Graph Contrastive Learning with Balanced Hard Negatives and Fine-grained Semantic-aware Positives

Hongshan Pu, Haoxu Zhang, Ye Liu*, Hongmin Cai

School of Future Technology, South China University of Technology, China
puhongshan016@163.com, ftzhx@mail.scut.edu.cn, {yliu03,hmcai}@scut.edu.cn

Abstract

Graph contrastive learning (GCL) aims to learn representations by bringing semantically similar graphs closer and pushing dissimilar ones farther apart without label supervision. Hard negatives, which refer to graphs that have different labels but similar embeddings to the target graph, play a key role in improving representation discrimination. However, current methods that generate both high-quality positives and hard negatives face two challenges: (1) Hard negative sample generation often suffers from class imbalance, resulting in unequal attention across classes and reduced discriminative power in the learned representations. (2) The typical binary positive sample generation approach, which divides the graph into important and unimportant semantic regions, overlooks regions that negatively impact semantics and mislead model predictions. To address these issues, we introduce a novel method named BalanceGCL, which enhance graph contrastive learning with balanced hard negatives and fine-grained semantic-aware positives. BalanceGCL comprises two modules: Balanced Hard Negative graphs generation (BHN) and Fine-grained Semantic-aware Positive graphs generation (FSP). Inspired by the counterfactual mechanism, BHN generates balanced hard negatives that remain structurally similar to the original graph while inducing a controlled semantic shift. To ensure class balance, BHN iteratively constructs one hard negative sample for each class, ensuring an even distribution of negative samples across all alternative categories. FSP leverages the semantic differences between original graphs and balanced hard negatives to identify positively contributing, negatively contributing, and unimportant regions. By enhancing the influence of positive contributors, suppressing negative ones, and perturbing unimportant areas, it generates more reliable and semantically complete positive samples. The proposed method outperforms state-of-the-art GCL techniques across 14 datasets in graph classification and transfer learning tasks, demonstrating its effectiveness in tackling class imbalance and identifying fine-grained semantic-aware regions.

Code — https://github.com/YeLiu-Lab/BalanceGCL

Introduction

Graph Contrastive Learning (GCL) (Li et al. 2022) enables effective unsupervised graph-level representation learning without relying on human-labeled data. The objective of GCL is to bring similar (positive) graphs closer in the embedding space while pushing dissimilar (negative) graphs farther apart (Tian et al. 2020). Consequently, a key difficulty in GCL is how to generate high-quality positive and negative graphs such that the contrastive objective can effectively capture meaningful semantic similarities and differences.

Existing GCL approaches generate high-quality positive graphs by designing diverse augmentation schemes that preserve critical structures or attributes while perturbing unimportant components. Earlier studies used human-defined heuristics for graph augmentation, e.g., unimportant edge perturbation (Zhu et al. 2021), unimportant node feature masking (Liu et al. 2022), important subgraph sampling (Qiu et al. 2020) and graph diffusion (Hassani and Khasahmadi 2020). Recently, learning-based augmentation methods have employed a data-driven manner to adaptively identify important regions. For example, GCS (Wei et al. 2023) identified the most semantically discriminative structures of a graph via contrastive learning, generating semantically meaningful augmentations by preserving discriminative structures with saliency information. Inspired by invariant rationale discovery, RGCL (Li et al. 2022) employs a learnable GNN-MLP generator to identify key structures by maximizing similarity between the original graph and its rationale subgraph. However, these methods typically use binary partitioning, preserving positively relevant regions while perturbing the rest, overlooking fine-grained regions with negative semantic impact. For instance, in toxic molecule classification, groups like nitro ($-\mathrm{NO_2}$) or hydroxyl ($-\mathrm{OH}$) often enhance toxicity and are correctly identified as positively contributing regions. In contrast, stabilizing groups such as methyl ($-\mathrm{CH_3}$) suppress toxicity (Pinheiro, Franco, and Fraga 2023), but are often misclassified as important or irrelevant, neglecting their negative semantic impact on representation learning.

On the other hand, for negative graphs generation, traditional methods like GraphCL (You et al. 2020) uniformly sample negatives from all graphs and inevitably introduce false negatives, meaning negative graphs that likely share the same label as the original instances, which ultimately degrades GCL performance. To mitigate this issue, a Prototypical Graph Contrastive Learning (PGCL) approach (Lin et al.

*Corresponding author.

2022) performed negative sampling from all clusters except the cluster of the target. Recently, several works improve the discrimination of GCL by creating hard negatives, which have a different label while their embeddings are close to that of the target. For example, (Luo et al. 2023) incorporated a generation branch called GraphACL that maximizes the contrastive learning loss in an adversarial way, enabling the hard negatives to approach the target graphs. (Yang et al. 2023a) introduced the counterfactual mechanism into GCL, minimizing structural differences between original and perturbed graphs while maximizing classification probability differences, thereby generating hard negative graphs. However, existing negative graph generation methods only ensure label difference from the target, ignoring class distribution. This often results in imbalanced negative samples, especially in datasets with uneven class distributions. Such imbalance may cause GCL to overfit common negatives and fail to learn sufficient information to distinguish rarer negatives, limiting its discriminative ability. To validate this, we generate negative samples by sampling from original datasets or CGC (Yang et al. 2023a) with three class distributions: Balanced, Random, and Highly Imbalanced. All other settings strictly follow those of GraphCL (You et al. 2020). Negative samples with a Balanced distribution are evenly selected across all classes. In the Random setting, samples are chosen randomly. For the Highly Imbalanced setting, 75% of negatives come from one class, with the remaining 25% evenly spread among others. As shown in Fig. 1, negative samples generated with a balanced distribution achieved significantly higher accuracy than other strategies, demonstrating a superior discriminative representation.

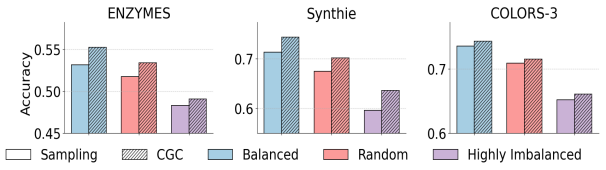

Figure 1: The impact of different negative sample distributions, including balanced, random, and highly imbalanced settings, on model accuracy using the ENZYMES, Synthie and COLORS-3 datasets. The model is based on the GraphCL framework, with negative samples either selected directly from the dataset (denoted as Sampling) or generated using the CGC method.

To addresses the aforementioned challenges, we propose a new method named BalanceGCL, which generates balanced hard negatives and fine-grained semantic-aware positive samples. BalanceGCL consists of two modules: Balanced Hard Negative graphs generation (BHN) and Fine-grained Semantic-aware Positive graphs generation (FSP). Specifically, BHN introduces structure and feature perturbations on the original graph to shift the prediction from the original class to a specific target class, while preserving confidence in other classes and minimizing deviation from the original graph. By applying this strategy to each class, BHN generates balanced hard negatives. FSP identifies regions with positive, negative, or negligible semantic impacts by comparing class-aware gradients between the original and balanced hard negative graphs. These regions facilitate the generation of fine-grained semantic positives by perturbing uninformative areas to enhance diversity, reinforcing positively contributing regions to preserve key semantics, and suppressing negatively contributing ones to minimize interference. This process ensures accurate semantic alignment with the original graph. The workflow of BalanceGCL is shown in Fig. 2. In short, the contributions can be summarized as follows:

- We propose a new method called BalanceGCL, which facilitates the GCL by generating balanced hard negative graphs and fine-grained semantic-aware positive graphs.
- Based on counterfactual explainability, the BHN module generates balanced hard negative graphs by minimally perturbing the original graph to decrease its original class probability, increase that of a target class, and maintain confidence in other classes.
- The FSP module is proposed to construct fine-grained semantic-aware positive graphs based on positively contributing, negatively contributing and unimportant regions calculated by the class-specific gradient of balanced hard negative graphs and original graphs.
- BalanceGCL is evaluated on 14 benchmark datasets from biology, chemistry, and social sciences across unsupervised graph classification and transfer learning tasks, demonstrating its state-of-the-art performance.

Related Work

Graph Contrastive Learning

Numerous works have been devoted to constructing positive graphs in GCL to maximize the similarity of positive pairs. A pioneering work GraphCL (You et al. 2020) generates positives using four types of random graph augmentations, including attribute masking, node dropping, edge perturbation, and subgraph sampling. However, random graph perturbations can cause semantic information loss, degrading GCL performance. To address this, many rule-based augmentations have been developed to preserve key structures and features. For example, (Zhu et al. 2021) used node centrality to highlight key structures and features, assigning lower perturbation probabilities to less important links and features, while (Li et al. 2023) employed motif centrality to identify dense subgraph patterns as important regions. Recently, many works have proposed learning-based augmentations to automatically discover and preserve important regions. For example, (Yin et al. 2022) and (Suresh et al. 2021) used learnable graph generators to model distributions for adaptive node dropping and feature masking. (Li et al. 2022) extracted discriminative rationale features for contrastive learning on rationale-aware views. (Wu et al. 2023) employed a graph generative adversarial network with a view generator and discriminator to learn graph distributions via a min-max game. However, these methods use a binary partitioning strategy dividing graphs into important and

unimportant regions, overlooking negatively contributing regions. This can cause suboptimal augmentation by misleading model predictions if such negative regions are not properly handled.

For the generation of negative graphs, traditional random graph augmentation, like GraphCL (You et al. 2020), may cause false negative graphs. To ensure generated negatives differ in label from the original graph, (Lin et al. 2022) selected negatives from clusters distinct from the original graph's cluster. (Yang et al. 2023b) obtained semantically meaningful positives and negatives using high-confidence clustering results. (Wei et al. 2023) first identified discriminative and irrelevant structures via contrastive learning, then generated diverse augmentations using graph contrastive saliency as positives and negatives. To further improve GCL's discriminative ability, recent works focus on generating hard negative graphs, which have different labels but are highly similar to the original graph. For example, (Luo et al. 2023) produced hard negative samples by utilizing alternating optimization, which aims to increase contrastive loss while reducing redundancy. (Yang et al. 2023a) used a counterfactual mechanism to minimize structural but maximize semantic differences, producing semantically distinct negatives. However, existing GCL methods do not ensure class-balanced negative graph distributions, leading to overfitting on frequent classes and poor discrimination of rare ones.

Graph Counterfactual Explainability

Graph Counterfactual Explainability (GCE) (Prado-Romero et al. 2024) explores how modifications to the input graph affect GNN predictions, helping to interpret the behavior of black-box GNN models under varying conditions. According to (Prado-Romero et al. 2024), GCE methods can be generally categorized into three types, including search, heuristic and learning-based approaches. Search-based methods (Liu et al. 2021; Faber, Moghaddam, and Wattenhofer 2020) identify existing instances in the dataset as counterfactuals for a given input, while heuristic-based approaches (Wellawatte, Seshadri, and White 2022; Abrate and Bonchi 2021) generate counterfactuals by perturbing the original graph to alter its prediction. Formally, given a graph $\mathcal{G} = \mathcal{V}, \mathcal{E}$ with node set $\mathcal{V}$ and edge set $\mathcal{E}$, a valid counterfactual $\mathcal{G}'$ should satisfy $\Phi(\mathcal{G}) \neq \Phi(\mathcal{G}')$, where $\Phi(\cdot)$ is the prediction of a trained model and $\mathcal{G}'$ is a counterfactual example constructed by introducing perturbation. Learning-based approaches employ a data-driven approach to learn proper perturbations during training, then they generate the counterfactual during inference. Among them, perturbation matrix-based methods (Lucic et al. 2022; Tan et al. 2022) are the most popular. These methods typically generate a counterfactual graph $\mathcal{G}'$ for a given graph $\mathcal{G}$ by minimizing the loss: $\min_{\mathcal{G}'} \mathcal{D}_{inst}(\mathcal{G}', \mathcal{G}) - \beta \mathcal{D}_{pred}(\Phi(\mathcal{G}), \Phi(\mathcal{G}'))$, where $\mathcal{D}_{pred}$ measures the prediction difference between $\mathcal{G}$ and $\mathcal{G}'$, $\mathcal{D}_{inst}$ quantifies the perturbation between them, and β balances the two terms. For example, CF-GNNExplainer (Lucic et al. 2022) created an optimal counterfactual example by minimizing both negative log-likelihood ($\mathcal{D}_{pred}$) and element-wise difference ($\mathcal{D}_{inst}$) between the original graph and counterfactual graph. CF^2 (Tan et al. 2022) minimizes the L_0 norm of perturbations on edges and features, while enforcing that the prediction probability of the original class is lower than that of at least one alternative class.

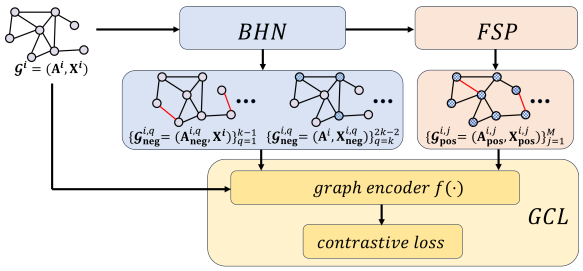

Figure 2: The framework of BalanceGCL.

Methodology

Balanced Hard Negative Graphs Generation

To address the class imbalance in hard negatives, inspired by the counterfactual mechanism, we propose a module called BHN to generate balanced hard negative graphs. The details of the BHN module are illustrated in Fig. 3(a).

Feature and Edge Perturbation. Given a graph $\mathcal{G}^i = (\mathcal{V}^i, \mathcal{E}^i, \mathbf{A}^i, \mathbf{X}^i)$ with node set $\mathcal{V}^i$ and edge set $\mathcal{E}^i$, where $\mathbf{X}^i \in \mathbb{R}^{N^i \times d}$ and $\mathbf{A}^i \in \mathbb{R}^{N^i \times N^i}$ represent its corresponding feature and adjacency matrix. For brevity, we denote $\mathcal{G}^i$ as $\mathcal{G}^i = (\mathbf{A}^i, \mathbf{X}^i)$ in the following. We generate its corresponding negative graphs by introducing two perturbation matrices $\delta_{\mathbf{X}}^{i,q} \in \mathbb{R}^{N^i \times d}$ and $\delta_{\mathbf{A}}^{i,q} \in \mathbb{R}^{N^i \times N^i}$ on features and structures, respectively. Mathematically, the perturbed feature matrix $\mathbf{X}_{\text{neg}}^{i,q}$ and adjacency matrix $\mathbf{A}^{i,q}$ are defined as

$$\mathbf{X}_{\text{neg}}^{i,q} = \mathbf{X}^i + \delta_{\mathbf{X}}^{i,q}, \mathbf{A}^{i,q} = \mathbf{A}^i + \delta_{\mathbf{A}}^{i,q} \tag{1}$$

To ensure a valid graph structure, we apply binarization to each element of $\mathbf{A}^{i,q}$ as:

$$\mathbf{A}_{\text{neg}}^{i,q} = \mathcal{I}\left(\text{sigmoid}(\mathbf{A}^{i,q}) \geq \omega\right), \tag{2}$$

where ω is a predefined threshold. The sigmoid function normalizes $\mathbf{A}^{i,q}$ into the range $[0, 1]$, and $\mathcal{I}(\cdot)$ denotes the indicator function that outputs 1 if the condition holds and 0 otherwise.

Negative Graphs Generation. After introducing perturbations, we aim to constrain these perturbations to generate balanced hard negative graphs that are structurally similar but semantically distinct while maintaining an even distribution across classes. To achieve this goal, we develop a strategy that encourages semantic shifts under explicitly regulating the class-wise distribution of generated negative samples, avoiding overwhelming generation into a few classes. Let k denote the total number of classes, and $g(\cdot) \in \mathbb{R}^k$ represent the pre-trained classifier. Given an original graph $\mathcal{G}^i = (\mathbf{X}^i, \mathbf{A}^i)$ with predicted pseudo label $p^i = \arg\max g(\mathcal{G}^i)$, to ensure class balance among the negative graphs, we generate $k - 1$ hard negative graphs

$\left\{\mathcal{G}_{\text{neg}}^{i,q} \,\middle|\, q=1,2,\ldots,k; q\neq p^i\right\}$ for $\mathcal{G}^i$, where each negative graph is associated with one of the remaining $k-1$ classes excluding the predicted pseudo-class p^i. The construction process is formulated as follows:

$$\begin{aligned}\mathcal{L}_c = \sum_{i=1}^{N}\sum_{\substack{q=1\\ q\neq p^i}}^{k}\Big(&\gamma\big(g_{p^i}(\mathcal{G}_{\text{neg}}^{i,q})^2 - g_q(\mathcal{G}_{\text{neg}}^{i,q})^2\big)\\ &+\frac{1}{|r|}\sum_{j\in r}\big(g_j(\mathcal{G}_{\text{neg}}^{i,q}) - g_j(\mathcal{G}^i)\big)^2\Big),\\ \text{with } p^i = \arg\max\ g(\mathcal{G}^i), &r=\{1,...,k\}\setminus\{p^i,q\},\end{aligned} \tag{3}$$

where $g_q(\mathcal{G}_{\text{neg}}^{i,q})$ denotes the q-th element of the prediction vector $g(\mathcal{G}_{\text{neg}}^{i,q})$, representing the predicted probabilities of assigning the negative sample $\mathcal{G}_{\text{neg}}^{i,q}$ to the q-th class. γ balances the two terms. The first term in Eq. (3) is designed to minimize the probability of assigning $\mathcal{G}_{\text{neg}}^{i,q}$ to the pseudo-label class p^i, while simultaneously maximizing its probability of being assigned to class q. This dual objective ensures that the negative sample is confidently categorized into class q, which is explicitly distinct from the original graph's pseudo label p^i. By using the squared term, the optimization process is accelerated, enabling faster convergence and more efficient model training. The second term in Eq. (3) enforces prediction consistency for the remaining classes, thereby constraining the semantic shift induced by perturbation to occur solely between the pseudo-label p^i and the target class q. By iterating over all remaining $k-1$ classes excluding the class p^i, Eq. (3) ensures that the generated negative samples for each graph are evenly distributed across the remaining classes.

Moreover, to further ensure that these balanced negative samples are hard negatives that are similar to the original graph $\mathcal{G}^i$ with respective to feature and structure, we employ the following formulation:

$$\begin{aligned}\mathcal{L}_{d_{\mathbf{A}}} &= \sum_{i=1}^{N}\sum_{q=1,q\neq p^i}^{k}\left\|\mathbf{A}^i - \mathbf{A}_{\text{neg}}^{i,q}\right\|_2^2\\ \mathcal{L}_{d_{\mathbf{X}}} &= \sum_{i=1}^{N}\sum_{q=1,q\neq p^i}^{k}\left\|\mathbf{X}^i - \mathbf{X}_{\text{neg}}^{i,q}\right\|_2^2\end{aligned} \tag{4}$$

where $p^i = \arg\max\ g(\mathcal{G}^i)$ and $\|\cdot\|_2$ is the L_2 norm. As a result, by combining Eq. (4) and Eq. (3), the total loss for balanced hard negative graphs generation is:

$$\mathcal{L}_{\mathbf{X}} = \alpha\mathcal{L}_{d_{\mathbf{X}}} + \mathcal{L}_c, \mathcal{L}_{\mathbf{A}} = \alpha\mathcal{L}_{d_{\mathbf{A}}} + \mathcal{L}_c, \tag{5}$$

Here, α is a hyperparameter that balances the perturbation term ($\mathcal{L}_{d_{\mathbf{X}}}$ or $\mathcal{L}_{d_{\mathbf{A}}}$) and the semantic constraint loss $\mathcal{L}_c$. By minimizing $\mathcal{L}_{\mathbf{X}}$ and $\mathcal{L}_{\mathbf{A}}$ separately, for each original input graph $\mathcal{G}^i$, we generate $2k-2$ balanced hard negative samples, denoted as $\{\mathcal{G}_{\text{neg}}^{i,q}\}_{q=1}^{2k-2}$, for contrastive training. The first $k-1$ negatives are constructed by perturbing the graph structure, i.e., $\{\mathcal{G}_{\text{neg}}^{i,q} = (\mathbf{A}_{\text{neg}}^{i,q}, \mathbf{X}^i)\}_{q=1}^{k-1}$, while the remaining $k-1$ are generated by perturbing the feature matrix: $\{\mathcal{G}_{\text{neg}}^{i,q} = (\mathbf{A}^i, \mathbf{X}_{\text{neg}}^{i,q})\}_{q=k}^{2k-2}$.

Fine-grained Semantic-aware Positive Graphs Generation

To ensure accurate semantic alignment between the positive graph and the original graph, we propose partitioning the graph into three types of semantically relevant regions: positively contributing, negatively contributing, and unimportant contributing regions. Different perturbation strategies are then applied to each region accordingly. The details of the FSP module are illustrated in Fig. 3(b).

Identification of Semantic-relevant Regions. The balanced hard negative samples generated by the BHN module are structurally similar to the original graph, but are predicted into different classes. This property enables a fine-grained analysis of how perturbations in structure or features can lead to significant semantic shifts in model predictions. Specifically, we first compute the element-wise partial derivative of $g_{p^i}(\mathcal{G}^i)$ with respect to the input feature matrix $\mathbf{X}^i$, defined as $\left(\mathbf{W}_{\text{o}}^{\mathbf{X}^i}\right)_{mn} = \frac{\partial g_{p^i}(\mathcal{G}^i)}{\partial \mathbf{X}_{mn}^i}$. This measures the sensitivity of the model's prediction for class p^i with respect to each input feature $\mathbf{X}_{mn}^i$. Accordingly, the element-wise product $(\mathbf{W}_{\text{o}}^{\mathbf{X}^i})_{mn} \circ \mathbf{X}_{mn}^i$ reflects the actual contribution of each element of original input feature to the prediction of class p^i. This process is formulated as: $\mathbf{R}^{\mathbf{X}^i} = \mathbf{W}_{\text{o}}^{\mathbf{X}^i} \circ \mathbf{X}^i, \mathbf{W}_{\text{o}}^{\mathbf{X}^i} = \frac{\partial g_{p^i}(\mathcal{G}^i)}{\partial \mathbf{X}^i}$, where $\circ$ denotes the Hadamard product. A higher value of $(\mathbf{R}^{\mathbf{X}^i})_{mn}$ indicates a greater contribution of the (m,n)-th feature element to the class p^i. Similarly, the average contribution of each feature element in the $k-1$ feature-perturbed negative samples to the prediction of class p^i can be measured as follows:

$$\mathbf{R}_{\text{neg}}^{\mathbf{X}^i} = \frac{1}{k-1}\sum_{\substack{q=1\\ q\neq p^i}}^{k}\mathbf{W}_{\text{neg}}^{\mathbf{X}^{i,q}} \circ \mathbf{X}_{\text{neg}}^{i,q}, \mathbf{W}_{\text{neg}}^{\mathbf{X}^{i,q}} = \frac{\partial g_{p^i}(\mathcal{G}_{\text{neg}}^{i,q})}{\partial \mathbf{X}_{\text{neg}}^{i,q}},$$

where $\mathcal{G}_{\text{neg}}^{i,q} = (\mathbf{A}^i, \mathbf{X}_{\text{neg}}^{i,q})$ is the q-th feature-perturbed negative sample of the i-th original graph. Next, we partition the features into three different types of regions based on the relative relationship between $\mathbf{R}_{\text{neg}}^{\mathbf{X}^i}$ and $\mathbf{R}^{\mathbf{X}^i}$.

Positively Contributing Region: Since feature-perturbed negative graphs are expected to yield predictions different from that of the original graph (i.e., class p^i), we expect the (m,n)-th feature element to contribute less to class p^i in the negative graphs if it had a positively contribution to class p^i in the original graph. In other words, if $(\mathbf{R}^{\mathbf{X}^i})_{mn}$ is larger, then the corresponding value $(\mathbf{R}_{\text{neg}}^{\mathbf{X}^i})_{mn}$ in the feature-perturbed negative graph should be lower. Therefore, the region contains features that contribute positively to classification of original graph is identified by the following:

$$\mathbf{I}_{\text{pos}}^{\mathbf{X}^i} = \text{ReLU}(\mathbf{R}^{\mathbf{X}^i} - \mathbf{R}_{\text{neg}}^{\mathbf{X}^i} - \theta_{\mathbf{X}}), \tag{6}$$

where $\theta_{\mathbf{X}}$ is a pre-defined threshold. Regions in the original graph are defined as positively contributing to the original category p^i only if its feature contribution to class p^i exceeds that in the corresponding negative graphs by a reasonable threshold.

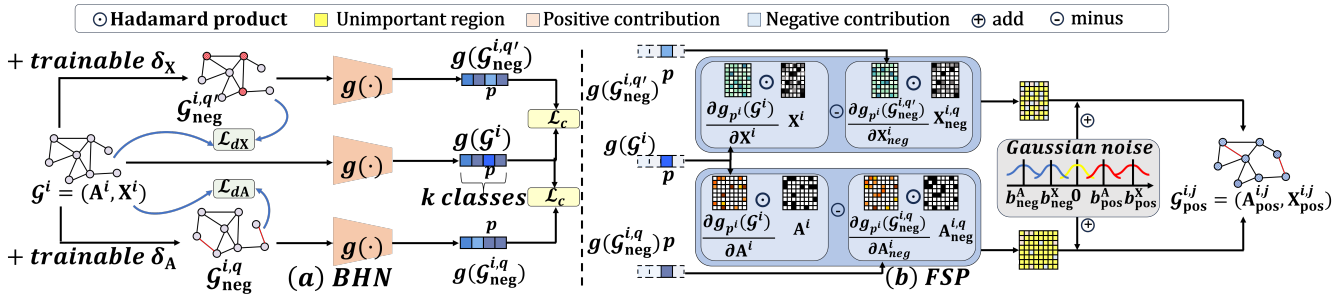

Figure 3: Detailed illustration of BHN and FSP modules. In BHN, perturbations are applied to node features and edges to generate balanced hard negatives. In FSP, gradient analysis identifies positively, negatively, and unimportant contributing regions of the graph, which are then differentially perturbed to ensure semantic consistency.

Negatively Contributing Region: Similarly, if a feature element contributes negatively to the prediction of class p^i in the original graph, we expect its contribution to class p^i to be higher in the corresponding negative graphs than in the original. In other words, a lower value of $(\mathbf{R}^{\mathbf{X}^i})_{mn}$ in the original graph should correspond to a higher value of $(\mathbf{R}^{\mathbf{X}^i}_{\text{neg}})_{mn}$ in the feature-perturbed negative graphs. Based on this principle, regions containing features that negatively influence the classification of the original graph are identified as

$$\mathbf{I}^{\mathbf{X}^i}_{\text{neg}} = \text{ReLU}(\mathbf{R}^{\mathbf{X}^i}_{\text{neg}} - \mathbf{R}^{\mathbf{X}^i} - \theta_{\mathbf{X}}). \tag{7}$$

A region in the original graph is considered to negatively contribute to class p^i if its feature contribution to p^i is significantly lower than that in the corresponding negative graphs, exceeding a predefined threshold.

Unimportant Region: If the contribution of a feature element to class p^i in the negative graph shows negligible difference compared to its contribution in the original graph, the corresponding region is defined as unimportant with respect to the classification decision. It is formalized as

$$\mathbf{I}^{\mathbf{X}^i}_{\text{uni}} = \text{ReLU}\left(\theta_{\mathbf{X}} - \left|\mathbf{R}^{\mathbf{X}^i} - \mathbf{R}^{\mathbf{X}^i}_{\text{neg}}\right|\right). \tag{8}$$

In parallel, we compute the contribution of graph structure, i.e., adjacency matrix $\mathbf{A}^i$, to the class that original graph belongs by a similar strategy,

$$\begin{aligned}
&\mathbf{R}^{\mathbf{A}^i} = \mathbf{W}^{\mathbf{A}^i}_{\text{o}} \circ \mathbf{A}^i, \quad \mathbf{W}^{\mathbf{A}^i}_{\text{o}} = \frac{\partial g_{p^i}(\mathcal{G}^i)}{\partial \mathbf{A}^i}, \\
&\mathbf{R}^{\mathbf{A}^i}_{\text{neg}} = \frac{1}{k-1} \sum_{\substack{q=1 \\ q \neq p^i}}^{k} \mathbf{W}^{\mathbf{A}^{i,q}}_{\text{neg}} \circ \mathbf{A}^{i,q}_{\text{neg}}, \mathbf{W}^{\mathbf{A}^{i,q}}_{\text{neg}} = \frac{\partial g_{p^i}(\mathcal{G}^{i,q}_{\text{neg}})}{\partial \mathbf{A}^{i,q}_{\text{neg}}}. \\
&\mathbf{I}^{\mathbf{A}^i}_{\text{pos}} = \text{ReLU}(\mathbf{R}^{\mathbf{A}^i} - \mathbf{R}^{\mathbf{A}^i}_{\text{neg}} - \theta_{\mathbf{A}}), \\
&\mathbf{I}^{\mathbf{A}^i}_{\text{neg}} = \text{ReLU}(\mathbf{R}^{\mathbf{A}^i}_{\text{neg}} - \mathbf{R}^{\mathbf{A}^i} - \theta_{\mathbf{A}}), \\
&\mathbf{I}^{\mathbf{A}^i}_{\text{uni}} = \text{ReLU}\left(\theta_{\mathbf{A}} - \left|\mathbf{R}^{\mathbf{A}^i} - \mathbf{R}^{\mathbf{A}^i}_{\text{neg}}\right|\right).
\end{aligned} \tag{9}$$

where $\mathbf{I}^{\mathbf{A}^i}_{\text{pos}}$, $\mathbf{I}^{\mathbf{A}^i}_{\text{neg}}$, and $\mathbf{I}^{\mathbf{A}^i}_{\text{uni}}$ denote the positively contributing, negatively contributing, and unimportant regions in the adjacency matrix $\mathbf{A}^i$ with respect to the class label of the original graph, respectively.

Positive Samples Generation. After partitioning the graph into three kinds of regions, to generate positive samples that preserve fine-grained semantics consistent with the original graph, we incorporate promotional, suppressed and random perturbations that amplify the influence of positively contributing regions, suppress the influence of negative ones and randomly perturb unimportant areas, respectively, as follows.

$$\begin{aligned}
\varepsilon^j_{\mathbf{X}^i} &= \mathbf{N}^{\mathbf{X}^i}_{\text{uni}} \circ \mathbf{I}^{\mathbf{X}^i}_{\text{uni}} + \mathbf{N}^{\mathbf{X}^i}_{\text{pos}} \circ \mathbf{I}^{\mathbf{X}^i}_{\text{pos}} + \mathbf{N}^{\mathbf{X}^i}_{\text{neg}} \circ \mathbf{I}^{\mathbf{X}^i}_{\text{neg}} \\
\varepsilon^j_{\mathbf{A}^i} &= \mathbf{N}^{\mathbf{A}^i}_{\text{uni}} \circ \mathbf{I}^{\mathbf{A}^i}_{\text{uni}} + \mathbf{N}^{\mathbf{A}^i}_{\text{pos}} \circ \mathbf{I}^{\mathbf{A}^i}_{\text{pos}} + \mathbf{N}^{\mathbf{A}^i}_{\text{neg}} \circ \mathbf{I}^{\mathbf{A}^i}_{\text{neg}},
\end{aligned} \tag{10}$$

where $\circ$ denotes the Hadamard product. $\mathbf{N}^{\mathbf{X}^i}_{\text{uni}}$ and $\mathbf{N}^{\mathbf{A}^i}_{\text{uni}}$ are independently sampled from the zero-mean Gaussian distribution $\mathcal{N}(0, \sigma_j^2)$; $\mathbf{N}^{\mathbf{X}^i}_{\text{pos}}$ and $\mathbf{N}^{\mathbf{A}^i}_{\text{pos}}$ are sampled from $\mathcal{N}(b^{\mathbf{X}}_{\text{pos}}, \sigma_j^2)$ and $\mathcal{N}(b^{\mathbf{A}}_{\text{pos}}, \sigma_j^2)$, respectively; and $\mathbf{N}^{\mathbf{X}^i}_{\text{neg}}$,$\mathbf{N}^{\mathbf{A}^i}_{\text{neg}}$ are sampled from $\mathcal{N}(b^{\mathbf{X}}_{\text{neg}}, \sigma_j^2)$ and $\mathcal{N}(b^{\mathbf{A}}_{\text{neg}}, \sigma_j^2)$, respectively. Here, $b^{\mathbf{X}}_{\text{pos}} > 0$ and $b^{\mathbf{A}}_{\text{pos}} > 0$ enforce promotional perturbations on positively contributing regions, while $b^{\mathbf{X}}_{\text{neg}} < 0$ and $b^{\mathbf{A}}_{\text{neg}} < 0$ apply suppressive perturbations on negatively contributing regions. Then, by leveraging distinct $\{\sigma_j\}_{j=1}^M$, we can introduce different levels of perturbations to generate M positive samples $\{\mathcal{G}^{i,j}_{\text{pos}} = (\mathbf{A}^{i,j}_{\text{pos}}, \mathbf{X}^{i,j}_{\text{pos}})\}_{j=1}^M$ for the i-th original graph, characterized by

$$\mathbf{X}^{i,j}_{\text{pos}} = \mathbf{X}^i + \varepsilon^j_{\mathbf{X}^i}, \mathbf{A}^{i,j}_{\text{pos}} = \mathcal{I}(\text{sigmoid}(\mathbf{A}^i + \varepsilon^j_{\mathbf{A}^i}) \geq \omega). \tag{11}$$

Here, $\mathbf{X}^{i,j}_{\text{pos}}$ and $\mathbf{A}^{i,j}_{\text{pos}}$ represent the perturbed feature matrix and adjacency matrix of the j-th positive sample, respectively. ω is a predefined threshold, and $\mathcal{I}(\cdot)$ denotes the indicator function, consistent with its definition in Eq. (2).

Contrastive Loss

The goal of GCL is to learn an encoder $f(\cdot) \in \mathbb{R}^h$, which maps each graph into a compact representation for the downstream tasks such as graph classification. After obtaining $\{\mathcal{G}^{i,j}_{\text{pos}}\}_{j=1}^M$ and $\{\mathcal{G}^{i,q}_{\text{neg}}\}_{q=1}^{2k-2}$, we utilize the InfoNCE contrastive loss (Oord, Li, and Vinyals 2018) to train the graph encoder: $\mathcal{L}_{contr} = -\sum_{i=1}^N \sum_{j=1}^M \log \frac{\exp(\text{sim}(\mathbf{z}_i, \mathbf{z}^{i,j}_{\text{pos}})/\tau)}{\sum_{q=1}^{2k-2} \exp(\text{sim}(\mathbf{z}_i, \mathbf{z}^{i,q}_{\text{neg}})/\tau)}$, where $\text{sim}(\cdot,\cdot)$

and τ are the cosine similarity and temperature parameter respectively. $\mathbf{z}_i = f(\mathcal{G}^i)$, $\mathbf{z}_{\text{pos}}^{i,j} = f(\mathcal{G}_{\text{pos}}^{i,j})$ and $\mathbf{z}_{\text{neg}}^{i,q} = f(\mathcal{G}_{\text{neg}}^{i,q})$ are the representation of the i-th anchor graph, the j-th positive graph and the q-th negative graph associated with anchor graph.

Experiments

Comparison with State-of-the-Art Methods

We compare our proposed BalanceGCL with ten state-of-the-art approaches, including three traditional positive sample generation methods, i.e., InfoGraph (Sun et al. 2020), GraphCL (You et al. 2020), and MVGRL (Hassani and Khasahmadi 2020), six advanced positive sample generation methods, i.e., AD-GCL (Suresh et al. 2021), JOAO (You et al. 2021), RGCL (Li et al. 2022), GCS (Wei et al. 2023), DRGCL (Ji et al. 2024), and GPS (Ju et al. 2025) and an advanced negative sample generation method CGC (Yang et al. 2023a). For unsupervised learning, we benchmark BalanceGCL on six established datasets in TU datasets. For transfer learning, we initially perform self-supervised pre-training on the ZINC-2M (Sterling and Irwin 2015) dataset. Afterward, the backbone model is fine-tuned on eight benchmark multi-task binary classification datasets from the biochemistry domain, which are part of MoleculeNet (Wu et al. 2018). To conduct evaluations on datasets without initial node attributes, we develop BalanceGCL-edge, which only applies edge perturbation in Eq. (2). Similarly, we construct BalanceGCL-attr by employing only attribute perturbation as defined in Eq. (1), enabling a systematic comparison between structural and feature perturbation strategies. Details of datasets, evaluation protocols, models, hyperparameters, and baselines are in Section III of the Appendix.

Graph Classification Result Analysis. Table. 1 presents the unsupervised graph-level classification results. The experimental results indicate that BalanceGCL consistently outperforms other state-of-the-art methods. Firstly, compared with traditional methods, such as GraphCL, InfoGraph, and MVGRL, our approach achieves notable improvements. Secondly, BalanceGCL significantly outperforms advanced positive sample generation methods, for example, BalanceGCL improves upon DRGCL by 2.0 %, AD-GCL by 3.2%, RGCL by 3.3%, and GCS by 3.4% on average. This is because these methods use a binary partitioning strategy, which fails to account for the negatively contributing region, resulting in degrading semantic integrity. In contrast, our method BalanceGCL incorporates fine-grained regional partitions in positive sample generation and applies tailored perturbations to each type of regions. Thirdly, BalanceGCL achieves an average performance gain of 3.0% over CGC by explicitly addressing class imbalance in the negative samples through Eq. (3), which enforces balanced attentions across different classes.

Transfer Learning Result Analysis. Since the initial node features in all transfer learning datasets are two-dimensional categorical attributes, we adopt the BalanceGCL-edge variant for pre-training. Table. 2 presents the transfer learning results. The baseline No pre-train, which trains the backbone model directly via supervised multi-task classification without pretraining, yields the lowest overall performance. Our BalanceGCL framework achieves the best performance on 7 out of 8 benchmark datasets, with an average ROC-AUC 1.8% higher than the second-best method. On the BACE dataset, its slightly lower performance is due to the nearly balanced class distribution (around 1:1), where the benefits of BalanceGCL, which is designed for class-imbalanced scenarios, are less pronounced. These results validate that more discriminative backbone model can be achieved by generating fine-grained semantic-aware positive and balanced hard negative graphs, which is essential for effective knowledge transfer to downstream tasks.

without node attr			
Model	**RDT-M5K**	**IMDB-M**	**COLLAB**
InfoGraph	53.44 ± 1.4	49.67 ± 1.1	73.84 ± 1.7
GraphCL	$\underline{56.81 \pm 1.5}$	50.80 ± 0.3	74.60 ± 1.5
MVGRL	52.03 ± 1.6	50.20 ± 0.3	$\underline{78.44 \pm 1.4}$
AD-GCL	55.65 ± 0.9	50.47 ± 0.4	74.38 ± 1.8
JOAO	56.43 ± 1.2	50.36 ± 0.4	73.22 ± 2.0
RGCL	56.69 ± 1.3	50.21 ± 0.2	73.28 ± 1.9
GCS	55.13 ± 1.4	48.82 ± 0.2	74.23 ± 1.4
DRGCL	56.71 ± 1.8	50.63 ± 0.3	72.96 ± 2.0
GPS	56.30 ± 1.2	$\underline{51.07 \pm 0.4}$	74.19 ± 1.6
BalanceGCL-edge	$\mathbf{57.41 \pm 1.3}$	$\mathbf{51.70 \pm 0.3}$	$\mathbf{78.73 \pm 1.9}$
with node attr			
Model	**ENZYMES**	**COLORS-3**	**Synthie**
InfoGraph	56.74 ± 0.7	73.74 ± 0.6	62.25 ± 0.6
GraphCL	53.54 ± 1.9	72.90 ± 0.7	69.75 ± 0.8
MVGRL	56.21 ± 1.3	77.23 ± 0.7	74.05 ± 0.7
AD-GCL	54.16 ± 0.7	73.51 ± 0.9	75.10 ± 0.7
JOAO	52.89 ± 1.3	76.83 ± 0.9	64.50 ± 1.0
RGCL	55.45 ± 0.9	77.26 ± 0.8	70.20 ± 0.5
CGC	54.84 ± 1.1	75.13 ± 0.8	74.25 ± 0.7
GCS	56.12 ± 0.8	76.58 ± 1.3	71.55 ± 0.9
DRGCL	56.33 ± 0.4	$\underline{78.90} \pm 0.8$	74.85 ± 0.4
GPS	55.73 ± 0.8	77.64 ± 1.0	72.65 ± 0.7
BalanceGCL-attr	$\underline{59.68 \pm 0.6}$	77.51 ± 1.0	77.45 ± 0.7
BalanceGCL-edge	58.37 ± 0.6	78.86 ± 0.7	$\underline{77.60 \pm 0.6}$
BalanceGCL	$\mathbf{60.29 \pm 0.5}$	$\mathbf{80.18 \pm 0.9}$	$\mathbf{78.75 \pm 0.6}$

Table 1: Unsupervised graph-level classification accuracy (%) on TU datasets. Bold and underline denote the best and second best performance respectively.

Ablation Studies

We conduct ablation studies to evaluate the effectiveness of BalanceGCL by designing the following variants: (1) **w/o BHN module**: This variant selects $2k - 2$ random samples from the dataset as negative samples for each graph. (2) **w/o class balance**: This variant model adopts $\mathcal{L}_c = -\sum_{i=1}^{N} \sum_{q=1, q\neq p^i}^{k} D_{\text{KL}}\left(p(\mathcal{G}^i), p(\mathcal{G}_{\text{neg}}^{i,q})\right)$ from CGC (Yang et al. 2023a) to replace $\mathcal{L}_c$ in Eq. (5). (3) **w/o FSP module**: This variant replaces the FSP module with random perturbations used in GraphCL (You et al. 2020), applying a perturbation rate of 0.2. (4) **w/o positively contributing region**: This variant eliminates the positively contributing regions by setting $\mathbf{I}_{\text{pos}}^{\mathbf{X}^i}$ and $\mathbf{I}_{\text{pos}}^{\mathbf{A}^i}$ in Eq. (10) as all-zero matrix. (5) **w/o negatively contributing region**: This variant

Model	BBBP	Tox21	ToxCast	SIDER	ClinTox	MUV	HIV	BACE	avg.
No Pre-Train	65.8 ± 4.5	74.0 ± 0.8	63.4 ± 0.6	57.3 ± 1.6	58.0 ± 4.4	71.8 ± 2.5	75.3 ± 1.9	70.1 ± 5.4	66.9
InfoGraph	68.5 ± 0.6	74.5 ± 0.6	63.2 ± 0.5	59.7 ± 0.8	77.3 ± 3.1	74.6 ± 1.7	76.9 ± 0.9	77.7 ± 0.9	71.5
GraphCL	69.7 ± 0.7	73.9 ± 0.7	62.4 ± 0.6	60.5 ± 0.9	76.0 ± 2.7	69.8 ± 2.7	78.5 ± 1.2	75.4 ± 1.4	70.8
MVGRL	69.0 ± 0.5	74.5 ± 0.6	62.6 ± 0.5	62.6 ± 0.6	77.8 ± 2.2	73.3 ± 1.4	77.1 ± 0.6	77.2 ± 1.0	71.8
AD-GCL	70.0 ± 1.1	$\underline{76.5 \pm 0.8}$	63.1 ± 0.7	$\underline{63.3 \pm 0.8}$	79.8 ± 3.5	72.3 ± 1.6	78.3 ± 1.0	78.5 ± 0.8	71.5
JOAO	70.2 ± 1.0	75.0 ± 0.3	62.9 ± 0.5	60.0 ± 0.8	81.3 ± 2.5	71.6 ± 1.4	76.7 ± 1.2	77.3 ± 0.5	71.9
RGCL	71.4 ± 0.7	75.2 ± 0.2	63.3 ± 0.2	61.4 ± 0.6	$\underline{83.4 \pm 0.9}$	$\underline{76.7 \pm 1.0}$	77.9 ± 0.8	76.0 ± 0.8	$\underline{73.2}$
GCS	70.4 ± 0.5	74.7 ± 0.3	63.7 ± 0.2	60.4 ± 0.4	80.7 ± 1.9	75.9 ± 1.4	78.2 ± 1.0	77.5 ± 0.4	72.7
DRGCL	71.2 ± 0.5	74.7 ± 0.5	64.0 ± 0.5	61.1 ± 0.8	78.2 ± 1.5	73.8 ± 1.1	$\underline{78.6 \pm 1.0}$	78.2 ± 1.0	72.5
GPS	$\underline{71.5 \pm 0.9}$	-	$\underline{64.4 \pm 0.3}$	-	82.1 ± 2.9	75.6 ± 1.7	79.0 ± 1.1	$\mathbf{80.1 \pm 0.8}$	-
BalanceGCL-edge	$\mathbf{72.8 \pm 0.7}$	$\mathbf{76.8 \pm 0.5}$	$\mathbf{64.9 \pm 0.4}$	$\mathbf{64.4 \pm 0.7}$	$\mathbf{83.9 \pm 1.9}$	$\mathbf{79.1 \pm 1.2}$	$\mathbf{79.8 \pm 0.8}$	$\underline{79.4 \pm 0.8}$	**75.0**

Table 2: Transfer learning performance of molecular property prediction on ZINC-2M (mean ROC-AUC (%) + std over 10 runs. Bold and Underline denote the best and second best performance respectively.

Variation	ENZYMES	COLORS-3	Synthie
(1) w/o BHN module	56.79 ± 0.5	75.53 ± 0.8	74.35 ± 0.7
(2) w/o class balance	57.34 ± 0.6	76.61 ± 0.9	76.20 ± 0.6
(3) w/o FSP module	56.65 ± 0.7	75.40 ± 0.8	75.65 ± 0.5
(4) w/o positively contributing region	57.86 ± 0.6	77.31 ± 1.0	76.85 ± 0.6
(5) w/o negatively contributing region	58.84 ± 0.6	76.98 ± 0.9	77.50 ± 0.5
(6) w/o unimportant region	59.32 ± 0.6	79.39 ± 1.0	77.80 ± 0.7
BalanceGCL	60.29 ± 0.5	80.18 ± 0.9	78.75 ± 0.6

Table 3: Ablation study results for BalanceGCL on three datasets, the accuracy of classification results is reported.

removes the negatively contributing regions by setting $\mathbf{I}_{\text{neg}}^{\mathbf{X}^i}$ and $\mathbf{I}_{\text{neg}}^{\mathbf{A}^i}$ in Eq. (10) as all-zero matrix. (6) **w/o unimportant region**: This variant removes the unimportant regions by setting $\mathbf{I}_{\text{uni}}^{\mathbf{X}^i}$ and $\mathbf{I}_{\text{uni}}^{\mathbf{A}^i}$ in Eq. (10) as all-zero matrix.

The results of ablation studies are shown in Table 3. First, replacing balanced hard negatives with random perturbations (**w/o BHN module**) significantly degrades performance across all datasets, highlighting the BHN module's importance. Additionally, removing class balance (**w/o class balance**) also causes notable drops, confirming that class-balanced negative graphs are essential for improving GCL discrimination. Secondly, we compare our method with four ablated variants: w/o FSP module, w/o positively contributing region, w/o negatively contributing region, and w/o unimportant region. Removing the entire FSP module causes the most significant performance drop, confirming its crucial role in preserving fine-grained semantics during positive sample generation. Excluding any of the three region types also leads to performance degradation, underscoring that jointly modeling positive, negative, and unimportant regions is essential for optimal performance. Finally, we evaluate the impact of feature and structure perturbations by comparing BalanceGCL-attr and BalanceGCL-edge in Table. 1. Results show that removing either perturbation degrades performance, with attribute perturbation having a slightly greater impact. This highlights the complementary roles of both in generating balanced hard negative graphs.

Visualization Analysis

We perform a visualization analysis on the Tox21 dataset for toxic substance classification. Specifically, we visualize the contribution of each bond (edge) to toxicity prediction by highlighting positively contributing, negatively contributing, and unimportant regions identified by BalanceGCL. These

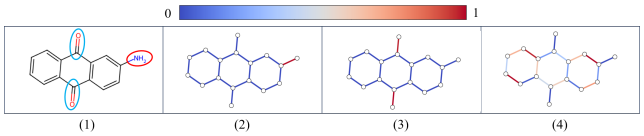

Figure 4: Visualization of semantically relevant regions on the Tox21. (1): the ground-truth molecular structures with toxic substructures highlighted in red and substructures that reduce toxicity marked in blue; (2)-(4): positively contributing, negatively contributing, and unimportant regions identified by BalanceGCL.

correspond to functional groups promoting toxicity, inhibiting toxicity, or having neutral impact, respectively. More details about this experiment are provided in Section III-E of the Appendix. As shown in Fig. 4, BalanceGCL reveals chemically meaningful regions consistent with known molecular substructures. This highlights BalanceGCL can accurately identify semantically aligned structural patterns.

Conclusion

We propose BalanceGCL to address two key limitations in existing GCL methods: (1) the coarse binary partition of semantic regions in positive graphs generation, and (2) the imbalanced distribution of hard negative graphs. BalanceGCL contains two modules: (1) BHN: This module generates class-balanced hard negative graphs based on counterfactual perturbations and a semantic loss. (2) FSP: This module generates fine-grained semantic-aware positive samples by identifying three types of semantic-contributing regions through comparisons between the original and balanced hard negative graphs. Comprehensive experiments on benchmark datasets demonstrate the superiority of BalanceGCL. Our future work will focus on improving the efficiency of generating positive and negative graphs.

Acknowledgments

This work is partly supported by the National Science and Technology Major Project (2024YFF1206600), the National Natural Science Foundation of China (62306118, 62325204), the Fundamental Research Funds for the Central Universities (2025ZYGXZR054).

References

Abrate, C.; and Bonchi, F. 2021. Counterfactual graphs for explainable classification of brain networks. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2495–2504.

Faber, L.; Moghaddam, A. K.; and Wattenhofer, R. 2020. Contrastive Graph Neural Network Explanation. In *Proceedings of the 37th Graph Representation Learning and Beyond Workshop at ICML 2020*, 28. International Conference on Machine Learning.

Hassani, K.; and Khasahmadi, A. H. 2020. Contrastive Multi-View Representation Learning on Graphs. In III, H. D.; and Singh, A., eds., *Proceedings of the 37th International Conference on Machine Learning*, volume 119, 4116–4126. PMLR.

Ji, Q.; Li, J.; Hu, J.; Wang, R.; Zheng, C.; and Xu, F. 2024. Rethinking Dimensional Rationale in Graph Contrastive Learning from Causal Perspective. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(11): 12810–12820.

Ju, W.; Gu, Y.; Mao, Z.; Qiao, Z.; Qin, Y.; Luo, X.; Xiong, H.; and Zhang, M. 2025. GPS: graph contrastive learning via multi-scale augmented views from adversarial pooling. *Science China Information Sciences*, 68(1).

Li, S.; Wang, X.; Zhang, A.; Wu, Y.; He, X.; and Chua, T.-S. 2022. Let Invariant Rationale Discovery Inspire Graph Contrastive Learning. In Chaudhuri, K.; Jegelka, S.; Song, L.; Szepesvari, C.; Niu, G.; and Sabato, S., eds., *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, 13052–13065. PMLR.

Li, W.-Z.; Wang, C.-D.; Lai, J.-H.; and Yu, P. S. 2023. Towards effective and robust graph contrastive learning with graph autoencoding. *IEEE Transactions on Knowledge and Data Engineering*, 36(2): 868–881.

Lin, S.; Liu, C.; Zhou, P.; Hu, Z.-Y.; Wang, S.; Zhao, R.; Zheng, Y.; Lin, L.; Xing, E.; and Liang, X. 2022. Prototypical graph contrastive learning. *IEEE Transactions on Neural Networks and Learning Systems*, 35(2): 2747–2758.

Liu, H.; Huang, Y.; Liu, X.; and Deng, L. 2022. Attention-wise masked graph contrastive learning for predicting molecular property. *Briefings in Bioinformatics*, 23(5): bbac303.

Liu, Y.; Chen, C.; Liu, Y.; Zhang, X.; and Xie, S. 2021. Multi-objective explanations of GNN predictions. In *2021 IEEE International Conference on Data Mining (ICDM)*, 409–418. IEEE.

Lucic, A.; Ter Hoeve, M. A.; Tolomei, G.; De Rijke, M.; and Silvestri, F. 2022. CF-GNNExplainer: Counterfactual Explanations for Graph Neural Networks. In Camps-Valls, G.; Ruiz, F. J. R.; and Valera, I., eds., *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, 4499–4511. PMLR.

Luo, X.; Ju, W.; Gu, Y.; Mao, Z.; Liu, L.; Yuan, Y.; and Zhang, M. 2023. Self-supervised graph-level representation learning with adversarial contrastive learning. *ACM Transactions on Knowledge Discovery from Data*, 18(2): 1–23.

Oord, A. v. d.; Li, Y.; and Vinyals, O. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Pinheiro, P. d. S. M.; Franco, L. S.; and Fraga, C. A. M. 2023. The magic methyl and its tricks in drug discovery and development. *Pharmaceuticals*, 16(8): 1157.

Prado-Romero, M. A.; Prenkaj, B.; Stilo, G.; and Giannotti, F. 2024. A survey on graph counterfactual explanations: definitions, methods, evaluation, and research challenges. *ACM Computing Surveys*, 56(7): 1–37.

Qiu, J.; Chen, Q.; Dong, Y.; Zhang, J.; Yang, H.; Ding, M.; Wang, K.; and Tang, J. 2020. GCC: Graph Contrastive Coding for Graph Neural Network Pre-Training. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, 1150–1160. New York, NY, USA: Association for Computing Machinery. ISBN 9781450379984.

Sterling, T.; and Irwin, J. J. 2015. ZINC 15–ligand discovery for everyone. *Journal of Chemical Information and Modeling*, 55(11): 2324–2337.

Sun, F.; Hoffmann, J.; Verma, V.; and Tang, J. 2020. InfoGraph: Unsupervised and Semi-supervised Graph-Level Representation Learning via Mutual Information Maximization. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Suresh, S.; Li, P.; Hao, C.; and Neville, J. 2021. Adversarial graph augmentation to improve graph contrastive learning. *Advances in Neural Information Processing Systems*, 34: 15920–15933.

Tan, J.; Geng, S.; Fu, Z.; Ge, Y.; Xu, S.; Li, Y.; and Zhang, Y. 2022. Learning and Evaluating Graph Neural Network Explanations based on Counterfactual and Factual Reasoning. In *Proceedings of the ACM Web Conference 2022*, WWW '22, 1018–1027. New York, NY, USA: Association for Computing Machinery. ISBN 9781450390965.

Tian, Y.; Sun, C.; Poole, B.; Krishnan, D.; Schmid, C.; and Isola, P. 2020. What makes for good views for contrastive learning? *Advances in Neural Information Processing Systems*, 33: 6827–6839.

Wei, C.; Wang, Y.; Bai, B.; Ni, K.; Brady, D. J.; and Fang, L. 2023. Boosting graph contrastive learning via graph contrastive saliency. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Wellawatte, G. P.; Seshadri, A.; and White, A. D. 2022. Model agnostic generation of counterfactual explanations for molecules. *Chemical Science*, 13(13): 3697–3705.

Wu, C.; Wang, C.; Xu, J.; Liu, Z.; Zheng, K.; Wang, X.; Song, Y.; and Gai, K. 2023. Graph Contrastive Learning with Generative Adversarial Network. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '23, 2721–2730. New York, NY, USA: Association for Computing Machinery. ISBN 9798400701030.

Wu, Z.; Ramsundar, B.; Feinberg, E. N.; Gomes, J.; Geniesse, C.; Pappu, A. S.; Leswing, K.; and Pande, V. 2018. MoleculeNet: a benchmark for molecular machine learning. *Chemical Science*, 9(2): 513–530.

Yang, H.; Chen, H.; Zhang, S.; Sun, X.; Li, Q.; Zhao, X.; and Xu, G. 2023a. Generating Counterfactual Hard Negative Samples for Graph Contrastive Learning. In *Proceedings of the ACM Web Conference 2023*, WWW '23, 621–629. New York, NY, USA: Association for Computing Machinery. ISBN 9781450394161.

Yang, X.; Liu, Y.; Zhou, S.; Wang, S.; Tu, W.; Zheng, Q.; Liu, X.; Fang, L.; and Zhu, E. 2023b. Cluster-Guided Contrastive Graph Clustering Network. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(9): 10834–10842.

Yin, Y.; Wang, Q.; Huang, S.; Xiong, H.; and Zhang, X. 2022. AutoGCL: Automated Graph Contrastive Learning via Learnable View Generators. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(8): 8892–8900.

You, Y.; Chen, T.; Shen, Y.; and Wang, Z. 2021. Graph Contrastive Learning Automated. In Meila, M.; and Zhang, T., eds., *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, 12121–12132. PMLR.

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

Zhu, Y.; Xu, Y.; Yu, F.; Liu, Q.; Wu, S.; and Wang, L. 2021. Graph Contrastive Learning with Adaptive Augmentation. In *Proceedings of the Web Conference 2021*, WWW '21, 2069–2080. New York, NY, USA: Association for Computing Machinery. ISBN 9781450383127.