Mitigating Dynamic Graph Distribution Shifts via Spectral Augmentation

Qianyu Song[1], **Chao Li**[1*], **Zhongying Zhao**[2], **Hua Duan**[3] and **Qingtian Zeng**[2]
[1]College of Electronic and Information Engineering, Shandong University of Science and Technology (SDUST), Qingdao, China
[2]College of Computer Science and Engineering, SDUST, Qingdao, China
[3]College of Mathematics and Systems Science, SDUST, Qingdao, China
qianasong@163.com, lichao@sdust.edu.cn, zzysuin@163.com, hduan@sdust.edu.cn, qtzeng@163.com

Abstract

Dynamic Graph Neural Networks (DyGNNs) are facing challenges with distribution shifts between training and test data that are similar but not identical. Existing DyGNNs for out-of-distribution scenarios primarily focus on discovering invariant patterns in the spatial domain, overlooking its impact on the structural properties in the spectral domain. In this paper, we propose a spectral-based graph augmentation framework designed to investigate and improve generalization behavior in dynamic graphs under distribution shifts. Specifically, we augment the input graph spectra into a mixture of shift components by maximizing the variance of spectral distance and propose an efficient approximation to reduce the computational cost brought by eigen-decomposition. Building on this, we develop a multi-encoder architecture in which each encoder targets a specific spectral shift component to generate referential representations. Our method adopts a new learning objective to encourage the model to rely on robust spectral properties for better generalization under distribution shifts. Extensive experiments on both real-world and synthetic datasets demonstrate that our proposed method significantly outperforms state-of-the-art approaches in node classification and link prediction tasks. Source codes are available at https://github.com/SSQiana/DSPA.

1 Introduction

Dynamic graphs can represent complex relationships in the real world, where nodes correspond to entities and edges indicate their interactions. Dynamic graph neural networks (DyGNNs) have been proposed to tackle complex structural and temporal information over dynamic graphs, and have demonstrated significant advancements in various domains, including social networks [Zhu *et al.*, 2024; Jiang *et al.*, 2024], recommendation systems [Zhang and Gan, 2024; Abolghasemi *et al.*, 2024], and transportation networks [Xia *et al.*, 2024; Chen *et al.*, 2024; Zhou *et al.*, 2020].

*Corresponding Author

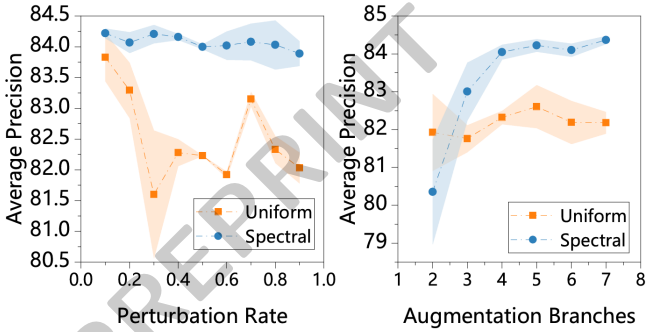


Figure 1: Comparison between uniform and spectral augmentations.

Despite their great success, most DyGNNs are trained and evaluated under the assumption that dynamic graph data are independent and identically distributed. However, this assumption does not always hold because distribution shifts naturally emerge as graphs evolve. For instance, in social networks, user interaction patterns may change significantly in response to trending topics, making past data less representative of current behaviors. If a classifier relies heavily on such spurious patterns from the training set, it inevitably fails to generalize to unseen test sets under distributions.

Invariant learning [Arjovsky *et al.*, 2019; Wu *et al.*, 2022] addresses the out-of-distribution (OOD) problem by identifying correlations that remain stable across training and testing environments. A prominent strategy within this paradigm is to create multiple training environments via data augmentation to develop robust predictors. For example, prior works have investigated distribution shifts caused by modifications to graph structure [Zhu *et al.*, 2021; Buffelli *et al.*, 2022], node features [Liu *et al.*, 2023; Wu *et al.*, 2024b], or graph sizes [Liu *et al.*, 2022b; Li *et al.*, 2022], under the assumption that target data conform to a specific shift type. The effectiveness of these methods makes the augmentation strategy a critical component for model generalization. However, the real-world distribution shifts and graph dynamics are unknown, which could be any combination of multiple distribution shift components. Fixed combinations of uniform augmentations, fail to comprehensively cover the potential shift configurations in practical scenarios, resulting in undesirable generalization on new unseen nodes.

To validate this insight, Figure 1 presents a preliminary analysis comparing augmentation strategies. We employ the EERM [Wu *et al.*, 2022] framework on the COLLAB dataset, which minimizes mean and variance risks across multiple augmentation branches. Specifically, we conduct a comparative analysis of two topology augmentation heuristics: 1) *uniform* augmentation, which perturbs edges uniformly at random; and 2) *spectral* augmentation, which flips edges within and across node clusters to induce principled changes in graph properties [Lin *et al.*, 2023; Lin *et al.*, 2022]. The results demonstrate that, under an equivalent perturbation budget, the spectral-based strategy consistently outperforms uniform augmentation across varying perturbation rates and numbers of branches. The performance gap between these two strategies motivates us to improve common uniform augmentations and explore effective graph augmentation strategies that better preserve graph invariance.

In this paper, we propose spectral augmentation as a surrogate for structural graph augmentation to enhance out-of-distribution generalization. Specifically, we define the spectral distance between an original graph and its perturbed counterpart measured by their Laplacian eigenvalues. The key idea of our approach is to decompose the input graph into multiple shift components by maximizing the variance of spectral distance and learn predictors that focus on robust spectral components. However, calculating the spectral distance requires the eigendecomposition of the graph Laplacian, which results in a prohibitive time complexity of $\mathcal{O}(N^3)$ with N nodes in a graph. We introduce an approximate solution based on the Rayleigh quotient, reducing the computational complexity from $\mathcal{O}(N^3)$ to $\mathcal{O}(m^2)$ with m non-isolated nodes in the graph. On top of this, we propose a novel objective that minimizes both the variance and the constructive risk across multiple spectral components. This objective encourages the classifier to focus on robust spectral components that remain stable under small edge perturbations (i.e., *environment-insensitive*) while reducing dependency on unstable components susceptible to perturbations (i.e., *environment-sensitive*). The main contributions of this work are summarized as follows:

- **A Novel Framework**: We propose a new approach to migrate dynamic graph distribution shifts by decomposing and mitigating complex distributional shifts on the spectral domain.
- **A Novel View**: We propose to extrapolate distribution shifts into a mixture of spectral components and encourage the model to encode crucial topological information that remains invariant under such shifts.
- **Superior Performance**: We conduct comprehensive experiments on both real-world and synthetic datasets to evaluate the performance of the proposed method. DSPA outperforms state-of-the-art methods on both link prediction and node classification tasks.

2 Related Works

OOD Generalization on Graphs. Out-of-distribution generalization on graphs often relies on invariant learning, which assumes that invariant factors related to the label remain stable across environments, while variant patterns may still be predictive. As a widely adopted framework, IRM [Arjovsky *et al.*, 2019] aims to learn a representation function that enables a shared classifier to perform optimally across multiple environments. Based on these principles, existing approaches for graph OOD generalization can be broadly categorized into two types: (i) Graph data augmentation [Liu *et al.*, 2023; Liu *et al.*, 2022a; Wu *et al.*, 2024b], which perturbs the graph structure in a rule-based manner; and (ii) Disentangled representation learning [Yu *et al.*, 2024; Zhang *et al.*, 2022; Zhang *et al.*, 2023], which attempts to separate feature representations into domain-invariant and domain-specific components. In this work, we focus on graph data augmentation to enhance the generalization capability of DyGNNs.

3 Preliminaries

Dynamic Graph. Let $\mathcal{G}$ be a dynamic graph defined over a global node set $\mathcal{V}$ with $|\mathcal{V}| = N$, which captures the temporal evolution of node interactions and attributes. The dynamic graph is represented as a sequence of discrete-time snapshots $\mathcal{G} = \{\mathcal{G}_t\}_{t=1}^{T}$, where each snapshot is defined as a tuple $\mathcal{G}_t = (\mathcal{V}_t, \mathcal{E}_t, \mathbf{X}_t, \mathbf{A}_t)$, comprising a node set $\mathcal{V}_t \subseteq \mathcal{V}$, an edge set $\mathcal{E}_t$, a node feature matrix $\mathbf{X}_t \in \mathbb{R}^{|\mathcal{V}_t| \times d_N}$, and an adjacency matrix $\mathbf{A}_t$. Here, d_N denotes the dimension of node features.

Graph Spectrum. For a given graph snapshot $\mathcal{G}_t = (\mathcal{V}_t, \mathcal{E}_t)$, the normalized Laplacian matrix is defined as: $\mathbf{L}_{\text{norm},t} = \mathbf{I} - \mathbf{D}_t^{-1/2}\mathbf{A}_t\mathbf{D}_t^{-1/2}$, where $\mathbf{I}$ is the identity matrix, $\mathbf{A}_t$ is the adjacency matrix, and $\mathbf{D}_t$ is the diagonal degree matrix. Since this matrix is real and symmetric, it admits the eigendecomposition $\mathbf{L}_{\text{norm},t} = \mathbf{U}_t\mathbf{\Lambda}_t\mathbf{U}_t^\top$. In this decomposition, $\mathbf{\Lambda}_t = \text{diag}(\lambda_{1,t}, \ldots, \lambda_{N,t})$ is the diagonal matrix of real eigenvalues; the set of these eigenvalues constitutes the graph spectrum. The corresponding matrix $\mathbf{U}_t = [\mathbf{u}_{1,t}, \ldots, \mathbf{u}_{N,t}]$ is an orthogonal matrix whose columns are the orthonormal eigenvectors, also known as the spectral bases.

OOD Generalization on Dynamic Graphs. In the context of dynamic graphs, we use the random variables $G_{1:t}$ and Y_t to indicate the sequence of ego-graphs up to time t and their true labels, while $\mathcal{G}_{1:t}$ and y_t refer to their specific empirical observations. We aim to learn a parameterized DyGNN mapping, $f_{\boldsymbol{\theta}} : \mathcal{G}_{1:t} \to \hat{Y}_t$, designed to generalize label predictions to unseen testing data.Aligning with standard practices in dynamic graph analysis [Gui *et al.*, 2022; Yang *et al.*, 2024], our research concentrates on node-level predictions. For any given node v associated with a temporal ego-graph $\mathcal{G}_{v,1:t}$, the training loss for the sample $(\mathcal{G}_{v,1:t}, y_t)$ is computed as $\ell(y_t, f_{\boldsymbol{\theta}}(\mathcal{G}_{v,1:t}))$. The core pursuit of OOD generalization is to ensure the model maintains stable performance across diverse testing environments, motivating the optimization objective below:

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \ \mathbb{E}_{(\mathcal{G}_{1:t}, Y_t, e)\sim p_{\text{tr}}}\Big[\ell(Y_t, f_{\boldsymbol{\theta}}(\mathcal{G}_{1:t}))\Big]. \quad (1)$$

4 Methodology

The framework of the proposed DSPA is illustrated in Figure 2. We formulate the generalization problem as a two-

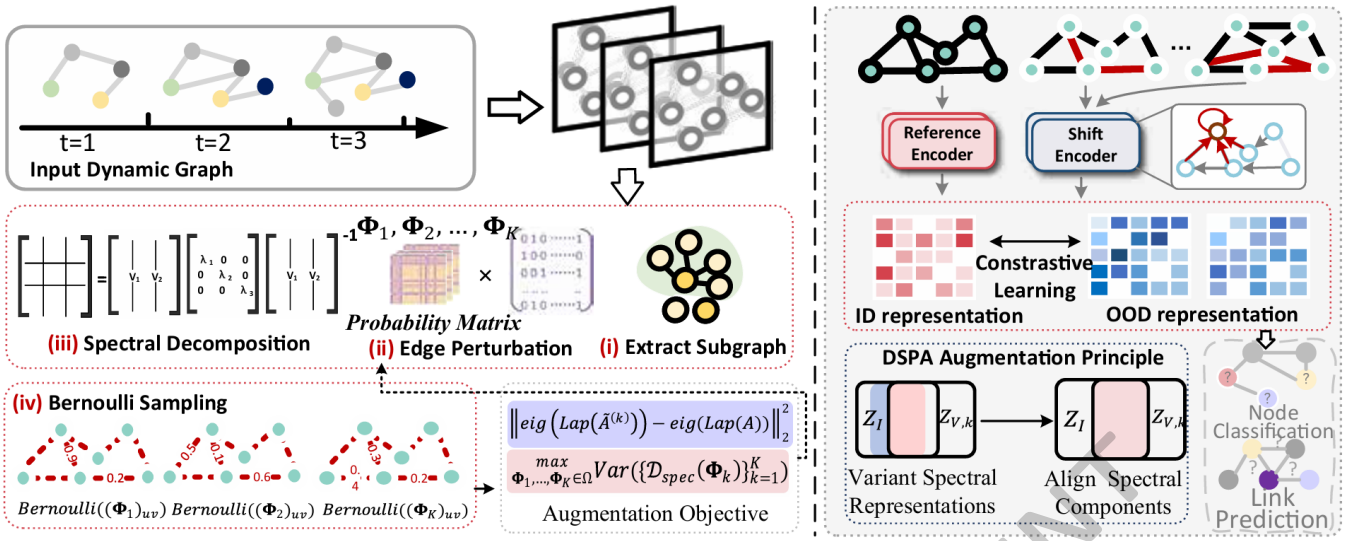


Figure 2: A conceptual diagram of the DSPA architecture. We construct DSPA and decompose the generalization problem into two distinct phases: (left) Decomposing the input graph spectrum into a mixture of spectral components to accommodate target shifts; (right) Mitigating distribution shifts by constraining each individual spectral component to reduce the corresponding surrogate shift.

stage framework: (i) **Surrogate estimation:** Decomposing the graph into a mixture of spectral components that serve as surrogates for target shifts, explicitly designed to capture the heterogeneity of the graph structure; and (ii) **Mitigation:** Neutralizing individual shift components via invariant learning to prioritize robust spectral component. We next present the specific methodology designed to realize this framework.

4.1 Spectral Augmentation for Shift Extrapolation

This section introduces our dynamic graph spectral augmentation strategy, which focuses on perturbing edges that induce significant changes in structural properties. We first formulate the augmentation principle as a spectral perturbation. Subsequently, we derive an approximation method to alleviate the computational overhead associated with large graphs.

Probabilistic Edge Flipping Mechanism

Instead of relying on fixed heuristic rules, we draw inspiration from SPAN [Lin *et al.*, 2023] and formulate topology augmentation as a learnable probabilistic process. We introduce a set of K learnable probability matrices, denoted as $\{\mathbf{\Phi}_1, \mathbf{\Phi}_2, \ldots, \mathbf{\Phi}_K\}$, where each $\mathbf{\Phi}_k \in [0,1]^{N\times N}$ governs the generation of the k-th augmented view.

For a specific view k, each entry $(\mathbf{\Phi}_k)_{uv}$ represents the probability of flipping the edge status between node u and v. To physically realize this augmentation, we first determine the permissible direction of modification using a structural direction matrix $\mathbf{D} \in \{-1, 1\}^{N\times N}$:

$$\mathbf{D} = \bar{\mathbf{A}} - \mathbf{A}, \tag{2}$$

where $\bar{\mathbf{A}}$ represents the adjacency of the complement graph (without self-loops). Consequently, $\mathbf{D}_{uv} = 1$ implies that the edge (u, v) does not exist and can be added, whereas $\mathbf{D}_{uv} = -1$ indicates an existing edge that can be removed.

To generate the augmented adjacency matrix $\tilde{\mathbf{A}}^{(k)}$, we sample a binary perturbation mask $\mathbf{B}^{(k)}$ from the Bernoulli distribution parameterized by $\mathbf{\Phi}_k$:

$$\mathbf{B}^{(k)}_{uv} \sim \text{Bernoulli}((\mathbf{\Phi}_k)_{uv}), \quad \tilde{\mathbf{A}}^{(k)} = \mathbf{A} + \mathbf{D} \circ \mathbf{B}^{(k)}. \tag{3}$$

Here, $\circ$ denotes the Hadamard product. If $\mathbf{B}^{(k)}_{uv} = 1$, the operation defined in $\mathbf{D}_{uv}$ is applied (adding or removing the edge); otherwise, the connection remains unchanged. The expectation of this stochastic augmentation is given by $\mathbb{E}[\tilde{\mathbf{A}}^{(k)}] = \mathbf{A} + \mathbf{D} \circ \mathbf{\Phi}_k$, making the topology shift differentiable with respect to the probability parameters.

Optimizing for Spectral Diversity

To ensure the augmented views cover a wide range of spectral shifts, we optimize the probability matrices $\{\mathbf{\Phi}_k\}_{k=1}^K$ to maximize the diversity of their resulting spectral properties. Specifically, we focus on the graph spectral distance variance.

Let $\mathcal{D}_{\text{spec}}(\mathbf{\Phi}_k)$ be the spectral distance of the expected augmented graph for the k-th view:

$$\mathcal{D}_{\text{spec}}(\mathbf{\Phi}_k) = \left\| \text{eig}\left(\text{Lap}\left(\tilde{\mathbf{A}}^{(k)}\right)\right) - \text{eig}(\text{Lap}(\mathbf{A})) \right\|_2^2. \tag{4}$$

The optimization objective is then formulated to maximize the variance of these distances across all K views:

$$\mathcal{L}_{aug} = \max_{\mathbf{\Phi}_1,\ldots,\mathbf{\Phi}_K \in \Omega} \text{Var}\left(\{\mathcal{D}_{\text{spec}}(\mathbf{\Phi}_k)\}_{k=1}^K\right), \tag{5}$$

where Ω constrains the perturbation budget. By maximizing this variance, the model is incentivized to generate views that exhibit distinct spectral characteristics.

Scalability

Directly optimizing the spectral distance in Eqn.4 requires performing eigendecomposition of $\mathcal{O}(N^3)$, which is computationally prohibitive for large graphs. To mitigate this, we employ a first-order spectral approximation based on matrix perturbation theory.

Proposition 1 (Spectral Approximation via Rayleigh Quotient)**.** *Let* $\mathbf{L}$ *be the Laplacian of the original graph and* $\{(\lambda_j, \mathbf{u}_j)\}$ *be its eigenpairs. Consider the perturbed Laplacian for the* k*-th view as* $\tilde{\mathbf{L}}^{(k)} = \mathbf{L} + \Delta\mathbf{L}(\mathbf{\Phi}_k)$*, where* $\Delta\mathbf{L}(\mathbf{\Phi}_k)$ *represents the Laplacian change induced by the probabilistic edge perturbation* $\mathbf{D} \circ \mathbf{\Phi}_k$*.*

Assuming the perturbation magnitude is bounded, the perturbed eigenvalues $\tilde{\lambda}_j^{(k)}$ *can be approximated using the* ***Rayleigh Quotient*** *of* $\tilde{\mathbf{L}}^{(k)}$ *evaluated on the original eigenvectors* $\mathbf{u}_j$*. Since the original eigenvectors are normalized, this simplifies to:*

$$\begin{aligned}\tilde{\lambda}_j^{(k)} \approx \mathcal{R}(\tilde{\mathbf{L}}^{(k)}, \mathbf{u}_j) &= \mathbf{u}_j^\top (\mathbf{L} + \Delta\mathbf{L}(\mathbf{\Phi}_k))\mathbf{u}_j \\ &= \lambda_j + \mathbf{u}_j^\top \Delta\mathbf{L}(\mathbf{\Phi}_k)\mathbf{u}_j. \end{aligned} \tag{6}$$

Proof. The Rayleigh Quotient $\mathcal{R}(\mathbf{M}, \mathbf{x}) = \frac{\mathbf{x}^\top \mathbf{M}\mathbf{x}}{\mathbf{x}^\top \mathbf{x}}$ is stationary at the eigenvectors of $\mathbf{M}$. Specifically, the gradient $\nabla_{\mathbf{x}}\mathcal{R}(\mathbf{M}, \mathbf{x})$ vanishes when $\mathbf{x}$ is an eigenvector. According to matrix perturbation theory, the perturbed eigenvector $\tilde{\mathbf{u}}_j$ can be expressed as $\tilde{\mathbf{u}}_j = \mathbf{u}_j + \mathcal{O}(\|\Delta\mathbf{L}\|)$. Substituting $\mathbf{u}_j$ into the Rayleigh Quotient of the perturbed matrix, we obtain:

$$\tilde{\lambda}_j^{(k)} = \mathcal{R}(\tilde{\mathbf{L}}^{(k)}, \tilde{\mathbf{u}}_j) \approx \mathcal{R}(\tilde{\mathbf{L}}^{(k)}, \mathbf{u}_j) + \mathcal{O}(\|\Delta\mathbf{L}\|^2). \tag{7}$$

□

Proposition 1 essentially states that utilizing the fixed basis $\mathbf{u}_j$ provides a robust first-order estimation of the spectral shift. Accordingly, we pre-compute the eigenvectors $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_N]$ of the original Laplacian $\mathbf{L}$ once. During the optimization of the probability matrices $\mathbf{\Phi}_k$, we treat $\mathbf{U}$ as a fixed basis and backpropagate gradients solely through the perturbation term $\Delta\mathbf{L}(\mathbf{\Phi}_k)$. This approximation avoids computationally expensive repetitive eigendecompositions.

Furthermore, in the context of dynamic graphs, isolated nodes in each snapshot are excluded from the spectral analysis, as they do not convey structural shift information. Restricting the analysis to the subgraph of m active nodes reduces the per-iteration complexity from $\mathcal{O}(N^3)$ to $\mathcal{O}(m^2)$, resulting in an overall training complexity of $\mathcal{O}(T \cdot L \cdot K \cdot m^2)$, where T, L, and K denote the number of time steps, optimization iterations, and augmented views, respectively.

4.2 Mitigating Shift Components by Invariant Learning

Having estimated the spectral surrogates via learnable probabilistic flipping, we propose an invariant spectral filtering to capture stable patterns in the spectral domain. We treat the original graph as a reference anchor and apply contrastive learning to align each augmented view with the original graph, thereby eliminating the effects of the induced spectral shift components.

The DSPA Workflow

We process the original graph and its augmentations through a shared encoder to capture both invariant and variant features. Formally, let $\mathcal{G}_t$ denote the ego-graph instance at snapshot t with adjacency matrix $\mathbf{A}_t$. The encoder is denoted as $h : \mathcal{G}_t \to \mathbb{R}^d$, which projects graph snapshots into a latent embedding space. h assumes two conceptual roles: the *Reference Encoder* serves as an in-distribution module that processes the original graph to extract the invariant component, whereas the *Shift Encoder* processes augmented graphs to generate variant components as illustrated in Figure 2. We refer to the scenario where h produces referentially invariant representations w.r.t. spectral surrogates while allowing heterogeneity across different spectral instances. The objective function ensures that h generalizes under multiple shifts.

Invariant Component ($\mathbf{Z}_I$): The encoder processes the original (in-distribution) graph $\mathcal{G}_t$, serving as a reference to produce a stable and distribution-invariant representation:

$$\mathbf{Z}_I = h(\mathcal{G}_t). \tag{8}$$

Variant Components ($\{\mathbf{Z}_{V,k}\}_{k=1}^K$): To capture potential distributional shifts, the encoder processes the K augmented views. Let $\mathcal{G}_t^{(k)}$ be the graph instance associated with the augmented adjacency $\tilde{\mathbf{A}}_t^{(k)}$ (from Eqn. 3). The variant representations are obtained as:

$$\mathbf{Z}_{V,k} = h(\mathcal{G}_t^{(k)}), \quad \text{for } k = 1, \ldots, K. \tag{9}$$

These generated representations are then propagated to a classifier μ. Consequently, the overall model is defined as $f = \mu \circ h$, with the prediction function articulated as:

$$f_\theta(\mathcal{G}_t) = \mu\big(h(\mathcal{G}_t), h(\mathcal{G}_t^{(1)}), \ldots, h(\mathcal{G}_t^{(K)})\big). \tag{10}$$

Unlike most DyGNNs that process all temporal snapshots simultaneously—which is both memory-consuming for large dynamic graphs and infeasible for true inductive learning. Instead, our approach processes snapshots sequentially. We leverage snapshot from the current timestep t to predict edges at the subsequent timestep $t + 1$. This strategy is suited for inductive learning and aligns with the evolutionary nature of dynamic graphs. We use GCN as the encoder h and train the model using stochastic gradient descent.

Training Objective

To learn representations that are robust to distributional shifts, we propose an objective that trains the model on both the original graph and its spectrally diverse augmentations.

• *Task Loss* ($\mathcal{L}_1$). The task loss guides the model to utilize invariant representations $\mathbf{Z}_I$ for downstream prediction. It is defined as:

$$\mathcal{L}_1 = \ell(\mu(\mathbf{Z}_I), \mathbf{Y}), \tag{11}$$

where μ is the prediction head, and ℓ is the task loss function.

• *Invariance Loss* ($\mathcal{L}_2$). To encourage consistency across spectrally shifted views, we minimize the variance of the task losses computed from the K augmented components. The invariance loss is defined as:

$$\mathcal{L}_2 = \mathrm{Var}\left(\{\ell_k\}_{k=1}^K\right) = \frac{1}{K}\sum_{k=1}^{K}\left(\ell_k - \bar{\ell}\right)^2, \tag{12}$$

where ℓ_k denotes the prediction logit corresponding to the k-th augmented view, and $\bar{\ell} = \frac{1}{K}\sum_{k=1}^K \ell_k$ represents the mean logit across all augmented views.

• *Contrastive Loss* ($\mathcal{L}_3$). Note that the augmentation strategy specifically targets edge perturbations that cause substantial

variations in the graph's structural characteristics. Drawing inspiration from the InfoMin principle [Zhao *et al.*, 2021; Lin *et al.*, 2023; Meng and Liu, 2023], we propose a contrastive learning objective that maximizes the correspondence between $\mathbf{Z}_I$ and $\mathbf{Z}_{V,k}$. This encourages the model to focus on *sensitive edges*, while edges that introduce structural instability are treated as noise and disregarded by the encoder. This ensures that the model's representations remain consistent across different augmented views.

Specifically, let D_i denote the denominator in the contrastive loss for sample i, computed as:

$$D_i = \sum_{\substack{j=1 \\ j\neq i}}^{N} e^{\frac{s(\mathbf{z}_{I,i},\mathbf{z}_{I,j})}{\tau}} + \sum_{l=1}^{K}\sum_{j=1}^{N} e^{\frac{s(\mathbf{z}_{I,i},\mathbf{z}_{V,l,j})}{\tau}}. \tag{13}$$

Then, the contrastive loss $\mathcal{L}_3$ is formulated as:

$$\mathcal{L}_3 = -\frac{1}{K}\sum_{k=1}^{K}\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp\left(\frac{s(\mathbf{z}_{I,i},\mathbf{z}_{V,k,i})}{\tau}\right)}{D_i}, \tag{14}$$

where $s(\cdot,\cdot)$ denotes the similarity function between two latent vectors, and τ is the temperature hyperparameter.

• *Total Loss.* Following recent advances in adversarial training on graphs [Wu *et al.*, 2022; Feng *et al.*, 2019], we optimize the network's parameters independently. We maximize the spectral diversity of perturbations via $\mathbf{\Phi}$ while simultaneously minimizing the overall objective with respect to the model parameters $\boldsymbol{\theta}$:

$$\min_{\boldsymbol{\theta}} \left(\mathcal{L}_1 + \mathcal{L}_2(\mathbf{\Phi}^*) + \alpha\mathcal{L}_3(\mathbf{\Phi}^*)\right) \tag{15}$$

$$\text{s.t.} \quad \mathbf{\Phi}^* = \arg\max_{\mathbf{\Phi}\in\Omega} \mathcal{L}_{\text{aug}}, \tag{16}$$

where α is a hyperparameter balancing the contributions, and $\mathcal{L}_{\text{aug}}$ is the spectral distance variance defined in Eqn. (5).

5 Experiments

5.1 Experimental Setup

Datasets. We conduct experiments on four node property prediction datasets: **COLLAB** [Tang *et al.*, 2012], **Yelp** [Sankar *et al.*, 2020], **ACT** [Kumar *et al.*, 2019], and **Aminer** [Zhang *et al.*, 2023]. Following [Yuan *et al.*, 2023; Zhang *et al.*, 2023], we construct distinct in-distribution (ID) and out-of-distribution (OOD) partitions for each dataset. For COLLAB, Yelp, and ACT, we adopt the challenging inductive future link prediction task, where the model is evaluated on a dynamic graph containing nodes that are unseen during training. For Aminer, we perform a node classification task on a citation network; here, nodes represent papers, and a timestamped edge from u to v indicates that paper u (published at time t) cites paper v.

Baselines. We compare DSPA with ERM and twelve baseline methods in four categories: (1) **Invariant Learning methods:** EERM [Wu *et al.*, 2022], IRM [Arjovsky *et al.*, 2019], CaNet [Wu *et al.*, 2024a]. (2) **Dynamic Graph Learning methods:** VGAE [Chen *et al.*, 2024], EGCN [Pareja *et al.*, 2020], DySAT [Sankar *et al.*, 2020]. (3) **Dynamic Graph OOD methods:** DIDA [Zhang *et al.*, 2022], I-DIDA [Zhang *et al.*, 2024], EAGLE [Yuan *et al.*, 2023], DyCIL [Zhang *et al.*, 2026]. (4) **Spectral Learning methods:** SPAN [Lin *et al.*, 2023], SILD [Zhang *et al.*, 2023].

Training Settings. All experiments are repeated five times with different random seeds. Validation set performance determines both model checkpoint selection and early stopping. While we present results for ID and OOD test sets, we focus primarily on OOD metrics to assess generalization. Models are optimized via Adam with weight decay for up to 200 epochs, and we report the test performance corresponding to the optimal validation epoch.

5.2 Distribution Shift on Real World Datasets

Settings. We use 3 real-world dynamic graph datasets: YELP, ACT, and Aminer. Following the setup in [Zhang *et al.*, 2023], we adopt the challenging inductive link prediction task on the COLLAB, YELP, and ACT datasets, where testing is performed on dynamic graphs from domains that are unseen during training. For the node classification task, we train on the Aminer dataset using a temporal split, where papers published between 2009 and 2014 are used for training, and those published after 2015 are used for testing.

Results. The results are presented in Table 1. Here, *ID* and *OOD* indicate whether distribution shifts are present in the test set. We have the following observations: **(i)** Although the baselines perform well on the training data, their effectiveness declines significantly during testing. These empirical results support our hypothesis that DyGNNs tend to overfit spurious correlations or non-causal features, leading to suboptimal generalization. **(ii)** When integrated into our framework, invariant methods such as EERM and IRM exhibit improvements over state-of-the-art baselines. This demonstrates that our training paradigm is more effective for handling distribution shifts in inductive settings. **(iii)** DSPA consistently outperforms the baselines under distribution shifts. We attribute this improvement to the training objective, which encourages invariance to distributional changes by leveraging diverse augmented views.

5.3 Distribution Shift on Synthetic Datasets

Settings. We follow [Zhang *et al.*, 2022] to generate additional node features that simulate distribution shifts. Specifically, we sample a mix of positive and negative links from the next time step's edges, $\mathbf{A}_{t+1}$. A time-dependent sampling probability $p(t)$ is used to select $p(t)|\mathbf{A}_{t+1}|$ positive links and $(1-p(t))|\mathbf{A}_{t+1}|$ negative links, where $p(t) = \bar{p} + \sigma\cos(t)$. The original node features and the synthesized features are concatenated as $[\mathbf{X}_t\|\mathbf{X}'_t]$ and fed into the model. A larger $p(t)$ introduces stronger spurious correlations between the perturbed features X'_t and future environments. To achieve better generalization, the model should avoid relying on variant patterns that exploit spurious correlations introduced by the additional features. In our experiments, we manipulate the level of spurious correlation by varying $\bar{p}$ during training. Specifically, we set $\bar{p}$ to 0.4, 0.6, and 0.8 during training, and fix it to 0.1 during the testing stage.

Results. The results are shown in Table 2. We have the following observations: **(i)** Our method better handles distribution shifts than the baselines, especially when the shifts are

Dataset		ACT		Yelp		Node Classification		
Split	Model	ID	OOD	ID	OOD	Aminer15	Aminer16	Aminer17
Base	ERM	92.20 ± 0.47	86.29 ± 0.33	71.79 ± 1.57	71.19 ± 1.51	50.04 ± 0.06	50.02 ± 0.03	42.91 ± 0.04
Invariant Learning	EERM	OOM	OOM	OOM	OOM	OOM	OOM	OOM
	IRM	87.92 ± 4.92	86.27 ± 3.20	58.89 ± 0.15	52.57 ± 0.09	51.24 ± 0.24	53.13 ± 0.22	48.42 ± 0.16
	CaNet	88.62 ± 0.23	87.61 ± 0.20	58.35 ± 0.48	57.71 ± 0.68	51.43 ± 0.03	51.61 ± 0.03	46.67 ± 0.02
Dynamic Graph Learning	VGAE	89.86 ± 0.43	87.31 ± 0.74	66.73 ± 0.79	62.83 ± 0.78	44.89 ± 0.03	48.07 ± 0.31	40.28 ± 0.23
	EGCN	69.17 ± 0.17	64.29 ± 0.09	71.36 ± 0.25	65.24 ± 1.31	38.54 ± 0.13	35.25 ± 0.21	30.31 ± 0.11
	DySAT	63.79 ± 3.27	64.12 ± 3.35	43.00 ± 0.83	45.58 ± 0.69	46.46 ± 0.33	42.42 ± 0.34	40.42 ± 0.04
Dynamic Graph OOD	DIDA	83.87 ± 0.27	77.03 ± 0.34	67.50 ± 2.54	71.52 ± 2.60	52.24 ± 0.19	43.25 ± 0.25	40.24 ± 0.06
	I-DIDA	85.61 ± 0.17	85.21 ± 0.18	75.53 ± 0.26	**73.19 ± 0.30**	52.32 ± 0.28	52.23 ± 0.47	45.52 ± 0.71
	EAGLE	84.44 ± 0.22	82.76 ± 0.78	67.86 ± 0.18	64.85 ± 0.65	OOM	OOM	OOM
	DyCIL	**93.35 ± 0.16**	85.51 ± 0.12	**81.79 ± 0.02**	51.54 ± 0.77	52.18 ± 0.35	54.27 ± 0.21	45.98 ± 0.15
Graph Spectral Learning	SPAN	88.73 ± 0.14	87.23 ± 0.42	69.47 ± 0.16	69.68 ± 0.27	52.47 ± 0.14	53.66 ± 0.63	**52.76 ± 0.30**
	SILD	72.89 ± 0.31	75.86 ± 0.52	65.11 ± 0.32	66.51 ± 0.01	51.43 ± 0.32	50.45 ± 1.24	46.13 ± 0.24
	DSPA	92.67 ± 0.59	**88.90 ± 1.10**	74.11 ± 0.90	72.30 ± 1.22	**52.63 ± 0.35**	**54.99 ± 0.41**	50.78 ± 0.23
	Improve	△−0.68	△+1.29	△−7.68	△−0.89	△+0.16	△+0.72	△−1.98

Table 1: Experimental results for the link prediction and node classification tasks are reported as mean AUC ± standard deviation. The best and second-best results are highlighted in **bold** and underline. "OOM" denotes an out-of-memory error under a 24GB memory constraint.

	Synthetic-Collab		
Model	$\bar{p}$=0.4	$\bar{p}$=0.6	$\bar{p}$=0.8
ERM	79.83 ± 0.83	81.20 ± 0.48	80.98 ± 0.32
EERM	OOM	OOM	OOM
IRM	80.57 ± 0.39	80.58 ± 0.51	80.50 ± 0.43
CaNet	81.92 ± 0.28	80.20 ± 0.21	82.86 ± 0.45
VGAE	80.67 ± 0.13	78.76 ± 0.64	79.33 ± 0.20
EGCN	80.14 ± 0.21	75.24 ± 0.14	74.24 ± 0.27
DySAT	84.29 ± 0.40	84.40 ± 0.40	84.36 ± 0.39
DIDA	83.19 ± 0.65	77.64 ± 0.24	62.78 ± 0.25
I-DIDA	61.99 ± 0.42	61.28 ± 0.21	61.08 ± 0.10
EAGLE	85.35 ± 0.16	86.16 ± 0.21	80.14 ± 0.56
DyCIL	**91.02 ± 0.12**	87.34 ± 0.04	83.09 ± 0.32
SPAN	85.78 ± 0.17	85.77 ± 0.19	85.75 ± 0.11
SILD	59.10 ± 0.28	59.80 ± 0.16	59.17 ± 0.32
Ours	86.98 ± 0.77	**87.59 ± 0.78**	**86.34 ± 0.52**
Improve	△−4.04	△+0.25	△+0.59

Table 2: Experimental results for the node classification task, reported as mean AUC ± standard deviation. The best and runner-up results are highlighted in **bold** and underlined, respectively.

strong. In the link prediction task, DSPA surpasses DyCIL by an average of 1.17%. **(ii)** Our method exploits invariant patterns across different levels of distribution shift. As the shift intensity increases, almost all baselines exhibit a significant performance drop. In contrast, the performance of DSPA degrades slightly. This demonstrates that DSPA learns robust structures and alleviates reliance on variant patterns, enabling the encoder to maintain stable outputs across varying shifts.

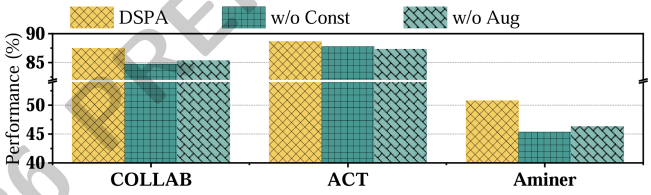


Figure 3: Results of ablation studies.

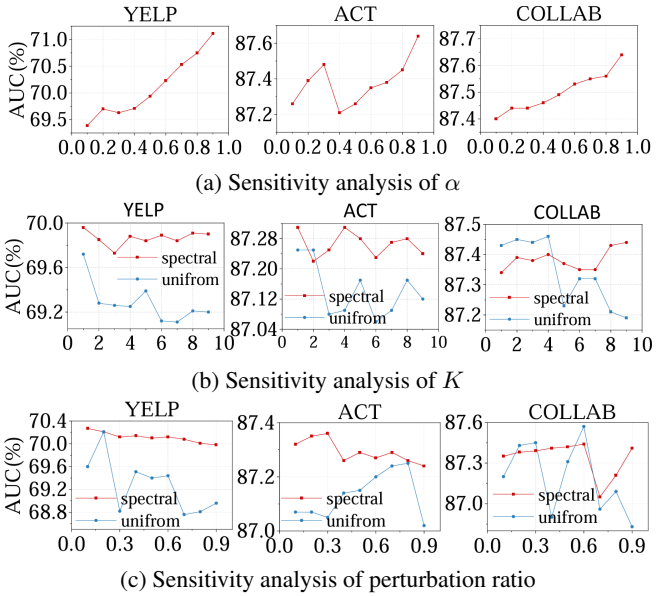
5.4 Ablation Study

We validate the effectiveness of various components in DSPA by individually removing these modules. The ablated version 'w/o Const' removes contrastive loss $\mathcal{L}_3$, and 'w/o Aug' is trained without the spectral augmentation only on the origin input graph. From Figure 3, we have the following observations. First, our proposed method consistently outperforms all its variants. This demonstrates the effectiveness of each component within our methodology. Second, the performance of both 'w/o Const' and 'w/o Aug' degrades significantly. This decline indicates that our spectral augmentation scheme and contrastive loss help the model focusing on invariant patterns under distribution shifts.

5.5 Hyperparameter Sensitivity Analysis

Impact of Parameter α. Figure 4(a) illustrates the performance of DSPA under different values of the hyperparameter α. As the weight of the contrastive loss $\mathcal{L}_3$ increases, the AUC improves across all datasets. This trend suggests that enhancing the alignment between invariant and perturbed representations by contrastive learning strengthens the model's robustness to structural perturbations, particularly in graphs with high sensitivity to edge modifications.

Impact of Parameter K. The hyperparameter K controls the number of augmentation branches. As shown in Fig-



(a) Sensitivity analysis of α

(b) Sensitivity analysis of K

(c) Sensitivity analysis of perturbation ratio

Figure 4: Parameter sensitivity analysis.

Datasets	YELP	ACT	COLLAB
#nodes	13,095	20,408	23,035
Pre-computation time (K=5)	69	95	67
Training time (iteration=100)	738	1,198	905

Table 3: Empirical running time (s) on four datasets.

ure 4(b), the model's OOD performance is generally insensitive to the choice of K on the YELP and ACT datasets, while exhibiting a moderate dependency on K for COLLAB. This discrepancy can be attributed to the more pronounced distribution shifts in COLLAB, which make generalization inherently more challenging. Moreover, the proposed spectral augmentation strategy consistently outperforms uniformly random edge perturbation, as the latter ignores the unequal influence of different edges on graph structural properties.

Impact of perturbation ratio. As shown in Figure 4(c), increasing the perturbation ratio slightly degrades performance on YELP and ACT, while generally improving performance on COLLAB. This difference is likely due to variations in graph density and structural redundancy across datasets. In particular, COLLAB is a dense network with abundant redundant structures, where a higher perturbation ratio helps generate more informative augmented views.

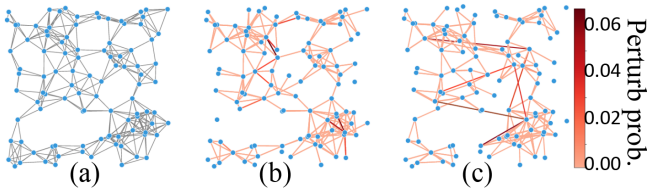


Figure 5: A case study on a random geometric graph.

Perturbation Ratio	0.1	0.3	0.5	0.7	0.9
SAE	0.058	0.129	0.180	0.225	0.260
AUC (Full)	72.35	72.14	72.01	71.57	71.33
AUC (Approx)	**72.49**	**72.46**	**72.51**	**72.05**	**71.92**

Table 4: Comparisons of approximation error (SAE) and AUC scores between Exact (Full) and Approximate (Approx) methods.

5.6 Spectral Approximation Analysis

To empirically validate the approximation bound in Proposition 1, we compare the Rayleigh quotient approximation (*Approx*) with exact eigendecomposition (*Full*) under different perturbation ratios on the COLLAB dataset. As shown in Table 4, the Spectral Approximation Error (SAE) increases monotonically with the ratio, empirically confirming that the error is dominated by higher-order terms $\mathcal{O}(\|\Delta\mathbf{L}\|^2)$. Despite this, the performance of *Approx* remains robust and consistently outperforms *Full*. This suggests that the first-order approximation acts as a *spectral regularizer*, effectively filtering out high-frequency structural noise captured by the exact decomposition while retaining semantically relevant shifts.

5.7 Visualization of Spectral Augmentation

To illustrate structural changes induced by spectral augmentation, we visualize a random geometric graph. Figure 5(a) shows the original topology, Figure 5(b) the graph maximizing $\mathcal{D}$spec, and Figure 5(c) the graph minimizing $\mathcal{D}$spec. Maximizing the spectral norm tends to remove inter-cluster edges, while minimizing it promotes cross-cluster connections. This augmentation mainly targets spurious edges affecting structural properties. Similar observations were reported in previous works [Lin *et al.*, 2023; Lin *et al.*, 2022]: smaller eigenvalues capture smooth intra-cluster similarities, whereas larger eigenvalues reflect sharp inter-cluster differences.

6 Conclusion

In this paper, we propose DSPA, a novel framework to migrate complex distribution shifts on dynamic graphs in the spectral domain. DSPA distinguishes itself from traditional invariant learning methods, which primarily rely on augmenting the training data to cover the test distribution shifts. Instead, our approach guides the encoder to identify sensitive edges, whose perturbations cause significant spectral variations. This encourages the model to focus on robust spectral components for improved generalization. Experimental results demonstrate that DSPA consistently outperforms baseline methods on both real-world and synthetic datasets.

Acknowledgements

This research is supported by the National Natural Science Foundation of China (Grant No. 62472263, 62072288), Humanities and Social Science Research Project of the Ministry of Education, China (Grant No. 24YJAZH058), the Natural Science Foundation of Shandong Province, China (Grant No. ZR2024MF034), the CAHE 2024 Higher Education Scientific Research Planning Project (Grant No. 24XX0101).

References

[Abolghasemi *et al.*, 2024] Roza Abolghasemi, Enrique Herrera Viedma, Paal Engelstad, Youcef Djenouri, and Anis Yazidi. A graph neural approach for group recommendation system based on pairwise preferences. *Information Fusion*, 107:102343, 2024.

[Arjovsky *et al.*, 2019] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

[Buffelli *et al.*, 2022] Davide Buffelli, Pietro Liò, and Fabio Vandin. Sizeshiftreg: a regularization method for improving size-generalization in graph neural networks. *Advances in Neural Information Processing Systems*, 35:31871–31885, 2022.

[Chen *et al.*, 2024] Minxiao Chen, Haitao Yuan, Nan Jiang, Zhifeng Bao, and Shangguang Wang. Urban traffic accident risk prediction revisited: Regionality, proximity, similarity and sparsity. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 281–290, 2024.

[Feng *et al.*, 2019] Fuli Feng, Xiangnan He, Jie Tang, and Tat-Seng Chua. Graph adversarial training: Dynamically regularizing based on graph structure. *IEEE Transactions on Knowledge and Data Engineering*, 33(6):2493–2504, 2019.

[Gui *et al.*, 2022] Shurui Gui, Xiner Li, Limei Wang, and Shuiwang Ji. Good: A graph out-of-distribution benchmark. *Advances in Neural Information Processing Systems*, 35:2059–2073, 2022.

[Jiang *et al.*, 2024] Jicun Jiang, Xiaodi Liu, Zengwen Wang, Weiping Ding, and Shitao Zhang. Large group emergency decision-making with bi-directional trust in social networks: A probabilistic hesitant fuzzy integrated cloud approach. *Information Fusion*, 102:102062, 2024.

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

[Li *et al.*, 2022] Haoyang Li, Xin Wang, Ziwei Zhang, and Wenwu Zhu. Out-of-distribution generalization on graphs: A survey. *arXiv e-prints*, pages arXiv–2202, 2022.

[Lin *et al.*, 2022] Lu Lin, Ethan Blaser, and Hongning Wang. Graph structural attack by perturbing spectral distance. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 989–998, 2022.

[Lin *et al.*, 2023] Lu Lin, Jinghui Chen, and Hongning Wang. Spectral augmentation for self-supervised learning on graphs. In *The Eleventh International Conference on Learning Representations*, 2023.

[Liu *et al.*, 2022a] Gang Liu, Tong Zhao, Jiaxin Xu, Tengfei Luo, and Meng Jiang. Graph rationalization with environment-based augmentations. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1069–1078, 2022.

[Liu *et al.*, 2022b] Songtao Liu, Rex Ying, Hanze Dong, Lanqing Li, Tingyang Xu, Yu Rong, Peilin Zhao, Junzhou Huang, and Dinghao Wu. Local augmentation for graph neural networks. In *International conference on machine learning*, pages 14054–14072, 2022.

[Liu *et al.*, 2023] Yang Liu, Xiang Ao, Fuli Feng, Yunshan Ma, Kuan Li, Tat-Seng Chua, and Qing He. Flood: A flexible invariant learning framework for out-of-distribution generalization on graphs. In *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining*, pages 1548–1558, 2023.

[Meng and Liu, 2023] En Meng and Yong Liu. Graph contrastive learning with graph info-min. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, page 4195–4199, 2023.

[Pareja *et al.*, 2020] Aldo Pareja, Giacomo Domeniconi, Jie Chen, Tengfei Ma, Toyotaro Suzumura, Hiroki Kanezashi, Tim Kaler, Tao Schardl, and Charles Leiserson. Evolvegcn: Evolving graph convolutional networks for dynamic graphs. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 5363–5370, 2020.

[Sankar *et al.*, 2020] Aravind Sankar, Yanhong Wu, Liang Gou, Wei Zhang, and Hao Yang. 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*, pages 519–527, 2020.

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

[Wu *et al.*, 2022] Qitian Wu, Hengrui Zhang, Junchi Yan, and David Wipf. Handling distribution shifts on graphs: An invariance perspective. In *International Conference on Learning Representations*, 2022.

[Wu *et al.*, 2024a] Qitian Wu, Nie Fan, Chenxiao Yang, Tianyi Bao, and Junchi Yan. Graph out-of-distribution generalization via causal intervention. In *The Web Conference*, 2024.

[Wu *et al.*, 2024b] Shirley Wu, Kaidi Cao, Bruno Ribeiro, James Zou, and Jure Leskovec. Graphmetro: Mitigating complex graph distribution shifts via mixture of aligned experts. *Advances in Neural Information Processing Systems*, 37:9358–9387, 2024.

[Xia *et al.*, 2024] Zhichao Xia, Yong Zhang, Jielong Yang, and Linbo Xie. Dynamic spatial–temporal graph convolutional recurrent networks for traffic flow forecasting. *Expert Systems with Applications*, 240:122381, 2024.

[Yang *et al.*, 2024] Kuo Yang, Zhengyang Zhou, Qihe Huang, Limin Li, Yuxuan Liang, and Yang Wang. Improving generalization of dynamic graph learning via environment prompt. *Advances in Neural Information Processing Systems*, 37:70048–70075, 2024.

[Yu *et al.*, 2024] Xi Yu, Huan-Hsin Tseng, Shinjae Yoo, Haibin Ling, and Yuewei Lin. Insure: an information theory inspired disentanglement and purification model for domain generalization. *IEEE Transactions on Image Processing*, 33:3508–3519, 2024.

[Yuan *et al.*, 2023] Haonan Yuan, Qingyun Sun, Xingcheng Fu, Ziwei Zhang, Cheng Ji, Hao Peng, and Jianxin Li. Environment-aware dynamic graph learning for out-of-distribution generalization. *Advances in Neural Information Processing Systems*, 36:49715–49747, 2023.

[Zhang and Gan, 2024] Hang Zhang and Mingxin Gan. Mbdl: Exploring dynamic dependency among various types of behaviors for recommendation. *Information Systems*, 124:102407, 2024.

[Zhang *et al.*, 2022] Zeyang Zhang, Xin Wang, Ziwei Zhang, Haoyang Li, Zhou Qin, and Wenwu Zhu. Dynamic graph neural networks under spatio-temporal distribution shift. *Advances in neural information processing systems*, 35:6074–6089, 2022.

[Zhang *et al.*, 2023] Zeyang Zhang, Xin Wang, Ziwei Zhang, Zhou Qin, Weigao Wen, Hui Xue, Haoyang Li, and Wenwu Zhu. Spectral invariant learning for dynamic graphs under distribution shifts. *Advances in Neural Information Processing Systems*, 36:6619–6633, 2023.

[Zhang *et al.*, 2024] Zeyang Zhang, Xin Wang, Haibo Chen, Haoyang Li, and Wenwu Zhu. Disentangled dynamic graph attention network for out-of-distribution sequential recommendation. *ACM Transactions on Information Systems*, 43(1):1–42, 2024.

[Zhang *et al.*, 2026] Xinxun Zhang, Pengfei Jiao, Mengzhou Gao, Tianpeng Li, and Xuan Guo. Towards ood generalization in dynamic graphs via causal invariant learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 16379–16387, 2026.

[Zhao *et al.*, 2021] Tong Zhao, Yozen Liu, Leonardo Neves, Oliver Woodford, Meng Jiang, and Neil Shah. Data augmentation for graph neural networks. In *Proceedings of the aaai conference on artificial intelligence*, volume 35, pages 11015–11023, 2021.

[Zhou *et al.*, 2020] Zhengyang Zhou, Yang Wang, Xike Xie, Lianliang Chen, and Hengchang Liu. Riskoracle: A minute-level citywide traffic accident forecasting framework. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 1258–1265, 2020.

[Zhu *et al.*, 2021] Qi Zhu, Natalia Ponomareva, Jiawei Han, and Bryan Perozzi. Shift-robust gnns: Overcoming the limitations of localized graph training data. *Advances in Neural Information Processing Systems*, 34:27965–27977, 2021.

[Zhu *et al.*, 2024] Haorui Zhu, Fei Xiong, Hongshu Chen, Xi Xiong, and Liang Wang. Incorporating a triple graph neural network with multiple implicit feedback for social recommendation. *ACM Transactions on the Web*, 18(2):1–26, 2024.