with the inputs along with previous observations to forecast next step, which called iterative forecasting. However, in multi-step traffic forecasting, iterative forecasting is prone to error accumulation, which leads to the gradual deterioration of the long-term forecasting accuracy. Fully-Connected decoder uses two fully-connected layers to forecast multi-step results at the same time. But the short-term forecasting performance of Fully-Connected decoder is less good than Time-Step decoder. To benefit from both, we fuse the outputs of Time-Step decoder and Fully-Connected decoder to obtain better forecasting performance. At last, we adopt the fusion of two decoders as:

$$\hat{Y} = W_f \odot \hat{Y}_{fc\_out} + (1 - W_f)\hat{Y}_{ts\_out} \tag{7}$$

where $\hat{Y}_{ts\_out}, \hat{Y}_{fc\_out} \in \mathbb{R}^{N \times M \times C}$ represent the forecasting of Time-Step decoder and Fully-Connected decoder respectively. W_f is learning parameters, $\hat{Y}$ is the final multi-step prediction results.

In this paper, we use the mean square error (MSE) as the loss function and minimize it through backpropagation.

$$\mathrm{L}(\theta) = \frac{1}{M}\sum_{i=t+1}^{t+M}(Y_i - \hat{Y}_i)^2 \tag{8}$$

where θ represents all learnable parameters of our model, $\hat{Y}_i$ denotes the model's forecasting results of all nodes at time step i, Y_i is the ground truth.

III. Experiment

A. Datasets

We evaluate AMF-STGCN on two mobile traffic datasets (Milan, Jiangsu) and two road traffic datasets (PEMS04, PEMS08). Milan dataset comes from the mobile traffic volume records of Milan provided by Telecom Italia [18]. Jiangsu dataset is mobile traffic volume sampled from 1051 cells in Jiangsu province. The road traffic datasets come from the Caltrans Performance Measurement System (PeMS) [14]. We summarize the statistics of the datasets in Table I.

TABLE I
Dataset description

Datasets	Samples	Nodes	Timespan	Timeslot	Input Length	Output Length
Milan	4320	900	Nov, 2013	10min	6	6
Jiangsu	8640	1051	Jan-Mar, 2021	15min	12	12
PEMS04	16992	307	Jan-Feb, 2018	5min	12	12
PEMS08	17856	170	Jul-Aug, 2016	5min	12	12

The ratios of train set, validation set and test set on four datasets are 2:0:1, 2:1:1, 6:2:2 and 6:2:2. All measurements are normalized to [0,1]. In the experiment, historical time window and the forecasting window on Milan dataset are set to 6, other datasets are set to 12. The hyperparameters of AMF-STGCN are determined by the performance on the validation set. The size of graph convolution kernels are set to 3×1, 1×3, 5×2, 3×2, 2×3, which are used in spatial-temporal inception branches respectively. We evaluate the performance of different methods using Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). Here, all results are denormalized before evaluation.

B. Experimental Settings

For Jiangsu, PEMS04 and PEMS08, adjacency matrix and parameters are same as in STGCN [1]. For Milan dataset, Spearman correlation coefficient was used to define the adjacency matrix, and threshold is set to 0.92.

We compare AMF-STGCN with other widely used baseline models, including HA (Historical Average method), ARIMA (Auto-Regressive Integrated Moving Average), LSTM [15], STGCN [1], ASTGCN [4], AGCRN [6] and STSGCN [7].

C. Comparison and Result Analysis

Table II shows the overall performance on four real-world datasets. Due to the huge computation cost, it is hard to measure the performance of STSGCN on mobile traffic datasets with more nodes. It is found that AMF-STGCN achieves state-of-the-art results on all datasets. The improvements of MAE on four datasets are between 1.94% and 14.18%, and the improvements of RMSE range from 3.76% to 11.48%. This also shows that AMF-STGCN is more robust in spatial-temporal forecasting tasks.

The performance improvements of AMF-STGCN on mobile traffic datasets are much greater than road traffic datasets. The

TABLE II
PERFORMANCE COMPARISON OF DIFFERENT APPROACHES

Dataset	**Milan**		**Jiangsu**		**PEMS04**		**PEMS08**	
Model	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**
HA	61.28	120.73	184.94	358.00	61.80	87.27	31.94	46.34
ARIMA	54.61	78.94	174.13	445.47	32.11	68.13	24.04	43.30
LSTM	42.93	80.29	198.37	300.95	29.04	45.63	25.93	38.64
STGCN	34.77	63.28	175.81	306.24	22.15	34.68	18.49	28.18
ASTGCN	39.42	72.04	164.28	300.20	22.99	35.42	18.89	28.64
AGCRN	42.27	135.18	131.83	279.51	20.11	32.88	16.19	25.56
STSGCN	\	\	\	\	21.29	33.81	17.18	26.85
AMF-STGCN	**29.84**	**57.73**	**129.28**	**247.41**	**19.45**	**31.57**	**15.53**	**24.60**
Improvements	+14.18%	+8.77%	+1.94%	+11.48%	+3.28%	+3.98%	+4.08%	+3.76%

improvements of MAE and RMSE on the mobile traffic datasets are 14.18% and 11.48%, while 4.08% and 3.98% on road traffic datasets. Based on previous data analysis results, we found that mobile traffic datasets have more obvious heterogeneity than road traffic datasets. Our model has more improvements in mobile traffic datasets, proving that it can extract spatial-temporal heterogeneity.

The methods based on GCN (STGCN, ASTGCN, AGCRN, STSGCN, AMF-STGCN) are more effective than time series forecasting methods (HA, ARIMA, LSTM), which illustrates the limitations of time series forecasting methods in modeling spatial-temporal data. Since STGCN and ASTGCN model the spatial-temporal dependencies separately and do not take heterogeneity into account, their effects are worse than AGCRN, STSGCN and AMF-STGCN in most datasets. Because AGCRN is limited by the dimension of node embedding, its generalization ability is weak. Besides, STSGCN only considers local spatial-temporal heterogeneity. Both of those methods are difficult to achieve optimal results. However, our AMF-STGCN can model spatial-temporal dependencies and nodes' heterogeneity at the same time, so it has advantages in modeling complex scenarios of traffic forecasting.

Fig. 6 shows changes of different metrics on the Milan and PEMS08 datasets with increasing prediction length. The metrics value of HA is too large to be displayed completely. As we can see from the Fig. 6, the error increases over time, indicating that forecasting task becomes more difficult. The performance of GCN-based methods is stable, which shows the effectiveness of capturing spatial information. Although our model has no outstanding performance in short-term forecasting tasks, it shows great advantages in medium and long-term forecasting tasks. This benefits from Fully-Connected decoder which alleviates the error accumulation of multi-step forecasting.

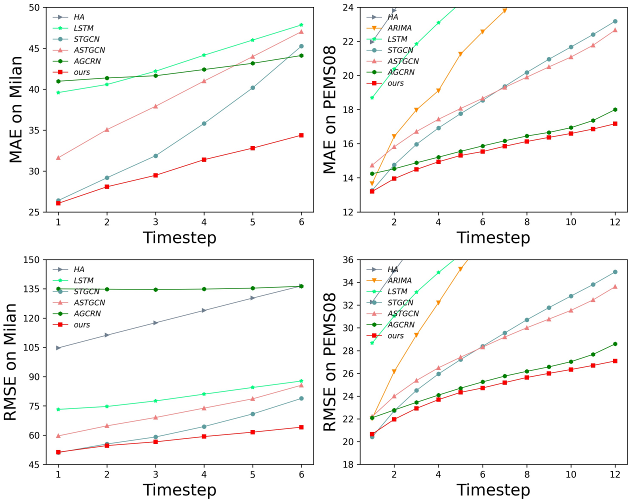


Fig. 6. Forecasting performance comparison at each time step.

D. Ablation Study

In this section, we further carry out ablation tests on Jiangsu and PEMS08 datasets to study the effectiveness of each key component used in AMF-STGCN, including: (1) AMF-STConv block, (2) AMF-Attention, (3) mask. Except for the different control variables, the settings of each experiment were the same.

Table III shows performance of AMF-STGCN against its ablations. We can observe that: (1) Replacing AMF-STConv block with spatial-temporal convolutional block of STGCN (w/o AMF-STConv block), the model performs worst compared with other ablation models. The result proves the superior ability of AMF-STConv block to capture heterogeneity of graph nodes and model spatial-temporal dependencies simultaneously. (2) Without attention in AMF-STConv block (w/o AMF-Att), the performance is second worst. We believe that model cannot extract heterogeneity. Take Jiangsu dataset as an example, the attention scores of Node0 and Node1 are [0.0015, 0.94, 0.003, 0.057] and [0.015, 0.31, 0.029, 0.65] respectively. It means Node0 is most affected by the second STConv kernel, whose effective affected range is within 5 hops in spatial dimension, and 2 time steps in temporal dimension. While Node1 is most affected by the fourth STConv kernel, followed by the second

TABLE III
ABLATION RESULTS OF AMF-STGCN

Dataset	**Jiangsu**		**PEMS08**	
Model	**MAE**	**RMSE**	**MAE**	**RMSE**
AMF-STGCN	129.28	247.42	9.71	15.53
w/o AMF-STConv block	141.25	263.73	10.53	17.01
w/o AMF-Att	134.64	252.95	9.94	16.18
w/o mask	129.93	248.07	9.73	15.56

kernel. (3) With the addition of the mask mechanism, the MAE is reduced by 0.56 and 0.2, and the RMSE is reduced by 0.55 and 0.03, respectively. This shows that the mask mechanism can further improve the robustness of the model.

E. Computational Complexity

We compare computational complexity between AMF-STGCN, STGCN, ASTGCN, AGCRN and STSGCN on the PEMS04 dataset. The results are shown in Table IV. The experiments are conducted on Tesla V100 server. STGCN uses a fully convolutional structure, so its training speed is fastest. ASTGCN introduces temporal and spatial attention on the basis of STGCN to capture dynamic spatial-temporal dependencies, so the amount of parameters and training time are greatly increased. Compared with AMF-STGCN, the above two methods have limitations in terms of interpretability and generalization ability. AGCRN introduces a large number of parameters, as it uses learnable node embedding to represent node status. In addition, AGCRN uses recurrent structures to model temporal dependence, which is more time-consuming. STSGCN constructs STSGC modules based on the complex localized spatial-temporal graphs to model localized spatial-temporal correlations. The parameters of STSGCM layers are not shared, resulting in a linear increase in the number of parameters. Compared with the aforementioned methods, AMF-STGCN is more suitable for applications.

At present, the proposed method has been deployed in the network of a certain province in China. By introducing the spatial-temporal forecasting model into operation and maintenance process, operators can identify cells with high volume of business in advance and plan the maintenance priority of these cells.

TABLE IV
THE COMPUTATIONAL COMPLEXITY ON THE PEMS04 DATASET

Models	Parameters	Training time(s/epoch)
AMF-STGCN	677591	45.88
STGCN	211596	10.81
ASTGCN	450031	26.74
AGCRN	748810	72.38
STSGCN	2872686	109.93

IV. CONCLUSION

In this paper, we elaborate on a new GCN-based model, AMF-STGCN. We evaluate our method on four real-world datasets from two different domains, and obtain consistent better results comparing to various advance models in the literature of prediction. In the future, it is valuable to model dynamics and abrupt changes of mobile traffic data, as well as solve the massive node scenarios problem.

REFERENCES

[1] B. Yu, H. Yin, and Z. Zhu, “Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting,” arXiv Prepr. arXiv1709.04875, 2017.

[2] B. M. Williams and L. A. Hoel, “Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: Theoretical basis and empirical results,” J. Transp. Eng., vol. 129, no. 6, pp. 664–672, 2003.

[3] Y. Li, R. Yu, C. Shahabi, and Y. Liu, “Diffusion convolutional recurrent neural network: Data-driven traffic forecasting,” arXiv Prepr. arXiv1707.01926, 2017.

[4] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan, “Attention based spatial-temporal graph convolutional networks for traffic flow forecasting,” AAAI Conference on Artificial Intelligence, vol. 33, pp. 922–929, 2019.

[5] K. He, Y. Huang, X. Chen, Z. Zhou, and S. Yu, “Graph attention spatial-temporal network for deep learning based mobile traffic prediction,” 2019 IEEE Glob. Commun. Conf. GLOBECOM 2019 - Proc., pp. 1–6, 2019.

[6] L. Bai, L. Yao, C. Li, X. Wang, and C. Wang, “Adaptive Graph Convolutional Recurrent Network for Traffic Forecasting,” in Advances in Neural Information Processing Systems, vol. 33, pp. 17804–17815, 2020.

[7] C. Song, Y. Lin, S. Guo and H. Wan, “Spatial-Temporal Synchronous Graph Convolutional Networks: A New Framework for Spatial-Temporal Network Data Forecasting,” AAAI Conference on Artificial Intelligence, vol. 34, pp. 914–921, 2020.

[8] C. Zheng, X. Fan, C. Wang, and J. Qi, “Gman: A graph multi-attention network for traffic prediction,” AAAI Conference on Artificial Intelligence, vol. 34, no. 01, pp. 1234–1241, 2020.

[9] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” arXiv Prepr. arXiv1609.02907, 2016.

[10] C. Szegedy et al., “Going deeper with convolutions,” IEEE conference on computer vision and pattern recognition, pp. 1–9, 2015.

[11] A. Vaswani et al., “Attention Is All You Need,” arXiv, 2017.

[12] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, “Squeeze-and-Excitation Networks,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 42, no. 8, pp. 2011–2023, 2020.

[13] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778, 2016.

[14] C. Chen, K. Petty, A. Skabardonis, P. Varaiya, and Z. Jia, “Freeway Performance Measurement System: Mining Loop Detector Data,” Transp. Res. Rec. J. Transp. Res. Board, vol. 1748, pp. 96–102, 2001.

[15] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” Neural Comput., vol. 9, no. 8, pp. 1735–1780, 1997.

[16] X. Li, W. Wang, X. Hu, and J. Yang, “Selective Kernel Networks,” 2019.

[17] B. Yu, M. Li, J. Zhang, and Z. Zhu, “3d graph convolutional networks with temporal graphs: A spatial information free framework for traffic forecasting,” arXiv Prepr. arXiv1903.00919, 2019.

[18] B. G et al., “A multi-source dataset of urban life in the city of milan and the province of trentino,” Sci. Data, vol. 2, 2015.