

AS-GCL: Asymmetric Spectral Augmentation on Graph Contrastive Learning

Ruyue Liu, Rong Yin*, Yong Liu, Xiaoshuai Hao, Haichao Shi, Can Ma, and Weiping Wang

Abstract—Graph Contrastive Learning (GCL) has emerged as the foremost approach for self-supervised learning on graph-structured data. GCL reduces reliance on labeled data by learning robust representations from various augmented views. However, existing GCL methods typically depend on consistent stochastic augmentations, which overlook their impact on the intrinsic structure of the spectral domain, thereby limiting the model’s ability to generalize effectively. To address these limitations, we propose a novel paradigm called AS-GCL that incorporates asymmetric spectral augmentation for graph contrastive learning. A typical GCL framework consists of three key components: graph data augmentation, view encoding, and contrastive loss. Our method introduces significant enhancements to each of these components. Specifically, for data augmentation, we apply spectral-based augmentation to minimize spectral variations, strengthen structural invariance, and reduce noise. With respect to encoding, we employ parameter-sharing encoders with distinct diffusion operators to generate diverse, noise-resistant graph views. For contrastive loss, we introduce an upper-bound loss function that promotes generalization by maintaining a balanced distribution of intra- and inter-class distance. To our knowledge, we are the first to encode augmentation views of the spectral domain using asymmetric encoders. Extensive experiments on eight benchmark datasets across various node-level tasks demonstrate the advantages of the proposed method.

Index Terms—Self-Supervised, Graph Contrastive Learning, GNN, Spectral Augmentation.

I. INTRODUCTION

GRAPH data play a significant role in multimedia applications, enabling cross-modal content analysis and generation. Graph neural networks (GNNs) have been widely adopted across diverse domains such as computer vision (CV) [1], natural language processing (NLP) [2], and information retrieval (IR) [3]. Notably, Graph Convolutional Networks (GCNs) [4] and Graph Attention Networks (GATs) [5] are specifically tailored to semi-supervised node classification tasks involving labeled nodes. Labeled data are particularly crucial for enhancing the performance of these GNN models.

Contrastive learning (CL) learns representations from supervision signals derived from data without relying on human-provided labels. CL has achieved great success in many

Ruyue Liu, Rong Yin, Haichao Shi, Can Ma, and Weiping Wang are with the Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100085, China. Ruyue Liu is with School of Cyberspace Security, University of Chinese Academy of Sciences, Beijing 100049, China. E-mail: {liuyue, yinrong, shihaichao, macan, wangweiping}@iie.ac.cn.

Yong Liu is with Renmin University of China, Beijing 100872, China. E-mail: liuyonggsai@ruc.edu.cn.

Xiaoshuai Hao is with Beijing Academy of Artificial Intelligence, Beijing 100028, China. E-mail: xshao@baai.ac.cn.

Rong Yin is the corresponding author.

domains, including computer vision [6], natural language processing [7], and signal processing [8]. Specifically, contrastive methods aim to build effective representations by joining semantically similar (positive) pairs and separating dissimilar (negative) pairs [9], [10]. Inspired by the success of contrastive methods in computer vision, these methods have recently been adopted for graphs [8], [11], [12].

While graph contrastive learning methods are effective on graph-related tasks, they often overlook a fundamental distinction between images and graphs. In the context of images, augmentation is well-defined and typically involves operations like cropping and rotation. However, in graphs, the representation of augmentation can become arbitrary. For example, the core semantics in images remain relatively stable even after they undergo random transformations [13]. In contrast, when the edges or nodes of a graph are perturbed, it is difficult to determine whether the augmented graph maintains a positive correlation with the original graph. This is mainly because graphs contain both semantic and structural information.

Due to the arbitrary behavior of augmentation in graphs, the quality of learned graph representations in augmentation-based contrastive methods highly depends on selecting an appropriate augmentation scheme [14]. Graph augmentation methods can be categorized into four main groups: subgraph extraction [15], [16], adaptive augmentation [17], [18], random augmentation [9], [19], [20], and other methods [11], [21]. Subgraph extraction involves partitioning each graph into multiple subgraphs, with subgraph selection determined by the specific requirements of different models. However, this category of methods incurs significant computational overhead that is often considered prohibitive. Adaptive augmentation leverages principles rooted in task labeling and information bottlenecks to identify optimal views to enhance learning, although these methods usually entail intricate model architectures. In random augmentation, contrastive views are constructed by applying random perturbations, such as discarding nodes or edges. While straightforward and readily implementable, random augmentation overlooks the nuanced impact of distinct edge perturbations on underlying graph structure, thus failing to benefit from the relational inductive bias of graph-structured data.

To address the severe structural damage caused by consistent random augmentation processes, we propose AS-GCL, a contrastive learning method utilizing spectral augmentation. To our knowledge, AS-GCL represents a pioneering effort in encoding spectral augmented views through an asymmetric encoder. The proposed method consists of three main components. The first component employs minimal spectral variation

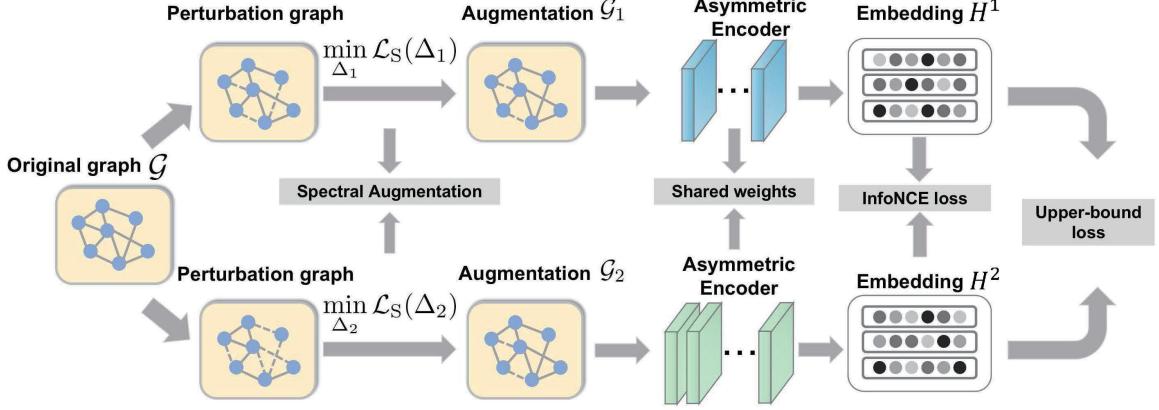


Fig. 1: The general framework of the proposed AS-GCL method. First, topology augmentation is optimized by generating paired augmented views \mathcal{G}_1 and \mathcal{G}_2 through spectral variation minimization. Then, encoders with shared parameters but different diffusion operators are used to generate representations for different views (\mathbf{H}^1 , and \mathbf{H}^2). Finally, a contrastive loss function is introduced to reduce intraclass variation and increase interclass contrast.

to guide topology augmentation, resulting in augmented views with minimal structural differences. In the second phase, we improve the GCN encoder by using an asymmetric encoder to generate different view representations. The final component achieves model optimization by maximizing the mutual information of the various view representations. We introduce a multiple loss function based on upper-bound loss, which prevents positive embeddings from deviating significantly from anchor embeddings, thereby minimizing intraclass differences. The main contributions of this paper are as follows.

- We propose the AS-GCL framework, employing a novel asymmetric spectral augmentation strategy that minimizes spectral variations and generates diverse, noise-resilient graph views through parameter-sharing encoders with different diffusion operators.
- This method introduces spectral-based augmentation and an upper-bound loss function, ensuring improved structural invariance, reduced noise, and improved generalizability by maintaining balanced intraclass and interclass distance.
- Extensive experiments on node classification and clustering tasks demonstrate that AS-GCL consistently outperforms state-of-the-art methods and shows strong robustness against adversarial attacks on graph structures.

The paper is organized as follows: Section II provides an overview of related work on graph contrastive learning. Section III introduces the notation and preliminaries used in the paper. Section IV presents the details of the proposed method. Section V describes a comprehensive set of experiments conducted to evaluate the effectiveness of the proposed method. Finally, Section VI summarizes the contributions of the paper and describes future research directions.

II. RELATED WORK

A. Graph Contrastive Learning

Recently, due to the significant success of contrastive learning methods for images, CL techniques have increasingly been

applied to graphs. DGI [22] is a notable pioneering work inspired by Deep InfoMax [23]. The objective of DGI is to acquire node representations by maximizing the mutual information of local graph blocks. GMI [24] and HDMI [25] further enhanced DGI by incorporating mutual information about edges and node attributes. Inspired by SimCLR [26], GRACE [27], which generates two augmented graph views by randomly perturbing nodes and edges, learns node representations by aligning representations of the same nodes across two augmented graphs while maintaining separation of the other nodes. Despite these advancements, a major criticism of graph contrastive learning (GCL) methods is so-called semantic drift, where the augmentation process can lead to distorted or inconsistent node representations, thereby reducing the quality of learned embeddings [28]. Additionally, most existing GCL frameworks use two view encoders with identical architectures, which may limit their ability to learn diverse features from augmented graph views [28]. In this study, we propose a novel paradigm for GCL. Unlike existing methods, the proposed AS-GCL method employs asymmetric view encoders with identical parameters but a different number of propagation layers. This design enables the encoders to capture information from both long-range and short-range connections in the graph, resulting in richer and more diverse representations. Structural asymmetry helps enhance the quality of the learned representations and opens new avenues for advancing contrastive learning on graph data.

B. Spectrum-based Augmentation

Most existing augmentation methods focus on the spatial domain, with relatively few studies exploring the spectral domain. Spectral-based methods can be broadly categorized as either utilizing the graph Laplacian spectrum or the feature spectrum. Methods leveraging the graph Laplacian spectrum focus primarily on capturing global graph structure. For example, SpCo [29] aims to capture the low-frequency information common to both augmentations by adjusting the

balance of high- and low-frequency components. Similarly, Amur Ghose et al. [30] proposed a method that enhances representations by rearranging various frequency components within graphs. GCIL [31] adopts a causal perspective by considering low-frequency components as causal variables and high-frequency components as noncausal variables. GCIL achieves graph augmentation by perturbing noncausal information and retaining causal information. Feature spectrum-based methods emphasize the local features of nodes or edges. For example, COSTA [11] decomposes hidden space representations and optimizes the alignment process by adding uniform noise (perturbation) to singular values. SFA [32] balances the contribution of each component within the feature spectrum to achieve an optimal representation. These methods typically rely on spectral filters to balance low- and high-frequency components, which requires careful tuning based on specific datasets and tasks. In contrast, our method focuses on minimizing Laplacian spectral variations to effectively mitigate the semantic drift caused by graph data augmentation. Unlike previous methods, our method does not require fine-tuning of frequency components for each dataset; rather, it automatically learns the structure with minimal variance. By reducing Laplacian spectral variation, our method ensures a more consistent global representation, leading to improved generalizability and robustness across different tasks.

III. PRELIMINARIES

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ represent an undirected graph, where $\mathcal{V} = \{v_1, \dots, v_N\}$ denotes the set of N nodes, and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ denotes the set of edges. The existence of edges is represented by an adjacency matrix $\mathbf{A} \in \{0, 1\}^{N \times N}$, where $A_{ij} \in \{0, 1\}$ signifies the relationship between nodes v_i and v_j in \mathcal{V} . Additionally, graph \mathcal{G} is associated with a node feature matrix $\mathbf{X} = [x_1, x_2, \dots, x_N] \in \mathbb{R}^{N \times d}$, where x_i represents a d -dimensional feature vector of node $v_i \in \mathcal{V}$. The node degree matrix is denoted as $\mathbf{D} = \text{diag}(d_1, \dots, d_N)$, where $d_i = \sum_{v_j \in \mathcal{V}} A_{ij}$ represents the degree of node $v_i \in \mathcal{V}$. The objective is to learn (in an unsupervised manner) a GNN encoder $f_\theta : (\mathbf{X}, \mathbf{A}) \rightarrow \mathbf{H} \in \mathbb{R}^{N \times m}$, which takes node features and graph structure as inputs and generates low-dimensional node representations, where m is significantly smaller than d .

Let $\mathbf{L} = \mathbf{D} - \mathbf{A}$ represent the unnormalized graph Laplacian of \mathcal{G} . If we denote the symmetric normalized adjacency matrix as $\hat{\mathbf{A}} = \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}}$, then the symmetric normalized graph Laplacian is $\hat{\mathbf{L}} = \mathbf{I} - \hat{\mathbf{A}} = \mathbf{D}^{-\frac{1}{2}} \mathbf{L} \mathbf{D}^{-\frac{1}{2}}$. Since $\hat{\mathbf{L}}$ is symmetric and normalized, its eigen-decomposition is $\mathbf{U} \Lambda \mathbf{U}^\top$, where $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_N)$ contains the eigenvalues of $\hat{\mathbf{L}}$, and $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_N] \in \mathbb{R}^{N \times N}$ contains the eigenvectors of $\hat{\mathbf{L}}$.

IV. PROPOSED METHOD

We propose a simple yet effective graph contrastive learning framework, AS-GCL, for generating more specific graph representations by leveraging the relational inductive bias of graph-structured data. The proposed method includes three key improvements: an augmentation scheme, an encoder, and a contrastive loss function. In AS-GCL's augmentation scheme,

the graph spectrum is utilized to capture structural properties, and the invariance of the graph spectrum is employed to represent structural invariance. With respect to the encoder, AS-GCL uses asymmetric encoders with a different number of diffusion layers to generate view embeddings. For contrastive loss, we introduce an upper-bound loss component to constrain the distance between positive and negative pairs.

A. Spectral Augmentation

The objective of our spectral augmentation scheme is to reduce dependence on unstable components and effectively mitigate the noise associated with data augmentation. Specifically, we define an edge perturbation-based topology augmentation scheme determined by a Bernoulli probability matrix. The augmentation principle is formulated as an optimization problem based on this topological augmentation scheme to minimize the spectral variation of the augmented graphs.

Topology Augmentation. To enhance graph topology diversity and improve the robustness of graph representation learning, we propose an edge perturbation-based topology augmentation scheme. This scheme introduces a Bernoulli probability matrix to control the probability of edge flipping, enabling the generation of diverse graph structures to enhance generalizability. Specifically, for a given adjacency matrix \mathbf{A} , we define topological augmentation matrix $\tilde{\mathbf{A}}$ as the result of sampling from a Bernoulli distribution $\mathcal{B}(\Delta)$, where $\Delta \in [0, 1]^{N \times N}$ is the parameter matrix that controls the probability of edge flipping. The element Δ_{ij} represents the probability of flipping the edge between nodes i and j , which can be flexibly adjusted based on prior knowledge or experimental experience.

Our choice to use a Bernoulli probability matrix to control edge perturbation is motivated by its flexibility and adaptability. Specifically, by adjusting the values in Δ_{ij} , we can control the intensity of the perturbation. This capability enables us to adjust the intensity of perturbations according to dataset requirements or specific experimental conditions. For example, using a lower perturbation probability helps maintain global structural graph integrity, ensuring that fundamental relationships between nodes are preserved. Conversely, a higher perturbation probability introduces more significant local changes, which can generate more diverse graph views. This flexibility provides balance during augmentation, enabling the model to retain the global characteristics of the original graph while introducing random perturbations to enhance robustness and generalizability across different views. Operationally, we first sample the edge perturbation matrix $\mathbf{M} \in \{0, 1\}^{N \times N}$, where $M_{ij} \sim \mathcal{B}(\Delta_{ij})$ determines whether to flip the edge between nodes i and j . If $M_{ij} = 1$, the edge is flipped; otherwise, it remains unchanged. The topological augmentation matrix is defined as follows:

$$\begin{aligned} \mathbf{C} &= \bar{\mathbf{A}} - \mathbf{A}, \\ \tilde{\mathbf{A}} &= \mathbf{A} + \mathbf{C} \circ \mathbf{M}, \end{aligned} \tag{1}$$

where $\bar{\mathbf{A}} = \mathbf{1}_N \mathbf{1}_N^\top - \mathbf{A}$ is the complement matrix of \mathbf{A} , $\mathbf{1}_N$ is an N -dimensional column vector of ones, and $\mathbf{1}_N \mathbf{1}_N^\top$ represents its outer product. To ensure effective edge perturbation,

we introduce the matrix \mathbf{C} , where C_{ij} indicates whether an edge operation is valid between nodes i and j . If $C_{ij} = 1$, edge addition is allowed, whereas $C_{ij} = -1$ allows edge deletion. This method enables the model to adapt to various structural changes during self-supervised learning, ultimately improving its robustness.

Minimization of spectral variations. We optimize Δ guided by the graph spectral domain instead of using a fixed value as employed in uniform perturbation. We aim to find the matrix Δ that minimizes the expected discrepancy in the graph spectra between the original graph and the graph augmented using the topology defined by Δ .

To find the desired perturbation matrix Δ , we formulate the following optimization problem:

$$\min \mathcal{L}_S = \min_{\Delta \in \mathcal{S}} \| \text{eig}(\text{Lap}(\mathbf{A} + \mathbf{C} \circ \Delta)) - \text{eig}(\text{Lap}(\mathbf{A})) \|_2^2, \quad (2)$$

where $\mathcal{S} = \{s \mid s \in [0, 1]^{N \times N}, \|s\|_0 \leq \epsilon \times N \times N\}$, with ϵ representing the strength of graph perturbation. $\text{Lap}(\mathbf{A})$ represents the normalized Laplacian matrix of \mathbf{A} , and $\text{eig}(\cdot)$ computes the eigenvalue vector.

We apply two perturbation ratios of varying sizes to enhance the graph topology. By solving Eq. (2), we derive optimal pairwise Bernoulli probability matrices, Δ_1 and Δ_2 , which produce two augmented views, \mathcal{G}_1 and \mathcal{G}_2 , to maintain structural invariance while minimizing deviation from the original graph. This enables the exploration and selection of a strategy closely aligned with the original graph's structural properties. Sampling the augmented views using the optimal Δ matrix effectively improves the stability and robustness of the graph representation while preserving key topological features.

To update the value of Δ^1 , we use the following equation:

$$\Delta^{(t)} = \Delta^{(t-1)} - \eta_t \nabla \mathcal{L}_S(\Delta^{(t-1)}), \quad (3)$$

where $\eta_t > 0$ represents the learning rate at step t . The gradient $\nabla \mathcal{L}_S(\Delta^{(t-1)})$ can be computed using the chain rule. For a real symmetric matrix \mathbf{L} , the derivative of its k -th eigenvalue λ_k is given by $\partial \lambda_k / \partial \mathbf{L} = \mathbf{u}_k \mathbf{u}_k^\top$, where \mathbf{u}_k is the corresponding eigenvector.

Notably, the derivative computation requires distinct eigenvalues, which may not hold for graphs that exhibit self-isomorphism. To address this limitation, we introduce a small noise term to the adjacency matrix, represented as $\varepsilon \times (\mathbf{E} + \mathbf{E}^\top)/2$, where each entry in \mathbf{E} is sampled from the uniform distribution $U(0, 1)$, and ε is a small constant. The inclusion of the term $(\mathbf{E} + \mathbf{E}^\top)/2$ ensures that the perturbed adjacency matrix maintains its symmetry, which is particularly important for undirected graphs. By averaging \mathbf{E} with its transpose, we ensure that the resulting perturbed adjacency matrix retains symmetric properties.

Remark 1. Directly capturing the structural invariance of graph contrastive learning methods is a complex challenge requiring the simultaneous consideration of various structural

attributes. Fortunately, the graph spectrum provides a comprehensive summary of several structural properties, including clustering, connectivity, and diameter [33], [34]. Empirically, relatively significant changes in the spectrum correspond to edge flips between nodes from different clusters, indicating more substantial structural alterations. Therefore, the perturbation spectrum can be used to control changes in these structural properties. By leveraging the graph spectrum to capture structural properties, our goal is to maintain spectral invariance as a proxy for structural invariance. Through graph spectral-guided optimization, we aim to enhance the spectral invariance of augmented graph representations, thereby improving both the accuracy and robustness of the models.

B. Asymmetric Encoders

In contrastive learning (CL), the learned representation includes both task-relevant information and task-irrelevant noise. CL methods aim to extract consistent information across different views while filtering out task-irrelevant noise specific to any single view. This requires a careful balance: the two views should not be too distant, as this reduces the amount of task-relevant information, nor should they be too close, as this can introduce excessive task-irrelevant noise. In the proposed spectral augmentation method (Section IV-A), we maintain the structural invariance of graph augmentations, thereby enhancing the consistency between views. However, this method may also increase the proximity between views, potentially leading to reduced differentiation. To address this issue, we employ asymmetric encoders with shared weight parameters but a different number of diffusion layers. This design helps manage the trade-off between view consistency and proximity, improving the overall effectiveness of representation learning.

We decompose each layer of the graph convolutional network (GCN) into two distinct operators, a diffusion operator and a transformation operator, denoted as f and g , respectively. These operators are defined as follows:

$$\begin{aligned} f(\mathbf{H}) &= \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} \mathbf{H}, \\ g(\mathbf{H}) &= \sigma(\mathbf{H} \mathbf{W}), \end{aligned} \quad (4)$$

where \mathbf{H} represents the node embeddings, $\mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}}$ is the graph filter matrix that performs diffusion, \mathbf{W} is the weight matrix for the linear transformation, and σ denotes a nonlinear activation function. This decomposition allows us to first capture the structural information of the graph through the diffusion operator and then transform the features with a linear transformation followed by a nonlinear activation function.

We express the l -layer GCN as a combination of multiple diffusion and transformation operators. Specifically, the l -layer GCN can be written as:

$$GCN(\mathbf{X}) = g^{(l)} \circ f \circ g^{(l-1)} \circ f \circ \dots \circ g^{(1)} \circ f(\mathbf{X}), \quad (5)$$

where \circ denotes the composition of two operators and $g^{(l)}$ represents the transformation operator applied in the l -th layer of the GCN. This expression shows that the GCN is formed by applying the diffusion operator f followed by the transformation operators $g^{(i)}$ for each layer iteratively.

¹Given that Δ_1 and Δ_2 are updated in an identical manner, we simplify the notation by omitting the subscripts and adopting Δ for consistency.

The proposed asymmetric encoder shares weight parameters but employs a different number of diffusion layers. This can be mathematically formalized as follows:

$$\begin{aligned}\mathbf{H}^1 &= g^{(i)} \circ f \circ \dots \circ g^{(1)} \circ f(\mathbf{X}), \\ \mathbf{H}^2 &= f^{[k]} \circ g^{(i)} \circ f \circ \dots \circ g^{(1)} \circ f(\mathbf{X}),\end{aligned}\quad (6)$$

where \mathbf{H}^1 and \mathbf{H}^2 represent the node embeddings generated by two different encoders. Here, $f^{[k]}$ denotes the composition of k diffusion operators f . Both encoders share i transformation layers, which include weight parameters and activation functions. However, the two encoders differ in the number of diffusion layers applied: the first encoder uses i diffusion layers, whereas the second encoder uses $i+k$ diffusion layers. These diffusion layers propagate the hidden representations from one layer to the next via the adjacency matrix. Note that the exact number of diffusion layers may vary depending on the design choices and specific requirements of the asymmetric encoder.

Remark 2. *By employing a varying number of diffusion layers, the two views generated by the asymmetric encoder are effectively pushed away from each other while ensuring that they are not overly distant. This design maintains balance between the views, preventing them from becoming too dissimilar. Moreover, the weight parameters shared between the encoders ensure that the views remain within an appropriate proximity range. This method enhances view diversity while minimizing noise in the graph convolutional network, thereby improving the robustness and accuracy of graph representation learning.*

C. Optimization Objective

InfoNCE loss. We employ a contrastive objective to evaluate node representations obtained from two different graph augmentations. For a given node v_a , the representations \mathbf{H}_a^1 and \mathbf{H}_a^2 from the respective graph augmentations form a positive pair. Conversely, the representations of other nodes in the two augmentations are considered negative pairs. The paired objective for each positive pair $(\mathbf{H}_a^1, \mathbf{H}_a^2)$ is defined as follows:

$$\begin{aligned}\mathcal{L}(\mathbf{H}_a^1, \mathbf{H}_a^2) = & \frac{e^{\theta(\mathbf{H}_a^1, \mathbf{H}_a^2)}}{e^{\theta(\mathbf{H}_a^1, \mathbf{H}_a^2)} + \sum_{b \neq a} (e^{\theta(\mathbf{H}_a^1, \mathbf{H}_b^2)} + e^{\theta(\mathbf{H}_a^1, \mathbf{H}_b^1)})},\end{aligned}\quad (7)$$

where $\theta(\cdot)$ denotes the cosine similarity function. The term $e^{\theta(\mathbf{H}_a^1, \mathbf{H}_a^2)}$ in the numerator represents the similarity between the same node a in two different views, essentially serving as a measure of similarity between positive samples. In the denominator, $e^{\theta(\mathbf{H}_a^1, \mathbf{H}_a^2)}$ represents the similarity for the positive pair, whereas the second term sums over all the negative sample pairs. Specifically, $e^{\theta(\mathbf{H}_a^1, \mathbf{H}_b^2)}$ represents negative pairs across views, i.e., the similarity between node a in the first view and node b in the second view. In addition, $e^{\theta(\mathbf{H}_a^1, \mathbf{H}_b^1)}$ indicates the similarity between nodes a and b within the same view. This formulation ensures that the negative sample comparisons encompass a broad range of potential combinations, promoting robust contrastive learning.

The overall objective is to maximize the average similarity of all positive pairs, which is expressed as:

$$\mathcal{L}_{InfoNCE} = \frac{1}{2N} \sum_{a=1}^N [\mathcal{L}(\mathbf{H}_a^1, \mathbf{H}_a^2) + \mathcal{L}(\mathbf{H}_a^2, \mathbf{H}_a^1)]. \quad (8)$$

Upper-bound loss. We propose an upper-bound loss function based on triplet state loss to address intraclass variation and enhance interclass variation. Specifically, the triplet loss for each positive pair $(\mathbf{H}_a^1, \mathbf{H}_a^2)$ is formulated as:

$$\mathcal{L}_{lower} = \sum_{a=1}^N \max(0, \|\mathbf{H}_a^1 - \mathbf{H}_a^2\|_2 - \|\mathbf{H}_a^1 - \mathbf{H}_a^-\|_2 + \alpha), \quad (9)$$

where \mathbf{H}_a^- denotes the negative pair of \mathbf{H}_a^1 and α is a hyperparameter that controls the minimum expected distance between the positive and negative pairs.

While this lower-bound loss \mathcal{L}_{lower} effectively separates negative pairs, it can inadvertently increase the distance between positive pairs. To address this issue, we introduce an upper-bound loss to constrain the distance between negative and positive pairs. This constraint helps reduce intraclass variation and maintain the proximity of positive samples. The upper-bound loss is defined as:

$$\mathcal{L}_{upper} = \sum_{a=1}^N \max(0, \|\mathbf{H}_a^1 - \mathbf{H}_a^-\|_2 - \|\mathbf{H}_a^1 - \mathbf{H}_a^2\|_2 - \beta), \quad (10)$$

where β is a hyperparameter that defines the acceptable range of distances between positive and negative pairs.

Combining InfoNCE, lower-bound, and upper-bound loss, the overall loss function for the proposed method AS-GCL is expressed as:

$$\mathcal{L} = \mathcal{L}_{InfoNCE} + \mathcal{L}_{lower} + \mathcal{L}_{upper}. \quad (11)$$

This comprehensive loss function ensures effective contrastive learning by balancing the distance between positive and negative pairs, thus enhancing the model's ability to generate robust and accurate graph representations.

Remark 3. *InfoNCE loss aims to minimize the distance between positive pairs and maximize the distance between negative pairs, which fosters the learning of effective graph representations. However, it does not directly address the issue of the large intraclass distance that can arise if the distance between an anchor and its positive embedding becomes too large. When this happens, the distance between the anchor and negative embeddings can become excessively large, potentially leading to infinite distances. Such large intraclass variation can negatively impact the generalization performance of the model. To address this limitation, AS-GCL incorporates both upper-bound and lower-bound loss. Upper bound loss ensures that the distance between the positive and anchor embeddings remains finite, effectively controlling intraclass variation. This constraint prevents the distance between positive pairs from becoming excessively large, maintaining a reasonable range for the distances between negative and positive pairs. By imposing this balance, the upper-bound loss helps to avoid extreme distances that could impair model performance. In*

TABLE I: Dataset statistics for node classification and clustering.

Datasets	Category	Node	Edge	Feature	Class
Cora [35]	Citation Network	2708	5429	1433	7
CiteSeer [35]	Citation Network	3327	4732	3703	6
PubMed [35]	Citation Network	19717	44338	500	3
Computers [36]	Purchase	13752	245861	767	10
Photo [36]	Purchase	7650	119081	745	8
CS [36]	Citation Network	18333	81894	6805	15
Physics [36]	Citation Network	34493	247962	8415	5
WikiCS [37]	Citation Network	11701	216123	300	10

summary, while InfoNCE loss focuses on differentiating between positive and negative pairs, the upper-bound loss in AS-GCL ensures that positive embeddings remain reasonably close to the anchor. This combined method promotes more stable and discriminative representations, enhancing the model's ability to generalize and improving overall performance.

V. EXPERIMENTAL EVALUATION

A. Experimental Setting

Datasets. We evaluate the performance of the proposed AS-GCL method by comparing it to that of existing state-of-the-art methods across eight real datasets: Cora, CiteSeer, PubMed, Amazon-Photo, Amazon-Computers, Coauthor-CS, Coauthor-Physics, and WikiCS. The characteristics of these datasets are summarized in TABLE I. Cora, CiteSeer, and PubMed are well-known citation networks where nodes represent documents and edges denote citation links between documents. Amazon-Photo (Photo) and Amazon-Computers (Computers) are subsets of the Amazon co-purchasing graph. In these datasets, nodes represent products, and edges indicate frequently co-purchased items. Coauthor-CS (CS) and Coauthor-Physics (Physics) are co-authorship graphs derived from academic papers in computer science and physics, respectively, wherein the nodes represent authors, and the edges signify collaborative relationships. WikiCS is a dataset derived from Wikipedia articles, with the nodes representing articles and the edges reflecting hyperlinks between articles. These datasets encompass a diverse range of graph-based structures, providing a broad testing ground for evaluating the effectiveness of our method.

Baselines. We examine a range of baseline methods for node-level tasks. These include random walk-based methods such as DeepWalk [38] and Node2Vec [39], as well as supervised learning techniques like MLP, GCN [4], and GAT [5]. We also investigate several self-supervised learning baseline methods categorized into three groups based on their methodology: random consistency perturbations (e.g., DGI [22], GMI [24], MVGRL [19], GRACE [40], and GraphCL [41]), generative methods (e.g., VGAE [42] and AdaGCL [43]), and learnable techniques such as GCA-SSG [44], GRADE [9], NCLA [45], GCIL [31], and LSGCL [8]. For node clustering tasks, we performed experiments using a variety of clustering and graph self-supervised learning methods. The clustering methods include K -means, spectral clustering, GAE [42], VGAE [42], DGI [22], DNGR [46], TADW [47], and GC-VAE

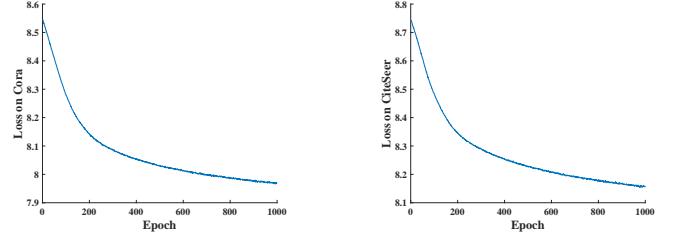


Fig. 2: Training loss for different numbers of rounds.

[48], and the self-supervised learning methods are NCLA [45] and GCIL [31].

Evaluation protocol. To assess the performance of AS-GCL, we utilize a two-layer GCN model where each layer has a hidden dimension of 256. A linear classifier is employed at the postprocessing stage for evaluation. For spectral augmentation, we set the number of training rounds to five and fix the edge perturbation rate ϵ at 0.2. During contrastive learning, we run 1000 training epochs with a batch size of 128. The diffusion layers are set as $i = 2$ and $k = 2$. The hyperparameters α and β are tuned for each dataset, with α ranging from 4-6 and β ranging from 8-10. For the downstream classification task, we adopt the cross-entropy loss function. The dataset is split as follows: 10% for training, 10% for validation, and 80% for testing. We use the Adam optimizer with a learning rate of 0.001 and a weight decay of 5×10^{-5} for optimization. To ensure unbiased results, each dataset is run five times with different random seeds, and we report the average accuracy and standard deviation. All the experiments are performed via PyTorch on a server equipped with eight NVIDIA GeForce 3090 GPUs, each with 24 GB of memory.

B. Experimental Results

Node Classification. TABLE II presents the node classification accuracy results, where the best-performing method in each column is highlighted in boldface font, and the second-best method is underlined. Fig. 2 depicts the training loss variation over the number of training rounds on the Cora and CiteSeer datasets. Notably, AS-GCL achieves state-of-the-art (SOTA) performance across seventeen graph benchmarks. Compared with other self-supervised approaches applied to the eight datasets, AS-GCL outperforms the strongest baseline, LSGCL, by an average of 1.7%, and surpasses the weakest baseline, VGAE, by 4.7%. These results demonstrate the effectiveness of AS-GCL on node classification tasks, further

TABLE II: Accuracy (%) on the eight benchmark datasets for the node classification task. The best result is represented by boldface font, and the second best result is underlined.

Method	Datasets								
	Cora	CiteSeer	PubMed	Computers	Photo	CS	Physics	WikiCS	Avg.Acc.
Deep Walk	69.1±0.3	43.5±0.2	65.1±0.4	85.7±0.1	89.5±0.1	84.2±0.4	91.7±0.2	74.4±0.2	75.4
Node2cec	71.0±0.6	47.3±0.5	66.3±0.8	84.6±0.2	89.7±0.2	85.2±0.1	91.2±0.1	71.8±0.1	75.9
MLP	49.9±0.4	49.3±0.3	69.1±0.3	73.9±0.2	79.5±0.3	90.3±0.2	93.5±0.4	72.0±0.4	72.2
GCN	81.6±0.2	70.3±0.4	79.3±0.2	84.5±0.3	91.6±0.3	<u>93.1±0.3</u>	93.7±0.2	73.0±0.1	83.4
GAT	83.1±0.3	72.4±0.3	79.5±0.1	85.8±0.1	91.7±0.2	<u>89.5±0.3</u>	93.5±0.3	72.6±0.3	83.5
DGI	82.3±0.3	71.3±0.4	79.4±0.3	84.9±0.3	91.6±0.1	92.1±0.4	93.7±0.2	75.3±0.1	83.8
GMI	83.3±0.2	72.6±0.2	79.8±0.4	82.2±0.1	90.7±0.2	92.6±0.2	94.3±0.4	74.9±0.2	83.8
VGAE	75.9±0.5	66.8±0.2	75.8±0.4	85.8±0.3	91.5±0.2	91.8±0.4	94.1±0.2	75.5±0.3	82.2
MVGRL	83.1±0.2	72.3±0.5	80.3±0.5	87.5±0.1	91.7±0.1	92.1±0.3	<u>95.1±0.2</u>	77.5±0.1	84.9
GRACE	81.9±0.6	71.4±0.4	80.5±0.4	86.3±0.5	92.1±0.3	84.2±0.4	92.9±0.2	78.1±0.1	83.4
GraphCL	82.1±0.3	72.8±0.2	81.7±0.3	85.6±0.2	92.5±0.1	92.6±0.4	93.7±0.2	76.8±0.3	84.7
GCA-SSG	83.9±0.4	<u>73.1±0.3</u>	81.3±0.4	<u>88.4±0.3</u>	89.5±0.1	92.4±0.1	93.4±0.2	78.2±0.3	85.0
GRADE	84.0±0.3	72.4±0.4	82.7±0.3	84.7±0.1	<u>92.6±0.1</u>	92.7±0.4	93.7±0.2	78.1±0.2	85.1
NCLA	82.2±1.6	71.7±0.9	82.0±1.4	83.7±1.1	90.2±1.3	91.5±0.7	92.2±0.9	78.6±0.8	84.0
AdaGCL	83.8±1.2	70.5±0.6	82.6±0.5	84.3±1.0	91.2±0.8	90.4±0.6	92.6±0.7	77.4±0.5	84.1
GCIL	84.2±0.5	69.1±0.4	81.6±0.7	83.4±0.6	90.5±0.4	91.8±0.9	92.8±0.4	76.7±0.6	83.8
LSGCL	84.9±0.3	72.0±0.4	84.5±0.3	85.6±0.1	91.5±0.2	91.8±0.2	92.9±0.2	<u>79.1±0.3</u>	85.2
Ours	85.2±0.3	73.9±0.1	84.9±0.3	89.9±0.3	93.5±0.2	93.5±0.3	95.2±0.2	79.5±0.1	86.9

TABLE III: ACC, NMI, F-score and ARI on three benchmark datasets for the node clustering task. The best result is in boldface font, and the second best result is underlined.

Datasets	Methods											
	Metirc	K-Means	Spectral	GAE	VGAE	DGI	DNGR	TADW	GC-VAE	NCLA	GCIL	Ours
Cora	ACC	0.493	0.396	0.597	0.592	0.590	0.419	0.562	<u>0.707</u>	0.694	0.688	0.728
	NMI	0.311	0.289	0.392	0.408	0.386	0.317	0.441	<u>0.537</u>	0.543	<u>0.552</u>	0.584
	F-score	0.376	0.332	0.415	0.456	0.432	0.389	0.418	0.695	0.485	0.521	0.683
	ARI	0.230	0.176	0.294	0.342	0.336	0.142	0.332	<u>0.482</u>	0.425	0.414	0.491
CiteSeer	ACC	0.541	0.318	0.413	0.603	0.577	0.326	0.455	<u>0.663</u>	0.485	0.623	0.671
	NMI	0.315	0.087	0.174	0.343	0.319	0.182	0.291	<u>0.407</u>	0.394	0.409	0.417
	F-score	0.413	0.256	0.297	0.462	0.452	0.403	0.415	<u>0.632</u>	0.505	0.548	0.651
	ARI	0.279	0.084	0.143	0.343	0.289	0.047	0.354	<u>0.415</u>	0.402	0.394	0.417
PubMed	ACC	0.562	0.498	0.608	0.619	0.499	0.468	0.355	<u>0.682</u>	0.532	0.487	0.687
	NMI	0.262	0.142	0.235	0.213	0.151	0.155	0.103	<u>0.294</u>	0.208	0.253	0.296
	F-score	0.559	0.473	0.497	0.478	0.432	0.452	0.426	<u>0.663</u>	0.452	0.424	0.669
	ARI	0.227	0.098	0.223	0.211	0.145	0.054	0.158	<u>0.273</u>	0.252	<u>0.277</u>	0.297

highlighting its ability to handle diverse datasets and graph structures with exceptional accuracy.

Node Clustering. Additionally, we evaluate the impact of the proposed method on node clustering by conducting a comprehensive performance analysis on three benchmark datasets: Cora, Citeseer, and Pubmed. The results, detailed in TABLE III, underscore the exceptional performance of AS-GCL compared with that of state-of-the-art graph clustering methods. The highest results in each column are highlighted in boldface font, and the second-highest results are underlined. AS-GCL consistently delivers strong results across multiple evaluation metrics, including Accuracy (ACC), Normalized Mutual Information (NMI), F-measure (F-score), and Adjusted Rand Index (ARI). Such consistent improvements across different benchmarks and metrics further validate the robustness and effectiveness of AS-GCL on node clustering tasks, highlighting its potential for broader applications in graph-based

learning.

Analysis. The excellent performance of AS-GCL is largely attributable to several key advantages of the proposed method. First, our spectral-based augmentation strategy plays a critical role in the enhanced performance of AS-GCL. By leveraging spectral information, AS-GCL captures both global and local structural patterns in graphs, ensuring more informative and diverse contrastive views. This enables the model to preserve essential graph properties while effectively reducing noise. Second, the asymmetric encoder design enhances learning by employing parameter-sharing encoders with different diffusion operators. This approach not only increases the diversity of the learned graph representations but also mitigates overfitting. As a result, AS-GCL is highly equipped to generalize across different datasets. Third, the upper-bound loss function incorporated in AS-GCL ensures balanced intraclass and inter-class distance. By addressing the issue of excessive intraclass

TABLE IV: Comparisons of empirical runtime (in seconds) on four representative node classification datasets with varying number of nodes. In AS-GCL, ‘Overall runtime’ denotes the sum of ‘Data augmentation time’ and ‘Contrastive training time’ for comparison with other baseline methods.

Dataset	Cora	Photo	Computers	PubMed
Number of nodes	2708	7650	13752	19717
AS-GCL (Ours)	96.4 257.8 354.2	472.4 838.2 1310.6	870.3 1103.0 1973.3	1235.7 1389.2 2624.9
GraphCL VGAE AdaGCL NCLA GCIL	289.4 314.6 364.3 342.0 372.5	1028.5 1205.3 1425.1 1152.3 1422.7	1531.7 1784.4 1967.4 1684.7 2008.6	1864.2 2287.3 2753.8 2432.5 2542.2

and insufficient interclass distance, our method enhances the discriminative power of learned representations, leading to improved classification performance.

C. Empirical Running Time

The time complexity for precomputing each round of spectral augmentation is $\mathcal{O}(mnT)$, where m represents the number of edges, n is the number of nodes, and T denotes the number of augmentation rounds. Typically, T is a small constant, often set to less than 5, making the overall precomputation cost manageable. TABLE IV shows the runtime (in seconds) of the proposed method along with those of typical graph contrastive learning methods. In these experiments, we performed 1000 training iterations on datasets of varying size, conducting five rounds of iterative training for spectral augmentation for our method. The precomputational overhead associated with our spectral augmentation method is reasonable and acceptable, especially in the context of overall runtime; this is because our method requires only a single data enhancement step before initiating formal training. Additionally, our approach demonstrates outstanding classification and clustering performance with time efficiency comparable to that of generative methods (see TABLE II and TABLE III). For large-scale graphs, where time and memory complexity are critical, techniques such as Chebyshev polynomial approximation of the graph Laplacian and subgraph sampling can further reduce the computational burden. These strategies enable more efficient batch training and enhance scalability. We emphasize that our work represents a preliminary step toward effective graph augmentation, with further research needed to scale the method to even larger graphs, a key focus for future investigation.

D. Spectral Variation Verification

We can quantify the changes in Laplace operators resulting from graph augmentation. To assess the impact of graph augmentation on the components of the adjacency matrix, we denote the symmetric normalized Laplace operator of the original graph as \hat{L} and the symmetric normalized Laplace operator of the augmented graph as \tilde{L} . We use the Frobenius norm $\|\hat{L} - \tilde{L}\|_F$ as a metric to measure the distance between these operators. In addition, we examine the effects of three commonly employed augmentations on the adjacency matrix: edge addition, edge removal, and graph diffusion. The

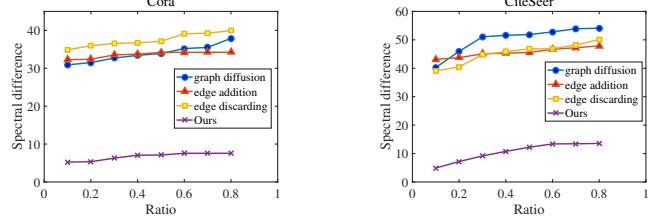


Fig. 3: Frobenius distance between the symmetric normalized Laplace matrices of the decomposition of the original and augmented graphs.

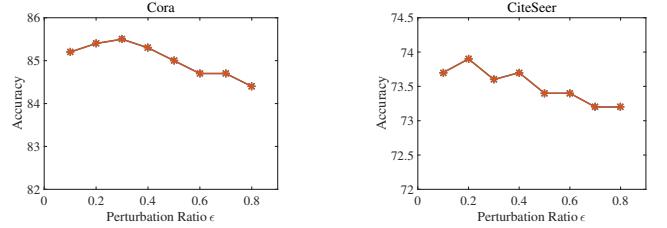


Fig. 4: Node classification accuracy with different edge perturbation ratios ϵ .

Frobenius distances between the symmetrically normalized Laplacian matrices of the augmented and original graphs are summarized in Fig. 3. Notably, significant alterations in the graph Laplace operator result from all three random topological augmentations; this highlights the limitations of using consistent random edge perturbations to preserve the structural properties of graphs. The proposed method addresses this limitation by using minimal spectral variations as a guide. As a result, the proposed method effectively captures the invariant spectral components associated with sensitive edges while reducing its reliance on unstable components, proposing a more efficient representation of the graph.

E. Ablation Studies

We conduct an ablation study to assess the individual contributions of various components within the unified learning framework of AS-GCL. The framework includes several key steps, i.e., graph data augmentation, view encoding, and contrastive loss computation. In particular, we focus on three crucial components: spectral augmentation for optimizing stochastic topology augmentation, the asymmetric encoder for enhancing view diversity and reducing noise, and the use of upper-bound and lower-bound loss to minimize intraclass distance. By systematically removing each component, we can evaluate the impact of each component on overall performance. In TABLE V, a decline in performance results when any of the components are omitted, highlighting the significance of each component in improving the model’s effectiveness. Both the optimized data augmentation approach and the mechanisms designed to enhance model robustness play a substantial role in AS-GCL’s performance, underscoring the importance of integrating all of the components to successfully address graph representation learning tasks.

TABLE V: Results of the ablation study of the proposed AS-GCL method. The best results are shown in boldface font.

Variant	Datasets							
	Cora	CiteSeer	PubMed	Computers	Photo	CS	Physics	WikiCS
AS-GCL	85.2	73.9	84.9	89.2	93.5	93.5	95.2	79.5
w/o spectral augmentation	80.1	61.3	81.6	84.5	87.7	86.4	92.0	72.2
w/o asymmetric encoders	83.1	68.5	82.6	84.9	90.9	90.1	91.2	73.3
w/o upper-bound loss	82.4	63.4	81.4	84.8	90.2	87.9	89.6	76.3
w/o lower-bound loss	84.6	69.8	82.6	83.0	90.8	89.8	90.1	72.5
w/o upper-bound and lower-bound losses	80.8	62.2	80.9	84.5	90.4	86.9	89.6	75.3

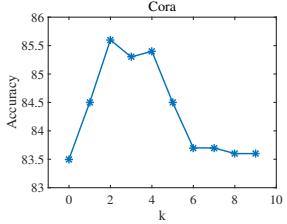
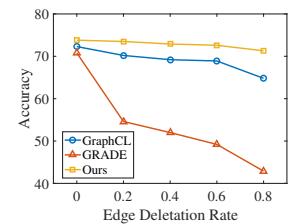
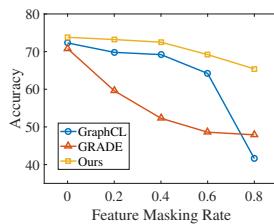
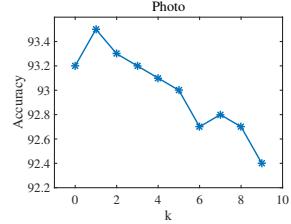
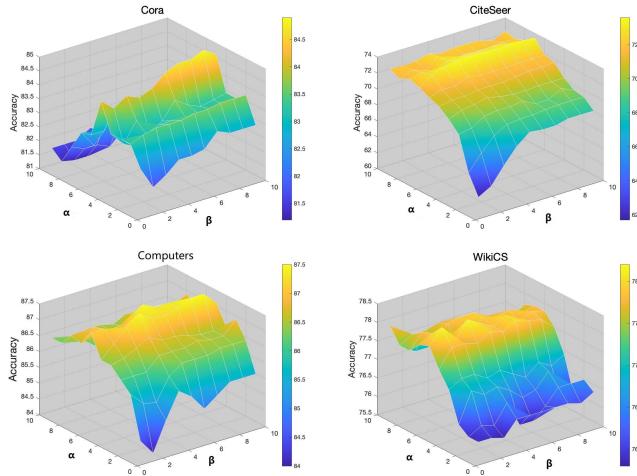
Fig. 5: Node classification accuracy with different diffusion layers k in asymmetric encoders.

Fig. 7: Test node classification accuracy when graph structures are perturbed by feature masking or edge deletion.

Fig. 6: Node classification accuracy of the proposed method under different parameter settings (α and β).

F. Parametric Experiments

Next, we evaluate the impact of four key hyperparameters on model performance: perturbation ratio ϵ , number of diffusion layers in the asymmetric encoder, and thresholds for upper and lower bound loss, denoted as α and β , respectively. As shown in Fig. 4, the model is not highly sensitive to changes in the perturbation ratio. This stability is attributed to the proposed spectral augmentation method, which effectively reduces the influence of noise introduced by perturbation, ensuring that performance remains stable even with increased perturbation levels.

Fig. 5 provides insights into the effect of the number of diffusion layers on model performance. With a relatively small number of layers, the asymmetric encoder efficiently filters out high-frequency noise and enhances the retention of important structural information. However, as the number of diffusion

layers increases, the information undergoes multiple rounds of propagation, leading to signal smoothing and the loss of key details and local features during aggregation and transmission. This over-smoothing effect ultimately reduces performance, suggesting a need to balance effective information propagation with the preservation of local structural details.

In addition, we investigate the effect of varying the values of α and β on node classification performance, as illustrated in Fig. 6. Extreme values—either too small or too large—adversely impact the performance of the model. When α is too small or β is too large, the model's ability to control the distance within and between classes weakens, leading to reduced intraclass compactness and diminished clustering effectiveness. When α is too large or β is too small, the intra-class distance increases while the interclass distance decreases, which hinders the model's ability to differentiate between distinct classes. Therefore, selecting appropriate values for α and β is essential to achieving a balanced trade-off between intraclass compactness and interclass separation, which in turn ensures effective classification. The hyperparameter evaluation presented here highlights the flexibility of the AS-GCL method under different configurations.

G. Robustness Analysis

To evaluate the robustness of the proposed AS-GCL method against adversarial graph perturbations, we test its performance on the Citeseer dataset under various attack intensities. In this evaluation, we simulate attacks by randomly altering the graph structure through feature masking and edge removal, with the ratio of modified edges and masked features varying from 0-0.8. We compare the performance of AS-GCL against two baseline methods, GraphCL and GRADE. As shown in Fig. 7, AS-GCL consistently delivers excellent or comparable performance in both attack scenarios. Notably, AS-GCL confers significant improvements with higher edge

removal rates, demonstrating its robustness in maintaining performance even amidst severe graph perturbations. This adaptability is especially valuable for practical applications where graph structures can be significantly altered or subjected to adversarial changes. The ability of our method to handle substantial structural modifications without considerable performance degradation underscores its effectiveness in robust graph representation learning.

VI. CONCLUSION

This paper proposes a novel method called Asymmetric Spectral Augmentation Graph Contrastive Learning (AS-GCL) that introduces a new paradigm for topological augmentation. The proposed method minimizes spectral variations and generates distinct view representations by using encoders with shared parameters but different diffusion operators. This strategy effectively reduces graph structure noise and produces more reliable graph views. Additionally, we introduce an upper-bound loss function to address the significant bias found between positive and anchor embeddings. Extensive experiments on various node-level tasks, including node classification and clustering, demonstrate that AS-GCL outperforms existing methods. Future work will focus on applying AS-GCL to larger datasets and more complex graph neural networks.

ACKNOWLEDGMENTS

This work is supported in part by the National Natural Science Foundation of China (No. 62106259, No. 62076234), the Beijing Outstanding Young Scientist Program (No. BJJWZYJH012019100020098), and the Beijing Natural Science Foundation (No. 4222029).

REFERENCES

- [1] D. Cai, S. Qian, Q. Fang, J. Hu, W. Ding, and C. Xu, “Heterogeneous graph contrastive learning network for personalized micro-video recommendation,” *IEEE Transactions on Multimedia*, vol. 25, pp. 2761–2773, 2023.
- [2] A. Shirian, S. Tripathi, and T. Guha, “Dynamic emotion modeling with learnable graphs and graph inception network,” *IEEE Transactions on Multimedia*, vol. 24, pp. 780–790, 2022.
- [3] H. Zhang, Y. Li, and X. Li, “Constrained bipartite graph learning for imbalanced multi-modal retrieval,” *IEEE Transactions on Multimedia*, vol. 26, pp. 4502–4514, 2024.
- [4] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” *arXiv preprint arXiv:1609.02907*, 2016.
- [5] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, “Graph attention networks,” in *6th International Conference on Learning Representations, ICLR 2018*, 2018.
- [6] M. Assefa, W. Jiang, J. Zhan, K. Gedamu, G. Yilma, M. Ayalew, and D. Adhikari, “Audio-visual contrastive and consistency learning for semi-supervised action recognition,” *IEEE Transactions on Multimedia*, vol. 26, pp. 3491–3504, 2024.
- [7] N. Rethmeier and I. Augenstein, “A primer on contrastive pretraining in language processing: Methods, lessons learned, and perspectives,” *ACM Computing Surveys*, vol. 55, no. 10, pp. 1–17, 2023.
- [8] Y. Gao, X. Li, and Y. Hui, “Rethinking graph contrastive learning: An efficient single-view approach via instance discrimination,” *IEEE Transactions on Multimedia*, vol. 26, pp. 3616–3625, 2024.
- [9] R. Wang, X. Wang, C. Shi, and L. Song, “Uncovering the structural fairness in graph contrastive learning,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 32 465–32 473, 2022.
- [10] H. Wang, J. Zhang, Q. Zhu, and W. Huang, “Augmentation-free graph contrastive learning with performance guarantee,” *arXiv preprint arXiv:2204.04874*, 2022.
- [11] Y. Zhang, H. Zhu, Z. Song, P. Koniusz, and I. King, “Costa: Covariance-preserving feature augmentation for graph contrastive learning,” in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2022, pp. 2524–2534.
- [12] R. Liu, R. Yin, Y. Liu, and W. Wang, “Unbiased and augmentation-free self-supervised graph representation learning,” *Pattern Recognition*, vol. 149, p. 110274, 2024.
- [13] X. Gong, C. Yang, and C. Shi, “Ma-gcl: Model augmentation tricks for graph contrastive learning,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023, pp. 4284–4292.
- [14] N. Lee, J. Lee, and C. Park, “Augmentation-free self-supervised learning on graphs,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 7, 2022, pp. 7372–7380.
- [15] F.-Y. Sun, J. Hoffmann, V. Verma, and J. Tang, “Infograph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization,” *arXiv preprint arXiv:1908.01000*, 2019.
- [16] W. Xia, Y. Li, and S. Li, “On the substructure countability of graph neural networks,” *IEEE Transactions on Knowledge and Data Engineering*, 2022.
- [17] M. Jing, Y. Zhu, T. Zang, J. Yu, and F. Tang, “Graph contrastive learning with adaptive augmentation for recommendation,” in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2022, pp. 590–605.
- [18] Y. Wang, Y. Cai, Y. Liang, H. Ding, C. Wang, S. Bhatia, and B. Hooi, “Adaptive data augmentation on temporal graphs,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 1440–1452, 2021.
- [19] K. Hassani and A. H. Khasahmadi, “Contrastive multi-view representation learning on graphs,” in *International Conference on Machine Learning*. PMLR, 2020, pp. 4116–4126.
- [20] R. Liu, R. Yin, Y. Liu, and W. Wang, “Aswt-sgnn: Adaptive spectral wavelet transform-based self-supervised graph neural network,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 12, 2024, pp. 13 990–13 998.
- [21] J. Xia, L. Wu, J. Chen, B. Hu, and S. Z. Li, “Simgrace: A simple framework for graph contrastive learning without data augmentation,” in *Proceedings of the ACM Web Conference 2022*, 2022, pp. 1070–1079.
- [22] P. Veličković, W. Fedus, W. L. Hamilton, P. Liò, Y. Bengio, and R. D. Hjelm, “Deep graph infomax,” *arXiv preprint arXiv:1809.10341*, 2018.
- [23] R. D. Hjelm, A. Fedorov, S. Lavoie-Marchildon, K. Grewal, P. Bachman, A. Trischler, and Y. Bengio, “Learning deep representations by mutual information estimation and maximization,” *arXiv preprint arXiv:1808.06670*, 2018.
- [24] Z. Peng, W. Huang, M. Luo, Q. Zheng, Y. Rong, T. Xu, and J. Huang, “Graph representation learning via graphical mutual information maximization,” in *Proceedings of The Web Conference 2020*, 2020, pp. 259–270.
- [25] B. Jing, C. Park, and H. Tong, “Hdmi: High-order deep multiplex infomax,” in *Proceedings of the Web Conference 2021*, 2021, pp. 2414–2424.
- [26] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, “A simple framework for contrastive learning of visual representations,” in *Proceedings of the International Conference on Machine Learning*. PMLR, 2020, pp. 1597–1607.
- [27] Y. Zhu, Y. Xu, F. Yu, Q. Liu, S. Wu, and L. Wang, “Deep graph contrastive representation learning,” *arXiv preprint arXiv:2006.04131*, 2020.
- [28] M. Sun, J. Xing, H. Wang, B. Chen, and J. Zhou, “Mocl: data-driven molecular fingerprint via knowledge-aware contrastive learning from molecular graph,” in *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, 2021, pp. 3585–3594.
- [29] N. Liu, X. Wang, D. Bo, C. Shi, and J. Pei, “Revisiting graph contrastive learning from the perspective of graph spectrum,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 2972–2983, 2022.
- [30] A. Ghose, Y. Zhang, J. Hao, and M. Coates, “Spectral augmentations for graph contrastive learning,” in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2023, pp. 11 213–11 266.
- [31] Y. Mo, X. Wang, S. Fan, and C. Shi, “Graph contrastive invariant learning from the causal perspective,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 8, 2024, pp. 8904–8912.
- [32] Y. Zhang, H. Zhu, Z. Song, P. Koniusz, and I. King, “Spectral feature augmentation for graph contrastive learning and beyond,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 9, 2023, pp. 11 289–11 297.
- [33] D. A. Spielman, “Spectral graph theory and its applications,” in *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS’07)*. IEEE, 2007, pp. 29–38.

- [34] F. Dörfler, J. W. Simpson-Porco, and F. Bullo, “Electrical networks and algebraic graph theory: Models, properties, and applications,” *Proceedings of the IEEE*, vol. 106, no. 5, pp. 977–1005, 2018.
- [35] Z. Yang, W. Cohen, and R. Salakhudinov, “Revisiting semi-supervised learning with graph embeddings,” in *International Conference on Machine Learning*. PMLR, 2016, pp. 40–48.
- [36] S. Suresh, P. Li, C. Hao, and J. Neville, “Adversarial graph augmentation to improve graph contrastive learning,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 15920–15933, 2021.
- [37] P. Mernyei and C. Cangea, “Wiki-cs: A wikipedia-based benchmark for graph neural networks,” *arXiv preprint arXiv:2007.02901*, 2020.
- [38] B. Perozzi, R. Al-Rfou, and S. Skiena, “Deepwalk: Online learning of social representations,” in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014, pp. 701–710.
- [39] A. Grover and J. Leskovec, “Node2vec: Scalable feature learning for networks,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 855–864.
- [40] Y. Zhu, W. Xu, J. Zhang, Q. Liu, S. Wu, and L. Wang, “Deep graph structure learning for robust representations: A survey,” *arXiv preprint arXiv:2103.03036*, 2021.
- [41] Y. You, T. Chen, Y. Shen, and Z. Wang, “Graph contrastive learning automated,” in *International Conference on Machine Learning*. PMLR, 2021, pp. 12121–12132.
- [42] T. N. Kipf and M. Welling, “Variational graph auto-encoders,” *stat*, vol. 1050, p. 21, 2016.
- [43] Y. Jiang, C. Huang, and L. Huang, “Adaptive graph contrastive learning for recommendation,” in *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2023, pp. 4252–4261.
- [44] H. Zhang, Q. Wu, J. Yan, D. Wipf, and P. S. Yu, “From canonical correlation analysis to self-supervised graph neural networks,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 76–89, 2021.
- [45] X. Shen, D. Sun, S. Pan, X. Zhou, and L. T. Yang, “Neighbor contrastive learning on learnable graph augmentation,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 8, 2023, pp. 9782–9791.
- [46] S. Cao, W. Lu, and Q. Xu, “Deep neural networks for learning graph representations,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2016.
- [47] C. Yang, Z. Liu, D. Zhao, M. Sun, and E. Y. Chang, “Network representation learning with rich text information,” in *Proceedings of the 24th International Conference on Artificial Intelligence*, 2015, pp. 2111–2117.
- [48] L. Guo and Q. Dai, “Graph clustering via variational graph embedding,” *Pattern Recognition*, vol. 122, p. 108334, 2022.



Ruyue Liu is a Ph.D. student at the Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China, and the School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China. Her current research interests include machine learning, data mining, self-supervised learning, federated learning, and graph representation learning.



Rong Yin received a Ph.D. from the Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China, and the School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China, in 2020. She is currently an Associate Professor with the Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China. Her current research interests include machine learning, data mining, statistical theory, distributed learning, and optimization algorithms.



Yong Liu was born in 1986. He received a Ph.D. in Computer Science from Tianjin University, Tianjin, China, in 2016. He is currently an Associate Professor with the Beijing Key Laboratory of Big Data Management and Analysis Methods, Gaoling School of Artificial Intelligence, Renmin University of China, Beijing, China. His current research interests include large-scale kernel methods, large-scale model selection, and machine learning.



Xiaoshuai Hao received the B.S. degree from Shandong University of Science and Technology China in 2017 and the Ph.D. degree from the Institute of Information Engineering, Chinese Academy of Sciences in 2023. He is currently a researcher in embodied multimodal large models at the Beijing Academy of Artificial Intelligence. His research interests include multimedia retrieval, multimodal learning and embodied intelligence.



Haichao Shi is an engineer at the Institute of Information Engineering, Chinese Academy of Sciences. He is a member of the Institute of Electrical and Electronics Engineers (IEEE), the International Computer Society (ACM), and the Chinese Society of Image and Graphics (CSIG). He has published more than twenty academic papers in TIP, TNNS, PR, AAAI, MM, and other internationally recognized journals and conferences in related fields and has applied for five invention patents.



Can Ma received his Ph.D. from the Institute of Computing Technology, Chinese Academy of Sciences, in 2012. He is currently a full senior engineer and Ph.D. supervisor at the Institute of Information Engineering, Chinese Academy of Sciences. His research interests include big data management and analysis in cyberspace and social networks.



Weiping Wang received a Ph.D. in Computer Science from the Harbin Institute of Technology, Harbin, China, in 2006. He is currently a Professor at the Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China. His research interests include big data, data security, databases, and storage systems. He has more than 100 publications in major journals and international conferences.