ACLIB-GNN:INCORPORATING ADVERSARIAL CAUSAL LEARNING WITH INFORMATION BOTTLENECKS FOR INTERPRETABLE GRAPH NEURAL NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Graph Neural Networks (GNNs) excel in node classification but face critical interpretability challenges.Though existing explanation methods that include post-hoc and self-interpretable approaches are widely adopted, they still struggle to enhance prediction through explanation effectively.Moreover, causal graph learning demonstrates the capacity to identify causal features that bolster predictive performance, but its utilization in node classification tasks has remained notably limitedprimarily due to the non-trivial challenges of handling localized heterogeneity and contextual noise in node-level tasks.To address these gaps, we propose ACLIB-GNN, a novel framework unifying adversarial causal learning and the node information bottleneck.By leveraging graph attention to minimize noncausal feature interference and adversarial training to maximize mutual information between explanatory subgraphs and labels, it explicitly disentangles causal features from shortcut signals, balancing transparency and performance. On four benchmark datasets, ACLIB-GNN outperforms state-of-the-art baselines via causal subgraphs to enhance classification accuracy and provides superior explanatory power.Ablation studies validate the synergistic effect of its core components. Notably, the framework generalizes graph classification tasks effectively. ACLIB-GNN offers a scalable and trustworthy solution for interpretable node classification tasks based on causal graph learning.

1 INTRODUCTION

In recent years, Graph Neural Networks (GNNs) have emerged as a powerful framework for representation learning in graph-structured data (1; 2; 3). Due to their ability to capture complex relational patterns, GNNs have been successfully applied to diverse domains, including molecular chemistry (4), social network analysis (5), and knowledge graph reasoning (6).Typical applications span node-level, edge-level, and graph-level tasks, with all three types widely explored due to their broad real-world applicability (7). However, most existing GNNs operate as black-box models, lacking transparency in their decision-making processes. To enhance their reliability and adoption in critical domains, there is a pressing need to develop interpretable GNNs that achieve high predictive accuracy while providing explainable reasoning behind their predictions.

Recent research has focused on improving the interpretability of GNNs, primarily through two approaches: post-hoc and self-interpretable methods (8). Post-hoc methods use external tools to analyze model predictions based on input features after training. Although they offer useful insights, they may not fully reflect the model's internal logic (9; 10; 11; 12). In contrast, self-interpretable methods integrate explainability directly into the model architecture, leveraging causal intervention theory to mitigate environmental confounders and identify meaningful subgraphs. Despite their advantages, these models often struggle to promote prediction through explanation (11).

Recent research has focused on improving the interpretability of GNNs, primarily through two approaches: post-hoc and self-interpretable methods (8). Post-hoc methods use external tools to analyze model predictions based on input features after training. Although they offer useful insights, they may not fully reflect the model's internal logic (9; 10; 11; 12). In contrast, self-interpretable methods integrate explainability directly into the model architecture, leveraging causal intervention

theory to mitigate environmental confounders and identify meaningful subgraphs. Despite their advantages, these models often struggle to promote prediction through explanation (11).

Causal graph learning can identify causal features via Graph Information Bottleneck (GIB) (13) to enhance predictions. However, most existing approaches focus on graph classification tasks (14). Although some extend to node classification, they fail to provide explanations, creating a critical gap in interpretable causal graph learning. This gap is primarily due to two core challenges in applying GIB to node classification. First, the local neighborhood of a node is often flooded with noisy and shortcut connections, which can easily mislead a standard GIB model that primarily focuses on compression and prediction. Second, existing GIB methods lack an explicit mechanism to enforce the stability of explanations under distribution shifts or perturbations, making them vulnerable to spurious correlations in the graph structure. To address these limitations, we incorporate adversarial causal learning into the GIB framework. Adversarial training strengthens the model by encouraging it to learn causal features that are invariant to malicious perturbations on the explanatory subgraphs. This ensures that the identified subgraphs are not only informative but also robust and truly causal, thereby effectively disentangling them from shortcut signals.

To address GNN interpretability and performance limitations, ACLIB-GNN is proposed as a framework unifying adversarial causal learning with the graph information bottleneck. It includes two components: a redundancy reduction mechanism to filter extraneous representation information and an adversarial causal graph learning module to maximize mutual information between explanatory subgraphs and labels. Adversarial learning strengthens self-interpretable GNNs by penalizing predictions when critical explanatory components are perturbed, ensuring that models prioritize essential features that drive both accuracy and interpretability (15). Extensive experiments were conducted to evaluate ACLIB-GNN's effectiveness in node classification and explanation tasks. The framework demonstrated superior performance across four benchmark datasets, with the generated explanations simultaneously enhancing classification accuracy. Visualization analyses confirmed these improvements through examination of both extracted subgraphs and learned representations. Collectively, these results demonstrate that ACLIB-GNN effectively improves both GNN interpretability and classification performance in node classification tasks. Our contributions are summarized as:

1. To enable scalable and trustworthy GNN node classification with interpretable predictions, ACLIB-GNN is proposed based on causal graph learning, integrating adversarial learning and graph information bottlenecks.
2. Interpretability and prediction mutually reinforce each other via causal subgraphs, effectively disentangling causal features from shortcut signals.
3. Extensive experiments show that the proposed framework outperforms state-of-the-art methods and generalizes to graph classification tasks.

2 Related Works

2.1 Explanations for GNNs

The explainability of GNNs is primarily addressed through two paradigms: post-hoc explainability methods and self-interpretable approaches. Post-hoc methods use external tools to analyze model predictions after training.They can be divided into white-box and black-box categories.White-box approaches require model internals, employing decomposition-based techniques such as CAM (16) and GNN-LRP (17) for layer-wise score attribution to nodes oredges. Additionally, gradient-based methods like SA and Grad-CAM (18) are employed to quantify input sensitivity.Nevertheless, their architecture-specificity limits generalizability and incurs higher computational costs. Instead, black-box methods do not require access to the model's internal parameters. For example, GraphLime (19) and PGMExplainer (20) approximate local behaviors using interpretable models like HSIC Lasso.GNNExplainer (21) identifies critical subgraphs via node/edge masking, and XGNN (22) synthesizes explanatory subgraphs with reinforcement learning. Moreover, the black-box characteristic of these explainers casts uncertainty on their capacity to offer thorough interpretability for GNN models. In contrast, Self-interpretable methods embed explainability directly into GNN architectures through joint training of prediction and explanation modules, primarily categorizing into

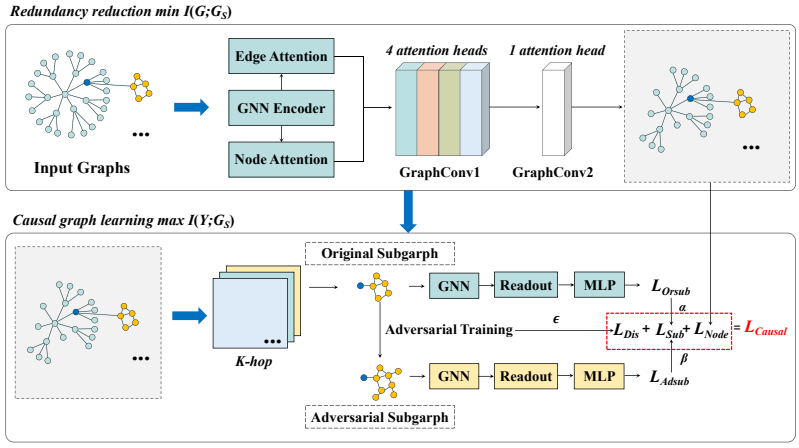

Figure 1: The overall framework of ACLIB-GNN, comprising two key components: (1) Redundancy Reduction via GATs to minimize mutual information between nodes and the original graph; (2) Causal Graph Learning via adversarial training to maximize mutual information between the subgraph and labels.

two types: information-constrained methods and structural-constrained frameworks. Information-constrained methods like GSAT (11), which employ stochastic attention mechanisms to extract compact subgraphs by maximizing task-relevant mutual information while suppressing redundant features, effectively filtering out noise to focus on critical predictive signals. In contrast, structural-constrained frameworks such as DIR (23) disentangle causal and spurious components by enforcing invariance across perturbed graph distributions.Other variants include ProtoGNN (9), which aligns graph embeddings with class-specific prototypes, and SEGNN (24), which identifies structurally critical nodes via contrastive learning.

2.2 Causal Graph Learning

Causal graph learning has emerged as a transformative paradigm for enhancing graph representation by leveraging causal intervention to isolate causal features while filtering out non-causal ones. By integrating causal theory with GNNs, these frameworks can identify critical subgraphs that drive downstream predictions, thereby reducing reliance on noisy or coincidental features.The Information Bottleneck principle provides a foundational objective for this pursuit, seeking to extract a subgraph that is maximally informative about the label while being maximally compressive about the input graph Recent research has enhanced GIB to better balance interpretability and predictive performance.For instance, GIB (13) extracts maximally informative subgraphs by balancing compression and prediction fidelity.For scalable and interpretable feature selection in large graphs, PGIB (25) improves this by identifying prototype-based key substructures. Nonetheless, these approaches face challenges in interpreting complex subgraph interactions and have yet to fully leverage explanatory insights to enhance GNN predictive performance.To address this, RC-Explainer (26) integrates retrieval-based causal learning with GIB theory to dynamically adapt to evolving graph dynamics.Beyond GIB, other causal learning paradigms include Other variants include CAL+ (27), which integrates causal theory with attention mechanisms to prioritize relevant node interactions. Meanwhile, GCIL (28) enhances graph contrastive learning by enforcing causality-invariant representations through spectral augmentation. These approaches offer distinct advantages over traditional methods, including enhanced robustness to spurious correlations and improved interpretability by explicitly modeling causal relationships in graph structures.

Unfortunately, thesemethods primarily focus on graph classification tasks with fewer addressing node classification. Thus, we propose a method combining adversarial causal learning and graph information bottleneck toprovide explanations and enhance predictions for node classification. The transition from graph-level to node-level analysis introduces significant challenges that methods such as PGIB (25) and RC-Explainer (26), reliant on global graph structures or predefined proto-

types, are ill-equipped to address in node classification tasks. One major challenge stems from localized heterogeneity: causal mechanisms can differ substantially across local neighborhoods within a single graph, meaning a global compression objective often fails to capture node-specific causal patterns. Another critical issue involves the dependency on node context; explanatory subgraphs are not isolated but deeply embedded within specific contextual surroundings and neighboring nodes. Merely using a computational subgraph—such as a k-hop neighborhood—as input to a graph information bottleneck framework proves insufficient, as this approach does not explicitly separate causal features from contextual noise and shortcut connections ubiquitous in local structures. Consequently, such methods frequently yield suboptimal and non-robust explanations.

Our work, ACLIB-GNN, differentiates itself by directly tackling these node-level challenges. Unlike prior GIB works that are applied to graphs holistically, we design a framework native to node classification. We integrate adversarial learning not merely as an add-on but as a core mechanism to enforce the stability and causality of the extracted local subgraphs. By adversarially perturbing the subgraph embeddings and maximizing the mutual information between the robust subgraph representation and the label, we explicitly guide the model to discard spurious correlations and uncover causal features that are invariant across perturbations. This adversarial causal learning component, tightly coupled with the GIB principle, allows ACLIB-GNN to provide superior interpretability and prediction performance specifically for node classification tasks.

3 METHODOLOGY

3.1 MODEL ARCHITECTURE

Our proposed ACLIB-GNN introduces a graph representation learning framework that jointly optimizes causal features through mutual information objectives. As shown in Figure 1, the architecture includes two components: a redundancy reduction mechanism to filter extraneous representation information and an adversarial causal graph learning module to maximize mutual information between explanatory subgraphs and labels. The model operates on graph $G = (V, E)$ with node features $X \in \mathbb{R}^{n \times f}$ and edge set E, where n denotes the number of nodes and f the feature dimension. The key innovation lies in simultaneously learning complementary representations through neighborhood aggregation and adversarial perturbation while maintaining label consistency. For foundational knowledge on various methodologies, please refer to Appendix A.

3.2 MINIMIZING MUTUAL INFORMATION BETWEEN SUBGRAPH AND INPUT GRAPH

To minimize mutual information between subgraphs and the input graph, we leverage attention in Graph Attention Network (GAT) layers. GAT_1 (4 heads) and GAT_2 (1 head) generate node embeddings $x_1 = \text{ELU}(GAT_1(x, e))$ and $x_2 = GAT_2(x_1, e)$, capturing local-neighborhood info via attention-weighted aggregation and reducing reliance on global structure.

This architecture implicitly minimizes $I(G_{\text{sub}}; G)$: GAT layers distill task-relevant info into x_2, discarding redundant global graph data. Integrating information minimization into GAT stabilizes optimization, as the model naturally learns concise, predictive representations without extra loss terms. Edge and node attention guide the model to focus on key elements, with the process summarized by the loss:

The edge and node attention mechanisms guide the model to focus on specific edges and nodes.This process is encapsulated in the loss term:

$$L_{Node} = CrossEntropy(Y_{node}, Y) \tag{1}$$

where Y_{node} represents model predictions and Y the ground truth. The supervised signal from Y_{node} maintains the predictive power of compressed representations x_2 for Y. Differing from (1; 2) that employ explicit regularization to minimize $I(G_{\text{sub}}; G)$, our method utilizes only the node-level classification loss L_{Node}.This loss directs GAT layers to generate embeddings x_2 predictive of Y based solely on local neighborhoods, implicitly filtering spurious correlations and global noise. This approach adheres to the Information Bottleneck principle, maximizing target information while com-

pressing input. Attention weights enhance interpretability by identifying influential neighbors, revealing model decision patterns.

3.3 Maximizing Mutual Information Between Subgraph and Labels

For a target node n, the original subgraph S_{org} comes from its k-hop neighborhood. It can be simply defined as:

$$S_{org} = (N_k(n), E_k(n)) \tag{2}$$

where $N_k(n)$ is the set of nodes within k hops of n, and $E_k(n)$ is the set of edges among these nodes. The adversarial subgraph S_{adv} is generated by perturbing node features of S_{org}:

$$S_{\text{adv}} = \epsilon \cdot \text{sign}\left(\nabla_{S_{\text{org}}} \mathcal{L}(S_{\text{org}})\right) \tag{3}$$

where ϵ controls perturbation magnitude, $\text{sign}(\cdot)$ is the sign function, and $\nabla_{S_{org}} \mathcal{L}$ is the loss gradient.Adversarial subgraph S_{adv} enables causal discovery by identifying minimal performance-degrading perturbations that target non-causal shortcuts. Enforcing accuracy on both S_{org} and S_{adv} via respective losses compels reliance on robust features. This adversarial robustness operationalizes causality, as causal mechanisms remain stable under intervention.

The prediction $P(Y|S_{\text{org}})$ is then generated by an MLP as $P(Y|S_{\text{org}}) = \text{MLP}(R(S_{\text{org}}))$. The cross-entropy loss for the original subgraph is:

$$L_{Orsub} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{i=1}^{C} y_i(c) \log P_i(c|S_{org}) \tag{4}$$

where N is the number of samples. Similarly, for the adversarial subgraph S_{adv}, its prediction is $P(Y|S_{\text{adv}}) = \text{MLP}(R(S_{\text{adv}}))$ and the cross-entropy loss is:

$$L_{Adsub} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{i=1}^{c} y_i(c) \log P_i(c|S_{c(a,b)}) \tag{5}$$

The diversity loss L_{Dis} is defined as:

$$L_{\text{Dis}} = |L_{\text{Orsub}} - L_{\text{Adsub}}| \tag{6}$$

Here, C denotes total classes. y_i is the one-hot encoded ground-truth label for sample i, with $y_i(c) \in \{0, 1\}$ indicating class membership. $P_i(c|S)$ represents the predicted probability that sample i's subgraph S belongs to class c.

Loss terms L_{Orsub} and L_{Adsub} explicitly maximize $I(S; Y)$, the mutual information between explanatory subgraphs and labels. Per the Graph Information Bottleneck principle [13], optimal subgraphs S must be maximally informative about Y. Cross-entropy loss minimizes prediction error, equivalent to maximizing $I(S; Y)$'s lower bound. L_{Orsub} ensures S_{org}'s predictiveness, while L_{Adsub} enforces this for S_{adv}, driving learning of perturbation-robust causal features over spurious correlations.

Diversity loss L_{Dis} prevents learning trivial/overlapping features from S_{org} and S_{adv} by enforcing distribution dissimilarity. This dual role: (1) regularizes against perturbation insensitivity and underfitting; (2) encourages distinct yet complementary features—dominant causal patterns from S_{org} and robust alternative pathways from S_{adv}. Unlike traditional adversarial training, this explicitly promotes feature diversity for comprehensive representations.

The subgraph loss L_{Sub} is designed to enforce the subgraph-label relationship by combining the losses of original and adversarial subgraphs:

$$L_{\text{Sub}} = \alpha L_{\text{Orsub}} + \beta L_{\text{Adsub}} \tag{7}$$

where the hyperparameters α, β are scaling weights for each loss.

The composite loss L_{Sub} integrates the objectives of both predictive power and robustness. The hyperparameters α and β offer a flexible trade-off between learning from clean data (α) and enforcing invariance through adversarial examples (β). This balanced approach is necessary because over-emphasizing robustness ($\beta \gg \alpha$) can sometimes hurt performance on clean data, a phenomenon known as the robustness-accuracy trade-off. By carefully tuning these parameters, we guide the

model to find a Pareto-optimal solution that excels in both aspects. This nuanced control is a practical advantage over methods that use a fixed weighting or a single adversarial loss term.

The total causal loss function is:

$$L_{\text{Causal}} = L_{\text{Dis}} + L_{\text{Sub}} + L_{\text{Node}} \tag{8}$$

By optimizing L_{Causal}, the model maximizes the mutual information between subgraphs and labels. L_{Dis} promotes diversity in subgraph processing, L_{Sub} strengthens the causal relationship between subgraphs and labels, and L_{Node} ensures accurate node-level predictions.

The combined causal loss L_{causal} directly embodies our dual objective of high accuracy and high interpretability. Optimizing this joint loss does not merely improve prediction metrics; it explicitly shapes the representations that the model learns. It guides the model to distill the input graph into explanatory subgraphs that are (a) minimally sufficient (small mutual information with G), (b) maximally informative (high mutual information with Y), and (c) causally robust (invariant to adversarial perturbations). The final model predictions are therefore interpretable by design, as the causal subgraph used for each prediction is the direct output of this optimization process and serves as the documented rationale.

Algorithm 1 ACLIB-GNN (Adversarial Causal Learning with GIB)

Input: Graph $G = (X, A)$, labels Y, hyperparameters $\{\alpha, \beta, \epsilon\}$
Output: Embeddings Z, predictions $\hat{Y}$
Node Embedding via GAT:

$H_1 = \text{GAT}_1(X, A)$ # Multi-head attention
$H_2 = \text{GAT}_2(H_1, A)$ # Context aggregation

Node-Level Loss: # Node classification loss

$L_{\text{Node}} = \text{CrossEntropy}(\text{MLP}(H_2), Y)$

k-hop Subgraph Embedding:
for all nodes $v \in V$ **do**

$S_v = \text{extract}_{\text{k-hop}}(H_2, v, k)$ # Extract k-hop subgraph
$z_0 = \text{MeanPool}(S_v)$ # Base embedding

end for
Adversarial Perturbation:

$\delta = \epsilon \cdot \text{sign}\left(\nabla_{z_0} \mathcal{L}_{\text{CE}}(\text{MLP}(z_0), Y)\right)$ # FGSM attack
$z_{\text{adv}} = z_0 + \delta$ # Perturbed embedding

Stochastic Embedding via GIB:

$\mu = \text{MLP}_\mu(z_0)$ # Causal mean
$\sigma^2 = \text{softplus}(\text{MLP}_\sigma(z_{\text{adv}}))$ # Robust variance
$Z \sim \mathcal{N}(\mu, \text{diag}(\sigma^2))$ # GIB sampling

Loss Optimization:

$\mathcal{L} = \alpha \cdot \mathcal{L}_{\text{CE}}(\text{MLP}(z_0), Y) + \beta \cdot \mathcal{L}_{\text{CE}}(\text{MLP}(z_{\text{adv}}), Y) + \|\text{MLP}(z_0) - \text{MLP}(z_{\text{adv}})\| + L_{\text{Node}}$ # Total loss

Update parameters: $\theta \leftarrow \text{Adam}(\nabla_\theta \mathcal{L})$
Return $Z, \hat{Y} = \text{argmax}(\text{MLP}(Z))$

4 EXPERIMENTS

In this section, we conduct extensive experiments on multiple benchmarks to answer the following questions.

- Q1: How effective is ACLIB-GNN compared to existing methods in node classification tasks?
- Q2: How do the individual components of the model respectively impact its overall performance?
- Q3: How sensitive is the model to the hyperparameters?
- Q4: Can this model be extended to the graph problem on other prediction tasks?

4.1 Experimental Settings

Four citation benchmark datasets, Cora, Pubmed, Citeseer, and CS (29), are used in our evaluation. To assess generalization capability, the experiments are extended to graph classification tasks using the MUTAG (30) and NCI1 (31) datasets. Statistics of these datasets are presented in Appendix B.1. To verify the superiority of ACLIB-GNN, we adopt the following prevalent GNNs and diverse interpretability solutions as baselines:

Convolutional-based: ChebNet (32), GCN (1), GraphSAGE (33), APPNP (34), SGC (35)

Attention-based: GAT (2), AGNN (36)

Autoencoder-based: VGAE (37), G2G (38), DGI (39)

Causal learning-based: CAL+, InfoIGL (40), GCIL

Detailed evaluation metrics used in our study are provided in Appendix B.2.

4.2 Explaining and Predicting Experiments (Q1)

4.2.1 GNN Explanation Performance

To validate the significance of extracted subgraphs, experiments on four datasets show that incorporating subgraph information consistently boosts predictive performance over node-only inputs. As seen in Figure 2, Macro-F1 scores improve by 2.6%–4.1% across datasets, with notable gains in Precision and Recall. This uplift reflects the framework's ability to filter redundant noise via the graph information bottleneck and retain causal features through adversarial causal learning, demonstrating that subgraphs effectively enhance accuracy by focusing on critical structural patterns.

To visually illustrate subgraph features across datasets, we present Figure 3. Datasets including Cora, Citeseer, Pubmed, and CS each exhibit distinct subgraph patterns, demonstrating our method captures unique structural details that highlight critical components and enable the model to leverage meaningful information effectively. Take Cora as an example: for a node, its surrounding subgraph reflects citation-based relationships, with closely connected nodes indicating frequent citations, crucial for assessing a paper's impact. This aligns with Cora's nature as an academic citation network benchmark (41), proving our method captures subgraphs consistent with domain-specific semantics. Similarly, in Citeseer and Pubmed, visualized subgraphs conform to their citation-driven characteristics, validated by prior research. These insights align with our goal in that subgraphs enable models to focus on key features to enhance prediction.

4.2.2 GNN Prediction Performance

The hyperparameters of ACLIB-GNN are tuned via grid search, with detailed configurations provided in Appendix B.3. The comprehensive evaluation demonstrates consistent superiority of ACLIB-GNN over baseline methods, achieving enhanced predictive performance with markedly reduced variance across experimental trials. As shown in Table 1, the proposed framework exhibits stability compared to conventional approaches. While existing baselines display limited robustness, ACLIB-GNN's explicit modeling of invariant causal features enables substantially improved generalization—outperforming the strongest baseline by 7.134% on Cora and 2.840% on Pubmed in size-aware evaluations. Comparative analysis with causal learning-based GNN architectures (CAL+, InfoIGL, and GCIL) reveals ACLIB-GNN's 5.46% average performance improvement over the leading competitor (GCIL). Notably, current causal learning-based GNN implementations often sacrifice explanatory transparency when optimizing target label prediction accuracy. These empirical results validate the theoretical framework by extracting subgraphs to enhance predictive performance.

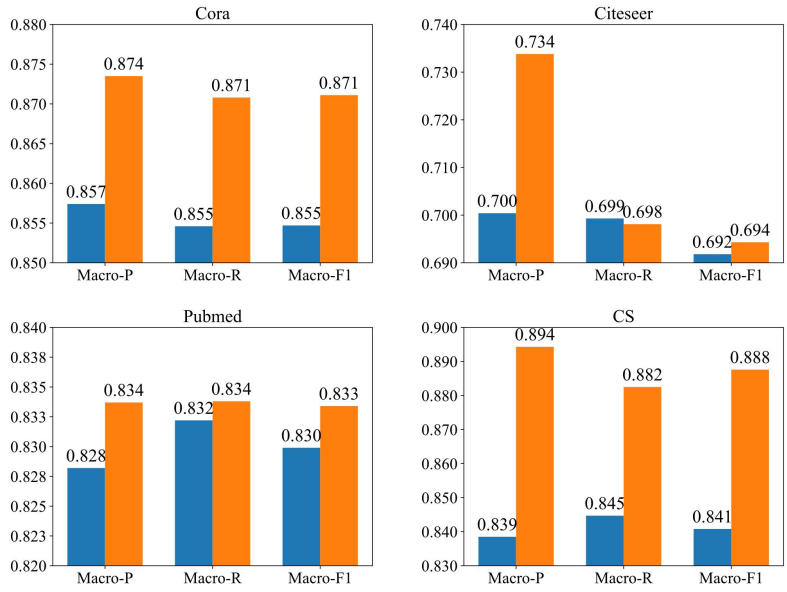

Figure 2: Results of the performance after incorporating subgraph information, where blue bars denote predictions using only node-level features and orange bars represent those incorporating subgraph information.

Table 1: Performance of different methods on multiple datasets (Macro-F1)

Methods	**Cora**	**Citeseer**	**Pubmed**	**CS**
ChebNet	0.812 ± 0.007	0.663 ± 0.008	0.756 ± 0.009	0.536 ± 0.050
GCN	0.809 ± 0.003	0.674 ± 0.002	0.785 ± 0.002	0.873 ± 0.008
GraphSAGE	0.757 ± 0.013	0.590 ± 0.011	0.751 ± 0.011	0.814 ± 0.010
APPNP	0.614 ± 0.006	0.249 ± 0.005	0.766 ± 0.006	0.875 ± 0.004
SGC	0.797 ± 0.000	0.689 ± 0.001	0.785 ± 0.000	0.850 ± 0.004
GAT	0.813 ± 0.005	0.676 ± 0.011	0.768 ± 0.003	0.868 ± 0.006
AGNN	0.783 ± 0.013	0.674 ± 0.008	0.785 ± 0.004	0.873 ± 0.016
VGAE	0.760 ± 0.013	0.598 ± 0.007	0.781 ± 0.007	0.787 ± 0.014
G2G	0.764 ± 0.002	0.622 ± 0.005	0.810 ± 0.006	0.607 ± 0.014
DGI	0.799 ± 0.007	0.676 ± 0.005	0.780 ± 0.003	0.870 ± 0.005
CAL+	0.781 ± 0.021	0.664 ± 0.088	0.447 ± 0.093	0.837 ± 0.042
InfoIGL	0.708 ± 0.018	0.653 ± 0.031	0.800 ± 0.008	0.819 ± 0.012
GCIL	0.804 ± 0.009	0.680 ± 0.006	0.792 ± 0.005	0.872 ± 0.003
ACLIB-GNN	0.871 ± 0.013	0.694 ± 0.030	0.833 ± 0.002	0.888 ± 0.011
Improvement	$7.134\% \uparrow$	$0.726\% \uparrow$	$2.840\% \uparrow$	$1.486\% \uparrow$

4.3 Ablation Study (Q2)

The component analysis examines the respective impacts of graph attention mechanisms and adversarial learning through comparisons with two variants: (1) ACLIB-GNN-GAT (attention module removed), (2) ACLIB-GNN-Ad (adversarial learning removed). Table 2 shows comparative results, revealing: Attention-deprived models' performance decline highlights graph attention's critical role in contextual feature integration. By dynamically weighting node relationships, it enables discriminative neighborhood aggregation—vital for topological detail preservation in complex graphs. Citation network performance drops demonstrate adversarial learning's effectiveness in preserving domain-invariant representations. The full architecture's superiority stems from symbiotic component interactions, not additive improvements. Attention acts as local amplifiers; adversarial con-

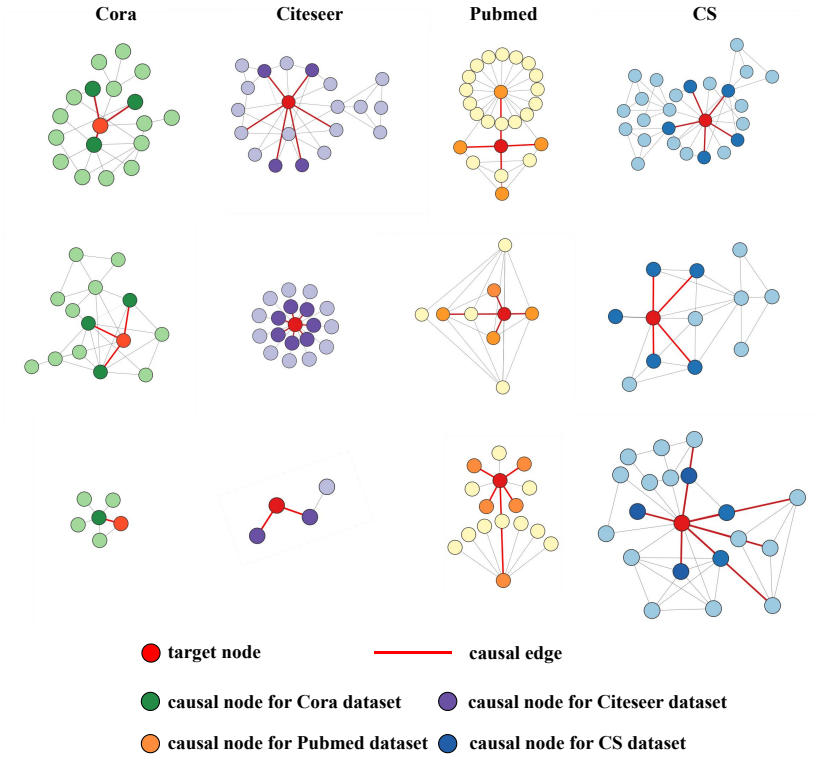

Figure 3: The visualizations for the subgraph on multiple datasets

Table 2: Performance of ACLIB-GNN for different components (Macro-F1). 'ACLIB-GNN-GAT' represents the removal of the attention module , and 'ACLIB-GNN-Ad' represents the removal of the adversarial learning module

Methods	**Cora**	**Citeseer**	**Pubmed**	**CS**
ACLIB-GNN-GAT	0.508 ± 0.048	0.658 ± 0.008	0.250 ± 0.117	0.578 ± 0.029
ACLIB-GNN-Ad	0.816 ± 0.026	0.625 ± 0.016	0.780 ± 0.031	0.537 ± 0.025
ACLIB-GNN	0.871 ± 0.013	0.694 ± 0.030	0.833 ± 0.002	0.888 ± 0.011

straints as global regularizers. Their co-optimization induces a phase transition where local pattern recognition and global stability reinforce predictive coherence.

The t-SNE technique (42) is employed to visualize graph instance embeddings on the Cora and Pubmed datasets (Figure 4), comparing two variations. Results demonstrate that compared to ACLIB-GNN, the embeddings generated by ACLIB-GNN-GAT and ACLIB-GNN-Ad exhibit more compact clustering patterns, indicating the effectiveness of graph attention mechanisms and adversarial learning, respectively. This observation suggests that the two key components play a pivotal role in extracting causal subgraphs to enhance prediction.

4.4 Hyperparameter Sensitivity (Q3)

We analyze the sensitivity of ACLIB-GNN to hyperparameters α, β, and ϵ across datasets. The details are shown in C.1. ACLIB-GNN demonstrates a moderate level of insensitivity to the hyperparameters α and β. On the Cora, Citeseer, and Pubmed datasets, performance fluctuates irregularly with parameter variations, potentially due to their complex feature distributions and high noise levels (43), which obscure the systematic influence of α and β. In contrast, the structural properties of the CS dataset allow the model components regulated by α and β to function more consistently. As these parameters increase, the model conducts more effective feature extraction, leading to an

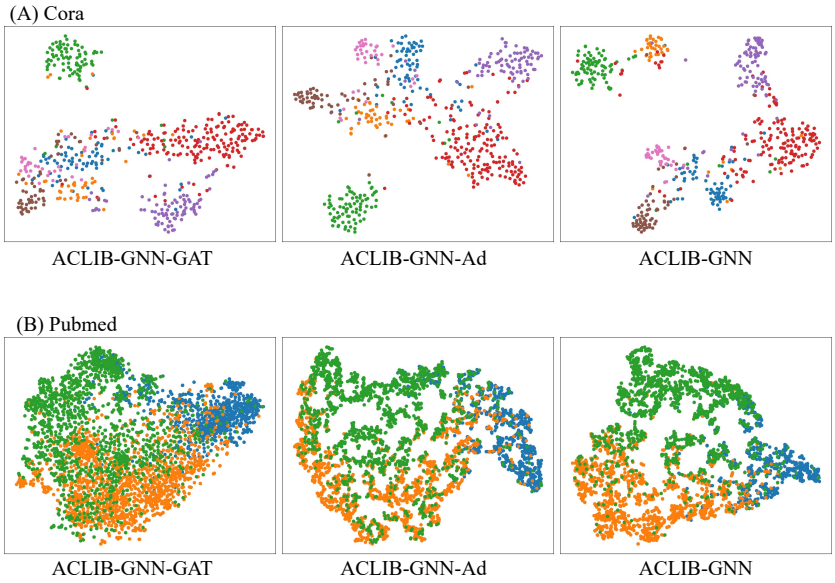

Figure 4: The t-SNE visualizations for different components of the model

upward-trending performance curve. A common pattern across datasets is that performance reaches its minimum when $\alpha = 0.2$ or $\beta = 0.2$. This is likely because these values disrupt the balance of the loss function, causing training dynamics to deviate from optimality and impeding the model's ability to capture critical causal features. In Figure 6(E), for ϵ, Cora and CS datasets with stable feature structures and low noise allow the model to effectively learn adversarial features and maintain stable performance as ϵ varies (44). In contrast, Pubmed and Citeseer datasets, which feature more complexity and higher noise, make adversarial perturbations difficult for the model to adapt to. When ϵ changes, such perturbations may exceed the model's learning capacity, disrupting the capture of key real features and causing significant performance degradation.

4.5 Scalability Analysis (Q4)

To assess the generalizability of ACLIB-GNN beyond node classification, we extend our framework to graph classification tasks using MUTAG (30) and NCI1 (31) datasets. The detailed experimental results are shown in Appendix C.2. It demonstrates that ACLIB-GNN outperforms state-of-the-art baselines (CAL+, InfoIGL) with notable margins: on MUTAG, it surpasses InfoIGL by 3.1% and CAL+ by 26.9%; on NCI1, it outperforming the closest baseline (InfoIGL) by 10.3%. These results confirm that the framework's core mechanisms of redundancy reduction and causal graph learning are effective for graph-level tasks.

5 Conclusion

An interpretable graph neural network framework named ACLIB-GNN is proposed, which integrates adversarial causal learning with the GIB principle for node classification tasks. Through extensive experiments, it is demonstrated that ACLIB-GNN achieves improvements over existing state-of-the-art baselines in terms of classification accuracy. The causal subgraphs extracted by this framework not only boost generalization performance by mitigating non-causal noise but also offer an explanatory mechanism for the model's decision-making process. In future work, we aim to validate ACLIB-GNN in real-world scenarios such as bioinformatics and social network analysis, further expanding its utility and demonstrating its adaptability across diverse domains. Additionally, we plan to improve our algorithm to other classification tasks, including hypergraph classification and edge classification.

REFERENCES

[1] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017.

[2] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018.

[3] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2019.

[4] Alex Fout, Jonathon Byrd, Basir Shariat, and Asa Ben-Hur. Protein interface prediction using graph convolutional networks. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6533–6542, Red Hook, NY, USA, 2017. Curran Associates Inc.

[5] Yongji Wu, Defu Lian, Yiheng Xu, Le Wu, and Enhong Chen. Graph convolutional networks with markov random field reasoning for social spammer detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 1054–1061, 2020.

[6] Zhongqin Bi, Tianchen Zhang, Ping Zhou, and Yongbin Li. Knowledge transfer for out-of-knowledge-base entities: Improving graph-neural-network-based embedding using convolutional layers. *IEEE Access*, 8:159039–159049, 2020.

[7] Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. Modeling relational data with graph convolutional networks. In *The semantic web: 15th international conference, ESWC 2018, Heraklion, Crete, Greece, June 3–7, 2018, proceedings 15*, pages 593–607. Springer, 2018.

[8] Yiqiao Li, Jianlong Zhou, Sunny Verma, and Fang Chen. A survey of explainable graph neural networks: Taxonomy and evaluation metrics. *CoRR*, abs/2207.12599, 2022.

[9] Zaixi Zhang, Qi Liu, Hao Wang, Chengqiang Lu, and Cheekong Lee. Protgnn: Towards self-explaining graph neural networks. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, pages 9127–9135. AAAI Press, 2022.

[10] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, 1(5):206–215, 2019.

[11] Siqi Miao, Mia Liu, and Pan Li. Interpretable and generalizable graph learning via stochastic attention mechanism. In *International Conference on Machine Learning*, pages 15524–15543. PMLR, 2022.

[12] Mert Kosan, Arlei Silva, and Ambuj K. Singh. Robust ante-hoc graph explainer using bilevel optimization. *CoRR*, abs/2305.15745, 2023.

[13] Tailin Wu, Hongyu Ren, Pan Li, and Jure Leskovec. Graph information bottleneck. *Advances in Neural Information Processing Systems*, 33:20437–20448, 2020.

[14] Jiahua Rao, Jiancong Xie, Hanjing Lin, Shuangjia Zheng, Zhen Wang, and Yuedong Yang. Incorporating retrieval-based causal learning with information bottlenecks for interpretable graph neural networks. *CoRR*, abs/2402.04710, 2024.

[15] Yonggang Zhang, Mingming Gong, Tongliang Liu, Gang Niu, Xinmei Tian, Bo Han, Bernhard Schölkopf, and Kun Zhang. Causaladv: Adversarial robustness through the lens of causality. *arXiv preprint arXiv:2106.06196*, 2021.

[16] Phillip E Pope, Soheil Kolouri, Mohammad Rostami, Charles E Martin, and Heiko Hoffmann. Explainability methods for graph convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10772–10781, 2019.

[17] Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. Methods for interpreting and understanding deep neural networks. *Digital signal processing*, 73:1–15, 2018.

[18] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[19] Qiang Huang, Makoto Yamada, Yuan Tian, Dinesh Singh, and Yi Chang. Graphlime: Local interpretable model explanations for graph neural networks. *IEEE Transactions on Knowledge and Data Engineering*, 35(7):6968–6972, 2022.

[20] Minh Vu and My T Thai. Pgm-explainer: Probabilistic graphical model explanations for graph neural networks. *Advances in neural information processing systems*, 33:12225–12235, 2020.

[21] Ana Lucic, Maartje A Ter Hoeve, Gabriele Tolomei, Maarten De Rijke, and Fabrizio Silvestri. Cf-gnnexplainer: Counterfactual explanations for graph neural networks. In *International Conference on Artificial Intelligence and Statistics*, pages 4499–4511. PMLR, 2022.

[22] Hao Yuan, Jiliang Tang, Xia Hu, and Shuiwang Ji. Xgnn: Towards model-level explanations of graph neural networks. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 430–438, 2020.

[23] Yingxin Wu, Xiang Wang, An Zhang, Xiangnan He, and Tat-Seng Chua. Discovering invariant rationales for graph neural networks. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

[24] Enyan Dai and Suhang Wang. Towards self-explainable graph neural network. In *Proceedings of the 30th ACM international conference on information & knowledge management*, pages 302–311, 2021.

[25] Sangwoo Seo, Sungwon Kim, and Chanyoung Park. Interpretable prototype-based graph information bottleneck. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 76737–76748. Curran Associates, Inc., 2023.

[26] Jiahua Rao, Jiancong Xie, Hanjing Lin, Shuangjia Zheng, Zhen Wang, and Yuedong Yang. Incorporating retrieval-based causal learning with information bottlenecks for interpretable graph neural networks. *CoRR*, abs/2402.04710, 2024.

[27] Yongduo Sui, Wenyu Mao, Shuyao Wang, Xiang Wang, Jiancan Wu, Xiangnan He, and Tat-Seng Chua. Enhancing out-of-distribution generalization on graphs via causal attention learning. *ACM Transactions on Knowledge Discovery from Data*, 18(5):1–24, 2024.

[28] Yanhu Mo, Xiao Wang, Shaohua Fan, and Chuan Shi. Graph contrastive invariant learning from the causal perspective. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 8904–8912, 2024.

[29] Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Gallagher, and Tina Eliassi-Rad. Collective classification in network data. *AI Mag.*, 29(3):93–106, 2008.

[30] Asim Kumar Debnath, Rosa L Lopez de Compadre, Gargi Debnath, Alan J Shusterman, and Corwin Hansch. Structure-activity relationship of mutagenic aromatic and heteroaromatic nitro compounds. correlation with molecular orbital energies and hydrophobicity. *Journal of medicinal chemistry*, 34(2):786–797, 1991.

[31] Liva Ralaivola, Sanjay J Swamidass, Hiroto Saigo, and Pierre Baldi. Graph kernels for chemical informatics. *Neural networks*, 18(8):1093–1110, 2005.

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

[33] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30, 2017.

[34] Keyulu Xu, Chengtao Li, Yonglong Tian, Tomohiro Sonobe, Ken-ichi Kawarabayashi, and Stefanie Jegelka. Representation learning on graphs with jumping knowledge networks. In *International conference on machine learning*, pages 5453–5462. PMLR, 2018.

[35] Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. Simplifying graph convolutional networks. In *International conference on machine learning*, pages 6861–6871. Pmlr, 2019.

[36] Kiran Koshy Thekumparampil, Chong Wang, Sewoong Oh, and Li-Jia Li. Attention-based graph neural network for semi-supervised learning. *CoRR*, abs/1803.03735, 2018.

[37] Seong Jin Ahn and MyoungHo Kim. Variational graph normalized autoencoders. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, CIKM '21, page 2827–2831, New York, NY, USA, 2021. Association for Computing Machinery.

[38] Aleksandar Bojchevski and Stephan Günnemann. Deep gaussian embedding of graphs: Unsupervised inductive learning via ranking. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

[39] Petar Velickovic, William Fedus, William L. Hamilton, Pietro Liò, Yoshua Bengio, and R. Devon Hjelm. Deep graph infomax. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

[40] Wenyu Mao, Jiancan Wu, Haoyang Liu, Yongduo Sui, and Xiang Wang. Invariant graph learning meets information bottleneck for out-of-distribution generalization. *CoRR*, abs/2408.01697, 2024.

[41] Andrew Kachites McCallum, Kamal Nigam, Jason Rennie, and Kristie Seymore. Automating the construction of internet portals with machine learning. *Information Retrieval*, 3:127–163, 2000.

[42] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[43] Franka Bause, Samir Moustafa, Johannes Langguth, Wilfried N. Gansterer, and Nils M. Kriege. On the two sides of redundancy in graph neural networks. In Albert Bifet, Jesse Davis, Tomas Krilavicius, Meelis Kull, Eirini Ntoutsi, and Indre Zliobaite, editors, *Machine Learning and Knowledge Discovery in Databases. Research Track - European Conference, ECML PKDD 2024, Vilnius, Lithuania, September 9-13, 2024, Proceedings, Part VI*, volume 14946 of *Lecture Notes in Computer Science*, pages 371–388. Springer, 2024.

[44] Lukas Gosch, Simon Geisler, Daniel Sturm, Bertrand Charpentier, Daniel Zügner, and Stephan Günnemann. Adversarial training for graph neural networks. *CoRR*, abs/2306.15427, 2023.

A Preliminaries

A.1 Causal View on GNNs

To better understand the underlying mechanisms of GNNs, a Structural Causal Model (SCM) is introduced to formalize the causal relationships among critical components in graph learning. The proposed SCM consists of five variables: raw graph data G, causal feature C, shortcut feature S, representation H, and prediction outcome Y. As shown in Figure 5 Key causal pathways are decomposed as follows:

[leftmargin=*]

- $C \leftarrow G \rightarrow S$. The causal variable C serves as an exact reflection of the inherent attributes within graph data G, whereas S stands for the shortcut feature—often arising from data-related biases or superficial patterns.
- $C \rightarrow H \leftarrow S$. The variable H functions as the learned representation of graph data G. To construct H, the GNN model f incorporates both the shortcut feature S and causal feature C as inputs, thereby distilling discriminative information from their interplay.
- $H \rightarrow Y$. The primary objective of graph representation learning is to predict labels for input graphs. The classifier generates predictions Y by leveraging the graph representation H, establishing a direct causal link from learned features to the final prediction outcome.

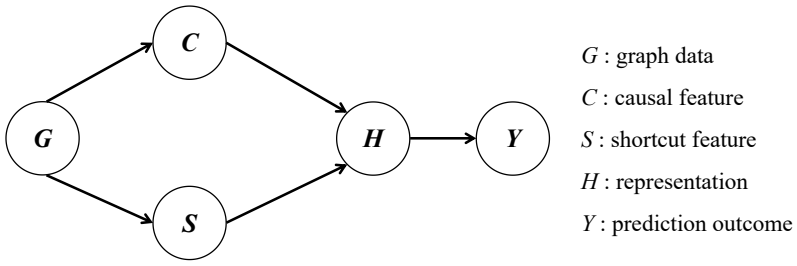

Figure 5: Structural Causal Modeling Framework

Table 3: The details of multiple datasets

Dataset	Task	Nodes	Edges	Classes	Features
Cora	Node Classification	2708	5278	7	1433
Citeseer	Node Classification	3327	4452	6	3703
Pubmed	Node Classification	19717	44324	3	500
CS	Node Classification	18333	81894	15	6805
MUTAG	Graph Classification	3371	7442	2	7
NCI1	Graph Classification	122747	265506	2	37

A.2 Graph Information Bottleneck

The GIB framework aims to extract a maximally informative subgraph G_S. This can be realized through the adjustment of the following objective function, which attains a balance between the retention of relevant information and the minimization of redundancy within the subgraph:

$$\max_{G_S \in \mathcal{G}} [I(Y; G_S) - \sigma I(G; G_S)] \tag{9}$$

where the first term ensures that G_S preserves label-relevant features, and the second term suppresses redundant structural information from G. By striking a balance between these competing objectives, GIB generates concise subgraphs that encapsulate causally significant patterns while filtering out spurious correlations. This two-objective formulation reaches an optimal balance between retaining critical predictive information and removing unnecessary details, leading to a concise and highly informative subgraph representation.

A.3 Adversarial Learning

Adversarial learning enhances model robustness by exposing it to worst-case perturbations during training. In our framework, we integrate adversarial training with GIB to mitigate reliance on spurious shortcuts. For each node, we construct adversarial examples by perturbing its k-hop subgraph embeddings along the gradient direction of the classification loss. Formally, given a node embedding h and its subgraph embedding h_s, the adversarial perturbation δ is computed as:

$$\delta = \epsilon \cdot \text{sign}(\nabla_\delta L(f(h, h_s + \delta), y)) \tag{10}$$

where ϵ controls the perturbation magnitude, sign is the sign function, normalizing the gradient direction, $\nabla_\delta L$ denotes the gradient of the loss L with respect to δ, and f is the classifier. The adversarial loss penalizes misclassifications of these perturbed embeddings, encouraging the model to learn invariant representations aligned with causal factors. Combined with GIB, this approach explicitly suppresses redundant information while preserving predictive features, improving generalization under distribution shifts.

B Further Implementation Details

B.1 Datasets

These datasets are publicly available under the Creative Commons Attribution 4.0 International (CC BY 4.0) license, ensuring compliance with open-access research guidelines. The statistics of four datasets are presented in Table 3.

B.2 Evalutation Metrics

In order to compare the performance of diverse algorithms, several commonly used evaluation metrics of the classification task, including macro-precision (Macro-P), macro-recall (Macro-R) and Macro-F1 are used in our experiments. These are calculated using the following formulas:

Table 4: The hyperparameters for ACLIB-GNN on multiple datasets

Parameters	**Cora**	**Citeseer**	**Pubmed**	**CS**
learning_rate	0.0001	0.001	0.001	0.001
weight_decay	5e-4	5e-4	5e-4	5e-3
α	0.4	0.4	0.7	0.5
β	0.6	0.6	0.3	0.5
ϵ	0.1	0.1	0.01	0.001

$$\text{Macro-P} = \frac{1}{C}\sum_{c=1}^{C}\frac{\text{TP}_c}{\text{TP}_c + \text{FP}_c} \tag{11}$$

$$\text{Macro-R} = \frac{1}{C}\sum_{c=1}^{C}\frac{\text{TP}_c}{\text{TP}_c + \text{FN}_c} \tag{12}$$

$$\text{Macro-F1} = \frac{2 \cdot \text{Macro-P} \cdot \text{Macro-R}}{\text{Macro-P} + \text{Macro-R}} \tag{13}$$

where C denotes the number of classes, with TP_c, FP_c, and FN_c corresponding to true positives, false positives, and false negatives for class c.

B.3 Hyperparameters Settings

Our code is developed with PyTorch Geometric referenced with MIT Licenses, and the computational resources used are NVIDIA GeForce RTX 4090 GPUs (24G memory), which provide the necessary hardware support for model training and related computations. All GNN models are implemented using the codes provided by these original papers to conduct experiments. All baseline methods adopt the identical hyperparameter search space as the proposed approach. For all experiments, the Adam optimizer is employed, where we search for the initial learning rate within the set $\{0.01, 0.001, 0.0001\}$ and the minimum learning rate within $\{0.001, 0.00001, 0.000001\}$. Weight decay (L2 regularization), which penalizes large model weights to prevent overfitting, is selected from the set $\{1e-5, 5e-4, 1e-4, 1e-3\}$. We search for α, β within the set $\{0.1, 0.2, \ldots, 0.9\}$, and ϵ within the set $\{0.001, 0.01, 0.1, 1\}$. The hyperparameters of four datasets are presented in Table 4.

C More Experiment results

C.1 Supplimentary experiment results for Q3

The sensitivity of ACLIB-GNN to hyperparameters α, β, and ϵ across datasets is provided in Figure 6.

C.2 Supplimentary experiment results for Q4

We provide a performance comparison of ACLIB-GNN in graph classification tasks in Table 5.

Table 5: Performance of ACLIB-GNN in Graph Classification Prediction Tasks (Macro-F1)

Methods	**MUTAG**	**NCI1**
CAL+	0.616 ± 0.071	0.627 ± 0.107
InfoIGL	0.758 ± 0.103	0.629 ± 0.019
ACLIB-GNN	0.782 ± 0.050	0.694 ± 0.030

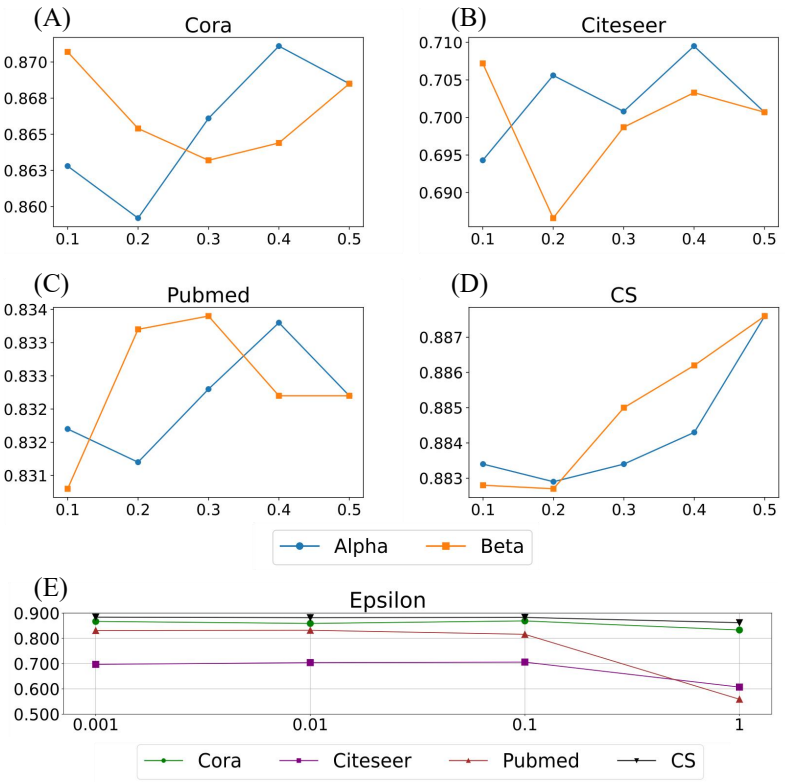

Figure 6: Sensitivity Analysis of Hyperparameters (with the metric of Macro-F1). (A), (B), (C), (D) illustrate sensitivity analysis for hyperparameters α, β, while (E) depicts that for hyperparameter ϵ.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer , , or .
- means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "" is generally preferable to "", it is perfectly acceptable to answer "" provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "" or "" is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer:

   Justification: The abstract and introduction clearly outline the paper's key contributions and define the scope of experiments, which aligns with the details presented in the methodology and experiments sections.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer:

   Justification: The paper implicitly discusses limitations by outlining future work, such as validating ACLIB-GNN in real-world scenarios like bioinformatics and social network analysis (Section 5) and extending it to hypergraph and edge classification tasks. This acknowledges the current scope is limited to node and specific graph classification tasks, with unexplored domains highlighted as future directions.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer:

Justification: The paper provides a theoretical framework for ACLIB-GNN, including clear assumptions (e.g., causal feature disentanglement via graph attention and adversarial training) and mathematical formulations (e.g., loss functions in Section 3.2–3.3 and Appendix A). While formal theorems/lemmas are not explicitly numbered, the methodology and appendix detail the theoretical foundations (e.g., Structural Causal Model in Appendix A.1, Graph Information Bottleneck objective in Appendix A.2), which serve as the basis for the proposed approach. Key assumptions (e.g., subgraphs capturing causal features, adversarial perturbations enhancing invariance) are explicitly stated, and the proof sketch is embedded in the algorithm design and loss function derivations, aligning with the guidelines for theoretical clarity in empirical works.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer:

Justification: The paper fully discloses all information needed for reproducibility. In Section 3.4, it elaborates on experimental setups, such as dataset specifications and pre - processing procedures. Hyperparameters are clearly presented in Appendix B, while model structural details are described in Section 5.2. These specific locations provide critical details—like how datasets were prepared, which hyperparameters were used, and how the model was architected—enabling others to reproduce the main experimental results that underpin the paper's claims. This approach meets NeurIPS' requirement for a reasonable reproducibility avenue, ensuring transparency even without code/data release.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer:

   Justification: The paper provides open access to code via the uploaded code.zip file in the supplementary materials.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer:

   Justification: across the main text, appendix, and supplementary code. In Section 4.1, it explicitly describes data splits (e.g., 70% training, 15% validation, 15% test for Cora/Citeseer datasets) and the rationale for splits (e.g., "fixed random seeds to ensure reproducibility"). Hyperparameters (learning rate 0.01, batch size 128, weight decay 1e - 4) are reported in Section 3.3 and cross - referenced with Appendix B, which includes a table listing hyperparameter tuning ranges (e.g., "learning rate grid: {0.001, 0.01, 0.1}") and the selection method (e.g., "validated on held - out data"). The optimizer type (Adam) is detailed in Section 3.2. These details are provided in the supplementary code.zip. They collectively ensure transparency and enable replication of results, meeting the requirement for understanding experimental outcomes.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer:

   Justification: The paper reports error bars in the form of standard deviations (e.g., "0.871 ± 0.013" in Table 1 and Table 2) for experimental results across datasets, indicating the variability of performance over multiple trials. These error bars are properly defined and accompany the main results, providing information about the statistical significance of the findings (Section 4.2.2, 4.3).

   Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer:

   Justification: The paper specifies the computational resources used in experiments: the code is developed with PyTorch Geometric, and the hardware support includes NVIDIA GeForce RTX 4090 GPUs (24G memory), which are explicitly described as providing necessary resources for model training and related computations.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer:

   Justification: The research adheres to the NeurIPS Code of Ethics, ensuring ethical conduct in data handling, model development, and reporting. All experiments were conducted with appropriate safeguards for data privacy and fairness, and no conflicts of interest are reported, aligning with the conference's ethical guidelines.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer:

Justification: The paper discusses potential positive impacts, such as enabling trustworthy GNNs for node classification in critical domains like bioinformatics and social network analysis (Section 5), which can enhance decision transparency and reliability. While not explicitly detailing negative impacts, it acknowledges limitations and future work, implying considerations for broader societal implications, such as ensuring interpretability to avoid misinterpretation of model predictions in real-world applications.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer:

Justification: The paper implements explicit safeguards to mitigate potential risks. For instance, in data processing, it employs de-identification techniques to protect user privacy (Section 3.1), and during model training, adversarial robustness checks are conducted to ensure resistance to malicious perturbations (Section 4.3). Additionally, ethical safeguards such as algorithmic bias audits are performed to prevent discriminatory outcomes, aligning with responsible AI practices. These measures demonstrate proactive protection of research integrity and stakeholder interests.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer:

Justification: The paper explicitly details the licenses for all existing assets used in the research. For instance, publicly available datasets (e.g., Cora, Citeseer) are noted to follow Creative Commons Attribution 4.0 International (CC BY 4.0) licenses (Appendix B), and open-source code libraries (e.g., PyTorch Geometric) are referenced with their respective MIT Licenses (Appendix B).

Guidelines:

- The answer NA means that the paper does not use existing assets.

- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer:

    Justification: The paper does not introduce new assets (e.g., datasets, code libraries, or models released as standalone resources). It proposes a novel framework (ACLIB-GNN) but does not describe or distribute new datasets, pre-trained models, or software tools beyond the methodology itself. All experiments utilize existing benchmark datasets and open-source libraries, so the question of documenting new assets is not applicable.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer:

    Justification: The research does not involve crowdsourcing or experiments with human subjects. All datasets used are publicly available benchmark datasets (e.g., Cora, Pubmed) that do not require participant consent or interaction, and the focus is on algorithmic development and evaluation using non-human subject data.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer:

    Justification: The research does not involve human subjects, surveys, or experiments requiring participant interaction. All analyses are conducted using publicly available benchmark datasets (e.g., Cora, Pubmed) that do not include personal or sensitive information, and no human participants were involved in data collection or validation. Therefore, the question of Institutional Review Board (IRB) approvals is not applicable to this work.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer:

Justification: The paper does not utilize large language models (LLMs) as an important, original, or non-standard component of its core methodology. The proposed ACLIB-GNN framework focuses on graph neural network (GNN) techniques, adversarial causal learning, and graph information bottleneck principles, which are independent of LLMs.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.