

# LOHA: Direct Graph Spectral Contrastive Learning Between Low-Pass and High-Pass Views

Ziyun Zou<sup>1</sup>, Yinghui Jiang<sup>2</sup>, Lian Shen<sup>1</sup>, Juan Liu<sup>3</sup>, Xiangrong Liu<sup>1,2\*</sup>

<sup>1</sup>Department of Computer Science and Technology, Xiamen University,

<sup>2</sup>National Institute for Data Science in Health and Medicine, Xiamen University,

<sup>3</sup>Pen-Tung Sah Institute of Micro-Nano Science and Technology, Xiamen University,  
{zoezzy, shenlian}@stu.xmu.edu.cn, stardjyeah@gmail.com, {cecyliu, xrliu}@xmu.edu.cn

## Abstract

Spectral Graph Neural Networks effectively handle graphs with different homophily levels, with low-pass filter mining feature smoothness and high-pass filter capturing differences. When these distinct filters could naturally form two opposite views for self-supervised learning, the commonalities between the counterparts for the same node remain unexplored, leading to suboptimal performance. In this paper, a simple yet effective self-supervised contrastive framework, LOHA, is proposed to address this gap. LOHA optimally leverages low-pass and high-pass views by embracing "harmony in diversity". Rather than solely maximizing the difference between these distinct views, which may lead to feature separation, LOHA harmonizes the diversity by treating the propagation of graph signals from both views as a composite feature. Specifically, a novel high-dimensional feature named spectral signal trend is proposed to serve as the basis for the composite feature, which remains relatively unaffected by changing filters and focuses solely on original feature differences. LOHA achieves an average performance improvement of 2.8% over runner-up models on 9 real-world datasets with varying homophily levels. Notably, LOHA even surpasses fully-supervised models on several datasets, which underscores the potential of LOHA in advancing the efficacy of spectral GNNs for diverse graph structures.

## Introduction

Graphs have been widely used to model the relationships between entities. In homophilic graphs, connected nodes tend to share similar features; such as in citation networks (Kipf and Welling 2017). Conversely, in heterophilic graphs, the opposite is true, as seen in dating networks (Lei et al. 2022). Mining graph information has proven immensely helpful in fields such as bioinformatics (Ruan et al. 2023; Jiang et al. 2023), link prediction (Li et al. 2024; Lu et al. 2023), recommendation systems (Jiang, Huang, and Huang 2023; Yang et al. 2023), and traffic prediction (Wang et al. 2023). However, traditional graph neural networks (such as GCN) adopt a low-pass filter encoder to smooth the representations of neighboring nodes, limiting the adaptability to heterophilic graphs (Zhu et al. 2020a; Nt and Maehara 2019).

Thus, more explorations towards designing delicate Spectral Graph Neural Networks (SGNNs) are conducted (Deferrard, Bresson, and Vandergheynst 2016; Wang and Zhang 2022; Xu et al. 2024). Being benefited from the ability to fit both low-pass and high-pass signals, it is determined as an effective solution to deal with graphs with different levels of homophily (He et al. 2021; He, Wei, and Wen 2022). When low-pass filters amplify the low frequencies and mine feature smoothnesses, high-pass filters are considered preferable for heterophilic graphs and could capture feature differences (Bo et al. 2021; Xu et al. 2019; Ghose et al. 2023). In supervised learning tasks, SGNNs with polynomial approximation excel in both homophilic and heterophilic settings by adaptively fitting graph filters of arbitrary shapes (Liu et al. 2024; Guo and Wei 2023). Yet, for real-world applications, node labels and homophily levels of graphs are mostly uncovered. It remains necessary but challenging to learn appropriate filters for different graphs without label supervision automatically (Chen, Lei, and Wei 2024; Wan et al. 2024).

In self-supervised graph representation learning, graph contrastive learning (GCL) has achieved significantly superior performance (Guo et al. 2023; Zhang et al. 2023; You et al. 2021). GRACE (Zhu et al. 2020b) correlates graph views by pushing closer representations of the same node in different views and pushing apart representations of different nodes. SPAN (Lin, Chen, and Wang 2023) constructs two views to maximize spectral changes, declaring it as the best way to preserve structure invariance. However, only the validations on homophilic graphs are confirmed. Thus, researchers conduct more exploitation to fit in more graphs. Equipping with the natural differences and specialties among low-pass and high-pass filters, recent researches naturally construct contradictions between the views shaped by these counterparts. Nevertheless, both PolyGCL (Chen, Lei, and Wei 2024) and S3GCL (Wan et al. 2024) refine the models based on backpropagation between low-pass or high-pass views with the final views instead of the direct contradiction between the counterparts. We concern these approaches cannot fully exploit the characteristics of these two sets of filters and the improvements are solely gained through the complex combination of filters. However, if we treat low-pass and high-pass views as mutual 'negative' samples from the filter specialties perspective—contrary to the traditional approach of maximizing mutual information

\*Corresponding Author

Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

between different views—it would cause the features of a node to diverge. This leads to our core question: *what 'positive' features(commonalities) should the two counterparts share to reunite the node's features back, ensuring the identification and consistency of features for each node?*

To address the aforementioned questions, we revisit existing graph contrastive learning methods and propose a novel spectral-based diagram. To the best of our knowledge, we are the first to leverage the opposing characteristics of low-pass and high-pass filters to construct natural negative samples, thereby guiding the model to learn optimal filtering strategies and better features. While maximizing the differences between the views of the same node might cause node features to diverge, we reckon the "harmony" in "diversity" could be achieved by considering the propagation of graph signals as a composite feature, thereby enhancing the identification of nodes. Specifically, we introduce a novel high-dimensional feature named spectral signal trend, which describes the propagation difference of signals on the graph, to serve as a basis for the composite feature. With the extra composite feature, we propose **LOHA**, a direct graph spectral contrastive learning framework between **LO**w-pass and **High**-pAss views. LOHA attains state-of-the-art results across various real-world datasets with different homophilic levels under a self-supervised learning framework, surpassing the second-best model by an average of 2.8% on nine datasets. Performances on several heterophilic graph datasets even surpass full-supervised learning models, further showing the superiority of our framework and the experimental effectiveness of our theoretical discovery. Our contributions can be summarized as follows:

- We propose LOHA, a simple yet effective self-supervised framework that directly contrasts low-pass and high-pass views based on their natural distinct specialties without additional data augmentations.
- We highlight that directly considering low-pass and high-pass views as opposing sides could better utilize the specialties of filters but might lead to feature separation within a node. Further, we propose using the relatively stable cooperative graph signal trends as a composite feature to reunite features. Our theoretical and experimental analyses validate the effectiveness of this approach.
- Extensive experiments on 9 real-world datasets across different homophily levels are conducted to verify the superiority of LOHA. Additional detailed ablation studies further corroborate our theoretical results.

## Preliminary

### Notations

Define the graph data as  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , where  $\mathcal{V}$  represents the set of  $N$  nodes, and  $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$  represents the edges between them. The feature matrix  $\mathbf{X} \in \mathbb{R}^{N \times F}$  contains node features  $\mathbf{x}_i$  for each node  $\mathbf{v}_i$ . The adjacency matrix  $\mathbf{A} \in \mathbb{R}^{N \times N}$  has entries  $\mathbf{A}_{ij} = 1$  if  $e_{ij} \in \mathcal{E}$  and 0 otherwise. The normalized adjacency matrix is defined as  $\hat{\mathbf{A}} = \mathbf{D}^{-1/2} \mathbf{A} \mathbf{D}^{-1/2}$ , where  $\mathbf{D}$  is the diagonal degree matrix with  $d_{ii} = \sum_j \mathbf{A}_{ij}$ . Homophily describes the tendency of node connections. As

described in (Zhu et al. 2020a), edge homophily degree is defined as:

$$h = \frac{|\{(v_i, v_j) \in \mathcal{E} : y_i = y_j\}|}{|\mathcal{E}|} \quad (1)$$

This metric ranges from 0 to 1, where  $h$  close to 1 indicates high homophily, and vice versa.

### Graph Spectral Filtering

The graph Laplacian is given by  $\mathbf{L} = \mathbf{D} - \mathbf{A}$ , and its normalized form is  $\tilde{\mathbf{L}} = \mathbf{I} - \hat{\mathbf{A}}$ , where  $\mathbf{I}$  is the identity matrix.  $\tilde{\mathbf{L}}$  can be decomposed as  $\tilde{\mathbf{L}} = \mathbf{U} \mathbf{\Lambda} \mathbf{U}^\top$ , with  $\mathbf{\Lambda} = \text{diag}(\lambda_0, \dots, \lambda_{N-1})$  being the diagonal matrix of eigenvalues and  $\mathbf{U}$  being the matrix of eigenvectors. Graph filtering is defined as  $\mathbf{U}g(\mathbf{\Lambda})\mathbf{U}^\top \mathbf{X}$ , where  $g(\mathbf{\Lambda})$  is a graph filter function. Polynomial approximations of  $g(\mathbf{\Lambda})$  is defined as  $\sum_{k=0}^K w_k \mathbf{L}^k$ , thus using  $K$ -order truncation to fit  $g(\mathbf{\Lambda})$  and avoid the  $O(N^3)$  complexity of eigendecomposition.

### Pre-analysis

To simplify, we use two sets of simplest low-pass filters  $\mathcal{F}_L$  and high-pass filters  $\mathcal{F}_H$  as examples:

$$\begin{aligned} \mathcal{F}_L &= \mathbf{I} + \mathbf{D}^{-1/2} \mathbf{A} \mathbf{D}^{-1/2} = \mathbf{I} - \mathbf{L}, \\ \mathcal{F}_H &= \mathbf{I} - \mathbf{D}^{-1/2} \mathbf{A} \mathbf{D}^{-1/2} = \mathbf{I} + \mathbf{L}. \end{aligned} \quad (2)$$

Thus, the graph signals  $\mathbf{X}$  are filtered as follows:

$$\begin{aligned} \mathcal{F}_L *_{\mathcal{G}} \mathbf{X} &= \mathbf{U}[\mathbf{I} - \mathbf{\Lambda}]\mathbf{U}^\top \mathbf{X} = \mathcal{F}_L \cdot \mathbf{X}, \\ \mathcal{F}_H *_{\mathcal{G}} \mathbf{X} &= \mathbf{U}[\mathbf{I} + \mathbf{\Lambda}]\mathbf{U}^\top \mathbf{X} = \mathcal{F}_H \cdot \mathbf{X}. \end{aligned} \quad (3)$$

As Eq.2 shows, when low-pass filters aggregate node features with the features of its neighbors, which represents the commonalities among nodes, high-pass filters focus on the differences between node features (Bo et al. 2021). This lays a solid foundation for treating the information from these two perspectives as mutually negative samples. Thus, recent research employs low-pass and high-pass filters in a self-supervised diagram, combining the encoders from two sub-views to construct the final-view encoders, thereby benefiting from both filtering techniques (Chen, Lei, and Wei 2024; Wan et al. 2024). However, currently these methods simply compare sub-views with final views, without the direct contraction between low- / high-pass views. Therefore, an intuitive question arises: Does the superb performance really benefit from the expectation differences between low-frequency and high-frequency information or simply from the complex combination of different filters? To answer this question, we conduct a set of simple demo experiments.

We change the low-pass and high-pass filters in PolyGCL (2024) to band-stop  $\gamma^s$  and band-pass  $\gamma^p$  sets and remain other settings unchanged to conduct comparison experiments. The filter parameters are defined as follows(also illustrated in Fig. 1(b)):

$$\begin{aligned} \gamma_i^s &= \begin{cases} \gamma_0^s - \sum_{j=1}^i \gamma_j^s & 1 \leq i < \lfloor \frac{K+1}{2} \rfloor \\ \gamma_{K-i}^s & \lfloor \frac{K+1}{2} \rfloor \leq i \leq K \end{cases}, \\ \gamma_i^p &= \begin{cases} \sum_{j=0}^i \gamma_j^p & 1 \leq i < \lfloor \frac{K+1}{2} \rfloor \\ \gamma_{K-i}^p & \lfloor \frac{K+1}{2} \rfloor \leq i \leq K \end{cases}, \end{aligned} \quad (4)$$

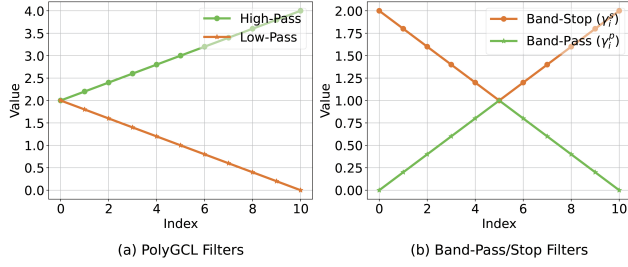


Figure 1: Initialized filters for demo experiments.

where the initial value for  $r_0^s$ ,  $r_K^s$  and  $r_0^p, r_k^p$  are set to 2 and 0, respectively, with all the other  $\gamma$  are initialized to  $\frac{2.0}{K}$ . Noted that all  $\gamma$  are trainable as in PolyGCL. Results are shown in Table 1, from where we can see that band-stop and band-pass filters achieve competitive results compared with low-high filter sets, which are expected to benefit from different specialties to capture similarities or differences from features. Thus, we argue whether constructing positive and negative samples through data augmentation between sub-views and final-view instead of the natural specialties of low-high pass filters is an optimal way to benefit from these two views or not. This leads to our main question: *when the direct contradiction exists in low-pass and high-pass views, what are the commonalities?* In other words, different from traditional contrastive learning like GRACE (2020b), which pushes representations in different views of the same node close, we propose to minimize the similarities between low-pass and high-pass views for one node to maximize the functionality of these two counterparts. However, when separating the features of one node between the low-high pass view, how to reunite features back for one node to be identifiable with other entities becomes a question.

## Theoretical Analysis

We reckon that the solution is to find an extra feature to serve as a landmark during training process. It should satisfy the following two requirements: (1) Relative stability. The relative differences between nodes cannot be changed to ensure consistency. Additionally, the features should be relatively stable to ensure a smooth training process. (2) Composite. A composite feature that combines the low-pass and high-pass views as a whole to help features reunite. Inspired by *Dirichlet Energy*, a global measure of how 'non-smooth' the graph information is, we propose a new basic feature to fit the first requirement. (Karhadkar, Banerjee, and Montufar 2023) define the *Dirichlet Energy*  $\mathcal{E}$  in vector field as:

**Definition 1 (Dirichlet Energy  $\mathcal{E}$ ).** Let  $\mathcal{G}$  be a connected graph with adjacency matrix  $\mathbf{A}$  and normalized Laplacian  $\tilde{\mathbf{L}}$ . For  $i \in \mathcal{V}$ , let  $d_i$  denote the degree of node  $i$ . For a vector field  $\mathbf{X} \in \mathbb{R}^{n \times p}$ ,  $x_{i,k}$  denotes the  $k$ th feature of  $\mathbf{x}_i$ , we define

$$\mathcal{E}(X) := \frac{1}{2} \sum_{i,j,k} \mathbf{A}_{i,j} \left( \frac{x_{i,k}}{\sqrt{d_i}} - \frac{x_{j,k}}{\sqrt{d_j}} \right)^2 = \text{Tr} \left( \mathbf{X}^T \tilde{\mathbf{L}} \mathbf{X} \right)$$

As *Dirichlet energy* is a global measure, we define the *Spectral Signal Trend* to describe the frequency variation of

Datasets		Low/High	Band-Pass/Stop
Homophilic	Cora	<b>86.5±0.8</b>	86.3±0.9
	Citeseer	<b>79.1±0.6</b>	78.9±0.8
	PubMed	86.9±0.3	<b>87.0±0.3</b>
Heterophilic	Cornell	82.3±3.4	<b>83.2±4.3</b>
	Texas	<b>86.9±1.5</b>	85.7±2.3
	Wisconsin	84.0±2.5	<b>86.2±3.5</b>

Table 1: Experiments results for comparison between low-/high- pass filters and band-stop/pass filters based on PolyGCL framework. Metrics are node classification accuracy(%).

the node information on the vector field to more finely display the changing trend of one node information propagation across the node neighbors.

**Definition 2 (Spectral Signal Trend  $\mathcal{T}_r$ ).** Given node feature  $\mathbf{x}_i \in \mathbb{R}^p$ , the spectral signal trend of node  $v_i$  across the graph can be defined as

$$\mathcal{T}_r(\mathbf{x}_i) = \left( \frac{x_{i,k}}{\sqrt{d_i}} - \sum_{j \in n_i^1} \frac{x_{j,k}}{\sqrt{d_j}} \right)^2, \quad (5)$$

for  $n_1 \in n_i^1$  stands for the 1-hop neighbors of node  $i$ .

The *Spectral Signal Trend* characterizes the changes in node features on the graph, compared with node features, it introduces the additional topological message. Further, since the relative differences of  $\mathcal{T}_r(\mathbf{x})$  between nodes are only dependent on the original nodes feature, it meets our first requirement. Secondly, to eliminate the influence of the differences between low-pass and high-pass views and propose a composite feature (Eq.6), where  $\Delta \mathbf{x}_i$  denotes aggregated original feature differences between node  $i$  and its neighbors. We consider the two simplest composition ways, subtraction or addition of  $\mathcal{T}_r(\mathbf{x})$  between low-pass and high-pass views, where  $\Delta g(\mathbf{\Lambda})$  is equal to  $(g^l(\mathbf{\Lambda}) - g^h(\mathbf{\Lambda}))$  and  $(g^l(\mathbf{\Lambda}) + g^h(\mathbf{\Lambda}))$  respectively. However, which one is more stable?

$$\mathcal{C}(\mathbf{x}_i^l, \mathbf{x}_i^h) = |\mathbf{U} \Delta g(\mathbf{\Lambda}) \mathbf{U}^T \Delta \mathbf{x}_i|. \quad (6)$$

To better conduct theoretical analysis, derived from (Ma et al. 2022), we make the following assumptions: consider a graph  $\mathcal{G}$ , where each node  $i$  has features  $\mathbf{x}_i \in \mathbf{R}^f$ . Denote a composition function  $\mathcal{C}(\mathbf{x}_i^l, \mathbf{x}_i^h)$  between low-pass and high-pass views. We assume that (1)  $\mathcal{C}$  for all nodes are sampled from the same distribution; (2)  $\mathcal{C}_i$  is independent to across each dimension; (3) The features in  $\mathbf{X}$  are bounded by a positive scalar  $B$ , thus  $\max_{i,j} |\mathbf{X}[i,j]| \leq B$ . Suppose that given graph filtering function  $\mathbf{U}g(\mathbf{\Lambda})\mathbf{U}^T \mathbf{X}$ , the bound of transformed feature  $\mathbf{X}'$  is  $\max_{i,j} |\mathbf{X}'[i,j]| \leq \max(\lambda') \cdot B$ , where  $\max(\lambda')$  denotes the maximum eigenvalue of  $\mathbf{U}g(\mathbf{\Lambda})\mathbf{U}^T$ . Thus, we have:

**Theorem 1.** Given the expectation of  $\mathcal{C}_i$  as  $\mathbb{E}(\mathcal{C}_i)$ ,  $d_i$  as the degree of node  $i$ . For any  $t > 0$ , the probability that the

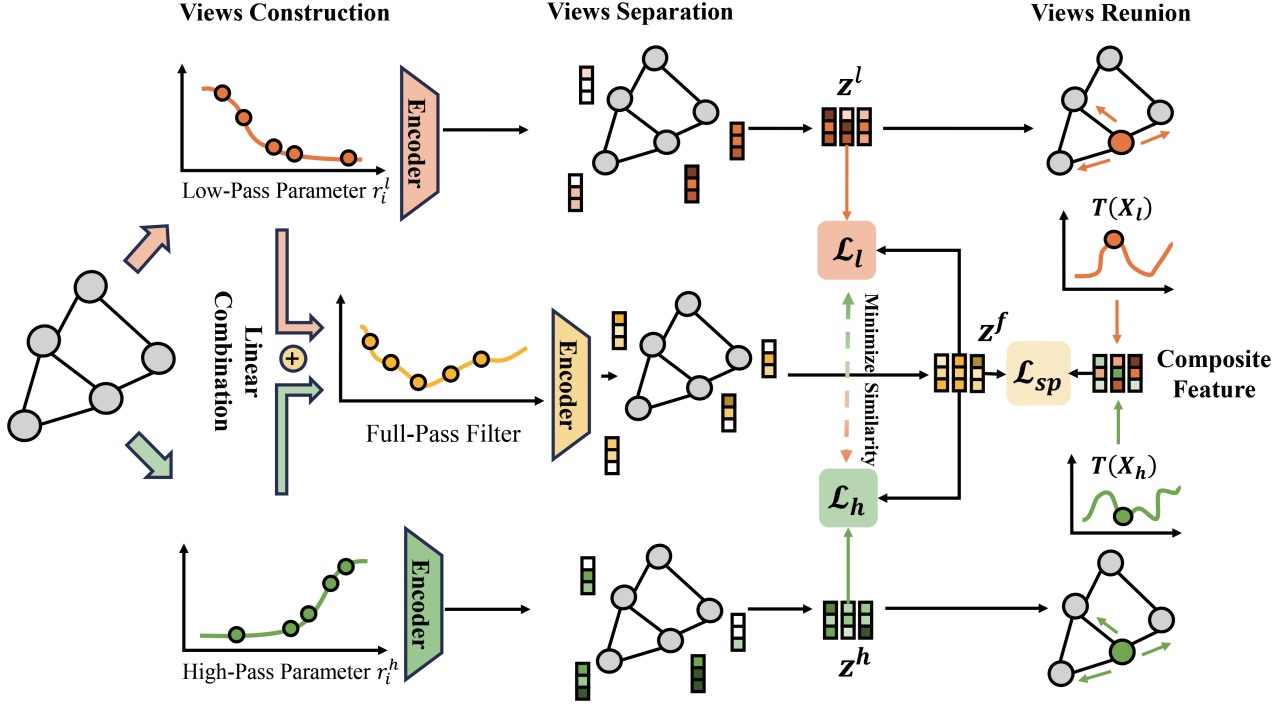


Figure 2: Illustration for the overall Pipeline of LOHA. Based on the natural opposite specialties between low-pass view and high-pass view, we minimize the mutual information between these two sides to construct contrastive views and separate sub-view features to learn better filters. Further, in the views reunion section, we propose to use a composite feature based on the spectral signal trends of the two views to reunite node features.

distance between  $C_i$  and its expectation is larger than  $t$  is bounded by:

$$\mathbb{P}(\|C_i - \mathbb{E}[C_i]\|_2 \geq t) \leq 2 \cdot f \cdot \exp\left(-\frac{d_i t^2}{2(\max(\lambda'))^4 B^2 f}\right). \quad (7)$$

Detailed proofs are provided in Appendix. From Theorem 1, smaller  $\max(\lambda')$  lead to more stable  $C_i$ , which meets our requirements. Thus, we subtract low-pass and high-pass views to form the composite features.

## Method

The overall pipeline of LOHA is illustrated in Fig. 2. In this section, we will provide a detailed explanation of our proposed method, LOHA, from the two main parts, Views Construction and Loss Design. In the Loss Design section, we separately introduce our loss design based on Views Separation and Views Reunion parts to show our insight more clearly.

### Views Construction

Following (He, Wei, and Wen 2022), we adopt Chebyshev polynomials with interpolation as base polynomials, which can be formulated as  $\sum_{k=0}^K w_k T_k(\hat{\mathbf{L}})\mathbf{X}$ , where  $\hat{\mathbf{L}}$  =

$2\tilde{\mathbf{L}}/\lambda_{\max} - \mathbf{I}$ , and  $w_k$  is reparameterized as follows:

$$w_k = \frac{2}{K+1} \sum_{j=0}^K \gamma_j T_k(x_j), \quad (8)$$

where  $x_j = \cos\left(\frac{j+1/2}{K+1}\pi\right)$ , ( $j = 0, \dots, K$ ) denote the Chebyshev nodes for  $T_{K+1}$ . As for the learnable parameter  $\gamma_j$ , we propose to use a learnable sliding cosine-parametric-based formulation to build our low-pass and high-pass filters based on parameters  $\gamma_j^l$  and  $\gamma_j^h$  as follows:

$$\begin{aligned} \gamma_j^l &= \sigma(\beta_a^l) - \frac{1}{2}\sigma(\beta_b^l) \left(1 + \cos\left(1 + \frac{\frac{1}{2}\tanh(\delta_l) + j}{K}\pi\right)\right), \\ \gamma_j^h &= \sigma(\beta_a^h) + \frac{1}{2}\sigma(\beta_b^h) \left(1 + \cos\left(1 + \frac{\frac{1}{2}\tanh(\delta_h) + j}{K}\pi\right)\right), \end{aligned} \quad (9)$$

where  $\sigma = \text{ReLU}(\cdot)$ ,  $\beta_a^l$  and  $\beta_b^l$  are initialized to 0 and 2 respectively whereas  $\beta_b^l$  and  $\beta_b^h$  are set to 2 and these four parameters are all trainable during the learning process. Noted that compared with filters proposed by (Wan et al. 2024), we add the learnable  $\tanh(\delta)$  parameters on  $j$  to achieve our “sliding” propose, which is a simple trick but shows powerful capability. All  $\delta$  are initialized to 0 and  $\frac{1}{2}\tanh(\cdot)$  function constrain the values to lie inside  $(-0.5, 0.5)$ , which ensures the  $\gamma_i$  will not intersect and strictly obey the low-pass or high-pass sequences accordingly. As stated in (Wan

et al. 2024), when the cosine-parameterized strategy effectively emphasizes key frequencies while reducing less important ones, compared with traditional equidistant interpolation, we believe that adding a sliding strategy could give the filters more flexibility. The capability of this strategy is validated in ablation experiments. To sum up, substituting  $\gamma_i$  with  $\gamma^l$  and  $\gamma^h$  in Eq.8 respectively, we could obtain two distinct views as in Eq.10.

$$\mathbf{Z}_l = MLP \left( \sum_{k=0}^K w_k^l T_k(\tilde{\mathbf{L}}) \mathbf{X} \right), \mathbf{Z}_h = MLP \left( \sum_{k=0}^K w_k^h T_k(\tilde{\mathbf{L}}) \mathbf{X} \right), \quad (10)$$

where  $MLP$  are shared between two views. Given feature  $\mathbf{Z}_l$  and  $\mathbf{Z}_h$ , we obtain the final embedding through linear combination  $\mathbf{Z}_f = \alpha \mathbf{Z}_l + \beta \mathbf{Z}_h$ . Next, we introduce our designed loss for contrastive learning.

### Loss Design

Our loss function consists of three parts:

$$\mathcal{L} = \mathcal{L}_l + \mathcal{L}_h + \mu \mathcal{L}_{sf}, \quad (11)$$

for which  $\mathcal{L}_l$  and  $\mathcal{L}_h$  belong to views separation part to push node features in low-pass view and high-pass view apart and  $\mathcal{L}_{sf}$  as views reunion part to reunite two views of one node to be aggregated. Hyper-parameter  $\mu$  is to balance  $\mathcal{L}_{sf}$  with  $\mathcal{L}_l$  and  $\mathcal{L}_h$ .

**Views Separation.** Inspired by InfoNCE loss (Chen et al. 2020), we formulate our  $\mathcal{L}_l$  and  $\mathcal{L}_h$  as follows:

$$\mathcal{L}_l = -\frac{1}{|\mathcal{V}|} \sum_{v_i \in \mathcal{V}} \left( \log \frac{s(z_i^f, z_i^l)}{\sum_{v_p \in \mathcal{V} \setminus v_i} s(z_i^f, z_p^l) + s(z_i^l, z_i^h)} \right), \quad (12)$$

$$\mathcal{L}_h = -\frac{1}{|\mathcal{V}|} \sum_{v_i \in \mathcal{V}} \left( \log \frac{s(z_i^f, z_i^h)}{\sum_{v_p \in \mathcal{V} \setminus v_i} s(z_i^f, z_p^h) + s(z_i^l, z_i^h)} \right).$$

Here, we define  $s(z_i^f, z_p^l) = \exp(\text{sim}(z_i^f, z_p^l)/\tau)$ , where  $z_i^f$ ,  $z_i^l$ , and  $z_i^h$  are embeddings from the final full-pass, low-pass, and high-pass filters respectively, and  $\tau$  is the contrast temperature. To enhance the training of the final full-pass filters, we consider the features obtained from the full-pass view  $z_i^f$  and the low-/high-pass views ( $z_i^l/z_i^h$ ) of a node  $v_i$  as the positive pair in the design of  $\mathcal{L}_l$  and  $\mathcal{L}_h$ . Furthermore, we treat features of other nodes  $v_p$  from the corresponding low-/high-pass ( $z_p^l/z_p^h$ ) as negative samples of  $z_i^f$ . To enforce the low-pass view to learn feature smoothness and the high-pass view to highlight neighborhood differences, we treat  $z_i^l$  and  $z_i^h$  of node  $v_i$  as negative samples of each other. This approach leverages the distinct properties of these two views effectively.

**Views Reunion.** As we discussed before, when  $z_i^l$  and  $z_i^h$  are both features for node  $v_i$  and we minimize their mutual information in the views separation part, we propose to use the differentiate of spectral signal trends of nodes to reunite node features. Our  $\mathcal{L}_{sf}$  are determined:

$$\mathcal{L}_{sf} = -\frac{1}{|\mathcal{V}|} \sum_{v_i \in \mathcal{V}} \left( \log \frac{s(z_i^f, C^-(z_i^l, z_i^h))}{\sum_{v_p \in \mathcal{V} \setminus v_i} s(z_i^f, C^-(z_p^l, z_p^h))} \right), \quad (13)$$

where  $C^-(\mathbf{x}_i^l, \mathbf{x}_i^h) = \left| \mathbf{U} \left( g^l(\mathbf{\Lambda}) - g^h(\mathbf{\Lambda}) \right) \mathbf{U}^T \Delta \mathbf{x}_i \right|$ .

Noted that here we define positive and negative pairs in a relative concept, for which we reckon that the similarity between features of the node  $v_i$  from full-filter views  $z_i^f$  and spectral differences  $C^-(z_i^l, z_i^h)$  should be higher than that with other nodes  $v_p$ .

What's more, to further validate the effectiveness of Eq.5, we propose two variants of it when computing  $C^-(\mathbf{x}_i^l, \mathbf{x}_i^h)$ . Specifically, by changing the input  $\mathbf{x}_i$ , we have:

$$\mathcal{T}_{r1}(\hat{\mathbf{x}}_i) = \left( \hat{\mathbf{x}}_i - \sum_{j \in n_i^1} \hat{\mathbf{x}}_j \right)^2 \quad (14)$$

$$\text{where } \hat{\mathbf{x}}_i = \frac{\tilde{\mathbf{x}}_i}{\sum_i \tilde{\mathbf{x}}_i} \quad \text{and} \quad \tilde{\mathbf{x}}_i = \sum_k \frac{\mathbf{x}_{i,k}}{\sqrt{d_i}},$$

$$\mathcal{T}_{r3}(\hat{\mathbf{x}}_i) = \left( \hat{\mathbf{x}}_i - \sum_j \hat{\mathbf{x}}_j \right)^2 \quad (15)$$

$$\text{where } \hat{\mathbf{x}}_i = \left[ \hat{\mathbf{x}}_i \parallel \text{mean} \left( \frac{\mathbf{x}_{i,k}}{\sqrt{d_i}} \right) \parallel \text{std} \left( \frac{\mathbf{x}_{i,k}}{\sqrt{d_i}} \right) \right],$$

where ' $\parallel$ ' means feature concatenation. Thus, the feature is squeezed from  $\mathcal{N} \times \mathcal{D}_{hidden}$  to  $\mathcal{N} \times 1$  for Eq.14 and  $\mathcal{N} \times 3$  for Eq.15, thus holds different information density.

## Experiment

In this section, we present a comprehensive evaluation of LOHA, through a series of node classification experiments on 9 real-world datasets. Comparison between LOHA with other baselines and ablation studies validate the effectiveness of LOHA and help us to gain further insights.

### Baselines and Settings

To show the superiority of LOHA, we select several common and typical full-supervised networks and novel self-supervised graph contrastive learning frameworks as our baseline as follows: (1) Full-Supervised Networks: MLP, GCN (2017), ChebNet (2016), BernNet (2021), and ChebNetII (2022); (2) Self-supervised Graph Contrastive Learning Frameworks: DGI (Veličković et al. 2019), GraphCL (You et al. 2020), GRACE (2020b), BGRL (Thakoor et al. 2022), GREET (2023), GraphACL (Xiao et al. 2023) and PolyGCL (2024).

**Evaluation Protocol.** Following (Veličković et al. 2019), we use the widely adopted two-stage linear evaluation pipeline in graph contrastive learning. In the first stage, the node features and structure information are self-trained without any label information. After convergence, the output feature of the first stage would then be fixed and used to train, validate, and test through a simple classifier. We follow the training and validation strategies as (Chien et al. 2021), where nodes are randomly split into 60%, 20%, and 20%. All comparative methods share the same fixed random splits. Output embedding size and hyper-parameters in stage 2 are also fixed for fair comparison. More detailed settings can be found in Appendix.

Methods	Cornell	Texas	Wisconsin	Actor	Chameleon	Amazon	Cora	Citeseer	PubMed	Avg.
Full-Supervised										
MLP	86.4±2.1	87.0±2.3	86.3±3.8*	40.2±0.5	46.1±1.2	39.1±1.7	76.5±0.8	74.3±1.0	85.9±0.4	69.1
GCN	68.6±3.8	78.2±3.1	74.5±3.1	34.2±0.9	58.8±2.4	43.3±0.4	85.3±1.0	76.7±0.5	85.7±0.2	67.3
ChebNet	83.6±7.5	85.3±2.5	82.5±4.9	37.6±1.7	59.7±2.4	47.8±0.7*	85.5±1.2	77.0±0.7	86.7±0.4	71.7
BernNet	86.7±0.4	87.2±1.5	84.4±1.8	41.8±1.3*	69.3±1.5	44.3±0.1	87.4±0.9*	78.0±0.9*	86.5±0.2	74.0
ChebNetII	88.4±1.6*	89.3±2.3*	85.6±2.5	40.1±0.8	71.5±1.2*	45.1±0.5	86.2±1.1	76.5±0.9	87.3±0.3*	74.4*
Self-Supervised										
DGI	72.4±6.9	81.5±3.2	76.4±1.9	32.6±1.2	58.2±0.8	42.8±0.4	84.9±0.9	76.4±0.9	83.1±0.6	67.6
GraphCL	61.5±5.7	68.0±5.0	61.5±3.9	32.8±1.3	59.2±1.4	39.0±0.3	<u>86.4±0.7</u>	78.0±0.5	85.2±0.2	63.5
GRACE	60.7±9.8	75.6±2.8	72.7±3.2	31.9±1.0	59.6±1.3	43.3±0.2	83.5±0.7	74.2±0.8	81.8±0.6	64.8
BGRL	59.8±3.0	70.7±2.8	63.1±4.0	33.1±0.7	64.0±1.3	41.6±0.5	84.3±0.7	74.5±1.0	83.2±0.3	63.8
GREET	78.4±3.9	80.0±3.9	83.6±3.7	38.3±0.9	63.2±1.0	42.0±0.2	85.7±1.4	<u>78.0±0.4</u>	85.6±0.3	70.5
GraphACL	60.2±0.5	70.8±0.7	73.6±0.5	29.6±0.1	72.2±2.3	41.4±0.6	86.3±1.0	<u>77.4±0.7</u>	85.8±0.3	66.4
PolyGCL	82.4±3.4	87.9±1.5	84.0±2.5	<u>41.8±0.8</u>	<u>71.9±1.2</u>	47.3±0.4	<b>87.6±0.7*</b>	<b>78.9±0.6*</b>	<u>86.9±0.3</u>	74.3
<b>LOHA</b>	<b>90.7±1.3*</b>	<b>93.4±1.3*</b>	<b>95.9±0.9*</b>	<b>42.8±0.8*</b>	<b>73.0±0.9*</b>	<b>48.8±0.3*</b>	85.1±1.3	77.1±0.5	<b>87.3±0.3*</b>	<b>77.1*</b>
LOHA-var-3	<u>89.0±2.3</u>	<u>91.4±1.5</u>	94.3±1.5	40.5±0.8	70.2±0.8	<u>48.0±0.5</u>	83.1±0.9	76.2±0.3	86.8±0.8	<u>75.5</u>
LOHA-var-1	88.4±2.1	89.3±1.6	<u>95.5±1.4</u>	38.7±1.2	68.4±1.0	47.3±0.8	83.5±1.0	74.9±0.8	86.3±1.3	74.7

Table 2: Mean node classification accuracy (%) on real-world graph. The best full-supervised results are marked by \*. For self-supervised models, **boldface** letters indicate the best results and underlining letters denote the runner-up results. The best results that surpass full-supervised performance are also marked with \*. LOHA-var-3 and LOHA-var-1 are variants correspond to models with Eq.15 and 14 respectively.

## Datasets

We choose widely used real-world datasets with different homophily levels to evaluate the performance of LOHA. (1) Homophilic Graphs: Cora, Citeseer, and PubMed from (Yang, Cohen, and Salakhudinov 2016). (2) Heterophilic Graphs: Cornell, Texas, Actor and Wisconsin from (Pei et al. 2020); Chameleon from (Rozemberczki, Allen, and Sarkar 2021); Amazon-ratings (Amazon for short in tables) from (Platonov et al. 2023). Details of the dataset can be found in Appendix.

## Results

The results, as presented in Table 2, demonstrate that LOHA consistently achieves strong performance across all datasets, with particularly notable success on heterophilic graphs. Specifically, LOHA surpasses all baseline models on 7 out of 9 benchmarks, indicating its robustness and adaptability across different graph structures. Notably, on the heterophilic graphs *Cornell*, *Texas*, and *Wisconsin*, LOHA exhibits substantial relative improvements over the runner-up models, achieving gains of 8.3%, 5.5%, and 11.9% respectively. These improvements highlight LOHA’s ability to effectively capture and leverage the structural nuances inherent in graphs with lower homophily levels.

Moreover, LOHA not only outperforms self-supervised baselines but also surpasses fully-supervised models on all heterophilic datasets, underscoring its capacity to generalize well in scenarios where node labels are limited or unavailable. This is particularly significant considering that fully-supervised methods typically have access to ground-truth labels during training. LOHA variants further reinforce the

framework’s effectiveness, consistently delivering superior performance across various settings. For example, LOHA-var-3 outperforms the next best self-supervised model by an average of 1.2% across all datasets, demonstrating the robustness of the proposed approach. Similarly, LOHA-var-1 also shows competitive results.

It is noteworthy that the baseline model, PolyGCL, utilizes a node shuffling strategy to increase the number of negative samples, which is actually a kind of data augmentation. In contrast, our LOHA is an ‘augmentation-free’ method that simply leverages the intrinsic differences between low-pass and high-pass views, but still shows incredible improvements. To more intuitively verify the effectiveness of the proposed loss, we replace the original loss function of PolyGCL with LOHA ones. The results, presented in Table 3, demonstrate the competitive performance of our approach and the advantages gained through the direct contradiction between low-pass and high-pass views.

**Variants of LOHA.** To analyze the effect of each part, we introduce three variants of LOHA: (1) w/o sliding: deleting the sliding trick in Eq.9; (2) w/o  $\mathcal{L}_{sp}$ : deleting the  $\mathcal{L}_{sp}$  in Eq.13; (3) w/o contrast: deleting  $s(z_i^l, z_i^h)$  in Eq.12. The results are shown in Fig. 3, which reveal that the full model consistently achieves the best performance across all datasets. The experimental results demonstrate the effectiveness of each of these three parts. Particularly, the absence of  $\mathcal{L}_{sp}$  leads to considerable performance degradation, especially on homophilic graphs like *Cora*. It indicates that  $\mathcal{L}_{sp}$  plays a crucial role in maintaining the model’s performance on graphs where feature smoothness is more prevalent. Similarly, when the contrastive term is excluded, this variant

Datasets	PolyGCL	Poly-LOHA	Poly-LOHA-3
Cora	<b>87.57±0.73</b>	86.21±0.65	83.32±0.13
Citeseer	<b>79.13±0.58</b>	78.05±0.91	75.21±0.38
PubMed	86.91±0.34	<b>86.96±0.32</b>	85.45±0.41
Cornell	82.36±3.42	<b>86.73±2.34</b>	85.12±3.32
Texas	87.92±1.46	<b>91.51±0.92</b>	90.36±1.45
Wisconsin	84.03±2.52	<b>91.06±1.64</b>	88.52±1.98
Actor	41.76±0.75	<b>43.12±0.63</b>	42.08±0.83
Chameleon	71.92±1.16	<b>73.25±1.14</b>	72.04±1.35
Amazon	47.31±0.41	<b>48.61±0.24</b>	46.38±0.51

Table 3: Experiments results(Node Classification Accuracy(%)) for ablation study between origin PolyGCL and the variant of PolyGCL which substitutes the loss function with LOHA loss(Poly-LOHA) and LOHA loss variant with Eq.15 (Poly-LOHA-3).

shows the largest performance drop across all datasets, underscoring its importance in capturing the essential differences between low-pass and high-pass views. Although sliding interpolation is just a minor trick, it also displays certain advantages. Overall, the results suggest that while each component contributes to LOHA’s effectiveness, their combination is particularly powerful, achieving optimal results by balancing the strengths of each.

## Related Work

### Spectral Graph Neural Network

As a low-pass filter, the suboptimal performance of traditional GCN on heterophilic graph simulated the exploration and design of complex SGNN. (Defferrard, Bresson, and Vandergheynst 2016) introduces ChebNet, which employs Chebyshev polynomials to achieve localized filters with reduced computational complexity, making it suitable for large-scale graphs. Furthermore, (He, Wei, and Wen 2022) enhances the original ChebNet by eliminating the Runge phenomenon. BernNet (2021) leverages the flexibility of Bernstein polynomials to design adaptive spectral filters that can better handle heterophilic graphs by learning complex filter shapes. Similarly, ASWT-SGNN (2024) uses flexible spectral wavelet transforms to capture multiscale features and enhance graph representation learning under self-supervision. However, when these methods work well in the supervised domain, how to obtain a set of optimal filters automatically and better utilize the advantages of both low-frequency and high-frequency information in self-supervised workflow remains uncovered.

### Graph Contrastive Learning

Graph contrastive learning aims to learn consistent representations from different views to mine invariant information of graph data. Through experimental discovery, SPAN (2023) points out that the insight of augmented GCL is to maximize spectral changes, thus proposing an augmentation method to maximize the spectral differences between two views. However, SpCo (2022) deems that the inner key lies

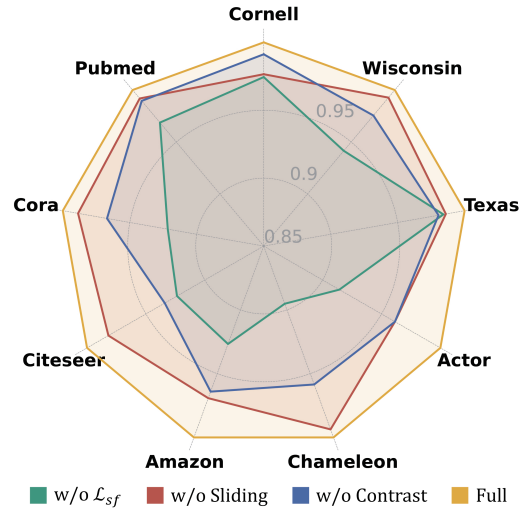


Figure 3: Experiments results for ablation study between LOHA and its variants. In order to better demonstrate the effect, we used the results of the full model as the denominator for each result for normalization.

in the difference between high-frequency and low-frequency parts, which the former one should be higher than the counterpart. Without augmentation, some methods contrast opposing views by inputting the same data into different encoders and pushing together the representations of the same entities from different views. For example, PolyGCL(2024) utilizes polynomial filters to construct low-pass and high-pass views and combine them through linear addition. Similarly, S3GCL(2024) uses MLP to better correlate the results of two views thus reducing inference time. However, these two methods lack a direct comparison between the two sub-views, which raises doubts about whether they truly and effectively apply the advantages of low-frequency and high-frequency information as expected. Thus, unlike the traditional methods, we propose to minimize the similarities between low-pass and high-pass views to force the model to maximize their specialties thus obtain better features.

## Conclusion

This paper introduces LOHA, a novel graph contrastive learning (GCL) framework that leverages the inherent opposition between low-pass and high-pass views. By focusing on maximizing the distinction between these views, LOHA enhances filter effectiveness and improves representation learning. Recognizing that excessive differentiation may cause node feature separation, we further introduce a high-dimensional feature, spectral signal trends, to serve as an extra feature that encourages feature reunion for individual nodes. In essence, LOHA is a simple yet powerful framework that optimizes filter learning through contrastive views and subsequently reunites node features using spectral signal trends. Extensive experiments and ablation studies demonstrate LOHA’s superiority across both homophilic and heterophilic graphs.

## Acknowledgments

This work was supported by the National Natural Science Foundation of China (grant nos. 62072384 and 62372391) and Fujian Provincial Major Science and Technology Project(2022YZ040011).

## References

- Bo, D.; Wang, X.; Shi, C.; and Shen, H. 2021. Beyond Low-frequency Information in Graph Convolutional Networks. In *AAAI*. AAAI Press.
- Chen, J.; Lei, R.; and Wei, Z. 2024. PolyGCL: GRAPH CONTRASTIVE LEARNING via Learnable Spectral Polynomial Filters. In *The Twelfth International Conference on Learning Representations*.
- Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, 1597–1607. PMLR.
- Chien, E.; Peng, J.; Li, P.; and Milenkovic, O. 2021. Adaptive Universal Generalized PageRank Graph Neural Network. In *International Conference on Learning Representations*.
- Defferrard, M.; Bresson, X.; and Vandergheynst, P. 2016. Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering. In *Advances in Neural Information Processing Systems*.
- Ghose, A.; Zhang, Y.; Hao, J.; and Coates, M. 2023. Spectral augmentations for graph contrastive learning. In *International Conference on Artificial Intelligence and Statistics*, 11213–11266. PMLR.
- Guo, X.; Wang, Y.; Wei, Z.; and Wang, Y. 2023. Architecture matters: Uncovering implicit mechanisms in graph contrastive learning. *Advances in Neural Information Processing Systems*, 36: 28585–28610.
- Guo, Y.; and Wei, Z. 2023. Graph neural networks with learnable and optimal polynomial bases. In *International Conference on Machine Learning*, 12077–12097. PMLR.
- He, M.; Wei, Z.; and Wen, J.-R. 2022. Convolutional Neural Networks on Graphs with Chebyshev Approximation, Revisited. In *NeurIPS*.
- He, M.; Wei, Z.; Xu, H.; et al. 2021. Bernnet: Learning arbitrary graph spectral filters via bernstein approximation. *Advances in Neural Information Processing Systems*, 34: 14239–14251.
- Jiang, Y.; Huang, C.; and Huang, L. 2023. Adaptive graph contrastive learning for recommendation. In *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining*, 4252–4261.
- Jiang, Y.; Jin, S.; Jin, X.; Xiao, X.; Wu, W.; Liu, X.; Zhang, Q.; Zeng, X.; Yang, G.; and Niu, Z. 2023. Pharmacophoric-constrained heterogeneous graph transformer model for molecular property prediction. *Communications Chemistry*, 6(1): 60.
- Karhadkar, K.; Banerjee, P.; and Montufar, G. 2023. FoSR: First-order spectral rewiring for addressing oversquashing in GNNs. In *International Conference on Learning Representations*.
- Kipf, T. N.; and Welling, M. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *International Conference on Learning Representations*.
- Lei, R.; Wang, Z.; Li, Y.; Ding, B.; and Wei, Z. 2022. EvenNet: Ignoring Odd-Hop Neighbors Improves Robustness of Graph Neural Networks. In *NeurIPS*.
- Li, J.; Shomer, H.; Mao, H.; Zeng, S.; Ma, Y.; Shah, N.; Tang, J.; and Yin, D. 2024. Evaluating graph neural networks for link prediction: Current pitfalls and new benchmarking. *Advances in Neural Information Processing Systems*, 36.
- Lin, L.; Chen, J.; and Wang, H. 2023. Spectral Augmentation for Self-Supervised Learning on Graphs. In *International Conference on Learning Representations*.
- Liu, N.; Wang, X.; Bo, D.; Shi, C.; and Pei, J. 2022. Revisiting Graph Contrastive Learning from the Perspective of Graph Spectrum. In *NeurIPS*.
- Liu, R.; Yin, R.; Liu, Y.; and Wang, W. 2024. ASWT-SGNN: Adaptive Spectral Wavelet Transform-based Self-Supervised Graph Neural Network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 13990–13998.
- Liu, Y.; Zheng, Y.; Zhang, D.; Lee, V.; and Pan, S. 2023. Beyond Smoothing: Unsupervised Graph Representation Learning with Edge Heterophily Discriminating. In *AAAI*.
- Lu, Y.; Gao, M.; Liu, H.; Liu, Z.; Yu, W.; Li, X.; and Jiao, P. 2023. Neighborhood overlap-aware heterogeneous hypergraph neural network for link prediction. *Pattern Recognition*, 144: 109818.
- Ma, Y.; Liu, X.; Shah, N.; and Tang, J. 2022. Is Homophily a Necessity for Graph Neural Networks? In *International Conference on Learning Representations*.
- Nt, H.; and Maehara, T. 2019. Revisiting graph neural networks: All we have is low-pass filters. *arXiv preprint arXiv:1905.09550*.
- Pei, H.; Wei, B.; Chang, K. C.-C.; Lei, Y.; and Yang, B. 2020. Geom-GCN: Geometric Graph Convolutional Networks. In *International Conference on Learning Representations*.
- Platonov, O.; Kuznedelev, D.; Diskin, M.; Babenko, A.; and Prokhorenkova, L. 2023. A critical look at evaluation of GNNs under heterophily: Are we really making progress? In *The Eleventh International Conference on Learning Representations*.
- Rozemberczki, B.; Allen, C.; and Sarkar, R. 2021. Multi-scale attributed node embedding. *Journal of Complex Networks*, 9(2): cnab014.
- Ruan, X.; Jiang, C.; Lin, P.; Lin, Y.; Liu, J.; Huang, S.; and Liu, X. 2023. MSGCL: inferring miRNA–disease associations based on multi-view self-supervised graph structure contrastive learning. *Briefings in Bioinformatics*, 24(2): bbac623.

- Thakoor, S.; Tallec, C.; Azar, M. G.; Azabou, M.; Dyer, E. L.; Munos, R.; Veličković, P.; and Valko, M. 2022. Large-Scale Representation Learning on Graphs via Bootstrapping. In *International Conference on Learning Representations*.
- Veličković, P.; Fedus, W.; Hamilton, W. L.; Liò, P.; Bengio, Y.; and Hjelm, R. D. 2019. Deep Graph Infomax. In *International Conference on Learning Representations*.
- Wan, G.; Tian, Y.; Huang, W.; Chawla, N. V.; and Ye, M. 2024. S3GCL: Spectral, Swift, Spatial Graph Contrastive Learning. In *Forty-first International Conference on Machine Learning*.
- Wang, B.; Zhang, Y.; Wang, X.; Wang, P.; Zhou, Z.; Bai, L.; and Wang, Y. 2023. Pattern expansion and consolidation on evolving graphs for continual traffic prediction. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2223–2232.
- Wang, X.; and Zhang, M. 2022. How powerful are spectral graph neural networks. In *International conference on machine learning*, 23341–23362. PMLR.
- Xiao, T.; Zhu, H.; Chen, Z.; and Wang, S. 2023. Simple and Asymmetric Graph Contrastive Learning without Augmentations. In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Xu, B.; Shen, H.; Cao, Q.; Cen, K.; and Cheng, X. 2019. Graph convolutional networks using heat kernel for semi-supervised learning. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence, IJCAI 19*, 1928–1934. AAAI Press. ISBN 9780999241141.
- Xu, H.; Yan, Y.; Wang, D.; Xu, Z.; Zeng, Z.; Abdelzaher, T. F.; Han, J.; and Tong, H. 2024. SLOG: An Inductive Spectral Graph Neural Network Beyond Polynomial Filter. In *Forty-first International Conference on Machine Learning*.
- Yang, Y.; Huang, C.; Xia, L.; and Huang, C. 2023. Knowledge Graph Self-Supervised Rationalization for Recommendation. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD '23*, 3046–3056. New York, NY, USA: Association for Computing Machinery. ISBN 9798400701030.
- Yang, Z.; Cohen, W.; and Salakhudinov, R. 2016. Revisiting semi-supervised learning with graph embeddings. In *International conference on machine learning*, 40–48. PMLR.
- You, Y.; Chen, T.; Shen, Y.; and Wang, Z. 2021. Graph contrastive learning automated. In *International Conference on Machine Learning*, 12121–12132. PMLR.
- You, Y.; Chen, T.; Sui, Y.; Chen, T.; Wang, Z.; and Shen, Y. 2020. Graph Contrastive Learning with Augmentations. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M. F.; and Lin, H., eds., *Advances in Neural Information Processing Systems*, volume 33, 5812–5823. Curran Associates, Inc.
- Zhang, Y.; Zhu, H.; Song, Z.; Koniusz, P.; and King, I. 2023. Spectral feature augmentation for graph contrastive learning and beyond. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 11289–11297.
- Zhu, J.; Yan, Y.; Zhao, L.; Heimann, M.; Akoglu, L.; and Koutra, D. 2020a. Beyond homophily in graph neural networks: Current limitations and effective designs. *Advances in neural information processing systems*, 33: 7793–7804.
- Zhu, Y.; Xu, Y.; Yu, F.; Liu, Q.; Wu, S.; and Wang, L. 2020b. Deep Graph Contrastive Representation Learning. In *ICML Workshop on Graph Representation Learning and Beyond*.