Out-of-Distribution Generalization on Graphs via Progressive Inference

Yiming Xu[1,2], **Bin Shi**[1,2*], **Zhen Peng**[1,2], **Huixiang Liu**[1,2], **Bo Dong**[2,3], **Chen Chen**[4]

[1]School of Computer Science and Technology, Xi'an Jiaotong University
[2]Shaanxi Provincial Key Laboratory of Big Data Knowledge Engineering, Xi'an Jiaotong University
[3]School of Distance Education, Xi'an Jiaotong University
[4]University of Virginia, Charlottesville, Virginia, USA
{xym0924, liuhxwork}@stu.xjtu.edu.cn, {shibin, dong.bo}@xjtu.edu.cn, zhenpeng27@outlook.com, zrh6du@virginia.edu

Abstract

The development and evaluation of graph neural networks (GNNs) generally follow the independent and identically distributed (i.i.d.) assumption. Yet this assumption is often untenable in practice due to the uncontrollable data generation mechanism. In particular, when the data distribution shows a significant shift, most GNNs would fail to produce reliable predictions and may even make decisions randomly. One of the most promising solutions to improve the model generalization is to pick out causal invariant parts in the input graph. Nonetheless, we observe a significant distribution gap between the causal parts learned by existing methods and the ground-truth, leading to undesirable performance. In response to the above issues, this paper presents GPro, a model that learns graph causal invariance with progressive inference. Specifically, the complicated graph causal invariant learning is decomposed into multiple intermediate inference steps from easy to hard, and the perception of GPro is continuously strengthened through a progressive inference process to extract causal features that are stable to distribution shifts. We also enlarge the training distribution by creating counterfactual samples to enhance the capability of the GPro in capturing the causal invariant parts. Extensive experiments demonstrate that our proposed GPro outperforms the state-of-the-art methods by 4.91% on average. For datasets with more severe distribution shifts, the performance improvement can be up to 6.86%.

Code — https://github.com/yimingxu24/GPro

Introduction

The powerful graph representation learning abilities of graph neural networks (GNNs) have been widely acknowledged in both academia and industry, and have been proven to be effective in a variety of applications, such as recommender systems (Niu et al. 2020; Xia et al. 2022; Seo et al. 2022; Yan et al. 2023), finance (Liu et al. 2021; Zhang et al. 2022; Shi et al. 2023; Zheng et al. 2023), life sciences (Hsieh et al. 2021; Zhu et al. 2022; Su et al. 2022; Fu et al. 2023) and autonomous driving (Gao et al. 2020; Xu et al. 2022). Despite their remarkable success, existing GNNs typically

rely on the assumption that training and testing data are independently and identically distributed (i.i.d.). However, this assumption often becomes untenable in realistic scenarios due to the uncontrollable underlying data generation mechanism (Bengio et al. 2019; Li et al. 2022b). Several recent studies have revealed the vulnerability of GNNs in the face of differently distributed data (Ding et al. 2021; Gui et al. 2022). The lack of out-of-distribution (OOD) generalization capabilities hinders the deployment of GNNs in multiple high-risk scenarios in the open world.

Recently, one of the most promising directions for improving out-of-distribution (OOD) generalization is the method based on causal invariant learning. Specifically, most existing studies (Sui et al. 2022a; Fan et al. 2022; Wu et al. 2022) obtain node representations by GNNs and identify causal invariant substructures and features from the input graph in a single-step manner, such as directly applying dot product operations or MLPs. Finally, they introduce specialized optimization objectives and constraints to minimize the risk of causal invariance across different distributions. However, the attention in existing works has focused on the design of optimization objectives, but ignored the exploration of model architectures. Unlike grid-like data, the intricate nature of graphs presents a substantial challenge to this problem since the topological structure leads to complex coupling associations between the causal and non-causal parts. This challenge raises a serious concern: *How powerful is this kind of single-step manner in uncovering causal substructures in the OOD scenarios?*

To validate this concern, we conduct an empirical study to investigate the effectiveness of existing methods in tackling this challenge. Specifically, in an OOD dataset, we visualize the causal features learned by existing methods and the ground-truth causal features (learned only by feeding the causal substructures into the GNN model) in the feature space. Unfortunately, our findings reveal a significant distribution gap between these two sets of features (details in Figure 4 and Figure 5). In other words, existing methods fail to capture high-quality causal invariant features, adversely affecting the generalization ability of the model. Indeed, when tackling complex problems, humans typically rely on multi-step inference rather than expecting immediate accurate results. For example, mathematicians break down difficult proofs into a series of sub-proofs and iteratively ad-
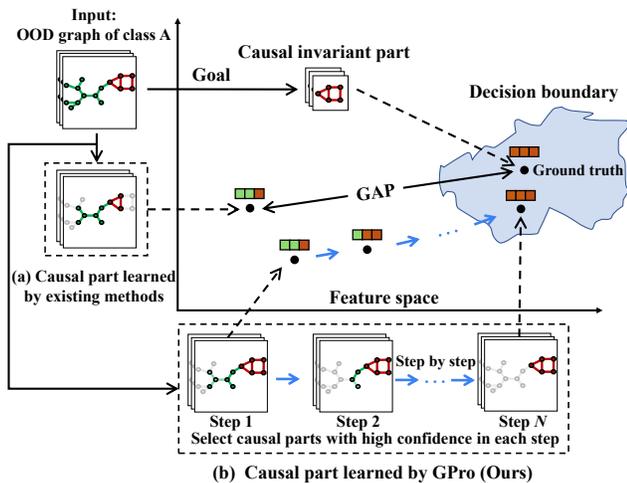
Figure 1: An illustration of the differences between existing methods and our proposed solution GPro. (a) The standard methods incorporate a significant amount of non-causal information (the green part in the input) in the learned features, resulting in a deviation from the decision boundary. (b) Our method is continuously refined via progressive inference to approach the ground truth.

vance from intermediate results to achieve conclusive solutions. Inspired by this insight, we explore the encoder architecture based on the progressive inference paradigm on graphs to emulate the cognitive processes employed by humans when solving complex problems, aiming to enhance generalization capabilities, as illustrated in Figure 1.

In this paper, we present a new framework to learn **G**raph causal invariance via **Pro**gressive inference, called GPro, which decomposes the complex problem of discerning causal substructures and features into multiple intermediate inference steps from easy to hard. Specifically, each inference step further separates out the non-causal substructure with high confidence from the intermediate result learned in the previous inference step via an attention-based substructure context inference block. By stacking multiple such blocks, GPro mimics a step-by-step thought process to refine an accurate answer. Since the causal and non-causal parts are complementary, instead of only focusing on identifying the causal substructures, GPro employs a dual-tower model that concurrently identifies the causal and non-causal substructures, which aims to facilitate mutual assistance. Furthermore, to make the progressive inference process better capture the causal invariant parts, we propose to enlarge the training distribution by constructing different counterfactual samples through two feature-level data augmentation techniques. We also propose a novel supervised contrastive learning loss in graph causal invariant learning that leverages the supervised signals within and between samples of a batch. Our main contributions are summarized as follows:

• We propose the new concept of progressive inference in graph out-of-distribution generalization, which transforms the invariant learning process into multiple inference steps. This overcomes the existing model's inability to effec-

tively disentangle the complex coupling associations between causal and non-causal substructures limitations.

• We introduce sophisticated feature augmentation strategies to enlarge the training distribution by generating counterfactual samples. Moreover, we present a novel supervised contrastive learning objective that effectively utilizes inter-sample supervised signals to further enhance the generalization ability of the model.

• The experimental results demonstrate that our GPro produces state-of-the-art results on 11 established baselines, and outperforms the sub-optimal baseline by 4.91% on average. Qualitative and quantitative analysis of progressive inference and ablation studies corroborate the effectiveness of each component in GPro.

Related Work

Graph neural networks have demonstrated impressive performance in a variety of applications (Qiu et al. 2018; Wang et al. 2019; Fu et al. 2022; Wang et al. 2022; Xue et al. 2022; Fu et al. 2024). However, most existing methods fail in terms of model generalization, which hinders the deployment of GNNs in high-risk applications in the open world. Recent studies are exploring how to improve the generalizability of GNNs in OOD scenarios, with efforts focusing on data-centric methods and causal invariant learning approaches. Data-centric methods (Sui et al. 2022b; Li et al. 2023) improve OOD generalization ability through data augmentation. Causal invariant learning methods (Wu et al. 2022; Li et al. 2022c,a) emphasize minimizing causal invariant risks in different distributions by introducing specialized optimization objectives and constraints. For example, StableGNN (Fan et al. 2023) extracts causal structures from input graphs to help the model eliminate spurious correlations. CAL (Sui et al. 2022a) and DisC (Fan et al. 2022) divide the input graph into causal and non-causal graphs, and encourages a stable relationship between causal estimates and predictions. CIGA (Chen et al. 2022) proposes an information-theoretic objective to capture the invariance of graphs to guarantee OOD generalization under various distributional shifts. FLOOD (Liu et al. 2023) constructs multiple environments from graph data augmentation and learns invariant representation under risk extrapolation. For more extensive work, please refer to (Li et al. 2022b). Although these methods show higher effectiveness, they still suffer from at least one of the following limitations: (1) Prior works ignore the important role of encoder architectures in OOD generalization. (Chen et al. 2022) highlights that it is promising to obtain better OOD generalization ability by incorporating more advanced architectures. As shown in Figure 4 and Figure 5, we confirm that existing methods are not sufficient to deal with this complex problem. (2) Some methods ignore the important role of increasing the diversity of training data, i.e., enlarging the training distribution, to improve generalization performance. (3) Existing methods do not fully consider supervised signals that exist within and between samples in a batch. Overall, the above limitations lead to sub-optimal solutions.
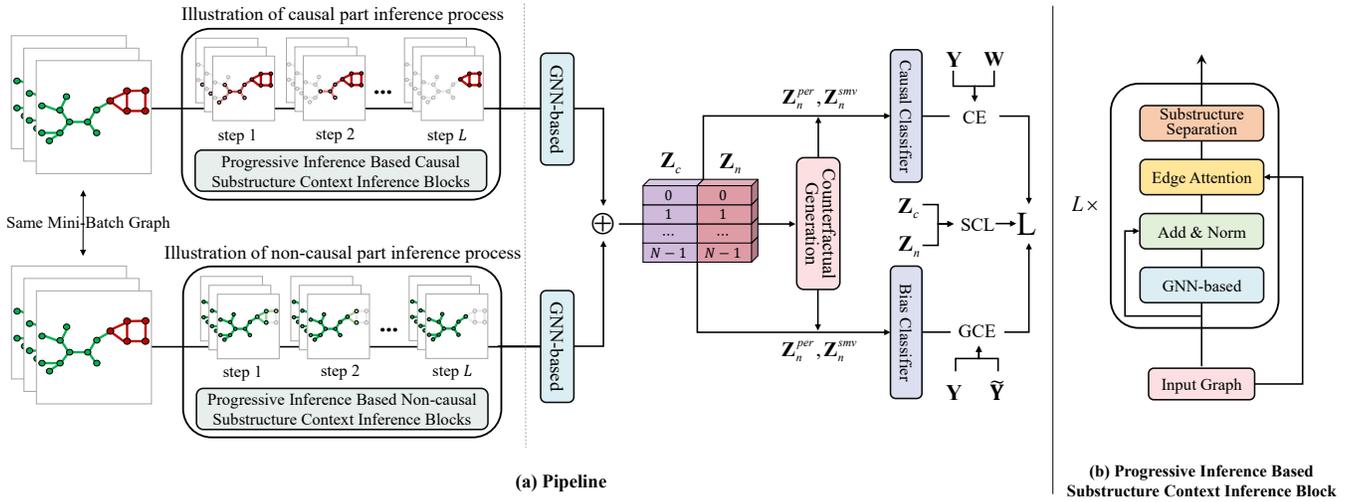
Figure 2: The pipeline and implementation details of the GPro. The basic idea is to decompose the complex problem of causal invariant learning on graphs into multiple intermediate inference steps, and finally extract causal features with generalization through progressive inference. Notably, in the input graph toy example of the leftmost, the red part and the green part are defined as causal and non-causal substructures.

Methodology

The pipeline of GPro is shown in Figure 2. It consists of three major components: progressive inference-based substructure context inference block, counterfactual graph sample generation, and causal learning loss function. First, the substructure context inference block extracts causal and non-causal representations via step-by-step inference. Then, two strategies are designed to generate counterfactual samples to enlarge the training distribution. Finally, the loss function promotes a causal relationship between causal representations and labels, while eliminating any misleading correlations between non-causal representations and labels.

Problem Formulation

Suppose we are given training and testing graph data $\mathcal{G}_{train} = \{(G_i, Y_i)\}_{i=1}^{N^{tr}}$ and $\mathcal{G}_{test} = \{(G_i, Y_i)\}_{i=1}^{N^{te}}$, drawn from distributions $P(\mathcal{G}_{train})$ and $P(\mathcal{G}_{test})$, respectively. $\mathcal{G}_{test}$ is unobserved in the training stage. In the out-of-distribution setting, our goal is to learn a graph predictor f that achieves a satisfactory generalization on a testing set with an unknown distribution:

$$f_\theta = \arg\min_{f_\theta} \mathbb{E}_{G,Y \sim P(\mathcal{G}_{test})}\left[\ell\left(f_\theta\left(G\right), Y\right)\right], \quad (1)$$

where the distribution shift exists in the training set and the unseen testing set, i.e., $P(\mathcal{G}_{train}) \neq P(\mathcal{G}_{test})$, and $\ell(\cdot, \cdot): \mathbb{Y} \times \mathbb{Y} \to \mathbb{R}$ denotes a loss function.

Substructure Context Inference Block

To address the challenges posed by the complex topology of graphs for causal invariant learning, we decompose the complex inference problem of learning causal structures and features into multiple intermediate inference steps. Since the

causal and non-causal parts are complementary, we employ a dual-tower model where one tower is responsible for identifying the non-causal part, while the other tower focuses on recognizing the causal part. These two towers work in tandem and provide mutual assistance to each other in the overall task. The illustration of our proposed GPro and its implementation details are shown in Figure 2.

Specifically, given an input graph $G = \{\mathbf{A}, \mathbf{X}\}$, where $\mathbf{A}$ and $\mathbf{X}$ are the adjacency matrix and node features, respectively, we first employ an edge attention layer to measure the causal importance of edges, and edge-level attention scores are estimated by considering three simple but effective encodings, namely GNN update node feature encoding, node centrality encoding, and inter-node similarity encoding. The node features are updated by a GNN encoder, and employ a residual connection (He et al. 2016) and batch normalization (Ioffe and Szegedy 2015) following the GNN layer:

$$\mathbf{H} = f(\mathbf{A}, \mathbf{X}). \quad (2)$$

Then, unlike previous methods that ignore node centrality, we realize the role of node centrality in measuring the importance of nodes (Ying et al. 2021), and additionally introduce the degree centrality of nodes to comprehensively portray their representations.

$$\mathbf{q}_i = \text{MLP}_{\text{node}}\left([|\mathcal{N}(i)|; \mathbf{h}_i]\right), \quad (3)$$

where $|\mathcal{N}(i)|$ is the degree of node i, $\mathbf{h}_i = \mathbf{H}[i, :]$ is the feature of node i updated by the GNN encoder f, and $[;]$ is the concatenation operation. The inter-node similarity is encoded through $\text{sim}(\cdot, \cdot)$. Finally, the calculation formula of edge-level attention α_{ij} for node i and node j is as follows:

$$\alpha_{ij} = \sigma\left(\text{MLP}_{\text{edge}}([\text{sim}(\mathbf{q}_i, \mathbf{q}_j); \mathbf{q}_i; \mathbf{q}_j])\right), \quad (4)$$

where $\alpha_{ij} \in (0, 1)$ denotes the edge-level attention score of edge (i, j) in the causal substructure. $\sigma(\cdot)$ is the

sigmoid function. Additionally, we define $\text{sim}(\mathbf{q}, \mathbf{k}) = \mathbf{q}^T \mathbf{k} / \|\mathbf{q}\| \|\mathbf{k}\|$.

To separate causal substructures and features from the original graph step-by-step, at each inference step, the substructure separation layer constructs a mask matrix $\mathbf{M}$ to further separate the ρ (e.g., 10%) substructures with the lowest score in the causal attention score matrix $\mathbf{E}$, where $\mathbf{E}[i, j] = \alpha_{ij}$, i.e., those should belong to the non-causal part, from the causal substructures learned from the previous inference step:

$$\mathbf{M} = \text{rank}\left(\mathbf{E}, \lfloor \rho |\mathcal{E}| \rfloor\right), \qquad (5)$$

where $\mathbf{M} = \{0, 1\}^{|\mathcal{V}| \times |\mathcal{V}|}$, $|\mathcal{V}|$ and $|\mathcal{E}|$ are the number of nodes and edges in the graph G, respectively. The rank function sorts the causal attention scores in $\mathbf{E}$, $\mathbf{M}_{ij} = 1/0$ indicates that the edge (i, j) is determined to be the causal/non-causal substructure in the current inference step.

Afterward, we update the adjacency matrix and edge-level attention scores in the next intermediate inference step through the mask matrix $\mathbf{M}$ constructed by the $(l - 1)$-th layer intermediate inference step:

$$\mathbf{A}^l = \mathbf{A}^{l-1} \odot \mathbf{M}, \qquad (6)$$

where $\odot$ denotes the Hadamard product of the matrix.

In the progressive inference process, each intermediate inference step is modeled by a substructure context inference block, involving Eq. (2) to Eq. (6), as illustrated in Figure 2(b). We obtain more reliable causal substructure $G_c = \left\{\mathbf{A}_c^L, \mathbf{H}_c^L\right\}$ and non-causal substructure $G_n = \left\{\mathbf{A}_n^L, \mathbf{H}_n^L\right\}$ through an L-step intermediate inference process, that is, stacking L layers of causal and non-causal substructure context inference blocks that do not share parameters. After deriving the final causal and non-causal substructures, we learn causal and non-causal graph-level representations through GNN encoders and the pooling operation:

$$\mathbf{Z}_c = f_{readout}\left(f_c\left(\mathbf{A}_c^L, \mathbf{H}_c^L\right)\right), \qquad (7)$$

$$\mathbf{Z}_n = f_{readout}\left(f_n\left(\mathbf{A}_n^L, \mathbf{H}_n^L\right)\right), \qquad (8)$$

where $f_{readout}(\cdot)$ is a readout function to generate the graph-level representation. $\mathbf{Z}_c, \mathbf{Z}_n \in \mathbb{R}^{N \times d}$ are causal and non-causal representation matrices in the mini-batch graph, respectively. The batch size is N.

Counterfactual Graph Sample Generation

To this point, we have extracted causal and non-causal representations in graphs through a complex multi-step inference process. To further improve the graph OOD generalization, we employ two strategies to generate counterfactual graph representations to eliminate correlations between causal and non-causal variables, while increasing the diversity of samples and enlarging the training distribution. As causal variables reflect invariant intrinsic properties in graph data, inappropriate interventions on causal representations may lead to changes in the semantics and labels of the input graph. However, there is no causality between non-causal representations and labels. Therefore, we could enlarge the training distribution through robust interventions on the non-causal representation.

The first counterfactual graph representation generation strategy is to randomly permute the non-causal representations. Random permute has proven to be effective in OOD problems in several domains (Lee et al. 2021; Sui et al. 2022a). The permute $(\cdot)$ function randomly permutes the order of the graphs in the mini-batch.

$$\text{idx} = \text{permute}(N), \qquad (9)$$

where idx is the new indices after random permutation. $\mathbf{Z}_n^{per}$ is the randomly permute non-causal representation matrix, i.e., $\mathbf{Z}_n^{per} = \mathbf{Z}_n[\text{idx}, :]$.

Inspired by (Tang et al. 2021), we design a new counterfactual sample generation strategy for graph-level representations. The core of the second strategy is to enlarge the training distribution by swapping the mean and variance between the non-causal representations of the samples in the mini-batch.

$$\mathbf{Z}_n^{smv} = \sigma_{\mathbf{Z}_n^{per}} \frac{\mathbf{Z}_n - \mu_{\mathbf{Z}_n}}{\sigma_{\mathbf{Z}_n}} + \mu_{\mathbf{Z}_n^{per}}, \qquad (10)$$

where $\mu_{\mathbf{Z}_n}, \sigma_{\mathbf{Z}_n}$ are the means and variances of the non-causal representations of each sample in the minibatch, and $\mu_{\mathbf{Z}_n^{per}}, \sigma_{\mathbf{Z}_n^{per}}$ are the means and variances of the non-causal graph representations after random permutation.

Causal Learning Loss Function

It is necessary to design reasonable loss functions to ensure causal relationships between causal features and labels while eliminating spurious correlations between non-causal features and labels. After counterfactual graph sample generation, given a mini-batch of graphs, we can extract three graph-level representations, i.e., a real graph representation $\mathbf{Z} = [\mathbf{Z}_c; \mathbf{Z}_n]$ and two counterfactual graph representations $\mathbf{Z}^{per} = [\mathbf{Z}_c; \mathbf{Z}_n^{per}]$, and $\mathbf{Z}^{smv} = [\mathbf{Z}_c; \mathbf{Z}_n^{smv}]$. Since the causal and non-causal parts are complementary, we employ a dual-tower model to identify the causal and non-causal parts, respectively. Therefore, we firstly design two classifiers, namely causal classifier Φ_c and non-causal classifier Φ_n to train this dual-tower model (note that, the loss from Φ_c is not back-propagated to the encoder model involved in generating non-causal features, and vice versa). The purpose of the causal branch is to estimate causal features, so we classify its representation to the ground-truth label. Thus, we define the supervised classification loss as cross-entropy (CE) loss to train the causal encoder. Meanwhile, we utilize the generalized cross-entropy (GCE) (Zhang and Sabuncu 2018) loss and target labels to train a non-causal encoder and classifier. GCE loss is described as:

$$\text{GCE}\left(\Phi_n(\mathbf{z}), \mathbf{y}\right) = \frac{1 - \Phi_n^y(\mathbf{z})^q}{q}, \qquad (11)$$

where y refers to the ground truth label, $\Phi_n(\mathbf{z})$ and $\Phi_n^y(\mathbf{z})$ indicate the softmax output of the non-classifier Φ_n and its probability belonging to the target class y, respectively. q is a hyperparameter. The GCE loss imposes a higher weight on the gradient of the CE loss for samples, which have high confidence Φ_n^y of the target category y. It is defined as follows:

$$\frac{\partial GCE(\Phi_n(\mathbf{z}), y)}{\partial \theta_n} = (\Phi_n^y)^q \frac{(\Phi_n(\mathbf{z}), y)}{\partial \theta_n}, \qquad (12)$$

where non-causal shortcut information is usually easier to learn and will have larger $(\Phi_n^y)^q$ as confirmed by prior work (Lee et al. 2021; Fan et al. 2022). GCE loss amplifies the gradient by $(\Phi_n^y)^q$ to emphasize the non-causal encoder and classifier Φ_n overfocus on non-causal information. Therefore, we train the causal and non-causal parts with CE and GCE losses, respectively. The mathematical definition of the objective function is as follows:

$$\mathcal{L}_{\text{dis}} = \text{CE}(\Phi_c(\mathbf{Z}), \mathbf{Y}) + \text{GCE}(\Phi_n(\mathbf{Z}), \mathbf{Y}). \qquad (13)$$

In addition, we also train the causal and non-causal encoders by the CE and GCE loss between the counterfactual graph representations $\mathbf{Z}^{per}$, $\mathbf{Z}^{smv}$ and the target labels, respectively. For the causal part, we maintain the consistency between causal features and the target label $\mathbf{Y}$, which is equivalent to expanding the training distribution, thereby better training the causal classifier. To make the spurious correlation between counterfactual graph representations and labels still exist, we permute the label $\widetilde{\mathbf{Y}} = \mathbf{Y}[\text{idx}]$ along with $\mathbf{Z}^{per}$ and $\mathbf{Z}^{smv}$ as the target labels for the output of Φ_n. This ensures that the non-causal encoder and classifier continuously focus on the non-causal information. Meanwhile, samples can be regarded as unbiased and high quality when the loss of the causal classifier is small, but the loss of the non-causal classifier is large. Inspired by (Lee et al. 2021), we enforce the causal encoder and classifier to learn causality by increasing the weights of counterfactual samples of unbiased samples by $\mathbf{W}(\mathbf{Z}) = \frac{\text{CE}(\Phi_n(\mathbf{Z}), \mathbf{Y})}{\text{CE}(\Phi_c(\mathbf{Z}), \mathbf{Y}) + \text{CE}(\Phi_n(\mathbf{Z}), \mathbf{Y})}$. Moreover, $\mathcal{L}_{\text{cou}}$ is not used during the initial training phase because the generated counterfactual graph representations are of low quality and may lead to label changes. $\mathcal{L}_{\text{cou}}$ is formally defined as follows:

$$\mathcal{L}_{\text{cou}} = \mathbf{W}(\mathbf{Z})(\text{CE}(\Phi_c(\mathbf{Z}^{per}), \mathbf{Y}) + \text{CE}(\Phi_c(\mathbf{Z}^{smv}), \mathbf{Y}))/2$$
$$+ \left(\text{GCE}\left(\Phi_n(\mathbf{Z}^{per}), \widetilde{\mathbf{Y}}\right) + \text{GCE}\left(\Phi_n(\mathbf{Z}^{smv}), \widetilde{\mathbf{Y}}\right)\right)/2. \qquad (14)$$

To enhance the disentanglement between causal and non-causal representations, a novel loss function is proposed in this work, which extends supervised contrastive learning (Khosla et al. 2020) (SCL) into the graph causal invariant learning. Specifically, by leveraging the label information, the proposed method pulls together causal graph representations that belong to the same class in a batch, while pushing apart causal graph representations from different classes and non-causal graph representations from all classes. The novel supervised contrastive loss of graph causal invariance principle is defined as follows:

$$\mathcal{L}_{\text{scl}} = \sum_{i \in I} \frac{-1}{|P(i)|} \log \frac{\sum_{p \in P(i)} \exp\left(\mathbf{z}_i^c \cdot \mathbf{z}_p^c / \tau\right)}{\sum_{j \in A(i)} \exp\left(\mathbf{z}_i^c \cdot \mathbf{z}_j^c / \tau\right) + \sum_{k \in I} \exp\left(\mathbf{z}_i^c \cdot \mathbf{z}_k^n / \tau\right)}, \qquad (15)$$

where $i \in I \equiv \{1...N\}$ is the index in the mini-batch, and $A(i) \equiv I \setminus \{i\}$. $P(i) \equiv \{p \in A(i) : \boldsymbol{y}_p = \boldsymbol{y}_i\}$ is the set of indices that have the same label as graph i, and τ is a

temperature parameter. $\mathbf{z}_i^c$ and $\mathbf{z}_k^n$ are the causal and the non-causal representation of graph i and k, respectively.

Note that the causal and non-causal substructure context inference blocks have the same architecture but do not share weights, we expect both encoders to make similar judgments on edge-level attention scores. We impose a consistency constraint on the context inference blocks of causal and non-causal substructures via mean squared error (MSE) loss.

$$\mathcal{L}_{\text{con}} = \text{MSE}(\mathbf{E}_c, \mathbf{E}_n), \qquad (16)$$

where $\mathbf{E}_c$ and $\mathbf{E}_n$ are the learned attention score matrices for the causal and non-causal substructure context downsampling blocks, respectively.

Finally, combining all the above defined losss functions, the total causal learning loss function is defined as:

$$\mathcal{L} = \mathcal{L}_{\text{dis}} + \lambda_1 \mathcal{L}_{\text{cou}} + \lambda_2 \mathcal{L}_{\text{scl}} + \lambda_3 \mathcal{L}_{\text{con}}, \qquad (17)$$

where λ_1, λ_2, and λ_3 are hyperparameters for weighing the importance of counterfactual loss, supervised contrastive loss, and consistency loss, respectively. The details of our algorithm are summarized in the Appendix.

Experiments

Experiment Preparation

Datasets We use three benchmark graph classification datasets in causal learning (Fan et al. 2022), namely CMNIST-75sp, CFashion-75sp, and CKuzushiji-75s, to evaluate the performance of the models on out-of-distribution (OOD) problems. The datasets consider three bias degrees 0.8, 0.9, 0.95, i.e., the causal and the non-causal substructures have 80%, 90%, and 95% probabilities of co-occurrence in the training set. For example, at a bias degree of 0.9 in the training set of CMNIST-75sp superpixel graph, 90% of the 0 digits come with a red background (i.e., biased samples), and the remaining 10% come with a random background color (i.e., unbiased samples). Thus, it enables the establishment of spurious correlations between the non-causal substructures and the labels. The datasets are divided into the training set: validation set: testing set in the ratio of 10K:5K:10K. The testing sets are all unbiased samples. Each dataset contains 10 classes. Statistics of the datasets are provided in the Appendix.

Baselines To verify that GPro produces consistent and significant improvements, we compare GPro with 11 state-of-the-art algorithms designed for in-distribution (ID) or out-of-distribution (OOD) learning. **In-Distribution Methods:** GCN (Kipf and Welling 2017), GIN (Xu et al. 2019), GC-NII (Chen et al. 2020), FactorGCN (Yang et al. 2020), and DiffPool (Ying et al. 2018). **Out-of-Distribution Methods:** LDD (Lee et al. 2021), StableGNN (Fan et al. 2023), CAL (Sui et al. 2022a), DisC (Fan et al. 2022), CIGA (Chen et al. 2022) and GALA (Chen et al. 2023). More details on the baselines can be found in the Appendix.

Implementation Details We use the Adam optimizer (Kingma and Ba 2014), and the learning rate is 0.01. For Eq. (7) and Eq. (8), we use the GCN (Kipf and Welling 2017) with 2 layers and 146 hidden dimensions as the

Dataset	CMNIST-75sp			CFashion-75sp			CKuzushiji-75sp		
Bias	0.8	0.9	0.95	0.8	0.9	0.95	0.8	0.9	0.95
GCN (Kipf and Welling 2017)	$50.43_{\pm4.13}$	$28.97_{\pm4.40}$	$13.50_{\pm1.38}$	$63.60_{\pm0.53}$	$57.22_{\pm0.93}$	$47.69_{\pm0.42}$	$38.45_{\pm1.1}$	$28.35_{\pm0.79}$	$20.70_{\pm0.88}$
GIN (Xu et al. 2019)	$57.75_{\pm0.78}$	$36.78_{\pm5.55}$	$16.04_{\pm1.14}$	$64.25_{\pm0.46}$	$58.03_{\pm0.40}$	$49.74_{\pm0.60}$	$41.83_{\pm0.78}$	$30.09_{\pm0.87}$	$21.18_{\pm1.63}$
GCNII (Chen et al. 2020)	$69.70_{\pm1.73}$	$57.68_{\pm1.68}$	$41.00_{\pm3.75}$	$66.68_{\pm0.59}$	$60.58_{\pm0.28}$	$53.18_{\pm0.08}$	$48.53_{\pm0.25}$	$36.23_{\pm0.20}$	$25.60_{\pm0.76}$
FactorGCN (Yang et al. 2020)	$72.30_{\pm1.18}$	$62.35_{\pm5.07}$	$42.50_{\pm4.91}$	$61.23_{\pm1.11}$	$53.50_{\pm1.29}$	$45.78_{\pm2.40}$	$42.87_{\pm1.19}$	$32.35_{\pm2.79}$	$23.87_{\pm0.12}$
DiffPool (Ying et al. 2018)	$73.79_{\pm0.02}$	$66.45_{\pm0.78}$	$47.12_{\pm1.04}$	$62.82_{\pm0.53}$	$57.50_{\pm0.39}$	$50.86_{\pm0.20}$	$45.46_{\pm0.65}$	$36.18_{\pm0.19}$	$27.45_{\pm0.26}$
StableGNN (Fan et al. 2023)	$77.65_{\pm1.64}$	$68.87_{\pm1.74}$	$51.33_{\pm0.87}$	$64.03_{\pm0.29}$	$58.26_{\pm0.09}$	$51.46_{\pm0.39}$	$49.41_{\pm0.09}$	$39.30_{\pm0.12}$	$28.26_{\pm0.14}$
LDD$_{GCN}$ (Lee et al. 2021)	$64.95_{\pm1.22}$	$56.65_{\pm2.18}$	$46.83_{\pm2.88}$	$63.85_{\pm1.17}$	$64.30_{\pm0.89}$	$62.28_{\pm0.48}$	$42.38_{\pm0.33}$	$38.75_{\pm0.49}$	$33.08_{\pm0.59}$
LDD$_{GIN}$ (Lee et al. 2021)	$64.88_{\pm1.45}$	$50.59_{\pm1.07}$	$31.23_{\pm2.48}$	$64.65_{\pm0.63}$	$57.10_{\pm0.43}$	$53.38_{\pm0.47}$	$37.83_{\pm0.54}$	$28.97_{\pm0.18}$	$22.13_{\pm0.34}$
LDD$_{GCNII}$ (Lee et al. 2021)	$78.03_{\pm0.66}$	$69.53_{\pm0.96}$	$51.05_{\pm3.87}$	$50.63_{\pm1.79}$	$54.09_{\pm2.54}$	$57.93_{\pm0.88}$	$48.70_{\pm1.98}$	$41.59_{\pm1.07}$	$33.93_{\pm0.71}$
CAL$_{GCN}$ (Sui et al. 2022a)	$77.10_{\pm1.01}$	$67.89_{\pm0.45}$	$51.42_{\pm1.39}$	$67.74_{\pm0.31}$	$60.90_{\pm0.71}$	$54.41_{\pm0.15}$	$52.18_{\pm0.32}$	$41.47_{\pm0.69}$	$31.39_{\pm0.65}$
CAL$_{GIN}$ (Sui et al. 2022a)	$76.50_{\pm0.40}$	$65.32_{\pm0.32}$	$44.43_{\pm1.28}$	$65.04_{\pm0.23}$	$59.82_{\pm0.39}$	$52.98_{\pm0.51}$	$50.71_{\pm0.41}$	$38.40_{\pm0.53}$	$29.46_{\pm0.49}$
CAL$_{GAT}$ (Sui et al. 2022a)	$88.21_{\pm0.50}$	$81.57_{\pm0.21}$	$69.18_{\pm1.10}$	$71.11_{\pm0.06}$	$66.22_{\pm0.36}$	$59.02_{\pm0.39}$	$64.54_{\pm0.16}$	$52.00_{\pm0.70}$	$37.93_{\pm0.81}$
DisC$_{GCN}$ (Fan et al. 2022)	$82.60_{\pm0.93}$	$78.14_{\pm2.14}$	$63.47_{\pm5.65}$	$66.85_{\pm1.11}$	$65.33_{\pm4.70}$	$63.93_{\pm1.50}$	$55.53_{\pm2.29}$	$48.13_{\pm2.59}$	$36.63_{\pm1.73}$
DisC$_{GIN}$ (Fan et al. 2022)	$82.10_{\pm1.50}$	$74.90_{\pm1.81}$	$58.58_{\pm4.24}$	$67.10_{\pm1.07}$	$59.90_{\pm1.31}$	$55.80_{\pm0.36}$	$55.18_{\pm1.00}$	$41.75_{\pm0.81}$	$30.25_{\pm1.63}$
DisC$_{GCNII}$ (Fan et al. 2022)	$79.50_{\pm2.48}$	$76.00_{\pm1.90}$	$60.54_{\pm5.33}$	$66.47_{\pm1.77}$	$65.48_{\pm0.70}$	$61.75_{\pm0.27}$	$54.90_{\pm1.30}$	$44.73_{\pm1.55}$	$36.95_{\pm0.70}$
CIGA (Chen et al. 2022)	$64.45_{\pm3.49}$	$48.56_{\pm6.44}$	$34.33_{\pm2.63}$	$59.37_{\pm0.89}$	$53.52_{\pm1.98}$	$45.37_{\pm2.15}$	$43.80_{\pm2.46}$	$31.74_{\pm2.18}$	$22.89_{\pm0.90}$
GALA (Chen et al. 2023)	$78.82_{\pm1.66}$	$64.73_{\pm2.39}$	$41.54_{\pm3.25}$	$65.64_{\pm0.49}$	$59.68_{\pm1.47}$	$51.72_{\pm1.36}$	$50.41_{\pm1.70}$	$33.69_{\pm2.76}$	$24.16_{\pm0.60}$
GPro	$\mathbf{88.87}_{\pm1.03}$	$\mathbf{87.58}_{\pm0.36}$	$\mathbf{79.34}_{\pm1.07}$	$\mathbf{75.41}_{\pm0.36}$	$\mathbf{70.57}_{\pm0.29}$	$\mathbf{64.72}_{\pm0.71}$	$\mathbf{66.46}_{\pm0.56}$	$\mathbf{58.35}_{\pm0.63}$	$\mathbf{47.56}_{\pm0.40}$

Table 1: Experimental results (%) for the graph classification task on three datasets with unbiased testing sets. We report the mean accuracy and standard error. Bold indicates the optimal and underline indicates the suboptimal.

encoder. We train the GPro with 200 epochs and add $\mathcal{L}_{\text{cou}}$ loss function at the 100th epoch. The batch size is 256. The default value for the number of causal and non-causal substructure context inference blocks is 2, and ρ are 0.9 and 0.8, respectively. We set q of GCE loss as 0.7 to amplify the focus on the non-causal part, λ_1 is 15, λ_2 is 0.01 and λ_3 is 1.

Comparison with State-of-the-Art

To comprehensively verify the effectiveness of GPro, we compared 11 state-of-the-art algorithms and their variants. Table 1 shows the experimental results (%) for the graph classification task in the three datasets. We report the mean accuracy and standard error. Bold indicates optimal and underline denotes suboptimal. On the basis of the experimental results, we can observe that GPro is optimal in 9 different dataset divisions. Specifically, the baselines developed based on ID are more likely to learn shortcut features from spurious correlations between non-causal parts and labels, resulting in performance that is typically inferior to OOD baselines. Compared to optimal ID-based baseline methods, GPro improves 22.81%, 10.05%, and 20.05% on average in three datasets, respectively. When spurious correlations are more severe in the training set, that is, the bias is larger, the performance of the baseline developed based on ID degrades severely. GPro improves 13.91%, 17.75%, and 21.29% on average over the ID-based design approach when the bias degree of the datasets is 0.8, 0.9, and 0.95, demonstrating that GPro has better debiasing causal learning ability. Algorithms designed for OOD often achieve better performance. Compared with state-of-the-art methods specially designed for OOD, our proposed model outperforms 4.91% on average. In the case of datasets with more severe distribution shifts, the performance improvement could reach 6. 86%. This further supports the observation in Figure 4 that existing methods are limited in disentangling the complex cou-
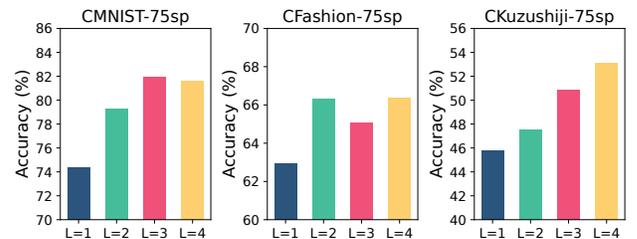

Figure 3: Quantitative sensitivity analysis of GPro for the number of progressive inference steps.

pled associations between causal and non-causal substructures in graphs, resulting in the failure to extract ground truth causal features.

In summary, the experimental results demonstrate that GPro obtains state-of-the-art OOD generalization capability through a well-designed progressive inference process, counterfactual sample generation, and causal loss functions.

Effectiveness of Progressive inference

This subsection evaluates progressive inference through quantitative and qualitative analyses.

Quantitative Evaluation We quantitatively evaluated our model by comparing the accuracy (%) across three challenging datasets: CMNIST-75sp-0.95, CFashion-75sp-0.95, and CKuzushiji-75sp-0.95. We assess the performance at 1, 2, 3, and 4 progressive inference steps, facilitated by stacking substructure context sampling blocks, with each block representing one step. Initial results, with a single inference step ($L = 1$), show a 4.02% improvement over the leading model, confirming the efficacy of GPro components. Performance typically improves as the number of inference

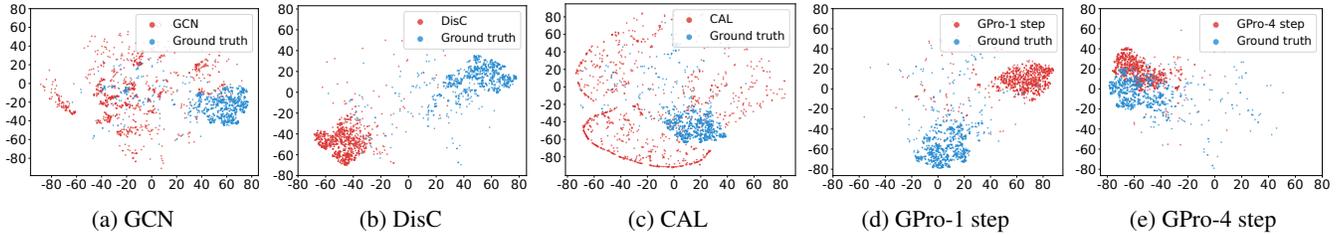(a) GCN   (b) DisC   (c) CAL   (d) GPro-1 step   (e) GPro-4 step

Figure 4: TSNE visualization of sample features of class 0 generated by the model in the CMNIST-75sp dataset. There is generally a significant distribution gap between the features learned by existing methods (such as GCN, DisC and CAL) and the ground-truth causal features. GPro learns causal features that are closer to the ground-truth via progressive inference.

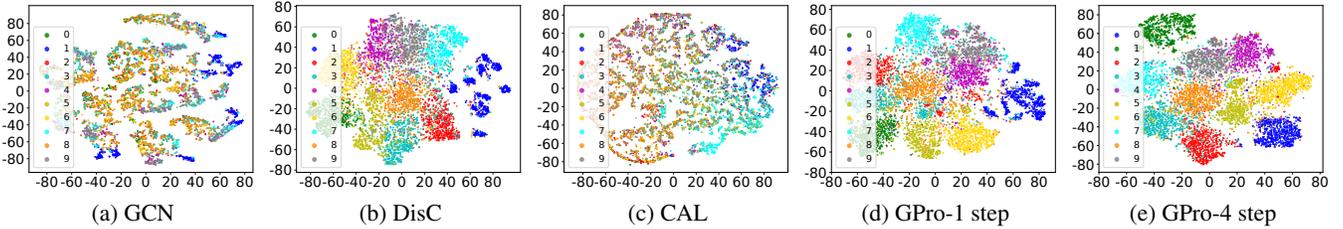(a) GCN   (b) DisC   (c) CAL   (d) GPro-1 step   (e) GPro-4 step

Figure 5: TSNE visualization of the features learned by GCN, DisC, CAL, and GPro in the CMNIST-75sp dataset, where labels are marked by colors. The features learned through GPro show that the clusters within each category exhibit compactness while the distance between clusters is maximized.

layers increases. These findings suggest that more complex datasets require additional inference steps to achieve optimal performance, while simpler datasets are well served by 2 to 3 steps. This experiment illustrates the importance of a multi-step approach in graph causality analysis.

Qualitative Visualization Evaluation We qualitatively evaluate the benefits of progressive inference in GPro using t-SNE visualization, as shown in Figure 4. The visualization highlights significant gaps between the causal features learned by ID methods like GCN and OOD methods such as DisC and CAL, compared to ground-truth causal features, which leads to the predictions made by the existing methods being still unreliable. GPro employs progressive inference to help bridge these gaps, with longer inference processes (4-step) yielding superior results compared to shorter inference processes (1-step). Moreover, we visualize the representations of all samples in the test set of CMNIST-75sp dataset learned by the above models. Figure 5a shows that each cluster mixes multiple classes, indicating GCN tends to learn shortcut features (non-causal features) from spurious correlations between non-causal parts and labels, and fails to capture generalized causal features. Single-step methods like DisC, CAL, and GPro-1step inadequately distinguish class features, resulting in blurred cluster boundaries. Conversely, employing longer inference steps results in tighter intra-class clusters and more distinct inter-class distances, showcasing the exceptional capability of progressive inference to capture causal features effectively.

Ablation Studies

To validate the validity of each component in GPro, we conduct ablation studies on CMNIST-75sp, CFashion-75sp, and

Method	CMNIST-75sp	CFashion-75sp	CKuzushiji-75sp
GPro	$87.58_{\pm 0.36}$	$70.57_{\pm 0.29}$	$58.35_{\pm 0.63}$
w/o $\mathbf{Z}_n^{per}$	$86.84_{\pm 0.52}\ (\downarrow_{0.74})$	$69.35_{\pm 0.86}\ (\downarrow_{1.22})$	$57.63_{\pm 0.58}\ (\downarrow_{0.72})$
w/o $\mathbf{Z}_n^{smv}$	$86.71_{\pm 0.41}\ (\downarrow_{0.87})$	$68.96_{\pm 0.38}\ (\downarrow_{1.61})$	$56.73_{\pm 0.58}\ (\downarrow_{1.62})$
w/o $\mathcal{L}_{cou}$	$82.99_{\pm 0.65}\ (\downarrow_{4.59})$	$63.82_{\pm 0.40}\ (\downarrow_{6.75})$	$48.25_{\pm 0.77}\ (\downarrow_{10.10})$
w/o $\mathcal{L}_{scl}$	$86.25_{\pm 1.61}\ (\downarrow_{1.33})$	$69.03_{\pm 0.32}\ (\downarrow_{1.54})$	$57.73_{\pm 0.64}\ (\downarrow_{0.62})$
w/o $\mathcal{L}_{con}$	$86.67_{\pm 1.05}\ (\downarrow_{0.91})$	$70.13_{\pm 0.33}\ (\downarrow_{0.44})$	$57.57_{\pm 0.31}\ (\downarrow_{0.78})$

Table 2: Ablation study on different variants.

CKuzushiji-75sp with all bias degrees of 0.9. Specifically, w/o $\mathbf{Z}_n^{per}$ and w/o $\mathbf{Z}_n^{smv}$ are designed to remove the counterfactual generation strategy of randomly permuting the non-causal representations and swapping the mean and variance between the non-causal representations, respectively. W/o $\mathcal{L}_{cou}$, w/o $\mathcal{L}_{scl}$ and w/o $\mathcal{L}_{con}$ are the GPro variant models for removing $\mathcal{L}_{cou}$, $\mathcal{L}_{scl}$, and $\mathcal{L}_{con}$ from the loss function Eq. (17), respectively. As shown in Table 2, we have the following observations: among the two counterfactual generation strategies, w/o $\mathbf{Z}_n^{per}$ performance decreases by 0.89%, and w/o $\mathbf{Z}_n^{smv}$ performance decreases by 1.37% on average in the three datasets. The effectiveness of two counterfactual generation strategies is demonstrated, while w/o $\mathbf{Z}_n^{smv}$ brings a more significant performance improvement. In addition, w/o $\mathcal{L}_{cou}$, w/o $\mathcal{L}_{scl}$, and w/o $\mathcal{L}_{con}$ show 7.15%, 1.16% and 0.71% performance degradation on the three datasets, respectively. Removing $\mathcal{L}_{cou}$ significantly reduces performance across all datasets. Overall, omitting any component in GPro leads to performance degradation, underscoring the importance of each component.

Conclusion

In this paper, we propose a novel approach to graph causal invariant learning via progressive inference perspective, called GPro. Specifically, we decompose the problem of identifying causal invariant parts of graphs into multiple intermediate inference steps, and extract causal features that are stable to distribution shifts through step-by-step inference. To make the progressive inference process better capture the causal invariant parts, we propose a novel feature augmentation method to generate counterfactual samples to enlarge the training distribution. Moreover, we propose a new supervised contrastive learning method to fully utilize supervised signals. We conduct comprehensive experiments on three datasets. Compared with the state-of-the-art method, our proposed model outperforms 4.91% on average. In the case of datasets with more severe distribution shifts, the performance improvement could be up to 6.86%. The experimental results demonstrate that our proposed method is superior to the state-of-the-art methods.

Acknowledgments

References

Bengio, Y.; Deleu, T.; Rahaman, N.; Ke, N. R.; Lachapelle, S.; Bilaniuk, O.; Goyal, A.; and Pal, C. 2019. A Meta-Transfer Objective for Learning to Disentangle Causal Mechanisms. In *International Conference on Learning Representations*.

Chen, M.; Wei, Z.; Huang, Z.; Ding, B.; and Li, Y. 2020. Simple and deep graph convolutional networks. In *International Conference on Machine Learning*, 1725–1735. PMLR.

Chen, Y.; Bian, Y.; Zhou, K.; Xie, B.; Han, B.; and Cheng, J. 2023. Does Invariant Graph Learning via Environment Augmentation Learn Invariance? In *Thirty-seventh Conference on Neural Information Processing Systems*.

Chen, Y.; Zhang, Y.; Bian, Y.; Yang, H.; Kaili, M.; Xie, B.; Liu, T.; Han, B.; and Cheng, J. 2022. Learning causally invariant representations for out-of-distribution generalization on graphs. *Advances in Neural Information Processing Systems*, 35: 22131–22148.

Ding, M.; Kong, K.; Chen, J.; Kirchenbauer, J.; Goldblum, M.; Wipf, D.; Huang, F.; and Goldstein, T. 2021. A Closer Look at Distribution Shifts and Out-of-Distribution Generalization on Graphs. In *NeurIPS 2021 Workshop on Distribution Shifts: Connecting Methods and Applications*.

Fan, S.; Wang, X.; Mo, Y.; Shi, C.; and Tang, J. 2022. Debiasing Graph Neural Networks via Learning Disentangled Causal Substructure. *NeurIPS*.

Fan, S.; Wang, X.; Shi, C.; Cui, P.; and Wang, B. 2023. Generalizing graph neural networks on out-of-distribution graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Fu, X.; Chen, C.; Dong, Y.; Vullikanti, A.; Klein, E.; Madden, G.; and Li, J. 2023. Spatial-Temporal Networks for Antibiogram Pattern Prediction. In *2023 IEEE 11th International Conference on Healthcare Informatics (ICHI)*, 225–234. IEEE.

Fu, X.; Chen, Z.; Zhang, B.; Chen, C.; and Li, J. 2024. Federated graph learning with structure proxy alignment. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 827–838.

Fu, X.; Zhang, B.; Dong, Y.; Chen, C.; and Li, J. 2022. Federated graph machine learning: A survey of concepts, techniques, and applications. *ACM SIGKDD Explorations Newsletter*, 24(2): 32–47.

Gao, J.; Sun, C.; Zhao, H.; Shen, Y.; Anguelov, D.; Li, C.; and Schmid, C. 2020. Vectornet: Encoding hd maps and agent dynamics from vectorized representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11525–11533.

Gui, S.; Li, X.; Wang, L.; and Ji, S. 2022. Good: A graph out-of-distribution benchmark. *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hsieh, K.; Wang, Y.; Chen, L.; Zhao, Z.; Savitz, S.; Jiang, X.; Tang, J.; and Kim, Y. 2021. Drug repurposing for COVID-19 using graph neural network and harmonizing multiple evidence. *Scientific reports*, 11(1): 1–13.

Ioffe, S.; and Szegedy, C. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, 448–456. PMLR.

Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; and Krishnan, D. 2020. Supervised contrastive learning. *Advances in neural information processing systems*, 33: 18661–18673.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kipf, T. N.; and Welling, M. 2017. Semi-supervised classification with graph convolutional networks. *ICLR*.

Lee, J.; Kim, E.; Lee, J.; Lee, J.; and Choo, J. 2021. Learning debiased representation via disentangled feature augmentation. *Advances in Neural Information Processing Systems*, 34: 25123–25133.

Li, H.; Wang, X.; Zhang, Z.; and Zhu, W. 2022a. Ood-gnn: Out-of-distribution generalized graph neural network. *IEEE Transactions on Knowledge and Data Engineering*.

Li, H.; Wang, X.; Zhang, Z.; and Zhu, W. 2022b. Out-of-distribution generalization on graphs: A survey. *arXiv preprint arXiv:2202.07987*.

Li, H.; Zhang, Z.; Wang, X.; and Zhu, W. 2022c. Learning invariant graph representations for out-of-distribution generalization. In *Advances in Neural Information Processing Systems*.

Li, X.; Gui, S.; Luo, Y.; and Ji, S. 2023. Graph structure and feature extrapolation for out-of-distribution generalization. *arXiv preprint arXiv:2306.08076*.

Liu, Y.; Ao, X.; Feng, F.; Ma, Y.; Li, K.; Chua, T.-S.; and He, Q. 2023. 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*, 1548–1558.

Liu, Y.; Ao, X.; Qin, Z.; Chi, J.; Feng, J.; Yang, H.; and He, Q. 2021. Pick and choose: a GNN-based imbalanced learning approach for fraud detection. In *Proceedings of the Web Conference 2021*, 3168–3177.

Niu, X.; Li, B.; Li, C.; Xiao, R.; Sun, H.; Deng, H.; and Chen, Z. 2020. A dual heterogeneous graph attention network to improve long-tail performance for shop search in e-commerce. In *SIGKDD*, 3405–3415.

Qiu, J.; Tang, J.; Ma, H.; Dong, Y.; Wang, K.; and Tang, J. 2018. Deepinf: Social influence prediction with deep learning. In *SIGKDD*, 2110–2119.

Seo, C.; Jeong, K.-J.; Lim, S.; and Shin, W.-Y. 2022. Siren: Sign-aware recommendation using graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*.

Shi, B.; Dong, B.; Xu, Y.; Wang, J.; Wang, Y.; and Zheng, Q. 2023. An edge feature aware heterogeneous graph neural network model to support tax evasion detection. *Expert Systems with Applications*, 213: 118903.

Su, X.; You, Z.-H.; Huang, D.-s.; Wang, L.; Wong, L.; Ji, B.; and Zhao, B. 2022. Biomedical knowledge graph embedding with capsule network for multi-label drug-drug interaction prediction. *IEEE Transactions on Knowledge and Data Engineering*.

Sui, Y.; Wang, X.; Wu, J.; Lin, M.; He, X.; and Chua, T.-S. 2022a. Causal attention for interpretable and generalizable graph classification. In *SIGKDD*, 1696–1705.

Sui, Y.; Wang, X.; Wu, J.; Zhang, A.; and He, X. 2022b. Adversarial Causal Augmentation for Graph Covariate Shift. *arXiv preprint arXiv:2211.02843*.

Tang, Z.; Gao, Y.; Zhu, Y.; Zhang, Z.; Li, M.; and Metaxas, D. N. 2021. Crossnorm and selfnorm for generalization under distribution shifts. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 52–61.

Wang, X.; He, X.; Cao, Y.; Liu, M.; and Chua, T.-S. 2019. Kgat: Knowledge graph attention network for recommendation. In *SIGKDD*, 950–958.

Wang, Y.; Wang, J.; Cao, Z.; and Barati Farimani, A. 2022. Molecular contrastive learning of representations via graph neural networks. *Nature Machine Intelligence*, 4(3): 279–287.

Wu, Y.-X.; Wang, X.; Zhang, A.; He, X.; and Chua, T.-S. 2022. Discovering invariant rationales for graph neural networks. *International Conference on Learning Representations*.

Xia, L.; Huang, C.; Xu, Y.; Dai, P.; and Bo, L. 2022. Multi-behavior graph neural networks for recommender system. *IEEE Transactions on Neural Networks and Learning Systems*.

Xu, K.; Hu, W.; Leskovec, J.; and Jegelka, S. 2019. How powerful are graph neural networks? *ICLR*.

Xu, Y.; Wang, L.; Wang, Y.; and Fu, Y. 2022. Adaptive Trajectory Prediction via Transferable GNN. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6520–6531.

Xue, J.; Jiang, N.; Liang, S.; Pang, Q.; Yabe, T.; Ukkusuri, S. V.; and Ma, J. 2022. Quantifying the spatial homogeneity of urban road networks via graph neural networks. *Nature Machine Intelligence*, 4(3): 246–257.

Yan, M.; Cheng, Z.; Gao, C.; Sun, J.; Liu, F.; Sun, F.; and Li, H. 2023. Cascading residual graph convolutional network for multi-behavior recommendation. *ACM Transactions on Information Systems*, 42(1): 1–26.

Yang, Y.; Feng, Z.; Song, M.; and Wang, X. 2020. Factorizable graph convolutional networks. *Advances in Neural Information Processing Systems*, 33: 20286–20296.

Ying, C.; Cai, T.; Luo, S.; Zheng, S.; Ke, G.; He, D.; Shen, Y.; and Liu, T.-Y. 2021. Do transformers really perform badly for graph representation? *Advances in Neural Information Processing Systems*, 34: 28877–28888.

Ying, Z.; You, J.; Morris, C.; Ren, X.; Hamilton, W.; and Leskovec, J. 2018. Hierarchical graph representation learning with differentiable pooling. *Advances in neural information processing systems*, 31.

Zhang, G.; Li, Z.; Huang, J.; Wu, J.; Zhou, C.; Yang, J.; and Gao, J. 2022. efraudcom: An e-commerce fraud detection system via competitive graph neural networks. *ACM Transactions on Information Systems (TOIS)*, 40(3): 1–29.

Zhang, Z.; and Sabuncu, M. 2018. Generalized cross entropy loss for training deep neural networks with noisy labels. *Advances in neural information processing systems*, 31.

Zheng, Q.; Xu, Y.; Liu, H.; Shi, B.; Wang, J.; and Dong, B. 2023. A Survey of Tax Risk Detection Using Data Mining Techniques. *Engineering*.

Zhu, J.; Wang, J.; Han, W.; and Xu, D. 2022. Neural relational inference to learn long-range allosteric interactions in proteins from molecular dynamics simulations. *Nature communications*, 13(1): 1–16.