Pattern Recognition 179 (2026) 113714

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

NeRS: Negative Relational Smoothing for Graph Contrastive Learning

Sheng Wan [a], Shougang Ren [a,*], Zicheng Zhao [b], Yan Zhu [a,*], Chen Gong [c]

[a] College of Artificial Intelligence, Nanjing Agricultural University, Nanjing, 211800, Jiangsu, China
[b] PCA Lab, Key Lab of Intelligent Perception and Systems for High-Dimensional Information of Ministry of Education, and Jiangsu Key Lab of Image and Video Understanding for Social Security, School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, 210094, Jiangsu, China
[c] School of Automation and Intelligent Sensing, Shanghai Jiao Tong University, Shanghai, 200240, China

ARTICLE INFO

Keywords:
Graph contrastive learning
Self-supervised learning
Representation learning
Graph heterophily

ABSTRACT

Graph Contrastive Learning (GCL) has emerged as a promising paradigm for self-supervised graph representation learning. However, existing GCL methods typically emphasize low-frequency signals, limiting their effectiveness in heterophilic graphs. To cope with graph heterophily, recent works mainly rely on the design of graph filters or the learning of graph structures. Nevertheless, these strategies are applied uniformly across the entire graph without distinguishing between homophilic and heterophilic relations. This could become problematic in real-world scenarios, where homophilic and heterophilic substructures often coexist. To address this challenge, we propose a new GCL framework with Negative Relational Smoothing (NeRS), where contrastive representation learning is performed based on the relational information contained in the graph. Specifically, our NeRS formulates representation learning as the optimization of a noise-robust contrastive objective. Here, a negative relational smoothing strategy is introduced to suppress the influence of noisy contrastive pairs induced by heterophilic relations, which enhances the robustness to varying levels of heterophily. To further improve the expressive power, we design a multi-view architecture that facilitates the construction of reliable contrastive signals under both homophilic and heterophilic substructures. Extensive experiments on multiple benchmark datasets demonstrate the effectiveness of our proposed NeRS.

1. Introduction

Graph representation learning [1] has become a fundamental technique for modeling and understanding large-scale relational data, which empowers applications such as recommender systems [2], economic systems [3], and prognostics and health management [4,5]. Due to the high cost of manual annotations, self-supervised graph representation learning has attracted increasing attention in recent years. Among them, Graph Contrastive Learning (GCL) has become a prominent paradigm for learning expressive representations without relying on label information [6].

The main goal of GCL is to learn informative graph representations via maximizing the agreement between different augmented views of the same graph while minimizing the agreement between views of different graphs. Most existing GCL techniques are built upon the homophily assumption, which implies that nodes connected by an edge tend to share similar class labels. However, in many practical scenarios, graphs may exhibit heterophily, where connected nodes often belong to different classes. For example, in economic networks, interactions often occur between nodes with different roles, such as cities with different industrial structures, which naturally leads to heterophilic connections [7,8]. Since the existing GCL methods typically employ low-pass filters as graph encoders, which aim to smooth the node representations within local neighborhood, their performance can be degraded significantly on heterophilic graphs. This degradation arises primarily due to the indiscriminate feature aggregation, which leads to feature ambiguity and ultimately impairs the discriminative power of the learned representations. Therefore, handling heterophily is critical for accurately modeling real-world networks, as it allows the learned representations to capture meaningful interactions among dissimilar nodes.

To improve the performance of GCL on heterophilic graphs, recent works have explored structure-aware augmentation strategies to construct more informative contrastive views [9]. Unlike random perturbations that may disrupt rare but important homophilic connections, these methods incorporate structural priors, such as feature similarity and local assortativity, to preserve the underlying class semantics [10]. Another line of research aims to enhance the implicit homophily by reshaping the graph structure or mitigating feature ambiguity during training [11]. Despite the refinement to augmentation strategies or graph structures, these approaches typically rely on low-pass filtering,

* Corresponding authors.
E-mail addresses: wansheng315@hotmail.com (S. Wan), rensg@njau.edu.cn (S. Ren), yanzhu@njau.edu.cn (Y. Zhu).

https://doi.org/10.1016/j.patcog.2026.113714
Received 22 December 2025; Received in revised form 25 March 2026; Accepted 10 April 2026
Available online 13 April 2026

which performs neighborhood aggregation without distinction, thereby still suffering from feature ambiguity in heterophilic settings. Inspired by the success of spectral filters in supervised graph representation learning, recent GCL methods have incorporated frequency-aware filtering mechanisms that jointly utilize high-pass and low-pass filters to capture different frequency components of graph signals [12]. However, the employed filters are typically applied uniformly at the graph level. In practice, real-world graphs contain both homophilic and heterophilic relations, with substructures exhibiting varying degrees of heterophily or homophily. Therefore, the filtering-based techniques fail to accurately model the relational information, which limits their effectiveness in GCL.

Different from the filtering-based techniques, in this paper, we propose a new GCL framework with Negative Relational Smoothing (NeRS), which formulates representation learning as the optimization of a noise-robust contrastive objective. Here, contrastive signals are constructed directly from local relations, treating node pairs connected by edges as positives. As the existence of heterophilic relations may result in noisy positives, we develop a negative relational smoothing strategy that models relational noise by assigning negative smoothing coefficients to unreliable contrastive pairs. By learning from the negatively smoothed relational distribution, NeRS emphasizes the clean contrastive pairs while suppressing the influence of the noisy ones, thereby achieving robustness to varying levels of heterophily. To further enhance the expressive power of NeRS across homophilic and heterophilic substructures, we build a multi-view architecture by jointly leveraging one-hop and two-hop neighbors. The intuition is that one-hop neighbors tend to preserve class-consistent information in homophilic settings, whereas two-hop neighbors are often effective under heterophily [13]. More detailed analysis is available in Section 4.5.1. This multi-view architecture facilitates the generation of reliable contrastive signals, which enables the learning of expressive representations in both homophilic and heterophilic substructures. Moreover, we provide a theoretical analysis showing that the devised NeRS loss serves as an upper bound on the clean cross-entropy objective, thereby ensuring robustness under heterophily.

2. Related work

In this section, we review some representative works on graph learning with heterophily and GCL, as they are closely related to this article.

2.1. Graph learning with heterophily

Graph-structured data are widely used to model complex relationships among entities in numerous real-world domains [14,15]. For example, in economic systems [16,17], causal relationships among markets and regions have been investigated using econometric and graph-based models. Besides, in industrial prognostics and health management [18], complex temporal dependencies are often modeled for accurate remaining useful life prediction. Recently, Graph Neural Networks (GNNs) [19,20] have emerged as a powerful approach for learning over graph-structured data. The success of GNNs is largely attributed to the homophily assumption, which refers to the tendency of connected nodes to exhibit similar labels or attributes. However, in many practical scenarios, graphs exhibit heterophily, where connected nodes may belong to different classes or have dissimilar attributes [21]. Under this circumstance, conventional GNNs often suffer performance degradation, as aggregating heterophilic features can introduce noise and distort the learned representations.

To address this challenge, a growing trend of research has focused on developing supervised learning methods for heterophilic graphs. One prominent line of research enhances dissimilarity modeling by employing high-pass filters or signed message passing to preserve heterophilic patterns [22]. Another line exploits global information or high-order neighbors to mitigate the limitations of local aggregation [23]. In parallel, several studies have proposed advanced architectures incorporating novel learning strategies [24]. Other methods adopt graph structure learning [25] and graph Transformer [26] to address the heterophily problem. In recent years, unsupervised learning methods have also emerged as a promising direction for handling heterophilic graphs. Different approaches have been proposed to decouple or reweight neighborhood aggregation to mitigate the adverse effects of heterophilic edges [12]. In addition, other techniques incorporate high-frequency signal modeling [27] or edge-level discrimination [28].

2.2. Graph contrastive learning

In recent years, GCL has emerged as a promising self-supervised paradigm to address the scarcity of labels in graph representation learning [29]. By maximizing agreement between augmented views of the same graph, GCL enables the extraction of informative and discriminative representations without reliance on manual annotations.

In GCL, graph augmentation plays a pivotal role in constructing semantically meaningful views for contrastive objectives. Various techniques have been designed to generate augmented graph views, which can be broadly categorized into rule-based and learning-based approaches. Rule-based methods apply stochastic perturbations, such as random edge or node dropping, feature masking, and subgraph sampling, as seen in GraphCL [30] and GRACE [31]. Meanwhile, GCC [32] utilizes ego-network sampling to preserve local topological semantics, while MVGRL [33] employs graph diffusion to encode global structural patterns. Differently, learning-based augmentation strategies generate augmented views in a data-driven manner through optimization-based objectives. For instance, AutoGCL [34] and SUBLIME [35] adopt structure learning to refine or reconstruct graph topologies, while AD-GCL [36] leverages adversarial training to craft informative and challenging views that enhance contrastive supervision. In addition to graph augmentation, the contrastive mode (*i.e.*, the level at which contrast is performed) also plays a critical role in GCL. Contrastive modes are commonly categorized into intra-scale and inter-scale contrast, depending on whether the paired representations belong to the same level of granularity. Intra-scale contrast aligns embeddings at the same scale, such as node-node or graph-graph pairs [30,31]. On the other hand, inter-scale contrast enhances representation consistency across hierarchical levels of the graph [33,37].

Recent studies in industrial prognostics and health management have shown that advanced representation learning methods are increasingly required to address limited labels, interpretability, uncertainty, and cross-domain shift in remaining useful life prediction [4,5]. In particular, graph-based modeling has been employed to capture complex structural dependencies in degradation data under federated settings, which further highlights the broad significance of developing reliable graph representation learning methods for challenging real-world scenarios.

Despite recent progress, many GCL methods utilize globally shared graph filtering operations to perform representation updating, which limits their effectiveness on real-world graphs containing both homophilic and heterophilic relations. To circumvent this limitation, we formulate representation learning as the optimization of a noise-robust contrastive objective that enables accurate modeling of relational information.

3. Preliminaries

In this section, we introduce the notations and the basic concepts of GCL, followed by a formal description of homophily.

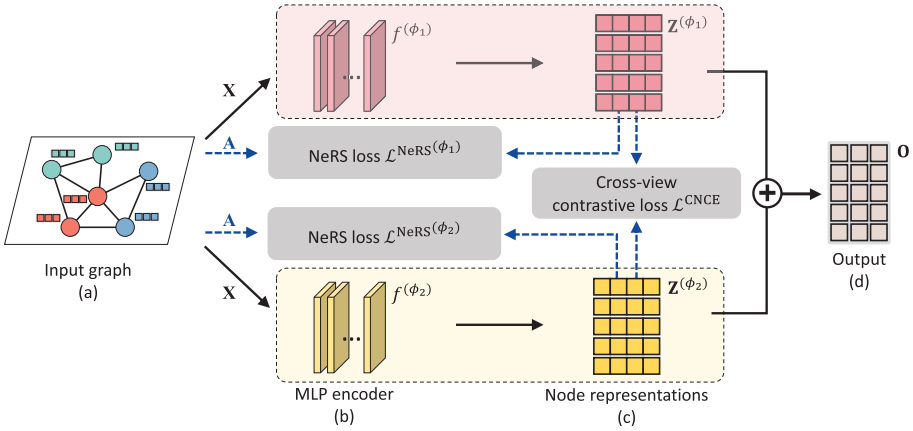


Fig. 1. The conceptual framework of our proposed method. In (a), the node features of the input graph are fed into (b) the MLP encoders in two views (*i.e.*, $f^{(\phi_1)}$ and $f^{(\phi_2)}$), respectively. Afterwards, in (c), the node representations $\mathbf{Z}^{(\phi_1)}$ and $\mathbf{Z}^{(\phi_2)}$ are generated from $f^{(\phi_1)}$ and $f^{(\phi_2)}$ correspondingly, which are then utilized to constitute the NeRS loss based on the relational information. Meanwhile, the cross-view contrastive loss is formed between $\mathbf{Z}^{(\phi_1)}$ and $\mathbf{Z}^{(\phi_2)}$. In (d), the final output is obtained by integrating the representations $\mathbf{Z}^{(\phi_1)}$ and $\mathbf{Z}^{(\phi_2)}$.

3.1. Notations and problem description

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ denote an undirected graph, where $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges, $\mathcal{V} = \{v_1, v_2, \ldots, v_n\}$ is the set of n nodes, and $\mathbf{X} \in \mathbb{R}^{n \times d}$ is the node feature matrix with $\mathbf{x}_i \in \mathbb{R}^d$ representing the feature vector of the ith node v_i. The adjacency matrix of $\mathcal{G}$ is denoted as $\mathbf{A} \in \{0,1\}^{n \times n}$, where $\mathbf{A}_{ij} = 1$ if there exists an edge between v_i and v_j, and $\mathbf{A}_{ij} = 0$ otherwise. The objective of GCL is to learn an encoder $f : \mathcal{G} \to \mathbb{R}^{n \times h}$ that maps each node v_i to the low-dimensional representation $\mathbf{z}_i \in \mathbb{R}^h$. The learned representations can subsequently be utilized for downstream tasks, such as node classification.

3.2. Homophily

Homophily refers to the tendency of nodes to connect with others that share the same class label. In graph learning, the homophily assumption underlies the success of many GNN models, where node representations are updated by aggregating information from neighbors. To quantitatively characterize the degree of homophily, in this paper, we adopt the edge homophily ratio [23], which is simply the fraction of homophilic edges, as the evaluation metric:

$$\mathcal{H} = \frac{\left|\left\{(v_i, v_j) \in \mathcal{E} \mid y_i = y_j\right\}\right|}{|\mathcal{E}|}, \tag{1}$$

where y_i denotes the class label of v_i and $|\mathcal{E}|$ is the number of edges. The edge homophily ratio measures the proportion of edges that connect nodes sharing the same class label. It provides a simple yet effective indicator of the extent to which a graph adheres to the homophily assumption.

4. Method

This section details our proposed NeRS (see Fig. 1). Specifically, we illustrate the critical components of NeRS by elaborating the negative relational smoothing strategy, presenting the representation enhancement with multi-view establishment, explaining the overall training procedure, and providing the theoretical analysis that ensures robustness of the objective.

4.1. GCL setup

Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, GCL constructs multiple augmented views of the graph and aims to maximize agreement between positive pairs while minimizing agreement between negative pairs. A common training objective in GCL is the InfoNCE loss, which can be simply presented as

$$\mathcal{L}^{\text{NCE}} = -\sum_{v_i \in \mathcal{V}' \cup \mathcal{V}''} \log \frac{e^{s_{ii^+}/\tau}}{e^{s_{ii^+}/\tau} + \sum_{v_j \in \mathcal{N}_i^-} e^{s_{ij}/\tau}}, \tag{2}$$

where $\mathcal{V}'$ and $\mathcal{V}''$ denote the sets of nodes in two augmented views, $s_{ii^+} = \text{sim}(\mathbf{z}_i, \mathbf{z}_i^+)$ denotes the similarity between the representations of node v_i and its positive counterpart v_i^+, $s_{ij} = \text{sim}(\mathbf{z}_i, \mathbf{z}_j)$ represents the similarity between the representations of v_i and v_j, and τ is the temperature parameter. Here, v_i^+ is obtained from the other augmented view of v_i, $\mathcal{N}_i^- = \left(\mathcal{V}' \cup \mathcal{V}''\right) \setminus \{v_i, v_i^+\}$ represents the set of negatives of v_i, and the similarity function $\text{sim}(\cdot,\cdot)$ is often implemented as cosine similarity. In most existing GCL methods, node representations are obtained through graph encoders that perform feature aggregation via graph filtering. However, the employed filters are typically defined by globally shared spectral parameters, which limits their capacity when applied to substructures with varying levels of homophily within the same graph.

4.2. Learning with negative relational smoothing

In this paper, we propose a new GCL framework that enables expressive representation learning across both homophilic and heterophilic substructures. Unlike conventional GCL methods, which rely on globally shared graph filters for representation updating, our approach performs contrastive representation learning by leveraging relational information in the graph. Here, the neighboring nodes are considered positive pairs, while all other node pairs are treated as negative. However, the presence of heterophilic relations may introduce noisy contrastive pairs. This challenge motivates us to formulate a noise-robust contrastive objective to enhance the learning of meaningful representations.

Recent theoretical and empirical studies have demonstrated that negative label smoothing is effective in high levels of label noise [38]. Unlike standard label smoothing, which softens one-hot labels by distributing part of the label weight uniformly across all classes, negative label smoothing uses a negative coefficient to combine the hard and soft labels, which can be expressed as

$$\mathbf{y}^{\text{NLS}} = (1 - r) \cdot \mathbf{y} + \frac{r}{c} \cdot \mathbf{1}, \tag{3}$$

where $\mathbf{y}$ denotes a one-hot label vector, $\mathbf{1}$ is an all-one vector, c represents the number of classes, and $r < 0$ is the smoothing coefficient. As such, negative label smoothing penalizes incorrect predictions more aggressively, compared with standard label smoothing, and encourages higher confidence on clean signals.

Inspired by negative label smoothing, we propose a negative relational smoothing strategy to handle graph heterophily. To be specific, for each node v_i, we first define a hard relational label vector $\mathbf{y}_i^{(\mathrm{re})} \in \{0,1\}^n$, where the jth element $\mathbf{y}_i^{(\mathrm{re})}[j] = 1$ if there exists an edge between v_i and v_j, and 0 otherwise. As such, v_i and v_j forms a positive pair when $\mathbf{y}_i^{(\mathrm{re})}[j] = 1$. On this basis, we can derive the smoothed relational label vector $\hat{\mathbf{y}}_i^{(\mathrm{re})}$ as follows:

$$\hat{\mathbf{y}}_i^{(\mathrm{re})}[j] = \begin{cases} \dfrac{1-r}{|\mathcal{N}_i^{(\mathrm{ne})}|}, & \text{if } v_j \in \mathcal{N}_i^{(\mathrm{ne})} \\ \dfrac{r}{n - |\mathcal{N}_i^{(\mathrm{ne})}| - 1}, & \text{if } v_j \in \tilde{\mathcal{N}}_i^{(\mathrm{ne})} \\ 0, & \text{otherwise,} \end{cases} \tag{4}$$

where $r < 0$ is the smoothing coefficient, $\mathcal{N}_i^{(\mathrm{ne})}$ is the set of neighbors of v_i, and $\tilde{\mathcal{N}}_i^{(\mathrm{ne})} = \mathcal{V} \setminus (\mathcal{N}_i^{(\mathrm{ne})} \cup \{v_i\})$. Afterwards, with the smoothed relational label vector $\hat{\mathbf{y}}_i^{(\mathrm{re})}$, we can derive the NeRS loss as follows:

$$\mathcal{L}^{\mathrm{NeRS}} = -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{n}\hat{\mathbf{y}}_i^{(\mathrm{re})}[j] \cdot \log\left(\mathbf{p}_i[j]\right). \tag{5}$$

In Eq. (5), $\mathbf{p}_i \in \mathbb{R}^n$ encodes the pairwise similarity of the node representations and can be expressed as

$$\mathbf{p}_i[j] = \frac{e^{s_{ij}/\tau}}{\sum_{k=1}^{n} e^{s_{ik}/\tau}}. \tag{6}$$

Here, the node representations are obtained via a Multi-Layer Perceptron (MLP) encoder and updated by optimizing the objective $\mathcal{L}^{\mathrm{NeRS}}$. This objective can be viewed as an implicit form of contrastive learning, where the similarity between each node pair (v_i, v_j) is modulated by $\hat{\mathbf{y}}_i^{(\mathrm{re})}[j]$. Specifically, $\hat{\mathbf{y}}_i^{(\mathrm{re})}[j] > 0$ encourages higher similarity between v_j and v_i (*i.e.*, a positive pair), while $\hat{\mathbf{y}}_i^{(\mathrm{re})}[j] < 0$ discourages similarity between them (*i.e.*, a negative pair). By leveraging the negative relational smoothing strategy, our proposed method can achieve robustness against heterophilic edges. Additionally, by focusing on relation-specific signals instead of graph-level operations, our proposed method is able to learn expressive representations across both homophilic and heterophilic substructures.

4.3. Enhancing NeRS with multi-view establishment

Although the NeRS loss $\mathcal{L}^{\mathrm{NeRS}}$ demonstrates robustness against graph heterophily, the reliance on a single neighborhood scope could limit its expressiveness. To be concrete, under varying levels of homophily, neighbors may exhibit different degrees of class consistency with the central node. As a consequence, constructing contrastive pairs based on a single neighborhood scope can be suboptimal. Motivated by this, we design a multi-view architecture to enhance representation learning in both homophilic and heterophilic substructures. Specifically, the one-hop and two-hop neighbors are employed to construct two structural views, denoted as ϕ_1 and ϕ_2, respectively. As demonstrated in the analysis of Section 4.5.1, one-hop neighbors tend to maximize class consistency under homophily, whereas two-hop neighbors often yield stronger consistency under heterophily, which is also in line with the balance theory (*i.e.*, "the enemy of my enemy is my friend") [39].

On this basis, we define the NeRS objectives in two views, *i.e.*, $\mathcal{L}^{\mathrm{NeRS}(\phi_1)}$ and $\mathcal{L}^{\mathrm{NeRS}(\phi_2)}$, following Eq. (5), and then the integrated NeRS loss can be presented as

$$\mathcal{L}^{\mathrm{NeRS}(\phi)} = \mathcal{L}^{\mathrm{NeRS}(\phi_1)} + \mathcal{L}^{\mathrm{NeRS}(\phi_2)}. \tag{7}$$

Note that the node representations of views ϕ_1 and ϕ_2, namely $\mathbf{Z}^{(\phi_1)}$ and $\mathbf{Z}^{(\phi_2)}$, are generated by two independent MLP encoders $f^{(\phi_1)}$ and $f^{(\phi_2)}$, respectively. Afterwards, following the InfoNCE formulation in Eq. (2), we can derive the cross-view contrastive objective for enhancing NeRS as follows:

$$\mathcal{L}^{\mathrm{CNCE}} = -\sum_{v_i \in \mathcal{V}} \log \frac{e^{\mathrm{sim}(\mathbf{z}_i^{(\phi_1)}, \mathbf{z}_i^{(\phi_2)})}}{\sum_{v_j \in \mathcal{V}} e^{\mathrm{sim}(\mathbf{z}_i^{(\phi_1)}, \mathbf{z}_j^{(\phi_2)})} + \sum_{v_k \in \mathcal{V} \setminus \{v_i\}} e^{\mathrm{sim}(\mathbf{z}_i^{(\phi_1)}, \mathbf{z}_k^{(\phi_1)})}}, \tag{8}$$

Algorithm 1 The Proposed NeRS Algorithm

Input: Feature matrix $\mathbf{X}$; adjacency matrix $\mathbf{A}$; maximum number of iterations T; smoothing coefficient r.
Output: The overall node representations $\mathbf{O}$.

```
Compute the smoothed relational label vector ŷ^(re) for each node according to Eq. (4);
// Training phase
for t = 1 to T do
    Obtain node representations Z^(φ1) and Z^(φ2) based on the MLP encoders f^(φ1) and f^(φ2), respectively;
    Compute the NeRS loss L^NeRS(φ) based on Eq. (7);
    Compute the cross-view contrastive objective L^CNCE according to Eq. (8);
    Compute the total loss L^Overall via Eq. (10);
    Update the network parameters by minimizing L^Overall;
end for
// Inference phase
Obtain the overall output based on Eq. (9).
```

where the temperature parameter τ is omitted for brevity.

The dual-view framework constructed based on first-order and second-order neighbors can provide informative relational signals in graphs with both homophilic and heterophilic substructures. On this basis, negative relational smoothing is utilized to improve the reliability of the observed relations by suppressing noisy positive pairs, so that the model can learn from refined relational supervision. In addition, the cross-view contrastive objective is employed to encourage semantic consistency across the two views, which further enhances the expressive power of the learned representations.

4.4. Model training

To obtain the overall output, we integrate the embedding results from different views to ensure robustness across both homophilic and heterophilic substructures, which is expressed as follows:

$$\mathbf{O} = \alpha^{(\phi_1)}\mathbf{Z}^{(\phi_1)} + (1 - \alpha^{(\phi_1)})\mathbf{Z}^{(\phi_2)}. \tag{9}$$

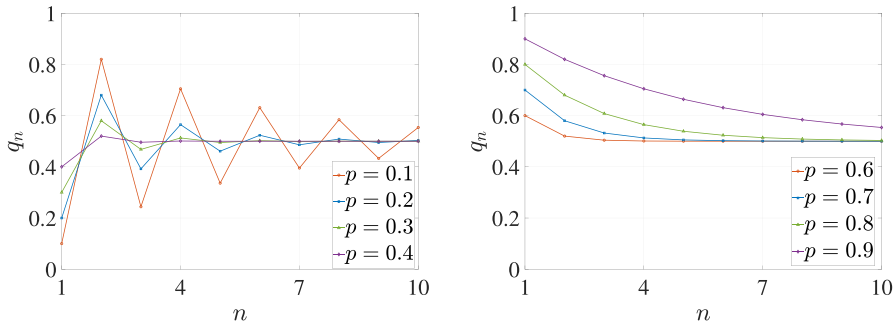
Here, $\alpha^{(\phi_1)}$ is a learnable balancing coefficient that controls the contributions of $\mathbf{Z}^{(\phi_1)}$. Subsequently, the overall objective of our proposed method arrives at

$$\mathcal{L}^{\mathrm{Overall}} = \mathcal{L}^{\mathrm{NeRS}(\phi)} + \lambda\mathcal{L}^{\mathrm{CNCE}}, \tag{10}$$

where the parameter λ is utilized to weight the importance of $\mathcal{L}^{\mathrm{CNCE}}$. In practice, the proposed negative relational smoothing introduces a repulsive effect for unreliable positive pairs, which helps reduce the risk of being misled by noisy relations, especially in heterophilic regions. In addition, it also helps mitigate representation collapse by discouraging excessive concentration of node embeddings and promoting better dispersion in the latent space. Meanwhile, the cross-view contrastive objective $\mathcal{L}^{\mathrm{CNCE}}$ encourages consistency between $\mathbf{Z}^{(\phi_1)}$ and $\mathbf{Z}^{(\phi_2)}$, thereby helping align the two views and reducing the risk of mutual deviation during training. As a result, the overall objective can provide reliable training signals for learning expressive node representations. The detailed description of our method is provided in Algorithm 1.

4.5. Theoretical analysis

We provide a theoretical analysis of NeRS by demonstrating the effectiveness of the multi-view establishment and proving that the NeRS loss guarantees robustness through an upper bound on the clean cross-entropy objective.



(a) (b)

Fig. 2. The probability that an n-hop neighbor shares the same class with the central node (*i.e.*, q_n) under different values of the edge homophily ratio ρ. (a) $\rho \in \{0.1, 0.2, 0.3, 0.4\}$; (b) $\rho \in \{0.6, 0.7, 0.8, 0.9\}$.

4.5.1. Rationale of multi-view establishment

Let $\rho \in [0, 1]$ denote the edge homophily ratio. Consider an undirected simple graph without self-loops. Assume that class labels are uniformly distributed and neighbors are conditionally independent given the central node, the probability that an n-hop neighbor shares the same label with the central node, which is denoted as q_n, can be approximated as

$$q_n = \rho \cdot q_{n-1} + (1-\rho) \cdot \left(\frac{1 - q_{n-1}}{C-1} \right) \tag{11}$$

for $n \geq 2$, where C is the number of classes and $q_1 = \rho$. The recurrence in Eq. (11) admits the closed-form solution as follows:

$$q_n = \frac{1}{C} + \left(\frac{\rho \cdot C - 1}{C-1} \right)^{n-1} \cdot \left(\rho - \frac{1}{C} \right) \tag{12}$$

for $n \geq 1$. In Eq. (12), it is clear that $q_n \to 1/C$ as n increases. When $\rho < 1/C$, which indicates the heterophilic tendency, q_n oscillates around $1/C$ with decaying magnitude. In this case, for all integers $k \geq 1$, we have $q_{2k} \geq 1/C \geq q_{2k+1}$, and the even-hop sequence $\{q_{2k}\}_{k \geq 1}$ is strictly decreasing. This implies that two-hop neighbors exhibit the highest probability of preserving class consistency among all hops. When $\rho > 1/C$, which indicates the homophilic tendency, q_n decreases monotonically with n. Hence, one-hop neighbors have the highest probability of preserving class consistency among all hops.

To better illustrate the effect of incorporating one-hop or two-hop neighbors, we present in Fig. 2 the theoretical variation of q_n derived from Eq. (12). For clarity, we consider the binary case (*i.e.*, $C = 2$), so $1/C = 0.5$. When $\rho < 1/C$, Fig. 2(a) displays an oscillatory trend, where the maximum values occur at $n = 2$. When $\rho > 1/C$, Fig. 2(b) shows a monotone decay of q_n from $n = 1$ toward 0.5 (*i.e.*, $1/C$), where the maximum values are attained at $n = 1$. All curves are obtained analytically from the closed-form expression.

4.5.2. Properties of NeRS loss

We first consider an ideal homophilic graph where the edges connect nodes with the same class labels. Under this assumption, the distribution of the label-consistent neighbors of node v_i (*i.e.*, the neighbors sharing the same class with the central node v_i) is denoted by $\mathbf{q}_i^*$. In practice, the observed graph may be corrupted, where each positive edge is dropped with probability η_+ and each negative edge is added with probability η_-. Here, an edge is positive if it connects two nodes with the same label and negative otherwise. The noisy graph is obtained by independently dropping each positive edge with probability η_+ and adding each negative edge with probability η_-. For simplicity, we define

$$\mathbf{q}_i^{\text{neg}}[j] = \frac{1}{m_i} \mathbb{1}\{ v_j \in \tilde{\mathcal{N}}_i^{(\text{ne})} \}, \tag{13}$$

where $m_i = n - b_i - 1$ and $b_i = \left| \mathcal{N}_i^{(\text{ne})} \right|$. Afterwards, the observed neighbor distribution can be expressed as follows:

$$\mathbf{q}_i^{\text{obs}} = (1 - \eta_+)\, \mathbf{q}_i^* + \eta_- \, \mathbf{q}_i^{\text{neg}}. \tag{14}$$

On this basis, the smoothed relational target in Eq. (4) becomes

$$\hat{\mathbf{q}}_i^{(\text{re})} = (1 - r)\, \mathbf{q}_i^{\text{obs}} + r\, \mathbf{q}_i^{\text{neg}}. \tag{15}$$

Taking expectation over $\hat{\mathbf{q}}_i^{(\text{re})}$ yields

$$\mathbb{E}\left[\hat{\mathbf{q}}_i^{(\text{re})} \right] = \big((1-r)(1-\eta_+) \big) \mathbf{q}_i^* + \big((1-r)\eta_- + r \big) \mathbf{q}_i^{\text{neg}}. \tag{16}$$

When the coefficient of the second term is close to zero, $\mathbb{E}[\hat{\mathbf{q}}_i^{(\text{re})}]$ becomes proportional to $\mathbf{q}_i^*$. Since cross-entropy is invariant to positive scaling of the target distribution, minimizing $\mathcal{L}^{\text{NeRS}}$ is equivalent up to a positive constant factor to minimizing the cross-entropy with the clean target $\mathbf{q}_i^*$. This ensures that NeRS remains aligned with the label-consistent relational signals, thereby improving robustness under heterophily. In addition, this suggests an empirical calibration of r:

$$r \approx -\frac{\eta_-}{1 - \eta_-}. \tag{17}$$

Furthermore, to correct for the imbalance induced by node degrees, a degree-normalized variant can be adopted:

$$r_i \approx -\frac{\eta_-}{1 - \eta_+} \cdot \frac{m_i}{b_i}. \tag{18}$$

Although the above analysis provides a theoretical calibration of r, the noise rates η_+ and η_- are unknown in practice and cannot be reliably estimated.

Based on the above-mentioned analysis, we will discuss the connection between $\mathcal{L}^{\text{NeRS}}$ and the cross-entropy objective with the clean target distribution $\mathbf{q}_i^*$. Substituting the decomposition of $\hat{\mathbf{q}}_i^{(\text{re})}$, $\mathcal{L}^{\text{NeRS}}$ can be rewritten as a linear combination of cross-entropy terms, namely

$$\mathcal{L}^{\text{NeRS}} = \frac{1}{n} \sum_{i=1}^{n} \Big\{ \beta\, \mathcal{L}^{\text{CE}}(\mathbf{q}_i^*, \mathbf{p}_i) + \gamma\, \mathcal{L}^{\text{CE}}(\mathbf{q}_i^{\text{neg}}, \mathbf{p}_i) \Big\}, \tag{19}$$

where $\beta = (1-r)(1-\eta_+)$ and $\gamma = (1-r)\eta_- + r$, $\mathbf{p}_i$ denotes the predicted distribution in Eq. (6), and $\mathcal{L}^{\text{CE}}$ indicates the cross-entropy loss. When $\gamma \geq 0$, the second term is lower bounded by $\gamma \log m_i$, so that

$$\mathbb{E}\big[\mathcal{L}^{\text{CE}}(\mathbf{q}_i^*, \mathbf{p}_i) \big] \;\leq\; \frac{\mathcal{L}^{\text{NeRS}} - \mathbb{E}[\gamma \log m_i]}{\beta}. \tag{20}$$

This establishes that $\mathcal{L}^{\text{NeRS}}$ serves as an upper bound of the cross-entropy objective with the clean target distribution $\mathbf{q}_i^*$, *i.e.*, $\mathcal{L}^{\text{CE}}(\mathbf{q}_i^*, \mathbf{p}_i)$. That is, minimizing $\mathcal{L}^{\text{NeRS}}$ is guaranteed to minimize an upper bound of $\mathcal{L}^{\text{CE}}(\mathbf{q}_i^*, \mathbf{p}_i)$, which makes NeRS robust under heterophilic graphs.

It is also notable that although the proposed negative relational smoothing may exhibit an effect similar to adaptive filtering mechanism, it should not be regarded as a spectral filter in the strict sense. Specifically, NeRS is not defined in the Laplacian eigenbasis and does not impose an explicit frequency response on graph signals. Instead, it operates by refining pairwise relational supervision in the contrastive objective. More detailed analysis is provided in the supplementary material.

4.6. Theoretical interpretation of negative relational smoothing

To further clarify the qualitative difference between the proposed negative relational smoothing strategy and the negative sampling mechanism used in conventional contrastive learning, we reinterpret NeRS from the perspective of target distributions. For an anchor node v_i, let $\mathbf{p}_i \in \mathbb{R}^n$ denote the predicted similarity distribution over all candidate nodes, as defined in Eq. (6). In standard contrastive learning, the objective is typically constructed at the sample level. To be concrete, a set of positive and negative pairs is first specified for each anchor, and then the model is optimized by maximizing the similarity of positive pairs while minimizing that of negative pairs. This process can be equivalently viewed as matching $\mathbf{p}_i$ to a sparse target distribution $\mathbf{t}_i$, *i.e.*,

$$\mathcal{L}^{\text{pair}} = -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{n}\mathbf{t}_i[j]\log\mathbf{p}_i[j], \tag{21}$$

where $\mathbf{t}_i[j] = 1$ if node v_j forms a positive pair with v_i, and $\mathbf{t}_i[j] = 0$ otherwise. In this formulation, the optimization is defined on individual pairs, where each pair is assigned a discrete role, such as positive or negative.

Unlike traditional contrastive learning, our proposed NeRS is defined at the relation level. Instead of explicitly sampling a finite set of negative pairs, NeRS constructs a signed relational target $\hat{\mathbf{y}}_i^{(\text{re})}$ over all nodes according to Eq. (4), where neighboring nodes receive positive weights and unconnected nodes receive negative weights. Substituting Eq. (4) into Eq. (5), the NeRS loss can be decomposed as

$$\mathcal{L}^{\text{NeRS}} = (1-r)\mathcal{L}^{\text{attr}} - r\mathcal{L}^{\text{rep}}, \tag{22}$$

where

$$\mathcal{L}^{\text{attr}} = -\frac{1}{n}\sum_{i=1}^{n}\frac{1}{|\mathcal{N}_i^{(\text{ne})}|}\sum_{v_j\in\mathcal{N}_i^{(\text{ne})}}\log\mathbf{p}_i[j], \tag{23}$$

and

$$\mathcal{L}^{\text{rep}} = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{|\tilde{\mathcal{N}}_i^{(\text{ne})}|}\sum_{v_j\in\tilde{\mathcal{N}}_i^{(\text{ne})}}\log\mathbf{p}_i[j]. \tag{24}$$

Since $r < 0$, minimizing $\mathcal{L}^{\text{NeRS}}$ increases the probability mass assigned to observed neighbors through $\mathcal{L}^{\text{attr}}$ and suppresses the probability mass assigned to unconnected node pairs through $\mathcal{L}^{\text{rep}}$, simultaneously. Therefore, the repulsive effect in our NeRS is achieved by redistributing probability mass over the entire relational structure induced by the graph.

To sum up, in conventional contrastive learning frameworks, negative information is incorporated through sampled negative pairs, and the optimization objective is defined over individual pairwise comparison. Differently, by introducing a relational target, our NeRS performs a form of distributional correction based on relational supervision, which suppresses unreliable relations while preserving the class-consistent relational signals. In this sense, our NeRS can be interpreted as a robust distributional extension of pairwise contrastive learning.

4.7. Time complexity analysis

We analyze the time complexity of our proposed NeRS by considering three key components. Firstly, the MLP-based encoding requires $\mathcal{O}(ndh)$ operations, where n is the number of nodes, d represents the dimension of the input features, and h is the dimension of the embeddings. Secondly, the NeRS loss in Eq. (5) requires computing pairwise similarity between all node embeddings, which results in $\mathcal{O}(n^2h)$ computational complexity. Thirdly, the cross-view contrastive objective (Eq. (8)) requires computing all pairwise similarity between the two structural views, which incurs $\mathcal{O}(n^2h)$ complexity. Finally, combining the above components, the total time complexity per training iteration is $\mathcal{O}(ndh) + \mathcal{O}(n^2h)$. Since $n \gg d$ in typical graph learning scenarios, the overall time complexity of our proposed NeRS is $\mathcal{O}(n^2h)$.

Table 1
Statistics of the homophilic datasets used in this paper, where "Hom." denotes the edge homophily ratio.

Dataset	Cora	CiteSeer	PubMed	Amazon computers	Wiki-CS
#Nodes	2708	3327	19,717	13,752	11,701
#Edges	5429	4732	44,338	245,861	216,123
#Features	1433	3703	500	767	300
#Classes	7	6	3	10	10
Hom.	0.81	0.74	0.80	0.38	0.40

Table 2
Statistics of the heterophilic datasets used in this paper, where "Hom." denotes the edge homophily ratio.

Dataset	Cornell	Texas	Wisconsin	Actor	Roman-Empire
#Nodes	183	183	251	7600	22,662
#Edges	295	309	466	33,544	32,927
#Features	1703	1703	1703	931	300
#Classes	5	5	5	5	18
Hom.	0.31	0.11	0.20	0.22	0.05

5. Experimental results

To reveal the effectiveness of our proposed NeRS, in this section, extensive experiments have been conducted on the task of self-supervised node classification across various real-world datasets.

5.1. Experimental settings

Here, we will introduce the experimental settings used to evaluate our proposed method, including the datasets, baseline methods, and evaluation protocol.

5.1.1. Datasets

We evaluate our method on ten widely used benchmark datasets, including both homophilic and heterophilic graphs. Importantly, while these datasets are often categorized as homophilic or heterophilic according to their overall homophily ratio, each of them contains both homophilic and heterophilic substructures. To be concrete, we choose three widely used citation network datasets, namely Cora, Citeseer, and Pubmed [40,41], where nodes represent documents and edges denote citation relationships. Meanwhile, we also adopt Wiki-CS [42] and Amazon Computers [43] datasets. Here, Wiki-CS is a Wikipedia-based network with nodes representing articles and edges denoting hyperlinks between them, and Amazon Computers is a subset of the Amazon co-purchase network, with nodes corresponding to products and edges indicating frequent co-purchase behaviors. In addition, we have also incorporated five heterophilic datasets, including Cornell, Texas, Wisconsin, Actor [44], and Roman-Empire [45]. Here, Cornell, Texas, and Wisconsin are web-page networks from the WebKB dataset, with nodes representing web pages and edges denoting hyperlinks. Actor is a co-occurrence network where nodes represent actors and edges indicate co-appearances in the same movie. Roman-Empire is a word-level graph built from a Wikipedia article, where nodes represent words and edges connect syntactically or sequentially related words. The statistics of the ten datasets adopted in this paper are summarized in Tables 1 and 2.

5.1.2. Baseline methods

To demonstrate the effectiveness of our method, we compare NeRS with a range of recently proposed baseline methods, including one supervised GNN model GCN [19] and fourteen self-supervised models, namely SGRL [46], EPAGCL [47], GraphCL [30], GRACE [31], DGI [37], GCA [48], GraphECL [49], PolyGCL [12], HeterGCL [9], GREET [28], ARIEL [50], IFL-GCL [51], AFECL [52], and E2Neg [53].

Table 3
Classification accuracies (%) of different methods on five homophilic datasets. The best result on each dataset is highlighted in bold and the second best is underlined. OOM denotes "out of memory".

Method	Cora	CiteSeer	PubMed	Amazon computers	Wiki-CS
GCN [19]	82.73 ± 0.69 **	70.18 ± 1.36 **	86.24 ± 0.30 **	82.75 ± 3.26 **	78.66 ± 0.65 **
SGRL [46]	83.52 ± 0.64 *	70.81 ± 0.58 **	86.15 ± 0.25 **	**89.32 ± 0.28**	79.23 ± 0.32 *
EPAGCL [47]	83.24 ± 0.54 *	69.53 ± 1.25 **	84.11 ± 0.50 **	86.17 ± 0.54 **	77.67 ± 0.40 **
GraphCL [30]	82.87 ± 0.62 **	71.43 ± 0.85 **	80.05 ± 0.62 **	86.20 ± 0.41 **	52.68 ± 4.71 **
GRACE [31]	83.30 ± 0.55 **	70.81 ± 0.53 **	84.24 ± 0.22 **	85.19 ± 0.72 **	79.39 ± 0.29
DGI [37]	82.41 ± 0.51 **	73.05 ± 0.32 *	78.61 ± 0.81 **	50.17 ± 0.94 **	77.76 ± 0.41 **
GCA [48]	82.84 ± 0.71 **	68.60 ± 1.20 **	83.52 ± 0.29 **	87.17 ± 0.19 **	74.54 ± 3.36 **
GraphECL [49]	83.12 ± 0.75 **	71.50 ± 0.93 **	84.62 ± 0.53 **	58.18 ± 0.64 **	58.20 ± 0.56 **
PolyGCL [12]	82.81 ± 0.32 **	72.31 ± 0.43 **	80.95 ± 0.23 **	79.17 ± 1.32 **	69.85 ± 0.53 **
HeterGCL [9]	81.70 ± 0.87 **	72.35 ± 0.51 **	84.01 ± 0.27 **	85.10 ± 0.83 **	76.82 ± 0.37 **
GREET [28]	82.98 ± 0.62 **	72.96 ± 0.74 **	OOM	OOM	77.76 ± 0.41 **
ARIEL [50]	83.60 ± 0.27 *	72.30 ± 0.24 **	85.77 ± 0.12 **	87.16 ± 0.17 **	78.53 ± 0.23 **
IFL-GCL [51]	83.14 ± 0.98 **	68.68 ± 1.23 **	85.66 ± 0.47 **	85.69 ± 0.53 **	78.82 ± 0.62 **
AFECL [52]	82.01 ± 0.65 **	72.03 ± 0.48 **	84.38 ± 0.38 **	OOM	OOM
E2Neg [53]	78.57 ± 0.79 **	70.42 ± 0.93 **	85.07 ± 0.16 **	86.88 ± 0.86 **	79.14 ± 0.28 *
NeRS	**84.03 ± 0.51**	**73.36 ± 0.38**	**86.80 ± 0.28**	87.96 ± 0.42	**79.54 ± 0.30**

Table 4
Classification accuracies (%) of different methods on five heterophilic datasets. The best result on each dataset is highlighted in bold and the second best is underlined. OOM denotes "out of memory".

Method	Cornell	Texas	Wisconsin	Actor	Roman-Empire
GCN [19]	43.91 ± 6.05 **	54.59 ± 2.97 **	49.90 ± 7.19 **	27.82 ± 0.67 **	44.48 ± 1.20 **
SGRL [46]	43.58 ± 1.57 **	54.34 ± 1.07 **	47.83 ± 3.61 **	27.36 ± 0.29 **	39.39 ± 0.22 **
EPAGCL [47]	38.10 ± 7.23 **	49.66 ± 8.37 **	42.59 ± 7.09 **	28.13 ± 0.89 **	32.51 ± 2.84 **
GraphCL [30]	43.77 ± 0.49 **	52.50 ± 4.46 **	49.66 ± 1.41 **	27.47 ± 0.60 **	25.73 ± 0.36 **
GRACE [31]	40.73 ± 3.11 **	53.49 ± 2.23 **	44.96 ± 2.37 **	27.19 ± 0.82 **	37.62 ± 0.23 **
DGI [37]	40.29 ± 5.34 **	57.54 ± 2.04 **	50.61 ± 3.69 **	29.36 ± 0.43 **	27.00 ± 0.18 **
GCA [48]	40.18 ± 2.66 **	51.43 ± 3.11 **	47.06 ± 2.39 **	28.94 ± 0.35 **	40.05 ± 0.34 **
GraphECL [49]	53.11 ± 3.58 **	59.93 ± 3.42 **	64.43 ± 3.18 **	33.12 ± 1.43 **	58.34 ± 1.34 **
PolyGCL [12]	55.19 ± 1.99 **	62.30 ± 1.54 **	62.00 ± 2.28 **	33.23 ± 0.22 **	59.25 ± 0.61 **
HeterGCL [9]	65.56 ± 2.50 *	75.00 ± 1.62 **	75.17 ± 2.44	36.38 ± 0.62 **	59.37 ± 0.67 **
GREET [28]	65.63 ± 3.10	70.47 ± 4.92 **	74.73 ± 3.28 **	35.06 ± 0.74 **	OOM
ARIEL [50]	41.56 ± 2.16 **	54.67 ± 1.14 **	54.56 ± 0.99 **	28.44 ± 0.91 **	48.29 ± 0.38 **
IFL-GCL [51]	44.20 ± 4.25 **	49.81 ± 2.54 **	49.13 ± 3.20 **	29.25 ± 0.91 **	OOM
AFECL [52]	43.25 ± 0.70 **	58.05 ± 2.31 **	46.21 ± 3.11 **	27.17 ± 0.52 **	37.97 ± 0.24 **
E2Neg [53]	43.47 ± 1.89 **	56.85 ± 1.12 **	46.90 ± 2.41 **	27.97 ± 0.92 **	40.57 ± 0.44 **
NeRS	**65.99 ± 2.10**	**75.55 ± 2.09**	**75.87 ± 1.81**	**37.09 ± 0.36**	**60.49 ± 0.40**

5.1.3. Evaluation settings

Here, we adopt the standard linear evaluation protocol as introduced by [37], which follows a two-stage paradigm. In the first stage, the model is trained in a fully self-supervised manner using only the node features and graph structure, without access to any label information. Once training is complete, the encoder is frozen, and the learned node representations are used in the second stage to train an MLP classifier. All methods are evaluated under the same data splits, with 10% of labeled nodes per class used for training, 10% for validation, and the remaining 80% for testing. For fair comparison, all baseline methods have been carefully re-tuned on each dataset. Specifically, we follow the hyperparameter ranges suggested in their original papers and select the best configurations based on validation performance under the same data splits. All the experiments are repeated ten times, and we report the average performance for fair evaluation. To statistically validate the superiority of our proposed NeRS over the baseline methods, the paired t-test is adopted in our experiments. Here, statistical significance is evaluated based on the p-values, where * and ** denote p-value < 0.1 and p-value < 0.05, respectively, when comparing each baseline method against NeRS.

The experiments are conducted on a Linux server equipped with one RTX 4090 GPU featuring 24 GB of VRAM. Our proposed algorithm is implemented via PyTorch 1.13.1, along with PyTorch Geometric 2.4.0. The network parameters are updated with the Adam optimizer. In our experiments, two different views are adopted to produce node representations, where the numbers of hidden units are kept identical across views.

5.2. Results and analysis

This subsection reports the experimental results and analyzes the performance of our proposed method in comparison with various baseline methods.

5.2.1. Performance on homophilic datasets

The comparison results on homophilic datasets are exhibited in Table 3. It can be noted that the GCL models devised for handling graph heterophily (*i.e.*, PolyGCL, HeterGCL, and GREET) generally achieve inferior performance, compared with SGRL, EPAGCL, and GRACE, which rely on the homophilic assumption. The performance gap arises because the models designed for heterophilic graphs primarily aim to capture complex non-homophilic relations. Hence, they often neglect the simple homophilic structures. Differently, our proposed NeRS outperforms the baseline methods on four out of five homophilic datasets and achieves competitive performance on the Amazon Computers dataset. In particular, it outperforms the models designed for heterophilic graphs by at least 1.05% on Cora and 1.78% on Wiki-CS, respectively. This performance gain can be largely attributed to the negative relational smoothing strategy, which effectively suppresses noisy contrastive signals while preserving relevant structural information. As a consequence, it is not surprising that our NeRS can produce expressive representations.

5.3. Performance under mixed homophily

To evaluate the effectiveness of our method in scenarios where homophilic and heterophilic substructures coexist, we conduct an additional analysis based on node-level edge homophily ratio. Specifically,

Table 5
Ablation study on the five homophilic datasets. The best result on each dataset is highlighted in bold.

Method	Cora	CiteSeer	PubMed	Amazon computers	Wiki-CS
NeRS (w/o $\mathcal{L}^{\mathrm{NeRS}(\phi_1)}$)	78.97 ± 0.40	70.23 ± 0.45	85.76 ± 0.36	82.59 ± 0.31	74.42 ± 0.34
NeRS (w/o $\mathcal{L}^{\mathrm{NeRS}(\phi_2)}$)	81.58 ± 0.38	72.40 ± 0.46	85.96 ± 0.28	87.46 ± 0.27	79.25 ± 0.29
NeRS (w/o $\mathcal{L}^{\mathrm{CNCE}}$)	83.89 ± 0.68	73.17 ± 0.40	86.42 ± 0.27	87.06 ± 0.37	79.04 ± 0.35
NeRS	**84.03 ± 0.51**	**73.36 ± 0.38**	**86.80 ± 0.28**	**87.96 ± 0.42**	**79.54 ± 0.30**

Table 6
Ablation study on the five heterophilic datasets. The best result on each dataset is highlighted in bold.

Method	Cornell	Texas	Wisconsin	Actor	Roman-Empire
NeRS (w/o $\mathcal{L}^{\mathrm{NeRS}(\phi_1)}$)	58.23 ± 3.46	74.79 ± 2.11	73.13 ± 3.43	36.95 ± 0.39	60.04 ± 0.48
NeRS (w/o $\mathcal{L}^{\mathrm{NeRS}(\phi_2)}$)	57.01 ± 2.86	68.01 ± 8.80	72.59 ± 3.64	36.71 ± 0.32	60.11 ± 0.29
NeRS (w/o $\mathcal{L}^{\mathrm{CNCE}}$)	57.89 ± 1.94	72.19 ± 1.94	74.63 ± 1.61	34.71 ± 0.43	59.11 ± 0.58
NeRS	**65.99 ± 2.10**	**75.55 ± 2.09**	**75.87 ± 1.81**	**37.09 ± 0.36**	**60.49 ± 0.40**

Table 7
Classification accuracy (%) on nodes with medium edge homophily ratio (*i.e.*, $[1/3, 2/3]$). The best result on each dataset is highlighted in bold.

Method	Cora	CiteSeer	Wisconsin	Texas
EPAGCL	57.54 ± 1.89	57.86 ± 2.56	40.78 ± 12.41	72.65 ± 3.56
HeterGCL	57.22 ± 1.95	63.06 ± 1.99	70.35 ± 4.19	73.95 ± 3.20
PolyGCL	59.16 ± 1.17	59.87 ± 0.85	62.44 ± 4.08	70.88 ± 3.71
AFECL	62.45 ± 1.65	63.70 ± 2.26	36.28 ± 5.17	73.16 ± 1.66
NeRS	**63.05 ± 1.21**	**66.74 ± 1.47**	**73.49 ± 4.24**	**74.17 ± 1.34**

we compute the edge homophily ratio of each node based on its neighborhood, and then select nodes with homophily values falling within the interval $[1/3, 2/3]$, which often corresponds to regions where homophilic and heterophilic relations are highly mixed. Here, we adopt four representative GCL baseline methods that are proposed in recent years (*i.e.*, 2024 or 2025). The results are exhibited in Table 7. Although there exist both homophilic and heterophilic relations, which may lead to ambiguous relational supervision for representation learning, we observe that the proposed NeRS still achieves the best performance across the datasets. This demonstrates that the proposed negative relational smoothing strategy is effective in handling scenarios with homophilic and heterophilic substructures, which helps enhance the expressive power of the learned representations.

5.3.1. Performance on heterophilic datasets

As shown in Table 4, models built upon the homophily assumption tend to exhibit performance degradation when evaluated on the five heterophilic datasets. This is due to that these models aggregate features indiscriminately from the neighborhood, which inevitably suppresses meaningful high-frequency signals. Unlike the statistics in Table 3, the models designed for handling heterophilic graphs generally obtain satisfactory performance, which demonstrates their effectiveness in heterophilic settings. Among these methods, our proposed NeRS achieves the best performance across all five datasets, which can be attributed to two key factors. Firstly, unlike conventional GCL methods relying on globally shared graph filters, the negative relational smoothing strategy is inherently robust to varying levels of heterophily. Secondly, by avoiding potentially destructive augmentation operations, the proposed NeRS can obtain reliable contrastive signals without distorting the original graph information.

To sum up, by combining negative relational smoothing with cross-view contrastive learning, our NeRS not only demonstrates strong robustness to varying levels of graph heterophily, but also reveals great potential in modeling homophilic relations. The results reported in Tables 3 and 4 collectively highlight the effectiveness and generalizability of our proposed NeRS in both homophilic and heterophilic graphs.

Table 8
Comparison of per-epoch training time (in milliseconds) of different self-supervised methods.

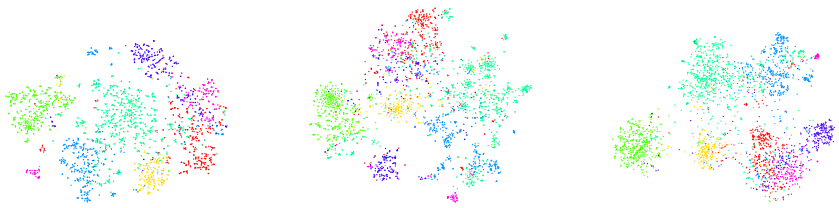
Method	Cora	CiteSeer	Wiki-CS	Cornell	Texas	Actor
SGRL [46]	45	60	94	24	36	36
EPAGCL [47]	33	50	180	156	11	342
GraphCL [30]	32	28	33	12	2	74
GRACE [31]	39	48	117	38	40	70
DGI [37]	66	1177	1491	5	4	2259
GCA [48]	11	16	121	10	10	31
GraphECL [49]	71	79	444	97	59	209
PolyGCL [12]	66	138	232	20	19	146
HeterGCL [9]	18	31	117	5	5	61
GREET [28]	99	208	1873	15	14	783
NeRS	35	62	244	18	17	108

5.3.2. Running time comparison

In addition to effectiveness, it is important to evaluate the computational overhead introduced by different self-supervised methods. Accordingly, Table 8 summarizes the per-epoch training time of the self-supervised methods on both homophilic and heterophilic datasets. Overall, the proposed NeRS demonstrates competitive computational efficiency across datasets of varying scales and heterophily levels. On small heterophilic datasets (*e.g.*, Cornell and Texas), NeRS maintains moderate per-epoch costs, which indicates that the proposed negative relational smoothing does not introduce excessive computational overhead. On larger datasets such as Wiki-CS and Actor, NeRS exhibits higher per-epoch training time than lightweight methods (*e.g.*, GCA), which can be attributed to the computation of pairwise similarities in the relational contrastive objective. Despite this additional cost, our NeRS generally achieves higher accuracies than the baseline methods. All these results indicate that NeRS strikes a favorable trade-off between effectiveness and efficiency, which offers robust representation learning without incurring prohibitive computational overhead.

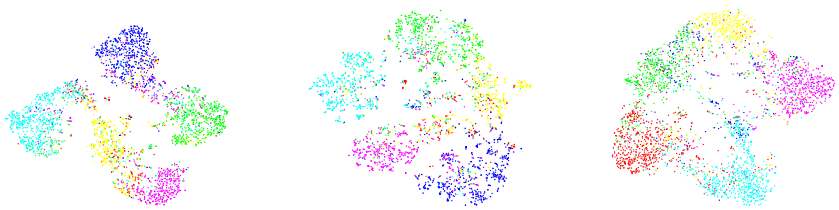
5.4. Ablation study

As is mentioned above, in our proposed method, the NeRS losses (*i.e.*, $\mathcal{L}^{\mathrm{NeRS}(\phi_1)}$ and $\mathcal{L}^{\mathrm{NeRS}(\phi_2)}$) and the cross-view contrastive loss (*i.e.*, $\mathcal{L}^{\mathrm{CNCE}}$) are critical in producing expressive representations in both homophilic and heterophilic substructures. To shed light on the contributions of these components, we report the classification results of our proposed method when each of them is removed on the previously-used ten datasets, where the data splits are kept identical with those described above. For simplicity, we adopt "NeRS (w/o $\mathcal{L}^{\mathrm{NeRS}(\phi_1)}$)", "NeRS (w/o $\mathcal{L}^{\mathrm{NeRS}(\phi_2)}$)", and "NeRS (w/o $\mathcal{L}^{\mathrm{CNCE}}$)" to represent the reduced models by removing the NeRS losses $\mathcal{L}^{\mathrm{NeRS}(\phi_1)}$, $\mathcal{L}^{\mathrm{NeRS}(\phi_2)}$, and the cross-view contrastive loss $\mathcal{L}^{\mathrm{CNCE}}$, respectively. The comparative results have been listed in Tables 5 and 6. It is evident that removing either loss



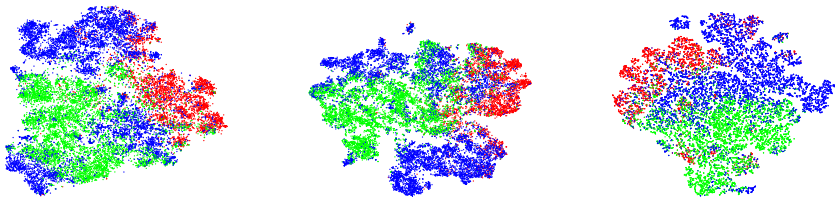
(a) (b) (c)

Fig. 3. t-SNE visualizations of node embeddings learned from different views on Cora dataset. (a) $\mathbf{Z}^{(\phi_1)}$; (b) $\mathbf{Z}^{(\phi_2)}$; (c) $\mathbf{O}$.



(a) (b) (c)

Fig. 4. t-SNE visualizations of node embeddings learned from different views on CiteSeer dataset. (a) $\mathbf{Z}^{(\phi_1)}$; (b) $\mathbf{Z}^{(\phi_2)}$; (c) $\mathbf{O}$.



(a) (b) (c)

Fig. 5. t-SNE visualizations of node embeddings learned from different views on PubMed dataset. (a) $\mathbf{Z}^{(\phi_1)}$; (b) $\mathbf{Z}^{(\phi_2)}$; (c) $\mathbf{O}$.

leads to a decline in classification accuracy, which highlights the essential roles of negative relational smoothing and cross-view contrastive learning in achieving strong performance. Notably, excluding $\mathcal{L}^{\mathrm{NeRS}^{(\phi_1)}}$ results in a more significant performance degradation on homophilic datasets, while omitting $\mathcal{L}^{\mathrm{NeRS}^{(\phi_2)}}$ has a larger negative impact on heterophilic graphs. These results suggest that one-hop neighbors are more informative in homophilic settings, whereas two-hop neighbors are more beneficial under heterophily, which proves the rationality of our multi-view architecture.

To qualitatively analyze the effect of the proposed multi-view architecture, we visualize the learned node representations on three representative datasets, namely Cora, CiteSeer, and PubMed, using t-SNE. For each dataset, we report the embeddings obtained from the one-hop view $\mathbf{Z}^{(\phi_1)}$, the two-hop view $\mathbf{Z}^{(\phi_2)}$, and the integrated representation $\mathbf{O}$, as shown in Figs. 3–5. From Figs. 3(a), 4(a), and 5(a), we observe that the one-hop view $\mathbf{Z}^{(\phi_1)}$ tends to produce relatively compact clusters, which is consistent with the fact that one-hop neighbors preserve strong class consistency under homophily. Nevertheless, some classes still exhibit partial overlap, which indicates limited discriminative power when relying on a single neighborhood scope. Notably, the integrated representation $\mathbf{O}$, shown in Figs. 3(c), 4(c), and 5(c), consistently yields the most compact clusters across all three datasets, which demonstrates the effectiveness of the proposed multi-view establishment.

To further investigate the interaction between the multi-view architecture and the smoothing coefficient r, we compare the performance among three model variants that incorporate the one-hop view (*i.e.*, "One-hop only"), the two-hop view (*i.e.*, "Two-hop only"), and the multi-view architecture (*i.e.*, "Multi-view"), respectively, without negative relational smoothing (*i.e.*, "$r = 0$") and with negative relational smoothing (*i.e.*, "$r = r^*$"). Here, r^* denotes the selected smoothing coefficient on each dataset. The results are reported in Table 9. When $r = 0$, we observe that the "One-hop only" variant performs better on the homophilic datasets (*i.e.*, Cora and CiteSeer), whereas the "Two-hop only" variant achieves better results on the heterophilic datasets (*i.e.*, Texas and Wisconsin). This observation is consistent with our analysis in Section 4.5.1, where one-hop neighbors are shown to preserve stronger class consistency under homophily, while two-hop neighbors become more informative under heterophily. By introducing negative relational smoothing, the performance of all variants is improved, which indicates that the negative relational smoothing effectively enhances the reliability of relational supervision within each view. It is also notable that the performance of the "Two-hop only" variant drops from 71.39% to 62.94% when switching from $r = r^*$ to $r = 0$ on Wisconsin dataset. This indicates that the benefit of the two-hop view is relatively limited without negative relational smoothing, which can be attributed to the existence of noisy relational signals. As negative relational smoothing helps suppress noisy relations, the effectiveness of the "Two-hop only" variant can be enhanced when $r = r^*$. Nevertheless, the two-hop view still demonstrates clear advantages in heterophilic graphs when $r = 0$ or $r = r^*$. In addition, we observe that the multi-view model generally outperforms the single-view variants under both $r = 0$ and $r = r^*$, which demonstrates the effectiveness of the proposed fusion strategy. More importantly, we find that the multi-view model combined with negative relational smoothing achieves the best performance on all four datasets. For instance, on Texas and Wisconsin, the accuracies of the multi-view model improve from 66.58% and 70.55% when $r = 0$ to 75.55% and 75.87% when $r = r^*$, respectively. These results suggest that negative relational smoothing can suppress noisy relational signals within each

Table 9
Interaction between the multi-view architecture and the smoothing coefficient r.

	Method	Cora	CiteSeer	Texas	Wisconsin
$r = 0$	One-hop only	77.91 ± 0.84	65.98 ± 0.60	56.78 ± 1.63	54.28 ± 1.71
	Two-hop only	77.36 ± 0.48	65.04 ± 0.82	71.10 ± 5.64	62.94 ± 1.91
	Multi-view	80.20 ± 0.53	67.41 ± 0.79	66.58 ± 9.63	70.55 ± 2.45
$r = r^*$	One-hop only	83.51 ± 0.48	72.71 ± 0.33	70.55 ± 1.71	70.65 ± 4.31
	Two-hop only	79.00 ± 0.46	70.66 ± 0.40	71.92 ± 1.24	71.39 ± 2.96
	Multi-view	**84.03 ± 0.51**	**73.36 ± 0.38**	**75.55 ± 2.09**	**75.87 ± 1.81**

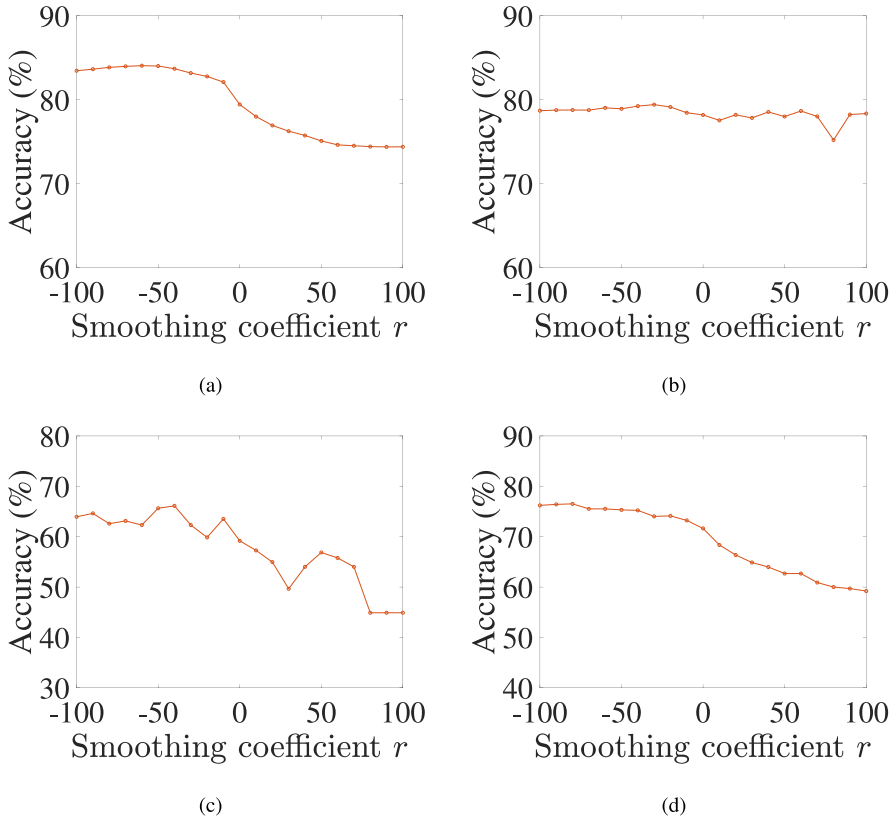


Fig. 6. Sensitivity analysis of the smoothing coefficient r on different datasets. (a) Cora; (b) Wiki-CS; (c) Cornell; (d) Wisconsin.

view based on the multi-view architecture. Overall, the results indicate that the multi-view architecture and negative relational smoothing can jointly contribute to improved performance.

5.5. Parametric sensitivity

In our proposed method, the smoothing coefficient r in Eq. (4) plays a crucial role in enhancing robustness to graph heterophily. Therefore, we evaluate in detail the sensitivity of the performance to different values of r. The results on two representative homophilic datasets (*i.e.*, Cora and Wiki-CS) and two heterophilic ones (*i.e.*, Cornell and Wisconsin) are reported in Fig. 6. We note that the performance deteriorates when $r > 0$, while shifting r to negative values generally leads to performance improvement. It can be inferred that negative smoothing penalizes overconfident contrastive pairs induced by heterophilic relations, thereby enhancing robustness. Besides, the performance remains relatively stable when $r < 0$, which indicates that the proposed method is insensitive to the choice of r.

In practical use, the smoothing coefficient r can be selected based on the silhouette score [54], which measures the compactness and separability of node embeddings. Specifically, for each candidate value of r, the NeRS model is first trained to obtain node representations, after which K-means clustering is applied to the learned embeddings. The silhouette score is then computed to quantify the clustering quality. The value of r that maximizes the silhouette score is adopted as the final hyperparameter. Moreover, the empirical results indicate that the model is not sensitive to this hyperparameter and thus r can be tuned without significant difficulty in practice.

6. Conclusion

In this paper, we present a new GCL framework with Negative Relational Smoothing (NeRS). Different from most existing GCL methods relying on graph filtering operations, we formulate the representation learning process as a noise-robust optimization problem, which enables effective representation learning under varying levels of heterophily. To further enhance expressiveness across both homophilic and heterophilic substructures, we design a multi-view architecture that facilitates the construction of reliable contrastive pairs. Experiments on ten benchmark datasets have revealed the effectiveness of our proposed method in handling both homophilic and heterophilic graphs.

Meanwhile, there still exist several limitations in this work. Specifically, the smoothing coefficient is globally shared and may not fully capture fine-grained structural variation. Additionally, the proposed method is mainly developed for undirected, simple, and static graphs. Nevertheless, the proposed relational smoothing strategy may provide useful insights for robust graph representation learning. In future

work, we will further explore adaptive smoothing for each node and extensions to directed and dynamic graphs.

CRediT authorship contribution statement

Sheng Wan: Writing – review & editing, Writing – original draft, Conceptualization. **Shougang Ren:** Methodology, Conceptualization. **Zicheng Zhao:** Visualization, Methodology. **Yan Zhu:** Writing – review & editing, Supervision. **Chen Gong:** Writing – review & editing, Supervision, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Nos: 62506171, 62336003, 12371510), the Natural Science Foundation of Jiangsu Province (No: BK20241469), and the Fundamental Research Funds for the Central Universities (No: YDZX2026052).

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.patcog.2026.113714.

Data availability

Data will be made available on request.

References

[1] M. Zhao, X. Huang, S. Zhang, A. Liu, Z. Lyu, Y. Wang, L. Cui, L. Bai, GraphProbe: Knowledge probing for graph representation learning, Pattern Recognit. (2025) 112518.
[2] Y. Hao, J. Ma, P. Zhao, G. Liu, X. Xian, L. Zhao, V.S. Sheng, Multi-dimensional graph neural network for sequential recommendation, Pattern Recognit. 139 (2023) 109504.
[3] B. Jin, X. Xu, Contemporaneous causal analysis of housing prices across Guangdong's major cities: Employing vector error-correction modeling and directed acyclic graphs, J. Uncertain Syst. (2026).
[4] J. Zhang, C. Wang, Q. Qian, S. Yin, Source-free domain adaptation for cross-domain remaining useful life prediction: A distributed federated learning perspective, Reliab. Eng. Syst. Saf. (2026) 112271.
[5] J. Zhang, K. Chen, F. Wu, Q. Qian, T. Huang, Y. Cheng, S. Yin, Remaining useful life prediction based on self-attention mechanism -sequential variational autoencoder: From a semi-supervised perspective, Adv. Eng. Inform. 71 (2026) 104242.
[6] Y. Xu, D. Huang, C.-D. Wang, J.-H. Lai, Deep image clustering with contrastive learning and multi-scale graph convolutional networks, Pattern Recognit. 146 (2024) 110065.
[7] B. Jin, X. Xu, A study of contemporaneous residential real estate price causation across major Jiangsu province cities: Methodology using vector error-correction models and directed acyclic graphs, Econ. Open (2025) 2550008.
[8] X. Xu, Contemporaneous causal orderings of CSI300 and futures prices through directed acyclic graphs, Econ. Bull. 39 (3) (2019) 2052–2077.
[9] C. Wang, Y. Liu, Y. Yang, W. Li, HeterGCL: Graph contrastive learning framework on heterophilic graph, in: International Joint Conference on Artificial Intelligence, 2024, pp. 2397–2405.
[10] C. Liu, C. Yu, N. Gui, Z. Yu, S. Deng, SimGCL: Graph contrastive learning by finding homophily in heterophily, Knowl. Inf. Syst. 66 (3) (2024) 2089–2114.
[11] D. He, J. Zhao, R. Guo, Z. Feng, D. Jin, Y. Huang, Z. Wang, W. Zhang, Contrastive learning meets homophily: Two birds with one stone, in: International Conference on Machine Learning, 2023, pp. 12775–12789.
[12] J. Chen, R. Lei, Z. Wei, PolyGCL: Graph contrastive learning via learnable spectral polynomial filters, in: International Conference on Learning Representations, 2024.
[13] D. Cartwright, F. Harary, Structural balance: A generalization of Heider's theory, Psychol Rev 63 (5) (1956) 277.
[14] H.-Y. Yao, C.-Y. Zhang, Z.-L. Yao, C.P. Chen, J. Hu, A recurrent graph neural network for inductive representation learning on dynamic graphs, Pattern Recognit. 154 (2024) 110577.
[15] Y. Fang, L. Zhang, Z. Wei, Z. Wu, S. Wang, C. Liu, X. Yang, ScaleGraph: A scalable self-supervised framework for cross-domain zero-shot graph learning, Pattern Recognit. (2025) 112482.
[16] X. Xu, Y. Zhang, An integrated vector error correction and directed acyclic graph method for investigating contemporaneous causalities, Decis. Anal. J. 7 (2023) 100229.
[17] X. Xu, Y. Zhang, Contemporaneous causality among office property prices of major Chinese cities with vector error correction modeling and directed acyclic graphs, J. Model. Manag. 19 (4) (2024) 1079–1093.
[18] J. Zhang, K. Chen, R. He, T. Huang, J. Tian, S. Wu, P. Yan, Y. Cheng, Remaining useful life prediction based on interpretable serialized variational autoencoder: A drift-diffusion stochastic equation perspective, IEEE Trans. Ind. Inform. (2026).
[19] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, in: International Conference on Learning Representations, 2017.
[20] Y. Li, L. Du, C. Guan, R. Cai, J. Gao, RA-GCN: Residual attention based graph convolutional network for multi-label pattern image retrieval, Pattern Recognit. (2025) 112647.
[21] K. Xu, C. Li, Y. Tian, T. Sonobe, K.-i. Kawarabayashi, S. Jegelka, Representation learning on graphs with jumping knowledge networks, in: International Conference on Machine Learning, 2018, pp. 5453–5462.
[22] J. Liu, M. He, G. Wang, Q.V.H. Nguyen, X. Shang, H. Yin, Imbalanced node classification beyond homophilic assumption, in: IJCAI International Joint Conference on Artificial Intelligence, 2023, pp. 7206–7214.
[23] J. Zhu, Y. Yan, L. Zhao, M. Heimann, L. Akoglu, D. Koutra, Beyond homophily in graph neural networks: Current limitations and effective designs, Adv. Neural Inf. Process. Syst. 33 (2020) 7793–7804.
[24] J. Choi, S. Hong, N. Park, S.-B. Cho, Gread: Graph neural reaction-diffusion networks, in: International Conference on Machine Learning, 2023, pp. 5722–5747.
[25] Y. Wang, S. Xiang, C. Pan, Improving the homophily of heterophilic graphs for semi-supervised node classification, in: IEEE International Conference on Multimedia and Expo, 2023, pp. 1865–1870.
[26] K. Kong, J. Chen, J. Kirchenbauer, R. Ni, C.B. Bruss, T. Goldstein, GOAT: A global transformer on large-scale graphs, in: International Conference on Machine Learning, 2023, pp. 17375–17390.
[27] Y. Li, B. Lin, B. Luo, N. Gui, Graph representation learning beyond node and homophily, IEEE Trans. Knowl. Data Eng. 35 (5) (2022) 4880–4893.
[28] Y. Liu, Y. Zheng, D. Zhang, V.C. Lee, S. Pan, Beyond smoothing: Unsupervised graph representation learning with edge heterophily discriminating, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 37, 2023, pp. 4516–4524.
[29] Z. Lv, S. Cheng, L. Xie, J. Li, M. Zhao, A graph contrastive learning network for change detection with heterogeneous remote sensing images, Pattern Recognit. (2025) 112394.
[30] Y. You, T. Chen, Y. Sui, T. Chen, Z. Wang, Y. Shen, Graph contrastive learning with augmentations, Adv. Neural Inf. Process. Syst. 33 (2020) 5812–5823.
[31] Y. Zhu, Y. Xu, F. Yu, Q. Liu, S. Wu, L. Wang, Deep graph contrastive representation learning, in: ICML Workshop on Graph Representation Learning and beyond, 2020.
[32] J. Qiu, Q. Chen, Y. Dong, J. Zhang, H. Yang, M. Ding, K. Wang, J. Tang, GCC: Graph contrastive coding for graph neural network pre-training, in: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2020, pp. 1150–1160.
[33] K. Hassani, A.H. Khasahmadi, Contrastive multi-view representation learning on graphs, in: International Conference on Machine Learning, 2020, pp. 4116–4126.
[34] Y. Yin, Q. Wang, S. Huang, H. Xiong, X. Zhang, AutoGCL: Automated graph contrastive learning via learnable view generators, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 36, 2022, pp. 8892–8900.
[35] Y. Liu, Y. Zheng, D. Zhang, H. Chen, H. Peng, S. Pan, Towards unsupervised deep graph structure learning, in: Proceedings of the ACM Web Conference, 2022, pp. 1392–1403.
[36] S. Suresh, P. Li, C. Hao, J. Neville, Adversarial graph augmentation to improve graph contrastive learning, Adv. Neural Inf. Process. Syst. 34 (2021) 15920–15933.
[37] P. Veličković, W. Fedus, W.L. Hamilton, P. Liò, Y. Bengio, R.D. Hjelm, Deep graph infomax, in: International Conference on Learning Representations, 2018.
[38] J. Wei, H. Liu, T. Liu, G. Niu, M. Sugiyama, Y. Liu, To smooth or not? When label smoothing meets noisy labels, in: International Conference on Machine Learning, 2022.
[39] R. Lei, Z. Wang, Y. Li, B. Ding, Z. Wei, EvenNet: Ignoring odd-hop neighbors improves robustness of graph neural networks, Adv. Neural Inf. Process. Syst. 35 (2022) 4694–4706.
[40] P. Sen, G. Namata, M. Bilgic, L. Getoor, B. Galligher, T. Eliassi-Rad, Collective classification in network data, AI Mag. 29 (3) (2008) 93–93.

[41] A. Bojchevski, S. Günnemann, Deep Gaussian embedding of graphs: Unsupervised inductive learning via ranking, in: International Conference on Learning Representations, 2018.
[42] P. Mernyei, C. Cangea, Wiki-CS: A wikipedia-based benchmark for graph neural networks, 2020, arXiv preprint arXiv:2007.02901.
[43] O. Shchur, M. Mumme, A. Bojchevski, S. Günnemann, Pitfalls of graph neural network evaluation, 2018, arXiv preprint arXiv:1811.05868.
[44] H. Pei, B. Wei, K.C.-C. Chang, Y. Lei, B. Yang, Geom-GCN: Geometric graph convolutional networks, in: International Conference on Learning Representations, 2020.
[45] O. Platonov, D. Kuznedelev, M. Diskin, A. Babenko, L. Prokhorenkova, A critical look at the evaluation of GNNs under heterophily: Are we really making progress? in: International Conference on Learning Representations, 2023.
[46] D. He, L. Shan, J. Zhao, H. Zhang, Z. Wang, W. Zhang, Exploitation of a latent mechanism in graph contrastive learning: Representation scattering, Adv. Neural Inf. Process. Syst. 37 (2024) 115351–115376.
[47] Y. Xu, S. Huang, H. Zhang, X. Li, Why does dropping edges usually outperform adding edges in graph contrastive learning? in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 39, 2025, pp. 21824–21832.
[48] Y. Zhu, Y. Xu, F. Yu, Q. Liu, S. Wu, L. Wang, Graph contrastive learning with adaptive augmentation, in: Proceedings of the ACM Web Conference, 2021, pp. 2069–2080.
[49] T. Xiao, H. Zhu, Z. Zhang, Z. Guo, C.C. Aggarwal, S. Wang, V.G. Honavar, Efficient contrastive learning for fast and accurate inference on graphs, in: International Conference on Machine Learning, 2024.
[50] S. Feng, B. Jing, Y. Zhu, H. Tong, ARIEL: Adversarial graph contrastive learning, ACM Trans. Knowl. Discov. Data 18 (4) (2024) 1–22.
[51] Z. Wang, B. Xu, Y. Yuan, H. Shen, X. Cheng, InfoNCE is a free lunch for semantically guided graph contrastive learning, in: Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2025, pp. 719–728.
[52] Y. Li, H. Zhang, Y. Yuan, Edge contrastive learning: An augmentation-free graph contrastive learning model, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 39, 2025, pp. 18575–18583.
[53] Y. Huang, J. Zhao, D. He, D. Jin, Y. Huang, Z. Wang, Does GCL need a large number of negative samples? Enhancing graph contrastive learning with effective and efficient negative sampling, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 39, 2025, pp. 17511–17518.
[54] P.J. Rousseeuw, Silhouettes: A graphical aid to the interpretation and validation of cluster analysis, J. Comput. Appl. Math. 20 (1987) 53–65.