Disentangling Invariant Subgraph via Variance Contrastive Estimation under Distribution Shifts

Haoyang Li [1] **Xin Wang** [1] **Xueling Zhu** [2] **Weigao Wen** [3] **Wenwu Zhu** [1]

Abstract

Graph neural networks (GNNs) have achieved remarkable success, yet most are developed under the in-distribution assumption and fail to generalize to out-of-distribution (OOD) environments. To tackle this problem, some graph invariant learning methods aim to learn invariant subgraph against distribution shifts, which heavily rely on predefined or automatically generated environment labels. However, directly annotating or estimating such environment labels from biased graph data is typically impractical or inaccurate for real-world graphs. Consequently, GNNs may become biased toward variant patterns, resulting in poor OOD generalization. In this paper, we propose to learn disentangled invariant subgraph via self-supervised contrastive variant subgraph estimation for achieving satisfactory OOD generalization. Specifically, we first propose a GNN-based invariant subgraph generator to disentangle the invariant and variant subgraphs. Then, we estimate the degree of the spurious correlations by conducting self-supervised contrastive learning on variant subgraphs. Thanks to the accurate identification and estimation of the variant subgraphs, we can capture invariant subgraphs effectively and further eliminate spurious correlations by inverse propensity score reweighting. We provide theoretical analyses to show that our model can disentangle the ground-truth invariant and variant subgraphs for OOD generalization. Extensive experiments demonstrate the superiority of our model over state-of-the-art baselines.

[1]Department of Computer Science and Technology, BNRist, Tsinghua University, Beijing, China [2]Department of Radiology, Xiangya Hospital, Central South University, Changsha, Hunan, China [3]Alibaba Group. Correspondence to: Xin Wang <xin_wang@tsinghua.edu.cn>, Wenwu Zhu <wwzhu@tsinghua.edu.cn>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

1. Introduction

Graph is ubiquitous in our daily life, which has been widely used to represent the complex relationships between entities in many fields, including social network (Qiu et al., 2018; Li et al., 2019), knowledge representation (Wang et al., 2017), recommendation systems (Wu et al., 2022b; Li et al., 2021a), multimedia (Hudson & Manning, 2019), computer vision (Zellers et al., 2018), natural language processing (Marcheggiani & Titov, 2017), etc. Among these popular applications, graph-level prediction tasks constitute a major branch. Generally speaking, the label for a graph depends on the information in its critical subgraph (i.e., the part that has invariant and truly predictive relations to the label in the task), rather than the whole graph (Luo et al., 2020; Yu et al., 2021; Fan et al., 2022). For example, the solubility of one molecule can be determined by the predictive functional group, rather than the molecular scaffold (Duvenaud et al., 2015; Hu et al., 2020). The label of one graph from the MNIST superpixel dataset (Dwivedi et al., 2023; Fan et al., 2022) has deterministic and invariant relations with the digit subgraph, rather than the background subgraph (Figure 1). Several representative works (Ying et al., 2018; Luo et al., 2020) are developed to explore capturing such critical subgraphs for learning graph representations effectively, which can largely improve graph-level prediction performance.

Despite their remarkable progress, the existing approaches are generally built upon the in-distribution (I.D.) hypothesis, namely the testing and training graphs are sampled from an identical distribution. Yet the graph data generation mechanism in real-world scenarios is uncontrollable and unobservable (Bengio et al., 2019), so that the distribution shifts can widely exist, making the *out-of-distribution* (OOD) generalization become one of the most crucial issues to be handled (Li et al., 2022b). The existing subgraph-based graph methods fail to capture critical subgraphs due to lacking OOD generalization ability, whose performance can degenerate significantly under distribution shifts. Although several pioneering works (Wu et al., 2022c;a; Li et al., 2022d; Yang et al., 2022; Li et al., 2023b) have been proposed to handle graph distribution shifts via learning environment invariant subgraphs (which are defined as critical

subgraphs that have invariant relations to the labels in any environment) against distribution shifts, they heavily rely on the predefined or automatically generated environment labels, i.e., multiple training environments. However, the environment labels are unavailable in most scenarios and directly annotating or generating environment labels is also impractical or inaccurate, especially in the graph dataset with severe bias (Fan et al., 2022; Qi et al., 2022), which largely limits these methods to capture the truly invariant subgraphs for OOD generalization.

To address this issue, we study the problem of handling graph distribution shifts with severe bias by explicitly estimating environment-related variant patterns and further discovering invariant subgraphs for OOD generalization, which remains largely unexplored in the literature due to the following challenges:

- It is highly non-trivial to disentangle invariant and variant subgraph patterns from their complex interactions within input graphs.
- It is challenging to estimate the degree of spurious correlations between the variant subgraphs and labels, since the variant subgraphs reflect the environment-related information.
- It is also challenging to learn invariant subgraphs based on the estimated degree of spurious correlations for OOD generalized predictions.

In this paper, we propose a novel inVarIant subgraph learning based on VAriance Contrastive Estimation (**VIVACE**) method, which is able to capture invariant subgraphs for achieving satisfactory OOD generalization under distribution shifts with severe bias. Specifically, we first propose an invariant and variant subgraph identification module to disentangle potentially invariant and variant patterns for input graphs. Then, we find that the variance information behind graph data is also important and should not be directly ignored as in most existing literature. So we propose a variant subgraph contrastive estimation module to explicitly model the degree of the spurious correlations between variant subgraphs and labels with self-supervised contrastive training, which is useful for predicting the graph labels under distribution shifts. Finally, we propose an inverse propensity weighting based invariant subgraph prediction module to reweight the invariant subgraph predictions for eliminating the spurious correlations and achieving OOD generalization. In this way, our **VIVACE** method can capture and utilize invariant subgraphs with stable power to make predictions under distribution shifts with severe bias. We provide comprehensive theoretical analyses to show that our proposed method can disentangle ground-truth invariant and variant subgraphs for achieving OOD generalization with a strong guarantee. We conduct extensive experiments to demonstrate the superiority of our method over state-of-the-art baselines.

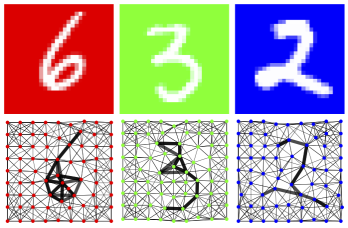


Figure 1. Example: the critical subgraph (i.e., digit subgraph, denoted by the bold lines) has truly predictive relations with the label (i.e., the digit) in the MNIST superpixel dataset (Fan et al., 2022; Dwivedi et al., 2023).

The contributions of our work are summarized as follows:

- We propose learning invariant subgraph via variant subgraph contrastive estimation, which can handle graph distribution shifts with severe bias. To the best of our knowledge, this is the first work to study how to explicitly utilize the variant subgraphs to facilitate accurate identification of invariant subgraphs under distribution shifts.
- We propose three mutually promoted modules to disentangle invariant and variant subgraphs, estimate the degree of spurious correlations, and make predictions based on the invariant subgraphs.
- We theoretically prove that our method can disentangle the ground-truth invariant and variant subgraphs which is a significant step towards OOD generalized predictions.
- Extensive experiments on several graph classification benchmark datasets demonstrate the superiority of our proposed method over state-of-the-art baselines.

The rest of the paper is organized as follows. We first formulate the problem in Section 2. In Section 3, we present the details of our proposed **VIVACE** method. We present the experimental results to show the effectiveness of the method in Section 4, including quantitative comparisons, ablation studies, hyper-parameter sensitivity, etc. We review the related works in Section 5. Finally, we conclude this work in Section 6.

2. Problem Formulation

The OOD generalization under distribution shifts on graphs can be formulated as:

Problem 1. The out-of-distribution generalization on graphs is to find an optimal graph predictor $f^*(\cdot)$ that maps input

graph G to the label y and performs well on all environments $\mathcal{E}$:

$$f^*(\cdot) = \arg\min_{f} \sup_{e\in\mathcal{E}} \mathcal{R}(f|e), \tag{1}$$

where $\mathcal{R}(f|e) = \mathbb{E}^e_{G,y}[\ell(f(G), y)]$ is the risk of the predictor f on the environment e, and ℓ denotes a loss function. We further decompose $f(\cdot) = w \circ h$, where $h(\cdot)$ is the encoder that maps input graph into the d-dimensional representation, and $w(\cdot)$ is the classifier.

Following the literature (Li et al., 2022b), we make the assumption:

Assumption 1. Given input graph G, there exists an invariant subgraph G_I^* satisfying:
a. Invariance assumption: $\forall e, e' \in \mathcal{E}, P^e(y|G_I^*) = P^{e'}(y|G_I^*)$.
b. Sufficiency assumption: $y = w_I(h_I(G_I^*)) + \epsilon,\ \epsilon \perp G$, where h_I denotes a graph encoder, w_I is the classifier, $\perp$ indicates statistical independence, and ϵ is random noise.

The invariance assumption means that there exists a subgraph inside the input graph that has invariant relations to the label across different environments. The sufficiency assumption means that the invariant subgraph has sufficient predictive capability for predicting the graph label. Overall, the graph OOD generalization problem can be solved by finding the invariant subgraph for the input graph.

3. Method

In this section, we introduce the details of our proposed **VIVACE**, which mainly consists of three key modules: invariant & variant subgraph identification module, variant subgraph contrastive estimation module, and inverse propensity weighting based invariant subgraph prediction module. The framework of **VIVACE** is shown in Figure 2.

3.1. Invariant & Variant Subgraph Identification

The graph invariant learning literature (Li et al., 2022d; Fan et al., 2022; Wu et al., 2022c) generally assumes that each graph G consists of an invariant subgraph $G_I \subset G$ (which is dominant to the label in the task and also has invariant relations to the label across different environments), and a variant subgraph (which could form spurious correlations and have variant relations with the label in different environments). Since variant subgraph is the complement of invariant subgraph in terms of the input graph, the accurate identification and estimation for variant subgraph can promote the modeling for invariant subgraph.

Given one input graph $G = (X, A)$, where X is the node feature matrix and A is the adjacency matrix of G, we first adopt the proposed invariant subgraph generator $\Phi(\cdot)$ to disentangle the invariant pattern in G under distribution shifts, so the invariant and variant subgraphs can be obtained:

$$G_I = \Phi(G),\ \ G_V = G \backslash G_I, \tag{2}$$

where $\Phi(\cdot)$ is instantiated as a learnable edge mask matrix $\mathbf{M}$ on G. Therefore,

$$G_I = (X, \mathbf{M} \odot A),\ \ G_V = (X, (1-\mathbf{M}) \odot A), \tag{3}$$

Here, we obtain the edge mask matrix $\mathbf{M}$ with a learnable GNN instead of directly learning the edge mask for each graph independently. It is because this design can identify the invariant patterns in a global view shared across the entire dataset and also easily generalize to handle unseen test graphs without retraining (Li et al., 2022d; Fan et al., 2022; Wu et al., 2022c). Specifically, each entry $\mathbf{M}_{i,j}$ of the edge mask matrix is calculated:

$$\mathbf{M}_{i,j} = \text{Sigmoid}\left(\text{MLP}\left([\mathbf{Z}_i^{(m)}, \mathbf{Z}_j^{(m)}]\right)\right),\ \ \mathbf{Z}^{(m)} = \text{GNN}^{\mathbf{M}}(G), \tag{4}$$

where $\mathbf{Z}^{(m)}$ is the node embedding for calculating the edge mask and $\text{MLP}(\cdot)$ is a multilayer perceptron. Finally, we can disentangle the invariant subgraph $G_{I,i}$ and variant subgraph $G_{V,i}$ for the i-th input graph G_i of the whole dataset[1]. We use soft edge weights to decompose input graph into invariant and variant subgraphs instead of hard splits, which is common in the literature for better optimization (Fan et al., 2022; Wu et al., 2022c). The Sigmoid function is employed to project the mask values into the interval $(0, 1)$, indicating the probability of the edge being classified into invariant subgraph.

3.2. Variant Subgraph Contrastive Estimation

Since GNNs tend to exploit the spurious correlations between the variant subgraphs and the labels to make predictions, which leads to poor OOD generalization, the existing literature aims to encourage GNNs only focusing on invariant subgraphs but ignoring the variant subgraphs. For achieving such a goal, they rely on the ideal predefined or automatically generated multiple environments for training. It is because only when we can explicitly observe environment-discriminative features among multiple environments, we can capture the variant subgraph that has variant correlations under different environments, so that we can disregard such variant subgraph but in turn capture the invariant subgraph that has invariant relationships between predictive graph structural information and the label for OOD generalization.

However, when the training graph data is with severe bias (e.g., imbalanced and less diverse), it is nearly impossible to obtain ideal multiple environments for learning invariant subgraphs. The variant pattern can not be fully disregarded from the invariant pattern (Qi et al., 2022), so that the performance in unseen testing graphs with distribution shifts is

[1]Note that for simplification, we omit the index i when there is no ambiguity.

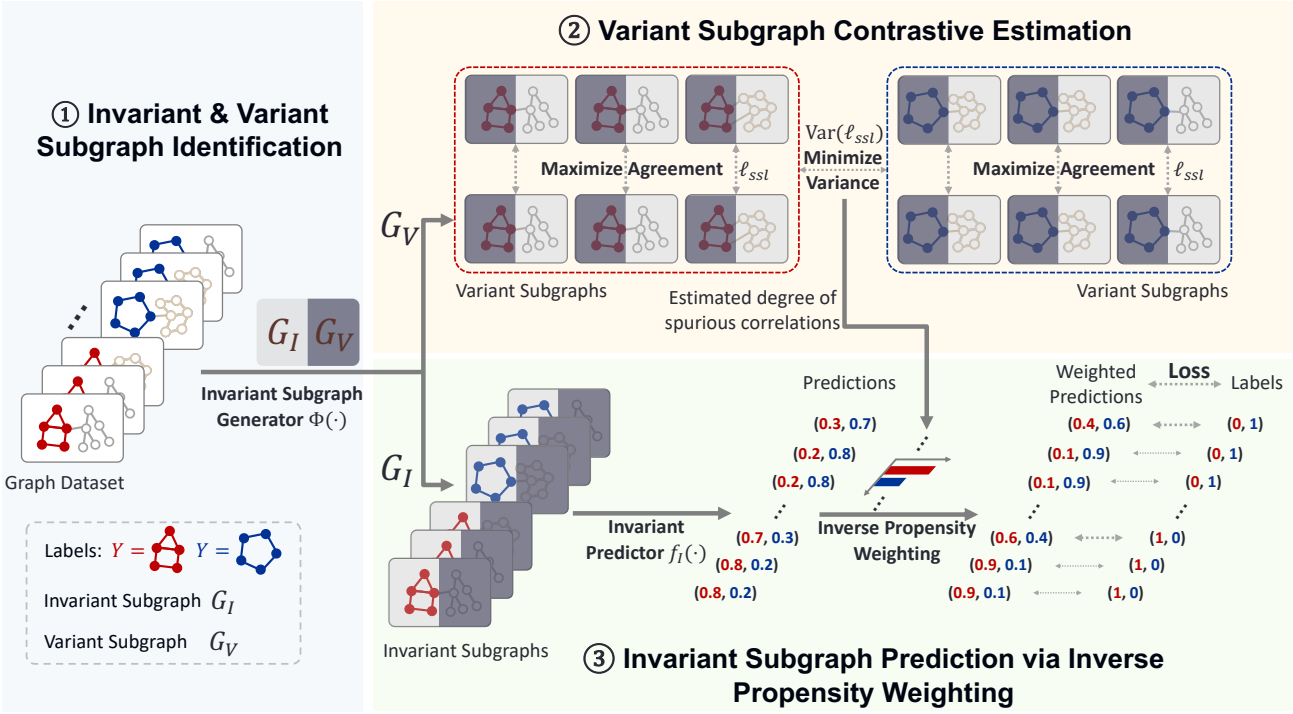


Figure 2. The framework of the proposed method **VIVACE**. **Training Stage:** Given the input graph dataset, the invariant subgraph generator $\Phi(\cdot)$ first disentangles the invariant subgraph G_I and variant subgraph G_V for each graph G. Then, the variant subgraphs are utilized for estimating the degree of the spurious correlation in a self-supervised contrastive manner. Finally, the invariant subgraph predictions are reweighted based on the accurate estimation of the spurious correlations between the variant subgraphs and labels, so as to eliminate the spurious correlations for OOD generalization. **Testing Stage:** We directly adopt the optimized invariant subgraph generator Φ^* and predictor f_I^* to make OOD generalized predictions $\hat{y} = f_I^*(\Phi^*(G))$.

not stable. Therefore, in this module, we propose a novel strategy to explicitly estimate the degree of the spurious correlations between the variant subgraph G_V and its label y for each input graph G, i.e., $p(y|G_V)$.

Let $f_V = w_V \circ h_V$ denote the variant subgraph predictor mapping the variant subgraph into the corresponding predicted label, which consists of the variant subgraph encoder h_V to learn representations and variant subgraph classifier w_V that makes predictions based on the learned representations. Since the invariant subgraph is truly predictive to the label, namely fully including the label information, the variant subgraph captures the environment-related information that is not available as the supervisions for training h_V. Therefore, we propose the following training objective with self-supervisions:

$$\ell_{V,ssl}(\Phi, h_V) = \frac{1}{|Y|}\sum_{k=1}^{|Y|} \ell_{ssl}(\Phi, h_V; k) + \alpha \mathrm{Var}\left(\ell_{ssl}\left(\Phi, h_V; k\right)\right). \tag{5}$$

Here Y denotes the label set of the graph dataset, ℓ_{ssl} is the contrastive loss for training the identification and prediction of the variant subgraph, and $\mathrm{Var}(\cdot)$ denotes the variation. More specifically, $\ell_{ssl}(\Phi, h_V; k)$ is defined as the contrastive loss for the graphs whose label is k, i.e.,

$$\ell_{ssl}(\Phi, h_V; k) = -\frac{1}{T_k}\sum_{y_i=k} \log \frac{\exp \phi(h_V(G_{V,i}), h_V(G'_{V,i}))}{\sum_{y_j=k} \exp \phi(h_V(G_{V,i}), h_V(G'_{V,j}))}, \tag{6}$$

where T_k is the number of training graph whose label is k and N is the total number of training graph in the dataset. Note that $\sum_{k=1}^{|Y|} T_k = N$. $G'_{V,i}$ is the augmented graph of $G_{V,i}$, where we adopt common graph augmentation strategies (You et al., 2020). This objective can maximize the agreement between G_V and G'_V. ϕ is the cosine similarity with temperature τ, i.e., $\phi(\mathbf{a}, \mathbf{b}) = \textsc{Cosine}(\mathbf{a}, \mathbf{b})/\tau$ and $\textsc{Cosine}(\mathbf{a}, \mathbf{b}) = \mathbf{a}^\top \mathbf{b}/(\|\mathbf{a}\|_2 \|\mathbf{b}\|_2)$.

However, only utilizing the self-supervised contrastive loss in Eq. (6) to train the variant subgraph encoder is not enough, since the invariant patterns will also be captured by the encoder during training. Therefore, inspired by the invariant learning literature (Krueger et al., 2021; Qi et al., 2022), we adopt the variation of the contrastive loss functions on all groups of graphs divided by the labels as the regularizer to encourage the encoder h_V only capturing the variant patterns. Intuitively, this regularizer can help the encoder

ignore the divergence among different groups but focus on their consistency. Since the training graphs are divided into different groups according to their labels, the invariant patterns that fully reflect the label information will be disregarded and only the variant patterns will be captured by the variant subgraph encoder h_V.

For better clarifications, we further explain how to prevent G_V from including invariant information. Rather than only optimizing the contrastive loss Eq. (6), we optimize the contrastive learning objective Eq. (5) with the second variance term to achieve this goal (where the contrastive loss in Eq. (6) is only the first term), as shown in Algorithm 1. The second variance term in Eq. (5) is proposed to minimize the difference (variance) among the contrastive loss from all G_V with different labels $1, 2, \ldots, |Y|$, where $|Y|$ is the number of unique labels. If G_V contains the invariant information with label by mistake, the variance term will increase since there exists a significant difference among label k and the other labels $1, \ldots, k-1, k+1, \ldots, |Y|$. Therefore, when the second variance term is properly optimized, it can encourage G_V to exclude invariant information to the label for achieving a minimum variance term. Finally, it will lead the variant subgraph encoder to capture the subgraph that only contains the variant subgraph.

Based on the trained variant subgraph encoder h_V, we further train the variant classifier w_V via the generalized cross entropy (GCE) loss (Lee et al., 2021; Fan et al., 2022) to fit the spurious correlations of the variant classifier:

$$\ell_V(\Phi, f_V) = \mathrm{GCE}(y, w_V(h_V(G_V))) = \frac{1 - (w_V^y(f_V(G_V)))^q}{q}, \tag{7}$$

where $w_V^y(f_V(G_V))$ means the predicted probability on the label y, i.e., the softmax output of the y-th dimensionality in practice. And $q \in (0, 1]$ is a hyperparameter that controls the degree of fitting the spurious correlations. Finally, we can rely on the trained variant subgraph predictor $f_V = w_V \circ h_V$ to accurately estimate the degree of the spurious correlations between the variant subgraph and its label $p(y|G_V)$, whose implementation will be introduced in the next module.

3.3. Invariant Subgraph Prediction via Inverse Propensity Weighting

Although we do not have any annotations to specify the variant subgraph in the dataset, we are able to estimate the relations between the variant subgraph and the label, which can largely help to learn accurate relations between the invariant subgraph and the label for OOD generalization. Specifically, we adopt propensity score techniques in causality (Jung et al., 2020; Seaman & Vansteelandt, 2018; Qi et al., 2022) to reweight the invariant subgraph predictions for eliminating the spurious correlations. Let f_I denote the invariant subgraph predictor mapping the invariant subgraph into the corresponding predicted label. We adopt the following inverse propensity weighted (IPW) loss (Jung et al., 2020; Seaman & Vansteelandt, 2018; Qi et al., 2022) to optimize the invariant subgraph predictor:

$$\ell_I(\Phi, f_I) = \sum_G \frac{1}{P(y|G_V)} \cdot \mathrm{CE}(y, f_I(\Phi(G))), \tag{8}$$

where CE is the standard cross-entropy loss, and the propensity score function $p(y|G_V)$ (Fan et al., 2022; Qi et al., 2022) is calculated by the following equation:

$$P(y|G_V) = \frac{\mathrm{CE}(y, f_V(G_V))}{\mathrm{CE}(y, f_I(\Phi(G))) + \mathrm{CE}(y, f_V(G_V))}. \tag{9}$$

Intuitively, for the input G, we expect the invariant subgraph generator $\Phi(\cdot)$ and predictor $f_I(\cdot)$ can capture the invariant ground-truth relation $p(y|\Phi(G))$ for OOD generalization. Therefore, we need to explicitly estimate the degree of the spurious correlations between the variant subgraph and the label:

$$p(y|G_V) = p(y|G \backslash \Phi(G)). \tag{10}$$

If $p(y|G_V)$ is large, the Eq. (9) can under-weight the prediction loss to reduce the over-large spurious correlations. And a small $p(y|G_V)$ means the variant subgraph has very few impacts on the prediction under distribution shifts, so the prediction loss should be up-weighted to encourage the invariant subgraph generator and predictor for capturing invariant patterns and improving OOD generalization.

For the optimization, we jointly optimize the self-supervised contrastive objective in Eq. (5), the supervised objectives in Eq. (8) and Eq. (7) as follows:

$$\Phi^*, f_I^*, f_V^* = \operatorname{argmin}\ \ell_I(\Phi, f_I) + \ell_V(\Phi, f_V) + \lambda \ell_{V,ssl}(\Phi, h_V). \tag{11}$$

3.4. Optimization Procedure

We present the pseudocode of **VIVACE** in Algorithm 1 to show the training procedure.

At the testing stage, we directly adopt the optimized Φ^* to capture the invariant subgraph for each input testing graph G_{te} and further feed it into optimized f_I^* to make prediction as follows:

$$\hat{y} = f_I^*(\Phi^*(G_{te})). \tag{12}$$

Since the predictions only depend on the subgraphs that have invariant relations with the labels, the OOD generalization performance can be largely improved.

3.5. Theoretical Analyses

We theoretically demonstrate that the proposed **VIVACE** model can achieve OOD generalization by disentangling the ground-truth invariant and variant subgraphs.

Algorithm 1 The training procedure of **VIVACE**.

Input: The graph dataset
Output: An optimized invariant subgraph generator $\Phi(\cdot)$ and predictor $f_I(\cdot)$ mapping each graph to its label

while not converge **do**
  for sampled minibatch $\mathcal{B}$ of the graph dataset **do**
    for each graph G and the corresponding label y in $\mathcal{B}$ **do**
      Generate the edge mask matrix $\mathbf{M}$ by Eq. (4).
      Generate the invariant subgraph G_I and the variant subgraph G_V by Eq. (3).
    end for
    Calculate the contrastive learning objective by Eq. (5).
    Calculate the objective of variant subgraph predictor by Eq. (7).
    Obtain the propensity score function for reweighting by Eq. (9).
    Calculate the objective of invariant subgraph predictor by Eq. (8).
    Obtain the overall objective by Eq. (11).
    Update model parameters by backpropagation.
  end for
end while

Theorem 1. *Denote the optimal invariant subgraph generator Φ^* that disentangles the ground-truth invariant subgraph G_I^* and variant subgraph G_V^* given the input graph G, where G_I^* satisfies Assumption 1 and denote the complement as $G_V^* = G \backslash G_I^*$. Assume the second variance term of Eq. (5) is minimized, we have that the first contrastive loss term is minimized iff the invariant subgraph generator Φ equals Φ^*.*

The proof is also shown in Appendix. The theorem shows that our proposed method can identify the ground-truth invariant and variant subgraphs by the variance contrastive estimation for accurately modeling the degree of the spurious correlations and further eliminating the spurious correlations to achieve OOD generalization.

3.6. Complexity Analysis

We provide the detailed time complexity analysis of the proposed **VIVACE** method as follows. The time complexity of **VIVACE** is $O(|E|d + |V|d^2)$, where $|V|$ and $|E|$ indicate the number of nodes and edges of the input graph, respectively, and d is the dimensionality of the representations. Specifically, we use GCN (Kipf & Welling, 2017), a message-passing GNN, to instantiate the GNN components in our method, which has a complexity of $O(|E|\,d + |V|\,d^2)$. We disentangle the invariant and variant subgraphs by generating mask for the existing edges of the input graph, so the time complexity is $O(|E|\,d + |V|\,d^2)$. The variant subgraph contrastive estimation module and inverse propensity weighting based invariant prediction module do not introduce a higher time complexity. Therefore, the time complexity of our proposed **VIVACE** method is comparable with the baselines, demonstrating its promising efficiency.

4. Experiments

In this section, we conduct extensive experiments to verify that our **VIVACE** method can effectively handle distribution shifts even on the severely biased graph datasets by capturing the invariant subgraphs, including the experimental setup, quantitative comparisons, ablation studies, the impact of the hyper-parameters, etc.

4.1. Experiment Setup

Baselines. We consider several representative methods that are widely used in the literature (Li et al., 2022a; Fan et al., 2022) as the baselines:

- GCN (Kipf & Welling, 2017): It follows the recursive neighborhood aggregation scheme and is considered one of the most popular GNNs.
- GIN (Xu et al., 2019): It is also one famous GNN and has shown to be one of the most expressive GNNs in the representation learning of graphs.
- FactorGCN (Yang et al., 2020): It is a representative graph disentangling model for graph classification.
- DiffPool (Ying et al., 2018): It is a graph pooling method that learns the cluster assignment for each node and outputs the coarsened graph.
- LDD (Lee et al., 2021): It can learn debiased representations via disentangled feature augmentation, which is a general debiasing method.
- DIR (Wu et al., 2022c): It generates multiple environments from biased graph data by conducting interventions on graphs and further capturing invariant explainable subgraphs for predictions.
- DisC (Fan et al., 2022): It is a notable debiasing method for GNNs, which aims to explicitly distinguish causal or bias patterns from the input graphs.

Datasets. We consider five widely adopted datasets in the literature for comprehensive evaluations. First, following (Fan et al., 2022; Dwivedi et al., 2023), we adopt three datasets, i.e., CMNIST, CFashion, and CKuzushiji, which are converted from image datasets using superpixels (Knyazev et al., 2019), since they have controllable bias degrees and clear human-understandable ground-truth invariant subgraphs for evaluations. Specifically, for CMNIST, each graph is converted from an image in MNIST (LeCun et al., 1998). The task is to classify each graph into the corresponding handwritten digit. And the spurious correlations are introduced via colorizing the background based on the correlations with the label. The degree of such correlation is controlled by r. We consider datasets with biased $r = \{0.8, 0.9, 0.95\}$

Table 1. Experimental results (%) of our method and baselines. The evaluation metric is accuracy for CMNIST, CFashion, and CKuzushiji, and ROC-AUC for MOLSIDER and MOLHIV. ± denotes the standard deviation. The best results are in bold for each row. Our **VIVACE** outperforms the baselines in all comparisons, indicating its superiority against graph distribution shifts.

Dataset	Bias	Methods							
	r	GCN	GIN	FactorGCN	DiffPool	DIR	LDD	DisC	**VIVACE**
CMNIST	0.8	50.43±4.13	57.75±0.78	72.30±1.18	73.79±0.02	9.98±0.33	64.95±1.22	82.60±0.93	**82.71±0.74**
	0.9	28.97±4.40	36.78±5.55	62.35±5.07	66.45±0.78	9.96±0.23	56.65±2.18	78.14±2.14	**79.46±1.87**
	0.95	13.50±1.38	16.04±1.14	42.50±4.91	47.12±1.04	10.03±0.27	46.83±2.88	63.47±5.65	**64.72±4.61**
CFashion	0.8	63.60±0.53	64.25±0.46	61.23±1.11	62.82±0.53	13.02±1.92	63.85±1.17	66.85±1.11	**67.09±1.23**
	0.9	57.22±0.93	58.03±0.40	53.50±1.29	57.50±0.39	12.80±1.67	64.30±0.89	65.33±4.70	**65.38±4.18**
	0.95	47.69±0.42	49.74±0.60	45.78±2.40	50.86±0.20	11.98±1.41	62.28±0.48	63.93±1.50	**63.96±1.27**
CKuzushiji	0.8	38.45±1.10	41.83±0.78	42.87±1.19	45.46±0.65	10.35±0.32	42.38±0.33	55.53±2.29	**55.58±1.87**
	0.9	28.35±0.79	30.09±0.87	32.35±2.79	36.18±0.19	10.72±0.27	38.75±0.49	48.13±2.59	**48.15±1.91**
	0.95	20.70±0.88	21.18±1.63	23.87±0.12	27.45±0.26	10.59±0.46	33.08±0.59	36.63±1.73	**37.01±1.67**
MOLSIDER		59.62±1.82	57.61±1.48	53.32±1.75	60.21±1.55	57.74±1.63	58.83±1.62	59.31±1.87	**62.15±1.10**
MOLHIV		76.13±1.01	75.63±1.41	57.18±1.54	76.32±1.48	77.05±0.57	76.91±1.81	76.97±1.03	**78.11±0.82**

for training and unbiased r for testing. We also adopt similar strategies to construct CFashion and CKuzushiji from Fashion-MNIST (Xiao et al., 2017) and Kuzushiji-MNIST (Clanuwat et al., 2018) datasets. Also, we consider two datasets, MOLSIDER and MOLHIV from Open Graph Benchmark (Hu et al., 2020). The default split separates structurally different molecules with different scaffolds into different subsets, i.e., training/validation/testing sets. We report the accuracy for CMNIST, CFashion, and CKuzushiji, ROC-AUC for MOLSIDER and MOLHIV.

4.2. Experiment Results

The experimental results are reported in Table 1. We have the following observations.

Our proposed **VIVACE** method consistently and significantly outperforms the GNN backbones (i.e., GCN and GIN) on the CMNIST, CFashion, and CKuzushiji datasets. When compared with the graph disentangling method FactorGCN or representative graph pooling method, our proposed **VIVACE** also achieves substantial performance gains. Besides, the representative graph invariant learning method DIR achieves unsatisfactory prediction performances. It is because DIR relies on datasets without severe bias to create interventional distributions, so that the invariant patterns captured by DIR are not accurate when this assumption is not valid. The general debiasing method LDD and graph debiasing method DisC show promising performance gains upon the other baselines, but our **VIVACE** still performs better than them in all comparisons. One plausible reason is that they can not fully leverage the informative variant patterns for capturing invariant subgraphs to achieve OOD generalized predictions. Our method can capture accurate invariant and variant subgraphs simultaneously by the mutually promoted two modules respectively, i.e., variant subgraph contrastive estimation module and propensity score based invariant subgraph prediction module, showing the remarkable OOD generalization ability in practice.

As the degree of spurious correlations increases, namely r grows larger, the performance of all the methods tends to decrease since there exists a larger degree of distribution shifts between testing and training graph data. Nevertheless, our proposed **VIVACE** is able to keep the most relatively stable performances, and demonstrates the effectiveness in handling graph distribution shifts. When r increases from 0.8 to 0.95 on CFashion dataset, GCN drops by about 15% accuracy, while our method with GCN as the backbone can drop no more than 4%, which verifies that our method can better capture the invariant and variant subgraphs, and thus well handle distribution shifts.

We also conduct comparisons on more challenging and large-scale OGB datasets. Since the molecules with different scaffolds are naturally split into different training/validation/testing sets, the generalization between molecules with different scaffolds is difficult. Nevertheless, our proposed **VIVACE** also achieves improvements upon the best results of the baselines. For example, **VIVACE** increases the ROC-AUC by 1.94% on MOLSIDER and 1.06% on MOLHIV against the strongest baselines respectively. Although the baseline DisC shows competitive performances on CMNIST, CFashion, CKuzushiji, it fails to achieve promising OOD generalization results on these two more challenging OGB datasets. In contrast, **VIVACE** achieves significant and consistent performance gains in all comparisons, demonstrating its effectiveness against distribution shifts.

Training Dynamics. In Figure 5 in Appendix, we plot the loss and accuracy in the training process on CMNIST (r = 0.9), while the results on the other datasets show similar patterns. We can empirically observe the convergence of

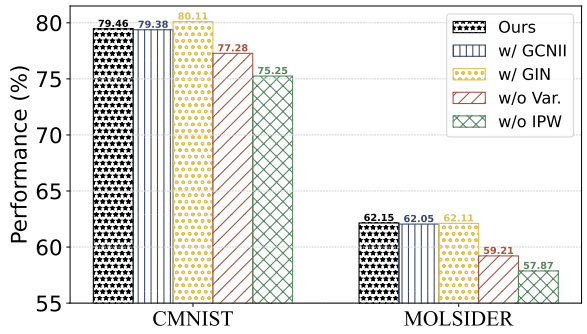
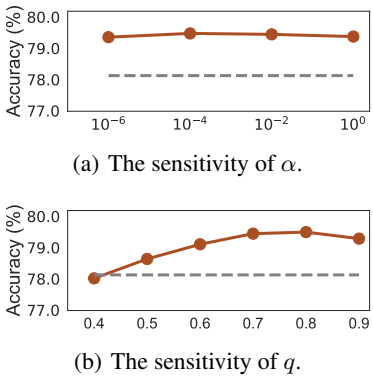


Figure 3. Ablation studies. 'w/ GCNII' and 'w/ GIN' denote adopting different backbones. 'w/o Var.' and 'w/o IPW' denote removing the variant subgraph contrastive module and inverse propensity weighting module.



(a) The sensitivity of α.

(b) The sensitivity of q.

Figure 4. Sensitivity analysis for some important hyper-parameters in our model. The red and grey lines denote the results of **VIVACE** and the best result of all baselines, respectively. **VIVACE** achieves better results with a wide range of hyper-parameter choices.

our proposed method. The loss and accuracy will converge before reaching the maximal training epoch.

4.3. Ablation Study

We perform comprehensive ablation studies to further validate the effectiveness of the key modules in our method.

- Variants 'w/GCNII' or "w/ GIN' means that we replace the instantiation of the GNN components from GCN (Kipf & Welling, 2017) to the other GNN backbone GCNII (Chen et al., 2020) or GIN (Xu et al., 2019).
- Variant 'w/o Var.' means that we remove the variant subgraph contrastive estimation module and Variant 'w/o IPW' further removes the inverse propensity weighting module.

The results are obtained from the CMNIST ($r = 0.9$) dataset. From Figure 3, we can first observe that our method is also compatible with the other popular GNNs even with slight performance improvements. By default, we use GCN (Kipf & Welling, 2017) as the backbone GNN model. Although the backbone GNN is replaced with GCNII or GIN model, our **VIVACE** can keep relatively stable performances or even achieve gains. Specifically, the generalization performance further increases by 0.65% when using GIN as the backbone that is a more expressive GNN.

Besides, the performance of the variants 'w/o Var.' and 'w/o IPW' drop drastically, indicating that it is important to explicitly identify variant subgraphs and further estimate their impact on the labels for the final goal of learning invariant subgraphs for OOD generalization. If the spurious correlations of all variant subgraphs with the labels are treated equally, the accurate identification of the invariant subgraphs will also be affected, leading to the unsatisfactory OOD generalization performance.

4.4. Hyper-parameter Sensitivity

We analyze the hyper-parameter sensitivity of our proposed method in terms of the invariance regularizer coefficient α in the variant subgraph contrastive estimation module and the coefficient q that controls the degree of fitting the spurious correlations in Eq. (7).

For simplicity, we only report the results on CMNIST ($r = 0.9$), while the results on other datasets show similar patterns. In Figure 4, the red and grey lines denote the results of **VIVACE** and the best result of all baselines, respectively. The hyper-parameter α has an influence on the OOD generalization performance, indicating that we need to properly balance the self-supervised contrastive learning loss and the invariance regularizer term. A small value may not be sufficient to encourage the invariance among different environments effectively, while a very large value may affect the self-supervised contrastive training. We also observe the hyperparameter q has a moderate impact on the model performance, verifying the significance to fit the spurious correlations. Overall, although an appropriate choice of the hyper-parameters can further achieve generalization performance gains, our method is not sensitive to them and is able to outperform the best baselines within a wide range of hyper-parameters choices.

5. Related Works

In this section, we review the related works of graph neural network, disentangled representation learning, and OOD generalization.

Graph Neural Network. Graph data has been widely used to model the complex relationships between different entities. In the field of graph machine learning, graph neural networks (GNNs) (Kipf & Welling, 2017; Veličković et al.,

2018; Li et al., 2023a; 2025; Chen et al., 2025), following a message passing paradigm to iteratively update node representations by the neighbors, have drawn ever-increasing attention and achieved enormous success in various applications across wide-ranging applications. Graph-level prediction task is one of the popular applications and can be abstracted as one main branch in addition to the node-level and link-level prediction tasks. The prediction of a graph is generally based on its critical subgraph rather than the whole input graph (Luo et al., 2020; Yu et al., 2021; Xuan et al., 2019), where the critical subgraph is regarded as the part that is truly predictive to the label in the task. There are some works (Ying et al., 2018; Luo et al., 2020; Sun et al., 2021; Monti et al., 2018) proposed to improve the graph-level prediction performances by capturing the critical subgraph for learning graph representations effectively. However, these methods are built upon the in-distribution (I.D.)hypothesis and fail to generalize under distribution shifts.

Disentangled Representation Learning. Disentangled representation learning (DRL) (Wang et al., 2024) aims to learn representations capable of identifying the underlying factors behind the observable data (Zhang et al., 2024b; Wang et al., 2022; Zhang et al., 2023a; Wang et al., 2025a). In addition to its success in image and video (Wang et al., 2025b; Chen et al., 2024b;a), DRL is also a promising direction in the domain of graph neural network (Li et al., 2021b; 2022c; Zhang et al., 2023c). For instance, DisenGCN (Ma et al., 2019) is the pioneering approach in this area, introducing a method to disentangle node representations by dynamically extracting factors that contribute to the formation of edges between a node and its neighbors. Building upon this, IPGDN (Liu et al., 2020) not only separates the factors connecting nodes to their neighbors but also ensures these factors to be independent. Both methods are guided by supervision from downstream node classification tasks. FactorGCN (Yang et al., 2020) further advances disentangled learning by applying a factoring mechanism at the input graph level. This approach disentangles input graph to create distinct factors, which are subsequently treated as separate graphs. However, their generalization abilities under distribution shifts remain relatively underexplored.

Out-of-Distribution (OOD) Generalization. Most machine learning models rely on the assumption that testing and training data are identically distributed. However, this assumption can be easily violated due to the widely existing but uncontrollable distribution shifts in the real world (Shen et al., 2021; Du et al., 2022; Yang et al., 2023; Sui et al., 2023). The performance will degenerate significantly if the machine learning models do not have strong OOD generalization ability. Graph neural networks, as the most popular models in the graph community recently, also face the same obstacle. Considering the increasing demand for handling in-the-wild unseen data (Feng et al., 2023), OOD generalization on graphs has drawn great attention (Li et al., 2022a; Gui et al., 2022; Cai et al., 2024). Several famous works are proposed to tackle this problem on graphs by learning subgraph backed by different theories or assumptions, including causality (Wu et al., 2022c; Sui et al., 2022), invariant learning (Wu et al., 2022a; Li et al., 2022d; Yao et al., 2024; Chen et al., 2024c; Zhang et al., 2023b), disentanglement (Fan et al., 2022; Li et al., 2024; Zhang et al., 2024a), information bottleneck (Miao et al., 2022). Different from these works that output explainable or invariant subgraphs under distribution shifts, some works directly learn generalizable graph representations for the problems where distribution shifts exist on graph size (Bevilacqua et al., 2021; Buffelli et al., 2022) or the other structural patterns (Wu et al., 2024). And the learned representations are expected to remain invariant across different environments. For capturing invariant subgraphs, most of them heavily rely on the predefined or automatically generated environment labels, i.e., multiple training environments to specify the variant information, and further learn invariant subgraphs. However, the environment labels are unavailable and directly generating environment labels during the training process is also impractical, leading to inaccurate modeling for the variant patterns and the estimation of the degree of the spurious correlations. Some of them rely on the strong causality assumptions that are hardly guaranteed to hold true all the time in real-world scenarios. How to explicitly estimate the impact of environment-related variant information and further eliminate the spurious correlations remains largely unexplored.

6. Conclusion

In this paper, we study learning invariant subgraph via self-supervised contrastive variant subgraph estimation, which can well handle graph distribution shifts with severe bias for OOD generalization. We find that the variance information behind graph data is also important and should not be directly ignored but be explicitly captured. We propose a novel **VIVACE** model, which consists of three tailored modules, i.e., invariant and variant subgraph identification module, variant subgraph contrastive estimation module, and inverse propensity weighting based invariant subgraph prediction module. The main technical contribution lies in that we design self-supervised contrastive learning on variant subgraphs to explicitly model the degree of the spurious correlations and further design the inverse propensity weighting strategy to reweight the invariant subgraph predictions to eliminate the spurious correlations for OOD generalization. We conduct extensive experiments on several graph classification benchmark datasets. The results demonstrate the superiority of our proposed method over state-of-the-art baselines against distribution shifts.

Acknowledgements

This work is supported by National Natural Science Foundation of China (No.62222209, 62473380, 62272481, 61972460, 62372470, 62172440), Beijing National Research Center for Information Science and Technology under Grant No.BNR2023TD03006, The Science and Technology Innovation Program of Hunan Province (2023RC1029).

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Bengio, Y., Deleu, T., Rahaman, N., Ke, R., Lachapelle, S., Bilaniuk, O., Goyal, A., and Pal, C. A meta-transfer objective for learning to disentangle causal mechanisms. *International Conference on Learning Representations*, 2019.

Bevilacqua, B., Zhou, Y., and Ribeiro, B. Size-invariant graph representations for graph classification extrapolations. In *International Conference on Machine Learning*, pp. 837–851. PMLR, 2021.

Buffelli, D., Liò, P., and Vandin, F. Sizeshiftreg: a regularization method for improving size-generalization in graph neural networks. *Advances in Neural Information Processing Systems*, 35:31871–31885, 2022.

Cai, J., Wang, X., Li, H., Zhang, Z., and Zhu, W. Multimodal graph neural architecture search under distribution shifts. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 8227–8235, 2024.

Chen, H., Wang, X., Zhang, Y., Zhou, Y., Zhang, Z., Tang, S., and Zhu, W. Disenstudio: Customized multi-subject text-to-video generation with disentangled spatial control. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 3637–3646, 2024a.

Chen, H., Zhang, Y., Wu, S., Wang, X., Duan, X., Zhou, Y., and Zhu, W. Disenbooth: Identity-preserving disentangled tuning for subject-driven text-to-image generation. *International Conference on Learning Representations*, 2024b.

Chen, H., Wang, X., Zhang, Z., Li, H., Feng, L., and Zhu, W. Autogfm: Automated graph foundation model with adaptive architecture customization. In *International Conference on Machine Learning*, 2025.

Chen, M., Wei, Z., Huang, Z., Ding, B., and Li, Y. Simple and deep graph convolutional networks. In *International conference on machine learning*, pp. 1725–1735. PMLR, 2020.

Chen, Y., Bian, Y., Zhou, K., Xie, B., Han, B., and Cheng, J. Does invariant graph learning via environment augmentation learn invariance? *Advances in Neural Information Processing Systems*, 36, 2024c.

Clanuwat, T., Bober-Irizar, M., Kitamoto, A., Lamb, A., Yamamoto, K., and Ha, D. Deep learning for classical japanese literature. *arXiv preprint arXiv:1812.01718*, 2018.

Du, X., Wu, Z., Feng, F., He, X., and Tang, J. Invariant representation learning for multimedia recommendation. In *Proceedings of the 30th ACM International Conference on Multimedia*, pp. 619–628, 2022.

Duvenaud, D. K., Maclaurin, D., Iparraguirre, J., Bombarell, R., Hirzel, T., Aspuru-Guzik, A., and Adams, R. P. Convolutional networks on graphs for learning molecular fingerprints. *Neural Information Processing Systems*, 28, 2015.

Dwivedi, V. P., Joshi, C. K., Luu, A. T., Laurent, T., Bengio, Y., and Bresson, X. Benchmarking graph neural networks. *Journal of Machine Learning Research*, 24 (43):1–48, 2023.

Fan, S., Wang, X., Mo, Y., Shi, C., and Tang, J. Debiasing graph neural networks via learning disentangled causal substructure. In *Advances in Neural Information Processing Systems*, 2022.

Feng, W., Li, H., Wang, X., Duan, X., Qian, Z., Liu, W., and Zhu, W. Multimedia cognition and evaluation in open environments. In *Proceedings of the 1st International Workshop on Multimedia Content Generation and Evaluation: New Methods and Practice*, pp. 9–18, 2023.

Fey, M. and Lenssen, J. E. Fast graph representation learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019.

Gui, S., Li, X., Wang, L., and Ji, S. Good: A graph out-of-distribution benchmark. *Advances in Neural Information Processing Systems*, 35:2059–2073, 2022.

Hu, W., Fey, M., Zitnik, M., Dong, Y., Ren, H., Liu, B., Catasta, M., and Leskovec, J. Open graph benchmark: Datasets for machine learning on graphs. *Neural Information Processing Systems*, 2020.

Hudson, D. and Manning, C. D. Learning by abstraction: The neural state machine. *Neural Information Processing Systems*, 2019.

Jung, Y., Tian, J., and Bareinboim, E. Learning causal effects via weighted empirical risk minimization. *Neural Information Processing Systems*, 33:12697–12709, 2020.

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

Kipf, T. N. and Welling, M. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017.

Knyazev, B., Taylor, G. W., and Amer, M. Understanding attention and generalization in graph neural networks. *Advances in neural information processing systems*, 32, 2019.

Krueger, D., Caballero, E., Jacobsen, J.-H., Zhang, A., Binas, J., Zhang, D., Le Priol, R., and Courville, A. Out-of-distribution generalization via risk extrapolation (rex). In *International Conference on Machine Learning*, pp. 5815–5826. PMLR, 2021.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998.

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

Li, H., Cui, P., Zang, C., Zhang, T., Zhu, W., and Lin, Y. Fates of microscopic social ecosystems: Keep alive or dead? In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 668–676, 2019.

Li, H., Wang, X., Zhang, Z., Ma, J., Cui, P., and Zhu, W. Intention-aware sequential recommendation with structured intent transition. *IEEE Transactions on Knowledge and Data Engineering*, 34(11):5403–5414, 2021a.

Li, H., Wang, X., Zhang, Z., Yuan, Z., Li, H., and Zhu, W. Disentangled contrastive learning on graphs. *Advances in Neural Information Processing Systems*, 34, 2021b.

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

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

Li, H., Zhang, Z., Wang, X., and Zhu, W. Disentangled graph contrastive learning with independence promotion. *IEEE Transactions on Knowledge and Data Engineering*, 2022c.

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

Li, H., Wang, X., and Zhu, W. Curriculum graph machine learning: A survey. *International Joint Conference on Artificial Intelligence*, 2023a.

Li, H., Zhang, Z., Wang, X., and Zhu, W. Invariant node representation learning under distribution shifts with multiple latent environments. *ACM Transactions on Information Systems*, 42(1):1–30, 2023b.

Li, H., Wang, X., Zhang, Z., Chen, H., Zhang, Z., and Zhu, W. Disentangled graph self-supervised learning for out-of-distribution generalization. In *International Conference on Machine Learning*, 2024.

Li, H., Wang, X., Zhang, Z., Wu, Z., Xiao, L., and Zhu, W. Self-supervised masked graph autoencoder via structure-aware curriculum. In *International Conference on Machine Learning*, 2025.

Liu, Y., Wang, X., Wu, S., and Xiao, Z. Independence promoted graph disentangled networks. In *Association for the Advancement of Artificial Intelligence*, volume 34, pp. 4916–4923, 2020.

Luo, D., Cheng, W., Xu, D., Yu, W., Zong, B., Chen, H., and Zhang, X. Parameterized explainer for graph neural network. *Neural Information Processing Systems*, 2020.

Ma, J., Cui, P., Kuang, K., Wang, X., and Zhu, W. Disentangled graph convolutional networks. In *International conference on machine learning*, pp. 4212–4221. PMLR, 2019.

Marcheggiani, D. and Titov, I. Encoding sentences with graph convolutional networks for semantic role labeling. *arXiv preprint arXiv:1703.04826*, 2017.

Miao, S., Liu, M., and Li, P. Interpretable and generalizable graph learning via stochastic attention mechanism. In *International Conference on Machine Learning*, pp. 15524–15543, 2022.

Monti, F., Otness, K., and Bronstein, M. M. Motifnet: a motif-based graph convolutional network for directed graphs. In *2018 IEEE Data Science Workshop (DSW)*, pp. 225–228. IEEE, 2018.

Qi, J., Tang, K., Sun, Q., Hua, X.-S., and Zhang, H. Class is invariant to context and vice versa: on learning invariance for out-of-distribution generalization. In *Computer Vision–ECCV 2022*, pp. 92–109, 2022.

Qiu, J., Tang, J., Ma, H., Dong, Y., Wang, K., and Tang, J. Deepinf: Social influence prediction with deep learning. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2110–2119, 2018.

Seaman, S. R. and Vansteelandt, S. Introduction to double robust methods for incomplete data. *Statistical science: a review journal of the Institute of Mathematical Statistics*, 33(2):184, 2018.

Shen, Z., Liu, J., He, Y., Zhang, X., Xu, R., Yu, H., and Cui, P. Towards out-of-distribution generalization: A survey. *arXiv preprint arXiv:2108.13624*, 2021.

Sui, Y., Wang, X., Wu, J., Lin, M., He, X., and Chua, T.-S. Causal attention for interpretable and generalizable graph classification. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1696–1705, 2022.

Sui, Y., Wu, Q., Wu, J., Cui, Q., Li, L., Zhou, J., Wang, X., and He, X. Unleashing the power of graph data augmentation on covariate distribution shift. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Sun, Q., Li, J., Peng, H., Wu, J., Ning, Y., Yu, P. S., and He, L. Sugar: Subgraph neural network with reinforcement pooling and self-supervised mutual information mechanism. In *Proceedings of the Web Conference 2021*, pp. 2081–2091, 2021.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., and Bengio, Y. Graph attention networks. In *International Conference on Learning Representations*, 2018.

Wang, Q., Mao, Z., Wang, B., and Guo, L. Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering*, 2017.

Wang, X., Chen, H., Zhou, Y., Ma, J., and Zhu, W. Disentangled representation learning for recommendation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(1):408–424, 2022.

Wang, X., Chen, H., Wu, Z., Zhu, W., et al. Disentangled representation learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

Wang, X., Chen, H., Pan, Z., Zhou, Y., Guan, C., Sun, L., and Zhu, W. Automated disentangled sequential recommendation with large language models. *ACM Transactions on Information Systems*, 43(2):1–29, 2025a.

Wang, X., Pan, Z., Chen, H., and Zhu, W. Divico: Disentangled visual token compression for efficient large vision-language model. *IEEE Transactions on Circuits and Systems for Video Technology*, 2025b.

Wu, Q., Zhang, H., Yan, J., and Wipf, D. Handling distribution shifts on graphs: An invariance perspective. In *International Conference on Learning Representations*, 2022a.

Wu, S., Sun, F., Zhang, W., Xie, X., and Cui, B. Graph neural networks in recommender systems: a survey. *ACM Computing Surveys*, 55(5):1–37, 2022b.

Wu, S., Cao, K., Ribeiro, B., Zou, J., and Leskovec, J. Graphmetro: Mitigating complex distribution shifts in gnns via mixture of aligned experts. *Neural Information Processing Systems*, 2024.

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

Xiao, H., Rasul, K., and Vollgraf, R. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Xu, K., Hu, W., Leskovec, J., and Jegelka, S. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2019.

Xuan, Q., Wang, J., Zhao, M., Yuan, J., Fu, C., Ruan, Z., and Chen, G. Subgraph networks with application to structural feature space expansion. *IEEE Transactions on Knowledge and Data Engineering*, 33(6):2776–2789, 2019.

Yang, N., Zeng, K., Wu, Q., Jia, X., and Yan, J. Learning substructure invariance for out-of-distribution molecular representations. *Advances in Neural Information Processing Systems*, 35:12964–12978, 2022.

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

Yang, Z., He, X., Zhang, J., Wu, J., Xin, X., Chen, J., and Wang, X. A generic learning framework for sequential recommendation with distribution shifts. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2023.

Yao, T., Chen, Y., Chen, Z., Hu, K., Shen, Z., and Zhang, K. Empowering graph invariance learning with deep spurious infomax. *arXiv preprint arXiv:2407.11083*, 2024.

Ying, Z., You, J., Morris, C., Ren, X., Hamilton, W., and Leskovec, J. Hierarchical graph representation learning with differentiable pooling. *Neural Information Processing Systems*, 31, 2018.

You, Y., Chen, T., Sui, Y., Chen, T., Wang, Z., and Shen, Y. Graph contrastive learning with augmentations. *Advances in Neural Information Processing Systems*, 33: 5812–5823, 2020.

Yu, J., Xu, T., Rong, Y., Bian, Y., Huang, J., and He, R. Graph information bottleneck for subgraph recognition. In *International Conference on Learning Representations*, 2021.

Zellers, R., Yatskar, M., Thomson, S., and Choi, Y. Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5831–5840, 2018.

Zhang, Y., Wang, X., Chen, H., and Zhu, W. Adaptive disentangled transformer for sequential recommendation. In *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining*, pp. 3434–3445, 2023a.

Zhang, Z., Wang, X., Zhang, Z., Qin, Z., Wen, W., Xue, H., Li, H., and Zhu, W. Spectral invariant learning for dynamic graphs under distribution shifts. *Advances in Neural Information Processing Systems*, 36:6619–6633, 2023b.

Zhang, Z., Wang, X., Zhang, Z., Shen, G., Shen, S., and Zhu, W. Unsupervised graph neural architecture search with disentangled self-supervision. *Advances in Neural Information Processing Systems*, 36:73175–73190, 2023c.

Zhang, Z., Wang, X., Chen, H., Li, H., and Zhu, W. Disentangled dynamic graph attention network for out-of-distribution sequential recommendation. *ACM Transactions on Information Systems*, 43(1):1–42, 2024a.

Zhang, Z., Wang, X., Qin, Y., Chen, H., Zhang, Z., Chu, X., and Zhu, W. Disentangled continual graph neural architecture search with invariant modular supernet. In *International Conference on Machine Learning*, 2024b.

A. Notations

For better clarifications, we first summarize the key notations in the proposed method and the corresponding descriptions in Table 2.

Table 2. The summary of the key notations and the corresponding descriptions.

Notation	Description
N	The number of graphs
G	The input graph
X, A	The node feature and adjacency matrix
Φ	The invariant subgraph generator
$G_I = \Phi(G)$	The invariant subgraph of G
$G_V = G \backslash G_I$	The variant subgraph of G
$\mathbf{M}$	The learnable edge mask matrix
f_I / f_V	The invariant/variant subgraph predictor
h_V	The variant subgraph encoder
w_V	The variant subgraph classifier
ℓ	The loss function
$\lvert Y \rvert$	The number of different labels

B. Proof of Theorem 1

Proof. Denote the first contrastive loss term of Eq. (5) as:

$$L_{ssl} = \frac{1}{|Y|}\sum_{k=1}^{|Y|}\ell_{ssl}(\Phi, h_V; k) = \frac{1}{|Y|}\sum_{k=1}^{|Y|} -\frac{1}{T_k}\sum_{y_i=k}\log\frac{\exp\left(\phi\left(h_V(G_{V,i}), h_V(G'_{V,i})\right)\right)}{\sum_{y_j=k}\exp\left(\phi\left(h_V(G_{V,i}), h_V(G'_{V,j})\right)\right)}. \tag{13}$$

Denote the second variance term of Eq. (5) as:

$$L_{\text{var}} = \text{Var}\left(\ell_{ssl}(\Phi, h_V; k)\right), \quad k = 1, \ldots, |Y|. \tag{14}$$

$\Leftarrow$**:** To prove the optimal invariant subgraph generator Φ^* can minimize the contrastive loss L_{ssl}, i.e.,

$$\Phi^* = \arg\min_{\Phi} L_{ssl}, \tag{15}$$

we assume there exists another $\Phi' \neq \Phi^*$, where the input graph G is disentangled into $G = (G'_I, G'_V)$ and

$$L_{ssl}(\Phi') < L_{ssl}(\Phi^*). \tag{16}$$

This implies that G'_V includes the ground-truth invariant subgraph information, i.e.,

$$G'_V \cap G^*_I = G'_V \cap (G \backslash G^*_V) \neq \emptyset. \tag{17}$$

Therefore, the contrastive loss $\ell_{ssl}(\Phi, h_V; k)$ among the environments partitioned by the graph label $k = 1, \ldots, |Y|$ is dependent on the graph label itself, i.e.,

$$\exists\, k_1, k_2 \in \{1, \ldots, |Y|\},\ k_1 \neq k_2,\ \text{such that}\ \ell_{ssl}(\Phi', h_V; k_1) \neq \ell_{ssl}(\Phi', h_V; k_2). \tag{18}$$

Thus,

$$L_{\text{var}}(\Phi') = \text{Var}(\ell_{ssl}(\Phi, h_V; k)) > 0, \quad k = 1, \ldots, |Y|. \tag{19}$$

However, G^*_V excludes all the ground-truth invariant information that is sufficiently predictive to the graph label. We have

$$\ell_{ssl}(\Phi^*, h_V; k = 1) = \ell_{ssl}(\Phi^*, h_V; k = 2) = \cdots = \ell_{ssl}(\Phi^*, h_V; k = |Y|), \tag{20}$$

i.e.,

$$L_{\text{var}}(\Phi^*) = \text{Var}\left(\ell_{ssl}(\Phi^*, h_V; k)\right) = 0, \quad k = 1, \ldots, |Y|. \tag{21}$$

Thus,

$$L_{\text{var}}(\Phi') > L_{\text{var}}(\Phi^*), \tag{22}$$

which contradicts the assumption that the variance term L_{var} is minimized. Therefore, we prove that the optimal invariant subgraph generator Φ^* minimizes the contrastive loss L_{ssl}.

$\Rightarrow$**:** To prove that minimizing the contrastive loss L_{ssl} implies $\Phi = \Phi^*$, i.e., the uniqueness of Φ^*, assume there exists another invariant subgraph generator $\Phi' \neq \Phi^*$ derived by minimizing L_{ssl}, where $G = (G_I', G_V')$ and

$$\Phi' = \arg\min_{\Phi} L_{ssl}. \tag{23}$$

Since L_{ssl} is minimized, G_V' preserves all the intrinsic features of the ground-truth variant subgraph, i.e.,

$$G_V^* \subseteq G_V'. \tag{24}$$

Meanwhile, because the second variance term L_{var} is minimized across graph label partitions $k = 1, \ldots, |Y|$, only ground-truth variant patterns can be included in G_V', i.e.,

$$G_V' \subseteq G_V^*. \tag{25}$$

Therefore,

$$G_V' = G_V^* \Rightarrow \Phi' = \Phi^*. \tag{26}$$

Thus, we conclude that there exists a unique Φ^* that minimizes the first contrastive loss term of Eq. (5).

□

C. Additional Experimental Details

C.1. Datasets

The datasets are publicly available as follows:

- **CMNIST**: `http://yann.lecun.com/exdb/mnist/` with license unspecified
- **CFashion**: `https://github.com/zalandoresearch/fashion-mnist` with MIT License
- **CKuzushiji**: `https://github.com/rois-codh/kmnist` with CC BY-SA 4.0 License
- **MOLSIDER**: `https://ogb.stanford.edu/docs/graphprop/` with MIT License
- **MOLHIV**: `https://ogb.stanford.edu/docs/graphprop/` with MIT License

C.2. Implementations

We use GCN (Kipf & Welling, 2017) as the backbone GNN model. The hyper-parameter q in Eq. (7) is 0.7. We adopt the Adam optimizer (Kingma & Ba, 2014). Note that we adopt the default hyperparameter settings following (Fan et al., 2022) for a fair comparison. We report the mean values with standard deviations of four repeated experiments.

All the experiments are conducted with:

- Operating System: Ubuntu 18.04.1 LTS
- CPU: Intel(R) Xeon(R) CPU E5-2699 v4@2.20GHz
- GPU: NVIDIA GeForce GTX TITAN Xp with 12GB of Memory
- Software: Python 3.6.5; NumPy 1.19.2; PyTorch 1.10.1; PyTorch Geometric 2.0.3 (Fey & Lenssen, 2019)

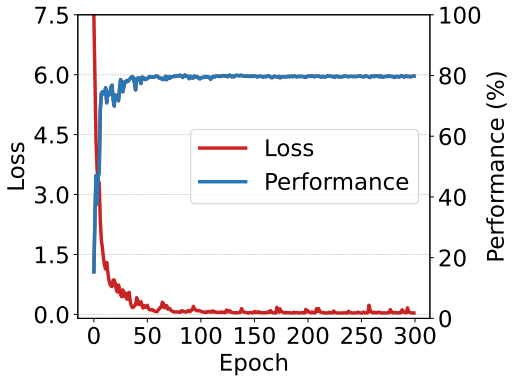


Figure 5. The training dynamics on CMNIST (r = 0.9). We can observe the convergence of our proposed method in practice.

D. Training Dynamics

We present the loss and accuracy during the training process on the CMNIST dataset ($r = 0.9$) in Figure 5. Similar trends are observed across other datasets. The results empirically demonstrate the convergence of our proposed method, with both the loss and accuracy stabilizing well before the maximum training epoch is reached.

E. Limitations

A potential limitation is that we only focus on OOD generalized prediction tasks on static and homogenous graphs, which are most common in the graph learning community. It is worth exploring to extend this work into dynamic and heterogeneous graphs in future.