Unveiling Concept Shift via Spatio-Temporal State Learning

Kuo Yang
University of Science and Technology of China
Hefei, China
yangkuo@mail.ustc.edu.cn

Qihe Huang
University of Science and Technology of China
Hefei, China
hqh@mail.ustc.edu.cn

Zhengyang Zhou*
University of Science and Technology of China (USTC)
Hefei, China
Suzhou Institute for Advanced Research, USTC
Suzhou, China
zzy0929@ustc.edu.cn

Abstract

Dynamic graphs are ubiquitous in the real world, presenting the temporal evolution of individuals within spatial associations. Recently, dynamic graph learning research is flourishing, striving to more effectively capture evolutionary patterns and spatial correlations. However, existing methods still fail to address the issue of concept shift in dynamic graphs. Concept shift manifests as a distribution shift in the mapping pattern between historical observations and future evolution. The reason is that some environment variables in dynamic graphs exert varying effects on evolution patterns, but these variables are not effectively captured by the models, leading to the intractable concept shift issue. To tackle this issue, we propose a **St**a**te**-driven environ**men**t inference framework (**Samen**) to achieve a dynamic graph learning framework equipped with concept generalization ability. Firstly, we propose a two-stage environment inference and compression strategy. From the perspective of state space, we introduce a prefix-suffix collaborative state learning mechanism to bidirectionally model the spatio-temporal states. A hierarchical state compressor is further designed to refine the state information resulting in concept shift. Secondly, we propose a skip-connection spatio-temporal prediction module, which effectively utilizes the inferred environments to improve the model's generalization capability. Finally, we select seven datasets from different domains to validate the effectiveness of our model. By comparing the performance of different models on samples with concept shift, we verify that our **Samen** gains generalization capacity that existing methods fail to capture.

CCS Concepts

• **Computing methodologies** → **Temporal reasoning**; **Learning latent representations**; • **Mathematics of computing** → **Graph algorithms**.

*Corresponding author: Zhengyang Zhou

WWW '25, April 28–May 2, 2025, Sydney, NSW, Australia.

ACM ISBN 978-1-4503-XXXX-X/18/06
https://doi.org/10.1145/XXXXXX.XXXXXX

Keywords

Dynamic graph learning, Spatio-temporal neural network, Concept shift, State space model.

ACM Reference Format:
Kuo Yang, Qihe Huang, and Zhengyang Zhou. 2025. Unveiling Concept Shift via Spatio-Temporal State Learning. In *Proceedings of the ACM Web Conference 2025 (WWW '25), April 28–May 2, 2025, Sydney, NSW, Australia.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/XXXXXX.XXXXXX

1 Introduction

Dynamic graphs are ubiquitous in the real world, encompassing social networks [3, 38], web-based platforms [10, 28], and traffic networks [23, 52]. Unlike static graphs, dynamic graphs involve the changes in nodes and edges over time, presenting a significant challenge in simultaneously modeling temporal evolution and spatial correlations [4, 51]. Recently, the spatio-temporal learning of coupled Graph Neural Networks and Time Series Models has started to flourish [42, 47, 48], gaining increasing improvement for its ability to effectively model both spatial dependencies and temporal dynamics.

Although existing dynamic graph learning frameworks are theoretically proven to effectively capture spatio-temporal correlations, modeling real-world dynamic graphs is challenging due to the constantly changing evolutionary patterns influenced by various complex factors [48, 50]. These factors include human noise, irregular patterns, and unpredictable external conditions, all of which can significantly disturb the evolution patterns [37, 40, 51].

One of the major challenges arising from changes in evolution patterns is the issue of concept shift. This issue manifests as similar or identical historical observations leading to different future evolutions as shown in the right panel of Fig. 1 [30, 36]. Actually, this phenomenon is widely present in real-world datasets [30]. Empirically, we observe that the proportion of samples exhibiting concept shift in the Yelp [27] and SD-2019 [23] datasets reached as high as 24.3% and 32.9%, respectively. The widespread existence of this phenomenon poses significant challenges for dynamic graph learning methods. As shown in the left panel of Fig. 1, we filter the samples that exhibit concept shift, and utilize them to evaluate existing dynamic graph learning frameworks. This distinct contrast highlights that concept shift is indeed a significant factor contributing to the lack of generalization in current dynamic graph learning models. Therefore, enhancing the model's ability to address the concept shift issue has become an urgent challenge to resolve.

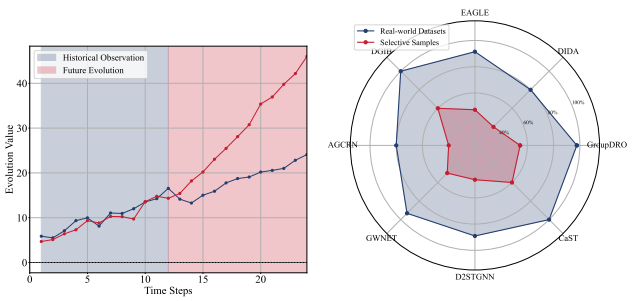

Figure 1: Our research motivation. *Left Panel*: a sample selected from the SD2019 dataset exhibiting concept shift. *Right Panel*: the performance discount comparison of existing models on the samples with a concept shift issue.

Unlike data distribution shift, which appears as discrepancies between the distributions of the training and test datasets [5, 7, 21, 34], concept shift is characterized by the shifts in the mapping relationship from data to its label. Some environment variables in dynamic graphs exert a varying effect on evolution patterns, and the failure of models to capture those information exacerbates this issue. In other words, some key environment variables often remain inactive in historical observations, but their emergence in future observations leads to significant changes in evolutionary trends. As shown in Fig. 2, clear weather often does not alter human interest patterns, resulting in their inactivity. However, as winter comes, previously inactive weather variables become activated, leading to significant shifts in human interests. Therefore, identifying environment variables with both inactive and active properties is a crucial step in addressing the concept shift.

In this work, we propose a state-driven dynamic graph learning framework to address the issue of concept shift. **Firstly**, from the perspective of information theory [1, 16], we theoretically investigate how to infer the environment variables that cause concept shift issue. Further, we summarize two practical guidelines to model the environment variables. **Secondly**, we introduce the insight of spatio-temporal state to uniformly model all potential environment variables in dynamic graphs. By designing a prefix-suffix state learning mechanism, we can bidirectionally capture environment variables. **Thirdly**, to refine the environments leading to concept shift, we propose a hierarchical state compressor to infer those pivotal environments with both inactive and active properties. By constructing hierarchical state learning tasks, we can maximize the extraction of variables from historical observations that may alter evolutionary patterns. **Finally,** we design a skip-connection state prediction framework to alleviate the limitations faced by parameterized models in addressing concept shift. **Empirically**, we select seven datasets from different domains to validate the effectiveness of our model. **Samen** achieves better results than all baselines on most datasets. Moreover, we filter all samples exhibiting concept shift from two cross-domain datasets. By comparing the performance of different methods on those samples, we corroborate the excellent generalization in addressing the concept shift issue. Our contributions are as follows:

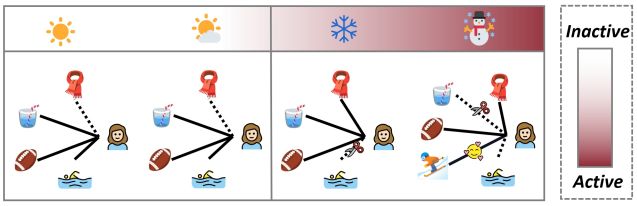

Figure 2: Human interests are influenced by the environments. Weather variables remain inactive in historical observations but become active in future observations.

- We conduct a theoretical analysis of the variational upper bounds of parameterized estimation methods to address the concept shift issue. Based on this, we introduce two primary practical guidelines to address this challenge.
- We model environment variables from the perspective of state space, and propose a two-stage state modeling strategy to infer the key states that contribute to concept shift. Technically, we introduce a novel prefix-suffix collaborative state learning mechanism, and a hierarchical state compressor to infer the key states.
- We design extensive experiments to validate the superior generalization in concept shift scenarios.

2 Preliminaries and Backgrounds

2.1 Preliminaries

Notation. We define a temporal graph with N nodes over T steps as $\mathbf{TG} = \{\mathcal{G}^t\}_{t=1}^{T}$. Each graph snapshot at time step t is indicated as $\mathcal{G}^t = (X^t, A^t)$, where $X^t \in \mathbb{R}^{N\times d}$ denotes the node-wise features, and $A^t \in \mathbb{R}^{N\times N}$ indicates the connections between nodes, where d uniformly represents features and representations dimensions.

Problem definition. Temporal graphs learning focuses on summarizing the evolutionary patterns from historical snapshots and predicting the future trends at subsequent K steps. Specific tasks include future node properties forecast and link prediction. With the historical observations $\mathbf{X} = \mathcal{G}^{1:T}$ as input, the temporal graph neural network learns future evolution patterns $\mathbf{Y} = \mathcal{G}^{T+1:T+K}$, aiming to predict $\widehat{\mathbf{Y}} \leftarrow \mathbf{X}$. Our primary objective is to achieve high-quality future predictions.

2.2 Backgrounds

Concept shift. Different from data distribution shift issue which has been widely studied [5, 7, 21, 40, 42, 45, 48], concept migration has different characteristics and reasons. Data distribution shift issue denotes the differences between the training and test data distributions, i.e., $\mathbb{P}(\mathbf{TG}_{train}) \neq \mathbb{P}(\mathbf{TG}_{te})$ [5, 7, 21]. However, concept shift is manifested in changes to the mapping distribution between the data and its labels,

$$\mathbb{P}(\kappa) \neq \mathbb{P}(\kappa'), \tag{1}$$

where $\mathbf{X} \xrightarrow{\kappa} \mathbf{Y}$ and $\mathbf{X} \xrightarrow{\kappa'} \mathbf{Y}'$. In other words, concept shift is an intra-sample phenomenon, whereas data distribution shift reflects inter-sample differences. Therefore, the differing causes of distribution

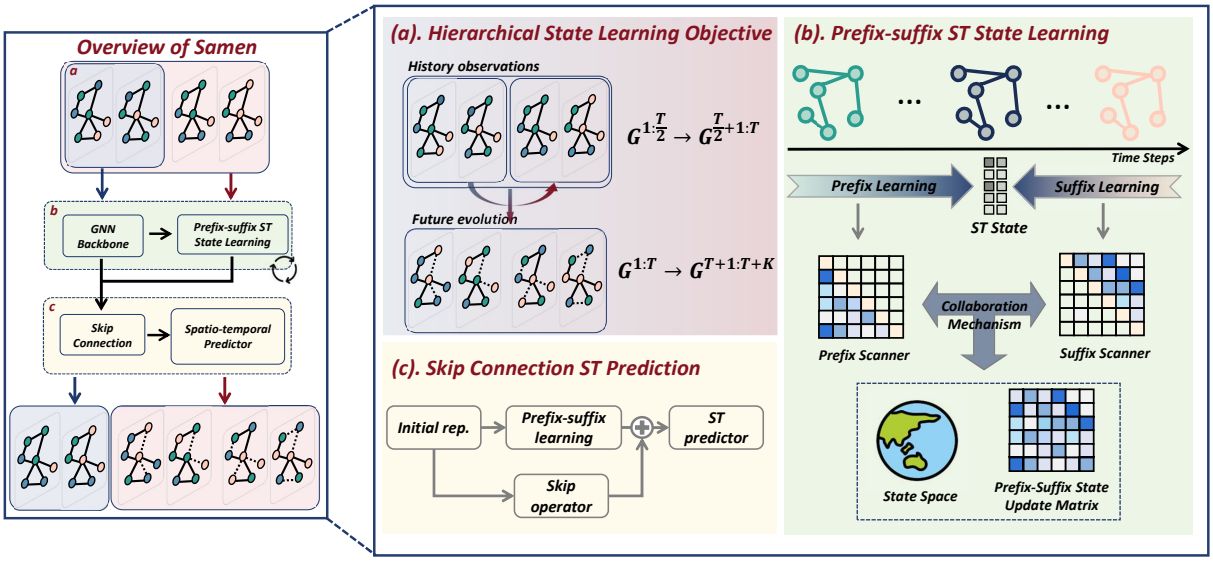

Figure 3: *Left Panel*: the overview of Samen. *Right Panel*: the detailed pipeline of Samen. Based on a hierarchical state learning objective, Samen mainly includes a prefix-suffix collaborative state learning mechanism and a skip-connection spatio-temporal prediction module.

shifts lead to distinct technical approaches for addressing these issues. The key to resolving data distribution shift lies in discovering the common feature among samples (invariant learning technique) [40, 51], while addressing concept shift involves uncovering additional features within the samples to bridge the gaps in the shifted mapping relationships (environment inference technique). In this work, we focus on a theoretical and practical investigation of the environment variables that contribute to concept shift from the perspective of state space.

State Space Model (SSM). SSM is a mathematical framework used to describe the dynamic evolution of a system [11–13, 32, 49]. It is widely applied in areas such as signal processing, control systems, and time series analysis. By utilizing a set of state variables, SSM represents the internal state of a system at any given point, allowing for the prediction of the system's future behavior or observations [17]. Most practices of SSM consist of two main components: the state transition equation (Eq. 2), which updates state based on the system evolves, and the output equation (Eq. 3), which relates the updated states to the final output.

$$h'(t) = Ah(t) + Bx(t), \tag{2}$$

$$y(t) = Ch'(t) + Dx(t), \tag{3}$$

where A, B, C, D are learnable parameters most practices [13], $h(t)$ and $x(t)$ denotes the state and input at step t respectively.

By integrating temporal dynamics and observational data, SSM can effectively captures long-term correlation. Empirically , SSM practices have achieved some outstanding performance, such as Mamba [11]. We argue that this success typically stems from two main objectives: the first is to efficiently learn prefix states, and the second is to adaptively adjust the update weight based on the actual sequence information. Formally, given a binary associative prefix operator $\circ$ (i.e. $(a \circ b) \circ c = a \circ (b \circ c)$) and a sequence of T graphs $[\mathcal{G}^1, \mathcal{G}^2, \cdots, \mathcal{G}^T]$, the prefix scanner of SSM is fed with historical observations to return the state sequences,

$$[G^1, (\mathcal{G}^1 \circ \mathcal{G}^2), \cdots, (G^1 \circ \mathcal{G}^2 \circ \cdots \circ \mathcal{G}^T)]. \tag{4}$$

Our proposed state-driven environment inference framework is built upon the foundation of SSM, but it is strictly different from existing works. Specifically, we observe that the state in dynamic graphs exhibit a unique duality. This requires that comprehensive state modeling must incorporate the cooperative learning mechanism of both prefix and suffix states.

3 Related Work

Dynamic Graph Learning. The widespread existence of spatio-temporal graphs in the real world has led to the rapid growth of dynamic graph learning. Dynamic graph models collaboratively use Graph Neural Networks (GNNs) [9, 43] to learn spatial structures and time series models [15, 46] to capture temporal evolution. In order to effectively capture the temporal evolution of spatially structured individuals, various works focus on studying temporal and spatial information in a decoupled manner [2, 20, 22, 25, 29, 31, 41, 44, 50]. These studies are based on the assumption that the training dateset and the test dateset have the same distribution, and there exists a consistent mapping distribution in different data. However, this assumption is often not satisfied in real-world data.

Generalization of Dynamic Graph Models. The issue of generalization is a significant challenge currently faced by dynamic graph learning methods. Numerous studies have focused on developing dynamic graph learning approaches with stronger generalization capabilities. Existing work primarily focuses on addressing out-of-distribution (OOD) generalization, i.e., $\mathbb{P}(\mathbf{TG}_{train}) \neq \mathbb{P}(\mathbf{TG}_{te})$ [5, 7, 21]. DIDA [48] is one of the earliest studies on the issue of OOD generalization on dynamic graphs, which involves achieve robust

spatio-temporal prediction by identifying invariant patterns. Xia et al. [42] propose causal spatio-temproal neural network termed CaST that performs the back-door adjustment and front-door adjustment to resolve temporal OOD issue. Yuan et al. [48] introduce a novel framework EAGLE for OOD generalization on dynamic graphs by modeling complex dynamic environments and exploiting spatio-temporal invariant patterns. Unlike existing works, we focus on concept shift in dynamic graphs. Concept shift is manifested in changes to the mapping distribution between the data and its labels $\mathbb{P}(\kappa) \neq \mathbb{P}(\kappa')$, where $\mathbf{X} \xrightarrow{\kappa} \mathbf{Y}$ and $\mathbf{X} \xrightarrow{\kappa'} \mathbf{Y}'$. In this work, we focus on solving the concept shift problem on dynamic graphs.

State Space Model. The State Space Model (SSM) is a mathematical model used to describe dynamic systems, which has a long history of development [39]. With the development of deep learning, SSM has been widely used in many time series scenarios. Rangapuram et al. [24] proposed a deep state space model to improve the performance of time series prediction. The HiPPO framework [12] introduces a way to effectively capture long-term dependencies in sequential data using a memory mechanism that integrates polynomial projections. LSSL [14] is proposed to achieve a discretization learning framework for continuous time SSM. SSM-S4 [13] is introduced to efficient model long sequences using structured state space models. SSM-S5 [32] modifies the internalstructure of the SSM-S4 layer, and replaces the frequency-domain approach used by SSM-S4 with a purely recurrent, time-domain approach leveraging parallel scans. Mamba [11] has the characteristics of adaptive selectivity and parallel scanning with higher efficiency and performance. All variants of SSM solely focus on learning prefix states. Based on a thorough analysis and understanding of dynamic graphs in the real world, we observe the duality of spatio-temporal states. This insight inspires us to propose a prefix-suffix collaborative state learning model.

4 Upper Bound for Tackling Concept Shift

The issue of concept shift in dynamic graphs fundamentally arises from the presence of unseen environment variables. This gap created by such limited data is often difficult to bridge through the design of network frameworks. Therefore, understanding the upper bound of the model in addressing concept shift is primary for tackling this challenge.

We first define the ground-truth distribution of historical observations $\mathbf{X}$ and future evolution $\mathbf{Y}$ as $\mathbb{Q}_{\mathbf{X}}$ and $\mathbb{Q}_{\mathbf{Y}}$. Our goal is to obtain an approximate $\hat{\mathbb{Q}}$, which can align the evolution patterns of $\mathbb{Q}_{\mathbf{X}}$ and $\mathbb{Q}_{\mathbf{Y}}$. This objective can be formalized as,

$$\hat{\mathbb{Q}} = \arg\min_{\hat{\mathbb{Q}}} -I(\mathbf{X}, \mathbf{E}; \mathbf{Y}). \tag{5}$$

To obtain this estimate, we need to understand the role of the environment variable $\mathbf{E}$ in historical observations $\mathbf{X}$ and future evolution $\mathbf{Y}$. Considering the characteristics of concept shift, the sudden emergence of environment variables in future evolution is the key factor that renders the model ineffective. In other words, in a history-future observations without concept shift, the environment variable still exists, but they remain hidden throughout the entire process. Therefore, the objective of Eq. 5 can be updated as,

$$\hat{\mathbb{Q}} = \arg\min_{\hat{\mathbb{Q}}} -I(\mathbf{X}, \mathbf{E}; \mathbf{Y}) - I(\mathbf{E}; \mathbf{Y}) + I(\mathbf{X}; \mathbf{E}). \tag{6}$$

It describes that the estimation of environment distribution $\hat{\mathbb{Q}}$ should focus on capturing the different variable components present in historical observations $\mathbf{E}$ and future evolution $\mathbf{Y}$. To optimize this objective controllably, we introduce a variational approximation $\mathbb{P}_\theta(\mathbf{Y}|\mathbf{E})$ for $\mathbb{P}(\mathbf{Y}|\mathbf{E})$, and a variational approximation $\mathbb{P}_\phi(\mathbf{E}|\mathbf{X})$ for $\mathbb{P}(\mathbf{E}|\mathbf{X})$. We can derive each term of Eq. 6 as follows.

For the first term $I(\mathbf{X}, \mathbf{E}; \mathbf{Y})$ of Eq. 6, there exists,

$$I(\mathbf{X}, \mathbf{E}; \mathbf{Y}) = \mathbb{E}_{\mathbf{X},\mathbf{Y},\mathbf{E}}\left[\log \frac{\mathbb{P}(\mathbf{Y}|\mathbf{E}, \mathbf{X})}{\mathbb{P}(\mathbf{Y})}\right]. \tag{7}$$

Considering the parameterized variational estimations, we can obtain the upper bound,

$$I(\mathbf{X}, \mathbf{E}; \mathbf{Y}) \geq \mathbb{E}_{\mathbf{X},\mathbf{E}}\left[\log \mathbb{P}(\mathbf{Y}|\mathbf{X}, \mathbf{E})\right] + H(\mathbf{Y}). \tag{8}$$

For the second term $I(\mathbf{E}; \mathbf{Y})$ of Eq. 6, there exists

$$I(\mathbf{E}; \mathbf{Y}) = \mathbb{E}_{\mathbf{E},\mathbf{Y}}\left[\log \frac{\mathbb{P}(\mathbf{Y}|\mathbf{E})}{\mathbb{P}(\mathbf{Y})}\right]. \tag{9}$$

Based on the parameterized variational estimations, we obtain the lower bound

$$I(\mathbf{E}; \mathbf{Y}) \geq \mathbb{E}_{\mathbf{E},\mathbf{Y}}\left[\log \mathbb{P}_\theta(\mathbf{Y}|\mathbf{E})\right] + H(\mathbf{Y}). \tag{10}$$

For the third term $I(\mathbf{X}; \mathbf{E})$ of Eq. 6, there exists

$$I(\mathbf{X}; \mathbf{E}) = \mathbb{E}_{\mathbf{X},\mathbf{E}}\left[\log \frac{\mathbb{P}(\mathbf{E}|\mathbf{X})}{\mathbb{P}(\mathbf{E})}\right]. \tag{11}$$

Considering the marginal distribution $\mathbb{P}(\mathbf{E})$ can be represented as $\mathbb{P}(\mathbf{E}) = \sum_{\mathbf{DG}} \mathbb{P}_\phi(\mathbf{E}|\mathcal{G})\mathbb{P}(\mathcal{G})$, we can obtain the upper bound,

$$I(\mathbf{X}; \mathbf{E}) \leq \mathbb{E}_{\mathcal{G}}\left[\mathrm{KL}(\mathbb{P}_\phi(\mathbf{E}|\mathcal{G})||\mathbb{P}(\mathbf{E}))\right]. \tag{12}$$

Given the above derivations, we can further update the objective of Eq. 5 by formalize the variational upper bound,

$$\begin{aligned}\hat{\mathbb{Q}} = \arg\min_{\hat{\mathbb{Q}}} \mathbb{E}[&-\log \mathbb{P}(\mathbf{Y}|\mathbf{X}, \mathbf{E}) + \log \mathbb{P}_\phi(\mathbf{E}|\mathcal{G}) - \log \mathbb{P}_\theta(\mathbf{Y}|\mathbf{E}) \\ &+ \mathrm{KL}(\mathbb{P}_\phi(\mathbf{E}|\mathcal{G})||\mathbb{P}(\mathbf{E}))].\end{aligned} \tag{13}$$

We can divide this variational upper bound into following two optimization objectives,

$$\min \mathbb{E}[\log \mathbb{P}_\phi(\mathbf{E}|\mathcal{G}) - \log \mathbb{P}_\theta(\mathbf{Y}|\mathbf{E})], \tag{14}$$

$$\min \mathbb{E}[-\log \mathbb{P}(\mathbf{Y}|\mathbf{X}, \mathbf{E}) + \mathrm{KL}(\mathbb{P}_\phi(\mathbf{E}|\mathcal{G})||\mathbb{P}(\mathbf{E}))]. \tag{15}$$

The parameterized variational estimates $\mathbb{P}_\theta(\mathbf{Y}|\mathbf{E})$ and $\mathbb{P}_\phi(\mathbf{E}|\mathbf{X})$ establish the inference line $\langle \mathbf{X} \rightarrow \mathbf{E} \rightarrow \mathbf{Y} \rangle$. Consequently, the optimization process of Eq. 14 is tractable. Even though, the optimization process Eq. 15 remains unaccessible.

The first term of Eq. 15 highlights that the inference process $\langle \mathbf{X} \rightarrow \mathbf{Y} \leftarrow \mathbf{E} \rangle$ (i.e., $\mathbb{P}(\mathbf{Y}|\mathbf{X}, \mathbf{E})$) remains critically important, alongside the $\langle \mathbf{X} \rightarrow \mathbf{E} \rightarrow \mathbf{Y} \rangle$ inference path. The second term of Eq. 15 indicates that we need to incorporate the inductive bias $\mathbb{P}(\mathbf{E})$ into the extraction process of the environment variables $\langle \mathbf{X} \rightarrow \mathbf{E} \rangle$. Therefore, current parameterized models that bridge the gap between $\mathbf{X}$ and $\mathbf{Y}$ are inherently constrained by the process for estimating and optimizing Eq. 15.

Practical Guidance. The theoretical analysis above provides us with practical insights in tackling concept shift issue.

- Understanding the inductive bias $\mathbb{P}(\mathbf{E})$ of concept shiftis essential for achieving accurate inference of environment variables from history observations $\langle \mathbf{X} \rightarrow \mathbf{E} \rangle$.

- Adopting the collaborative inference lines of $\langle \mathbf{X} \rightarrow \mathbf{E} \rightarrow \mathbf{Y} \rangle$ and $\langle \mathbf{X} \rightarrow \mathbf{Y} \leftarrow \mathbf{E} \rangle$ to model the unseen environment variables that cause concept shift provides theoretical insights.

In the following research, we build on these two practical guidances to enhanced generalization capability in addressing concept shift issue.

5 Environment Inference from the Perspective of Spatio-Temporal State

In this section, we aim to infer the environment variables $\mathbf{E}$ that cause concept shift issue, i.e., the inference path $\langle \mathbf{X} \rightarrow \mathbf{E} \rangle$. This corresponds to the first practical guidance derived from our theoretical analysis. we adopt a two-stage modeling strategy, initially extracting environment variables comprehensively and subsequently refining the information that causes concept shift issue.

To achieve this insight, we propose a prefix-suffix collaborative state learning mechanism and a hierarchical state compressor. The former is designed to comprehensively capture the environment factors in dynamic graphs from the perspective of state space, while the latter focuses on refining the state variables that lead to concept shift.

5.1 Prefix-Suffix Collaborative State Learning Mechanism

In dynamic graphs, there are numerous variables associated with evolutionary patterns, making it difficult to isolate the environment variables that contribute to concept shift. Therefore, we adopt a two-stage modeling strategy, initially extracting environment variables comprehensively and subsequently refining the information that causes concept shift issue.

To comprehensively infer the environment factors, we introduce the spatio-temporal state (ST state $\mathbf{S}$) to construct a unified perspective for capturing environment information $\mathbf{E}$ in dynamic graphs. This design is inspired by the exciting performance achieved in the studies of sequence data from the perspective of state space. Compared to regular sequence data, the state in dynamic graphs is complex and therefore challenging to capture. Specifically, the ST state exhibit a *duality*, characterized by both *continuity* and *hysteresis*.

- The *continuity* of ST state signifies that each individual state is often reflected by the accumulation of past state patterns. For example, the timestamp state is typically the cumulative result of historical evolutionary processes.
- The *hysteresis* of ST state means that certain individual state information is retroactively extrapolated from future evolution patterns. For instance, unrecorded weather conditions or individual affinities can only be inferred through future evolutionary processes.

Given the *duality* of ST state, we introduce a prefix-suffix collaborative state learning method. This design incorporates a state learning operator that effectively models prefix information while also enabling the inference of suffix information, distinguishing it from traditional State Space Models (SSMs) that focus exclusively on learning prefix information. Additionally, existing works have confirmed that achieving data-adaptive state updates is equally important. This means the model should dynamically adjust the weights of state updates based on the characteristics of sequence data. Consequently, we aim to implement a ST state learning framework that satisfies the following two requirements.

- **Bidirectional state learning.** The ability to capture both prefix and suffix information in the state learning process.
- **Data-adaptive state updating.** A mechanism that adjusts the state updates dynamically based on the characteristics of the input sequence.

To achieve this target, we propose a Bidirectional Mixing Method (BMM), which encompasses a bidirectional mixing aggregation process and a adaptive state update operator. We first design a bidirectional aggregation kernel to achieve prefix-suffix state aggregation.

Bidirectional feature mixing kernel. Given the parameterized graph neural network backbone GNN_α, we first conduct spatial encoding on the graph, resulting in a temporal static representation $Z^t = \text{GNN}_\alpha(\mathcal{G}^t) \in \mathbb{R}^{N \times d}$ at step t. Next, we achieve bidirectional mixing of local steps through a message-passing kernel,

$$\hat{Z}^t_{pre} = [Z^t || Z^{t-1}] \cdot [I || W_\psi]^T, \tag{16}$$

$$\hat{Z}^t_{suf} = [Z^t || Z^{t+1}] \cdot [I || W_\xi]^T, \tag{17}$$

where $\hat{Z}^1_{pre} = Z^1$ and $\hat{Z}^T_{suf} = Z^T$, $[\cdot||\cdot]$ indicates the concatenation operation. $W_\xi \in \mathbb{R}^{d \times d}$ and $W_\psi \in \mathbb{R}^{d \times d}$ are formalized as learnable aggregation kernel. Although the aggregation between the local sequences does not capture the long-term evolutionary patterns, it distinctly determines the direction of information compression. Building on this certain direction, we achieve state updates for long-term observations by amplifying the dependencies of the short sequences

State update operation. After each bidirectional feature mixing layer, we perform a global state update based on the local mixed information. Inspired by the successful practices of existing state space models, we introduce learnable state transition matrix $A \in \mathbb{R}^{N \times d \times d}$ and input control matrix $B \in \mathbb{R}^{N \times d \times d}$ to update state,

$$S^t_{pre}(n) = A_n S^{t-1}_{pre}(n) + B_n \hat{Z}^{t-1}_{pre}(n), \tag{18}$$

$$S^t_{suf}(n) = A_n S^{t+1}_{suf}(n) + B_n \hat{Z}^{t+1}_{suf}(n), \tag{19}$$

where $S^1_{pre} = \hat{Z}^1_{suf}$, $S^T_{suf} = \hat{Z}^T_{suf}$ and $n \in [1, ..., N]$ denotes the node. Thus, we can obtain the states derived from both prefix learning and suffix learning, i.e., $S_{pre} \in \mathbb{R}^{T \times N \times D}$ and $S_{suf} \in \mathbb{R}^{T \times N \times D}$. We then couple them into a unified state space representation,

$$S = S_{pre} \odot S_{suf}, \tag{20}$$

where $\odot$ is the element-wise product. Guided by the *dualityt* of ST state in dynamic graphs, we comprehensively model the ST state $S = \{S^1, S^2, ..., S^T_n\} \in \mathbb{R}^{T \times N \times d}$. The next challenge lie in refining the information that contributes to conceptual distribution shift.

5.2 Hierarchical Compressor for Inferring Activable States

We start by investigating the characteristics of those states that contribute to the concept shift issue. We argue that the concept shift is controlled by those ST states, which remain inactive in historical observations but become active in future evolution. We refer to

these as activatable states. The appearance of these states naturally leads to a significant proportion of samples in the dataset where the historical observations are identical, yet the future evolutions differ.

Therefore, we propose a hierarchical compressor for inferring activable states. Specifically, we capture those states with both inactive and active properties by constructing hierarchical historical-future scenarios. Before introducing the implementation details of proposed compressor, we first present Assumption 5.1.

Assumption 5.1. Given a sample with concept shift issue, it contains historical observations $\mathbf{X} = \mathcal{G}^{1:T}$ and future evolution $\mathbf{Y} = \mathcal{G}^{T+1:T+K}$. The states S that causes the concept shift stay active in $\mathbf{Y}$, but shows a gradual transition from fully inactive to progressively deactivated in $\mathbf{X}$. The steps in $\mathbf{X}$ that are closer to the future exhibit a slightly increasing activation trend for S.

This assumption is to ensure the availability of the $\mathbb{P}(S|\mathbf{X})$ process, i.e., $\mathbb{P}(\mathbf{E}|\mathbf{X})$. Actually, it can be supported by real-world datasets. Our compressor aim to extend the prediction task from $\mathbf{X} = \mathcal{G}^{\mathbf{1:T}}$ to $\mathbf{Y} = \mathcal{G}^{\mathbf{T+1:T+K}}$ by introducing an auxiliary task focused on state learning within the historical observations. Specifically, we conduct a fine-grained partition of the historical data $\mathcal{G}^{1:T}$ to create a hierarchical history-future data form $\mathcal{G}^{1:T} = (\mathcal{G}^H, \mathcal{G}^F)$, where $H \cup F = \{1, \cdots, T\}$. In the implementation, H and F are divided equally over the set $\{1, \cdots, T\}$. Therefore, we construct a hierarchical learning tasks,

$$\mathcal{G}^H \to \mathcal{G}^F, \quad \mathcal{G}^{1:T} \to \mathcal{G}^{T+1:T+K}. \tag{21}$$

We then obtain corresponding training objectives,

$$\mathcal{L}_{obs} \coloneqq \mathcal{L}(\mathcal{G}^H \to \mathcal{G}^F), \ \ \mathcal{L}_{fut} \coloneqq \mathcal{L}(\mathcal{G}^{1:T} \to \mathcal{G}^{T+1:T+K}). \tag{22}$$

In this section, we undertake a comprehensive investigation of environment variables $\mathbf{E}$ from the perspective of state space. We innovatively summarize two pivotal inductive biases: the *duality* of states and the *activatable property* of states that contribute to concept shift. We then propose an effective modeling framework for the spatio-temporal states leading to the concept shift issue.

6 Skip-connection Dynamic Graph Prediction

We aim to develop a dynamic graph prediction module that leverages the inferred states to address concept shift issue, i.e., the collaborative inference paths of $\langle \mathbf{X} \to \mathbf{E} \to \mathbf{Y} \rangle$ and $\langle \mathbf{X} \to \mathbf{Y} \leftarrow \mathbf{E} \rangle$. This corresponds to the second practical guidance derived from our theoretical analysis. Based on the definition of conditional probability, we can derive it by marginalizing over $\mathbf{X}$,

$$\mathbb{P}(\mathbf{Y}|\mathbf{E}) = \sum_{\mathbf{E}} \mathbb{P}(\mathbf{Y}|\mathbf{X}, \mathbf{E}) \mathbb{P}(\mathbf{X}|\mathbf{E}). \tag{23}$$

Therefore, $\mathbb{P}(\mathbf{Y}|\mathbf{E})$ can be obtained by marginalizing over $\mathbf{X}$. This involves integrating the probabilities of $\mathbf{X}$ to derive the probability distribution of $\mathbf{Y}$ given $\mathbf{E}$. We can further utilize a unified parameterized framework $\mathbb{P}(\mathbf{Y}|\hat{\mathbf{E}})$ to align the collaborative mechanisms ($\langle \mathbf{X} \to \mathbf{E} \to \mathbf{Y} \rangle$ & $\langle \mathbf{X} \to \mathbf{Y} \leftarrow \mathbf{E} \rangle$) for predicting future evolution, where $\hat{\mathbf{E}}$ is guided to encompass the information from $\mathbf{X}$ and $\mathbf{E}$.

To achieve this target, we propose a skip-connection dynamic graph prediction framework. Given the initial representation $Z \in \mathbb{R}^{T \times N \times d}$ obtained from the GNN encoder and the ST state $S \in \mathbb{R}^{T \times N \times d}$ learned from the prefix-suffix modeling mechanism, we implement a connection based on skip manner,

$$\hat{S} = S \oplus Z \cdot D, \tag{24}$$

where $D \in \mathbb{R}^{d \times d}$ represents learnable parameters demoting the skip weights and $\oplus$ is the element-wise addition. Then, skip-connected output $\hat{S} \sim \hat{\mathbf{E}}$ will be fed into a spatio-temporal prediction layer $\mathbb{P}_\theta$ to predict future evolution. The learning objective $\mathcal{L}_{fut}$ for predicting future evolution $\mathbf{Y}$ based on historical observations $\mathbf{X}$ is,

$$\mathcal{L}_{fut} = -\mathbb{E}[\log \mathbb{P}_\theta(\mathbf{Y}|\hat{S})] + \mathbb{E}[\log(\mathbb{P}_\phi(\hat{S}|\mathbf{X}))], \tag{25}$$

where $\phi = \{\alpha, W_\psi, W_\xi, A, B, D\}$ denotes the parameterized prefix-suffix state learning mechanism, θ indicates the spatio-temporal prediction process. Considering our proposed hierarchical state compressor, the learning objective of the auxiliary task is,

$$\mathcal{L}_{obs} = -\mathbb{E}[\log \mathbb{P}_\theta(\mathcal{G}^U|\hat{S}^H)] + \mathbb{E}[\log(\mathbb{P}_\phi(\hat{S}^H|\mathcal{G}^H))]. \tag{26}$$

The overall training objectives of the proposed **Samen** is,

$$\mathcal{L}_{ST} = \mathcal{L}_{fut} + \mathcal{L}_{obs}. \tag{27}$$

Algorithm 1 The training process of **Samen**

Input: historical dynamic graph data $\mathbf{X} = \mathcal{G}^{1:T}$
Initial: graph encoder GNN_α, state learning module $\phi = \{\alpha, W_\psi, W_\xi, A, B, D\}$, spatio-temporal prediction module θ, future evolution horizons K, the layers of state learning L
Training:
$Z^t = \text{GNN}_\alpha(\mathcal{G}^t)$
Prefix-suffix State learning mechanism:
for $i = 1$ **to** L **do**
$\hat{Z}^t_{pre} = [Z^t || Z^{t-1}] \cdot [I || W_\psi]^T$
$\hat{Z}^t_{suf} = [Z^t || Z^{t+1}] \cdot [I || W_\xi]^T$
$S^t_{pre}(n) = A_n S^{t-1}_{pre}(n) + B_n \hat{Z}^{t-1}_{pre}(n)$
$S^t_{suf}(n) = A_n S^{t+1}_{suf}(n) + B_n \hat{Z}^{t+1}_{suf}(n)$
$S = S_{pre} \odot S_{suf}$
$Z = S + \text{GNN}_\alpha(S)$
end for
Future evolution prediction module:
$\hat{S} = S \oplus Z \cdot D$
$\hat{\mathbf{Y}} = \mathbb{P}_\theta(\mathbf{Y}|\hat{S})$
Optimizing:
$\mathcal{L}_{obs} \coloneqq \mathcal{L}(\mathcal{G}^H \to \mathcal{G}^F), \ \ \mathcal{L}_{fut} \coloneqq \mathcal{L}(\mathcal{G}^{1:T} \to \mathcal{G}^{T+1:T+K})$
$\mathcal{L}_{ST} = \mathcal{L}_{fut} + \mathcal{L}_{obs}$

7 Experiment

We conduct extensive experiments to evaluate the effectiveness of **Samen**[1] in addressing the concept shift issue. Specifically, we evaluate our **Samen** by answering the following questions.

- **Q1**. Does our **Samen** demonstrate competitiveness compared to existing state-of-the-art methods on real-world datasets?
- **Q2**. Does our approach effectively tackle the issues that remain unresolved in existing works?
- **Q3**. Does each component in our **Samen** effectively enhance the generalization capacity?

[1] Our source code is available at https://anonymous.4open.science/r/Samen-CE73.

Table 1: AUC score (%) of future link prediction task on real-world social relationship datasets. The best results are shown in bold and the second best results are underlined.

Model	COLLAB	Yelp	ACT
DySAT	88.77±0.23	78.87±0.57	78.52±0.40
VREx	88.31±0.32	79.04±0.16	83.11±0.29
GroupDRO	88.76±0.12	79.38±0.42	85.19±0.53
DIDA	91.97±0.05	78.22±0.40	89.84±0.82
EAGLE	92.45±0.21	78.97±0.31	92.37±0.53
DGIB	92.68±0.20	79.53±0.20	**94.89±0.20**
Samen	**92.71±0.15**	**79.96±0.21**	94.41±0.19

- **Q4**. Does our framework operate with high efficiency?

7.1 Experiment Setup

We introduce datasets, baselines and experiment settings briefly.

7.1.1 ***Datasets.*** We employ seven cross-domain real-world dynamic graph datasets to evaluate our **Samen**.

- COLLAB [35] is an academic collaboration dataset consisting of papers published over a span of 16 years. Yelp [27] is a dataset that contains business reviews, while ACT [19] tracks students' activities on a MOOC platform over a 30-day period.
- PEMS08 and PEMS04 [33] are classic traffic network datasets from California, featuring 5-minute intervals. SD-2019 and GBA-2019 [23] are newly introduced large-scale traffic network datasets.

7.1.2 ***Baselines.*** We compare **Samen** against two categories of baselines: six social link forecasting methods and six traffic flow prediction frameworks.

- Social link forecasting methods consist of DySAT[27], VREx[18], GroupDRO[26], DIDA[48], EAGLE [48], DGIB [47].
- Traffic flow prediction baselines include AGCRN [2], GWNET [41], Z-GCNETs [6], D2STGNN [31], STGNCDE [8], CaST [42].

7.1.3 ***Experiment Settings.*** In addition to evaluating model performance on regular real-world datasets, we also filter samples exhibiting concept shift and investigate the effectiveness of our **Samen** on these samples.

- The task of social relationship analysis is to exploit past graphs to make link prediction in the next time step. The principle for selecting samples that exist concept shift is as follows. For a given sample $(\mathbf{X}, \mathbf{Y})$, if there exists another sample $(\mathbf{X}', \mathbf{Y}')$ satisfying $\mathbf{X} = \mathbf{X}'$ and $\mathbf{Y} \neq \mathbf{Y}'$, the current sample will be identified as exhibiting concept shift.
- The traffic flow forecast task is to predict the next 12 steps based on historical 12 steps observations ($12 \rightarrow 12$). The principle for selecting samples that exist concept shift: given a sample $(\mathbf{X}, \mathbf{Y})$, if there exists another sample $(\mathbf{X}', \mathbf{Y}')$ satisfying $\mathrm{MAD}(\mathbf{X}, \mathbf{X}') \leq \sigma$ and $\mathrm{MAD}(\mathbf{Y}, \mathbf{Y}') \geq 10\sigma$, the current sample will be identified as exhibiting concept shift [2].

[2]MAD denotes Mean Absolute Distance, which follows the same calculation method as MAE (Mean Absolute Error). In practice, σ is set to 0.5.

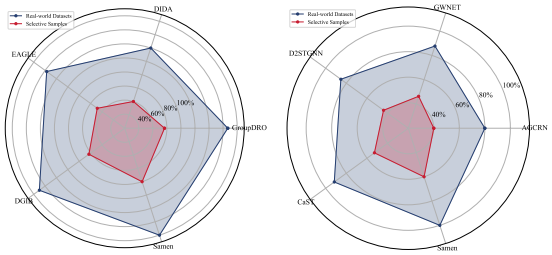

Figure 4: The performance of different methods on both the raw dataset and shifted dataset. The value here represents a performance discount, not a true metric value.

7.2 Performance Analysis on Real-world Datasets

We comprehensively evaluate the performance of **Samen** across seven real-world datasets from different domains to answer **Q1**.

Social link forecasting. Tab. 1 presents the performance of **Samen** on social link prediction tasks. Our model outperforms all baselines on two datasets. We also observe that DGIB [47] achieves one best performance on ACT, and DGIB [48] achieve many sub-optimal results. It proves that the approaches of tackling data out-of-distribution issue can also bring some insights into solving concept shift.

Traffic flow prediction. Tab. 2 shows the results of **Samen** on traffic flow dataset. We can observe that **Samen** outperforms all baselines on two datasets. We note that CaST [42] obtains suboptimal results on most datasets and even optimal results on PEMS04. This indicates that practices centered on inferring environment variables may offer potential solutions for addressing both data distribution shift and concept shift. Based on the comparison of different methods across datasets from two domains, we observe that **Samen** is competitive with existing state-of-the-art frameworks. More importantly, we emphasize the performance of our method in addressing challenges that have not been tackled by previous works.

7.3 Generalization Analysis for Tackling Concept Shift

To answer **Q2**, we compare the performance degradation of **Samen** with existing methods on samples affected by concept shift. We adopt the concept shift samples filtered from the Yelp and SD-2019 datasets, as used in Fig. 1. As shown in Fig. 4, we provide the performance comparison of the models on the raw dataset and the concept-shifted sample set. We can observe that **Samen** exhibits the least performance degradation on the samples with concept shift, and significantly outperforms all existing methods. We can conclude that our framework demonstrates excellent generalization ability in countering concept shift issue.

7.4 Ablation Study

To answer **Q3**, we investigate each component of **Samen**. Specifically, we conduct ablation studies to explore the effectiveness of prefix-suffix cooperative state learning mechanism, hierarchical

Table 2: The performance of traffic prediction tasks (12 → 12) on four real-world datasets. The best results are shown in bold and the second best results are underlined.

Model	PEMS08		PEMS04		SD-2019		GBA-2019	
	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
GWNET	16.67±0.9	26.77±1.1	20.69±0.8	33.02±1.3	17.74±1.1	29.62±2.9	20.91±2.6	**33.41±2.3**
AGCRN	15.95±0.2	25.75±0.6	19.83±0.3	32.26±0.2	18.09±1.2	32.01±1.3	21.01±1.6	34.25±2.0
Z-GCNETs	15.76±0.3	26.31±0.2	19.50±0.5	31.91±0.7	18.21±1.7	32.76±1.8	21.84±1.2	35.12±2.2
D2STGNN	15.69±0.3	26.41±0.4	19.55±0.7	31.99±0.5	17.85±1.5	29.51±1.7	20.71±2.0	33.65±2.7
STGNCDE	15.32±0.6	26.38±0.3	18.94±0.6	32.39±0.4	19.55±0.7	33.57±1.5	21.79±1.5	35.37±2.4
CaST	15.08±0.3	25.16±0.4	18.56±0.9	**31.44±0.3**	17.61±1.3	29.12±1.4	21.30±1.5	34.67±1.5
Samen	**14.70±0.2**	**24.26±0.6**	**17.91±0.4**	31.62±0.3	**17.18±1.2**	**28.32±1.4**	**20.26±1.5**	33.54±1.8

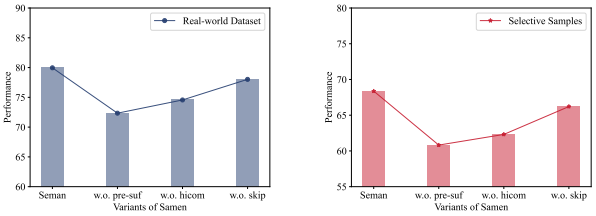

Figure 5: Ablation Study. We design three variants of Samen and compare their performance on raw Yelp (*left*) and sample set with concept shift (*right*).

compressor and skip-connection dynamic graph predictor. We design three variants of **Samen** for each of the three main modules, i.e, *w.o. pre-suf*, *w.o.hicom* and *w.o. skip*. They respectively represent the variant that removes suffix state learning, the variant that removes the hierarchical compressor and focuses solely on single-prediction mode, and the variant that remove skip connections. As shown in Fig. 5, we observe significant performance degradation in each variant, with the degradation being particularly pronounced on samples affected by concept shift. By comparing the degree of performance degradation, we find that prefix-suffix state learning module plays the most critical role, followed by the hierarchical state compressor. Therefore, we infer that learning suffix state in dynamic graph scenarios helps the model to comprehensively capturing the spatio-temporal evolution patterns. This practice of suffix learning provides a useful insight for the later research on concept shift.

7.5 Efficiency Analysis

The time complexity of **Samen** is $O(T \cdot L \cdot N(2d^2+1) + T \cdot L \cdot \mathcal{E})$, where N represents the number of nodes, $\mathcal{E}$ denotes the number of edges, d is the feature dimension, T represents the step of history observation, L denotes number of layers of the prefix-suffix state learning mechanism. The time cost of our **Samen** primarily lies in the ST state learning process with $O(2 \cdot T \cdot L \cdot N \cdot d^2)$, and the time complexity for graph learning with $O(L \cdot (N+\mathcal{E}))$. The factor of 2 in state update process arises from our prefix-suffix bidirectional aggregation. Our method has linear time complexity with high training efficiency.

We empirically investigate the training efficiency of **Samen**, as shown in Fig. 3. We also conduct efficiency comparisons of **Samen**, DIDA, and EAGLE, measuring the time taken per epoch. All experiments are running on a NVIDIA A100-PCIE-40GB. We observe that the training efficiency of our method is competitive with existing approaches, with high efficiency noted in some dynamic graphs.

Table 3: The training efficiency of Samen with other baselines on COLLAB, Yelp and ACT (s/epoch).

Models	DIDA	EAGLE	Samen
COLLAB	11.27	12.23	10.83
Yelp	7.92	6.89	7.12
ACT	10.27	9.34	9.23

8 Conclusion

In this work, we theoretically unveil the intractable challenge posed by concept shift in dynamic graphs, and propose a state-driven environment inference framework (**Samen**) to address this issue. Building upon a comprehensive investigation of concept shift from an information theory perspective, we distill two practical guidelines to design dynamic graph learning model with enhanced generalization ability. Unifying the environment factors in dynamic graph within the state space, we propose a prefix-suffix collaborative state learning mechanism to bidirectionally capture the environment information, and introduce a hierarchical state compressor to refine the environment information leading to concept shift. Extensive experiments over seven datasets verify the effectiveness of our model. The **limitation** of our work is the lack of interpretable strategy to validate that the environment variables learned from state updates align with the real-world ground truth. This challenge arises from the environment variables affecting concept shift in the real world are often multivariate and unavailable. Consequently, the inherent unavailability of the existing data restrict our design to interpretablely identify such ground truth. Additionally, the intertwining of spatial and temporal environments also hinders our efforts to propose interpretable methods.

References

[1] Robert B Ash. 2012. *Information theory*. Courier Corporation.

[2] Lei Bai, Lina Yao, Can Li, Xianzhi Wang, and Can Wang. 2020. Adaptive graph convolutional recurrent network for traffic forecasting. *Advances in neural information processing systems* 33 (2020), 17804–17815.

[3] Tanya Y Berger-Wolf and Jared Saia. 2006. A framework for analysis of dynamic social networks. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. 523–528.

[4] Da Cao, Yawen Zeng, Meng Liu, Xiangnan He, Meng Wang, and Zheng Qin. 2020. Strong: Spatio-temporal reinforcement learning for cross-modal video moment localization. In *Proceedings of the 28th ACM international conference on multimedia*. 4162–4170.

[5] Yongqiang Chen, Yatao Bian, Kaiwen Zhou, Binghui Xie, Bo Han, and James Cheng. 2024. Does invariant graph learning via environment augmentation learn invariance? *Advances in Neural Information Processing Systems* 36 (2024).

[6] Yuzhou Chen, Ignacio Segovia, and Yulia R Gel. 2021. Z-GCNETs: Time zigzags at graph convolutional networks for time series forecasting. In *International Conference on Machine Learning*. PMLR, 1684–1694.

[7] Yongqiang Chen, Yonggang Zhang, Yatao Bian, Han Yang, MA Kaili, Binghui Xie, Tongliang Liu, Bo Han, and James Cheng. 2022. Learning causally invariant representations for out-of-distribution generalization on graphs. *Advances in Neural Information Processing Systems* 35 (2022), 22131–22148.

[8] Jeongwhan Choi, Hwangyong Choi, Jeehyun Hwang, and Noseong Park. 2022. Graph neural controlled differential equations for traffic forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 6367–6374.

[9] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. 2016. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in neural information processing systems* 29 (2016).

[10] Chen Gao, Xiang Wang, Xiangnan He, and Yong Li. 2022. Graph neural networks for recommender system. In *Proceedings of the fifteenth ACM international conference on web search and data mining*. 1623–1625.

[11] Albert Gu and Tri Dao. 2023. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752* (2023).

[12] Albert Gu, Tri Dao, Stefano Ermon, Atri Rudra, and Christopher Ré. 2020. Hippo: Recurrent memory with optimal polynomial projections. *Advances in neural information processing systems* 33 (2020), 1474–1487.

[13] Albert Gu, Karan Goel, and Christopher Ré. 2021. Efficiently modeling long sequences with structured state spaces. *arXiv preprint arXiv:2111.00396* (2021).

[14] Albert Gu, Isys Johnson, Karan Goel, Khaled Saab, Tri Dao, Atri Rudra, and Christopher Ré. 2021. Combining recurrent, convolutional, and continuous-time models with linear state space layers. *Advances in neural information processing systems* 34 (2021), 572–585.

[15] Qihe Huang, Lei Shen, Ruixin Zhang, Shouhong Ding, Binwu Wang, Zhengyang Zhou, and Yang Wang. 2023. CrossGNN: Confronting Noisy Multivariate Time Series Via Cross Interaction Refinement. In *Thirty-seventh Conference on Neural Information Processing Systems*.

[16] Edwin T Jaynes. 1957. Information theory and statistical mechanics. *Physical review* 106, 4 (1957), 620.

[17] Nikolas Kantas, Arnaud Doucet, Sumeetpal S Singh, Jan Maciejowski, and Nicolas Chopin. 2015. On particle methods for parameter estimation in state-space models. (2015).

[18] David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. 2021. Out-of-distribution generalization via risk extrapolation (rex). In *International Conference on Machine Learning*. PMLR, 5815–5826.

[19] Srijan Kumar, Xikun Zhang, and Jure Leskovec. 2019. Predicting dynamic embedding trajectory in temporal interaction networks. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 1269–1278.

[20] Shiyong Lan, Yitong Ma, Weikang Huang, Wenwu Wang, Hongyu Yang, and Pyang Li. 2022. Dstagnn: Dynamic spatial-temporal aware graph neural network for traffic flow forecasting. In *International conference on machine learning*. PMLR, 11906–11917.

[21] Haoyang Li, Ziwei Zhang, Xin Wang, and Wenwu Zhu. 2022. Learning invariant graph representations for out-of-distribution generalization. *Advances in Neural Information Processing Systems* 35 (2022), 11828–11841.

[22] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. 2017. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926* (2017).

[23] Xu Liu, Yutong Xia, Yuxuan Liang, Junfeng Hu, Yiwei Wang, Lei Bai, Chao Huang, Zhenguang Liu, Bryan Hooi, and Roger Zimmermann. 2023. LargeST: A Benchmark Dataset for Large-Scale Traffic Forecasting. *arXiv preprint arXiv:2306.08259* (2023).

[24] Syama Sundar Rangapuram, Matthias W Seeger, Jan Gasthaus, Lorenzo Stella, Yuyang Wang, and Tim Januschowski. 2018. Deep state space models for time series forecasting. *Advances in neural information processing systems* 31 (2018).

[25] Emanuele Rossi, Ben Chamberlain, Fabrizio Frasca, Davide Eynard, Federico Monti, and Michael Bronstein. 2020. Temporal graph networks for deep learning on dynamic graphs. *arXiv preprint arXiv:2006.10637* (2020).

[26] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. 2019. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. *arXiv preprint arXiv:1911.08731* (2019).

[27] Aravind Sankar, Yanhong Wu, Liang Gou, Wei Zhang, and Hao Yang. 2020. Dysat: Deep neural representation learning on dynamic graphs via self-attention networks. In *Proceedings of the 13th international conference on web search and data mining*. 519–527.

[28] Franco Scarselli, Sweah Liang Yong, Marco Gori, Markus Hagenbuchner, Ah Chung Tsoi, and Marco Maggini. 2005. Graph neural networks for ranking web pages. In *The 2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI'05)*. IEEE, 666–672.

[29] Youngjoo Seo, Michaël Defferrard, Pierre Vandergheynst, and Xavier Bresson. 2018. Structured sequence modeling with graph convolutional recurrent networks. In *Neural Information Processing: 25th International Conference, ICONIP 2018, Siem Reap, Cambodia, December 13-16, 2018, Proceedings, Part I 25*. Springer, 362–373.

[30] Zezhi Shao, Zhao Zhang, Fei Wang, Wei Wei, and Yongjun Xu. 2022. Spatial-temporal identity: A simple yet effective baseline for multivariate time series forecasting. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 4454–4458.

[31] Zezhi Shao, Zhao Zhang, Wei Wei, Fei Wang, Yongjun Xu, Xin Cao, and Christian S Jensen. 2022. Decoupled dynamic spatial-temporal graph neural network for traffic forecasting. *arXiv preprint arXiv:2206.09112* (2022).

[32] Jimmy TH Smith, Andrew Warrington, and Scott W Linderman. 2022. Simplified state space layers for sequence modeling. *arXiv preprint arXiv:2208.04933* (2022).

[33] Chao Song, Youfang Lin, Shengnan Guo, and Huaiyu Wan. 2020. Spatial-temporal synchronous graph convolutional networks: A new framework for spatial-temporal network data forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 34. 914–921.

[34] Qingyun Sun, Jianxin Li, Hao Peng, Jia Wu, Yuanxing Ning, Philip S Yu, and Lifang He. 2021. Sugar: Subgraph neural network with reinforcement pooling and self-supervised mutual information mechanism. In *Proceedings of the Web Conference 2021*. 2081–2091.

[35] Jie Tang, Sen Wu, Jimeng Sun, and Hang Su. 2012. Cross-domain collaboration recommendation. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1285–1293.

[36] Binwu Wang, Jiaming Ma, Pengkun Wang, Xu Wang, Yudong Zhang, Zhengyang Zhou, and Yang Wang. 2024. Stone: A spatio-temporal ood learning framework kills both spatial and temporal shifts. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2948–2959.

[37] Xu Wang, Lianliang Chen, Hongbo Zhang, Pengkun Wang, Zhengyang Zhou, and Yang Wang. 2023. A Multi-graph Fusion Based Spatiotemporal Dynamic Learning Framework. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*. 294–302.

[38] Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. 2019. Kgat: Knowledge graph attention network for recommendation. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 950–958.

[39] G Welch. 1995. An Introduction to the Kalman Filter. (1995).

[40] Qitian Wu, Hengrui Zhang, Junchi Yan, and David Wipf. 2022. Handling distribution shifts on graphs: An invariance perspective. *arXiv preprint arXiv:2202.02466* (2022).

[41] Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Chengqi Zhang. 2019. Graph wavenet for deep spatial-temporal graph modeling. *arXiv preprint arXiv:1906.00121* (2019).

[42] Yutong Xia, Yuxuan Liang, Haomin Wen, Xu Liu, Kun Wang, Zhengyang Zhou, and Roger Zimmermann. 2023. Deciphering spatio-temporal graph forecasting: A causal lens and treatment. *arXiv preprint arXiv:2309.13378* (2023).

[43] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. 2018. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826* (2018).

[44] Menglin Yang, Min Zhou, Marcus Kalander, Zengfeng Huang, and Irwin King. 2021. Discrete-time temporal network embedding via implicit hierarchical learning in hyperbolic space. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 1975–1985.

[45] Junchi Yu, Jian Liang, and Ran He. 2023. Mind the Label Shift of Augmentation-based Graph OOD Generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 11620–11630.

[46] Yong Yu, Xiaosheng Si, Changhua Hu, and Jianxun Zhang. 2019. A review of recurrent neural networks: LSTM cells and network architectures. *Neural computation* 31, 7 (2019), 1235–1270.

[47] Haonan Yuan, Qingyun Sun, Xingcheng Fu, Cheng Ji, and Jianxin Li. 2024. Dynamic Graph Information Bottleneck. In *Proceedings of the ACM on Web Conference 2024*. 469–480.

[48] Haonan Yuan, Qingyun Sun, Xingcheng Fu, Ziwei Zhang, Cheng Ji, Hao Peng, and Jianxin Li. 2023. Environment-Aware Dynamic Graph Learning for Out-of-Distribution Generalization. *arXiv preprint arXiv:2311.11114* (2023).
[49] Michael Zhang, Khaled K Saab, Michael Poli, Tri Dao, Karan Goel, and Christopher Ré. 2023. Effectively modeling time series with simple discrete state spaces. *arXiv preprint arXiv:2303.09489* (2023).
[50] Zeyang Zhang, Xin Wang, Ziwei Zhang, Haoyang Li, Zhou Qin, and Wenwu Zhu. 2022. Dynamic graph neural networks under spatio-temporal distribution shift. *Advances in Neural Information Processing Systems* 35 (2022), 6074–6089.
[51] Zhengyang Zhou, Qihe Huang, Kuo Yang, Kun Wang, Xu Wang, Yudong Zhang, Yuxuan Liang, and Yang Wang. 2023. Maintaining the Status Quo: Capturing Invariant Relations for OOD Spatiotemporal Learning. (2023).
[52] Zhengyang Zhou, Yang Wang, Xike Xie, Lianliang Chen, and Hengchang Liu. 2020. RiskOracle: A minute-level citywide traffic accident forecasting framework. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 34. 1258–1265.

A Datasets

Our experimental design included the selection of seven real-world dynamic graph datasets from two distinct domains. The detailed statistics of the datasets are as shown in Tab. 4.

Table 4: Statistics of the datasets.

Dataset	# Nodes	# Edges	# Snapshots
COLLAB	23,035	151,790	16
Yelp	13,095	65,375	24
ACT	20,408	202,339	30
PEMS08	170	276	17,856
PEMS04	307	338	16,992
SD-2019	716	17,319	525,888
GBA-2019	2,352	61,246	525,888

COLLAB [35] is an academic collaboration dataset comprising papers published between 1990 and 2006, spanning 16 graph snapshots. In this dataset, nodes represent authors, and edges represent co-authorship relationships. The edges include five attributes based on co-authored publications: "Data Mining", "Database", "Medical Informatics", "Theory" and "Visualization".

Yelp [27] is a dataset containing customer reviews on businesses. In this dataset, nodes represent customers and businesses, while edges capture review behaviors. The edges are associated with five attributes based on business categories: "Pizza", "American (New) Food", "Coffee & Tea", "Sushi Bars" and "Fast Food".

ACT [19] characterizes student interactions on a MOOC platform over a span of one month, consisting of 30 graph snapshots. In this dataset, nodes represent students or the targets of actions, while edges signify various student actions.

PEMS08 [33] is collected from the Caltrans Performance Measurement System (PeMS), which records the real traffic network flow data from 07/01/2016 to 08/31/2016. It delineates a dynamic graph data of a traffic network with 170 sensors across 17,856 steps. Among the known traffic datasets, it falls into the category of small-scale dataset.

PEMS04 [33] records the real traffic network flow data from 01/01/2018 to 02/28/2018. It describes a dynamic graph data of a traffic network with 307 sensors across 16,992 steps. It belongs to a medium-scale traffic dataset.

SD [23] is a sub-dataset of the large-scale dataset CA proposed by [23]. It comprises traffic flow data recorded by 716 sensors in San Diego county from 01/01/2017 to 12/31/2021. We select all samples from 2019.

GBA [23] is a larger traffic dataset than SD, which is also a sub-dataset of the large-scale dataset CA. It contains traffic flow data provided by 2,352 sensors in 11 counties situated in the Greater Bay Area from 01/01/2017 to 12/31/2021. We select all samples from 2019.

B Metrics

Link connection prediction task. We use AUC value to evaluate the performance of the models in binary link prediction task. We first define four variables:

- True Positives (TP): The number of positive instances correctly classified.
- False Positives (FP): The number of negative instances incorrectly classified as positive.
- True Negatives (TN): The number of negative instances correctly classified.
- False Negatives (FN): The number of positive instances incorrectly classified as negative.

Based on those four variables, we can calculate the value of True Positive Rate (TPR) and False Positive Rate (FPR):

$$TPR = \frac{TP}{TP + FN}, \quad FPR = \frac{FP}{FP + TN}. \tag{28}$$

For a prediction task given N samples, we can calculate AUC:

$$AUC = \sum_{i=1}^{N-1} (TPR_{i+1} + TPR_i) \cdot (FPR_{i+1} - FPR_i). \tag{29}$$

Traffic prediction task. We utilize Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) to assess the performance of our Samen and other baselines. Both metrics quantify the error between model predictions and actual observations in regression tasks. A smaller value for these metrics indicates better model performance. MAE is less sensitive to outliers due to its use of absolute differences, while RMSE is more sensitive to large errors due to its use of squared differences. Given the actual observation Y_i and the corresponding predicted value $\hat{Y}_i$ for n samples, two metrics are calculated as follows:

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |Y_i - \hat{Y}_i|, \tag{30}$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2}. \tag{31}$$

C Detailed Implementation

We implement our Samen and parts of baselines with PyTorch 1.10.1 on a server with NVIDIA A100-PCIE-40GB. All experiments are repeated with 10 different random seeds of [1,2,3,4,5,6,7,8,9,10]. The reported results include the mean and standard deviation obtained from these 10 runs. During the training, we employ the Adam optimizer.