ComS2T: A Complementary Spatiotemporal Learning System for Data-Adaptive Model Evolution

Zhengyang Zhou, *Member, IEEE*, Qihe Huang, Binwu Wang, Jianpeng Hou, Kuo Yang, Yuxuan Liang, *Member, IEEE*, Yu Zheng, *Fellow, IEEE*, and Yang Wang, *Senior Member, IEEE*

***Abstract*—Spatiotemporal (ST) learning has become a crucial technique to enable smart cities and sustainable urban development. Current ST learning models capture the heterogeneity via various spatial convolution and temporal evolution blocks. However, rapid urbanization leads to fluctuating distributions in urban data and city structures, resulting in existing methods suffering generalization and data adaptation issues. Despite efforts, existing methods fail to deal with newly arrived observations, and the limitation of those methods with generalization capacity lies in the repeated training that leads to inconvenience, inefficiency and resource waste. Motivated by complementary learning in neuroscience, we introduce a prompt-based complementary spatiotemporal learning termed ComS2T, to empower the evolution of models for data adaptation. We first disentangle the neural architecture into two disjoint structures, a stable neocortex for consolidating historical memory, and a dynamic hippocampus for new knowledge update. Then we train the dynamic spatial and temporal prompts by characterizing distribution of main observations to enable prompts adaptive to new data. This data-adaptive prompt mechanism, combined with a two-stage training process, facilitates fine-tuning of the neural architecture conditioned on prompts, thereby enabling efficient adaptation during testing. Extensive experiments validate the efficacy of ComS2T in adapting various spatiotemporal out-of-distribution scenarios while maintaining effective inferences.**

***Index Terms*—Spatiotemporal learning, complementary learning system, OOD generalization, urban computing.**

Received 29 February 2024; revised 12 May 2025; accepted 1 June 2025. Date of publication 5 June 2025; date of current version 5 September 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 12227901 and Grant 62402414, in part by the Natural Science Foundation of Jiangsu Province under Grant BK.20240460, and in part by the State Key Laboratory of Resources and Environmental Information System. This work was also supported in part by Guangzhou Municipal Science and Technology Project under Grant 2023A03J0011. Recommended for acceptance by X. Li. *(Zhengyang Zhou and Qihe Huang contributed equally to this work.) (Corresponding author: Yang Wang.)*

Zhengyang Zhou and Yang Wang are with the University of Science and Technology of China, Hefei 230026, China, and also with the Suzhou Institute for Advanced Research, University of Science and Technology of China, Suzhou 215000, China (e-mail: zzy0929@ustc.edu.cn; angyan@ustc.edu.cn).

Qihe Huang, Binwu Wang, Jianpeng Hou, and Kuo Yang are with the University of Science and Technology of China, Hefei 230026, China (e-mail: hqh@mail.ustc.edu.cn; wbw2024@ustc.edu.cn; enterpr1se@mail.ustc.edu.cn; yangkuo@mail.ustc.edu.cn).

Yuxuan Liang is with the Hong Kong University of Science and Technology, Guangzhou 511458, China (e-mail: yuxliang@outlook.com).

Yu Zheng is with JD.COM, JD Intelligent Cities Research, Beijing 100176, China (e-mail: msyuzheng@outlook.com).

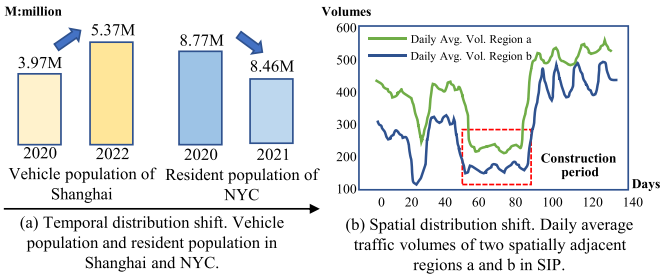
The code is available on https://github.com/hqh0728/ComS2T.

Digital Object Identifier 10.1109/TPAMI.2025.3576805

I. Introduction

SPATIOTEMPORAL (ST) learning, which is inherited from spatial learning [1], [2], [3] and equipped with temporal tendency extractor [4], has become a pivotal technique to improve the quality of urban life and the intelligence of cities. Current spatiotemporal forecasting models usually incorporate various spatial convolution blocks and temporal dependence extractors to achieve predictions, enabling diverse multi-variate urban series forecasting [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], including traffic conditions [7], [16], [17], [18], [19], natural climates [20], [21], [22], [23], as well as key indexes of environments [24], [25], [26], [27]. Despite prosperity, most existing methods assume that training and testing data are both independent and identically distributed (i.i.d.) where the principle does not hold in real-world scenarios. Actually, urban spatiotemporal elements tend to expand and increase with urbanization and evolution of cities. In Fig. 1, we take human mobility prediction as an instance. From a macro perspective, the vehicle population of Shanghai increased from 3.97 million in 2020 to 5.37 million in 2022, whereas the demographic population of NYC decreased from 8.77 million to 8.46 million from 2020 to 2021. From a microscopic perspective, if a region experiences the construction of a shopping mall, the mobility intensity will decrease suddenly during the construction period, followed by an increase after construction. As a result, the shifts regarding temporal distribution and urban structures pose out-of-distribution (OOD) challenges on respective temporal and spatial perspectives to current ST models. Therefore, a data-adaptive spatiotemporal learning framework with timely model updates is highly needed.

Although the majority of spatiotemporal learning methods fail to adapt their models to new OOD instances, learning on graphs with OOD settings has increasingly raised the attention of researchers [28], [29], [30], [31]. In general, environment is a fundamental concept in OOD learning where researchers can explicitly capture the invariance across environments for transfer. In our spatiotemporal learning scheme, we can inherit the environment concept and classify spatiotemporal environments into temporal aspect and local structures. To address temporal distribution shifts, AdaRNN defines covariate shifts in time series and designs distribution characterization to aggregate previous series for weighted prediction [32]. Following



(a) Temporal distribution shift. Vehicle population and resident population in Shanghai and NYC.

(b) Spatial distribution shift. Daily average traffic volumes of two spatially adjacent regions a and b in SIP.

Fig. 1. Examples of urban dynamics along city evolution.

AdaRNN, CauSTG reflects complex spatial-temporal dependencies by learnable parameters and takes invariant learning from causal perspective as a priori [33]. Unfortunately, CauSTG only captures stable neural structure yet fails to improve the dynamic structure for data adaptation. With respect to emerging urban structures, pioneering works take continuous learning into consideration. TrafficStream updates the neural structure by identifying the most dissimilar new nodes and consolidates historical information with experience reply [34], while PECPM manages the ST pattern bank with newly involved nodes, reducing memory burdens [35]. However, experience-reply or memory-based solutions require unlimited data space and multiple model retraining, leading to increased computational and storage space requirements. Modeling data distributions from environments [36], [37] can actually empower generalization by imitating the perturbation on features and extending the boundary of training set thus enlarging the receptive field of models. Nevertheless, they still fall short into a closed training set even with augmented samples, leading to the difficulty of addressing new instances with emerging patterns and shapes.

Therefore, we can summarize two serious problems in existing solutions to OOD challenges. First, these methods explicitly model environments but still have nothing to do with the newly arrived data, especially lack the designs to accommodate model evolution and data-adaptive update [33], [36], [37]. Second, current frameworks including continuous learning, suffer computational burdens and space complexity from preserving the historical regularity [33] and new patterns [34], [35], thus limiting the efficiency of model generalization when the patterns and structures vary.

Fortunately, there are some updates on the understanding of memory mechanism in human brain, i.e., different regions in our brain usually carry out distinctive roles and work in a complementary manner to consolidate historical memory and assimilate fresh knowledge [38], [39], [40], [41]. In particular, it reveals that the neocortex neural module gradually acquires structured and well-learned historical knowledge whereas the hippocampus structure tends to efficiently learn specific individual instance-level skills [39], [42]. This insight, formally referred to complementary learning system (CLS) [38], opportunely provides clues to consolidate and update model parameters in a complementary perspective for adapting streaming spatiotemporal observations.

Recently, the complementary learning system has been investigated to realize continuous learning [42], but it still has never been coupled with spatiotemporal frameworks. Given the inherent property of spatiotemporal data, i.e., revealing complex spatial and temporal dependencies with interactions among environmental factors, and the nature of complementary learning, i.e., requiring disjoint learning structures and effective update strategy, introducing CLS into ST learners still presents the following challenges.

- How to seamlessly couple the complex spatiotemporal learners with complementary learning in a unified and efficient framework, i.e., given an ST learner, how to efficiently identify both stable neocortex neural module and dynamic hippocampus structure for transferability and model update, respectively?
- How to cooperatively model spatiotemporal observations with environment features in a holistic perspective, and then appropriately deal with unseen data to adapt the hippocampus structure to new environments?
- How to design the training strategy to simultaneously preserve the historical information and enable online model to update upon new patterns with limited consumption?

In this work, inspired by complementary learning system in neuroscience, we propose a prompt-based data adaptive Complementary ST learning System (ComS2T) to tackle the OOD challenge and endow the model with evolution capacity. Our ComS2T actively identifies the respective stable and dynamic subspace of learning weights to instantiate the complementary learning. First, by reflecting the spatial-temporal dependencies into learnable parameters, we disentangle the full neural weights into two complementary subspaces, stable neocortex and dynamic hippocampus. Second, interactions between environment factors and spatiotemporal observations are often multi-layered and complex. To disentangle such interactions and refine them as prompts to train the following neural architecture, we take spatial location description and temporal signals as basic environment signals for prompts. These basic signals are utilized to train learnable spatial-temporal prompts by reconstructing the input main observations with parameterized distribution. We then exploit the well-learned prompts to fine-tune hippocampus structures of our spatial-temporal blocks, allowing the whole architecture to evolve with new input observations. Finally, we devise a two-stage training process with spatiotemporal warm-up and prompt-based fine-tune, which progressively learns the mapping functions conditional on prompts and allows efficient adaptation during testing stage. To this end, our complementary learning empowers model evolution on both training and testing stages. Specifically, along the training process, our CLS simultaneously preserves the historical information and allows the flexibility of hippocampus. During testing process, the designs of adapting spatial-temporal prompts to testing data with limited self-training further enables the model adaption. The contributions of this study are summarized as follows.

- It is the first attempt to couple complementary learning in neuroscience with spatiotemporal models to realize generalization and data adaptation, where an efficient neural architecture disentanglement is devised through two well-preserved variation matrices.

- A self-supervised prompt learning is proposed to bridge the gap between environment factors and distribution of main observation, which not only allows prompts for neural network fine-tune, but also enables the dynamics and evolution of model parameters sensitive to data distributions.
- Our framework can simultaneously deal with shifts on both spatial and temporal aspects, and four OOD scenarios are constructed to imitate the data adaptation for model verification. Experiments show that our ComS2T can improve performances from 0.73% to 10.79% under temporal shifts, while promote 1.19% to 14.48% under structural shifts.

II. Preliminaries

A. Problem Formulation and Basic Structures

Given spatiotemporal graphs with T steps, $\mathbb{G} = \{\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_t, \ldots, \mathcal{G}_T\}$, each $\mathcal{G}_t$ is described as $\{\mathcal{V}, \boldsymbol{X}_t, \mathcal{E}\}$ where $\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$ is the node set, and $\mathcal{E}$ describes the graph structure. In the observed spatiotemporal graph, we denote $\boldsymbol{X}_t \in \mathbb{R}^{N\times F}$ as the deterministic main observations of $\mathcal{G}_t$, and take $\mathbf{E} = \{\mathbf{E}_s, \mathbf{E}_t\}$ as the observed contextual environments, consisting of spatial environment $\boldsymbol{e}_s \in \mathbf{E}_s$ such as geographical encoding and location index, and temporal environment $\boldsymbol{e}_t \in \mathbf{E}_t$, such as day of week, timestamps of day, etc. Spatiotemporal learning aims to predict next consecutive l steps by exploiting previous κ steps, i.e., $\widehat{\boldsymbol{Y}} = f(\boldsymbol{X})$ where $(\boldsymbol{X}, \boldsymbol{Y}) = (\boldsymbol{X}_{t-\kappa+1:t}, \boldsymbol{X}_{t+1:t+l})$, and f is a spatiotemporal learner. Generally, for an ST learning framework $f(\cdot)$, it usually consists of two main components, graph-based spatial learning $F_S(\boldsymbol{X})$ and temporal convolution module $\Gamma_T(\boldsymbol{X}_S)$, where the two components can learn alternately. Given input sequential observations $\mathbb{X} = \{\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_\kappa\}$, we formalize the spatial representation $\boldsymbol{X}_s$, and the output of spatial learning block $F_s(\cdot)$ as,

$$\boldsymbol{X}_S = F_S(\boldsymbol{X}) = \text{GCN}(\boldsymbol{A}\boldsymbol{X}\boldsymbol{W}_{sp}) \tag{1}$$

The output of temporal learning block is $\boldsymbol{X}_{ST}$ which is constructed by feeding $\boldsymbol{X}_S$ into Γ_T, i.e.,

$$\boldsymbol{X}_{ST} = \Gamma_T(\boldsymbol{X}_S) = \text{TCN}(\boldsymbol{X}_S; \boldsymbol{W}_T) \tag{2}$$

where $\boldsymbol{W}_s = \{\boldsymbol{A}, \boldsymbol{W}_{sp}\}$ and $\boldsymbol{W}_T$ respectively account for learnable parameters on spatial and temporal perspectives. Given the training and testing data $\mathcal{D}_{train}$, $\mathcal{D}_{test}$, the data distribution shift refers to the changes of distribution over training and testing observations, i.e., $P_{train}(\boldsymbol{X}) \neq P_{test}(\boldsymbol{X})$, and the goal for data-adaptive model evolution is to derive a new mapping $\widehat{\boldsymbol{Y}} = f^*(\boldsymbol{X}; \boldsymbol{P})$ simultaneously containing an invariant relation component and a data-adaptive dynamic component based on prompts $\boldsymbol{P}$. The learning objective can be determined as,

$$\min_{(\boldsymbol{x},\boldsymbol{y})\in\mathbb{G}_s} \mathcal{R}(\widehat{\boldsymbol{y}}; f^*(\boldsymbol{x})) \tag{3}$$

B. Theoretical Analysis for Complementary Spatiotemporal Learning

To analyze the superiority of complementary learning, we first provide the definition and process of CLS, and make some fundamental assumptions of spatiotemporal learning to facilitate the derivation, then demonstrate the superiority of coupling CLS with ST learners by careful derivation.

1) Complementary Learning System: From the neuroscience, theory of Complementary Learning System (CLS) delivers how different structures of our brain work cooperatively to consolidate the knowledge and acquire new skills. Specifically, CLS in our brain is consisted of two major components, hippocampus $\mathcal{M}_H$ and neocortex $\mathcal{M}_N$ structures. The hippocampus $\mathcal{M}_H$ captures the short-term and new episodic information in a real-time manner while neocortex $\mathcal{M}_N$ accounts for slow learning of structured information and then replays such consolidated memory [40]. Moreover, the neural structures in brains are activated by outer stimulation and activation signals can be transmitted by pre-synaptic neurotransmitters [43]. Concretely, the cooperation and update function $\mathcal{U}$ can be formulated as two steps [44], [45],

$$\mathcal{M}_N \overset{cons}{\longleftarrow} \mathcal{M}_H; \quad \mathcal{U}(\mathcal{M}_N, \mathcal{M}_H) = \mathcal{M}_N || \mathcal{M}_H \tag{4}$$

where $cons$ indicates the consolidating process from short-term memory $\mathcal{M}_H$ to stable neocortex memory $\mathcal{M}_N$, $||$ denotes the neural structure concatenation.

2) Fundamental Assumptions on ST Learning: To analyze learning process of spatiotemporal forecasting, we make fundamental assumptions following EERM [30].

Assumption 1 (Dependence between main observation and their environments): Given sequential spatiotemporal observations $\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_N$, we suppose the distributions $\mathbf{P}(\boldsymbol{X})$ can be dependent on the contextual environments $\mathbf{E} = \{\mathbf{E}_s, \mathbf{E}_t\}$.

Different from modeling categorical environments in other literature [36], [37], we do not assume a closed set for environments, instead we instantiate them with continuous geographic and timestamp embeddings, indicating the urban structure and overall temporal shifts. Then the distribution shifts over $\boldsymbol{X}_i$ can be attributed to changes of the virtual environment $\mathbf{E}$.

Assumption 2 (Invariance property): Even though the covariate distributions changing over environments, there must exist some invariant relations. Given two environments $\boldsymbol{e}_i, \boldsymbol{e}_j \in \mathbf{E}$, $\exists (p, q), s.t.\ \mathbf{P}(x_p, x_q|\boldsymbol{e}_i) = \mathbf{P}(x_p, x_q|\boldsymbol{e}_j)$, where x_p and x_q are specific observations of $\boldsymbol{X}$. To this end, we can decompose all the relations between $\boldsymbol{X}$ and $\boldsymbol{Y}$ into invariant parts and dynamic parts, accounting for respective causal and non-causal components.

In the dynamic graph regression, given node v_i, let degree of v_i, and proportions of neighbors with causally invariant relations to v_i denote as d_i, p_i with $d_i > 1, 0 < p_i < 1$. Considering the covariate shifts, i.e., we train the model on samples following Gaussian distribution $\mathbb{G}_s \sim N(\mu_0, \sigma_0|e_0)$ and test on samples following $\mathbb{G}_t \sim N(\mu_q, \sigma_q|e_q)$.

3) Failure on Traditional ST Learning: For GNN-based representation learning, we consider one-time GNN aggregation of its neighbors, from T-step to achieve the expected regression prediction of $T+1$-step. Based on above assumptions, for node v_i, we take $\mathcal{N}_c(v_i)$ as the causally correlated neighbor set of v_i while $\mathcal{N}_s(v_i)$ denotes the set of non-causally correlated neighbors. Given degree d_i, the traditional one-time aggregation for v_i with all nodes can be formulated by decomposing the causal

and non-causal parts, and the hidden representation of node i at time step T can be written as,

$$E(h_i^T) = \frac{x_i^T + \sum_{c_j \in \mathcal{N}_c(v_i)} w_{ij}^c x_{c_j}^T + \sum_{s_j \in \mathcal{N}_s(v_i)} w_{ij}^s x_{s_j}^T}{1 + d_i} \tag{5}$$

where c_j and s_j are the subscripts of two neighborhood sets, $\mathcal{N}_c(v_i)$ and $\mathcal{N}_s(v_i)$, corresponding to nodes with potential stable relations to node v_i and spurious correlations to v_i. Learnable weights w_{ij}^c and w_{ij}^s are denoted for causal parts and non-causal parts. By calculating the difference between the aggregated expectation $E(h_i^T)$ and ground-truth x_i^{T+1}, we can derive the prediction error ε_0 after one-time aggregation by neglecting the non-linear activations, i.e.,

$$\begin{aligned} \varepsilon_0 &= ||E(h_i^T) - x_i^{T+1}|| \\ &= ||\frac{\begin{array}{c} x_i^T + \sum_{c_j \in \mathcal{N}_c(v_i)} w_{ij}^c x_{c_j}^T \\ + \sum_{s_j \in \mathcal{N}_s(v_i)} w_{ij}^s x_{s_j}^T - (1+d_i) x_i^{T+1} \end{array}}{1 + d_i}|| \end{aligned} \tag{6}$$

Assume that observations on both current step x_i^T and next step x_i^{T+1} follow the same Gaussian distribution $N(\mu_0, \sigma_0)$, and $p_i = \frac{||\mathcal{N}_c(v_i)||}{||\mathcal{N}(v_i)||}$ accounts for the proportion of causal neighbors.

To facilitate the expression, we let μ_0^t, μ_0^{t+1} be the expectation of observation x_i at t and $t+1$, and μ_0^c, μ_0^s represent the expectation of the expected observation of its causal neighborhood and non-causal neighborhood. The initial error ε_0 can be modified by,

$$\varepsilon_0 = \frac{\mu_0^t + p_i d_i \mu_0^c w_i^c + (1-p_i) d_i \mu_0^s w_i^s - (1+d_i)\mu_0^{t+1}}{1+d_i} \tag{7}$$

where we ignore the sign for absolute value, and assume that the expectation and learnable weights all preserve positive.

Since the non-causal based learning is formulated by regression function of $\widehat{y}_i = w_i^c x_c + w_i^s x_s$, the prediction residual res_i will be derived by $res_i = \widehat{y}_i - w_i^c x_c = w_i^s x_s$. Therefore, we can substitute the difference between aggregated causal parts and ground-truth with aggregated non-causal part, and obtain the following equations,

$$\begin{aligned} \varepsilon_0 &= \frac{\mu_0^t + p_i d_i \mu_0^c w_i^c - (1+d_i)\mu_0^{t+1} + (1-p_i) d_i \mu_0^s w_i^s}{1+d_i} \\ &= \frac{2(1-p_i) d_i \mu_0^s w_i^s}{1+d_i} \end{aligned} \tag{8}$$

With (8), we can arrive that the derived error is not reducible as $w_i^c \neq 0$. And we further disentangle the influence factors of this error. As causal parts are defined on the stable relations while non-causal parts are defined on highly variant correlations across distributions, we can impose the distribution assumption of corresponding learnable weights by,

$$\begin{aligned} &w_i^c \sim N(\mu_w, \sigma_{wc}), \quad w_i^s \sim N(\mu_w, \sigma_{ws}) \\ &s.t.\ \sigma_{ws} \gg \sigma_{wc} \end{aligned} \tag{9}$$

Given that if one random variable follows Gaussian distribution, then 99.73% of the samples fall into the ranges between $[\mu - 3\sigma, \mu + 3\sigma]$, according to the 'Three Sigma Principe'. This principle tells us almost all samples must fall into above ranges excluding very few extreme values. Then we can approximate error ε_0 with restoring μ_0^s to μ_0,

$$\varepsilon_0 \sim \frac{2(1-p_i) d_i \mu_0 (\mu_w \pm 3\sigma_{ws})}{1+d_i} \tag{10}$$

Since two variables of p_i satisfying $0 < p_i < 1$ and $\sigma_{ws} \gg \sigma_{wc}$ are constants that cannot be ignorable, the errors for non-invariance learning will be positively proportional to both $\mu_0 \mu_w$ and σ_{ws} while negatively correlated with p_i. That's to say, the less causal parts within observations, i.e., smaller p_i, and the larger variations of relations across environments, i.e., larger σ_{ws}, the performance deterioration will be more serious. Moreover, when f is transferred to OOD testing set $N(\mu_q, \sigma_q | e_q)$ satisfying $\mu_q = q\mu_0$ where $q \in \mathbb{N}^+$. The approximated error of OOD testing risk is amplified to,

$$\varepsilon_q \sim \frac{2(1-p_i) d_i q \mu_0 (\mu_w \pm 3\sigma_{ws})}{1+d_i} \tag{11}$$

To this end, with $\mu_w \neq 0$, this derivation manifests that learning over OOD scenarios suffer $q - times$ errors of in-distribution (ID) ones, resulting in an unacceptable amplification of error bounds from ID to OOD samples. Thus, the traditional non-invariant relation learning is inclined to fail on OOD regressions. □

4) Superiority of Complementary ST Learning: In CLS, given $\widehat{\boldsymbol{y}}_i = \boldsymbol{w}_i^c \boldsymbol{x}_c + \boldsymbol{w}_i^s \boldsymbol{x}_s$, when the invariant neural architecture and dynamic context-sensitive architectures are disentangled, then the learnable weights can be explicitly separated into two sections, i.e., $\boldsymbol{W} = \{\boldsymbol{w}^c, \boldsymbol{w}^s\}$, for stable neocortex and dynamic hippocampus structures. The two learnable parts are explicitly considered as satisfying the following distributions,

$$\boldsymbol{w}^c \sim N(\mu_w, \sigma_{wc}), \quad \boldsymbol{w}^s \sim N(\mu_w, \sigma_{ws}) \tag{12}$$

In OOD scenarios of our CLS, $\boldsymbol{w}^c$ should be static thus transferable across environments, while $\boldsymbol{w}^s$ can be variable and timely updated upon the distribution of main observation changes. Then given node v_i in spatiotemporal graph, (7) becomes reducible and can be suppressed by optimizing w_i^s. Let (7) $= 0$, the w_i^s must have an analytical solution to this optimization, shown as,

$$w_i^s = \frac{(1+d_i)\mu_0^{t+1} - (\mu_0^t + p_i d_i \mu_0^c w_i^c)}{(1-p_i) d_i \mu_0^s} \tag{13}$$

We then conclude that our ComS2T can potentially converge to optimal results under the distribution shifts with disentanglement and update mechanisms for data adaptation.

III. Methodology

A. Framework Overview

Inspiring by neuroscience, ComS2T unifies the invariance and dynamics into a complementary spatiotemporal learning system, as illustrated in Fig. 2. First, it efficiently disentangles the learnable neural weights into two complementary subspaces, where two structures work cooperatively to dynamically adapt streaming spatiotemporal data. Second, ComS2T pre-trains the spatial-temporal prompts via self-supervised learning, bridging

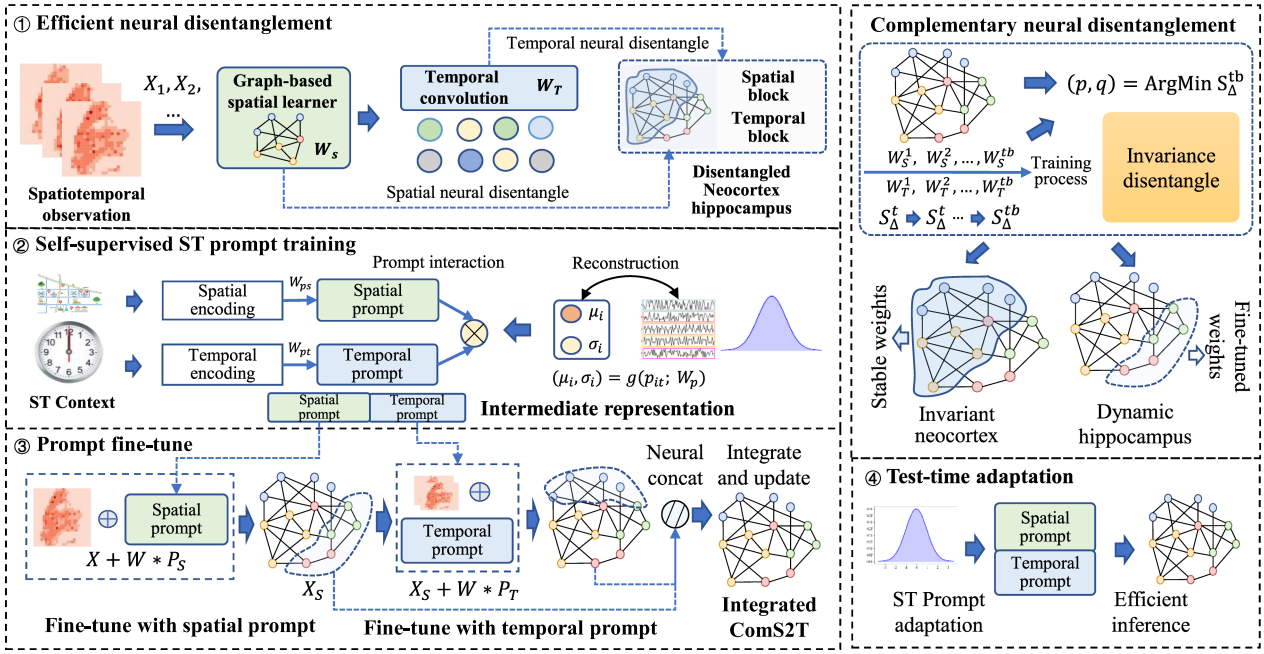


Fig. 2. Framework overview of ComS2T, consisting of four major components. The right top Complementary neural disentanglement is the detailed illustration of the first component for decoupling stable and dynamic weight spaces in CLS.

the gap between learnable prompts and specific data patterns. Such pre-training strategy allows test-time training and model sensitivity to distribution shifts. Finally, we devise a progressive learning architecture, consisting of four major components, i.e., efficient neuron disentanglement, prompt pre-training, prompt-based fine-tune and test-time self-adaptation, to realize the data adaptation from coarse to fine-grained.

B. Efficient Neural Disentanglement

Following the complementary learning that different neural structures play respective roles, we propose an efficient disentanglement to decouple the potential neocortex and hippocampus structures from neural networks.

To ensure an interpretable and aligned disentanglement, we take spatial module and temporal module as separated units. Assuming there are K layers for spatial aggregation and L times of temporal convolution, we take both the spatial adjacency and feature-level scaling as the spatial learnable space $\boldsymbol{W}_S = \{\boldsymbol{A}_i, \boldsymbol{\omega}_{s_i}\}_{i=0,1,\ldots,K}$ and the set of temporal learnable weights is designated as $\boldsymbol{W}_T = \{\boldsymbol{w}_{t_i}\}_{i=0,1,\ldots,L}$. For easier notation, we take all learnable parameters in a spatiotemporal learner $f(\cdot)$ as $\boldsymbol{W} = \{\boldsymbol{W}_s, \boldsymbol{W}_T\}$, and denote $w_{s_{ij}}$ and $w_{t_{ij}}$ as the specific deterministic element in $\boldsymbol{W}_s \in \mathbb{R}^{P\times Q}$ and $\boldsymbol{W}_T \in \mathbb{R}^{M\times N}$, where $P \times Q$ and $M \times N$ are the virtual dimensions for representing these two weight sets for convenient description, respectively .[1]

To obtain an efficient neural disentanglement, we describe the model behavior by weight series, i.e., $\boldsymbol{W}^0, \boldsymbol{W}^1, \ldots, \boldsymbol{W}^{tb}$, where tb is the training unit consisting of several batches or epochs in the learning process, and we can take one epoch as a training process unit. The variations over different neuron-level elements can be observed when the learning process achieves relative stability. To capture the evolution behavior along training process, we propose a differential accumulation strategy. Specifically, we instantiate two matrices, the absolute differential values $\Delta\widetilde{\boldsymbol{W}}^{tb}$, which characterizes the differences between adjacent training units, and the accumulated differences $\boldsymbol{S}_{\Delta}^{tb}$, which is the summation of all variations. Given the unit number within training process tb, the element-level model evolution can be characterized by,

$$\begin{cases} \Delta\widetilde{\boldsymbol{W}}^{tb} = |\boldsymbol{W}^{tb} - \boldsymbol{W}^{tb-1}| \\ \boldsymbol{S}_{\Delta}^{tb} = \Delta\widetilde{\boldsymbol{W}}^{tb} + \boldsymbol{S}_{\Delta}^{tb-1} \end{cases} \tag{14}$$

Note that $\boldsymbol{S}_{\Delta}^{0} = \boldsymbol{0}$ is the initialized status, and the absolute differential values on learnable parameters can directly outline the quantified variations on streaming training samples, alleviating the influence of signed variation on final accumulation. We calculate (14) in an element-wise manner, and impose such differential operation on both spatial learning space $\boldsymbol{W}_s$ and temporal convolution space $\boldsymbol{W}_T$, capturing stable relations in an element-wise subtraction manner with interpretability. As smaller values in $\boldsymbol{S}_{\Delta}^{t}$ indicate more stable relations along the training process, we separate the stable and unstable neurons by taking the minimal-$\tau\%$ variations as a threshold. This cutting-off threshold is respectively imposed over the spatial and temporal learning blocks. The τ is a hyperparamter, indicating that the model fluctuates over training samples and can be optimized empirically. To this end, weight space with smallest variations, accounting for $\tau\%$ least values in $\boldsymbol{S}_{\delta}^{t}$, are considered as the

[1] We take virtual dimensions for convenient model derivation, where the actual parameter dimension is the addition among various neural layers that is hard to characterize.

Algorithm 1: Training Procedure of ComS2T.

Input: Main observations $\boldsymbol{X}$, Observed environment description $\mathbf{E}$;

Output: Neocortex structure on spatial and temporal blocks $\boldsymbol{W}_{ne}^{S}, \boldsymbol{W}_{ne}^{T}$; hippocampus structure on spatial and temporal blocks $\boldsymbol{W}_{hp}^{S}, \boldsymbol{W}_{hp}^{T}$;

1: **for** iteration = 1, 2,..., Q **do**
2: Neural disentanglement decouples the learning neural spaces into initial neocortex structure $\boldsymbol{W}_{ne}^{S}, \boldsymbol{W}_{ne}^{T}$ and hippocampus structure $\boldsymbol{W}_{hp}^{S}, \boldsymbol{W}_{hp}^{T}$ based on (14) to (16).
3: Self-supervised prompt training based on (17) to (19) to obtain $\boldsymbol{P}_S, \boldsymbol{P}_T$.
4: Fine-tune hippocampus structure with prompts $\boldsymbol{P}_S, \boldsymbol{P}_T$ based on (20) to (23), and achieve updated $\boldsymbol{W}_{hp}^{S}, \boldsymbol{W}_{hp}^{T}$.
5: **end for**

neocortex neural parts for capturing invariant and stable spatial-temporal correlations, while the complementary set (indexes for remaining $(1-\tau\%)$ largest values within $\boldsymbol{S}_{\Delta}^{t}$) of the neural parameters is considered as the hippocampus neural structure for data-adaptive update. Given the training unit tb, we denote $\boldsymbol{S}_{sp\Delta}^{tb}, \boldsymbol{S}_{tp\Delta}^{tb}$ as the summarized variations on spatial and temporal learning blocks. The neocortex and hippocampus structures can then be determined by highlighting the most stable learnable weights as follows,

$$\begin{cases} \{(p,q)_S\} = \underset{\substack{\text{Min}-\tau\% \\ 1\leqslant p\leqslant P, 1\leqslant q\leqslant Q}}{\arg\min} \left(\{\boldsymbol{S}_{sp\Delta}^{tb}(p,q)\}\right) \\ \{(m,n)_T\} = \underset{\substack{\text{Min}-\tau\% \\ 1\leqslant m\leqslant M, 1\leqslant n\leqslant N}}{\arg\min} \left(\{\boldsymbol{S}_{tp\Delta}^{tb}(m,n)\}\right) \end{cases} \tag{15}$$

The index sets $\{(p,q)_S\}, \{(m,n)_T\}$ are index sets of selected neuron-level neocortex elements in spatial and temporal aspects $\boldsymbol{W}_s$ and $\boldsymbol{W}_T$. After that, the disentanglement process can be implemented by,

$$\begin{cases} \boldsymbol{W}_{ne}^{S} = \underset{\substack{\text{Min}-\tau\% \\ i,j\in\{(p,q)_S\}}}{\text{Avg}} \left(\boldsymbol{W}_S(i,j)\right), \quad \boldsymbol{W}_{hp}^{S} = \boldsymbol{W}^{S} - \boldsymbol{W}_{ne}^{S} \\ \boldsymbol{W}_{ne}^{T} = \underset{\substack{\text{Min}-\tau\% \\ i,j\in\{(m,n)_T\}}}{\text{Avg}} \left(\boldsymbol{W}_T(i,j)\right), \quad \boldsymbol{W}_{hp}^{T} = \boldsymbol{W}^{T} - \boldsymbol{W}_{ne}^{T} \end{cases} \tag{16}$$

where the $\text{Avg}(\cdot)$, which averages the learnable weights in the set, can be viewed as a smooth strategy to ensure the generality and smoothness of transferable neocortex structures. And $A-B$ denotes as the complementary set of B to set A. $\boldsymbol{W}_{ne}^{S}, \boldsymbol{W}_{ne}^{T}$ are decoupled neocortex neural structures on respective spatial and temporal blocks while $\boldsymbol{W}_{hp}^{S}, \boldsymbol{W}_{hp}^{T}$ are hippocampus neural structures for two blocks. Comparing with existing OOD generalization, the efficiency and adaptions of our proposed neural disentanglement lies in that we do not require too much memory, but only update the $\boldsymbol{S}_{sp\Delta}^{tb}, \boldsymbol{S}_{tp\Delta}^{tb}$ every training unit, then the disentanglement can be implemented along the usual training process.

C. Self-Supervised Spatial-Temporal Prompt Learning

Complementary learning system (CLS) is proposed to enable adaptive model evolution with data distribution. To this end, two important issues rise the attention, i.e., 1) how to endow the model with capacity of data adaptation, especially exploiting non-labeled samples for model evolution, 2) In a memory system, a brief but powerful summary can help remember better, thus how to design brief and informative prompts to guide the update of model becomes important.

In this section, we design a distribution-supervised pre-training strategy to achieve continuous prompt representations in a self-supervised manner, thus we can take spatial-temporal prompts as an intermediate variable to deliver the variation of data to main models. First, we respectively select informative spatial and temporal signals as the basic elements of prompts. We take longitude, latitude, location index as basic spatial information i.e., $\boldsymbol{e}_s(i) = [lat, long; loc\_no]_{v_i} \in \mathbb{R}^{2\times E}$, while consider day of week, time step, and time-series trend as representative temporal signal, i.e., $\boldsymbol{e}_t(t) = [Dw, Ts; Tr]_t \in \mathbb{R}^{2\times E}$, where ';' denotes the division of line in the matrix, E is the embedding dimension, and each line carries their distinctive semantic meanings. Second, we explicitly model the distribution as a data summary over sequential observations, and construct a question-answer pair between spatial-temporal prompts and data summary to empower prompts sensitive to data distribution. Given the spatial location $\boldsymbol{e}_s(i)$ and temporal step $\boldsymbol{e}_t(t)$ at node i and step t in the spatiotemporal graphs, we model the continuous κ-step observations at corresponding spatial-temporal context as an observed distribution by corresponding parameters $(\mu_i^t, \sigma_i^t) \sim \boldsymbol{X}_i^{t:t+\kappa}$. Then we can easily regress these parameters through a carefully designed learning blocks. To explore the relations between data distribution and spatial-temporal context, we construct a Spatial-Temporal Interaction Module (STIM) $g(\cdot)$ to capture the interactions between spatial and temporal contexts. It consists of a Compressed Interaction Network and an MLP structure, which allows field-level interactions between spatial and temporal prompts via inheriting the property of Deep Factorization Machine [46]. With STIM, our self-supervised pre-training on prompts can be considered as a regression task,

$$\begin{aligned} &(\widehat{\mu}_i, \widehat{\sigma}_i)_t \\ &\quad = g\left(\left[\text{MLP}\left(\boldsymbol{e}_s(i); \boldsymbol{W}_{ps}\right) \odot \text{MLP}\left(\boldsymbol{e}_t(t); \boldsymbol{W}_{pt}\right)\right]; \boldsymbol{W}_P\right) \end{aligned} \tag{17}$$

where $\boldsymbol{e}_s(i) \in \boldsymbol{E}_s$ and $\boldsymbol{e}_t(t) \in \boldsymbol{E}_t$ are basic elements of spatial and temporal signals for construction of prompts, also accounting for the environment signals in our Fundamental Assumptions. The $\boldsymbol{W}_{ps}, \boldsymbol{W}_{pt}, \boldsymbol{W}_P$ are learnable weights for transforming spatial prompts, temporal prompts as well as overall prompt representation to predictive distribution parameters. Through the regression of parameters over sequential observations, the intermediate representations of corresponding spatial-temporal signals are taken as prompts respectively,

$$\boldsymbol{P}_S = \text{MLP}\left(\boldsymbol{e}_s(i); \boldsymbol{W}_{ps}\right), \quad \boldsymbol{P}_T = \text{MLP}\left(\boldsymbol{e}_t(t); \boldsymbol{W}_{pt}\right) \tag{18}$$

The above $\boldsymbol{P}_S, \boldsymbol{P}_T$ become the well-learned spatial and temporal prompts by imposing the following self-supervised learning

objective,

$$Loss_{self} = \min \Sigma_i^N \Sigma_t^T \left(((\hat{\mu_i})_t - (\mu_i)_t)^2 + ((\hat{\sigma_i})_t - (\sigma_i)_t)^2 \right) \tag{19}$$

The above parameterized distribution can flexibly guide the fine-tune on spatial and temporal prompts upon accessing new observations, regardless of the learning phases of training or testing. Moreover, such adjustment on prompts can deliver dynamics to hippocampus structure of our main model, thus empowering generalization to OOD scenarios and endowing it with evolving capacity.

D. Progressive Spatiotemporal Learning

In this section, we couple the prompt learning with a two-stage training, consisting of both warm-up and fine-tune, progressively achieving the evolution capacity. Concretely, for the whole architecture of ComS2T, it composes of four stages, spatiotemporal model warm up and invariance decoupling, self-supervised pre-training, prompt-based fine-tune during model training, and test-time adaptation for adaptive testing.

Spatiotemporal model warm up and invariance decoupling: Following Section III-B, we train spatial and temporal blocks with pair-wise main observations $\{(\boldsymbol{X}, \boldsymbol{Y})\}$, namely model warm up, until achieving stability of learnable parameters. Specifically, we refer the learning divergence to the differences of training errors between two batches and the stability refers to such differences converge at one specific with limited variations. By denoting the training error at t batch as ε_t, the stability of training process can be formulated as $|\varepsilon_{t+1} - \varepsilon_t| < \varepsilon_0$ where ε_0 is the threshold describing stability.

We can activate the efficient neural disentanglement at the end of warm-up stage. We characterize the model behavior by retrieving the accumulated variations of learnable weights at the stopped training unit tb, i.e., $\boldsymbol{S}_{sp\Delta}^{tb}, \boldsymbol{S}_{tp\Delta}^{tb}$. The neurons with $\tau\%$ smallest values in accumulated variations are disentangled as neocortex that can be considered as stable neuron structure, while the complementary neuron set are classified as hippocampus that can be viewed as dynamically updated structure. We can easily obtain the neural structure divisions, in spatial perspective $\boldsymbol{W}_{ne}^S, \boldsymbol{W}_{hp}^S$, and temporal perspective $\boldsymbol{W}_{ne}^T, \boldsymbol{W}_{hp}^T$. With all the parameters learned during warm up, $\boldsymbol{W}_s$ and $\boldsymbol{W}_T$ can be the initialization for following fine-tune process and enable the model preliminary to adapt spatial-temporal observations.

Self-supervised pre-training: We then construct pair-wise training samples and exploit the distribution reconstruction to learn informative semantic spatial-temporal prompts. This allows the prompt learned by only accessing the distribution over main observations without predicted future observations.

ST prompt-based fine-tune: This stage allows the prompt as additional inputs of our framework with fine-tuning process. We leverage the neocortex to preserve stable weights for transferring invariant relations across environments, and take spatial-temporal prompts to guide hippocampus to update with distribution shifts where the meta information on spatial and temporal aspects can reflect most data changes. To facilitate the gradient propagation and semantic information aggregation, we inject the spatial and temporal prompts separately into the hippocampus structures regarding respective spatial and temporal learning blocks with careful dimension alignment. Specifically, we first freeze the neurons within neocortex structure, and integrate well-learned prompts with main observations as input to hippocampus structure. Given the well-learned spatial prompt $\boldsymbol{P}_s$ and temporal prompt $\boldsymbol{P}_T$, in our fine-tune stage, the input of spatial learning block becomes,

$$\boldsymbol{X}_{in} = \boldsymbol{X} \oplus \left(\boldsymbol{W}_{ps}^{al} * \boldsymbol{P}_s \right) \tag{20}$$

where $\oplus$ indicates element-wise addition. $\boldsymbol{W}_{ps}^{al} * \boldsymbol{P}_s$ accounts for dimension alignment between spatial prompt and main observations. After that, we freeze neocortex structure $\boldsymbol{W}_{ne}^S$ and let hippocampus parameters $\boldsymbol{W}_{hp}^S$ update. The output of spatial blocks can be written as,

$$\boldsymbol{X}_S = F_S \left(\boldsymbol{X}_{in}; \boldsymbol{W}_{hp}^S | \boldsymbol{W}_{ne}^S \right) \tag{21}$$

Similarly, we update the input of temporal learning block by pre-alignment,

$$\boldsymbol{X}_{ST} = \boldsymbol{X}_S \oplus \left(\boldsymbol{W}_{pt}^{al} * \boldsymbol{P}_T \right) \tag{22}$$

Then the output of the temporal blocks can be written as,

$$\boldsymbol{Y} = \Gamma_T \left(\boldsymbol{X}_{int}; \boldsymbol{W}_{hp}^T, \boldsymbol{W}_{ne}^T \right) \tag{23}$$

With complementary learning strategy, the neocortex structure transfers stable relations into OOD scenarios while introduces prompt signals derived from external factors to update hippocampus structures, allowing the model to be dynamic with data distribution changing. Our design fixes invariant relations within spatial-temporal observations, and regress the residual observations with contexts, which enables fine-tune only requiring limited computations. To this end, the whole architecture integrates spatial and temporal learning blocks where each block consists of corresponding neocortex and hippocampus structures. Formally, the integrated model becomes,

$$\boldsymbol{Y} = f^* \left(\boldsymbol{X}, \boldsymbol{P}_S, \boldsymbol{P}_T; \left(\boldsymbol{W}_{hp}^S || \boldsymbol{W}_{ne}^S \right) || \left(\boldsymbol{W}_{hp}^T || \boldsymbol{W}_{ne}^T \right) \right) \tag{24}$$

The symbol $||$ represents the concatenation between neuron structures. In addition, to follow up on the latest findings of complementary learning [44], [45], i.e., neocortex can slowly update to accommodate the new knowledge and dynamic hippocampus structure, we further devise an alternate learning strategy to update them with different frequencies, i.e., the neocortex regularly update every K_b batches while hippocampus structure updates every batch.

Test-time adaptation: To enable a true evolvable model, we take advantage of self-supervised learning to fine-tune the prompts during test time, which enables the efficient update on partial model parameters. When new observations arrive, we can sample a small number of batches of observations with their counterpart spatial and temporal descriptions, $\widetilde{\boldsymbol{X}}_{test} \sim \boldsymbol{X}_{test}$. Then we can fine-tune the prompt representation $\boldsymbol{W}_{ps}$ and $\boldsymbol{W}_{pt}$ with new data distribution, and update the intermediate embedding $\widetilde{\boldsymbol{P}}_S, \widetilde{\boldsymbol{P}}_T$ based on (17) and (19). Finally, we can correspondingly obtain the new learning outputs by re-exploiting

Algorithm 2: Testing Procedure of ComS2T.

Input: Main testing observations $\boldsymbol{X}_{test}$, Observed environment description $\mathbf{E}_{test}$, Well-learned parameters of ComS2T $\boldsymbol{W}^S_{ne}, \boldsymbol{W}^T_{ne}, \boldsymbol{W}^S_{hp}, \boldsymbol{W}^T_{hp}$;

Output: Prediction results $\widehat{\boldsymbol{Y}}_{test}$;

1: Sample partial observations from testing set $\widetilde{X} \sim \boldsymbol{X}_{test}$ and compute the parameterized distribution $(\widetilde{\mu}, \widetilde{\sigma})$;

2: Update $\boldsymbol{P}_S, \boldsymbol{P}_T$ into $\widetilde{\boldsymbol{P}}_S, \widetilde{\boldsymbol{P}}_T$ based on (17) to (19);

3: Implement prediction based on (25) and output $\widehat{\boldsymbol{Y}}_{test}$.

TABLE I
MODEL EFFICIENCY COMPARISONS ACROSS BASELINES

Model	Number of parameter updates under temporal shift
CauSTG	$KL + LP\gamma\%$
PECPM	$L + LP\gamma\%$
TrafficStream	$L + LP\gamma\%$
ComS2T	$L + L\gamma\% + PE_P$

(24),

$$\boldsymbol{Y}_{test} = f^*\left(\boldsymbol{X}_{test}, \widetilde{\boldsymbol{P}}_S, \widetilde{\boldsymbol{P}}_T; (\boldsymbol{W}^S_{hp} || \boldsymbol{W}^S_{ne}) || \left(\boldsymbol{W}^T_{hp} || \boldsymbol{W}^T_{ne}\right)\right) \tag{25}$$

Learning objective: We take Mean Absolute Error (MAE) as the main learning objective for training in both warm-up and fine-tune stages. For self-supervised learning, we exploit the reconstruction loss $Loss_{self}$ as the objective.

Model efficiency analysis: We compare model efficiency against three continuous learners by number of updated parameters. First, assuming that each model is equipped with L parameters, the streaming data will experience P times of distribution shifts. The number of parameters regarding prompt update can be denoted as $|\boldsymbol{W}_{ps}| + |\boldsymbol{W}_{pt}| + |\boldsymbol{W}_P| = E_P \ll L$. There are K temporal environments, above three models will update $\gamma\%$ parameters, where $\gamma\% = 1 - \tau\%$ in ComS2T. Specifically, CauSTG learns K submodels across temporal environments and then updates model during testing scenario. PECPM and TrafficStream will update the $\gamma\%$ parameters of the whole model. For PECPM and TrafficStream, PECPM updates partial parameters of model when the distribution changes while TrafficStream takes an experience-reply strategy for model update. In contrast, ComS2T only undergoes two stages, i.e., warm-up learning and prompt-based fine-tune, then only a few parameters of spatial-temporal prompts will update when distribution changes. The number of updated model parameters have been listed in Table I. Formally, we assume $E_P \ll L$ as the updated parameters are a smaller proportion of whole parameters and the updated coefficient empirically satisfies $0.1 < \gamma < 0.5$. Let $E_P = \eta L(\eta < 0.1)$ where η is the scaling coefficient for all-set parameters L. When P is increasing, i.e., $P \to +\infty$, we can derive that,

$$\begin{aligned} Diff &= L \times \gamma \times (P-1) - PE_P \\ &= L \times \gamma \times (P-1) - PL \times \eta \\ &= L \times (\gamma \times P - \gamma - P\eta) \\ &= L(P(\gamma - \eta) - \gamma) \end{aligned} \tag{26}$$

where $P(\gamma - \eta) - \gamma > 0$. We can arrive $L \times \gamma \times (P-1) - PE_P > 0$, indicating the efficiency of once training for updates.

Summary: Our ComS2T is efficient and reliable, and its efficiency lies in its potential of decoupling associations during training, and the use of prompt modeling to obtain a continuous mapping function between context environment and real data distribution over main observations. It allows delivering the distribution change from observation to prompt and subsequently the hippocampus of ComS2T. The learning and testing procedures of our ComS2T are respectively described in Algorithms 1 and 2, and detailed experiment settings can be found in Section IV-C.

TABLE II
DATASET STATISTICS

Dataset	Node #	Time step #	Time span	Interval length
SIP	108	25,920	01/01/2017-03/31/2017	5min
Metr-LA	207	34,272	03/01/2012-06/30/2012	5min
KnowAir	184	11,688	01/01/2015-12/31/2018	3h
Temperature	184	11,688	01/01/2015-12/31/2018	3h

IV. EXPERIMENT

We collect four types of spatial-temporal data and design various learning-testing scenarios to imitate distribution shifts in both spatial and temporal dimensions.

A. Dataset Description

We take different categories of spatiotemporal datasets such as traffics, air quality and smart grids, for verification of our data-adaptive learning architecture. The statistics of datasets can be found in Table II.

- *SIP (Traffic):* It is the camera surveillance capturing traffic volumes in Suzhou Industry Park (SIP), Suzhou.
- *Metr-LA (Traffic):* Traffic attributes such as speed, detected by highway loop detectors of Los Angeles, USA. We reach this dataset via literature [18].
- *KnowAir (Air quality):* PM2.5 concentrations, covering 184 main cities of China [47].
- *Temperature (Climate):* Urban numerical temperature covering the same 184 cities as KnowAir [47].

B. Learning With Spatial-Temporal OOD Settings

We construct data distribution shifts on temporal perspective and imitate structure shifts on spatial aspect, as illustrated in Fig. 3. First, **temporal distribution shift** can be imitated by two

training-testing divisions according to data distribution characteristics over different datasets.

- *Interval-level division:* For traffic datasets of SIP and Metr-LA those are highly dynamic, it is observed that evolution patterns on two half days are totally different, and we can well imitate the temporal distribution shift. We thus organize training sets by collecting all the same day intervals (e.g., every 8:00-16:00) for model learning, while perform testing on the other unseen day intervals (e.g., every 1:00-7:00).
- *Month-level division:* For air quality and climate datasets those are relatively static within short term but can vary seasonally, we divide the whole-year records into four trimesters, where we train with the two trimester while test on one of the trimesters.

Second, **spatial distribution shifts** are realized by the involvement of new nodes and removal of existing nodes.

- *Node involvement:* We actively mask a series of existing nodes during training and add them back during testing stage to simulate the new connections of the graph structure. For inference stage, we propose to fine-tune the spatial prompts with new locations, and exploit the node copy strategy [48] from recommendation system to find similar nodes in existing node sets and copy its neighbors to the new ones.
- *Node removal:* Similarly, we remove some existing nodes during testing stage for imitating the node disappearance in the dynamic graph structure. For inference stage, we first re-train the spatial and temporal prompts to fine-tune the model, and mask the lines and columns of the removed nodes in adjacent matrix for dimension alignment. Then the prediction can be implemented.

C. Implementation Details

Dataset processing: For each dataset, we organize them as sample groups following settings in Section IV-B. For **SIP and Metr-LA**, we take 1/3 of samples for training, i.e., every 8:00-16:00, samples every 16:00-24:00 for validation, and samples every 0:00-7:00 for testing where 0:00-1:00 periods are for test-time data adaptation. For air quality and climate datasets **KnowAir** and **Temperature**, we take samples during every first six months (January to June) for training, samples during July and August for validation, samples during October to December for testing where samples of September is considered for test-time data adaptation. We follow this division for each baseline to ensure fair comparison.

Regarding data processing, we encode the categorical context with one-hot embedding and transfer them into fixed-length vectors. Our target is to construct the data-adaptive model to predict next 12 slots based on the current 12 frames ($\tau = 12$) under both spatial and temporal OOD settings.

Deep learning implementation: For general settings, all the methods are implemented using PyTorch 1.10.0 and evaluated on one Tesla V100 GPU. To guarantee fair comparison, we perform grid search to tune the hyperparameters for all baselines over three datasets. The hyperparameter configurations of our ComS2T can be found in Table III.

TABLE III
CONFIGURATION OF COMS2T

Parameter	Concrete values
Backbone of ComS2T	GraphWaveNet (GWN)
Learning rate	1e-4
Dimension of spatial-temporal prompts $\boldsymbol{E}$	(64,16,16,32)
Percentage of stable neocortex τ	(60%, 60%, 60%, 70%)
Hidden dimension of GNNs	32
TCN kernel dimension	(12,6,3,3)
Batch size	64
Optimizer	Adam

The configurations are displayed in the order of SIP, Metr-LA, KnowAir and Temperature if values should be specified by datasets.

Prediction details under OOD settings: The training process of our ComS2T can be three-fold, self-supervised pre-training for spatial-temporal prompts, model warm up and hippocampus structure disentanglement, and prompt-incorporated model fine-tune. With the placeholders of prompts, we can deliver conditional prompts forward to main learning structure, then the prompts and hippocampus structures will be jointly fine-tuned to allow the increase of generalization capacity during fine-tune stage. During testing stages under temporal shifts, we exploit the temporally nearest samples for test-time model adaptation, which allows the update of prompts conditional on distribution shifts.

For structure shifts, when a new node is introduced, we exploit the distribution-supervised learning scheme to update the spatial and temporal prompts based on parameterized distribution of new observations. For adjacent matrix, by inheriting the collaborative filtering in recommendation system for overcoming cold-start issue [48], we impose a node copy strategy, which finds the most proximal nodes to new ones and copies the adjacencies of existing similar nodes to new ones. Thus, an extended relational spatial adjacency is constructed for testing. When an existing node is removed from the spatiotemporal graph, we re-train spatial and temporal prompts, and mask the corresponding line and column of the removed nodes in adjacent matrix for dimension alignment.

Evaluation metrics: Baselines and our ComS2T are implemented five times and the averaged errors are reported. We take Mean Absolute Error (MAE) as the main metric for evaluation. The error can be written as, i.e.,

$$\text{MAE} = \frac{1}{TN}\sum_{t=1}^{T}\sum_{j=1}^{N}|y_i^t - \widehat{y}_i^t| \tag{27}$$

where $\widehat{y}_i^t$ is the predicted observation of node i at time step t, while y_i^t is corresponding ground-truth.

D. Baseline

Our baselines are three-fold, including five satisfactory ST learners, three causal-based ST learners, and three Continuous learning-based ST learners.

- *MTGNN*: A graph-based multi-variate time series learning without defining explicit graph topology [5] (ST learner).

TABLE IV
PERFORMANCE COMPARISONS ON OOD SCENARIOS AGAINST BASELINES (METRIC: MAE)

	SIP			Metr-LA			KnowAir			Temperature		
	Temp shift	Node involve	Node removal	Temp shift	Node involve	Node removal	Temp shift	Node involve	Node removal	Temp shift	Node involve	Node removal
MTGNN	44.25	49.80	46.28	6.86	6.26	6.85	36.73	46.38	39.56	6.87	7.22	6.97
GWN	44.17	50.76	47.57	6.16	5.52	7.64	38.19	36.48	40.05	7.47	8.43	7.92
ST-SSL	44.55	45.23	46.55	6.58	5.76	7.77	39.17	38.72	43.09	8.66	8.17	7.95
HiGP	47.13	48.59	47.87	6.81	7.05	7.31	37.89	37.03	40.40	7.42	7.18	7.12
TimeMixer	43.46	47.23	46.35	4.91	5.53	6.60	37.23	37.89	39.56	7.31	7.75	7.69
CauSTG	43.47	50.10	46.70	5.97	4.68	6.82	37.86	35.75	39.32	7.07	7.44	7.59
CaST	43.52	48.67	46.54	5.59	4.98	7.46	38.88	**33.42**	38.35	7.35	7.16	7.04
IRM+GWN	43.73	47.43	47.14	5.53	4.97	7.47	39.50	35.62	39.61	7.33	8.28	7.22
PECPM	44.78	43.51	47.78	6.18	4.65	6.80	38.94	36.27	39.53	7.56	7.25	7.11
TrafficStream	45.67	45.32	47.95	6.47	4.86	6.92	39.15	37.03	40.48	7.73	8.22	7.90
TTT-ST	45.45	45.87	49.36	5.52	5.65	6.73	37.77	38.13	39.64	7.03	7.23	7.31
ComS2T	**41.12**	**39.88**	**44.52**	**4.38**	**4.35**	**5.62**	**35.32**	34.48	**38.21**	**6.82**	**6.90**	**6.88**
Beyond second best	5.38%	8.36%	3.79%	10.79%	6.40%	14.48%	3.83%	-3.17%	0.36%	0.73%	3.63%	1.19%

The **best results** are bold while second best are underlined.

- *GraphWaveNet (GWN)*: A graph-based traffic prediction model that integrates TCNs and GCNs [49] (ST learner).
- *ST-SSL*: A State-of-the-Art learning architecture explicitly considering discrimination on spatial and temporal dimensions [17] (ST learner).
- *TimeMixer*: A decomposable multi-scale mixing method for multi-variate time-series forecasting [50](ST learner).
- *HiGP*: A graph-based multi-variate time series clustering solution hierarchical forecasting [51](ST learner).
- *CauSTG*: An emerging Causal-based invariant learning for spatial-temporal data [33] (Causal learner).
- *CaST*: A causal lens spatial-temporal learning framework, which explicitly models the environments and imposes the backdoor adjustment [36] (Causal learner).
- *IRM+GWN*: We especially integrate the invariant risk minimization with GraphWaveNet to test the generality of its framework (Causal learner).
- *PECPM*: A memory-based continuous learning via pattern expansion along urban expansion [35] (Continuous learner).
- *TrafficStream*: An experience reply-based continuous learning framework for traffic flow prediction [34] (Continuous learner).
- *TTT-ST*: It is the first test-time training framework for spatiotemporal forecasting [52](Continuous learner).

E. Analysis of Performances Against Competitors

The comprehensive experimental comparisons are shown in Table IV. Note that the temporal shift, node-level involvement and node-level removal are abbreviated as 'Temp shift, Node involve and Node removal' in our result tables. According to the characteristics of datasets, we take interval-level division to imitate the temporal shift on Traffic datasets (SIP and Metr-LA) while take month-level division for air quality and climate datasets (KnowAir and Temperature). The improvements beyond second best baseline are illustrated at the bottom of Table IV. Overall, our ComS2T achieves consistent superior performances against baselines under most scenarios, improving the performances from 0.73% to 10.79% under temporal distribution shifts, and promoting 1.19% to 14.48% under structural shifts. More specifically, it shows a significant improvement on Metr-LA, and this may be attributed to the nice regularity between spatial-temporal prompts and main observations. The detailed four observations on respective categories of solutions are elaborated as follows.

Obs1. Comparison against traditional ST learners: Although traditional ST learners reveal satisfactory performances on settings of consecutive sequence forecasting, but they still fall short under distribution shifts, especially on two traffic datasets. Promisingly, TimeMixer shows superiority under temporal shifts, while MTGNN and ST-SSL reveal some robustness to structral shifts. It is mostly because that 1) TimeMixer is well-designed from temporal dependence learning, 2) for spatial shifts, the learnable adjacencies are well-transferred to new nodes with node copy while the step-wise and node-wise self-supervised signals may play vital role in obtaining distinguished patterns for generalization. Thus, the potential advantage of SSL learning is the self supervision, which is also inherited into ComS2T.

Obs2. Comparison with ST model with invariant learning: Some pioneering models have taken invariance and transferability across environments into consideration to counteract the temporal distribution shifts, e.g., CauSTG, and IRM+GWN. The empirical results show that they can exactly improve the OOD learning capacity but they are still inferior to our ComS2T. The underlying reason can lie in that they only transfer the invariance to OOD scenarios while no specific solutions for model update and data adaptation. These two methods reasonably trap into suboptimal performances.

Obs3. Comparison with ST continuous learning: Further, for those prediction models explicitly considering

environment variations, e.g., CauSTG, CaST, and TrafficStream, which either exploits the closed environment division [33] and codebooks [36], or employs experience reply to re-train the model [34]. Even so, they still fail to fully exploit the available environment information to improve the adaptation capacity. In contrast, our ComS2T leverages both advantages of self-supervised prompts and complementary learning to accommodate spatial and temporal prompts by establishing bridges between main observations and environment prompts, contributing mostly above 3% improvement under temporal shifts and at least 3.63% improvement under structural shifts.

Obs4. Comparison under structural shifts: Finally, even though CauSTG has considered the spatial shifts and PECPM focuses on the issue of road network expansion and achieves several second-best results under structural shifts, it is still empirically inferior to ComS2T. Our work explicitly involves spatial structural contexts by updating spatial prompts with new observations. It is observed that our solution can significantly outperform both CauSTG and PECPM, for instance, it accounts for the improvement of 3.01% against CauSTG under temporal shift of SIP, and 17.30% promption against PECPM under node removal of Metr-LA. Besides, these mentioned solutions still suffer the efficiency issue for multiple training of submodels (CauSTG) and computation of pattern-level matching (PECPM). In our ComS2T, when the urban structure changes, we only require preliminarily update on spatial-temporal prompts with a few observations, and then it can be well-generalized on testing set with new structures, inclusion or exclusion of new nodes.

In summary, we can conclude that our ComS2T is superior to all other baselines on two aspects, i.e., 1) Without sacrificing the memory storage for new pattern preservation and computational burden on sequence-level pattern matching, our ComS2T directly disentangle the stable and dynamic neural architectures and actively update the neural networks in an overall pipeline, resulting in its superior efficiency. 2) Our ComS2T takes both advantage of self-supervised prompt with distribution reconstruction and the complementary learning architecture, which allows flexible prompt updates with new observations and exactly realizes the data adaptation spatiotemporal learning framework.

F. Ablation Study

In ablation studies, we remove each well-designed module or learning strategy in our ComS2T to verify their contributions. We conduct experiments on two OOD scenarios regarding both temporal shifts and structural node-level shifts, where the results are illustrated as two sub-columns for each dataset in Table V. To ensure comprehensive evaluation, we specifically investigate the decoupled influence of each module on respective spatial and temporal blocks. Therefore, our evaluation is divided into overall strategy removal, removing specific strategy on spatial block (S-) and removing specific strategy on temporal block (T-). The detailed descriptions of ablative variants are elaborated according to learning stages as below. When perform these ablation studies, other settings still follow the original ComS2T where best hyperparameters settings are adopted for this testing.

TABLE V
PERFORMANCE COMPARISONS ON VARIANTS OF COMS2T (METRIC: MAE)

Datasets	SIP		Metr-LA		KnowAir		Temperature	
OOD scenarios	Temp shift	Node involve	Temp shift	Node involve	Temp shift	Node involve	Temp shift	Node involve
Non-Hip	54.48	44.62	5.48	5.10	36.47	38.08	6.91	7.05
S-Hip	55.81	40.72	5.40	4.95	36.41	37.27	6.84	7.09
T-Hip	45.58	47.80	5.18	5.26	37.56	35.65	7.03	7.05
Non-SSL	45.73	40.47	4.82	4.74	36.46	37.80	6.98	7.32
S-SSL	42.84	44.92	5.17	5.02	37.44	36.32	7.11	7.18
T-SSL	46.44	43.87	5.35	4.75	35.70	40.03	6.90	7.02
Non-Prompt	49.21	46.47	4.99	4.59	38.33	37.43	6.99	7.09
SSL-KL	42.46	42.10	5.32	5.14	35.81	36.04	6.93	6.98
Prompt-Con	44.31	41.18	5.30	5.05	36.67	36.16	6.87	6.98
S-FTP	44.57	40.80	5.33	5.18	35.53	36.16	7.07	6.95
T-FTP	45.37	50.31	5.38	4.73	37.41	38.10	7.17	7.75
Non-TTF	45.12	44.11	4.83	4.44	36.14	36.56	7.27	6.92
S-TTF	42.11	50.06	5.21	4.77	35.70	39.05	6.92	7.00
T-TTF	45.86	43.29	5.28	4.82	38.14	40.24	7.41	7.19
ComS2T	**41.12**	**39.88**	**4.38**	**4.35**	**35.32**	**34.48**	**6.82**	**6.90**

The **best results** are bold and the second best are underlined.

Investigation on hippocampus structure: For stage 1, we first remove the overall hippocampus structure and take the architecture without efficient neural disentanglement for OOD inference. We train and update the whole neural architecture with prompts without explicitly identifying the hippocampus and neocortex structures, which helps verify the effectiveness of hippocampus structure. We name such variant as **Non-Hip**. For influence of spatial and temporal blocks, we ablate the neural disentanglement on GCN and retain only stable neuron disentanglement in TCN, and ablate it on TCN with GCN neuron disentangled. We respectively designate the **S-Hip** and **T-Hip** for ablation of removing neural disentanglement for spatial block and temporal block.

Investigation on self-supervised prompt learning: Regarding stage 2, we remove the self-supervision signals of learning both spatial and temporal prompts, and take random initialization to replace the prompt training process. The overall ablative variant is called **Non-SSL**. We also enable spatial prompt to learn with distribution parameters and let temporal prompt be an embedding associated with corresponding timestamps, where we name this variant as **T-SSL**. Similarly, we ablate the learnable spatial prompts with fixed embedding and name it as **S-SSL**. Since how reconstruct the series determines the quality of self-supervised learning, we devise an **SSL-KL** by measuring the series-level distance with KL divergence in self-supervised reconstruction instead of parameterized regression.

Disentangling impacts of w/w.o. Prompts: Regarding stage 3, we construct the ablative variant via updating the hippocampus structure without any spatial-temporal prompts, just with pairwise $\{(\boldsymbol{X}, \boldsymbol{Y})\}$. The variant ablating both spatial and temporal aspects is called **Non-Prompt**. Also, we remove fine-tune process for spatial and temporal blocks as **S-Prompt** and **T-Prompt**. Further, to verify prompt fusion design, we take concatenation to fuse main representations with spatial or temporal prompt, instead of element-wise addition, and name such variant as **Prompt-Con**.

Disentangling impacts of prompt update during testing: Regarding the last stage, we conduct the ablation research without **updating** both spatial and temporal prompts during testing stage, and name this overall ablative variant as **Non-TTF**. Its

TABLE VI
PERFORMANCE COMPARISONS ON MIXED FREEZE/UPDATE VARIANTS AND RANDOM NEURON UPDATES. (METRIC: MAE) THE **BEST RESULTS** ARE BOLD AND THE SECOND BEST ARE UNDERLINED

Datasets	SIP		Metr-LA		KnowAir		Temperature	
Scenarios	Temp shift	Node involve	Temp shift	Node involve	Temp shift	Node involve	Temp shift	Node involve
Fre-Hip, Up-Neo	49.71	40.04	9.23	18.71	36.03	38.05	6.87	7.19
Fre-Hip, Fre-Neo	53.64	53.98	37.28	35.53	40.69	40.39	6.96	6.96
Up-Hip, Up-Neo	50.72	47.79	5.41	5.02	36.77	35.67	6.85	7.12
Up-Hip, Fre-Neo (Equal to ComS2T)	41.12	39.88	4.38	4.35	35.32	34.48	6.82	6.90
Rand-Hip-Fre50%	47.52	41.44	5.34	4.90	36.26	**34.47**	7.16	7.07
Rand-Hip-Fre60%	42.05	42.32	5.18	4.99	36.02	36.52	6.84	7.04
Rand-Hip-Fre70%	42.78	40.43	5.41	5.16	36.17	35.70	6.85	7.33
Rand-Hip-Fre80%	41.22	40.73	5.09	5.26	35.84	35.53	7.04	7.22
Integrated ComS2T (60%,60%,60%,70%)	**41.12**	**39.88**	**4.38**	**4.35**	**35.32**	34.48	**6.82**	**6.90**

(Metric: MAE) The **best results** are bold and the second best are underlined.

variant versions on spatial and temporal prompt are designated as **S-TTF** and **T-TTF** for description.

Investigation on semantic function of hippocampus structure: To validate whether the disentangled 'hippocampus' and 'neocortex' correspondingly responsible for their semantic functions, we implement the experiment for validation, i.e., respectively freezing/updating the hippocampus/neocortex for testing. Moreover, we implement a test to isolate the effectiveness of Efficient Neural Disentanglement, i.e., randomly freezing the same scale of parameters in spatial and temporal learning blocks, which can be viewed as randomly selecting updatable hippocampus neurons, and then observe the performance variation. The proportion of neocortex structures τ is set to range among $\{50\%, 60\%, 70\%, 80\%\}$ for tests. We provide the testing results of mixed matrix and random freezing on Metr-LA and Temperature in Table VI.

Findings on ablation studies: As shown in results of Tables V and VI, our ComS2T clearly beat against all variants and achieves the best performances. First, regarding the ablative variants, the performance experiences a prominent drop when the hippocampus structure is disabled on traffic datasets (SIP and Metr-LA), verifying the exact effectiveness of our complementary architecture. For KnowAir and Temperature, these two datasets are respectively sensitive to prompt description and test-stage adaptation, which are also served as two vital components for data adaptation. The reason for the heterogeneous sensitivity of components to datasets may be the different characteristics of datasets, where traffics are with extensive dynamic patterns, while air quality and climate observations are more regular by with seasonal and location-based prompts. Second, regarding how specified designs on spatial or temporal blocks influence the final generalization, it is reasonable that variants with specific fine-tuning process under spatial or temporal prompts tend to be with more powerful generalization capacity in corresponding scenarios, i.e., remaining spatial fine-tune results in acceptance performances on node involvement, where 'S/T-FTP, S/T-TTF' mostly satisfy such regularity. However, the results are not always supporting such viewpoint (e.g., 'S/T-Hip, S/T-SSL'), and we speculate it can be attributed to that both temporal and spatial blocks are coupled altogether and play important jointed roles for final prediction. Third, on how parameterized regression guides the reconstruction, and how prompt fusion influence results, we find that given two parameters, such regression in self-supervised manner is more easier to learn than KL divergence-based one. And compared to concatenation-based prompt fusion, the element-wise addition can easily complete the fusion without dimension increase and redundancy. To this end, the above results and corresponding analysis verifies the effectiveness and superiority of our original designs. Fourth, when mixingly freezing or updating hippocampus and neocortex structures, the results become inferior to ComS2T, and if we randomly select the neurons as neocortex for freezing, the performances also decrease. These results further verify the rationales of selecting most fluctuated neurons as hippocampus structures while maintaining stable neurons as neocortex. More concretely, the frozen weights fail to fit the OOD patterns of Metr-LA due to extensive gaps of fluctuations and temporal patterns between training and testing sets. On the contrast, with fine-tuning process, ablative variants can consistently obtain moderate performances as they still allow partial parameters updates to fit for new patterns. To conclude, the consistent drops in variants and better performances of ComS2T confirm the designs and intuitions of coupling complementary learning with spatiotemporal forecasting.

G. Hyperparameter Analysis

To test the sensitivity of our ComS2T, we select two crucial hyperparameters to observe how the model behave along with the parameter changes. Our experiments are conducted over temporal distribution shift on all datasets, and empirically optimized hyperparamters are listed as below,

- Percentage accounting for stable neocortex τ, we let it range from $\{50\%, 60\%, 70\%, 80\%\}$.
- The dimensions of spatial and temporal prompts, we consider them as the same dimension E, and let it change within $\{16, 32, 64, 128\}$.

The model performance variations can be found in Fig. 4. For temporal shift scenarios, Metr-LA and KnowAir both achieve best at 60% proportion of neocortex with the prompt dimension of 16. SIP reaches its best with 60% proportion of neocortex and prompt dimension of 64, while Temperature reaches its best with 70% proportion of neocortex and dimension of 32. The higher stable proportion on Temperature suggests that temperature observations are with higher regularity and stability than other urban attributes such as traffics and air quality under temporal shifts. And the larger hidden dimension for SIP demonstrates the potential dynamics of corresponding set requires more fitting capacity of neural networks. Regarding structural shifts, four datasets respectively adapt the preserved stable ratios to 80%, 70%, 50% and 60% at the best performances. For the dimension of prompts, we observe that SIP and Metr-LA both achieve the best performance at the hidden dimension of 128, while KnowAir and Temperature both perform best at the dimension of 16. This is because the dynamics of traffics requires more fitting capacity. In summary, in this subsection, we not only achieve satisfactory results with our hyperparameter studies, but provide insightful urban analysis for further research on cities.

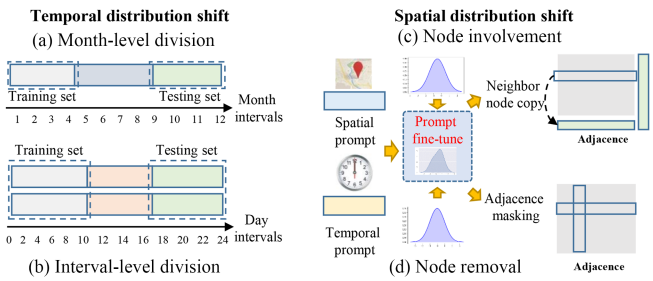


Fig. 3. Experimental settings of four OOD scenarios.

H. Detailed Case Study and Model Exploration

Our case studies are provided to answer the following two **R**esearch **Q**uestions (**RQ**), with empirical and visualized results on specific cases and detailed analysis. **RQ1:** How prompts interpret the dynamic spatial and temporal contexts, and can the prompts adapt to changes in the distribution of main observations? **RQ2:** How the learnable parameters of main ComS2T architecture behave along with the learning process, whether the disentanglement and partial update of neural structures are effective for performance improvement and model generalization.

Visualization on prompts with distinctiveness: We first illustrate the truncated adjacency with 19 nodes from the whole node set on Metr-LA [2] in Fig. 5(a), since it is difficult to visualize all node-level adjacency with limited space. From this subfigure, we observe node 2 and 11 are highly correlated with each other and then we visualize the well-learned spatial and temporal prompts at node 2 and 11 for illustration. The prompts are representations and visualized in the formation of heatmaps, while the corresponding sequential 12-step observations of traffic speed (time series in figure), are plotted with different lines. The selected periods, i.e., 8:00-9:00 a.m. and 21:00-22:00 for visualization are workday morning peak hours and evening hours, where observations at different timestamps are shown as respective two lines separated by a dotted line.

In Fig. 5(b)–(c), it is observed that the two sequences are with distinctive patterns, where the former peak hours experience larger speed variations with underlying high volumes across all steps while the latter one shows relatively stability with time goes on. And also deliver that partial similarity of prompts is also obtained when series reveal similarities, e.g., two nodes share similarity on corresponding peak hours. These visualized intermediate results with reasonable distinctiveness and similarity demonstrate that our self-supervised learning signals can effectively guide the optimization process of prompts and obtain distinguished prompts against different distributions, allowing data adaptation to be delivered forward to update of hippocampus structures.

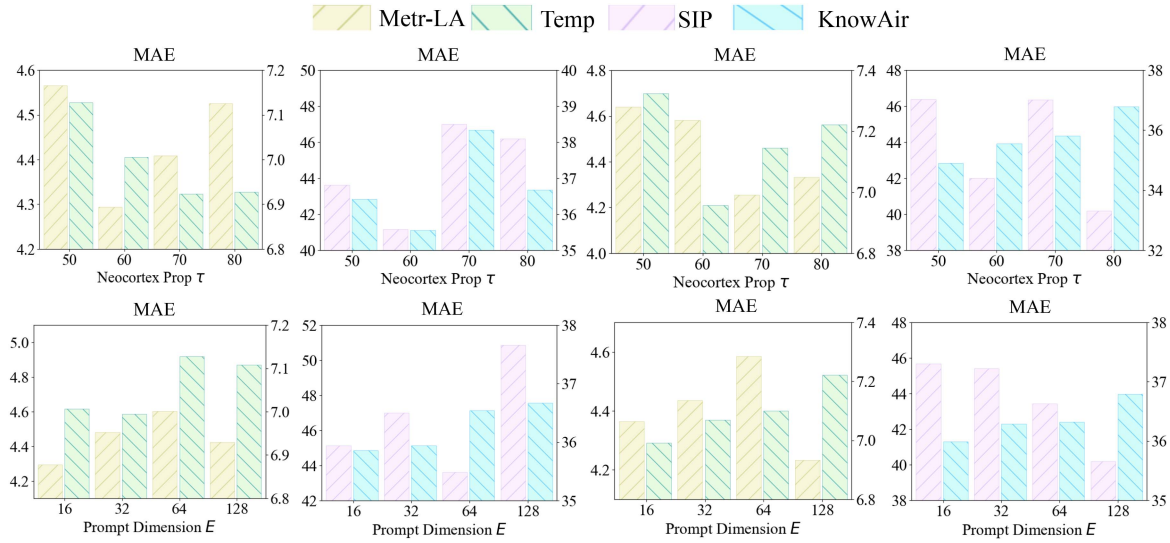
Training process visualization: Second, we illustrate the parameter behaviors and corresponding performance variations along with the training procedure on Metr-LA and Temperature in Figs. 6 to 7. To be specific, the parameter behaviors are demonstrated **in two ways**, i.e., the collective behavior and individual behaviors. 1) We visualize the variations of expectation of the parameters within neocortex and hippocampus neural structures, as collective parameter behaviors, and take performance indicator of MAE errors along with the number of training epochs in Fig. 6. We have marked the knee point of the beginning of prompt-conditioned hippocampus update with dashed line in our Fig. 2) To investigate how individual parameters behave, we visualize fine-grained parameters regarding both spatial and temporal prompts during learning process in Figs. 7 and 8. For collective parameter behaviors, it is observed that the hippocampus structure experiences a heavy fluctuation at the knee point while fluctuation slows down as the learning process continues. For performances, the errors first increase and then decrease to reach the stability, where the increases of errors and fluctuation of parameters reflect the adaptation and adjustment process of neural architecture with incorporation of spatial-temporal prompts. With progressive learning reaching informative prompt signals, the errors are decreasing and eventually outperforming the performance of warm-up stage. For individual ones, the prompt parameters show distinctiveness during each stage, where it also shows inter-stage similarity between warm-up and prompt training stages, as well as between hippocampus update and testing stages. To provide a more intuitive similarity measurement among these prompt parameters, we especially propose to adopt the correlation coefficient for quantifying their relationship. Specifically, we flatten the elements of matrix into the vector, and adopt the Pearson correlation coefficient to measure the similarity between matrices. By denoting the vector in spatial prompt parameters at step t and q as $\boldsymbol{W}_S^t$ and $\boldsymbol{W}_S^q$ where $w(i)_S^t \in \boldsymbol{W}_S^t, w(j)_S^q \in \boldsymbol{W}_S^q$ are elements in respective prompts, we can derive the similarity between them,

$$r(W_S^t, W_S^q) = \frac{\sum_i (w_S^t(i) - \overline{W_S^t})(W_S^q(i) - \overline{W_S^q})}{\sqrt{\sum_i (w_S^t(i) - \overline{W_S^t})^2}\sqrt{\sum_i (w_S^q(i) - \overline{W_S^q})^2}} \tag{28}$$

where $\overline{W_S^t}, \overline{W_S^q}$ are the averaged value of respective vectors. To this end, we can compute the similarity of above four groups of similarity on Metr-LA and Temperature in Fig. 9, and explicitly mark them in Figs. 7 and 8. From above, it suggests that each update process reveals distinguishment between (warm up, prompt training) and (hippocampus update, testing update) stages, while shows similarity within corresponding adjacent pairs. Thus, we can conclude that ComS2T achieves data adaption and the hippocampus updates during two-stage fine-tune process and exactly impose sufficient data-driven impacts for data adaptation.

The above two cases jointly illustrate the distinctiveness of spatial-temporal prompt representation, and the effectiveness of our disentanglement and update process in ComS2T. In essence, the coupling of the prompt-hippocampus for update and generalization, can work cooperatively to construct the mappings between spatial-temporal contexts and the prediction residuals in traditional learning process.

[2] It is a dataset of traffic speed.



(a) Hyperparameter study under temporal shifts (b) Hyperparameter study under spatial shifts

Fig. 4. Hyperparameter study under both spatial and temporal shifts.

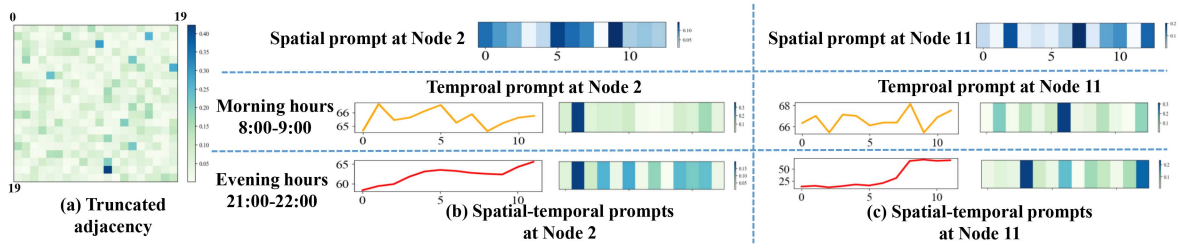


Fig. 5. Visualization of spatial-temporal prompts under different contexts.

TABLE VII
SPACE AND TIME COMPLEXITY COMPARISONS

Models	MAE	Time complexity	Space complexity
GWN	44.17	9.1320 s/epoch	0.26 M
MTGNN	44.25	15.771 s/epoch	0.64 M
CauSTG	43.47	23.451 s/epoch	1.32 M
CaST	43.52	9.3470 s/epoch	0.22 M
TTT-ST	45.45	17.885 s/epoch	1.01 M
ComS2T	**41.12**	**8.0980 s/epoch**	**0.11 M**

Empirical analysis of time and space complexity: We choose several comparative baselines, i.e., having comparable prediction performances with ComS2T, and conduct further experiments on the time complexity and space complexity. Specifically, the time complexity is defined as the time cost per epoch (second/epoch) at model update, and the space complexity is the number of total updated parameters when model updates. The comparison results are provided in Table VII. According to the results, we can observe that our proposed ComS2T arrives the smallest MAE with the least time and space complexity for model update.

V. RELATED WORK

Spatiotemporal learning: Great efforts have been made to empower diverse exciting spatiotemporal applications from traffic prediction [35], [53], [54], [55], [56], [57], environmental modeling [24], [25], [58], to housing price prediction [59], [60]. Among them, various grid convolution [61], [62] or graph convolution networks [5], [16], [63] are devised to capture spatial correlations while temporal convolution [5] or variants of RNN [6], [64], [65], [65], [66], [67] are well-designed to explicitly model temporal dependencies. Actually, the urban elements and city structures are never static, but almost all of existing spatiotemporal models assume the same data distribution between training and testing sets. Therefore, it poses great challenges to maintain same performances with the expansion of cities as well as the increases of vehicles.

OOD generalization on spatiotemporal learning: There are two research lines to counteract the OOD challenges in spatiotemporal forecasting, i.e., continuous learning based and the causal perspective based [28], [33], [34], [35], [36], [37], [68], [69], [70], [71]. Continuous spatiotemporal learning updates model with new data instance with experience reply [34], [35], [69] where they re-train the partial neural architecture by

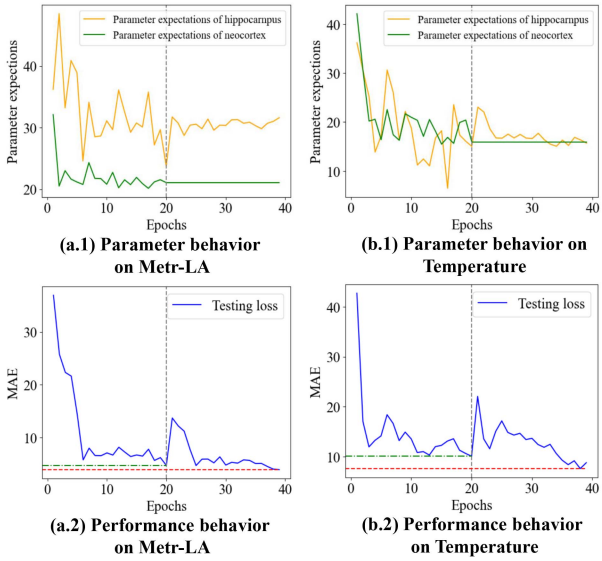


Fig. 6. Learning behavior visualization of ComS2T.

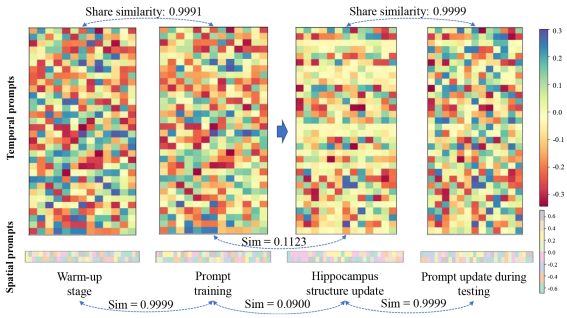


Fig. 7. Fine-grained prompt parameter visualization during learning stages on Metr-LA.

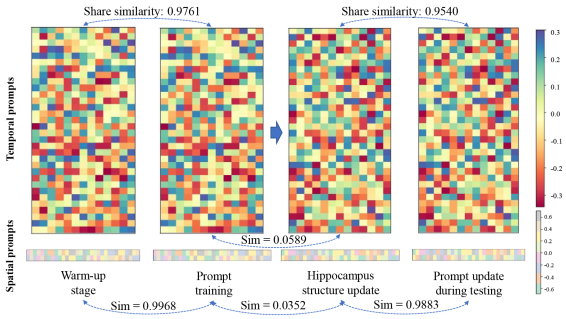


Fig. 8. Fine-grained prompt parameter visualization during learning stages on Temperature.

re-arranging the training set. Specifically, Wang, et al. proposes a historical-data replay strategy, TrafficStream, to update the neural network with all nodes [34], while PECPM dynamically manages a spatiotemporal pattern bank with conflict nodes, which reduces the memory storage burdens [35]. Unfortunately, memory-based methods will inevitably increase the storage space when the network expands and new pattern occurs. To

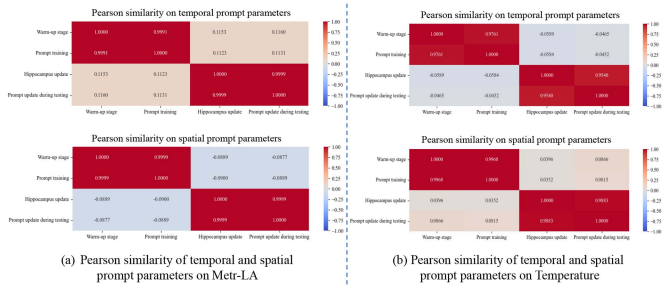


Fig. 9. Mixed matrices for prompt parameter visualization during learning stage on Metr-LA and Temperature.

this end, causal-based learning models temporal environments and captures invariant stable correlations or representations across environments by observing model behaviors. CauSTG reflects complex spatial-temporal dependencies via learnable parameters and transfer the relatively stable weights from historical environment to unseen scenarios [33]. CaST [36] and EAGLE [37] explicitly model the environment by disentangling the environment-aware representation and imitate the OOD scenarios via generating new environments. Even prosperity, CauSTG requires multiple training of same neural architectures, while CaST, EAGLE and sUrban [71] mimic the perturbation and extends the boundary of well-learned data with environment reconstruction-sampling strategy. However, the unseen environments are infinite and the boundary of sample space cannot be unlimitedly extended, thus these solutions are still short of adapting model to brand-new data and environments. To summarize, methods for spatiotemporal OOD generalization either requires high computational resources, or fail to deal with new distribution instances, thus lacking capacity in model evolution.

Learning from neuroscience: Neuroscience is a discipline investigating how human brain works for remembering, learning, and consolidation [72], [73], [74], [75], [76]. Early researches have revealed the similarity between machine learning and human studying behavior, where the neural network is initially developed by imitating the brain neural architectures [77] and it will be activated when the information flow exceeds a fixed threshold [75]. Biological neural networks can be capable of flexibly modulating synaptic plastic to respond to dynamic inputs. The inspired strategies can be summarized as weight regularization [76] from stabilization of previously-learned synaptic changes, memory extension and formation from the number expansion and pruning of functional connections [35], [78], [79], as well as meta-learning, stemmed from activity-dependent synaptic plasticity [11]. However, these methods cannot well interpret what exact to remember and forget along the learning process. As for a collaborative strategy, the complementary learning system (CLS) theory uncovers that regions in brain consist of complementary functions for remembering knowledge, the hippocampus space learns new skills quickly, whereas the neocortex structure progressively learns stable and long-term knowledge [39]. Various machine learning methods are developed from CLS [42], to enable a more generalizable model and

research evidence shows that it shares similarity with continuous learning [80], [81] to adapt multi-task learning. To this end, the structure of the biological brain is sophisticated and informative, exploring the potential structure-reaction of biological brains are valuable to improve machine learning efficiency and potentially address the challenges unsolved currently.

Our work: Rather than taking efforts on series-level pattern computation [34] and pattern expansion [35], or repeated training on divided datasets [33], we couple the complementary learning of neuroscience theory, with spatiotemporal neural architecture to accommodate the streaming and dynamic series observations. This design simultaneously preserves the transferrable stable knowledge from existing data and quickly adapts our model to new arrival instances with new patterns via respectively updating neocortex and hippocampus neural structures. As a result, our ComS2T can efficiently and effectively empower the ST learner with the capacity of evolving with spatiotemporal shifts in a unified framework.

VI. Conclusion and Discussion

In this work, motivated by neuroscience, we couple the complementary learning with spatiotemporal forecasting as a ComS2T, to equip the model with data adaptation and evolution capacity. We first decouple the spatial-temporal learning neural network into two disjoint architectures, stable neocortex and dynamic hippocampus. To enable efficient model evolution, we instantiate additional environments with spatial-temporal prompts to characterize the data distribution and enable prompts learnable with self-supervision. Then we disentangle the neural architecture and incorporate informative prompts into dynamic hippocampus for fine-tuning. ComS2T allows model adaptation conditioned on environment prompts during training stage, thus the fine-tune of prompts can be extended to testing stages when environments change, which empowers model evolution upon new data arrives. Extensive experiments have been conducted on four urban datasets with spatial and temporal shifts. The empirical results demonstrate that ComS2T counteracts the OOD challenges over streaming urban data, improving performances 0.73%$\sim$10.79% and 1.19%$\sim$14.48% respectively under temporal and structural shifts. The substantial visualized case studies illustrate the semantic intermediate results and effective disentanglement learning scheme, enhancing the interpretability of ComS2T.

Discussion of coupling neuroscience and computer science: Neuroscience is a discipline full of mystery and values. Our ComS2T can be an initial practice of incorporating neuroscience into machine learning system, which takes advantage of the learning schemes in both human brain and ANNs. The success of this coupling scheme provides insights into developing more generalizable machine learning systems by investigating interesting and practical mechanisms in neuroscience, e.g., how learning new skills and retrieving consolidated memory interact with each other to improve learning generalization, how to exploit the signal activation scheme to prompt memories. We believe these inherent mechanisms can benefit better designs of both artificial neural architecture and training strategies.

Future works can be divided on two-fold. First, we will continue to improve the spatiotemporal complementary learning by further promoting the training-testing efficiency and tackling semantic alignment between environment and main observations. Second, we are going to find more interesting mechanism in brains such as relations between memory and learning, and facilitate model designs to enable more intelligent machine learning systems countering learning challenges.

Acknowledgment

The AI-driven experiments, simulations and model training were performed on the robotic AI-Scientist platform of Chinese Academy of Sciences.

References

[1] H. Zhang, Y. Zhu, and X. Li, "Decouple graph neural networks: Train multiple simple GNNs simultaneously instead of one," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 11, pp. 7451–7462, Nov. 2024.

[2] H. Zhang, J. Shi, R. Zhang, and X. Li, "Non-graph data clustering via $\mathcal{O}(n)$ bipartite graph convolution," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 7, pp. 8729–8742, Jul. 2023.

[3] Y. Xu, S. Huang, H. Zhang, and X. Li, "Why does dropping edges usually outperform adding edges in graph contrastive learning," 2024, *arXiv:2412.08128*.

[4] Y. Liu et al., "Multivariate time-series forecasting with temporal polynomial graph neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, Art. no. 1411.

[5] Z. Wu, S. Pan, G. Long, J. Jiang, X. Chang, and C. Zhang, "Connecting the dots: Multivariate time series forecasting with graph neural networks," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2020, pp. 753–763.

[6] Q. Huang et al., "CrossGNN: Confronting noisy multivariate time series via cross interaction refinement," in *Proc. 37th Conf. Neural Inf. Process. Syst.*, 2023, Art. no. 2031.

[7] Y. Zheng et al., "DiffUFlow: Robust fine-grained urban flow inference with denoising diffusion model," in *Proc. 32nd ACM Int. Conf. Inf. Knowl. Manage.*, 2023, pp. 3505–3513.

[8] G. Jin, Y. Liang, Y. Fang, J. Huang, J. Zhang, and Y. Zheng, "Spatio-temporal graph neural networks for predictive learning in urban computing: A survey," 2023, *arXiv:2303.14483*.

[9] S. Wang, J. Cao, and S. Y. Philip, "Deep learning for spatio-temporal data mining: A survey," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 8, pp. 3681–3700, Aug. 2022.

[10] B. Lu, X. Gan, W. Zhang, H. Yao, L. Fu, and X. Wang, "Spatio-temporal graph few-shot learning with cross-city knowledge transfer," in *Proc. 28th ACM SIGKDD Conf. Knowl. Discov. Data Mining*, 2022, pp. 1162–1172.

[11] Z. Pan et al., "Spatio-temporal meta learning for urban traffic prediction," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 3, pp. 1462–1476, Mar. 2022.

[12] Y. Liu et al., "iTransformer: Inverted transformers are effective for time series forecasting," in *Proc. 12th Int. Conf. Learn. Representations*, 2023.

[13] J. Dong, H. Wu, H. Zhang, L. Zhang, J. Wang, and M. Long, "SimMTM: A simple pre-training framework for masked time-series modeling," in *Proc. Adv. Neural Inf. Process. Syst.*, 2024, Art. no. 1306.

[14] H. Wu, T. Hu, Y. Liu, H. Zhou, J. Wang, and M. Long, "TimesNet: Temporal 2D-variation modeling for general time series analysis," in *Proc. 11th Int. Conf. Learn. Representations*, 2022.

[15] Y. Liu, H. Wu, J. Wang, and M. Long, "Non-stationary transformers: Rethinking the stationarity in time series forecasting," 2022, *arXiv:2205.14415*.

[16] Z. Zhou, Y. Wang, X. Xie, L. Chen, and C. Zhu, "Foresee urban sparse traffic accidents: A spatiotemporal multi-granularity perspective," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 8, pp. 3786–3799, Aug. 2022.

[17] J. Ji et al., "Spatio-temporal self-supervised learning for traffic flow prediction," in *Proc. AAAI Conf. Artif. Intell.*, 2023, Art. no. 486.

[18] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," in *Proc. Int. Conf. Learn. Representations*, 2018.

[19] Z. Zhou, Y. Wang, X. Xie, L. Chen, and H. Liu, "RiskOracle: A minute-level citywide traffic accident forecasting framework," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 1258–1265.

[20] H. Wu, H. Zhou, M. Long, and J. Wang, "Interpretable weather forecasting for worldwide stations with a unified deep model," *Nature Mach. Intell.*, vol. 5, pp. 602–611, 2023.

[21] R. Castro, Y. M. Souto, E. Ogasawara, F. Porto, and E. Bezerra, "STConvS2S: Spatiotemporal convolutional sequence to sequence network for weather forecasting," *Neurocomputing*, vol. 426, pp. 285–298, 2021.

[22] K. Chen et al., "FengWu: Pushing the skillful global medium-range weather forecast beyond 10 days lead," 2023, *arXiv:2304.02948*.

[23] Y. Zhang et al., "Skilful nowcasting of extreme precipitation with NowcastNet," *Nature*, vol. 619, no. 7970, pp. 526–532, 2023.

[24] Y. Liang et al., "AirFormer: Predicting nationwide air quality in China with transformers," in *Proc. AAAI Conf. Artif. Intell.*, 2023, pp. 14329–14337.

[25] W. Du et al., "Deciphering urban traffic impacts on air quality by deep learning and emission inventory," *J. Environ. Sci.*, vol. 124, pp. 745–757, 2023.

[26] L. Chen, J. Xu, B. Wu, and J. Huang, "Group-aware graph neural network for nationwide city air quality forecasting," *ACM Trans. Knowl. Discov. Data*, vol. 18, no. 3, pp. 1–20, 2023.

[27] J. Li, Q. Sun, H. Peng, B. Yang, J. Wu, and S. Y. Philip, "Adaptive subgraph neural network with reinforced critical structure mining," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 7, pp. 8063–8080, Jul. 2023.

[28] K. Wang et al., "Brave the wind and the waves: Discovering robust and generalizable graph lottery tickets," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 5, pp. 3388–3405, May 2024.

[29] Y. Wu, X. Wang, A. Zhang, X. He, and T.-S. Chua, "Discovering invariant rationales for graph neural networks," in *Proc. Int. Conf. Learn. Representations*, 2021.

[30] Q. Wu, H. Zhang, J. Yan, and D. Wipf, "Handling distribution shifts on graphs: An invariance perspective," in *Proc. Int. Conf. Learn. Representations*, 2022.

[31] S. Li, X. Wang, A. Zhang, Y. Wu, X. He, and T.-S. Chua, "Let invariant rationale discovery inspire graph contrastive learning," in *Proc. Int. Conf. Mach. Learn.*, PMLR, 2022, pp. 13052–13065.

[32] Y. Du et al., "AdaRNN: Adaptive learning and forecasting of time series," in *Proc. 30th ACM Int. Conf. Inf. Knowl. Manage.*, 2021, pp. 402–411.

[33] Z. Zhou et al., "Maintaining the status quo: Capturing invariant relations for OOD spatiotemporal learning," in *Proc. 29th ACM SIGKDD Conf. Knowl. Discov. Data Mining*, 2023, pp. 3603–3614.

[34] X. Chen, J. Wang, and K. Xie, "TrafficStream: A streaming traffic flow forecasting framework based on graph neural networks and continual learning," 2021, *arXiv:2106.06273*.

[35] B. Wang et al., "Pattern expansion and consolidation on evolving graphs for continual traffic prediction," in *Proc. 29th ACM SIGKDD Conf. Knowl. Discov. Data Mining*, 2023, pp. 2223–2232.

[36] Y. Xia et al., "Deciphering spatio-temporal graph forecasting: A causal lens and treatment," in *Proc. 37th Conf. Neural Inf. Process. Syst.*, 2023, Art. no. 1611.

[37] H. Yuan et al., "Environment-aware dynamic graph learning for out-of-distribution generalization," in *Proc. 37th Conf. Neural Inf. Process. Syst.*, 2023, Art. no. 2164.

[38] R. C. O'Reilly, R. Bhattacharyya, M. D. Howard, and N. Ketz, "Complementary learning systems," *Cogn. Sci.*, vol. 38, no. 6, pp. 1229–1248, 2014.

[39] D. Kumaran, D. Hassabis, and J. L. McClelland, "What learning systems do intelligent agents need? Complementary learning systems theory updated," *Trends Cogn. Sci.*, vol. 20, no. 7, pp. 512–534, 2016.

[40] J. L. McClelland, B. L. McNaughton, and A. K. Lampinen, "Integration of new information in memory: New insights from a complementary learning systems perspective," *Philos. Trans. Roy. Soc. B*, vol. 375, no. 1799, 2020, Art. no. 20190637.

[41] C. S. Lee and A. Y. Lee, "Clinical applications of continual learning machine learning," *Lancet Digit. Health*, vol. 2, no. 6, pp. e279–e281, 2020.

[42] E. Arani, F. Sarfraz, and B. Zonooz, "Learning fast, learning slow: A general continual learning method based on complementary learning system," in *Proc. Int. Conf. Learn. Representations*, 2022.

[43] R. Bertram, A. Sherman, and E. F. Stanley, "Single-domain/bound calcium hypothesis of transmitter release and facilitation," *J. Neuriophysiol.*, vol. 75, no. 5, pp. 1919–1931, 1996.

[44] W. S. Grant, J. Tanner, and L. Itti, "Biologically plausible learning in neural networks with modulatory feedback," *Neural Netw.*, vol. 88, pp. 32–48, 2017.

[45] J. L. McClelland, "Incorporating rapid neocortical learning of new schema-consistent information into complementary learning systems theory," *J. Exp. Psychol.: Gen.*, vol. 142, no. 4, pp. 1190–1210, 2013.

[46] H. Guo, T. Ruiming, Y. Ye, Z. Li, and X. He, "DeepFM: A factorization-machine based neural network for CTR prediction," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, 2017, pp. 1725–1731.

[47] S. Wang, Y. Li, J. Zhang, Q. Meng, L. Meng, and F. Gao, "PM2.5-GNN: A domain knowledge enhanced graph neural network for PM2.5 forecasting," in *Proc. 28th Int. Conf. Adv. Geographic Inf. Syst.*, 2020, pp. 163–166.

[48] M. A. Hameed, O. Al Jadaan, and S. Ramachandram, "Collaborative filtering based recommendation system: A survey," *Int. J. Comput. Sci. Eng.*, vol. 4, no. 5, pp. 859–876, 2012.

[49] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang, "Graph wavenet for deep spatial-temporal graph modeling," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, 2019, pp. 1907–1913.

[50] S. Wang et al., "TimeMixer: Decomposable multiscale mixing for time series forecasting," in *Proc. 12th Int. Conf. Learn. Representations*, 2024.

[51] A. Cini, D. Mandic, and C. Alippi, "Graph-based time series clustering for end-to-end hierarchical forecasting," in *Proc. Int. Conf. Mach. Learn.*, PMLR, 2024, pp. 8985–8999.

[52] C. Chen, Y. Liu, L. Chen, and C. Zhang, "Test-time training for spatial-temporal forecasting," in *Proc. SIAM Int. Conf. Data Mining*, SIAM, 2024, pp. 463–471.

[53] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan, "Attention based spatial-temporal graph convolutional networks for traffic flow forecasting," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 922–929.

[54] S. Wang, H. Miao, H. Chen, and Z. Huang, "Multi-task adversarial spatial-temporal networks for crowd flow prediction," in *Proc. 29th ACM Int. Conf. Inf. Knowl. Manage.*, 2020, pp. 1555–1564.

[55] 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.*, vol. 36, no. 1, pp. 62–76, Jan. 2024.

[56] Y. Zheng, L. Capra, O. Wolfson, and H. Yang, "Urban computing: Concepts, methodologies, and applications," *ACM Trans. Intell. Syst. Technol.*, vol. 5, no. 3, pp. 1–55, 2014.

[57] Y. Liang et al., "Revisiting convolutional neural networks for citywide crowd flow analytics," in *Proc. Eur. Conf. Mach. Learn. Knowl. Discov. Databases*, Ghent, Belgium, Springer, 2021, pp. 578–594.

[58] F. Amato, F. Guignard, S. Robert, and M. Kanevski, "A novel framework for spatio-temporal prediction of environmental data using deep learning," *Sci. Rep.*, vol. 10, no. 1, 2020, Art. no. 22243.

[59] X. Liu, "Spatial and temporal dependence in house price prediction," *J. Real Estate Finance Econ.*, vol. 47, no. 2, pp. 341–369, 2013.

[60] P. Wang, C. Ge, Z. Zhou, X. Wang, Y. Li, and Y. Wang, "Joint gated co-attention based multi-modal networks for subregion house price prediction," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 2, pp. 1667–1680, Feb. 2023.

[61] 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.

[62] J. Ye, L. Sun, B. Du, Y. Fu, X. Tong, and H. Xiong, "Co-prediction of multiple transportation demands based on deep spatio-temporal neural network," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2019, pp. 305–313.

[63] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," in *Proc. Int. Joint Conf. Artif. Intell.*, 2018, pp. 3634–3640.

[64] L. Bai, L. Yao, C. Li, X. Wang, and C. Wang, "Adaptive graph convolutional recurrent network for traffic forecasting," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2020, Art. no. 1494.

[65] Y. Wang et al., "PredRNN: A recurrent neural network for spatiotemporal predictive learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 2, pp. 2208–2225, Feb. 2023.

[66] Z. Yao, Y. Wang, H. Wu, J. Wang, and M. Long, "ModeRNN: Harnessing spatiotemporal mode collapse in unsupervised predictive learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 11, pp. 13281–13296, Nov. 2023.

[67] Y. Wang, Z. Gao, M. Long, J. Wang, and S. Y. Philip, "PredRNN: Towards a resolution of the deep-in-time dilemma in spatiotemporal predictive learning," in *Proc. Int. Conf. Mach. Learn.*, PMLR, 2018, pp. 5123–5132.

[68] Y. Yuan, C. Shao, J. Ding, D. Jin, and Y. Li, "A generative pre-training framework for spatio-temporal graph transfer learning," 2024, *arXiv:2402.11922*.

[69] G. I. Parisi, J. Tani, C. Weber, and S. Wermter, "Lifelong learning of spatiotemporal representations with dual-memory recurrent self-organization," *Front. Neurorobot.*, vol. 12, pp. 78, 2018.
[70] Y. Yuan, J. Ding, J. Feng, D. Jin, and Y. Li, "UniST: A prompt-empowered universal model for urban spatio-temporal prediction," in *Proc. 30th ACM SIGKDD Conf. Knowl. Discov. Data Mining*, 2024, pp. 4095–4106.
[71] Q. Wang, B. Guo, L. Cheng, and Z. Yu, "sUrban: Stable prediction for unseen urban data from location-based sensors," *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 7, no. 3, pp. 1–20, 2023.
[72] G. W. Lindsay, "Attention in psychology, neuroscience, and machine learning," *Front. Comput. Neurosci.*, vol. 14, pp. 29, 2020.
[73] M. W. Mathis and A. Mathis, "Deep learning tools for the measurement of animal behavior in neuroscience," *Curr. Opin. Neurobiol.*, vol. 60, pp. 1–11, 2020.
[74] A. Bessadok, M. A. Mahjoub, and I. Rekik, "Graph neural networks in network neuroscience," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 5, pp. 5833–5848, May 2023.
[75] N. Gupta et al., "Artificial neural network," *Netw. Complex Syst.*, vol. 3, no. 1, pp. 24–28, 2013.
[76] F. Zenke, B. Poole, and S. Ganguli, "Continual learning through synaptic intelligence," in *Proc. Int. Conf. Mach. Learn.*, PMLR, 2017, pp. 3987–3995.
[77] J. Zou, Y. Han, and S.-S. So, "Overview of artificial neural networks," in *Proc. Artif. Neural Netw.: Methods Appl.*, 2009, pp. 14–22.
[78] J. Kirkpatrick et al., "Overcoming catastrophic forgetting in neural networks," *Proc. Nat. Acad. Sci. USA*, vol. 114, no. 13, pp. 3521–3526, 2017.
[79] G. M. Van de Ven, H. T. Siegelmann, and A. S. Tolias, "Brain-inspired replay for continual learning with artificial neural networks," *Nature Commun.*, vol. 11, no. 1, 2020, Art. no. 4069.
[80] L. Wang et al., "Incorporating neuro-inspired adaptability for continual learning in artificial intelligence," *Nature Mach. Intell.*, vol. 5, pp. 1356–1368, 2023.
[81] L. Wang, X. Zhang, H. Su, and J. Zhu, "A comprehensive survey of continual learning: Theory, method and application," 2023, *arXiv:2302.00487*.

Zhengyang Zhou (Member, IEEE) received the PhD degree from the University of Science and Technology of China, in 2023. He is now an associate researcher with Suzhou Institute for Advanced Research, University of Science and Technology of China (USTC). He has published more than 30 papers on top conferences and journals, such as ICML, NeurIPS, ICLR, KDD, *IEEE Transactions on Knowledge and Data Engineering*, WWW, and AAAI. His mainly research interests include spatiotemporal data minining and human-centered urban computing. He is now especially interested in improving model generalization capacity for streaming and spatiotemporal data.

Qihe Huang received the BS degree from the Nanjing University of Information Science and Technology, in 2022. He is currently working toward the PhD degree with the University of Science and Technology of China. His research centers on spatiotemporal data mining and time-series forecasting. He has published more than 10 top journals and conferences on time-series learning and spatiotemporal intelligence including NeurIPS, ICML, ICLR, WWW, AAAI, and IJCAI.

Binwu Wang is now an associate researcher with Suzhou Institute for Advanced Research, University of Science and Technology of China (USTC). His research interests include spatial-temporal data mining and human-centered urban computing. He has published more than 10 refereed journal and conference papers in the field of data mining, including AAAI, ICLR, *IEEE Transactions on Intelligent Transportation Systems*, DASFAA, IEEE ICDM, WSDM, etc.

Jianpeng Hou received the BS degree from Shandong University, in 2023. He is currently working toward the MS degree with the University of Science and Technology of China. His research centers on OOD generalization.

Kuo Yang received the BE degree from Northeastern University at Qinhuangdao, China, in 2021. He is now working toward the PhD degree with the School of Data Science, USTC. His mainly research interests are graph representation learning, spatiotemporal data mining and especially subgraph-driven spatiotemporal graph learning.

Yuxuan Liang (Member, IEEE) received the PhD degree from NUS. He is an assistant professor with Intelligent Transportation Thrust, Hong Kong University of Science and Technology (Guangzhou). He is currently working on the research, development, and innovation of spatio-temporal data mining and AI, with a broad range of applications in smart cities. He published more than 40 peer-reviewed papers in refereed journals and conferences, such as KDD, WWW, NeurIPS, ICLR, ECCV, AAAI, IJCAI, Ubicomp, and *IEEE Transactions on Knowledge and Data Engineering*. Those papers have been cited more than 2,100 times (Google Scholar H-Index: 21). He was recognized as 1 out of 10 most innovative and impactful PhD students focusing on data science in Singapore by Singapore Data Science Consortium (SDSC).

Yu Zheng (Fellow, IEEE) is the vice president of JD.COM and head JD Intelligent Cities Research. Before Joining JD.COM, he was a senior research manager with Microsoft Research. He currently serves as the editor-in-chief of *ACM Transactions on Intelligent Systems and Technology* and has served as the program co-chair of ICDE 2014 (Industrial Track), CIKM 2017 (Industrial Track) and IJCAI 2019 (industrial track). He is also a keynote speaker of AAAI 2019, KDD 2019 Plenary Keynote Panel and IJCAI 2019 Industrial Days. His monograph, entitled Urban Computing, has been used as the first text book in this field. In 2013, he was named one of the Top Innovators under 35 by MIT Technology Review (TR35) and featured by Time Magazine for his research on urban computing. In 2016, he was named an ACM distinguished scientist and elevated to an IEEE fellow, in 2020 for his contributions to spatio-temporal data mining and urban computing.

Yang Wang (Senior Member, IEEE) received the PhD degree from the University of Science and Technology of China, in 2007. He is now a full professor with the School of Computer Science and Technology, School of Software Engineering, and School of Data Science in USTC. Since then, he keeps working with USTC till now as a postdoc and an associate professor successively. Meanwhile, he also serves as the vice dean of the School of Software Engineering, USTC. His research interest mainly includes wireless (sensor) networks, data mining, and machine learning, and he is also interested in all kinds of applications of AI and data mining technologies especially in urban computing and AI4Science.