Expand Horizon: Graph Out-of-Distribution Generalization via Multi-Level Environment Inference

Jiaqiang Zhang[1,2], **Songcan Chen** [1,2] *

[1] College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics
[2] MIIT Key Laboratory of Pattern Analysis and Machine Intelligence
{zhangjq, s.chen}@nuaa.edu.cn

Abstract

Graph neural networks (GNNs) are widely used for node classification tasks, but when encountering distribution shifts due to environmental change in real-world scenarios, they tend to learn unstable correlations between features and labels. To overcome this dilemma, a powerful class of approaches views the environment as the root cause of those unstable correlations, thereby their key focus is to infer the environment involved, enabling the model to avoid capturing environment-sensitive correlations. However, their inferences rely solely on the *single-level* information from *one* low-hop ego-graph, neglecting both global information and multi-granularity information in local ego-graphs with different hops. Although applying deeper GNNs on the high-hop ego-graph could capture global information, it will bring the side effect of over-smoothing node representations. To tackle these issues, we propose a novel Multi-Level Environment Inference model named MLEI, which effectively broadens the horizon of training GNNs under node-level distribution shifts. Specifically, MLEI first leverages a linear graph transformer to surpass the scope of ego-graph, efficiently enabling high-level global environment inference. This global environment is in turn used as an overview to assist layer-by-layer environment inference on local multi-hop ego-graphs. Finally, we combine the environment from global and local views and utilize the designed objective function to capture stable predictive patterns. Extensive experiments on real-world datasets demonstrate that our model achieves satisfactory performance compared with the state-of-the-art methods under various distribution shifts.

Introduction

Graphs are ubiquitous data structures in many fields, such as social networks (Fan et al. 2020) and knowledge graphs (Nickel et al. 2015). In recent years, graph neural networks (GNNs) have emerged as a powerful tool for learning the representations of graph data, in which the core operation is message passing based on structures (Wu et al. 2020). Owing to their exceptional performance, GNNs have been widely applied to a range of graph analysis tasks, including but not limited to node classification (Velickovic et al. 2018) and traffic prediction (Wang, Cao, and Yu 2020).

*Corresponding author.

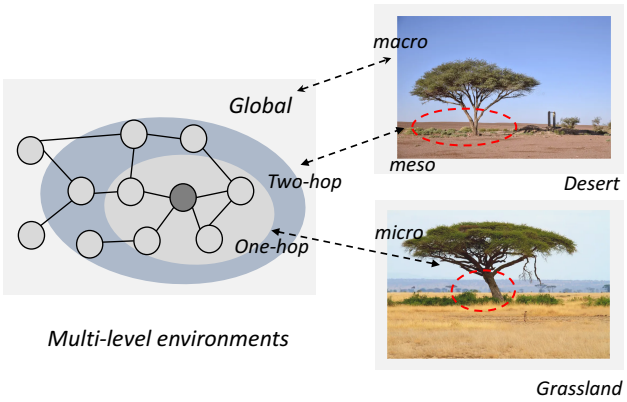

Figure 1: Nodes can be analogized to the trunk in an image, where the multi-hop neighboring information of a node corresponds to the environment at different ranges surrounding the tree, eg. (meso (micro) *vs* two (one)-hop). The global information corresponds to the macro environment where the tree is located, such as a desert or grassland.

Despite their significant advances, most existing GNN models are built on the assumption of *i.i.d.*, i.e., training and testing nodes are drawn from the same distribution (Li et al. 2022). However, in the real world, this assumption can be easily violated due to environmental changes during the data generation (Chen et al. 2022; Fan et al. 2022; Chen et al. 2024). For instance, in the Twitch dataset, users (nodes) and their categories (labels) are heavily influenced by their geographical locations, resulting in distribution shifts among nodes from different regions (Wu et al. 2022; Yu, Liang, and He 2023). In such cases, vanilla GNNs are prone to capture unstable correlations between features and labels, inevitably exhibiting degraded performance on out-of-distribution (OOD) data.

To tackle this OOD problem, there has been a growing interest in viewing the environment as the root cause of unstable dependencies through *causal analysis*. Consequently, their focus is on inferring the environment involved to prevent the model from capturing environment-sensitive correlations (Yuan et al. 2023). The main challenge in doing so mainly lies in the fact that, unlike image data, where datasets

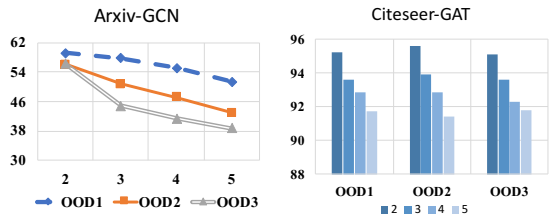

Figure 2: The performance variation of stacking different numbers of GNN layers on the Arxiv and Citeseer datasets.

typically include contextual information as environment labels, in graphs, environment labels for nodes are usually unavailable and laborious to collect (Wu et al. 2024). If we attempt to infer these labels, another challenge emerges: the environment is inherently related to the interconnected nature within nodes (Li et al. 2023), which requires the model to adapt to structural features. As a result, existing models utilize the ego-graphs that can reflect the local structure of each node for modeling. For example, EERM (Wu et al. 2022) and LiSA (Yu, Liang, and He 2023) take node representations from a *fixed*-hop ego-graph for environment-invariant learning, while CaNet (Wu et al. 2024) infers environments through variational inference based on the node representations from the *low*-hop ego-graph. It is evident that their modeled environment is *short-sighted*, neglecting both global information and multi-granularity information in local ego-graphs with different hops. As shown in Fig. 1, we can analogize the node to the trunk in an image, where the multi-hop neighbors (ego-graphs) of a node correspond to environments at different ranges surrounding the tree. It is clear that a node has access to rich contextual information across multiple levels. However, using global-level information, such as the macro-environment (desert, grassland) where the trees are located, as an example, current models generally neglect this aspect. A potential remedy is to employ a high-order ego-graph to model such global relationships, but it suffers from the well-known problem of over-smoothing node representations, potentially leading to degraded performance on test data. To validate this point, we apply deep GNNs with the CaNet model on Arxiv and Citeseer datasets. As shown in Fig. 2, the generalization performance decreases as the number of layers increases.

To alleviate the above drawbacks, we propose a novel Multi-Level Environment Inference model (MLEI) to effectively expand the horizon of training GNNs under node-level distribution shifts. **Firstly**, MLEI employs a linear graph transformer to go beyond the scope of the ego-graph, effectively achieving advanced global environment inference through probabilistic resampling. **Secondly**, this global environment serves as a general environment that assists local relations inference across multi-granularity ego-graphs. These approaches can identify the environments from different perspectives so that we can separate environment-sensitive correlations more accurately. **Finally**, the two complementary (global & local) parts undergo shift-robust learning with an objective function derived from causal analysis. Extensive experiments show that MLEI enjoys satisfactory performance improvements over baselines on six node classification datasets with four types of distribution drift.

Our main contributions are summarized as follows:

- We propose a multi-level learning model for graph OOD generalization, which effectively expands the modeling horizon by utilizing global and local information.
- We perform environment inference with the linear transformer and multi-hop ego graphs, capturing rich information from different granularities.
- We conduct extensive experiments on six benchmark datasets under various distribution shifts. The results verify the effectiveness of our proposal compared with existing state-of-the-art methods.

Related Work

Graph Neural Networks

Graph Neural Networks (GNNs) enable high-quality representation learning for graph-structured data (Zhou et al. 2020; Wu et al. 2020). Existing GNNs are generally categorized into two types: spatial-based and spectral-based (Bo et al. 2021). The former propagates information recursively based on structures, while the latter follows operations in the spectral domain inspired by signal processing. Numerous GNN models and their variants, such as SGC (Wu et al. 2019), APPNP (Gasteiger, Bojchevski, and Günnemann 2018), and LPSL (Han et al. 2023), have been proposed, achieving state-of-the-art performance in tasks such as recommendation systems (Zhang et al. 2023), protein function prediction (Shi et al. 2020), and traffic forecasting (Wang, Cao, and Yu 2020). However, most existing models focus on improving performance on in-distribution data. When encountering data with different distributions during the testing phase, their performance significantly deteriorates (Li et al. 2022; Lu et al. 2024). Therefore, to improve the practical effectiveness of GNNs, more attention needs to be given to handling distribution shifts.

Out-of-Distribution Generalization on Graphs

Out-of-Distribution (OOD) generalization on graphs, which does not rely on the assumption that training and testing data come from the same distribution, is increasingly receiving attention from the research community (Fan et al. 2023; Liu et al. 2023; Yang et al. 2022; Gui et al. 2023). Recent studies (Chen et al. 2023; Yuan et al. 2023; Li et al. 2023) have focused on modeling environments by uncovering the fundamental causes of distribution shifts and integrating them with learning strategies to enhance generalization. However, due to the complex interactions in graphs, the environment labels for nodes are either unavailable or costly to obtain, making it challenging to incorporate them directly. Therefore, some approaches (Yu, Liang, and He 2023; Wu et al. 2022) adopt different graph editing strategies to heuristically generate augmented environments and then combine them with the principles of invariant learning for model training. Another approach (Wu et al. 2024) leverages variational inference to sample environments based on node representations and combines it with causal intervention for stable re-

lationship learning. Nonetheless, these methods largely depend on low-order ego-graphs, which restrict the modeling of multi-level relationships in the data, resulting in suboptimal generalization performance.

Problem Formulation

In this section, we provide necessary terminology and main problem definitions. Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X})$, where $\mathcal{V}$ and $\mathcal{E}$ denote the node set and edge set, $\mathbf{X} = \{\mathbf{x}_v | v \in \mathcal{V}\} \in \mathbb{R}^{N \times D}$ is the feature matrix, N indicates the number of nodes and D is the input feature dimension. Meanwhile, the structure can be defined as an adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$, $\mathbf{A}_{i,j} = 1$ if there is an edge between node $v_i \in \mathcal{V}$ and $v_j \in \mathcal{V}$ and $\mathbf{A}_{i,j} = 0$ otherwise. For each node v, its corresponding one-hot label is denoted as: $\mathbf{y}_v \in \{0, 1\}^C$, where C is the number of class.

Node-Level Predictive Task on Graphs

Given a graph $\mathcal{G}$ with the node features $\mathbf{X}$, the adjacency matrix $\mathbf{A}$, and the labels of the training nodes: $\{\mathbf{y}_v\}_{v \in \mathcal{V}_{tr}}$, where the $\mathcal{V}_{tr}$ is train set, the goal of this node-level predictive task is to predict the node labels in the test set: $\mathcal{V}_{te}$. Here, the distributions of the training and testing nodes can be defined as: $p_{tr}(G, Y)$ and $p_{te}(G, Y)$, where G denotes a random variable of related neighbors for node and Y denotes as a random variable for node labels. Most models rely on the *i.i.d.* assumption, which means that the training and testing node are drawn independently from the same distribution: $p_{tr}(G, Y) = p_{te}(G, Y)$. Meanwhile, they always model G from node's low-order ego-graph. Our model is not limited to this and can be extended to include neighboring information with potential relationships on a global scale.

Out-of-Distribution Generalization on Graphs

In real world, the phenomenon of distribution shift, i.e. $p_{tr}(G, Y) \neq p_{te}(G, Y)$, is widely prevalent. By exploring the shift from the data-generative perspective, there is a general consensus that the fundamental cause of the drift is the environment E. The data generation process can then be further characterized as $P(G, Y|E) = P(G|E)P(Y|G, E)$. Their dependence is illustrated in Fig. 3 (a) by a causal diagram. Unlike image data, which has clear contextual environmental information, the environment label of nodes is unavailable and laborious to obtain. For example, in citation networks, the environmental information of a node is related to its publication time, as well as the properties of nodes connected to it within multi-hop distances. Building upon these foundations, the goal of graph out-of-distribution (OOD) generalization is to eliminate the impact of complex environmental information as much as possible during model training, enabling the derived model to perform well on test data with different distributions.

Methodology

In this section, we provide a detailed introduction to the designed model, which is divided into three main parts. First, for each node, we extend beyond the scope of the ego-graph and apply a linear-complexity transformer to decouple the global relationship, upon which we perform the environment inference. Second, we leverage multi-hop ego-graphs to infer multi-granularity local relations. Finally, under the objective function derived from causal intervention, we conduct model learning with the representations and environments learned from both local and global views. The overall framework is shown in Fig. 3 (b). Before diving into the methodology, we outline the technique of causal intervention (*do*-operation) by using backdoor adjustment.

Causal Treatment with Backdoor Adjustment

In the causal literature (Wu et al. 2024; Pearl, Glymour, and Jewell 2016), the *do*-operation aims to remove the dependence of the target on other variables. Here, as shown in Fig. 3(a), we intend to cancel out the influence of E on G, so that the unstable correlation between the node and labels is no longer captured: $p_\theta(\hat{Y}|do(G))$, where θ is the trainable parameter of predictor, $\hat{Y}$ is a random variable for predicted node labels. The ideal way to achieve $p_\theta(\hat{Y}|do(G))$ is through a randomized controlled trial, which involves conducting controlled experiments across all possible contexts in the physical scenario to eliminate environmental biases, enabling the model to learn stable correlations from G to Y (Wu et al. 2024). However, this operation is difficult to achieve in practice due to limited resources. Therefore, we have adopted the backdoor adjustment (Pearl, Glymour, and Jewell 2016) to implement, as described below :

$$\begin{aligned} p_\theta(\hat{Y}|do(G)) &= \sum_e p_\theta(\hat{Y}|do(G), E = e) p_\theta(E = e|do(G)) \\ &= \sum_e p_\theta(\hat{Y}|G, E = e) p_\theta(E = e|do(G)) \\ &= \sum_e p_\theta(\hat{Y}|G, E = e) p_\theta(E = e), \end{aligned} \tag{1}$$

Since the environmental labels are unknown, we further introduce distributions ($q_\phi(E = e|G)$) to obtain the variational lower bound as our learning objective:

$$\begin{aligned} &\log p_\theta(\hat{Y}|do(G)) = \log \sum_e p_\theta(\hat{Y}|G, E = e) P(E = e) \\ &= \log \sum_e p_\theta(\hat{Y}|G, E = e) p_0(E = e) \frac{q_\phi(E = e|G)}{q_\phi(E = e|G)} \\ &\geq \mathbb{E}_{q_\phi(E|G)}[\log p_\theta(\hat{Y}|G, E)] - KL(q_\phi(E|G) \| p_0(E)), \end{aligned} \tag{2}$$

where these two terms correspond to the prediction and regularization terms respectively, $p_0(E)$ is the prior distribution of environments. Based on this, our efforts can focus on the environment inference: $q_\phi(E|G)$ and the learning of stable predictive pattern: $p_\theta(\hat{Y}|G, E)$. Next, we will implement them from global and local views.

Global Relations Modeling with Transformer

In this part, we introduce the global relations modeling based on the transformer, which includes global representation learning and environment inference. The global attention mechanism in the transformer can capture the implicit

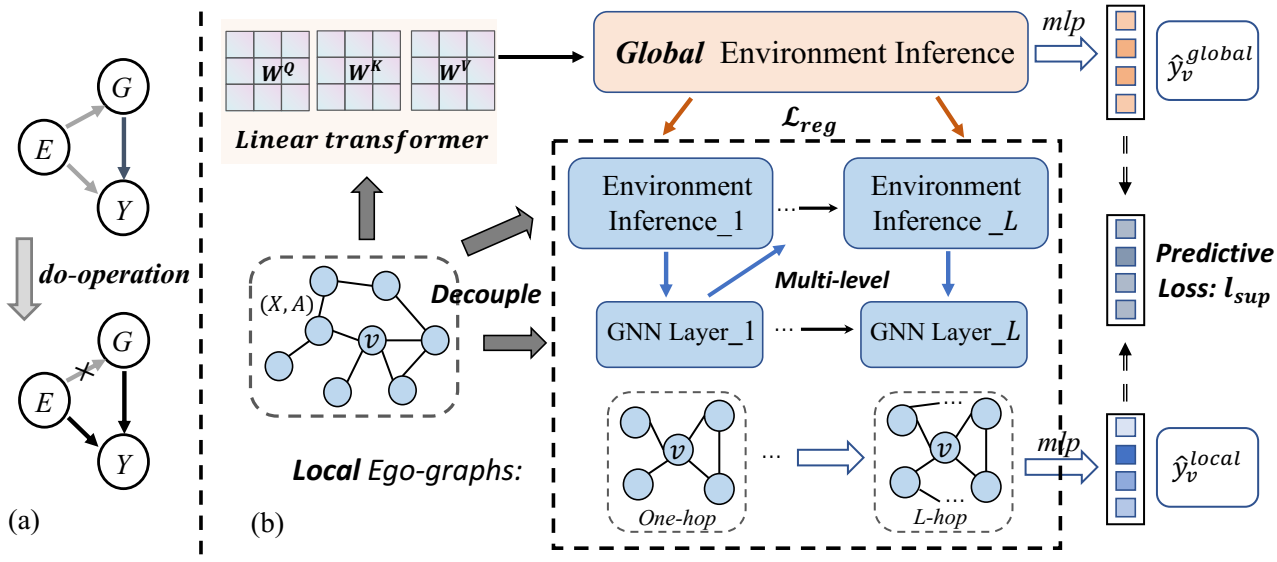

Figure 3: Illustration of the *do*-operation on a causal graph for (a) and the overall framework of MLEI for (b).

inter-dependencies between nodes that are not explicitly expressed by the input structure. However, the time and space complexity of a vanilla transformer typically scales quadratically with the number of nodes, which limits its application on large-scale graphs. To address these issues, building upon the previous work (Wu et al. 2023), we leverage matrix operation rules to achieve linear complexity. The specific operations are described as follows:

Global Represetation Learning First, we use a one-layer MLP $f(\cdot)$ to map the initial node features $\mathbf{X}$ into the latent space $\mathbf{Z}^0 = f(\mathbf{X})$. Next, we compute the linear attention based on these latent representations, i.e., projecting the representations $\mathbf{Z}^0$ into the query space, key space, and value space. The query vectors are used to calculate the attention weights, while the key vectors represent the importance of the inputs. This process is formulated as follows:

$$\begin{aligned} \mathbf{Q} &= f_Q(\mathbf{Z}^0), \quad \tilde{\mathbf{Q}} = \frac{\mathbf{Q}}{\|\mathbf{Q}\|}, \\ \mathbf{K} &= f_K(\mathbf{Z}^0), \tilde{\mathbf{K}} = \frac{\mathbf{K}}{\|\mathbf{K}\|}, \quad \mathbf{V} = f_V(\mathbf{Z}^0), \end{aligned} \tag{3}$$

where f_Q, f_K, f_V are shallow neural networks (one-layer MLP), $||\cdot||$ denotes the the Frobenius norm.

Based on the above representations, we can perform all-pair attentive propagation:

$$\begin{aligned} \mathbf{D} &= diag\left(\mathbf{1} + \frac{1}{N}\tilde{\mathbf{Q}}(\tilde{\mathbf{K}}^\top \mathbf{1})\right), \\ \mathbf{Z}^G &= \mathbf{D}^{-1}\left[\mathbf{V} + \frac{1}{N}\tilde{\mathbf{Q}}(\tilde{\mathbf{K}}^\top \mathbf{V})\right], \end{aligned} \tag{4}$$

where $\mathbf{1}$ is an N-dimensional all-one column vector, $\mathbf{D}$ denotes the attention normalizer, $diag(\cdot)$ is an operation that changes the N-dimensional column vector into a $N \times N$ diagonal matrix, $\mathbf{Z}^G = \{\mathbf{z}_v^g | v \in \mathcal{V}\}$ is the final representation which includes global relations.

Global Environment Inference $q_{\phi 1}^g(E|G)$ For each node v, we assume $\mathbf{e}_v^g \in \mathbb{R}^K$ as its inferred environment, which can be sampled from a categorical distribution $\mathcal{S}(\boldsymbol{\pi}_v^g)$. Here, the probabilities $\boldsymbol{\pi}_v^g$ are modeled by the global node representation $\mathbf{z}_v^g$:

$$\boldsymbol{\pi}_v^g = \text{Softmax}(\mathbf{W}_g \mathbf{z}_v^g), \tag{5}$$

where $\mathbf{W}_g \in \mathbb{R}^{H_g \times K}$ is a trainable weight matrix, H_g is the hidden dimension of transformer. Since the sampling process is non-differentiable and hinders back-propagation in the training stage, we modify this process with the reparametrization trick (Maddison, Mnih, and Teh 2017) :

$$e_{vk}^{(g)} = \frac{\exp\left(\left(\pi_{vk}^g + t_k^g\right)/\tau\right)}{\sum_k \exp((\pi_{vk}^g + t_k^g)/\tau)}, \quad t_k^g \sim \text{Gumbel}(0,1), \tag{6}$$

where t_k^g is i.i.d sampled from Gumbel distribution and τ is a temperature coefficient. Based on the above operations, for each node v, we can further formalize the prediction $p_{\theta_1}^{global}(\hat{Y}|G,E)$:

$$\hat{\mathbf{y}}_v^{global} = f_P\left(\sigma\left(\mathbf{W}_T \sum_{k=1}^{K} e_{v,k}^{(g)} \mathbf{z}_v^g + \mathbf{b}\right)\right), \tag{7}$$

where σ denotes the activation function, here we use the ReLU. $\mathbf{W}_T$ and $\mathbf{b}$ are the trainable parameter matrix and bias term respectively, f_P is a one-layer MLP for predicting labels, which will be shared with the local view. Finally, the loss function in this global view can be further defined as:

$$\begin{aligned} &Loss^{global} = \\ &\frac{-1}{|\mathcal{V}_{tr}|} \sum_{v \in \mathcal{V}_{tr}} [\mathbf{y}_v^\top \log \hat{\mathbf{y}}_v^{global} - \sum_{k=1}^{K} [e_{vk}^{(g)} \log \pi_{vk}^g + e_{vk}^{(g)} \log K]], \end{aligned} \tag{8}$$

where these two items correspond to the prediction term and regularization term in the function (2) respectively.

Local Relations Modeling with Ego-Graphs

In this part, we introduce the local relations modeling with multi-hop ego-graphs. Since the contextual information under different receptive fields is different, as shown in Fig.3, we model the ego-graph of different hops *layer by layer* to capture this multi-granularity information. Specifically, for the l-th layer, similar to the process on the global level, local environment inference $q^l_{\phi 2}(E|G)$ can be defined as follows (the difference is that the node representation $\mathbf{z}^{(l)}_v$ comes from the feature transformation of the l-hop ego-graph by GNNs) :

$$\boldsymbol{\pi}^{(l)}_v = \text{Softmax}(\mathbf{W}_l \mathbf{z}^{(l)}_v), \tag{9}$$

$$e^{(l)}_{vk} = \frac{\exp\left(\left(\pi^{(l)}_{vk} + t^l_k\right)/\tau\right)}{\sum_k \exp((\pi^{(l)}_{vk} + t^l_k)/\tau)}, \quad t^l_k \sim \text{Gumbel}(0,1), \tag{10}$$

where $\mathbf{W}_l$ is the trainable parameter matrix at the l-layer. Furthermore, the node representation combined with the local environment information can be defined as:

$$\mathbf{z}^{(l+1)}_v = \sigma\left(\sum_{k=1}^{K} e^{(l)}_{vk} \sum_{u,\mathbf{A}_{v,u}=1} \frac{1}{\sqrt{d_u d_v}} \mathbf{W}^{(l,k)}_D \mathbf{z}^{(l)}_u + \mathbf{W}^{(l,k)}_S \mathbf{z}^{(l)}_v\right), \tag{11}$$

where $\mathbf{W}^{(l,k)}_S$ and $\mathbf{W}^{(l,k)}_D$ are trainable weight matrices, d_u denotes the degree of node u.

Multi-Level Local & Global Information Fusion

In this section, we introduce the fusion of global and local information. First, for each layer, after capturing information from the ego-graph, we integrate the global environment as a general outline to further enhance the modeling of local relationships. The specific process is defined as follows:

$$\mathbf{z}^{(l_g)}_v = \sigma\left(\sum_{k=1}^{K} e^{(g)}_{vk} \sum_{u,\mathbf{A}_{v,u}=1} \frac{1}{\sqrt{d_u d_v}} \mathbf{W}^{(l,k)}_D \mathbf{z}^{(l)}_u + \mathbf{W}^{(l,k)}_S \mathbf{z}^{(l)}_v\right) \tag{12}$$

$$\mathbf{z}^{local\_l}_v = \frac{\mathbf{z}^{(l+1)}_v + \mathbf{z}^{(l_g)}_v}{2}, \tag{13}$$

Therefore, for this local view, after L-layer propogation, we can obtain the label prediction $p^{local}_{\theta_2}(\hat{Y}|G,E)$:

$$\hat{\mathbf{y}}^{local}_v = f_P(z^{local\_L}_v), \tag{14}$$

where f_P is a one-layer MLP for predicting labels. At this time, the loss function in the local view can be defined as:

$$Loss^{local} = \frac{-1}{|\mathcal{V}_{tr}|} \sum_{v \in \mathcal{V}_{tr}} [\mathbf{y}^\top_v \log \hat{\mathbf{y}}^{local}_v - \frac{1}{L} \sum_{l=1}^{L} \sum_{k=1}^{K} [e^{(l)}_{vk} \log \pi^{(l)}_{vk} + e^{(l)}_{vk} \log K]] \tag{15}$$

Finally, the overall loss function can be further rewritten as:

$$Loss = Loss^{local} + \lambda Loss^{global}, \tag{16}$$

where λ is a tuneable hyper-parameter.

Experiments

Datasets

We evaluate our proposed model on six datasets with different scales and properties, including Cora, Citeseer, Pubmed (Sen et al. 2008), Twitch (Rozemberczki and Sarkar 2021), Arxiv (Hu et al. 2020), and Elliptic (Pareja et al. 2020). The statistics of these datasets is provided in Table 2. Following the previous work (Wu et al. 2024), the construction of ID and OOD data for each dataset is as follows. For Cora, Citeseer, and Pubmed, we utilize randomly initialized GNNs to synthesize spurious features to introduce OOD data. For Arxiv, papers published between 2005 and 2014 are treated as ID data, while those published after 2014 are considered OOD data. For Twitch, we use the nodes from subgraphs DE, PT, and RU as ID data, and the nodes from ES, FR, and EN as OOD data. For Elliptic, we use the first five graph snapshots as ID data, and the remaining snapshots are divided into eight groups of the same size as OOD data.

Baselines and Experimental Settings

We mainly compare two categories of models. One category is designed for OOD generalization on general data, including IRM (Arjovsky et al. 2019), DeepCoral (Sun and Saenko 2016), DANN (Ganin et al. 2016), GroupDRO (Sagawa et al. 2019), and Mixup (Zhang et al. 2017). The other category is for OOD generalization specifically for learning on graphs, including the state-of-the-art models: SR-GNN (Zhu et al. 2021), EERM (Wu et al. 2022), and CANeT (Wu et al. 2024), with the latter two also focusing on environment modeling in graph data. Furthermore, for these models, we adopt GCN and GAT as backbone networks respectively for a comprehensive comparison. For each dataset, we randomly split the ID data into 50%/25%/25% proportions for training, validation, and testing. Following existing work, we use accuracy to evaluate performance on Cora, Citeseer, Pubmed, and Arxiv, and use ROC-AUC and macro F1 score as metrics for Twitch and Elliptic, respectively. Additionally, we conduct five trials with different initializations.

Result and Analysis

We report the test performance of each dataset in Table 1, Table 3, and Figure 4. It can be observed that regardless of whether GCN or GAT is used as the backbone network, MLGI achieves almost the best performance across different distribution shifts, showcasing its superiority in handling various complex real-world testing scenarios. Specifically, the accuracy on the Pubmed dataset is 3.79% higher than the state-of-the-art method CaNet. Meanwhile, our method exhibits a relatively low standard deviation on Cora, Citeseer, and Pubmed, which to some extent indicates its stability. For testing on the dynamic graph Elliptic, we split these test snapshots into eight equal-sized subsets in chronological order. Our method performs optimally on all OOD data, with the highest improvement being 4.42% on OOD4 with a GCN backbone, further proving its capability to generalize to unseen graphs. Finally, the impressive performance on the Twitch and the large graph dataset Arxiv highlight the method's efficiency in handling complex distribution shifts.

Backbone	Method	Cora		Citeseer		Pubmed	
		OOD	ID	OOD	ID	OOD	ID
GCN	ERM	74.30 ± 2.66	94.83 ± 0.25	74.93 ± 2.39	85.76 ± 0.26	81.36 ± 1.78	92.76 ± 0.10
	IRM	74.19 ± 2.60	94.88 ± 0.18	75.34 ± 1.61	85.34 ± 0.46	81.14 ± 1.72	92.80 ± 0.12
	Coral	74.26 ± 2.28	94.89 ± 0.18	74.97 ± 2.53	85.64 ± 0.28	81.56 ± 2.35	92.78 ± 0.11
	DANN	73.09 ± 3.24	95.03 ± 0.16	74.74 ± 2.78	85.75 ± 0.49	80.77 ± 1.43	93.20 ± 0.42
	GroupDRO	74.25 ± 2.61	94.87 ± 0.25	75.02 ± 2.05	85.33 ± 0.36	81.07 ± 1.89	92.76 ± 0.08
	Mixup	92.77 ± 1.27	94.84 ± 0.30	77.28 ± 5.28	85.00 ± 0.50	79.76 ± 4.44	92.68 ± 0.13
	SRGNN	81.91 ± 2.64	95.09 ± 0.32	76.10 ± 4.04	85.84 ± 0.37	84.75 ± 2.38	93.52 ± 0.31
	EERM	83.00 ± 0.77	89.17 ± 0.23	74.76 ± 1.15	83.81 ± 0.17	OOM	OOM
	CaNeT	95.60 ± 0.93	97.96 ± 0.17	93.05 ± 0.91	95.18 ± 0.24	88.05 ± 1.09	97.47 ± 0.08
	Our	**97.32 ± 0.29**	97.85 ± 0.10	**94.61 ± 0.74**	95.27 ± 0.15	**91.84 ± 0.59**	97.16 ± 0.25
GAT	ERM	91.10 ± 2.26	95.57 ± 0.40	82.60 ± 0.51	89.02 ± 0.32	84.80 ± 1.47	93.98 ± 0.24
	IRM	91.63 ± 1.27	95.72 ± 0.31	82.73 ± 0.37	89.11 ± 0.36	84.95 ± 1.06	93.89 ± 0.26
	Coral	91.82 ± 1.30	95.74 ± 0.39	82.44 ± 0.58	89.05 ± 0.37	85.07 ± 0.95	94.05 ± 0.23
	DANN	92.40 ± 2.05	95.66 ± 0.28	82.49 ± 0.67	89.02 ± 0.31	83.94 ± 0.84	93.46 ± 0.31
	GroupDRO	90.54 ± 0.94	95.38 ± 0.23	82.64 ± 0.61	89.13 ± 0.27	85.17 ± 0.86	94.00 ± 0.18
	Mixup	92.94 ± 1.21	94.66 ± 0.10	82.77 ± 0.30	89.05 ± 0.05	81.58 ± 0.65	92.79 ± 0.18
	SRGNN	91.77 ± 2.43	95.36 ± 0.24	82.72 ± 0.35	89.10 ± 0.15	83.40 ± 0.67	93.21 ± 0.29
	EERM	91.80 ± 0.73	91.37 ± 0.30	74.07 ± 0.75	83.53 ± 0.56	OOM	OOM
	CaNeT	97.52 ± 0.12	96.07 ± 0.44	95.33 ± 0.22	89.12 ± 0.52	89.50 ± 1.07	95.03 ± 0.26
	Our	**97.92 ± 0.27**	97.44 ± 0.08	**96.78 ± 0.30**	94.51 ± 0.28	**91.23 ± 0.77**	97.16 ± 0.16

Table 1: Test Accuracy ↑ (mean±standard deviation) for citation networks on out-of-distribution (OOD) and in-distribution (ID) data. OOM indicates an out-of-memory error on a GPU with 24GB memory.

Datasets	#Nodes	#Edges	#Classes	#Features
Cora	2708	5429	7	1433
Citeseer	3327	4732	6	3703
Pubmed	19717	44338	3	500
Twitch	34120	892346	2	2545
Arxiv	169343	1166243	40	128
Elliptic	203769	234355	2	165

Table 2: Statistics for experimental datasets.

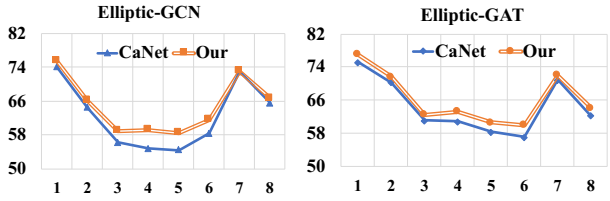

Figure 4: Macro F1 score on eight testing sets (by chronologically grouping the testing snapshots) of Elliptic.

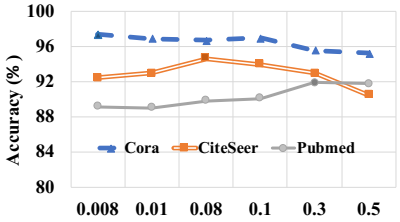

Figure 5: Model performance with different λ.

Ablation Study

In this section, we conduct ablation experiments to verify the effectiveness of each component. Among them, we remove the regularization term, global relationship modeling, local relationship modeling, and environment inference on multi-hop ego-graphs (i.e., leave global inference and one inference across all layers in the local view). The results, shown in Figure 6, illustrate that each component has a certain degree of influence on the model, varying with the degree of distribution drift in the data. The relationship modeling at different levels has a positive impact on out-of-distribution generalization. Specifically, on the Arxiv dataset, as the time gap between training and test data becomes larger, the impact of each part becomes more pronounced, which indirectly reflects the importance of multi-level environmental inference under regularization constraints.

Hyper-parameter Study

In this section, we study the impact of three hyper-parameters, including the number of pseudo environments K, the trade-off factor λ, and the temperature coefficient τ in the Gumbel-Softmax. The results are presented in Figures 5, 7, and 8. Regarding K, we select values from 1 to 5 for the experiment. It is observed that in most cases on both datasets, the optimal value is obtained at K=3. Notably, the impacts of OOD2 and OOD3 on Arxiv are relatively significant, which correlates with the degree of distribution shift between OOD data and training data. Regarding the trade-

Backbone	Method	Arxiv				Twitch			
		OOD 1	OOD 2	OOD 3	ID	OOD 1	OOD 2	OOD 3	ID
GCN	ERM	56.33 ± 0.17	53.53 ± 0.44	45.83 ± 0.47	59.94 ± 0.45	66.07 ± 0.14	52.62 ± 0.01	63.15 ± 0.08	75.40 ± 0.01
	IRM	55.92 ± 0.24	53.25 ± 0.49	45.66 ± 0.83	60.28 ± 0.23	66.95 ± 0.27	52.53 ± 0.02	62.91 ± 0.08	74.88 ± 0.02
	Coral	56.42 ± 0.26	53.53 ± 0.54	45.92 ± 0.52	60.16 ± 0.12	66.15 ± 0.14	52.67 ± 0.02	63.18 ± 0.03	75.40 ± 0.01
	DANN	56.35 ± 0.11	53.81 ± 0.33	45.89 ± 0.37	60.22 ± 0.29	66.15 ± 0.13	52.66 ± 0.02	63.20 ± 0.06	75.40 ± 0.02
	GroupDRO	56.52 ± 0.27	53.40 ± 0.29	45.76 ± 0.59	60.35 ± 0.27	66.82 ± 0.26	52.69 ± 0.02	62.95 ± 0.11	75.03 ± 0.01
	Mixup	56.67 ± 0.46	54.02 ± 0.51	46.09 ± 0.58	60.09 ± 0.15	65.76 ± 0.30	52.78 ± 0.04	63.15 ± 0.08	75.47 ± 0.06
	SRGNN	56.79 ± 1.35	54.33± 1.78	46.24 ± 1.90	60.02 ± 0.52	65.83 ± 0.45	52.47 ± 0.06	62.74 ± 0.23	75.75 ± 0.09
	EERM	OOM	OOM	OOM	OOM	67.50 ± 0.74	51.88 ± 0.07	62.56 ± 0.02	74.85 ± 0.05
	CaNeT	59.01 ± 0.30	56.88 ± 0.70	56.27 ± 1.21	61.42 ± 0.10	67.47 ± 0.32	53.59 ± 0.19	64.24 ± 0.18	75.10 ± 0.08
	Our	**60.03** ± **0.48**	**57.47** ± **0.51**	**57.64** ± **0.87**	60.92 ± 0.44	**68.59** ± **0.46**	**53.77** ± **0.18**	**64.59** ± **0.26**	75.36 ± 0.24
GAT	ERM	57.15 ± 0.25	55.07 ± 0.58	46.22 ± 0.82	59.72 ± 0.35	65.67 ± 0.02	52.00 ± 0.10	61.85 ± 0.05	75.75 ± 0.15
	IRM	56.55 ± 0.18	54.53 ± 0.32	46.01 ± 0.33	59.94 ± 0.18	67.27 ± 0.19	52.85 ± 0.15	62.40 ± 0.24	75.30 ± 0.09
	Coral	57.40 ± 0.51	55.14 ± 0.71	46.71 ± 0.61	60.59 ± 0.30	67.12 ± 0.03	52.61 ± 0.01	63.41 ± 0.01	75.20 ± 0.01
	DANN	57.23 ± 0.18	55.13 ± 0.46	46.61 ± 0.57	59.72 ± 0.14	66.59 ± 0.38	52.88 ± 0.12	62.47 ± 0.32	75.82 ± 0.27
	GroupDRO	56.69 ± 0.27	54.51 ± 0.49	46.00 ± 0.59	60.03 ± 0.32	67.41 ± 0.04	52.99 ± 0.08	62.29 ± 0.03	75.74 ± 0.02
	Mixup	57.17 ± 0.33	55.33 ± 0.37	47.17 ± 0.84	59.84 ± 0.50	65.58 ± 0.13	52.04 ± 0.04	61.75 ± 0.13	75.72 ± 0.07
	SRGNN	56.69 ± 0.38	55.01 ± 0.55	46.88 ± 0.58	59.39 ± 0.17	66.17 ± 0.03	52.84 ± 0.04	62.07 ± 0.04	75.45 ± 0.03
	EERM	OOM	OOM	OOM	OOM	66.80 ± 0.46	52.39 ± 0.20	62.07 ± 0.68	75.19 ± 0.50
	CaNeT	60.44 ± 0.27	58.54 ± 0.72	59.61 ± 0.28	62.91 ± 0.35	**68.08** ± **0.19**	53.49 ± 0.14	63.76 ± 0.17	76.14 ± 0.07
	Our	**61.03** ± **0.20**	**59.73** ± **0.23**	**60.28** ± **0.76**	62.37 ± 0.34	67.93 ± 0.19	**54.61** ± **0.15**	**64.69** ± **0.08**	75.98 ± 0.20

Table 3: Test Accuracy ↑ (mean±standard deviation) for Arxiv and ROC-AUC ↑ for Twitch on different subsets of out-of-distribution data. For data splits, publication years and subgraphs are used on Arxiv and Twitch.

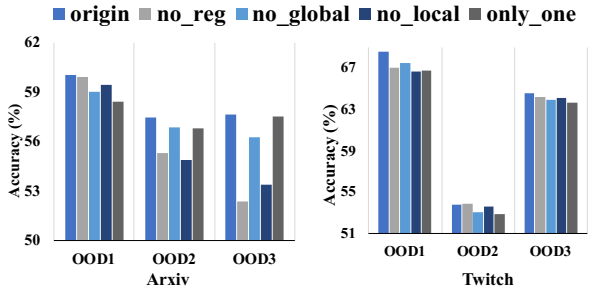

Figure 6: Ablation studies on Arxiv and Twitch.

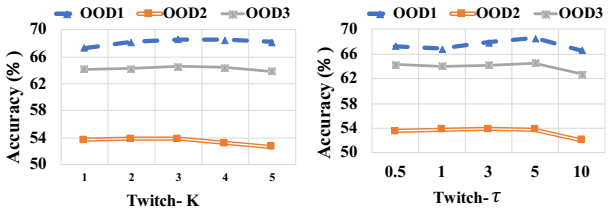

Figure 7: Performance with different K and τ on Twitch.

off factor λ , Cora, Citeseer, and Pubmed achieve the best value at 0.008, 0.08, and 0.3, respectively. This parameter appears to vary with the dataset size. In a dataset with a larger number of nodes, the richer the global relationship, the greater the weight assigned to it, which aligns with intuition. Regarding τ, in most cases, the best performance is achieved when τ=1. A too-large τ value tends to result in

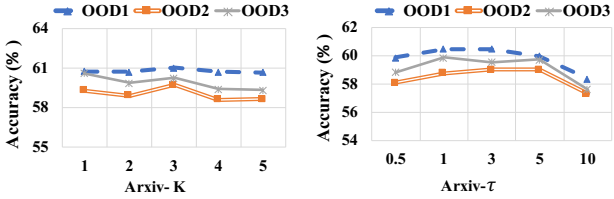

Figure 8: Performance with different K and τ on Arxiv.

over-smoothed outputs, collapsing to a uniform distribution and diminishing the information content.

Conclusion

In this paper, we explore the node-level out-of-distribution generalization by proposing a multi-level environment inference model, named MLEI. In contrast to existing models that focus solely on the local ego-graph, MLEI can effectively capture the multi-granularity structural information. Within these rich environments, MLEI integrates an objective function derived from causal analysis, enabling the model to identify environment-insensitive predictive patterns. We validate the effectiveness of MLEI through extensive experiments across datasets of varying scales and shifts.

Acknowledgments

This work was Supported by the National Natural Science Foundation of China (Grant No. 62376126).

References

Arjovsky, M.; Bottou, L.; Gulrajani, I.; and Lopez-Paz, D. 2019. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*.

Bo, D.; Wang, X.; Shi, C.; and Shen, H. 2021. Beyond low-frequency information in graph convolutional networks. In *AAAI*, volume 35, 3950–3957.

Chen, Y.; Bian, Y.; Zhou, K.; Xie, B.; Han, B.; and Cheng, J. 2023. Does invariant graph learning via environment augmentation learn invariance? *Advances in Neural Information Processing Systems*, 36.

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.

Chen, Z.; Xiao, T.; Kuang, K.; Lv, Z.; Zhang, M.; Yang, J.; Lu, C.; Yang, H.; and Wu, F. 2024. Learning to Reweight for Generalizable Graph Neural Network. In *AAAI*, volume 38, 8320–8328.

Fan, S.; Wang, X.; Mo, Y.; Shi, C.; and Tang, J. 2022. Debiasing graph neural networks via learning disentangled causal substructure. *Advances in Neural Information Processing Systems*, 35: 24934–24946.

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*.

Fan, W.; Ma, Y.; Li, Q.; Wang, J.; Cai, G.; Tang, J.; and Yin, D. 2020. A graph neural network framework for social recommendations. *IEEE Transactions on Knowledge and Data Engineering*, 34(5): 2033–2047.

Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; March, M.; and Lempitsky, V. 2016. Domain-adversarial training of neural networks. *Journal of machine learning research*, 17(59): 1–35.

Gasteiger, J.; Bojchevski, A.; and Günnemann, S. 2018. Predict then propagate: Graph neural networks meet personalized pagerank. *arXiv preprint arXiv:1810.05997*.

Gui, S.; Liu, M.; Li, X.; Luo, Y.; and Ji, S. 2023. Joint learning of label and environment causal independence for graph out-of-distribution generalization. *Advances in Neural Information Processing Systems*, 36.

Han, H.; Liu, X.; Shi, F.; Torkamani, M.; Aggarwal, C.; and Tang, J. 2023. Towards Label Position Bias in Graph Neural Networks. *Advances in Neural Information Processing Systems*, 36.

Hu, W.; Fey, M.; Zitnik, M.; Dong, Y.; Ren, H.; Liu, B.; Catasta, M.; and Leskovec, J. 2020. Open graph benchmark: Datasets for machine learning on graphs. *Advances in Neural Information Processing Systems*, 33: 22118–22133.

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

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

Liu, S.; Li, T.; Feng, Y.; Tran, N.; Zhao, H.; Qiu, Q.; and Li, P. 2023. Structural re-weighting improves graph domain adaptation. In *ICML*, 21778–21793. PMLR.

Lu, B.; Zhao, Z.; Gan, X.; Liang, S.; Fu, L.; Wang, X.; and Zhou, C. 2024. Graph out-of-distribution generalization with controllable data augmentation. *IEEE Transactions on Knowledge and Data Engineering*.

Maddison, C. J.; Mnih, A.; and Teh, Y. W. 2017. The concrete distribution: A continuous relaxation of discrete random variables. *ICLR*.

Nickel, M.; Murphy, K.; Tresp, V.; and Gabrilovich, E. 2015. A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, 104(1): 11–33.

Pareja, A.; Domeniconi, G.; Chen, J.; Ma, T.; Suzumura, T.; Kanezashi, H.; Kaler, T.; Schardl, T.; and Leiserson, C. 2020. Evolvegcn: Evolving graph convolutional networks for dynamic graphs. In *AAAI*, volume 34, 5363–5370.

Pearl, J.; Glymour, M.; and Jewell, N. P. 2016. *Causal inference in statistics: A primer*. John Wiley & Sons.

Rozemberczki, B.; and Sarkar, R. 2021. Twitch gamers: a dataset for evaluating proximity preserving and structural role-based node embeddings. *arXiv preprint arXiv:2101.03091*.

Sagawa, S.; Koh, P. W.; Hashimoto, T. B.; and Liang, P. 2019. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. *arXiv preprint arXiv:1911.08731*.

Sen, P.; Namata, G.; Bilgic, M.; Getoor, L.; Galligher, B.; and Eliassi-Rad, T. 2008. Collective classification in network data. *AI magazine*, 29(3): 93–93.

Shi, C.; Xu, M.; Zhu, Z.; Zhang, W.; Zhang, M.; and Tang, J. 2020. Graphaf: a flow-based autoregressive model for molecular graph generation. *ICLR*.

Sun, B.; and Saenko, K. 2016. Deep coral: Correlation alignment for deep domain adaptation. In *Computer Vision–ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part III 14*, 443–450. Springer.

Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2018. Graph Attention Networks. In *ICLR*.

Wang, S.; Cao, J.; and Yu, P. 2020. Deep learning for spatio-temporal data mining: A survey. *IEEE transactions on knowledge and data engineering*, 34(8): 3681–3700.

Wu, F.; Souza, A.; Zhang, T.; Fifty, C.; Yu, T.; and Weinberger, K. 2019. Simplifying graph convolutional networks. In *ICML*, 6861–6871. PMLR.

Wu, Q.; Nie, F.; Yang, C.; Bao, T.; and Yan, J. 2024. Graph Out-of-Distribution Generalization via Causal Intervention. In *WWW*, 850–860.

Wu, Q.; Zhang, H.; Yan, J.; and Wipf, D. 2022. Handling distribution shifts on graphs: An invariance perspective. *ICLR*.

Wu, Q.; Zhao, W.; Yang, C.; Zhang, H.; Nie, F.; Jiang, H.; Bian, Y.; and Yan, J. 2023. Simplifying and empowering transformers for large-graph representations. *Advances in Neural Information Processing Systems*, 36.

Wu, Z.; Pan, S.; Chen, F.; Long, G.; Zhang, C.; and Philip, S. Y. 2020. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, 32(1): 4–24.

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

Yu, J.; Liang, J.; and He, R. 2023. Mind the label shift of augmentation-based graph ood generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11620–11630.

Yuan, H.; Sun, Q.; Fu, X.; Zhang, Z.; Ji, C.; Peng, H.; and Li, J. 2023. Environment-Aware Dynamic Graph Learning for Out-of-Distribution Generalization. *Advances in Neural Information Processing Systems*, 36.

Zhang, D.; Zhu, Y.; Dong, Y.; Wang, Y.; Feng, W.; Kharlamov, E.; and Tang, J. 2023. ApeGNN: node-wise adaptive aggregation in GNNs for recommendation. In *WWW*, 759–769.

Zhang, H.; Cisse, M.; Dauphin, Y. N.; and Lopez-Paz, D. 2017. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*.

Zhou, J.; Cui, G.; Hu, S.; Zhang, Z.; Yang, C.; Liu, Z.; Wang, L.; Li, C.; and Sun, M. 2020. Graph neural networks: A review of methods and applications. *AI open*, 1: 57–81.

Zhu, Q.; Ponomareva, N.; Han, J.; and Perozzi, B. 2021. Shift-robust gnns: Overcoming the limitations of localized graph training data. *Advances in Neural Information Processing Systems*, 34: 27965–27977.