Spatio-Temporal Memory Augmented Multi-Level Attention Network for Traffic Prediction

Yan Liu, Bin Guo, *Senior Member, IEEE*, Jingxiang Meng, Daqing Zhang, *Fellow, IEEE*, and Zhiwen Yu, *Senior Member, IEEE*

***Abstract*—Traffic prediction is one of the fundamental spatio-temporal prediction tasks in urban computing, which is of great significance to a wide range of applications, e.g., traffic controlling, vehicle scheduling, etc. Recently, with the expansion of the city and the development of public transportation, long-range and long-term spatio-temporal correlations play a more important role in traffic prediction. However, it is challenging to model long-range spatial dependencies and long-term temporal dependencies simultaneously in two aspects: 1) complex influential factors, including spatial, temporal and external factors. 2) multiple spatio-temporal correlations, including long-range and short-range spatial correlations, as well as long-term and short-term temporal correlations. To solve these issues, we propose a spatio-temporal memory augmented multi-level attention network for fine-grained traffic prediction, entitled ST-MAN. Specifically, we design a spatio-temporal memory network to encode and memorize fine-grained spatial information and representative temporal patterns. Then, we propose a multi-level attention network to explicitly model both short-term local spatio-temporal dependencies and long-term global spatio-temporal dependencies at different spatial scales (i.e., grid and region levels) and temporal scales (i.e., daily and weekly levels). In addition, we design an external component that takes external factors and spatial embeddings as inputs to generate location-aware influence of the external factors much more efficiently. Finally, we design an end-to-end framework optimized with the contrastive objective and supervised objective to boost model performance. Empirical experiments over coarse-grained and fine-grained real-world datasets demonstrate the superiority of the ST-MAN model compared to several state-of-the-art baselines.**

***Index Terms*—Attention network, memory network, spatio-temporal prediction, traffic prediction, urban computing.**

Manuscript received 15 June 2022; revised 30 August 2023; accepted 30 September 2023. Date of publication 16 October 2023; date of current version 19 April 2024. This work was supported in part by the National Science Fund for Distinguished Young Scholars under Grant 62025205, in part by the National Key R&D Program of China under Grant 2019YFB1703901, in part by the National Natural Science Foundation of China under Grants 62032020, 61960206008, and 61725205, in part by the Young Scientists Fund of the National Natural Science Foundation of China under Grant 62302017, and in part by China Postdoctoral Science Foundation under Grant 2023M730058. Recommended for acceptance by K. Zheng. *(Corresponding author: Bin Guo.)*

Yan Liu is with Peking University, Beijing 100871, China (e-mail: yan_emily@outlook.com).

Bin Guo is with Northwestern Polytechnical University, Xi'an, Shaanxi 710129, China (e-mail: guob@nwpu.edu.cn).

Jingxiang Meng is with the Industrial and Commercial Bank of China, Xi'an, Shaanxi 710129, China (e-mail: mengjx@sdc.icbc.com.cn).

Daqing Zhang is with Peking University, Beijing 100871, China, and also with Télécom SudParis, Évry, 91011 Essonne, France (e-mail: dqzhang@sei.pku.edu.cn).

Zhiwen Yu is with Harbin Engineering University, Harbin, Heilongjiang 150001, China, and also with the Northwestern Polytechnical University, Xi'an, Shaanxi 710129, China (e-mail: zhiwenyu@nwpu.edu.cn).

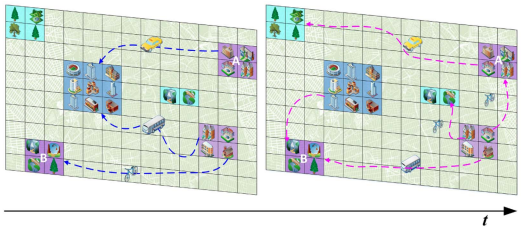
Digital Object Identifier 10.1109/TKDE.2023.3322405

I. Introduction

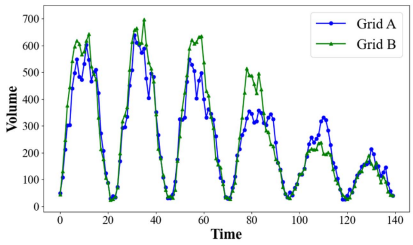
In recent years, spatio-temporal prediction has become increasingly significant in our daily life by providing useful and necessary information, such as traffic prediction [1], crowd flow prediction [2], air quality prediction [3], etc. Traffic prediction is one of the fundamental spatio-temporal prediction tasks, which plays a more and more important role in urban computing due to the increasing availability of large-scale traffic data. Traffic prediction [4], [5], [6] aims to predict the potential traffic volume (e.g., traffic in/out flow volume, passenger pickup/dropoff demand volume, etc.) based on historical observations, and it can provide insights to the government for traffic controlling, vehicle scheduling, etc. However, it could be largely affected by a broad range of complicated spatial and temporal factors and thus still suffers from some challenges.

One of the main pain points in traffic prediction that must be addressed is to model complicated spatio-temporal dependencies, since the potential traffic volume in a region not only be related to its previous observations, but also be influenced by neighbors' histories. Various spatio-temporal prediction models [2], [7], [8], [9] have been proposed to capture spatial and temporal correlations. Traditional works focus on obtaining sequential patterns based on historical observations by some times-series approaches, such as ARIMA [10], but they typically ignore spatial dependencies between different regions, and fail to model complex non-linear relations. Recently, a lot of deep learning models [11], [12] have achieved promising performance in spatio-temporal prediction tasks. Convolutional neural network (CNN) based methods have the ability to extract spatial dependencies among different regions, such as DeepST [13]. Recurrent neural network (RNN) based approaches are good at modeling temporal correlations by embedding sequential records into a hidden state vector [14], [15].

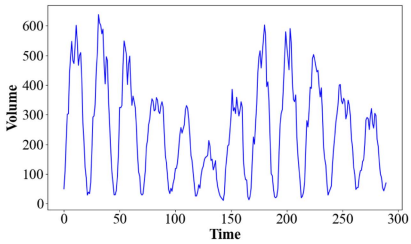
With the expansion of the city and the development of public transportation, *long-range and long-term spatio-temporal factors* play a more important role in urban traffic prediction. Fig. 1 illustrates a grid-based traffic flow use case in a city, where the city is divided into numerous small grids, and each region consists of multiple grid cells with similar functions, such as office regions and residential regions. *For spatial dependencies, the traffic volume between distant regions could be related to each other.* On one hand, with the expansion of human mobility, more and more people working in the center of the city (e.g., blue grids in Fig. 1(a)) tend to live in the suburban area (e.g.,



(a) Traffic Flow Transitions among different regions.



(b) Long-range Spatial Dependencies.



(c) Long-term Temporal Dependencies.

Fig. 1. Long-range and long-term spatio-temporal information are necessary for traffic volume prediction. For clarity, the city is divided into a lot of small grids, and each region consists of many grid cells with similar functions, such as office regions (blue grids), residential regions (purple grids) and leisure regions (green grids), etc.

purple grids in Fig. 1(a)), implying the long-range geographical correlations of traffic flow transitions. On the other hand, due to the functional division of the city, two distant regions may reflect similar traffic patterns (e.g., Grid A and Grid B in Fig. 1(b)), implying the long-range semantic correlations of traffic flow patterns. *For temporal dependencies, the traffic volume exhibits long-term periodic patterns, e.g., daily and weekly.* For example, on the daily scale, the traffic volume in the region follows a similar trend that increases during the morning and decreases during the night; on the weekly scale, the traffic flow pattern trends to repeat every week, as shown in Fig. 1(c).

Some advances have put much effort to address these issues. Most works model long-range spatial dependencies among regions by stacking many convolutional layers, since the convolutional layer just captures short-range dependencies at a local scale. For example, ST-ResNet [16] and DeepSTN+ [17] resort to the residual mechanism to enable a deeper convolutional neural network. However, stacking too many convolutional layers to capture long-range spatial dependencies could result in high computational costs and optimization difficulties [18], which limits the predictive performance when the city is rasterized with a lot of regions. In addition, some empirical knowledge is introduced to capture various temporal patterns. MDL [19] and MVGCN [20] take account of multiple temporal properties (e.g., closeness, period and trend) based on different timestamps, which are concatenated together to model temporal dependencies. However, these temporal features are modeled in different channels respectively, combing them directly cannot imply long-term temporal patterns of different regions since some complex temporal information could be lost. In general, these existing models focus on modeling either long-range spatial features or long-term temporal features, which still have limited prediction ability.

There are few works to model long-range spatial dependencies and long-term temporal dependencies simultaneously due to complicated spatio-temporal correlations. Intuitively, it is reasonable to combine CNN and RNN to capture both spatial and temporal correlations. For example, STDN [21] adopts CNN to extract spatial features, which are then fed into RNN to further model temporal dependencies. However, modeling long-range spatial dependencies and long-term temporal dependencies separately cannot capture the inner-connection of spatio-temporal correlations. Further, researchers attempt to design additional memory cells into standard ConvLSTM units to encode long-range spatial relations [22]. Nevertheless, they still have limited representation power in long-term temporal dependencies, because the latent vector is typically too small to express complex spatio-temporal correlations. Hence, *it is still challenging to capture complicated long-range and long-term spatio-temporal dependencies.*

To bridge the gap, we aim to introduce the external memory that attempts to encode and memorize spatio-temporal information. Recently, memory networks [23] have shown their promising performance in many sequential prediction tasks, such as sequential recommendation [24], question answering [25], etc. Compared with other traditional RNN/LSTM models, memory networks employ an external memory component to store the hidden vector by appropriate reading and writing operations [23]. However, conventional memory networks cannot be directly utilized to encode complex spatio-temporal correlations, and we face three specific challenges:

1) *How to design the memory network to encode and memorize spatial and temporal information simultaneously?* Previous external memory networks focus only on storing sequential features, and the spatial information is ignored [26]. However, spatial dependencies among regions in the city are crucial to model the spatio-temporal correlations for traffic prediction.
2) *How to extract long-range spatial dependencies and long-term temporal dependencies?* Although some effective strategies [27] have been designed to read and update information from external memories, these works neglect complicated inner-connection between temporal patterns and spatial distributions.
3) *How to effectively model the location-aware impact of external factors on each grid cell?* Previous studies utilize the fully-connected layers to learn the influence of external factors (e.g., weather conditions, holidays) by mapping

features to a lot of grids in the high-dimensional flow map. However, this approach leads to a significant increase in parameters as the number of grid cells grows.

In this paper, we propose a spatio-temporal memory augmented multi-level attention network for traffic prediction, named ST-MAN. Compared with existing methods, our model has the ability to learn both long-range spatial dependencies and long-term temporal patterns for efficient spatio-temporal prediction. Specifically, we design a spatio-temporal memory network (STMN) to tackle the first challenge. STMN consists of the key memory matrix to encode global spatial correlations as prior knowledge by spatial embedding, as well as the value memory matrix to capture long-term temporal patterns via memory encoding. For the second challenge, we propose a multi-level attention network (MAN) that incorporates attention mechanisms to explicitly model both short-term local and long-term global spatio-temporal correlations. MAN consists of two primary attention modules: The short-term local attention module focuses on capturing the geographic spatio-temporal dependencies of traffic flow transitions at two spatial granularities, namely cross-grid and cross-region flow transitions. The long-term global attention module aims to model the semantic spatio-temporal dependencies related to traffic flow patterns in regions with similar functions. This module leverages the spatio-temporal memory network to store and retrieve long-term and long-range spatio-temporal information. To tackle the third challenge, we introduce an external component that learns the location-aware influence of external factors. This component generates specific responses for each grid cell based on spatial embeddings, which helps to mitigate the need for a large number of parameters. Finally, an end-to-end framework is designed for traffic prediction, which is trained based on supervised objective and contrastive objective simultaneously.

In summary, our contributions are concluded as follows:

- We propose a spatio-temporal memory augmented multi-level attention network for traffic prediction, named ST-MAN, which models both long-range spatial dependencies and long-term temporal dependencies.
- We design a spatio-temporal memory network, which is capable of encoding and memorizing fine-grained spatial information and temporal patterns. To the best of our knowledge, this is the first try that the external memory network is introduced for spatio-temporal prediction to enrich the expressiveness of the spatio-temporal model.
- We introduce a multi-level attention network to effectively capture both short-term local spatio-temporal dependencies among geographic neighbors and long-term global correlations among semantic neighbors with similar functions.
- Extensive experiments on coarse-grained and fine-grained real-world datasets demonstrate that our model achieves significant improvement compared with state-of-the-art models.

The remainder of this article is organized as follows. Section II presents the problem formulation and system framework. Section III elaborates on the detailed design of the ST-MAN. Section IV reports the empirical evaluation. In Section V, we briefly review the related work. Finally, the conclusions and future work are discussed in Section VI.

TABLE I
DESCRIPTION OF NOTATION

Symbol	Description
$\mathbf{M}_t \in \mathbb{R}^{K\times H\times W}$	Traffic volume map at time step t, H and W are the height and width of the grid-based map, K is the number of traffic volume measurements
$\mathbf{X}_c \in \mathbb{R}^{Kl_c\times H\times W}$	Temporal closeness, l_c is the input length of closeness
$\mathbf{X}_p \in \mathbb{R}^{Kl_p\times H\times W}$	Temporal period, l_p is the input length of period (i.e., daily)
$\mathbf{X}_q \in \mathbb{R}^{Kl_q\times H\times W}$	Temporal trend, l_q is the input length of trend (i.e., weekly)
$\mathbf{X}_{ext} \in \mathbb{R}^{l_e}$	External factors, l_e is the number of external factors
$\mathbf{C}_k \in \mathbb{R}^{H\times W}$	Correlation matrix between region g_k and other regions
$\mathbf{M}_k \in \mathbb{R}^{H\times W\times D_{km}}$	Key matrix in memory network
$\mathbf{M}_v \in \mathbb{R}^{H\times W\times D_{vm}}$	Value matrix in memory network
$\mathbf{Q}_c, \mathbf{K}_c, \mathbf{V}_c$	Query, key and value matrices of short-term local attention module
$\mathbf{Q}_m, \mathbf{K}_m, \mathbf{V}_m$	Query, key and value matrices of long-term global attention module
$\mathbf{F}_c$	Short-term local features
$\mathbf{F}_m$	Long-term global features
$\mathbf{F}_{final}$	Final spatio-temporal features
α, β, γ	Trade-off parameters

II. PRELIMINARIES

In this section, we first describe the formulation of the traffic volume prediction problem. Then, we present the overview of our proposed framework. For brevity, we present a table of notations used in our work, as depicted in Table I.

A. Problem Formulation

Definition 1 (Grid Cell): Following previous works [13], we represent a city area as a rectangle, and partition it into a $H \times W$ grid map along the longitude and latitude, denoted as $C \in \mathbb{R}^{H\times W}$. As shown in Fig. 1, there are a total of $H \times W$ grid cells, and they have the same size.

Definition 2 (Region): Due to the functional division of the city, neighboring grid cells often exhibit similar functions, as illustrated by the office regions (blue grids) and residential regions (purple grids) in Fig. 1. The utilization of coarse-grained regions, in contrast to the fine-grained grid-based map, allows for the effective extraction of global semantic neighborhood information.

Definition 3 (Grid-based Traffic Volume Map): Let $\mathcal{T} = \{T_1, T_2, \ldots\}$ represent a collection of mobility trips, each trip $T_i \in \mathcal{T}$ contains a sequence of geospatial coordinates across p continuous time intervals. We define the traffic volume within grid $g_{i,j}$ at time step t as the number of mobility trips within that grid during the t^{th} time interval. Formally, the traffic volume map can be denoted as a 3D tensor $\mathbf{M}_t \in \mathbb{R}^{K\times H\times W}$ over the grid-based city map, where K is the number of traffic volume

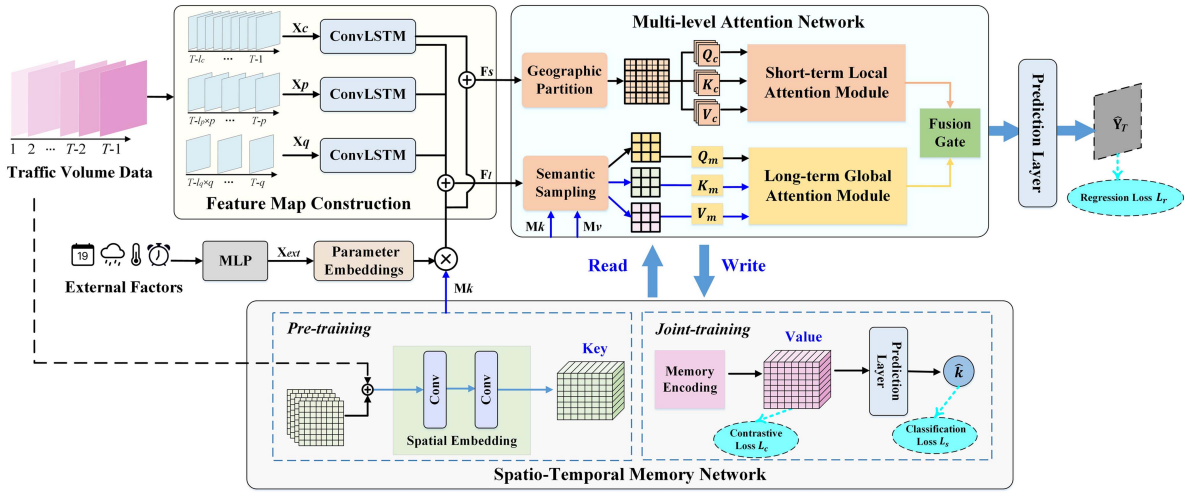


Fig. 2. Overall architecture of ST-MAN. It contains four major components: feature map construction, spatio-temporal memory network (STMN), multi-level attention network (MAN), and traffic volume prediction.

measurements. For example, K is set to 2 when considering the start/end traffic volume, and $(\mathbf{M}_t)_{0,i,j}$ and $(\mathbf{M}_t)_{1,i,j}$ indicate the number of mobility trips getting into/leaving grid $g_{i,j}$ during time interval t, respectively.

Problem Statement (Traffic Volume Prediction):

Given the historical observations $\{\mathbf{M}_t \mid t = 1, \ldots, T-1\}$ *over previous* $T-1$ *time intervals, the traffic volume prediction problem aims to predict the traffic volume map* $\mathbf{M}_T \in \mathbb{R}^{K \times H \times W}$ *at next time interval* T.

B. System Framework

The framework of ST-MAN is illustrated in Fig. 2, which consists of four major components:

1) *Feature Map Construction:* Given the input sequence of historical traffic volume data $\{\mathbf{M}_t \mid t = 1, \ldots, T-1\}$, our objective is to construct the input feature map for the prediction model, which incorporates spatial, temporal, and external information. On one hand, we consider all grid cells distributed throughout the city and select their corresponding recent, daily, and weekly timesteps as the input data, denoted as $\mathbf{X}_c$, $\mathbf{X}_p$, and $\mathbf{X}_q$, and they are further fed into the ConvLSTM to obtain spatial and temporal information. On the other hand, some external factors at each timestep, denoted as $\mathbf{X}_{ext}$, such as weather conditions and holidays, are also provided to capture the external information by parameter embeddings.
2) *Spatio-Temporal Memory Network (STMN):* STMN aims to encode fine-grained spatial dependencies and representative temporal patterns, and then stores them in key memory matrix and value memory matrix as prior knowledge to enhance the expressiveness of the model. Note that the key memory matrix is pre-trained via spatial embedding to obtain the embedding vectors of each region, while the value memory matrix is learned jointly with the prediction model. Especially, to improve memory fidelity, STMN encodes the discriminating and invariant temporal features among different regions by minimizing a contrastive objective.
3) *Multi-Level Attention Network (MAN):* MAN employs attention mechanisms to explicitly model both short-term local and long-term global spatio-temporal dependencies. Specifically, MAN comprises two attention modules designed to capture different types of spatial-temporal correlations separately. The first is the short-term local attention module, which utilizes the information obtained from recent observations by ConvLSTM. It focuses on capturing short-term local spatio-temporal dependencies in traffic flow transitions between geographic grid cells. The second is the long-term global attention module, which extracts long-range spatial dependencies and long-term temporal dependencies by incorporating essential global information from the spatio-temporal memory network. This module is responsible for capturing similar traffic flow patterns among semantic cells in the city. To obtain comprehensive feature representations, a fusion module is employed to adaptively integrate the different aspects of features. This fusion process ensures that the final feature representations capture the combined influence of short-term local and long-term global spatio-temporal dependencies.
4) *Traffic Volume Prediction:* For the traffic volume prediction task, the extracted high-level spatio-temporal features are fed into the prediction layer (e.g., the convolutional layer) to generate a predicted traffic volume map $\hat{\mathbf{Y}}_T$ at time step T.

III. Methodology

This section elaborates our proposed model in detail. We first construct the feature map for the prediction model based on ConvLSTM and the external component, including spatial and temporal information, as well as external information. Then, we present the spatio-temporal memory network

to encode fine-grained spatial information and representative temporal patterns. Furthermore, the multi-level attention work built on spatio-temporal memory network is proposed to explicitly model short-term local dependencies and long-term global dependencies, which are adaptively integrated via an effective fusion mechanism to enhance the expressiveness of the model for traffic prediction. Finally, we provide an end-to-end approach to optimize the model by adopting a joint loss function, including the contrastive objective and supervised objective.

A. Feature Map Construction

The input data plays a crucial role in traffic prediction as it encompasses various complex factors, including spatial, temporal, and external factors. In particular, the traffic volume data of a grid cell not only relies on its own historical observations but also exhibits correlations with the histories of its neighboring grids. These spatial and temporal dependencies are important aspects to consider in traffic prediction. Furthermore, traffic volume data is strongly influenced by external factors, such as weather conditions and holidays. Due to the diverse nature of the information contained in different types of data, it is necessary to construct features that are specifically tailored for traffic prediction, taking into account the unique characteristics of raw data. To address this, we begin by creating an input feature map based on the raw traffic volume data. This feature map serves as a foundation for the model to learn effective and discriminative representations, enabling it to capture the underlying patterns and dynamics of urban traffic.

1) ConvLSTM Component: The unique physical topology and social relationship structures inherent to cities give rise to urban traffic data that exhibit diverse temporal and spatial dependencies across different scales, such as short-term and long-term temporal correlations, as well as short-range and long-range spatial correlations. To model the complex spatio-temporal correlations of traffic data, we first turn historical traffic volume throughout a city at each time step into an image-like matrix respectively (i.e., grid-based traffic volume map) in the spatial dimension, and then divide the time steps into three fragments in the temporal dimension, including recent time steps, daily and weekly periodic time steps. Specifically, given the traffic volume data $\{\mathbf{M}_t \mid t = 1, \ldots, T-1\}$ over previous $T-1$ time steps, we construct three input sequences for a specific future time T, i.e., closeness, period and trend [16]. For closeness, we consider the traffic volume data at previous l_c time steps, denoted as $\mathbf{X}_c = [\mathbf{M}_{T-l_c}, \mathbf{M}_{T-(l_c-1)}, \ldots, \mathbf{M}_{T-1}] \in \mathbb{R}^{Kl_c \times H \times W}$. For period, we select the daily histories at previous l_p time steps, $\mathbf{X}_p = [\mathbf{M}_{T-l_p \times p}, \mathbf{M}_{T-(l_p-1)\times p}, \ldots, \mathbf{M}_{T-p}] \in \mathbb{R}^{Kl_p \times H \times W}$, where p is the time span (e.g., 24 hours per day). For trend, we select the weekly histories at previous l_q time steps, $\mathbf{X}_q = [\mathbf{M}_{T-l_q \times q}, \mathbf{M}_{T-(l_q-1)\times q}, \ldots, \mathbf{M}_{T-q}] \in \mathbb{R}^{Kl_q \times H \times W}$, where q is the time span (e.g., 24×7 hours per week).

After the stage of data preparation, three kinds of input sequences are then fed into the component, which aims to convert the prepared traffic data to the embeddings to model temporal and spatial properties. Fortunately, ConvLSTM [28] has shown its superior performance in modeling spatio-temporal correlations. Convolutional Long Short-term Memory (i.e., ConvLSTM) network [28], is an extension of the fully-connected LSTM network, which replaces all linear layers with convolution operations to capture spatial dependencies apart from temporal dependencies. In this paper, ConvLSTM is leveraged to capture the spatio-temporal information from different kinds of input traffic volume. Note that three ConvLSTM components share the same network structure, but use non-shared parameters. The structure of the ConvLSTM unit is shown in Fig. 3. Formally, let $\mathbf{H}_{t-1}$ and $\mathbf{C}_{t-1}$ denote the hidden state and cell state in a memory cell at time interval $t-1$. Given the input data $\mathbf{X}_t$ at current time t, we could model the spatio-temporal representation $\mathbf{H}_t$ and obtain the new memory $\mathbf{C}_t$ at time t by updating operations:

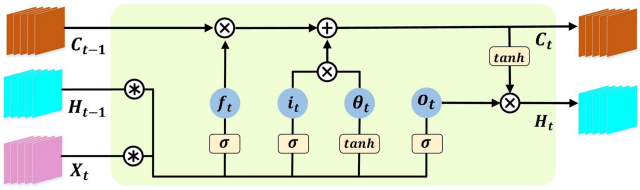


Fig. 3. ConvLSTM Component.

$$
\begin{aligned}
i_t &= \sigma\left(\mathbf{W}_{xi} * \mathbf{X}_t + \mathbf{W}_{hi} * \mathbf{H}_{t-1} + \mathbf{b}_i\right) \\
f_t &= \sigma\left(\mathbf{W}_{xf} * \mathbf{X}_t + \mathbf{W}_{hf} * \mathbf{H}_{t-1} + \mathbf{b}_f\right) \\
o_t &= \sigma\left(\mathbf{W}_{xo} * \mathbf{X}_t + \mathbf{W}_{ho} * \mathbf{H}_{t-1} + \mathbf{b}_o\right) \\
\theta_t &= \tanh\left(\mathbf{W}_{xc} * \mathbf{X}_t + \mathbf{W}_{hc} * \mathbf{H}_{t-1} + \mathbf{b}_c\right) \\
\mathbf{C}_t &= f_t \circ \mathbf{C}_{t-1} + i_t \circ \theta_t \\
\mathbf{H}_t &= o_t \circ \tanh\left(\mathbf{C}_t\right)
\end{aligned} \tag{1}
$$

where $*$ represents convolutional operations, $\circ$ represents the Hadamard product. $\mathbf{W}$ and $\mathbf{b}$ are learnable parameters, σ is logistic sigmoid function. i_t, f_t, o_t are the input gate, forget gate, and output gate, respectively.

2) External Component: External factors, such as weather conditions and holidays, have a significant impact on the traffic volume in different regions of a city. For instance, traffic patterns on weekends can differ greatly from those on weekdays. Previous studies mainly use fully-connected layers to map the effects of external factors onto each grid cell [16], [17], including the embedding layers to combine each factor and the final layer to map these embeddings to high-dimensional features with the same shape as the input flow map. However, when predicting traffic volume at a fine-grained grid level(e.g., 128×128 traffic volume map), the increasing number of grid cells poses a challenge in terms of parameter explosion. Specifically, the last layer of the model would require a massive number of parameters, directly proportional to the number of grid cells in the final layer.

To address the aforementioned challenge, we propose an external component to learn the location-aware influence of external factors in different grid cells. Previous research [29] provides insights into the relationship between external factors and grid cells, suggesting that correlated grid cells may exhibit

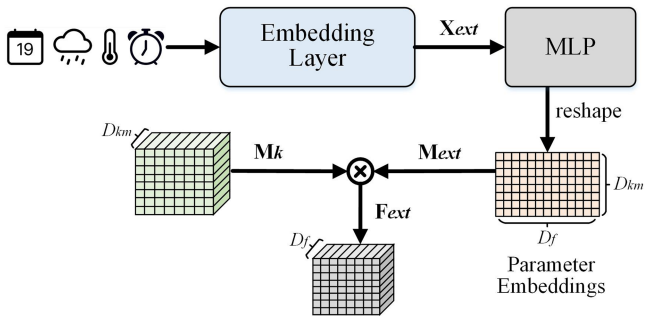


Fig. 4. Diagram of the external component.

similar responses to external factors. Therefore, we aim to leverage the information specific to each grid cell to generate grid-specific responses to the dynamic external factors, independent of the number of grid cells. In Section III-B1, we introduce the spatial embedding method, which captures spatial correlations among grids within the city. The embedding vectors derived from this method encapsulate unique information for each grid, serving as valuable prior knowledge. By leveraging these grid embeddings, we compute the influence of external factors on each grid cell, thereby avoiding the need to introduce a large number of additional parameters.

For external factors, we first utilize an embedding layer followed by the fully-connected layers to generate the parameter embeddings according to the recent work, which computes cell-specific responses based on Matrix Factorization [29]. For simplicity, we use $\mathbf{X}_{ext} \in \mathbb{R}^{l_e}$ to denote the embedding vector of the external feature, which is further fed into two fully-connected layers to generate the parameter embeddings $\mathbf{M}_{ext} \in \mathbb{R}^{l_p}$. To enable the location-aware impact of external factors, we introduce grid-cell embeddings to learn the specific response for each grid cell. Specifically, given the grid-cell embeddings $\mathbf{M}_k \in \mathbb{R}^{H \times W \times D_{km}}$ (i.e., the key matrix in spatio-temporal memory network presented in Section III-B1), we can reshape the parameter embeddings as $\mathbf{M}_{ext} \in \mathbb{R}^{D_{km} \times D_f}$, where $l_p = D_{km} \times D_f$. In this way, we could compute the cell-specific external features $\mathbf{F}_{ext} \in \mathbb{R}^{H \times W \times D_f}$, which satisfies $\mathbf{F}_{ext} = \mathbf{M}_k \mathbf{M}_{ext}$. The detailed computation process of our proposed external component is illustrated in Fig. 4.

B. Spatio-Temporal Memory Network

Despite the limited ability of traditional CNN-based and RNN-based models to model both long-range and long-term spatio-temporal correlations, we aim to introduce the prior knowledge in an explicit and adaptive manner, which could encode valuable knowledge based on historical data to enhance the expressiveness of the model. With the power to read and write long-term memory in memory slots, memory network [24] has been used to provide the additional representation of knowledge to increase the model capacity in many sequential tasks. However, most memory networks are insufficient to capture and store spatial information of different regions in the city. Therefore, we propose spatio-temporal memory network (STMN) to provide additional spatial and temporal information simultaneously for spatio-temporal prediction.

We introduce the key matrix and value matrix to store the spatial correlations and temporal patterns of each region in memory slots respectively, and then design appropriate training operations on two matrices in a more effective manner. Specifically, the grid-level spatial information is first encoded by spatial embedding as prior knowledge in the key matrix. With the incorporation of spatial embedding vectors in the key matrix, we could obtain the global relations among regions. Further, the value matrix is designed so that the model can learn to use keys to query relevant memories with respect to current information. Note that we leave how to effectively read the memory in the next section, and focus on how to design the spatio-temporal memory to effectively memorize available information in this section. In general, with the ability to encode, store, and update spatio-temporal patterns, STMN could enrich the representation capacity of the model.

1) Spatial Embedding: Generally, for the traffic volume prediction, each grid in the city is not independent, while they are correlative to each other. Therefore, we aim to encode global spatial correlations among grids in the city, which will be stored in a key memory matrix as grid-level prior knowledge. In particular, we measure the correlations of grids from two aspects: *spatial distribution* and *functional similarity*. For spatial distribution, the potential traffic volume in a grid could be impacted by nearby grids. For functional similarity, some grids could share similar temporal patterns with other grids in one type of area (e.g., residential area).

Inspired by word2vec [30], we present a traffic-volume-based spatial embedding approach to encode each grid into a vector, so that relevant grids would be close in the latent space. Intuitively, the embedding matrix should be learned by minimizing the distance of relevant grids in the latent space. In order to learn the embedding matrix in an effective manner, we further construct the training instances by sampling positive instances and negative instances for each grid in view of the correlation of grids. Formally, given historical traffic volume maps $\mathbf{M}_t$, for each grid g_k in the city, we first calculate a distance matrix $\mathbf{D}_k \in \mathbb{R}^{H \times W}$ based on Manhattan Distance to measure the spatial distribution, and compute a similarity matrix $\mathbf{S}_k \in \mathbb{R}^{H \times W}$ based on Pearson Correlation Coefficient to measure the functional similarity. Then they are combined to measure the correlation between grid g_k and other grids in the city, $\mathbf{C}_k = \mathbf{S}_k - \lambda \mathbf{D}_k$, where λ is the trade-off parameter.

Let $\mathbf{M_k} \in \mathbb{R}^{N \times D_{km}}$ denotes the key matrix to store the embedding vector of each grid, where $N = H \times W$, D_{km} is the dimension of embedding vector. To adaptively obtain and update the spatial embedding vector according to various mobility patterns, we unitize the convolutional network to integrate the information of historical volume data $\mathbf{M_a}$ and learned embedding matrix $\mathbf{M_b}$ in the previous stage. As shown in Fig. 5, $\mathbf{M_k} = \mathbf{W}_r * (\mathbf{M_a} \oplus \mathbf{M_b})$, where $*$ denotes the convolutional operation, $\mathbf{W}_r$ is the learnable parameter matrix. To learn the embedding matrix effectively, we sample positive and negative pairs for each grid based on $\mathbf{C}_k$ and a threshold μ to construct training instances. Specifically, for grid g_k, we sample a grid g_p

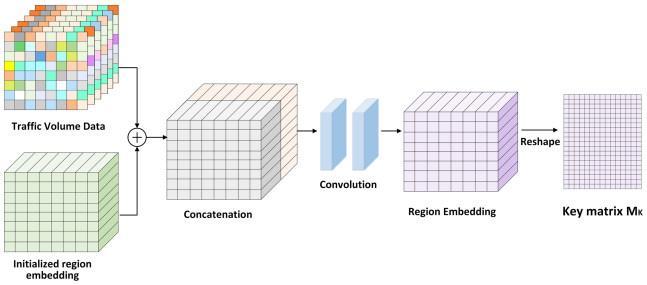


Fig. 5. Traffic-volume-based spatial embedding.

whose correlation value c_{kp} is larger than μ as positive instance, while sample a grid g_n whose correlation value c_{kn} is less than μ as negative instance, denoted as triples $(g_k, g_p, 1)$ and $(g_k, g_n, 0)$. Then the optimization objective of spatial embedding is defined as:

$$\hat{y} = \left(\mathbf{M_k} \cdot \mathbf{z}_i\right)^T \left(\mathbf{M_k} \cdot \mathbf{z}_j\right)$$
$$\mathcal{L}^r = -\frac{1}{N}\sum_{n=1}^{N}\left[y_n \times \log\left(\hat{y}_n\right) + \left(1 - y_n\right) \times \log\left(1 - \hat{y}_n\right)\right] \tag{2}$$

where $\hat{y}_n$ is prediction value, and $y_n \in (0, 1)$ is ground truth. $\mathbf{z}_i$ and $\mathbf{z}_j$ are one-hot vectors that denote the index of grids in each training instance, thus $\mathbf{M_k} \cdot \mathbf{z}_i$ and $\mathbf{M_k} \cdot \mathbf{z}_j$ represent the embedding vectors of grids g_i and g_j, respectively.

2) Memory Encoding: With the incorporation of the key matrix, we further introduce a value matrix to memorize long-term spatio-temporal patterns. To maintain effective information in the memory component, some work [23] design writing/updating operations inspired by neural turing machine (NTM), where the memory matrix in the memory network will be erased first before new information is added at each step. However, these updating operations ignore the spatial correlations among different grids. For example, some grids in business areas could share similar temporal patterns, which are different from grids in residential areas. Therefore, we design a memory encoding strategy to learn and write representative temporal patterns into the value matrix.

For the value matrix in STMN, our goal is not only to encode long-term spatio-temporal patterns, but also to learn good representations that are transferrable to other grids with the same functionality. Intuitively, for traffic volume prediction tasks, long-term spatio-temporal patterns are more stable than short-term patterns over a period of time. Therefore, we jointly learn the value matrix with the prediction model as shared knowledge instead of traditional writing operations at each step. Here, we present the optimization objective of the memory network, and the reading operation will be introduced in the next section.

Recently, many models based on contrastive learning have achieved outstanding performance in unsupervised learning tasks due to the powerful ability to learn good representations [31]. The goal of contrastive learning approaches is to learn an embedding space in which similar sample pairs stay close to each other while dissimilar ones are far apart, so we can obtain the representations invariant to different views of the same instance. Motivated by contrastive learning [32], we aim to learn the value matrix to encode the representative features invariant to different grids, which is trained with a contrastive NT-Xent loss.

Let $\mathbf{M_v} \in \mathbb{R}^{N \times D_{vm}}$ be the value matrix to store the feature representation of each grid, where $N = H \times W$, D_{vm} is the dimension of feature representation. We first cluster all grids into S classes by clustering historical observations using K-means ++ [33]. Then, we randomly sample a minibatch of S grids from S clusters $B^1 = (g_1^1, \ldots g_i^1 \ldots, g_S^1)$, and then sample another minibatch $B^2 = (g_1^2, \ldots g_i^2 \ldots, g_S^2)$. Given two grids in a cluster (g_i^1, g_i^2), we treat the pair as a positive example, and treat other $(2S-2)$ pairs of B^1 and B^2 derived from the minibatch as negative examples. Let $\mathbf{v_i}$ denote the feature vector of grid g_i in the value matrix. Then the loss function for a positive pair (g_i^1, g_i^2) is defined as:

$$\mathcal{L}\left(g_i^1, g_i^2\right) = -\log\frac{\exp\left(\mathrm{sim}\left(\mathbf{v}_i^1, \mathbf{v}_i^2\right)/\tau\right)}{\varphi_1 + \varphi_2}$$
$$\varphi_1 = \sum\nolimits_{s=1}^{S} \mathbb{I}_{[s \neq i]} \exp\left(\mathrm{sim}\left(\mathbf{v}_i^1, \mathbf{v}_s^1\right)/\tau\right)$$
$$\varphi_2 = \sum\nolimits_{s=1}^{S} \mathbb{I}_{[s \neq i]} \exp\left(\mathrm{sim}\left(\mathbf{v}_i^1, \mathbf{v}_s^2\right)/\tau\right) \tag{3}$$

where $\mathrm{sim}(\mathbf{v}_1, \mathbf{v}_2) = \frac{\mathbf{v}_1^T \mathbf{v}_2}{\|\mathbf{v}_1\|\|\mathbf{v}_1\|}$ is the cosine similarity measure, τ is the temperature parameter that regulates the level of similarity measurement.

The final loss is obtained across all positive pairs:

$$\mathcal{L}_c = \frac{1}{2S}\sum_{i=1}^{S}\mathcal{L}\left(g_i^1, g_i^2\right) + \mathcal{L}\left(g_i^2, g_i^1\right) \tag{4}$$

To further learn the discriminative representation to enhance the robustness of the model, a fully-connected layer is leveraged to predict the class of each grid $c_i \in \{1, \ldots, S\}$. Output $\hat{\mathbf{c}}_i$ is an S-way softmax that predicts the probability distribution over S different clusters, and N is the total number of grids in the city.

$$\hat{\mathbf{c}}_i = \tanh\left(\mathbf{W}_c \mathbf{v}_i + \mathbf{b}_c\right)$$
$$\mathcal{L}_s = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{S} c_{ij} \log\left(\hat{c}_{ij}\right) \tag{5}$$

C. Multi-Level Attention Network

Generally speaking, urban traffic prediction involves two main types of spatio-temporal dependencies: *geographic and semantic spatio-temporal correlations*. The first type of dependency is derived from the geographic interactions between traffic flows in different areas, known as dynamic traffic flow transitions. This type of dependency is prevalent among neighboring areas and is primarily influenced by recent traffic flow. We refer to this as *short-term local spatio-temporal dependencies*. The second type of dependency arises from the semantic associations of traffic flow patterns, specifically the long-term variations in traffic volume. Furthermore, due to the functional division of a

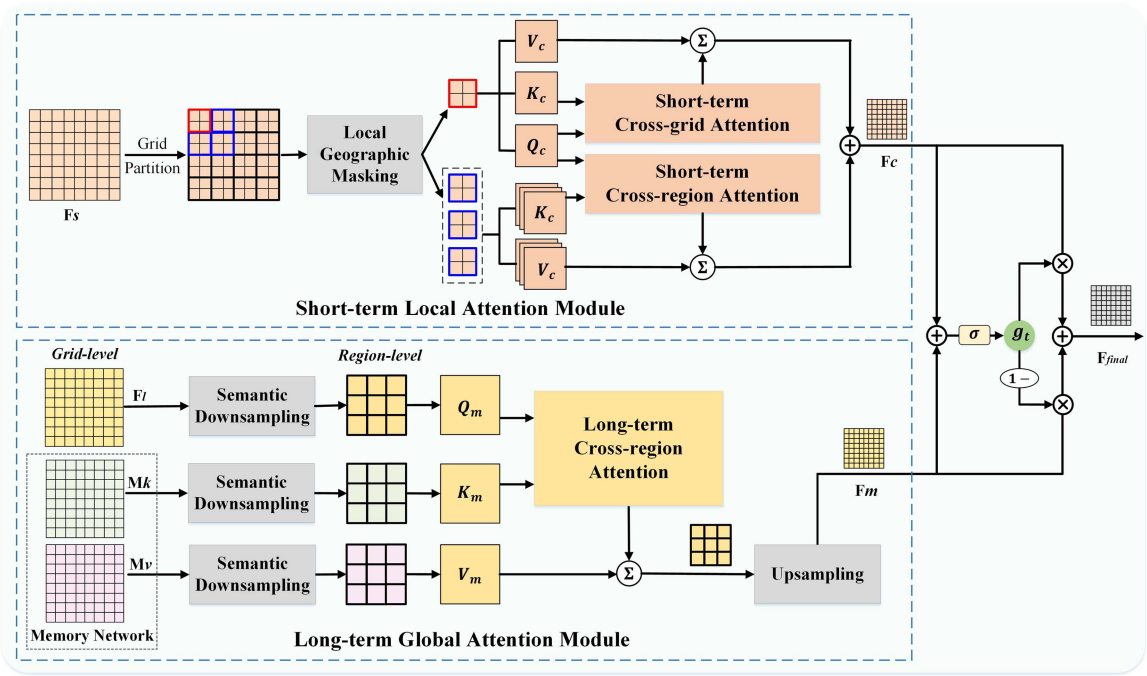


Fig. 6. Multi-level attention network. It contains two attention modules: short-term local attention (SLA) and long-term global attention (LGA).

city, distant areas may exhibit similar traffic patterns, leading to *long-term global spatio-temporal dependencies*. These two types of spatio-temporal dependencies have distinct essences, yet they are closely interconnected and play decisive roles in urban traffic prediction.

Although ConvLSTM has achieved impressive performance on some spatio-temporal prediction tasks, it still has limited representation power in modeling long-range and long-term spatio-temporal dependencies, which are significant in traffic volume prediction. Inspired by the self-attention mechanism [34], which is capable of aggregating features among all spatial positions, we leverage self-attention to capture spatial correlations at a global scale. In addition, STMN presented in the previous section is used to encode and store long-range grid-level spatial dependencies and long-term representative temporal patterns. Therefore, we propose the Multi-Level Attention Network (MAN), which harnesses the power of the external memory network and attention network to capture long-term global correlations and short-term local correlations simultaneously. Particularly, we use the features computed by ConvLSTM as the input of the self-attention module instead of the original raw traffic volume, because it encodes sequential information of input data. The framework of MAN is shown in Fig. 6, and it consists of two modules: short-term local attention and long-term global attention.

1) Short-Term Local Attention Module (SLA): The goal of SLA is to model short-range spatial dependencies and short-term temporal dependencies at the local scale, because most traffic flow transitions involve the interaction between nearby areas. As shown in Fig. 7, we can see that the majority of traffic flow transitions occur within a distance of 10 km or less, and there are relatively few traffic flows that span the entire city.

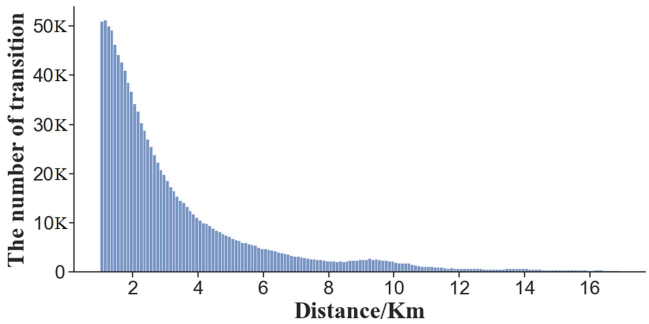


Fig. 7. Example of traffic flow transitions in New York.

Therefore, considering the geographic attributes of human mobility trajectories, it is reasonable to model short-term local spatial dependencies. As presented in Section III-A1, the output $\mathbf{F}_c$ of ConvLSTM captures the short-term temporal information from the input sequence of closeness, so we use it as the input features of SLA. In addition, considering the external factors, the input feature map of SLA could be concatenated with external features F_{ext}. Next, we feed the input map $\mathbf{F}_s$ to SLA to extract short-term local spatio-temporal features, where $\mathbf{F}_s = \mathbf{F}_c + \mathbf{F}_{ext}, \in \mathbb{R}^{H \times W \times D}$.

For the simple self-attention mechanism in the traditional spatial attention module, each grid in the city will interact with all other grids. However, from the short-range spatial view, only the interaction between nearby grids is necessary. Therefore, we introduce a geographic masking strategy for SLA to model short-term local spatio-temporal dependencies. Specifically, the interaction between two grids is considered in SLA when their distance is short. To facilitate explanation, we will introduce the

local attention module from two aspects, including cross-grid attention and cross-region attention.

To simplify the representation, we consider a city at two spatial scales, grid-based map $C \in \mathbb{R}^{H \times W}$ (e.g., 128×128), and region-based map $C' \in \mathbb{R}^{H' \times W'}$ (e.g., 32×32), where a region consists of some grids. We utilize a binary masking matrix $\mathbf{M}_{mask}$ to capture the interactions between nearby grids. The weight of $\mathbf{M}_{mask}$ is set to 1 when the distance between two grids falls within a certain range, and 0 otherwise. Intuitively, all grids within a region are geographically correlated as they are relatively close to each other, leading us to set their masking weights to 1. In contrast, for regions, the masking weights are set to 1 only when the distance between the two regions where the grids are located is less than a threshold. As shown in the SLA module illustrated in Fig. 6, for the red region, there is cross-grid interaction within the region, and only the grids within the blue region engage in cross-region interaction.

In general, both cross-grid and cross-region interactions fundamentally entail interactions among grids. The grid partition introduced in the last section is simply for the purpose of illustrating how to construct the masking matrix more intuitively. After obtaining the masking matrix $\mathbf{M}_{mask}$, we define grid-level spatial attention. Specifically, the short-term local attention module maps $\mathbf{F}_s$ into three matrices by convolutional operations with 1×1 convolutions: the query $\mathbf{Q}_c \in \mathbb{R}^{N \times \tilde{C}}$, the key $\mathbf{K}_c \in \mathbb{R}^{N \times \tilde{C}}$, and the value $\mathbf{V}_c \in \mathbb{R}^{N \times C}$, where $N = H \times W$, $\tilde{C}$ and C are number of channels. Let $\mathbf{A}_c \in \mathbb{R}^{N \times N}$ be the attention map, and each element in $\mathbf{A}_c$ explicitly indicate the correlation between two grids in the input feature map. Finally, the spatio-temporal representation $\mathbf{F}_c$ is obtained by summing over $\mathbf{V}_c$ weighted by $\mathbf{A}_c$. For simplicity, we define the above short-term local attention module as follows, where $\mathbf{W}_c^Q$, $\mathbf{W}_c^K$ and $\mathbf{W}_c^V$ are projection matrices to be learned.

$$\begin{aligned}
&\mathbf{Q}_c = \mathbf{W}_c^Q \mathbf{F}_s, \mathbf{K}_c = \mathbf{W}_c^K \mathbf{F}_s, \mathbf{V}_c = \mathbf{W}_c^V \mathbf{F}_s \\
&\mathbf{A}_c = \text{softmax}\left(\left(\mathbf{Q}_c \left(\mathbf{K}_c\right)^T\right) \odot \mathbf{M}_{mask}\right) \\
&\mathbf{F}_c = \text{SLA}\left(\mathbf{Q}_c, \mathbf{K}_c, \mathbf{V}_c, \mathbf{M}_{mask}\right) = \mathbf{A_c} \mathbf{V}_c
\end{aligned} \tag{6}$$

2) Long-Term Global Attention Module (LGA): LGA module aims to capture long-range global spatio-temporal dependencies by considering that distant areas with similar functions may exhibit similar traffic flow patterns. Different from the input features of SLA module, three types of temporal features obtained by ConvLSTM components (i.e., $\mathbf{F}_c, \mathbf{F}_p, \mathbf{F}_q$), as well as external features $\mathbf{F}_{ext}$ will be fed into the LGA module for global feature extraction, denoted as $\mathbf{F}_l \in \mathbb{R}^{H \times W \times D}$, where $\mathbf{F}_l = \mathbf{F}_c + \mathbf{F}_p + \mathbf{F}_q + \mathbf{F}_{ext}$. In addition, the fine-grained spatial relations and temporal traffic patterns are encoded and stored in STMN. Therefore, LGA module incorporates the information of STMN to explicitly model long-range spatial and long-term temporal dependencies.

Given the key memory matrix $\mathbf{M}_k \in \mathbb{R}^{H \times W \times D_{km}}$ and value memory matrix $\mathbf{M}_v \in \mathbb{R}^{H \times W \times D_{vm}}$ in STMN, as well as obtained features $\mathbf{F}_l \in \mathbb{R}^{H \times W \times D}$, LGA aims to extract long-range spatial features and long-term temporal features from internal memories and external memories. However, despite the large number of grids in a city (e.g., 128×128), capturing the spatio-temporal dependencies at the grid level could result in inefficient computation. Hence, the goal of LGA is to effectively learn long-range spatial dependencies and long-term temporal features by reading essential information from the external memory network.

Instead of considering the fine-grained information about each grid, we first perform a downsampling operation from grid space to region space to obtain the semantic information, which is more friendly to capture global dependencies. Specifically, we adopt a convolutional layer to map the grid-level input of LGA into the region-level features:

$$\begin{aligned}
\mathbf{F}'_l &= Conv(\mathbf{F}_l) \in \mathbb{R}^{H' \times W' \times D} \\
\mathbf{M}'_k &= Conv(\mathbf{M}_k) \in \mathbb{R}^{H' \times W' \times D_{km}} \\
\mathbf{M}'_v &= Conv(\mathbf{M}_v) \in \mathbb{R}^{H' \times W' \times D_{vm}}
\end{aligned} \tag{7}$$

To model the semantic spatio-temporal dependencies about traffic flow patterns of regions with similar functions, the attention mechanism is utilized to read the long-range spatial information and long-term temporal information from the memory network. we map region-level features into new feature spaces to improve the model flexibility, where $\mathbf{Q}_m \in \mathbb{R}^{N \times \tilde{C}}$, $\mathbf{K}_m \in \mathbb{R}^{N \times \tilde{C}}$ and $\mathbf{V}_m \in \mathbb{R}^{N \times C}$ denote the new query matrix, key matrix and value matrix respectively, $M = H' \times W'$. Then, the attention weights $\mathbf{A}_m \in \mathbb{R}^{M \times M}$ between all region pairs are calculated, which could indicate the global spatial relation. Therefore, the long-term global spatio-temporal features $\mathbf{F}'_m$ could be obtained by calculating a weighted sum of the memory value of all regions in the city. Notably, to obtain fine-grained features corresponding to each grid, we further employ the N^2-$Normalization$ method [35] to map region-level features $\mathbf{F}'_m$ of $H' \times W'$ regions to grid-level features $\mathbf{F}_m$ of $H \times W$ grids.

$$\begin{aligned}
&\mathbf{Q}_m = \mathbf{W}_m^Q \mathbf{F}'_l, \mathbf{K}_m = \mathbf{W}_m^K \mathbf{M}'_k, \mathbf{V}_m = \mathbf{W}_m^V \mathbf{M}'_v \\
&\mathbf{A_m} = \text{softmax}\left(\mathbf{Q}_m \left(\mathbf{K}_m\right)^T\right) \\
&\mathbf{F}'_m = \text{LGA}\left(\mathbf{Q}_m, \mathbf{K}_m, \mathbf{V}_m\right) = \mathbf{A}_m \mathbf{V}_m \\
&\mathbf{F}_m = N^2\text{-}Normalization\left(\mathbf{F}'_m\right)
\end{aligned} \tag{8}$$

3) Fusion Mechanism: Some existing works [36] have proved that fusing multiple features can improve the performance of the model. Hence, we present a fusion mechanism to integrate two kinds of spatio-temporal features to enrich the representation ability of the model. Specifically, we utilize the neural gating technique to dynamically aggregate features extracted from two attention modules, which are capable of adapting to different regions and time periods by controlling the importance of the two types of spatio-temporal features. Given the short-term local features $\mathbf{F}_c$ and long-term global features $\mathbf{F}_m$, we could obtain the integrated feature presentation $\mathbf{F}_{final} \in \mathbb{R}^{H \times W \times D}$ by a weighted sum of two representations controlled by the fusion gate g_f. The process can be summarized

TABLE II
DATASET DESCRIPTION OF FIVE DATASETS

Dataset	TaxiNYC	TaxiDC	BikeNYC	BikeDC	TaxiBJ+
Grids	10×20	16×16	10×20	16×16	128×128
Start time	1/1/2015	5/1/2015	1/1/2017	1/1/2017	7/1/2013
End time	7/1/2015	1/1/2016	12/31/2017	12/31/2017	10/30/2013
Time interval	1 hour	1 hour	1 hour	1 hour	30 minutes
External factors	/	/	/	/	7

as:

$$\mathbf{F}_{final} = g_f \circ \mathbf{F}_c + (1 - g_f) \circ \mathbf{F}_m$$
$$g_f = \sigma\left(\mathbf{W}_{cg} * \mathbf{F}_c + \mathbf{W}_{mg} * \mathbf{F}_m + \mathbf{b}_g\right) \quad (9)$$

where $\mathbf{W}_{cg}, \mathbf{W}_{mg}, \mathbf{b}_g$ are projection matrices to be learned.

Finally, the predicted traffic volume at T time step $\hat{\mathbf{Y}}_T \in \mathbb{R}^{H \times W \times K}$, is obtained based on extracted spatio-temporal features $\mathbf{F}_{final}$, which is computed by :

$$\hat{\mathbf{Y}}_T = \tanh\left(Conv\left(\mathbf{F}_{final}\right)\right) \quad (10)$$

where $Conv$ is the convolution operation, $\tanh$ is a hyperbolic tangent that ensures the output values are between -1 and 1.

D. Optimization

We formulate the traffic volume prediction as a regression task, and propose an end-to-end approach to train the model by minimizing the following loss function:

$$\mathcal{L} = \mathcal{L}_r^p + \alpha \mathcal{L}_s^m + \beta \mathcal{L}_c^m + \gamma \|\Theta\|_F^2 \quad (11)$$

where Θ is the model parameter set, α, β and γ are trade-off parameters. $\mathcal{L}_r^p$ is the Mean Squared Error (MSE) to evaluate the prediction performance between prediction value $\hat{\mathbf{Y}}_T$ and ground truth $\mathbf{M}_T$. $\mathcal{L}_s^m$ and $\mathcal{L}_c^m$ are losses of the memory network, as introduced in the previous section. The final term regularizes all the model parameters to avoid over-fitting.

IV. EXPERIMENTS

In this section, we first provide an overview of the experimental settings, including datasets, baselines, metrics and implementations. Subsequently, we conduct a comprehensive evaluation of our proposed model's performance on five public datasets.

A. Experimental Settings

1) Datasets: We conduct experiments on five traffic datasets to evaluate the performance of ST-MAN. Table II details five real-world datasets. The first two datasets, TaxiNYC[1] and BikeNYC,[2] consist of about 6 million taxicab trip records and 8 million bike trip records in NewYork City. The next two datasets, TaxiDC[3] and BikeDC,[4] sources from trajectories of about 16 million taxicabs and 3 million bikes in Washington DC. The final dataset, TaxiBJ+ contains over 30 thousand taxicabs trajectories in Beijing .[5]

2) Baselines: We compare ST-MAN with four groups of baselines. Approaches of the first type are time series regression models, including Historical Average (HA) and Auto-Regressive Integrated Moving Average (ARIMA); Approaches of the second type are traditional regression models, including Linear Regression (LR) and Tree model; Approaches of the third type are classic DNN-based models, including Multiple Layer Perceptron (MLP), Convolutional Neural Network (CNN), Long-Short Term Memory Network (LSTM), and Convolutional LSTM (ConvLSTM); Approaches of the last type are state-of-the-art spatio-temporal models, including DMVST-Net, DeepST, ST-ResNet, STDN, DeepSTN+, SA-ConvLSTM and PDFormer.

- *HA:* Historical Average simply uses the average value of historical data to predict the future value.
- *ARIMA:* Auto-Regressive Integrated Moving Average predicts the future value by a linear combination of historical values and residual operations. The ARIMA model is trained separately for each target region in our experiments.
- *LR:* Linear Regression models the linear relationship between current observations and historical data. We build a global regression model based on historical data from all regions to calculate the predicted values.
- *Tree model:* It uses a tree structure to fit the complex correlations between data.
- *MLP:* We employ a fully connected network for traffic prediction task. The number of neurons in the hidden layer are 256, 128 and 64, respectively.
- *CNN:* Convolutional operations are used to model spatial dependencies in the neural network. The CNN model in the experiment consists of 3 convolutional layers, where the filter number is 64 and the size of the convolutional kernel is 3×3.
- *LSTM:* The Long Short-term Memory Network is built with 128 hidden units, and we select the previous 12 frames to predict the next frame.
- *ConvLSTM:* Convolutional LSTM replaces fully connected layers in LSTM units with convolutional layers to capture both temporal and spatial dependencies.
- *DMVST-Net* [37]: Deep Multi-View spatio-Temporal Network combines CNN and LSTM to model temporal and spatial relationships simultaneously. Spatial features are first extracted by two convolutional layers, which are then fed into a fully connected LSTM layer to extract temporal features.
- *DeepST* [13]: It is the first Deep-learning-based network for spatio-temporal prediction, which leverages the framework of convolutional neural networks to capture spatial and temporal dependencies based on three kinds of sequential data.
- *ST-ResNet* [16]: Deep Spatio-Temporal Residual Networks, a state-of-the-art spatio-temporal model, applies

[1]https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page
[2]https://www.citibikenyc.com/system-data
[3]https://opendata.dc.gov/search?q=taxi
[4]https://www.capitalbikeshare.com/system-data
[5]https://github.com/yoshall/UrbanFM

TABLE III
PERFORMANCE COMPARISON OF DIFFERENT METHODS IN TERMS OF RMSE AND MAE

Datasets	**TaxiNYC**		**TaxiDC**		**BikeNYC**		**BikeDC**		**TaxiBJ+**	
Metrics	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
HA	10.0573	5.1072	33.6695	14.1543	30.8275	15.6316	12.445	7.0751	26.5702	18.8546
ARIMA	9.0998	5.0213	28.6089	10.7037	23.4854	11.6232	10.7253	6.4053	22.8099	15.7216
LR	7.4235	4.5586	13.0652	6.4329	15.2656	7.9387	8.1601	4.5318	15.4554	9.9221
Tree	8.8234	5.0987	14.7149	7.0631	16.1438	8.4188	8.0506	4.8498	16.7589	10.8695
MLP	9.8668	5.4152	19.2622	7.9245	11.8643	6.2067	5.5737	3.4883	9.4351	4.6531
CNN	5.1658	2.4539	5.628	1.8363	7.2354	3.4372	2.2232	0.8729	6.7147	3.0062
FC-LSTM	5.6171	2.6308	6.5573	2.0517	8.5537	3.249	2.6236	0.8415	6.961	3.2109
ConvLSTM	4.5731	2.0385	4.9085	1.4229	5.8726	2.8113	1.6271	0.5915	6.2903	3.0024
DMVST-Net	5.3104	2.7974	6.0765	2.666	6.4303	2.7589	1.7522	0.6548	6.6276	3.1431
DeepST	4.5453	2.0262	5.624	1.8486	5.5124	2.715	1.6292	0.6833	6.6942	3.1975
ST-ResNet	4.4784	2.3474	5.5259	1.8994	5.3987	2.6863	1.4385	0.5343	6.6561	3.1331
STDN	4.3754	1.9862	5.3398	1.6181	5.4237	2.7146	1.4962	0.6032	6.4877	3.0007
DeepSTN+	4.2504	2.2154	5.0929	1.8931	5.0076	2.3536	1.2872	0.4872	5.6976	3.0612
SAConvLSTM	4.2646	1.8785	4.9074	1.441	5.1384	2.4891	1.3471	0.5108	5.5434	2.7641
PDFormer	4.2369	1.9475	4.8228	1.5389	4.9562	2.3047	1.2939	0.4903	5.5075	2.6828
ST-MAN	**4.1396**	**1.8272**	**4.5205**	**1.4052**	**4.8264**	**2.1524**	**1.2397**	**0.4751**	**5.3876**	**2.5264**

the residual mechanism to further improve the general convolution framework in DeepST.

- *STDN* [21]: Spatio-Temporal Dynamic Network models temporal relationships over long periods of time through an attentional mechanism.
- *DeepSTN+* [17]: It improves the fusion mechanism in ST-ResNet, and models a large range of spatial dependencies using the ConvPlus module.
- *SA-ConvLSTM* [38]: Self-Attention ConvLSTM combines the attention mechanism and ConvLSTM to model a large range of spatial dependencies.
- *PDFormer* [39]: The state-of-the-art method for urban flow prediction, which captures dynamic temporal and spatial dependencies by self-attention modules.

3) Metrics: We evaluate the model performance based on two common metrics for traffic volume prediction: Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE). Smaller values of RMSE and MAE indicate better model performances.

4) Implementations: We implement ST-MAN and all baselines by PyTorch 1.8 and Python 3.8. In our experiment, we divided New York City, Washington DC, and Beijing into grid cells of sizes 10×20, 16×16, and 128×128, respectively. We also defined region cells of sizes 5×10, 8×8 and 16×16 for the corresponding cities. For each dataset, we choose the last 20% data as the testing data, and all data before that for training and validation. For temporal closeness, period, and trend, we set lengths of three segments as follows: $l_c \in \{3, 4, 5\}$, $l_p \in \{3, 4, 5\}$ and $l_q \in \{2, 3, 4\}$. Moreover, the dimensions (i.e., D_{km} and D_{vm}) of key and value matrices are set to 32 and 64. All convolution layers use 64 filters, and the size of the kernel is set as 3×3. The depth of ConvLSTM layers is searched over $\{2, 3, 4\}$. The hidden dimension D of attention modules is searched over $\{16, 32, 64, 128\}$, while the depth of attention layers is searched over $\{1, 2, 3, 4\}$. Temperature parameter τ is fixed to 1.0, and trade-off parameters α, β, γ in the loss function are set as 0.05, 0.1 and 0.001. In the training process, the model is optimized by Adam, the learning rate is 10^{-4}, and the batch size is set to 32.

B. Experimental Results

1) Performance Comparison: To evaluate our proposed model, we compare the performance of different models over five datasets, where TaxiBJ+ is a fine-grained dataset with 128×128 grid cells. The results are shown in Table III. From Table III, we find that ST-MAN achieves an improvement over all the baselines on five real-world datasets with two evaluation metrics, which demonstrates the superiority of our proposed model. We have the following main observations:

- It can be seen that DNN-based prediction models significantly outperform time series regression models and traditional regression methods, because they have limited ability to capture complicated non-linear spatial and temporal relations.
- By comparing state-of-the-art spatio-temporal models against traditional DNN models, including MLP, CNN, FC-LSTM and ConvLSTM, we can see that spatio-temporal models achieve better performance. The main reason is that most spatio-temporal models take into account the characteristics of spatio-temporal data to design corresponding modules. Take ConvLSTM for example, though ConvLSTM extends FC-LSTM to have convolutional structures to better capture spatio-temporal correlations, it still fails to capture long-range spatial dependencies and long-term temporal dependencies. This emphasizes that the particular module should be designed to model complex spatio-temporal correlations to improve the representation ability of the prediction model.
- ST-MAN obtains much better performance than some state-of-the-art spatio-temporal prediction models over

TABLE IV
ABLATION ANALYSIS ON FIVE DATASETS

Datasets	TaxiNYC		TaxiDC		BikeNYC		BikeDC		TaxiBJ+	
Metrics	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
ST-MAN w/o ConvLSTM	4.1952	2.0028	5.5153	1.7381	5.0138	2.3182	1.5446	0.6792	5.6374	2.8183
ST-MAN w/o STMN	4.4257	2.2412	5.3895	1.7637	5.2064	2.3236	1.5732	0.5643	5.6694	2.9095
ST-MAN w/o MAN	4.1881	1.9269	5.0817	1.5258	4.9607	2.2017	1.3856	**0.4732**	5.8325	3.0816
ST-MAN	**4.1396**	**1.8272**	**4.5205**	**1.4052**	**4.8264**	**2.1524**	**1.2397**	0.4751	**5.3876**	**2.5264**

five datasets. The reason is that CNN-based models (i.e., DeepST, ST-ResNet, DeepSTN+) mainly focus on capturing long-range spatial dependencies by convolutional layers, but they neglect long-term temporal sequential dynamics. DMVST-Net and STDN combine the local CNN and RNN to learn spatial dependencies and temporal dependencies respectively. However, the dynamic correlations between spatial and temporal factors are ignored. In addition, these models are insufficient to capture long-range spatial dependencies because CNN-based models need to stack too many convolutional layers, which cannot be deployed in most spatio-temporal tasks. Our proposed ST-MAN is capable of capturing short-term local spatio-temporal dependencies and long-term global spatio-temporal dependencies based on the memory network and multi-level attention network.

- We can observe that ST-MAN outperforms SAConvLSTM in all five prediction tasks. Although SAConvLSTM introduces the attention mechanism into ConvLSTM to extract long-range spatial dependencies, it still is insufficient to capture long-term temporal dependencies. In addition, although PDFormer utilizes the transformer structure to capture dynamic temporal and spatial features, it still is insufficient to capture long-term temporal dependencies. In addition, it suffers from significant redundant computation when dealing with a large number of grids in the city. Traditional self-attention mechanisms compute attention scores between all grids, resulting in a significant computational cost. Instead of considering the fine-grained information about each grid, we perform a downsampling operation from grid space to region space to obtain the semantic information, which is more friendly to capture global dependencies with lower computational cost. So our ST-MAN can effectively learn long-range spatial dependencies and long-term temporal features by reading essential information from the external memory network. In addition, we also introduce an external component to generate location-aware influence from external factors based on grid-level spatial embeddings. In general, the improvement of ST-MAN comes not only from the attention mechanism, but also from the unique structure of the external memory network.

2) Ablation Study: To study the contributions of different components in ST-MAN to the performance gain, we evaluate three variants of our model to conduct an ablation study:

- *ST-MAN w/o ConvLSTM* removes the ConvLSTM component in the stage of feature map construction.
- *ST-MAN w/o STMN* removes the external spatio-temporal memory network from ST-MAN.
- *ST-MAN w/o MAN* is the base prediction model, which only utilizes the ConvLSTM to extract features.

The results are summarized in Table IV. We can observe that the performance improves by incorporating STMN and MAN, which demonstrates that the memory network is able to enhance the expressiveness of the model and improve the performance. First, it can be observed that the performance of ST-MAN w/o ConvLSTM is worse than ST-MAN, indicating the benefit of utilizing the feature map obtained through ConvLSTM for modeling complex spatio-temporal dependencies. Second, the performance of ST-MAN without the attention module is also inferior to ST-MAN, and it exhibits variations across different datasets. The main reason for this discrepancy lies in the distinct traffic flow distributions in different cities. In addition, we find that removing the key and memory unit in STMN degrades the performance to some extent. The major reason is that the key memory unit provides information on global spatial correlation by spatial embedding, and ignoring spatial dependencies weakens the feature representation ability of the model. In a nutshell, the good performance of ST-MAN demonstrates the effectiveness of our design in introducing the external memory network to capture long-range and long-term spatio-temporal dependencies.

3) Effect of Attention Mechanism: This experiment investigates the effectiveness of our proposed attention module to model spatio-temporal features. We implement three simplified versions of ST-MAN to study whether the attention module effectively captures spatio-temporal features. Specifically, ST-MAN w/o SGLA removes the short-term cross-grid local attention module from the multi-level attention network, ST-MAN w/o SRLA removes the short-term cross-region local attention from the multi-level attention network, and ST-MAN w/o LGA removes the long-term cross-region global attention from spatio-temporal memory network, which only uses ConvLSTM for feature extraction.

The results are shown in Table V. We can see that ST-MAN achieves better performance than other variants, which demonstrates that our proposed attention mechanism can effectively extract spatio-temporal features and improve the prediction performance of the model. In addition, it can be observed that ST-MAN w/o LGA performs the worst, indicating that both long-range and short-range spatio-temporal features are useful for traffic volume prediction. Furthermore, it can be observed that the performance of ST-MAN w/o SRLA is worse than ST-MAN w/o SGLA, indicating the significance of cross-region traffic flow transitions.

TABLE V
EFFECT OF ATTENTION MODULES ON FIVE DATASETS

Datasets	TaxiNYC		TaxiDC		BikeNYC		BikeDC		TaxiBJ+	
Metrics	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
ST-MAN w/o SGLA	4.1556	2.0133	4.9256	1.6048	4.9819	2.223	1.2975	0.4923	5.7185	3.0278
ST-MAN w/o SRLA	4.1803	2.0849	5.1267	1.6163	5.0736	2.275	1.3256	0.5162	5.9274	3.1246
ST-MAN w/o LGA	4.4257	2.2412	5.3895	1.7637	5.2064	2.3236	1.5732	0.5643	5.6694	2.9095
ST-MAN	**4.1396**	**1.8272**	**4.5205**	**1.4052**	**4.8264**	**2.1524**	**1.2397**	**0.4751**	**5.3876**	**2.5264**

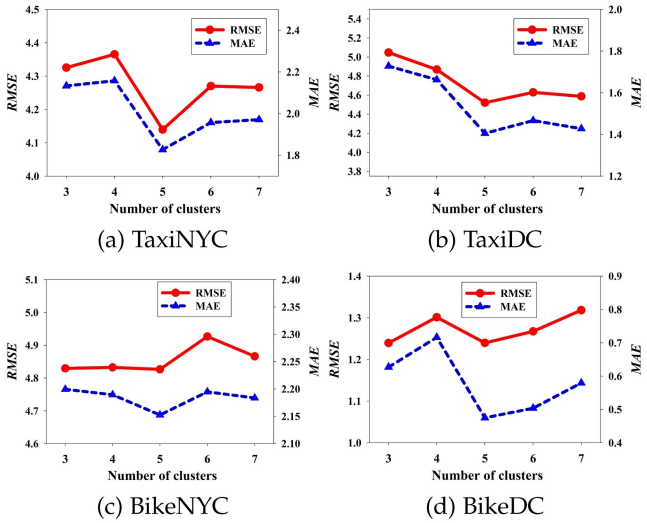


Fig. 8. Effect of memory cluster number.

One possible reason could be that people's mobility within the city has increased due to the development of transportation, making cross-region interactions more important.

4) Effect of Memory Cluster Number: To obtain prior knowledge about different areas of the city from spatio-temporal memory network, we classify areas into different types, where the number of potential urban functional areas obtained by clustering is a major parameter of ST-MAN model. Therefore, we investigate the effect of the number of different functional area categories in the value memory matrix on ST-MAN model.

In this experiment, the range of the number of clustered functional areas K is set to [3,4,5,6,7]. Fig. 8 shows the prediction performance of ST-MAN model over four datasets with increasing K values. From Fig. 8, we can observe that ST-MAN achieves the best performance when the value of K is set to 5. One possible reason could be that most of the cities are already well developed and the functional areas of the cities are stabilizing, thus the disappearance or emergence of certain types of functional areas may not occur in the short term. For example, business area, residential area, administrative area, college area, and scenic area could be five primary urban functional areas in most cities.

V. RELATED WORK

In this section, we briefly review some related work focusing on urban traffic prediction and the memory network.

A. Urban Traffic Prediction

With the increasing availability of large-scale traffic data, urban traffic prediction has been an appealing domain in urban computing. A lot of researches have been made to predict the potential traffic volume based on historical traffic data. Classic works view traffic prediction as a time series prediction problem, and apply time series approaches to model temporal patterns. For example, the historical average model (HA) simply uses the average value of historical data to predict future value. The Auto-Regressive Integrated Moving Average (ARIMA) model [10], one of the typical times series models, predicts the future value by a linear combination of historical values and residual operations. However, they have a limited ability to capture complicated non-linear temporal relations, and the spatial information is ignored to model spatial dependencies.

In recent years, deep learning has achieved promising performance in a lot of tasks [40], [41]. Particularly, CNNs have been successfully used to extract spatial features in the computer vision field [42], and RNNs encode temporal information for sequence learning tasks by embedding historical sequential records into a hidden state vector [43]. Many researchers have started to utilize deep learning approaches to solve traffic prediction problems [44], [45], which have shown superior performance compared with traditional methods. DeepST [13] is the first CNN-based network for spatio-temporal prediction, and it leverages the framework of convolutional neural networks to capture spatial dependencies. ST-ResNet [16], one of the representative spatio-temporal models, improves the framework of DeepST by introducing the residual mechanism to enable deep structures, which could model large citywide dependencies. [17] proposes a context-aware spatio-temporal neural network DeepSTN+, which applies the ConvPlus structure to capture long-range spatial correlations in different regions. Furthermore, many works [46], [47] utilize the graph convolution neural networks to model the trajectories and road networks for traffic prediction. In addition, a lot of works combine the benefits of CNNs with RNNs to model both spatial and temporal correlations. [37] proposes a Deep Multi-View spatio-Temporal Network (DMVST-Net) to model both spatial and temporal relations for taxi demand prediction. [21] proposes a spatio-Temporal Dynamic Network (STDN) to solve the problem of dynamic temporal shifting in real-world cases.

These existing works have limited prediction ability due to insufficient spatio-temporal features. On the one hand, stacking many convolutional layers in CNN-based networks to capture long-range spatial dependencies could result in high computational costs and optimization difficulties. On the other hand, the

historical data is compressed into a hidden state to encode temporal dependencies in RNN-based networks that have limited capability to express long-term temporal patterns. By contrast, we aim to model long-range spatial dependencies and long-term temporal dependencies simultaneously in urban traffic prediction.

B. Memory Network

Memory network is a recurrent attention model, which is designed to solve the problem that RNNs have difficulty in performing memorization. It leverages an external memory component to read and write long-term memory in memory slots, which provides the additional representation of knowledge to increase the model capacity [23].

Recently, memory network has been applied successfully to many domains, such as question answering, and sequential recommendation. [24] proposes a memory-augmented neural network (MANN) based on collaborative filtering to store and update users' interests for recommendation. Specifically, two external user memory matrices are leveraged to encode item-level and feature-level information. [48] proposes memory-to-sequence (Mem2Seq) model for end-to-end task-oriented dialog systems, which integrates the multi-hop mechanism and external memory to process long sequences. Besides the original memory network, Key-Value Memory Networks (KV-MN) [49] generalizes the memory component that can integrate with other knowledge sources, which is more flexible to encode prior knowledge to enhance the model. [50] first proposes the Key-Value Memory Networks to make reading documents more viable. Specifically, the memory component is structured as (key, value) pairs, and the addressing stage and reading stage are based on key memory and value memory respectively. [36] proposes a Dynamic Memory-based Attention Network (DMAN) for long sequential recommendation, which includes a set of memory blocks to store the long-term interests of users. [51] proposes a topic-enhanced memory network (TEMN) for personalized point-of-interest recommendations. Particularly, the memory network component is employed to encode users' historical check-in records and capture the local relationships between different regions.

Above all works mainly focus on enabling the memory network to store sequential information from hidden vectors, and there are few works toward spatio-temporal prediction because spatial information is ignored in the memory network. Our work is distinct from all the above works, we introduce a new perspective for spatio-temporal prediction, where the memory network is utilized to encode and memorize both spatial information and temporal information to enhance the expressive power of spatio-temporal prediction model. To the best of our knowledge, we are the first to employ the external memory network for spatio-temporal prediction tasks, and study the long-range and long-term spatio-temporal correlations with complex traffic states.

VI. Conclusion

This paper proposes a novel traffic prediction approach, spatio-temporal memory augmented multi-level attention Network, entitled ST-MAN. To the best of our knowledge, this is the first try that the external memory network is introduced for spatio-temporal prediction tasks to enrich the expressiveness of the prediction model, by encoding and memorizing fine-grained spatial information and temporal patterns. Furthermore, we combine the benefits of spatio-temporal memory network and multi-level attention network to explicitly model long-range spatial dependencies and long-term temporal dependencies simultaneously. Our proposed ST-MAN is evaluated over five real-world datasets, and the experimental results show that ST-MAN is more effective compared with state-of-the-art baselines. In our future work, we plan to utilize the memory network to learn domain-invariant spatio-temporal patterns, and transfer valuable knowledge from the data-rich city to the data-sparse city to improve the performance.

References

[1] C. Zheng et al., "Spatio-temporal joint graph convolutional networks for traffic forecasting," *IEEE Trans. Knowl. Data Eng.*, early access, Jun. 13, 2023, doi: 10.1109/TKDE.2023.3284156.

[2] X. Ouyang, Y. Yang, W. Zhou, Y. Zhang, H. Wang, and W. Huang, "Citytrans: Domain-adversarial training with knowledge transfer for spatio-temporal prediction across cities," *IEEE Trans. Knowl. Data Eng.*, early access, Jun. 07, 2023, doi: 10.1109/TKDE.2023.3283520.

[3] J. Han, H. Liu, H. Xiong, and J. Yang, "Semi-supervised air quality forecasting via self-supervised hierarchical graph neural network," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 5, pp. 5230–5243, May 2023.

[4] F. Li et al., "Dynamic graph convolutional recurrent network for traffic prediction: Benchmark and solution," *ACM Trans. Knowl. Discov. Data*, vol. 17, no. 1, pp. 1–21, 2023.

[5] J. Ji, J. Wang, Z. Jiang, J. Jiang, and H. Zhang, "STDEN: Towards physics-guided neural networks for traffic flow prediction," in *Proc. AAAI Conf. Artif. Intell.*, 2022, pp. 4048–4056.

[6] J. Qi, Z. Zhao, E. Tanin, T. Cui, N. Nassir, and M. Sarvi, "A graph and attentive multi-path convolutional network for traffic prediction," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 7, pp. 6548–6560, Jul. 2023.

[7] H. Lin, R. Bai, W. Jia, X. Yang, and Y. You, "Preserving dynamic attention for long-term spatial-temporal prediction," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2020, pp. 36–46.

[8] M. Liang, R. W. Liu, Y. Zhan, H. Li, F. Zhu, and F.-Y. Wang, "Fine-grained vessel traffic flow prediction with a spatio-temporal multigraph convolutional network," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 12, pp. 23694–23707, Dec. 2022.

[9] R. Jiang et al., "DeepCrowd: A deep model for large-scale citywide crowd density and flow prediction," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 1, pp. 276–290, Jan. 2023.

[10] 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.

[11] J. Ji et al., "Spatio-temporal self-supervised learning for traffic flow prediction," in *Proc. AAAI Conf. Artif. Intell.*, 2023, pp. 4356–4364.

[12] H. Yao, Y. Liu, Y. Wei, X. Tang, and Z. Li, "Learning from multiple cities: A meta-learning approach for spatial-temporal prediction," in *Proc. World Wide Web Conf.*, 2019, pp. 2181–2191.

[13] J. Zhang, Y. Zheng, D. Qi, R. Li, and X. Yi, "DNN-based prediction model for spatio-temporal data," in *Proc. 24th ACM SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.*, 2016, pp. 1–4.

[14] Y. Liu, C. Gong, L. Yang, and Y. Chen, "DSTP-RNN: A dual-stage two-phase attention-based recurrent neural network for long-term and multivariate time series prediction," *Expert Syst. Appl.*, vol. 143, 2020, Art. no. 113082.

[15] D. Chen, H. Wang, and M. Zhong, "A short-term traffic flow prediction model based on autoencoder and GRU," in *Proc. 12th Int. Conf. Adv. Comput. Intell.*, 2020, pp. 550–557.

[16] J. Zhang, Y. Zheng, and D. Qi, "Deep spatio-temporal residual networks for citywide crowd flows prediction," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 1655–1661.

[17] Z. Lin, J. Feng, Z. Lu, Y. Li, and D. Jin, "DeepSTN : Context-aware spatial-temporal neural network for crowd flow prediction in metropolis," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 1020–1027.
[18] W. Luo, Y. Li, R. Urtasun, and R. Zemel, "Understanding the effective receptive field in deep convolutional neural networks," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst.*, 2016, pp. 4905–4913.
[19] J. Zhang, Y. Zheng, J. Sun, and D. Qi, "Flow prediction in spatio-temporal networks based on multitask deep learning," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 3, pp. 468–478, Mar. 2020.
[20] J. Sun, J. Zhang, Q. Li, X. Yi, Y. Liang, and Y. Zheng, "Predicting citywide crowd flows in irregular regions using multi-view graph convolutional networks," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 5, pp. 2348–2359, May 2022.
[21] H. Yao, X. Tang, H. Wei, G. Zheng, and Z. Li, "Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 5668–5675.
[22] Y. Wang, J. Zhang, H. Zhu, M. Long, J. Wang, and P. S. Yu, "Memory in memory: A predictive neural network for learning higher-order non-stationarity from spatiotemporal dynamics," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 9154–9162.
[23] J. Weston, S. Chopra, and A. Bordes, "Memory networks," 2014, *arXiv:1410.3916*.
[24] X. Chen et al., "Sequential recommendation with user memory networks," in *Proc. 11th ACM Int. Conf. Web Search Data Mining*, 2018, pp. 108–116.
[25] A. Kumar et al., "Ask me anything: Dynamic memory networks for natural language processing," in *Proc. Int. Conf. Mach. Learn.*, PMLR, 2016, pp. 1378–1387.
[26] J. Huang, W. X. Zhao, H. Dou, J.-R. Wen, and E. Y. Chang, "Improving sequential recommendation with knowledge-enhanced memory networks," in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2018, pp. 505–514.
[27] C. Ma, L. Ma, Y. Zhang, J. Sun, X. Liu, and M. Coates, "Memory augmented graph neural networks for sequential recommendation," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 5045–5052.
[28] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 802–810.
[29] Y. Liang et al., "Fine-grained urban flow prediction," in *Proc. Web Conf.*, 2021, pp. 1833–1845.
[30] K. W. Church, "Word2vec," *Natural Lang. Eng.*, vol. 23, no. 1, pp. 155–162, 2017.
[31] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, PMLR, 2020, pp. 1597–1607.
[32] P. Khosla et al., "Supervised contrastive learning," 2020, *arXiv: 2004.11362*.
[33] A. Likas, N. Vlassis, and J. J. Verbeek, "The global k-means clustering algorithm," *Pattern Recognit.*, vol. 36, no. 2, pp. 451–461, 2003.
[34] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.
[35] Y. Liang et al., "UrbanFM: Inferring fine-grained urban flows," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2019, pp. 3132–3142.
[36] Q. Tan et al., "Dynamic memory based attention network for sequential recommendation," 2021, *arXiv:2102.09269*.
[37] H. Yao et al., "Deep multi-view spatial-temporal network for taxi demand prediction," in *Proc. AAAI Conf. Artif. Intell.*, 2018, pp. 2588–2595.
[38] Z. Lin, M. Li, Z. Zheng, Y. Cheng, and C. Yuan, "Self-attention ConvLSTM for spatiotemporal prediction," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 11531–11538.
[39] J. Jiang, C. Han, W. X. Zhao, and J. Wang, "PDFormer: Propagation delay-aware dynamic long-range transformer for traffic flow prediction," in *Proc. AAAI Conf. Artif. Intell.*, 2023, pp. 4365–4373.
[40] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1440–1448.
[41] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system: A survey and new perspectives," *ACM Comput. Surv.*, vol. 52, no. 1, pp. 1–38, 2019.
[42] J. Gu et al., "Recent advances in convolutional neural networks," *Pattern Recognit.*, vol. 77, pp. 354–377, 2018.
[43] Y. Yu, X. Si, C. Hu, and J. Zhang, "A review of recurrent neural networks: Lstm cells and network architectures," *Neural Comput.*, vol. 31, no. 7, pp. 1235–1270, 2019.
[44] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with Big Data: A deep learning approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 865–873, Apr. 2015.
[45] R. Vinayakumar, K. Soman, and P. Poornachandran, "Applying deep learning approaches for network traffic prediction," in *Proc. Int. Conf. Adv. Comput., Commun. Inform.*, 2017, pp. 2353–2358.
[46] H. Peng et al., "Spatial temporal incidence dynamic graph neural networks for traffic flow forecasting," *Inf. Sci.*, vol. 521, pp. 277–290, 2020.
[47] M. Li and Z. Zhu, "Spatial-temporal fusion graph neural networks for traffic flow forecasting," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 4189–4196.
[48] A. Madotto, C.-S. Wu, and P. Fung, "Mem2Seq: Effectively incorporating knowledge bases into end-to-end task-oriented dialog systems," 2018, *arXiv: 1804.08217*.
[49] J. Zhang, X. Shi, I. King, and D.-Y. Yeung, "Dynamic key-value memory networks for knowledge tracing," in *Proc. 26th Int. Conf. World Wide Web*, 2017, pp. 765–774.
[50] A. Miller, A. Fisch, J. Dodge, A.-H. Karimi, A. Bordes, and J. Weston, "Key-value memory networks for directly reading documents," 2016, *arXiv:1606.03126*.
[51] X. Zhou, C. Mascolo, and Z. Zhao, "Topic-enhanced memory networks for personalised point-of-interest recommendation," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2019, pp. 3018–3028.

Yan Liu received the ME and PhD degrees in computer science from the School of Computer Science, Northwestern Polytechnical University, Xi'an, China, in 2017 and 2022, respectively. She is currently a post-doc researcher with Peking University, Beijing, China. Her research interests include urban computing, mobile computing, spatio-temporal prediction, and recommendation.

Bin Guo (Senior Member, IEEE) received the PhD degree in computer science from Keio University, Japan, in 2009, and then was a postdoc researcher with Institut Telecom SudParis, France. He is a professor with Northwestern Polytechnical University, Xi'an, China. His research interests include ubiquitous computing, mobile crowd sensing, and HCI. He has served as an associate editor of the *IEEE Communications Magazine* and the *IEEE Transactions on Human-Machine-Systems*, the guest editor of the *ACM Transactions on Intelligent Systems and Technology* and the *IEEE Internet of Things*, the general co-chair of IEEE UIC'15, and the program chair of IEEE CPSCom'16, ANT'14, and UIC'13.

Jingxiang Meng is currently an engineer with the Industrial and Commercial Bank of China, Xi'an, China. His research interests include data mining and machine learning.

Daqing Zhang (Fellow, IEEE) received the PhD degree from the University of Rome "La Sapienza", Italy, in 1996. He is a chair professor with the Key Laboratory of High Confidence Software Technologies (Ministry of Education), School of Computer Science, Peking University, China, and Telecom Sud-Paris, IP Paris, France. His research interests include context-aware computing, urban computing, mobile computing, Big Data analytics, pervasive elderly care, etc. He has authored more than 300 technical papers in leading conferences and journals. He was the general or program chair for more than 17 international conferences, giving keynote talks at more than 20 international conferences. He is the associate editor for the *IEEE Pervasive Computing*, *ACM Transactions on Intelligent Systems and Technology*, and the *Proceeding of ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*. He is the winner of the Ten-Years CoMoRea Impact Paper Award at IEEE PerCom 2013 and Ten-Years Most Influential Paper Award at IEEE UIC 2019, the Honorable Mention Award at ACM UbiComp 2015 and 2016, and the Distinguished Paper Award at ACM UbiComp 2021.

Zhiwen Yu (Senior Member, IEEE) received the PhD degree of engineering in computer science and technology from Northwestern Polytechnical University, Xi'an, China, in 2005, where he is currently working as a professor. He has worked as a research fellow with the Academic Center for Computing and Media Studies, Kyoto University, Japan, from 2007 to 2009, and a post-doctoral researcher with the Information Technology Center, Nagoya University, Japan, in 2006-2007. His research interests include pervasive computing, context-aware systems, human-computer interaction, mobile social networks, and personalization. He is the associate editor or editorial board member for the *IEEE Communications Magazine*, the *IEEE Transactions on Human-Machine Systems*, *Personal and Ubiquitous Computing*, and *Entertainment Computing* (Elsevier). He was the general chair of IEEE CPSCom'15, and IEEE UIC'14. He served as a vice program chair of PerCom'15, the program chair of UIC'13, and the workshop chair of UbiComp'11. He is a member of the ACM, and a council member of the China Computer Federation (CCF).