Multiplex Experts Governance Collaboration for Label Noise-Resistant Graph Representation Learning

Sichao Fu, Qinmu Peng, Yiu-Ming Cheung, *Fellow, IEEE*, Yizhuo Xu, Bin Zou, Xiao-Yuan Jing, and Xinge You, *Senior Member, IEEE*

***Abstract*—Recently emerged label noise-resistant graph representation learning (LNR-GRL) has received increasing attention, which aims to enhance the generalization of graph neural networks (GNNs) in semi-supervised node classification with noisy and limited labels. Most of the existing LNR-GRL tend to introduce more complex sample selection strategies developed in nongraph areas to distinguish more noisy nodes to alleviate their misguidance. However, these proposed methods neglect the importance of inaccurate graph structure relationships rectification, and information collaboration between inaccurate graph structure relationships and noisy node label rectification in improving the quality of noisy node identification and its rectified node labels. To solve the above-mentioned issues, we propose a novel multiplex experts governance collaboration (MEGC) framework for LNR-GRL. Specifically, an unsupervised graph structure governance expert is first designed to rectify inaccurate graph structure relationships. Based on the rectified graph structure, a simple label noise governance expert is proposed to accurately identify noisy node labels and further improve the quality of noisy nodes' rectified labels and unlabeled nodes' pseudo-labels. Finally, the above-proposed governance experts can be effectively combined with GNNs to jointly guide their training via the introduced cross-view graph contrastive loss and cross-entropy loss, which can maximally limit the effect of noisy node labels and discover more effective supervision guidance from data itself for GNNs optimization. Extensive experiments on three benchmarks, two label noise types, four noise rates, and four training label rates demonstrate the superiority of the proposed method in comparison to the existing LNR-GRL methods.**

***Index Terms*—Graph neural networks (GNNs), information collaboration, label noise, node classification, semi-supervised learning.**

Received 28 December 2024; revised 24 February 2025; accepted 25 July 2025. Date of publication 21 August 2025; date of current version 18 September 2025. This work was supported in part by the National Key Research and Development Program of China under Grant 2022YFF0712300, and in part by the National Natural Science Foundation of China under Grant 62172177. This article was recommended by Associate Editor Z. Yu. *(Corresponding author: Qinmu Peng.)*

Sichao Fu, Qinmu Peng, Yizhuo Xu, and Xinge You are with the School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: fusichao_upc@163.com; pengqinmu@hust.edu.cn; xuyizhuo@hust.edu.cn; youxg@mail.hust.edu.cn).

Yiu-Ming Cheung is with the Department of Computer Science, Hong Kong Baptist University, Hong Kong, SAR, China (e-mail: ymc@comp.hkbu.edu.hk).

Bin Zou is with the Faculty of Mathematics and Statistics, Hubei Key Laboratory of Applied Mathematics, Hubei University, Wuhan 430062, China (e-mail: zoubin0502@hubu.edu.cn).

Xiao-Yuan Jing is with the School of Computer Science, Wuhan University, Wuhan 430072, China (e-mail: xiaoyuanjing@whu.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TSMC.2025.3595183.

Digital Object Identifier 10.1109/TSMC.2025.3595183

I. Introduction

With the development of technologies, such as the Internet, cloud computing, and the Internet of Things, the connection between things in real-world applications is everywhere. As a kind of ubiquitous data structure, the graph can more reasonably describe complex topological structure relationships among objects owing to its powerful expression ability [1]. In recent years, the emergence of graph neural networks (GNNs) has provided a powerful tool for semi-supervised node classification with complex topological structure relationships [2], [3], [4]. Nevertheless, GNNs' generalization highly relies on the effective guidance of the accurate training node labels. The node labels for GNNs' optimization are often noisy and limited due to some unpredictable real-world factors. Misleading information propagation between noisily labeled and unlabeled nodes caused by the message-passing mechanism on the provided (original) graph structure will seriously hurt their performance [5], [6]. Therefore, how to develop effective label noise-resistant and label-efficient GNNs to improve the performance of semi-supervised node classification with limited and noisy labels is a very challenging and important issue in real-world applications.

The main challenge to addressing the above issue is how to make the most of the clean node label supervision information while maximum limiting the impact of noisy node labels. To improve the robustness of the existing GNNs on graphs with noisy and limited labels, label noise-resistant graph representation learning (LNR-GRL) is proposed to address this issue, which has recently received increasing research interest across various graph-related tasks [7]. The existing LNR-GRL can be divided into two groups: 1) loss correction-based LNR-GRL and 2) sample selection-based LNR-GRL.

Loss correction-based LNR-GRL enhances the ability to combat the label noise of GNNs on graphs with noisy and limited labels by correcting the loss value of noisily labeled training nodes. For example, D-GNN [6] simultaneously introduced the label noise estimation and loss correction modules to improve the tolerance of GNNs on graphs with noisy and limited labels, where label noise estimation is

introduced to generate an estimated correction matrix for GNNs optimization. NCR [8] proposed an effective neighbor consistency regularization to preserve the consistency of each node and its neighbor nodes' similarities in the feature and predicted label spaces, which weakens the effect of noisily labeled nodes.

Sample selection-based LNR-GRL introduces some strategies to distinguish the noisily labeled nodes of the provided training nodes, and then select the accurately labeled nodes for GNNs optimization. For example, SSGs [9] generated an affinity graph to regularize the distinguishing process of noisily labeled nodes via entropy rate superpixel and adaptive combination, i.e., the higher the similarity of a pixel, the more similar its corresponding labels. RTGNN [10] trained two GNNs and regarded the labeled nodes with small-loss values at the same time in each GNNs as the cleanly labeled nodes. After that, the selectively clean node labels, rectified noisy node labels, and pseudo-labels of partially unlabeled nodes are simultaneously used for GNNs optimization. CGNN [11] identified noisily labeled nodes by calculating the consistency score of each labeled node and its neighborhood nodes label, where the pseudo-label of unlabeled nodes is generated by the pretrained GNNs with the multiview graph contrastive loss guidance. MTS-GNN [12] regarded the GNNs with the earlier iterations as teacher models and guided the student model to perform noisy node label correction via knowledge distillation.

Although many researchers have made abundant attempts to improve the adaptability of GNNs on graphs with noisy and limited labels, these proposed works independently consider the noisy node identification process by simply introducing more complex strategies proposed in nongraph-related tasks. For example, NRGNN [7] introduced the edge prediction module to generate the structure relationships between unlabeled nodes and labeled nodes to alleviate the effect of the noisy node labels. RTGNN [10] introduced the small-loss criterion to identify the clean and noisy node labels. Owing to the complex interactive relationships between nodes, the message passing of GNNs will cause the propagation of misleading information from noisily labeled nodes to their neighbor nodes, which further exacerbates the influence of noisy node labels for GNNs optimization. Simply improving the quality of noisy node identification or inaccurate graph structure relationships rectification cannot minimize the impact of noisy node labels. Thus, how to enhance the information collaboration between noisy node labels and inaccurate graph structure relationships rectification for improving the quality of noisy node identification and its rectified node labels is very important and also has not been effectively explored so far.

To solve the above-mentioned issue, we propose a simple and effective LNR-GRL method to better enhance the generalization of GNNs in semi-supervised node classification with noisy and limited labels, termed multiplex experts governance collaboration (MEGC). The main difference between the existing LNR-GRL and the proposed MEGC is that our proposed MEGC simultaneously focuses on making the most of all node label supervision information and maximum limiting the impact of noisy node labels. Specifically,

1) an effective unsupervised graph structure governance expert, including graph structure learning and cross-view graph contrastive learning module is proposed to improve the accuracy of graph structure relationships, which alleviates the effect of inaccurate graph structure relationships in noise information propagation. The graph structure learning module aims to adaptively generate an optimal graph structure for different graph-structured data, which makes the rectified graph structure better preserve the truly local geometric distribution between nodes. Cross-view graph contrastive learning module regards the rectified graph structure and the provided graph structure as different views of the same input nodes, which aims to guide the graph structure learning module's optimization by maximizing their mutual information.
2) Based on the rectified graph structure, we utilize the classic K-means to generate the cluster pseudo-label of each labeled node and select the clean node labels by preserving the consistency of the pseudo-label and its true label, i.e., the nodes with the same pseudo-label have similar true labels. To effectively utilize the supervision information from the noisy labeled nodes and unlabeled nodes, the label propagation algorithm is introduced to generate their pseudo-label information via the label propagation between the clean node labels and other nodes on the rectified graph structure. Through the above K-means and label propagation operation (termed label noise governance expert), we can further alleviate the effect of noisy node labels in noise information propagation.
3) We jointly train the unsupervised graph structure governance expert, label noise governance expert, and GNNs for semi-supervised node classification in an end-to-end way via the introduced cross-view graph contrastive loss and cross-entropy loss, where the clean node labels, the pseudo-label of noisy nodes and unlabeled nodes acquired by the above-proposed governance experts can provide more effective supervision guidance for GNNs optimization. By the information collaboration between noisy node identification and inaccurate graph structure relationships rectification, our proposed MEGC can better improve the quality of noisy node identification, noisy nodes rectified labels, and unlabeled nodes generated pseudo-labels, which indirectly enhances the robustness of GNNs on graphs with noisy and limited labels.

The main contributions are summarized as follows.

1) We propose a novel MEGC framework for LNR-GRL from a MEGC perspective. To the best of our knowledge, we are the first attempt to reveal the importance of information collaboration between noisy node labels and inaccurate graph structure relationships rectification in improving the quality of noisy node identification and its rectified node labels.
2) To alleviate the effect of inaccurate graph structure relationships and noisy node labels for GNN optimization during noise information propagation, an effective

unsupervised graph structure governance expert and a simple label noise governance expert are simultaneously proposed to solve these issues. Through an end-to-end training way, the proposed MEGC framework can minimize the impact of noisy node labels and discover more effective supervision guidance from the data itself for GNNs optimization.

3) To demonstrate the effectiveness of the proposed method, we conduct extensive experiments on three benchmarks, two label noise types, four noise rates, and four training label rates. These experimental results show the superior performance of our proposed MEGC framework.

II. Related Works

A. Graph Neural Networks

In the past few years, GNNs have become a dominant method in graph representation learning [13], [14], [15]. Nowadays, a large number of GNNs variants have been proposed and have also succeeded in their application domains. These emerged GNNs-related works can mainly be split into two categories: spatial-based GNNs and spectral-based GNNs.

Spatial-based GNNs define the graph convolution operation as the information aggregation process between nodes via path convolution. For instance, GAT [16] stacked multi-masked self-attentional layers to adaptively encode pairwise connection relationships between given arbitrary node features. LC-GAT [17] aimed to update the wrong/uncertain connection relationships according to the label distribution information generated on a pretrained two-layer GAT. In single-image super-resolution, Yan et al. [18] utilized GAT to fully mine similarity information between different patches in low-resolution images for enhancing their texture details. Zhou et al. [19] proposed a unified deep sparse GAT for solving the expensive computational complexity and the inaccurate pruning problem of the scene graph generation task, where the multiview features, including the subject node, object node, and the edge features from multiple subspaces are introduced to jointly model complex connection relationships. GATv2 [20] aimed to solve the unconditioned ranking problem of the attention scores between different nodes in GAT by modifying the order of internal operations.

Spectral-based GNNs regard the convolution operation on graph-structured data as the information denoising process via a graph Fourier filter from the graph signal processing aspect. For instance, SGC [21] reduced the unnecessary complexity and redundant computation of the spectral graph convolution operation by removing the nonlinearities of the hidden layer. HGNN [22] utilized the hypergraph structure to encode the beyond pairwise connection relationships between nodes. EFGCN [23] given the definition of the spectral example-feature graph from the existing spectral graph convolution, which can better utilize the complementary information from different graph structure types. GCN-RW [24] revealed the feasibility of random features in improving the learning efficiency of GNNs from a mathematical theoretical aspect. It further designed an effective GNNs training acceleration model by introducing the graph convolutional layer with random filters and least squares loss regularization constraint. PEGFAN [25] proposed the Haar-type graph structure description method with permutation equivariance property to extract multiscale information of heterophilous graph, which avoids the over-smoothing issue of the existing multihop aggregation methods.

While more and more GNNs variants have achieved extensive application in various areas, their superior performance seriously depends on an ideal training data hypothesis with accurate training node labels. Owing to some unpredictable real-world factors, this hypothesis is difficult to meet in most cases, which limits its scope of application. Nevertheless, the existing LNR-GRL independently improves the quality of noisy node identification or inaccurate graph structure relationships rectification to enhance the robustness of GNNs on graphs with noisy and limited labels, which cannot minimize the impact of noisy node labels. This article first investigates the importance of information collaboration between noisy node labels and inaccurate graph structure relationship rectification, and then designs a novel LNR-GRL model from a MEGC perspective.

B. Deep Graph Structure Learning

The superior performance of GNNs depends on a fundamental assumption: the collected graph structure can accurately describe the complex interaction relationships between nodes for GNNs optimization. Unfortunately, real-world applications are difficult to meet this assumption requirement owing to some unpredictable factors, such as privacy policy, expensive computational costs, human error, and so on. Thus, GNNs cannot generate their robust feature representations for each node via the message-passing mechanism on the collected graph structure, which seriously hurts the generalization and applicability of GNNs in various graph-related tasks [26]. To alleviate the effect of inaccurate graph structure relationships for GNNs optimization, a large number of deep graph structure learning (DGSL) works are proposed to optimize the wrong, missing, and uncertain interaction relationships of collected graph structure [27]. Existing DGSL methods can mainly be categorized into two groups: 1) supervised DGSL and 2) unsupervised DGSL.

Supervised DGSL utilizes the label information from a specific downstream task to guide the joint optimization of graph structure relationships along with GNNs. For instance, PGN [28] regarded the dynamic inference of connection relationships as the addition/deletion of connection relationships at time t via dot-product self-attention. GEN [26] inferred a more precise graph structure by introducing the structure and observation model to constrain the graph generation and multi-faceted information fusion process, respectively. MRGCN [29] introduced a manifold regularization term about predicted label information to the objective function of GNNs for enhancing graph signals' smoothness between neighborhood nodes. LSPE [30] made GNNs generate more expressive node

features by decoupling positional and structural representations of given nodes to fully capture the global neighbor relationships of each node.

Unsupervised DGSL aims to improve the accuracy of graph structure relationships by designing some effective self-supervised tasks to discover effective supervision information from the data itself. For instance, SUBLIME [31] maximized the consistency of mutual information between the provided graph-structured data and learned graph-structured data to learn universal and edge-unbiased graph structure relationships. FedSKA [32] generated a client graph that facilitates to prediction of the client representations to improve the effectiveness of personalization federated learning by maximizing the similarity between structure-based features and local features. BrainUSL [33] incorporated a topology-aware encoding module into a graph generation module to rectify the redundant and noise connections of functional connectivity between brain regions, which can further boost the performance of deep learning-based brain disease diagnosis approaches. MCGMAE [34] trained robust graph-masked autoencoders to learn an optimal graph structure applied to arbitrary downstream tasks under the guidance of self-supervised information from the feature-encoder, feature reconstruction, and graph-structure reconstruction tasks.

Although DGSL has received increasing interest in various graph-related tasks, no study has attempted to reveal the insufficiency of DGSL technology in resisting the impact of noisy node labels on graphs. At the same time, no study also has attempted to enhance the information collaboration between DGSL and noisy node label rectification technology to improve the generalization of GNNs on graphs with noisy and limited labels.

III. Proposed Method

In this section, we describe how to enhance information collaboration between noisy node labels rectification and inaccurate graph structure relationships rectification to train a more robust LNR-GRL model in detail from a MEGC perspective. The overview of the proposed MEGC framework is shown in Fig. 1, which is composed of three components: 1) unsupervised graph structure governance expert; 2) label noise governance expert; and 3) model training and testing.

A. Unsupervised Graph Structure Governance Expert

Mathematical Notations: $\mathcal{G} = (\mathcal{V}, \mathcal{E}, X)$ is introduced to describe an undirected and unweighted graph, where $\mathcal{V} = \{v_1, \ldots, v_N\}$ denotes the set of N nodes, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ represents the set of node edges, and $X = \{x_1, \ldots, x_N\} \in R^{N \times d}$ is the set of N node features. N denotes the number of nodes, x_N denotes the feature vector of the node v_N, and d denotes the dimension of each node embedding. $A = \{A_{11}, \ldots, A_{NN}\} \in \{0, 1\}^{N \times N}$ represents the adjacency matrix of $\mathcal{G}$, where $A_{ij} = 1$ if node v_i and node v_j exits a connected edge, otherwise $A_{ij} = 0$. $\mathcal{Y} = \{y_1, \ldots, y_N\} \in R^{N \times C}$ is the set of N node labels, where $y_N \in \{0, 1\}^C$ denotes the one-hot label vector of node v_N, and C denotes the number of node classes.

In this article, we focus on the robust training problem of GNNs against noisy labels on semi-supervised node classification, which aims to infer the label information of unlabeled nodes $\mathcal{V}_{\mathcal{U}} = \mathcal{V} - \mathcal{V}_{\mathcal{L}}$ under the guidance of a moderate amount of labeled nodes $\mathcal{V}_{\mathcal{L}}$. $\mathcal{V}_{\mathcal{L}} = \{v_1, \ldots, v_l\}$ $(l \ll N)$ denotes the set of labeled nodes. Owing to some unpredictable factors in real-world applications, the provided node labels can be corrupted by various noises, where $\mathcal{Y}_{\mathcal{L}}^{\mathcal{T}} = \{y_1^T, \ldots, y_l^T\} \in R^{l \times C}$ and $\mathcal{Y}_{\mathcal{L}}^{\mathcal{N}} = \{y_1^N, \ldots, y_l^N\} \in R^{l \times C}$ denote the true label information and noisy label information of labeled nodes $\mathcal{V}_{\mathcal{L}}$.

Problem Definition 1: Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, X)$ with a set of noisy node labels $\mathcal{Y}_{\mathcal{L}}^{\mathcal{N}}$, the main objective of the proposed MEGC framework is to train a robust GNNs model on graphs with noisy and limited labels, which can improve the prediction accuracy of unlabeled nodes' true labels, i.e. $f(\mathcal{G}, \mathcal{Y}_{\mathcal{L}}^{\mathcal{N}}) \rightarrow \hat{\mathcal{Y}}_{\mathcal{U}}$. f is a robust classifier that we aim to learn, and $\hat{\mathcal{Y}}_{\mathcal{U}}$ denotes the set of unlabeled nodes' predicted labels.

Existing LNR-GRL assumes that the provided graph structure is credible enough to describe the local geometric distribution between nodes correctly. Nevertheless, graph structure collected from various real-world applications will have some wrong, missing, and uncertain connection relationships [27], [34] due to some unpredictable factors, which seriously hurt the generalization of the existing LNR-GRL. For example, media users who share the same interests on social networks will miss numerous potential friends owing to space and time limitations.

To alleviate the effect of inaccurate graph structure relationships for LNR-GRL, an effective unsupervised graph structure governance expert is proposed to solve this issue. To be concrete, we utilize a graph encoder–decoder-based graph structure learning module to learn an optimal graph structure, which can better preserve the truly local geometric distribution between nodes, i.e.,

$$E = A'\sigma(A'XW_0^0)W_0^1 \tag{1}$$

$$S = \sigma(EE^T) \tag{2}$$

where $A' = \tilde{D}^{-(1/2)}(A + I_N)\tilde{D}^{-(1/2)}$. W_0^0 and W_0^1 denote the weight parameter matrix of each graph convolution layer. $\tilde{D}$ and I_N represent the degree matrix about $A+I_N$ and the identity matrix. E and S are the output node embedding matrix and rectified graph structure. σ and T denote a nonlinear function and the transpose operation of the matrix.

To improve the computational efficiency of the proposed unsupervised graph structure governance expert, a simple K-nearest neighbors (KNN) sparsification operation is introduced to refine the rectified graph structure into a sparse matrix S'

$$S'_{ij} = \begin{cases} S_{ij}, & j \in \text{topK}(i) \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

where $S' = \{S'_{11}, \ldots, S'_{NN}\}$ denotes the adjacency matrix after a post-processor. topK(i) is the set of nearest neighbor nodes that node v_i with topK minimum non-negative values.

Although a large number of graph structure learning methods have shown competitive performance in semi-supervised node classification, they cannot guarantee the accuracy of the rectified graph structure when provided labeled nodes'

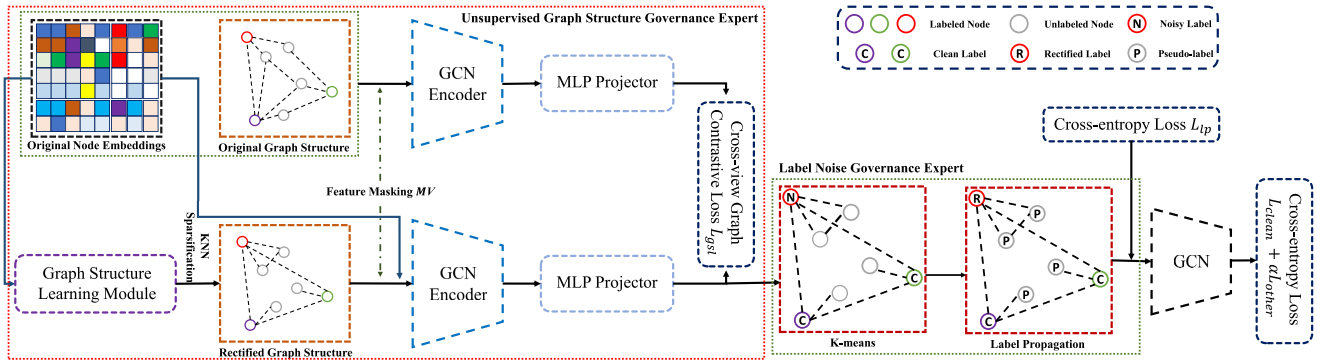


Fig. 1. Basic framework of the proposed MEGC model. There are a total of 6 steps: (1) Send original node embeddings X to an unsupervised graph structure governance expert to alleviate the effect of inaccurate graph structure relationships from A. Graph structure learning module is introduced to adaptively generate an optimal graph structure S' for the given X; (2) Graph data with the above different graph structure is sent to the same GCN encoder and MLP projector to learn their low-dimensional node embeddings in turn; (3) Cross-view graph contrastive loss $\mathcal{L}_{\text{gsl}}$ with feature masking scheme $\mathcal{MV}$ aims to guide the optimization of S' by the self-supervised information from the data itself; (4) The S' can be effectively combined with label noise governance expert to further alleviate the effect of noisy node labels from $\mathcal{Y}_{\mathcal{L}}^{\mathcal{N}}$. K-means module aims to identify clean node labels from $\mathcal{Y}_{\mathcal{L}}^{\mathcal{N}}$; (5) Label propagation module is introduced to generate the rectified-label of noisy nodes or pseudo-label of unlabeled nodes from $\mathcal{Y}_{\mathcal{L}}^{\mathcal{N}}$; (6) The above-proposed governance experts and GNNs are integrated into an end-to-end framework via $\mathcal{L}_{\text{clean}} + \mathcal{L}_{lp} + \alpha\mathcal{L}_{\text{other}} + \beta\mathcal{L}_{\text{gsl}}$, which can minimize the impact of noisy node labels and discover more effective supervision guidance for GNNs. These operations effectively improve the generalization of GNNs in semi-supervised node classification with noisy and limited labels.

supervision information is noisily and sparsely. Inspired by multiview contrastive learning, we propose to cross-view graph contrastive learning framework to discover the effective supervision information from the data itself, which can guide the training of the above-adopted graph structure learning module. The provided graph $\mathcal{G}_1 = (A, X)$ and the rectified graph $\mathcal{G}_2 = (S', X)$ are regarded as the different graph views of the same nodes, where $\mathcal{G}_1 = (A, X)$ plays a teacher role that provides stable and correct guidance for the training of the graph structure learning module. To enhance the fast adaptation of the graph structure learning module in different graph-structured data, we utilize the feature masking scheme to increase the challenges of training tasks, i.e.,

$$X' = [x_1 \odot mv^1, \ldots, x_N \odot mv^N]^T \tag{4}$$

where X' and $\odot$ denote the augmented node features matrix and the Hadamard product. $\mathcal{MV} = \{mv^1, \ldots, mv^N\} \in \{0, 1\}^d$ is the feature masking matrix. After that, two augmented graphs $\mathcal{G}_1' = (A, X')$ and $\mathcal{G}_2' = (S', X')$ are sent to a graph encoder module, including the GCN-based encoder [35] and MLP-based projector f_ψ to simultaneously learn the feature representations of all input nodes

$$\begin{cases} H_1 = A'\sigma(A'X'W_1^0)W_1^1 \\ H_2 = S''\sigma(S''X'W_2^0)W_2^1 \end{cases} \tag{5}$$

$$\begin{cases} Z_1 = f_\psi(H_1) \\ Z_2 = f_\psi(H_2) \end{cases} \tag{6}$$

where $S'' = \tilde{D}^{-(1/2)}(S' + I_N)\tilde{D}^{-(1/2)}$, $H_1 \in R^{N\times d_1}$, $H_2 \in R^{N\times d_1}$, $Z_1 \in R^{N\times d_2}$, and $Z_2 \in R^{N\times d_2}$. d_1 and d_2 denote the output feature dimension of the encoder and projector. W_1^0 and W_1^1 are the weight parameter matrix of the first and second graph convolution layers of the encoder module under the $\mathcal{G}_1'$ view. W_2^0 and W_2^1 are the weight parameter matrix of the first and second graph convolution layers of the encoder module under the $\mathcal{G}_2'$ view. ψ denotes the weight parameter matrix of projector f_ψ. The graph convolution layers number of the encoder and projector is all set to 2. The target of the projector f_ψ is to alleviate the influence of inconsistent measurement scale problems in H_1 and H_2.

After obtaining the node embedding Z_1 and Z_2, a symmetric normalized temperature-scaled cross-entropy loss $\mathcal{L}_{\text{gsl}}$ (cross-view graph contrastive loss) is introduced to maximize the mutual information between node embedding based on the provided graph structure A and rectified graph structure S', which can guide the optimization of the graph structure learning module

$$\mathcal{L}(Z_{1,i}, Z_{2,i}) = \log \frac{e^{\text{sim}(Z_{1,i}, Z_{2,i})/t}}{\sum_{j=1}^{N} e^{\text{sim}(Z_{1,i}, Z_{2,j})/t}} \tag{7}$$

$$\mathcal{L}_{\text{gsl}} = \frac{1}{2N}\sum_{i=1}^{N}\left[\mathcal{L}(Z_{1,i}, Z_{2,i}) + \mathcal{L}(Z_{2,i}, Z_{1,i})\right] \tag{8}$$

where t and $\text{sim}(Z_{1,i}, Z_{2,i})$ denote the temperature parameter and the cosine similarity between node embedding $Z_{1,i}$ and $Z_{2,i}$, respectively. $\mathcal{L}(Z_{1,i}, Z_{2,i})$ and $\mathcal{L}(Z_{2,i}, Z_{1,i})$ are two symmetric graph contrastive losses.

B. Label Noise Governance Expert

Apart from the governance of graph structure, how to reduce the impact of noisy node labels for LNR-GRL is equally important. Recently, some works have incorporated the effective sample selection strategies developed in computer vision into GNNs, which aim to remove the noisy node labels during GNNs optimization [36]. These emerging works assume that the loss distribution of clean labeled nodes is different from noisily labeled nodes. Nevertheless, such an assumption is often violated in real-world applications, which seriously reduces the effectiveness of LNR-GRL [37].

To improve the accuracy of noisy node identification and its rectified node labels, a simple label noise governance expert is proposed to solve this issue. Specifically, we send the rectified graph structure S' to the above-designed cross-view graph

contrastive learning framework to learn feature embedding of labeled nodes $\mathcal{V}_{\mathcal{L}}$. And then, the classic K-means algorithm is introduced to generate their cluster pseudo-labels, where the number of K is equal to the class number of the corresponding dataset. The selected nodes ${\mathcal{V}_{\mathcal{L}}}^{\text{clean}}$ that belong to the same class label ($\mathcal{Y}_{\mathcal{L}}^{\mathcal{N}}$) and cluster pseudo-label are regarded as clean nodes. For example, the set of randomly selected 8 nodes that belong to class 1 according to their noisy label information $\mathcal{Y}_{\mathcal{L}}^{\mathcal{N}}$ is $\mathcal{V}_L^1 = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\}$. The set of cluster pseudo-labels of these selected nodes $\mathcal{V}_L^1$ is $\mathcal{Y}_{\text{cluster}}^1 = \{1, 7, 4, 1, 5, 6, 1, 1\}$. Finally, we regard nodes $\{v_1, v_4, v_7, v_8\}$ with cluster pseudo-label 1 as the clean nodes in $\mathcal{Y}_{\mathcal{L}}^{\mathcal{N}}$. More importantly, the effectiveness of the introduced K-means-based sample selection strategy has been demonstrated by [36] from a theoretical analysis perspective.

Theorem 1: The majority of labeled nodes have true label information. The set of all noisy nodes that belong to class c is denoted as $\mathcal{V}_L^c$ and $||\mathcal{V}_L^c|| = m$, according to their noisy labels $\mathcal{Y}_{\mathcal{L}}^{\mathcal{N}}$. We assume the cluster accuracy of these selected nodes as Δ and their label noise rate $p < 0.5$. According to the above-mentioned sample selection example, $100 \times [((1-p) \times \Delta \times m)/((1-p) \times \Delta \times m + \tau)]$ nodes have correct labels, if $\tau \ll (1-p) \times \Delta \times m$ and $\Delta > (1/[2-2p])$.

Then, the clean nodes selected by the introduced K-means-based sample selection strategy can be used for GNNs optimization in semi-supervised node classification. Owing to the sparsity of cleanly labeled nodes after sample selection, only a small fraction of nodes would receive effective supervision signals. Such an imbalance phenomenon will seriously hurt the generalization of GNNs for semi-supervised node classification. In recent years, graph self-supervised learning that aims to discover supervision information from data itself has shown impressive performance in improving the generalization of GNNs on graphs with scarce labels. Inspired by those, we introduce a simple label propagation method to generate noisy nodes' rectified labels and unlabeled nodes' pseudo-labels, i.e.,

$$\mathcal{F}_t = \lambda S' \mathcal{F}_{t-1} + (I_N - \lambda)\mathcal{Y}' \tag{9}$$

where $\mathcal{F}_{t-1} \in R^{N \times C}$ represents the predicted labels at the timestamp $t-1$. $\mathcal{F}_0 = \mathcal{Y}' = [\mathcal{Y}_{\text{clean}}; \mathcal{Y}_{\text{noise}}; \mathcal{Y}_{\text{unlabeled}}] \in R^{N \times C}$ denotes the label set of clean nodes ${\mathcal{V}_{\mathcal{L}}}^{\text{clean}}$, noisy nodes ${\mathcal{V}_{\mathcal{L}}}^{\text{noisy}}$, and unlabeled nodes $\mathcal{V}_{\mathcal{U}}$, where $\mathcal{Y}'_{ij} = 1$ if node v_i is from the set of clean nodes ${\mathcal{V}_{\mathcal{L}}}^{\text{clean}}$ and $y_i = j$, otherwise $\mathcal{Y}'_{ij} = 0$. $\lambda \in (0, 1)$ is the control coefficient of the information propagation range. The closed-form solution $\mathcal{F}^*$ of $\mathcal{F}_t$ can be written as follows:

$$\mathcal{F}^* = (I_N - \lambda S')^{-1}\mathcal{Y}'. \tag{10}$$

To guarantee the quality of noisy node identification, noisy nodes' rectified labels, and unlabeled nodes' pseudo-labels, the cross-entropy loss $\mathcal{L}_{lp}$ is introduced to compute the loss values between the true labels $\mathcal{Y}'_{ij}$ and predicted scores $\mathcal{F}^{**}_{ij}$ of clean nodes for all parameter optimization

$$\mathcal{F}^{**}_{ij} = \text{SoftMax}(\mathcal{F}^*_{ij}) = \frac{\exp(\mathcal{F}^*_{ij})}{\sum_{k=1}^{C} \exp(\mathcal{F}^*_{ik})} \tag{11}$$

$$\mathcal{L}_{lp} = \sum_{i \in {\mathcal{V}_{\mathcal{L}}}^{\text{clean}}} \sum_{j=1}^{C} -\mathcal{Y}'_{ij} \log \mathcal{F}^{**}_{ij}. \tag{12}$$

Here, $\mathcal{F}^{**}$ denotes the normalized probabilistic score of $\mathcal{F}^*$ via the SoftMax function. $\mathcal{F}^*_{ij}$ is the jth element of $\mathcal{F}^*_i$.

C. Model Training and Testing

With the above-proposed unsupervised graph structure governance expert and label noise governance expert, the rectified graph structure S'', the true labels $\mathcal{Y}'_i$ ($i \in {\mathcal{V}_{\mathcal{L}}}^{\text{clean}}$) of clean nodes, and the pseudo-labels $\mathcal{F}^{**}_i$ ($i \in {\mathcal{V}_{\mathcal{L}}}^{\text{noisy}} \cup \mathcal{V}_{\mathcal{U}}$) of noisy nodes and unlabeled nodes all can be used to guide the optimization process of GNNs from semi-supervised node classification, i.e.,

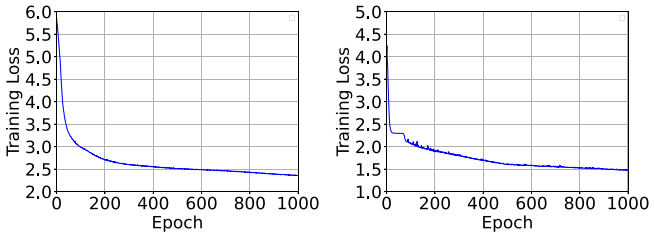
$$H = S''\sigma(S''XW^0)W^1 \tag{13}$$

where H denotes the output feature embedding matrix of all nodes. W^0 and W^1 are the weight parameter matrix of GNNs. In this article, we utilize an end-to-end training way to jointly optimize the unsupervised graph structure governance expert, label noise governance expert, and GNNs for semi-supervised node classification, which can maximally improve the generalization of GNNs on graphs with noisy and limited labels. The overall objection function $\mathcal{L}$ of the proposed MEGC framework can be written as follows:

$$\begin{aligned}\mathcal{L} &= \mathcal{L}_{\text{clean}} + \mathcal{L}_{lp} + \alpha\mathcal{L}_{\text{other}} + \beta\mathcal{L}_{\text{gsl}} \\ &= \sum_{i \in {\mathcal{V}_{\mathcal{L}}}^{\text{clean}}} \sum_{j=1}^{C} -\mathcal{Y}'_{ij} \log H'_{ij} + \mathcal{L}_{lp} \\ &\quad + \alpha \sum_{i \in {\mathcal{V}_{\mathcal{L}}}^{\text{noisy}} \cup \mathcal{V}_{\mathcal{U}}} \sum_{j=1}^{C} -\mathcal{F}^{**}_{ij} \log H'_{ij} + \beta\mathcal{L}_{\text{gsl}}\end{aligned} \tag{14}$$

where α and β denote the parameters to balance the contribution of different constraints for guiding the optimization process of GNNs from semi-supervised node classification. H' is the normalized probabilistic score of H via the SoftMax function. By minimizing verall objection function values $\mathcal{L}$ of the proposed MEGC framework, all trainable parameters W_0^0, W_0^1, W_1^0, W_1^1, W_2^0, W_2^1, ψ, λ, W^0, and W^1 can be updated via gradient descent. After multiple iterations, GNNs with the optimal parameters (W^0, W^1) trained on graphs with noisy and limited labels can be used to predict the label information of unlabeled nodes $\mathcal{V}_{\mathcal{U}}$.

Interaction Between Two Governance Experts: First, if we remove the unsupervised graph structure governance expert, the label noise governance expert will utilize the original graph structure to identify noisy node labels. Owing to the inaccuracy of the original graph structure relationships, the quality of noisy nodes' rectified labels and unlabeled nodes' pseudo-labels will be seriously influenced. The supervision information from clean node labels, noisy nodes' rectified labels, and unlabeled nodes' pseudo-labels will mislead GNNs optimization. Second, if we remove the label noise governance expert, we will utilize the rectified graph structure and noisy node labels without rectification for GNNs optimization. Owing to the inaccuracy of noisy node label supervision



(a) (b)

Fig. 2. Convergence analysis of MEGC on node classification task with 20% pair noise rate and 5% training label rate. (a) Cora. (b) Citeseer.

information and invalid utilization of unlabeled node self-supervision information, the generalization of GNNs will be seriously damaged. Thus, these two governance experts interact with each other. Simply improving the quality of noisy node identification or inaccurate graph structure relationships rectification cannot minimize the impact of noisy node labels.

D. Time Complexity Analysis and Convergence Analysis

In this part, we analyze the time complexity of each module of the proposed MEGC framework in detail. Specifically, for unsupervised graph structure governance expert, its corresponding graph structure learning, KNN sparsification, feature masking, GCN encoder and MLP projector is $\mathcal{O}(2md + 2nd^2)$, $\mathcal{O}(ndb)$, $\mathcal{O}(d)$, $\mathcal{O}(2md_1 + 2nd_1^2)$, $\mathcal{O}(2nd_2^2)$ in turns, where m represents the number of edges and b denotes the number of the batch size. In the label noise governance expert, the time complexity of K-means and label propagation is $\mathcal{O}(nd_2K)$, $\mathcal{O}(nd')$, respectively. d' is the average degree of nodes. In the proposed MEGC framework, its total time complexity is mainly related to the KNN sparsification, GCN encoder, MLP projector, K-means, and label propagation modules, i.e. $\mathcal{O}(ndb + 2md_1 + 2nd_1^2 + 2nd_2^2 + nd_2K + nd')$. To improve the computational efficiency of the proposed MEGC framework for practical implementation and application in real-world scenarios, the sky is to avoid $\mathcal{O}(n^2)$ time complexity. Thus, we introduce a mini-batch data load strategy and KNN sparsification post-processing. Owing to the small batch size (b), feature dimensions (d_1, d_2), category number (K) and average degree (d'), the total time complexity of the proposed MEGC framework is acceptable on the Tesla-V100 GPU. It indirectly reveals that the proposed MEGC framework can effectively scale to very large graphs. The change of node numbers only increases the data load time caused by the small batch size, when applying MEGC to very large graphs. To make a balance between efficiency and accuracy, we can utilize the widely used distributed training strategy for training acceleration of MEGC in very large graphs with millions of nodes. Fig. 2 further reports the training loss versus various epochs on the node classification task with 20% pair noise rate and 5% training label rate. As shown in Fig. 2, we can find that the value of the training loss eventually stabilizes within a certain stable threshold, as the number of iterations increases. In addition, graph datasets with more nodes require more convergence epochs. But generally speaking, MEGC can reach a stable convergence stage with only a relatively small number of epochs, which indirectly demonstrates the effectiveness of the proposed framework.

IV. Experiments

In this section, we conduct extensive experiments on three benchmarks, two label noise types, four noise rates and four training label rates, which aim to demonstrate the effectiveness of the proposed MEGC framework in improving the generalization of GNNs for semi-supervised node classification with noisy and limited labels. We first show the used datasets in Section IV-A, followed by the experiment setting of the proposed MEGC framework in Section IV-B. Sections IV-C–IV-E give a detailed experiment analysis for the reported experimental results.

A. Datasets

Citeseer, Cora [35] and BlogCatalog [38] are all graph data, which belong to the citation network and social network type, respectively. Specifically, Citeseer [35] contains 3327 machine learning papers. All nodes are categorized into 6 classes with a size of 3703 unique words, including Information Retrieval, Artificial Intelligence, Database, Agents, Human-Computer Interaction, and Machine Language. Cora [35] consists of 2708 papers collected from machine learning publications. It belongs to seven classes, such as rule learning, genetic algorithms, reinforcement learning, case-based, neural networks, theory, and probabilistic methods. Each paper is described by 0/1 with a size of 1433. BlogCatalog [38] is composed of a total of 10312 media users about 39 media users' interests. Each edge denotes the interest relationships between different media users.

B. Experimental Setup

In the semi-supervised node classification task, we randomly select 10%, 10%, and 80% nodes from each dataset as training samples, validation samples, and test samples, where $p\%$ ($p < 10$) nodes are further chosen from training samples as the labeled nodes. In this study, we introduce two classic label noise types to corrupt true label information of labeled training samples and validation samples, i.e., uniform noise and pair noise. For uniform noise, the true label information of labeled training samples and validation samples has a specific probability $\varrho/(C-1)$ to be uniformly flipped to other categories. For pair noise, the true label information of labeled training samples and validation samples has a specific probability ϱ to flip to their very similar pair categories.

For the proposed MEGC model, the feature dimension of the hidden units and output layer of the graph structure learning module, GCN-based encoder, and MLP-based projector f_ψ are searched in $\{8, 16, 32, 64, 128, 256, 512, 1024\}$. Feature masking rate $\mathcal{MV}$, balance parameter α and β are selected from 0.1 to 0.9. The neighbor numbers K' of the KNN sparsification operation is tuned amongst $\{10, 20, 30, 40, 50, 60, 70, 80, 90, 100\}$. All model parameters are initialized by the Adam optimizer with a total of 200 or 500 epochs, a dropout rate of 0.5, a learning rate of 0.001, and a weight decay of $5e-4$. To avoid any bias introduced

TABLE I
EXPERIMENT COMPARISON WITH THE STATE-OF-THE-ART LNR-GRL METHODS ON NODE CLASSIFICATION TASK WITH VARIOUS NOISE RATES AND 5% TRAINING LABEL RATE. THE BEST RESULTS ARE MARKED IN BLACK

Dataset	Noise Type	Uniform Noise				Pair Noise			
	Noise Rate (ϱ)	10%	20%	30%	40%	10%	20%	30%	40%
Cora	GCN [35]	78.1±1.4	74.6±3.5	69.3±1.2	61.6±6.1	77.5±2.2	73.5±3.9	66.6±2.3	60.3±1.9
	GIN [39]	-	72.3±0.9	-	-	-	74.7±1.4	-	-
	Adaptive Layer [40]	78.1±1.3	74.7±3.5	69.8±1.6	62.1±2.5	77.1±1.9	73.4±3.9	67.2±1.7	62.1±2.5
	Label Smoothing [41]	78.7±1.2	74.7±3.4	70.0±1.4	61.4±7.3	78.1±2.2	73.5±4.2	66.2±2.8	61.7±2.7
	Forward [42]	-	73.7±0.7	-	-	-	76.0±0.7	-	-
	GCE [43]	78.5±1.3	74.3±3.4	69.7±1.7	61.2±6.1	78.2±1.8	73.6±3.1	66.2±2.1	60.7±1.9
	Co-teaching [44]	78.2±1.5	74.7±3.0	68.8±1.4	62.6±3.9	77.3±1.0	72.2±4.1	64.4±2.2	59.9±4.2
	SCE [45]	78.9±0.4	77.0±0.8	69.8±0.9	60.0±3.4	80.5±0.2	77.4±1.0	67.8±2.4	57.8±1.3
	JoCoR [46]	79.3±0.2	76.9±0.3	72.9±1.4	68.2±2.5	80.7±0.3	77.6±0.3	69.2±1.4	63.8±0.9
	APL [47]	79.4±0.3	74.4±0.6	69.2±1.4	59.5±3.2	80.0±0.5	75.2±2.3	67.4±0.9	59.8±2.2
	Co-teaching+ [48]	76.4±1.2	73.6±1.7	64.4±1.3	54.8±3.6	78.2±1.2	73.8±1.4	63.6±3.9	59.4±2.5
	Self-Training [49]	-	75.6±1.8	-	-	-	76.4±1.4	-	-
	D-GNN [6]	78.1±1.4	74.7±3.5	69.3±1.1	61.5±6.1	77.5±2.2	73.5±3.8	66.5±2.4	61.5±6.1
	NRGNN [7]	81.5±1.5	79.0±2.3	77.0±2.0	70.5±2.5	80.5±1.4	76.1±3.3	72.2±2.5	63.7±2.0
	CP [50]	76.0±0.6	74.8±1.3	67.3±1.0	61.4±2.7	78.2±0.7	75.2±1.4	63.1±1.3	58.6±1.0
	RTGNN [10]	78.6±1.5	75.8±0.9	74.3±2.1	67.9±8.2	78.1±2.2	70.7±4.0	69.0±2.7	59.2±5.8
	MEGC (ours)	**84.3±0.3**	**82.1±0.4**	**81.1±0.7**	**79.0±0.4**	**83.6±0.5**	**81.2±0.2**	**78.6±0.4**	**72.7±0.3**
Citeseer	GCN [35]	69.2±1.7	65.8±2.3	61.3±3.2	57.1±4.5	68.1±2.4	64.9±2.4	60.6±2.5	54.5±5.5
	GIN [39]	-	65.7±2.1	-	-	-	61.6±1.0	-	-
	Adaptive Layer [40]	69.3±2.1	65.7±2.5	61.5±3.5	57.8±3.8	68.0±2.2	65.0±2.2	61.2±2.5	57.3±4.4
	Label Smoothing [41]	69.0±2.1	65.7±2.4	61.8±3.6	57.3±3.5	67.2±2.1	64.9±2.7	59.5±3.0	56.0±6.9
	Forward [42]	-	65.0±1.5	-	-	-	61.6±0.4	-	-
	GCE [43]	70.0±1.6	66.8±2.4	62.1±3.2	58.9±3.7	68.2±2.3	66.4±2.4	62.0±1.9	56.7±6.7
	Co-teaching [44]	69.9±1.5	67.8±2.5	63.4±2.6	61.7±2.2	68.4±2.1	66.3±1.9	63.4±2.9	57.6±2.4
	SCE [45]	70.2±0.4	69.4±0.7	64.9±1.6	56.2±1.4	71.1±0.2	67.8±0.8	59.8±0.6	57.9±1.0
	JoCoR [46]	72.2±0.3	71.3±0.7	69.1±0.3	58.1±2.3	72.1±0.6	69.6±0.4	64.4±2.5	58.1±1.6
	APL [47]	70.5±0.6	69.6±0.6	65.6±1.6	56.7±2.0	71.3±0.4	68.1±0.5	59.7±0.6	57.1±1.5
	Co-teaching+ [48]	66.9±2.3	66.4±1.3	60.5±5.8	53.0±4.0	66.5±2.2	65.1±2.1	60.0±5.0	50.5±3.3
	Self-Training [49]	-	67.8±1.4	-	-	-	62.0±1.6	-	-
	D-GNN [6]	69.5±1.9	66.0±2.7	61.3±3.2	57.1±4.5	67.9±2.6	64.9±2.4	60.7±2.5	54.7±5.3
	NRGNN [7]	70.5±1.0	68.0±1.5	65.2±3.5	57.6±5.8	69.4±0.8	67.2±2.2	64.2±5.1	57.6±5.8
	CP [50]	68.3±0.8	66.0±1.6	65.0±1.8	57.0±1.7	64.9±0.9	62.0±1.0	60.3±0.7	53.0±1.0
	RTGNN [10]	69.5±2.2	67.7±0.8	65.0±1.5	61.5±2.9	68.8±1.5	65.5±3.8	61.1±5.9	59.0±3.2
	MEGC (ours)	**73.2±0.1**	**72.7±0.2**	**71.4±0.6**	**70.0±0.8**	**72.8±0.2**	**70.7±0.6**	**68.0±0.3**	**65.7±2.7**
BlogCatalog	GCN [35]	69.9±1.0	67.5±2.3	65.4±1.8	63.6±1.7	69.9±1.2	65.1±0.9	60.5±2.5	52.7±3.3
	Adaptive Layer [40]	70.3±1.4	68.4±1.8	65.5±1.5	62.9±2.5	70.8±1.1	66.3±1.1	62.6±2.0	54.4±3.2
	Label Smoothing [41]	69.7±1.5	67.3±2.2	65.3±2.0	62.3±3.1	70.6±1.6	64.9±1.3	60.6±1.8	51.8±2.7
	GCE [43]	71.4±1.5	70.4±1.6	69.7±0.7	66.6±2.7	70.4±1.1	69.4±1.1	65.2±2.9	58.3±2.0
	Co-teaching [44]	71.4±1.7	68.3±2.2	66.5±3.0	63.5±2.6	**71.9±1.3**	67.7±0.8	60.3±5.9	52.2±1.3
	SCE [45]	70.7±0.5	68.7±0.8	67.5±0.9	66.5±1.0	69.4±0.7	64.3±1.4	60.6±1.0	57.0±0.9
	JoCoR [46]	70.9±0.4	69.6±0.4	69.3±1.1	66.3±1.1	70.5±0.7	66.3±1.1	59.5±1.1	58.3±2.2
	APL [47]	70.9±0.5	69.2±0.4	68.7±0.6	65.8±0.9	70.3±0.6	68.4±0.8	61.5±1.3	58.4±1.6
	D-GNN [6]	69.9±1.1	67.5±2.3	65.0±1.7	63.8±0.9	70.1±1.0	65.2±0.8	60.1±2.0	52.5±3.3
	NRGNN [7]	71.4±1.5	70.4±1.6	69.7±0.7	66.6±2.7	71.8±1.1	69.4±1.1	65.2±2.9	58.3±2.0
	RTGNN [10]	71.1±0.8	70.9±1.3	70.5±1.0	**70.3±2.0**	70.8±1.0	71.4±0.8	**70.3±2.0**	60.8±3.4
	MEGC (ours)	**71.9±0.9**	**71.3±0.7**	**71.0±0.3**	69.3±0.4	71.4±0.4	**71.6±0.3**	69.5±0.2	**62.5±1.1**

by the random partitioning of data, we carried out many times independently to report the average accuracy of all modes. During the model testing phase, we utilize the optimal parameters of GNNs (13) to validate the accuracy of semi-supervised node classification. All hyperparameters are tuned based on the accuracy of the validation samples via the popular grid search technique.

C. Comparison With State-of-the-Art Methods

In this article, we compare the proposed MEGC framework with three types of methods, including two GNNs without label noise limitation methods [35], [39], nine label noise-resistant nongraph representation learning methods (LNR-NGRL) [40], [41], [42], [43], [44], [45], [46], [47], [48], and five LNR-GRL methods [6], [7], [10], [49], [50]. Table I reports the classification accuracy of all models on node classification tasks with various noise rates. From these results, we can acquire the following observations.

1) As the label noise rate increases, the classification performance of both GCN [35] and GIN [39] methods is getting worse and worse, and these GNNs without label noise limitation methods also perform even worse than other models in most cases. These results powerfully reveal the serious influence of noisy node label information on GNNs optimization. Compared with GCN and GIN, LNR-NGRL achieves better performance, especially under low noise rates. This implies that sample selection or loss correction can alleviate the negative effect of label noise on GNNs optimization, which also indirectly demonstrates the inevitability of noisy node label information limitation.

TABLE II
EXPERIMENT COMPARISON WITH THE STATE-OF-THE-ART LNR-GRL METHODS ON NODE CLASSIFICATION TASK WITH 20% NOISE RATE AND VARIOUS TRAINING LABEL RATES. THE BEST RESULTS ARE MARKED IN BLACK

Dataset	Noise Type	Uniform Noise				Pair Noise			
	Label Rate (p)	3%	5%	7%	9%	3%	5%	7%	9%
Cora	Co-teaching+ [48]	64.9±1.0	73.6±1.7	74.0±1.6	75.2±1.6	68.4±1.0	73.8±1.4	74.9±1.6	76.0±2.3
	D-GNN [6]	67.9±0.8	74.7±3.5	76.1±0.8	77.0±1.9	70.8±1.1	73.5±3.8	73.9±0.8	75.4±1.9
	NRGNN [7]	73.8±0.7	79.0±2.3	80.2±0.6	80.5±1.6	73.4±1.6	76.1±3.3	78.9±1.1	79.4±1.3
	CP [50]	66.9±1.0	74.8±1.3	75.7±0.8	77.2±0.6	70.2±1.8	75.2±1.4	75.8±0.8	76.8±0.6
	RTGNN [10]	74.2±1.3	75.8±0.9	78.5±0.9	81.2±1.5	67.8±1.2	70.7±4.0	75.1±1.0	81.9±0.5
	MEGC (ours)	**75.3±0.4**	**82.1±0.4**	**82.6±0.2**	**83.7±0.6**	**77.1±0.1**	**81.2±0.2**	**81.9±0.5**	**83.0±0.4**

TABLE III
ABLATION STUDY ON NODE CLASSIFICATION TASK WITH VARIOUS NOISE RATES AND 5% TRAINING LABEL RATE

Dataset	Noise Type	Uniform Noise				Pair Noise			
	Noise Rate (ϱ)	10%	20%	30%	40%	10%	20%	30%	40%
Cora	MEGC (w/o UGSGE, w/o LNGE)	80.1±0.4	78.2±0.2	66.6±1.0	64.9±0.5	79.8±0.7	75.5±0.8	72.8±0.6	65.7±1.2
	MEGC (w/o UGSGE)	82.4±0.2	78.9±0.6	74.3±1.0	70.0±0.7	81.3±0.4	78.1±0.6	76.1±0.8	67.3±1.0
	MEGC (w/o LNGE)	82.3±0.4	79.2±0.3	78.2±0.8	70.9±1.1	81.6±0.7	79.8±0.8	74.9±0.5	69.6±0.8
	MEGC (w/o L_{other}, w/o L_{lp})	81.3±0.2	79.0±0.4	74.3±0.3	70.3±0.4	81.1±0.3	77.2±1.5	74.0±0.5	67.4±0.9
	MEGC (w/o L_{other})	83.7±0.3	81.4±0.3	79.5±0.4	77.6±0.5	82.5±0.6	80.7±0.4	77.8±0.5	71.9±1.6
	MEGC	**84.3±0.3**	**82.1±0.4**	**81.1±0.7**	**79.0±0.4**	**83.6±0.5**	**81.2±0.2**	**78.6±0.4**	**72.7±0.3**

2) Although these LNR-NGRL models improve the classification performance of semi-supervised node classification with noisy and limited labels to some extent, they only achieve limited improvements and even perform worse than GCN [35] in some cases, such as APL [47] and Co-teaching+ [48]. The main reason is that neglecting the graph structure relationships between nodes during noisy node label information limitation cannot more accurately identify more noisy nodes. Besides, the noisy node labels have a serious influence on GNNs optimization owing to the semi-supervised learning setting. To improve the quality of noisy node identification, LNR-GRL is proposed to effectively utilize the underlying graph structure relationships between nodes. Compared with LNR-NGRL, LNR-GRL achieves competitive results, especially under higher noise rates, which demonstrates the benefit of graph structure relationship utilization in noisy node identification.
3) Our proposed MEGC framework achieves the best and runner-up classification performance on all benchmarks, noise rates, and label noise types than all models. Specifically, MEGC obtains the 2.8%, 3.1%, 4.1%, 8.5%, 1.0%, 1.4%, 2.3%, and 8.3% improvements in comparison to the state-of-the-art methods on the Cora and Citesee datasets with various uniform noise rates, respectively. [i.e., NRGNN (10%, 20%, 30%, and 40%), JoCoR (10%, 20%, and 30%) and Co-teaching (40%)]. On the one hand, it demonstrates the superiority of the proposed MEGC framework in semi-supervised node classification with noisy and limited labels; on the other hand, it also reveals the benefit of information collaboration between noisy node labels and inaccurate graph structure relationships rectification in improving the quality of noisy node identification and its rectified node labels in comparison to graph structure optimization-based LNR-GRL method [7] and sample selection-based LNR-GRL methods [10], [49], [50].
4) As shown in Table I, a common phenomenon is that the classification accuracy of all comparison methods drops dramatically when the label noise rate increases from 10% to 40% in sequence. On the contrary, our proposed MEGC has stable performance and lower performance dropping rates in most cases, for example, on the Citeseer dataset with various noise rates and types. These results imply that our MEGC is more resistant to alleviating the negative effect of label noise.

To further demonstrate the effectiveness of the proposed MEGC under various training label rates, we report the classification performance of the proposed MEGC and other comparison methods in Table II. From these results, we can see the following.

1) Our MEGC outperforms all comparison methods by a large margin, especially when the training label rate is very small. The main reason is that our method can more accurately discriminate noisy nodes and further provide more effective self-supervision information for GNNs optimization via the information collaboration between noisy node labels and inaccurate graph structure relationships rectification.
2) As the training label rate increases, the performance between the proposed MEGC and the state-of-the-art methods decreases, especially under high training label rates. When the training label rate is high, the labeled samples for GNNs optimization are sufficient, which will reduce the sensitivity and importance of GNNs for noisy nodes and their rectified node labels.

D. Ablation Experiments

In Table III, we conduct extensive experiments to analyze the influence of each component. w/o denotes without a specific module. UGSGE and LNGE represent the unsupervised graph structure governance expert and label noise governance expert modules, respectively. MEGC (w/o UGSGE) represents

utilizing the original graph structure for noisy node labels recognition and GNNs optimization. MEGC (w/o LNGE) represents utilizing the rectified graph structure and noisy node labels without rectification for GNNs optimization. MEGC (w/o L_{other}, w/o L_{lp}) denotes only utilizing the selected clean nodes' label for GNNs optimization. MEGC (w/o L_{other}) denotes without introducing L_{lp} to guarantee their quality when MEGC utilizes the noisy nodes' rectified labels and unlabeled nodes' pseudo-labels for GNNs optimization. From the first, second and third columns of Table III, we can observe that MEGC (w/o UGSGE) and MEGC (w/o LNGE) all outperform MEGC (w/o UGSGE, w/o LNGE), which implies LNGE and UGSGE are helpful to alleviate the negative effect of label noise for GNNs optimization. For example, MEGC (w/o UGSGE) and MEGC (w/o LNGE) achieve gains of 2.3%, 0.7%, 7.7%, 5.1%, 2.2%, 1%, 11.6%, 6% than MEGC (w/o UGSGE, w/o LNGE) under 10%, 20%, 30%, and 40% uniform noise rates, respectively. On the one hand, these results indicate that the LNGE module plays a significant role in alleviating the negative effect of label noise than the UGSGE module; on the other hand, they indirectly indicate the insufficient of the existing graph structure relationships rectification constraint and the importance of a unified and effective standard for direct constraint the inaccurate graph structure relationships optimization. From the second, third, and sixth columns of Table III, we can see that MEGC acquires large performance improvements in comparison to MEGC (w/o UGSGE) and MEGC (w/o LNGE), such as MEGC obtains 1.9%, 3.2%, 6.8%, 9%, 2%, 2.9%, 2.9%, and 8.1% improvement than MEGC (w/o UGSGE) and MEGC (w/o LNGE) under 10%, 20%, 30%, and 40% uniform noise rates, respectively. These results powerfully reveal the importance of information collaboration between noisy node labels and inaccurate graph structure relationships rectification in improving the quality of noisy node identification and its rectified node labels. From the fourth, fifth, and sixth columns of Table III, MEGC (w/o L_{other}, w/o L_{lp}) perform worse than MEGC (w/o L_{other}) and MEGC, especially under high noise rate, which indicates the benefit of self-supervision information from the noisy nodes and unlabeled nodes. For instance, MEGC (w/o L_{other}) and MEGC achieve improvement of 2.4%, 3%, under 10% uniform noise rates in turns. Although MEGC (w/o L_{other}) outperforms MEGC (w/o L_{other}, w/o L_{lp}) by a large margin, it is difficult to guarantee that the self-supervision information from the noisy nodes and unlabeled nodes is not misleading for GNNs optimization. In summary, Table III has powerfully demonstrated the effectiveness of all components in alleviating the negative effect of label noise.

E. Parameters Sensitivity

In this part, we further conduct extensive experiments to implement the sensitivity analysis of MEGC under three important parameters, including neighbor numbers K', feature masking rate $\mathcal{MV}$, balance parameters α and β in Figs. 3–5.

Effect of the Neighbor Numbers K': Fig. 3 shows the performance of MEGC with various K' values on node classification task with 20% pair noise rate and 5% training label rate, when we change K' from 10 to 100. The size of the neighbor numbers K' is closely related to the sample numbers in each category. As is demonstrated in Fig. 3, MEGC all rise first and then decrease on different datasets, and a common phenomenon is that a too small K' (Citeseer) and too large K' (Cora) both damage their classification performance. For example, MEGC with $K' = 80$ achieve drops of 6.2% and 4.6% in comparison to MEGC with the best K' values on the Cora ($K' = 40$) and Citeseer ($K' = 100$) datasets, respectively. The main reason is that a too large K' on the Cora dataset with the small sample sizes can cause the over-smoothing problem between the updated node features via GNNs owing to the existence of inaccurate and misleading connection relationships. On the one hand, these results demonstrate that the proposed MEGC is more sensitive to the selection of neighbor numbers K'; on the other hand, they indirectly reveal how to select the best K' values for different datasets is very important.

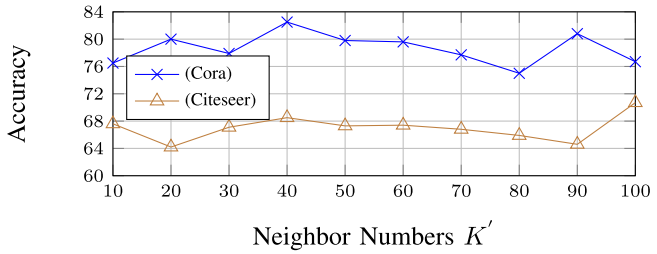


Fig. 3. Sensitivity analysis of the neighbor numbers K' on node classification task with 20% pair noise rate and 5% training label rate.

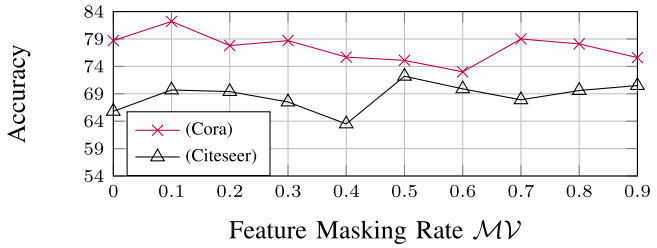


Fig. 4. Sensitivity analysis of the feature masking rates $\mathcal{MV}$ on node classification task with 20% pair noise rate and 5% training label rate.

Effect of the Feature Masking Rate $\mathcal{MV}$: Fig. 4 reports the influence of the feature masking rate $\mathcal{MV}$ for MEGC on the Cora and Citeseer datasets. As shown in Fig. 4, the performance of MEGC will drop dramatically when we remove the feature masking scheme. For example, MEGC without the feature masking operation obtains drops of 2.5% and 4.9% in comparison to the best results on the Cora ($\mathcal{MV} = 0.1$) and Citeseer ($\mathcal{MV} = 0.5$) datasets, respectively. These results reveal the effectiveness of the feature masking scheme with disturbing the node embeddings in improving the generalization of the graph structure learning module. In addition, a too large $\mathcal{MV}$ (Cora) and too small $\mathcal{MV}$ (Citeseer) both result in poor performance, such as MEGC with $\mathcal{MV} = 0.1$ improves by 8.2% and 5.6% in comparison to MEGC with $\mathcal{MV} = 0.6$ and MEGC with $\mathcal{MV} = 0.9$ in turns. MEGC with $\mathcal{MV} = 0.5$ improves by 1% and 7.2% in comparison

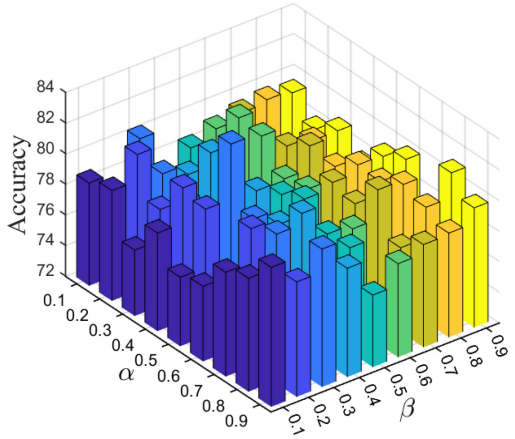


Fig. 5. Average accuracy on the Cora dataset versus balance parameters α and β of the proposed MEGC on node classification task with 20% pair noise rate and 5% training label rate.

to MEGC with $\mathcal{MV} = 0.1$ and MEGC with $\mathcal{MV} = 0.4$, respectively. We conjecture that

1) For the Cora dataset with low-dimensional node features, the origin node embeddings will be seriously undermined due to a too large $\mathcal{MV}$. This operation makes the graph encoder module unable to learn more discriminative node features, which indirectly influences the guidance process of cross-view graph contrastive learning.
2) For the Citeseer dataset with high-dimensional node features, a too small $\mathcal{MV}$ will reduce the robustness of rectified graph structure relationships owing to the easy challenge of self-supervised learning tasks. These results reveal the sensitivity of MEGC in the selection of feature masking rate $\mathcal{MV}$ and the importance of the reasonable selection of feature masking rate $\mathcal{MV}$ for task performance.

Effect of the Balance Parameters α and β: To show the relationships between balance parameters (α and β) and the classification performance of MEGC, Fig. 5 reports the accuracy versus different balance parameters α and β values on the Cora dataset with 20% pair noise rate and 5% training label rate. As shown in Fig. 5, MEGC with $\alpha = 0.5$ and $\beta = 0.3$ achives gains of 0.9 % and 4.7 % in comparison to MEGC with $\alpha = 0.1$ and $\beta = 0.3$, and MEGC with $\alpha = 0.5$ and $\beta = 0.1$, respectively. In this article, MEGC with $0 < \alpha < 0.6$ and $0 < \beta < 0.6$ achieves the best performance in most cases, and a too large α and β will result in worse performance. The main reason that excessive trust for UGSGE and LNGE will mislead the optimization process of GNNs, owing to the unreliable of the rectified graph structure relationships optimized by $\mathcal{L}_{gsl}$, and noisy nodes' rectified labels and unlabeled nodes generated pseudo-labels by LNGE. Thus, how to reasonably execute information collaboration between noisy node labels and inaccurate graph structure relationships rectification for LNR-GRL is vital.

V. Conclusion

In this article, we propose a simple and effective LNR-GRL method from a MEGC perspective to effectively enhance information collaboration between noisy node labels rectification and inaccurate graph structure relationships rectification, which aims to maximally alleviate the effect of inaccurate graph structure relationships and noisy node labels in the information propagation process of GNNs. To be concrete, to alleviate the effect of inaccurate graph structure relationships, an unsupervised graph structure governance expert is proposed to learn an optimal graph structure that describes the local geometric distribution between nodes correctly. To further alleviate the effect of noisy node labels, we propose a rectified graph structure-based label noise governance expert to improve the quality of noisy node identification, noisy nodes rectified labels, and unlabeled nodes' pseudo-labels. Finally, a simple end-to-end training framework is proposed to effectively improve the generalization of GNNs on graphs with noisy and limited labels under the guidance of cross-view graph contrastive loss and cross-entropy loss. Extensive results under three benchmarks, two label noise types, and four noise rates and four training label rates validate the superiority of MEGC over the existing methods.

While the proposed MEGC improves the performance of semi-supervised node classification with noisy and limited labels by a large margin, our MEGC still faces the following challenges to be addressed.

1) Unquantifiable graph structure relationships rectification constraint. The proposed MEGC indirectly achieves the optimization of inaccurate graph structure relationships by maximizing the mutual information between node embeddings. Owing to the lack of a unified and effective standard for directly constraining the inaccurate graph structure relationships optimization, MEGC makes the identified clean nodes still exist with part noisy nodes and further misleads the parameters optimization of GNNs.
2) Ineffectiveness in resisting multiple types of noise. Label noise, structure noise and feature noise are key factors affecting the robustness of graph representations for GNNs. The proposed MEGC typically addresses only one or two types of noise, while label noise, structure noise and feature noise often co-occur in practical applications. The accumulation of multiple types of noise makes it challenging for MEGC to effectively handle such complex noise scenarios. In future works, we will try to solve the above-mentioned challenges from different perspectives and further propose effective LNR-GRL models to improve the generalization of the existing GNNs on graphs with the above diverse noise.

References

[1] C.-Y. Zhang, Z.-L. Yao, H.-Y. Yao, F. Huang, and C. L. P. Chen, "Dynamic representation learning via recurrent graph neural networks," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 53, no. 2, pp. 1284–1297, Feb. 2023.
[2] P. Pham, L. T. Nguyen, N. T. Nguyen, W. Pedrycz, U. Yun, and B. Vo, "ComGCN: Community-driven graph convolutional network for link prediction in dynamic networks," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 52, no. 9, pp. 5481–5493, Sep. 2022.

[3] Y. Huang, L. Feng, M. Li, Y. Wang, Z. Zhu, and K. C. Tan, "Fast vehicle routing via knowledge transfer in a reproducing kernel Hilbert space," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 53, no. 9, pp. 5404–5416, Sep. 2023.

[4] J. Gou, Y. Cheng, Y. Zhan, B. Yu, W. Ou, and Y. Zhang, "Graph convolutional networks with collaborative feature fusion for sequential recommendation," *IEEE Trans. Big Data*, vol. 11, no. 2, pp. 735–747, Apr. 2025.

[5] D. Arpit et al., "A closer look at memorization in deep networks," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 233–242.

[6] N. T. Hoang, C. J. Jin, and T. Murata, "Learning graph neural networks with noisy labels," in *Proc. Int. Conf. Learn. Represent. Workshop*, 2019, pp. 1–5.

[7] E. Dai, C. Aggarwal, and S. Wang, "NRGNN: Learning a label noise resistant graph neural network on sparsely and noisily labeled graphs," in *Proc. ACM SIGKDD Conf. Knowl. Disc. Data Min.*, 2021, pp. 227–236.

[8] A. Iscen, J. Valmadre, A. Arnab, and C. Schmid, "Learning with neighbor consistency for noisy labels," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 4672–4681.

[9] J. Jiang, J. Ma, and X. Liu, "Multilayer spectral–spatial graphs for label noisy robust hyperspectral image classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 2, pp. 839–852, Feb. 2022.

[10] S. Qian et al., "Robust training of graph neural networks via noise governance," in *Proc. ACM Int. Conf. Web Search Data Min.*, 2023, pp. 607–615.

[11] J. Yuan, X. Luo, Y. Qin, Y. Zhao, W. Ju, and M. Zhang, "Learning on graphs under label noise," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2023, pp. 1–5.

[12] Y. Liu et al., "Multi-teacher self-training for semi-supervised node classification with noisy labels," in *Proc. ACM Int. Conf. Multimedia*, 2023, pp. 2946–2954.

[13] J. Gou, N. Xie, Y. Yuan, L. Du, W. Ou, and Z. Yi, "Reconstructed graph constrained auto-encoders for multi-view representation learning," *IEEE Trans. Multimedia*, vol. 26, pp. 1319–1332, 2023.

[14] M. Li et al., "Guest editorial: Deep neural networks for graphs: Theory, models, algorithms, and applications," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 4, pp. 4367–4372, Apr. 2024.

[15] S. Fu et al., "Finding core labels for maximizing generalization of graph neural networks," *Neural Netw.*, vol. 180, Dec. 2024, Art. no. 106635.

[16] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–12.

[17] B. Xu, J. Huang, L. Hou, H. Shen, J. Gao, and X. Cheng, "Label-consistency based graph neural networks for semi-supervised node classification," in *Proc. ACM SIGIR Int. Conf. Res. Develop. Inf. Retrieval*, 2020, pp. 1897–1900.

[18] Y. Yan, W. Ren, X. Hu, K. Li, H. Shen, and X. Cao, "SRGAT: Single image super-resolution with graph attention network," *IEEE Trans. Image Process.*, vol. 30, pp. 4905–4918, 2021.

[19] H. Zhou, Y. Yang, T. Luo, J. Zhang, and S. Li, "A unified deep sparse graph attention network for scene graph generation," *Pattern Recognit.*, vol. 123, Mar. 2022, Art. no. 108367.

[20] S. Brody, U. Alon, and E. Yahav, "How attentive are graph attention networks?" in *Proc. Int. Conf. Learn. Represent.*, 2022, pp. 1–26.

[21] F. Wu, A. Souza, T. Zhang, C. Fifty, T. Yu, and K. Weinberger, "Simplifying graph convolutional networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6861–6871.

[22] Y. Feng, H. You, Z. Zhang, R. Ji, and Y. Gao, "Hypergraph neural networks," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 3558–3565.

[23] S. Fu, W. Liu, K. Zhang, and Y. Zhou, "Example-feature graph convolutional networks for semi-supervised classification," *Neurocomputing*, vol. 461, pp. 63–76, Oct. 2021.

[24] C. Huang, M. Li, F. Cao, H. Fujita, Z. Li, and X. Wu, "Are graph convolutional networks with random weights feasible?" *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 3, pp. 2751–2768, Mar. 2023.

[25] J. Li, R. Zheng, H. Feng, M. Li, and X. Zhuang, "Permutation equivariant graph framelets for heterophilous graph learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 9, pp. 11634–11648, Sep. 2024.

[26] R. Wang et al., "Graph structure estimation neural networks," in *Proc. Web Conf.*, 2021, pp. 342–353.

[27] Y. Zhu, W. Xu, J. Zhang, Q. Liu, S. Wu, and L. Wang, "Deep graph structure learning for robust representations: A survey," 2021, *arXiv:2103.03036*.

[28] P. Veličković, L. Buesing, M. Overlan, R. Pascanu, O. Vinyals, and C. Blundell, "Pointer graph networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 2232–2244.

[29] W. Liu, S. Fu, Y. Zhou, Z.-J. Zha, and L. Nie, "Human activity recognition by manifold regularization based dynamic graph convolutional networks," *Neurocomputing*, vol. 444, pp. 217–225, Jul. 2021.

[30] V. P. Dwivedi, A. T. Luu, T. Laurent, Y. Bengio, and X. Bresson, "Graph neural networks with learnable structural and positional representations," in *Proc. Int. Conf. Learn. Represent.*, 2022, pp. 1–25.

[31] Y. Liu, Y. Zheng, D. Zhang, H. Chen, H. Peng, and S. Pan, "Towards unsupervised deep graph structure learning," in *Proc. ACM Web Conf.*, 2022, pp. 1392–1403.

[32] P. Zhang et al., "BrainUSL: Unsupervised graph structure learning for functional brain network analysis," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assisted Intervent.*, 2023, pp. 205–214.

[33] X. Li, X. Chen, B. Tang, S. Wang, Y. Xuan, and Z. Zhao, "Unsupervised graph structure-assisted Personalized federated learning," in *Proc. Eur. Conf. Artif. Intell.*, 2023, pp. 1430–1438.

[34] S. Fu et al., "Multilevel contrastive graph masked autoencoders for unsupervised graph-structure learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 36, no. 2, pp. 3464–3478, Feb. 2025.

[35] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–14.

[36] J. Yuan, H. Yu, M. Cao, J. Song, J. Xie, and C. Wang, "Self-supervised robust graph neural networks against noisy graphs and noisy labels," *Appl. Intell.*, vol. 53, no. 21, pp. 25154–25170, 2023.

[37] H. Song, M. Kim, D. Park, Y. Shin, and J.-G. Lee, "Learning from noisy labels with deep neural networks: A survey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 11, pp. 8135–8153, Nov. 2023.

[38] J. Wu, J. He, and J. Xu, "Net: Degree-specific graph neural networks for node and graph classification," in *Proc. ACM SIGKDD Int. Conf. Knowl. Disc. Data Min.*, 2019, pp. 406–415.

[39] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How powerful are graph neural networks?" in *Proc. Int. Conf. Learn. Represent.*, 2019, pp. 1–17.

[40] S. Sukhbaatar, J. Bruna, M. Paluri, L. Bourdev, and R. Fergus, "Training convolutional networks with noisy labels," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–11.

[41] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2818–2826.

[42] G. Patrini, A. Rozza, A. Krishna Menon, R. Nock, and L. Qu, "Making deep neural networks robust to label noise: A loss correction approach," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1944–1952.

[43] Z. Zhang and M. Sabuncu, "Generalized cross entropy loss for training deep neural networks with noisy labels," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 8792–8802.

[44] B. Han et al., "Co-teaching: Robust training of deep neural networks with extremely noisy labels," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 8536–8546.

[45] Y. Wang, X. Ma, Z. Chen, Y. Luo, J. Yi, and J. Bailey, "Symmetric cross entropy for robust learning with noisy labels," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 322–330.

[46] H. Wei, L. Feng, X. Chen, and B. An, "Combating noisy labels by agreement: A joint training method with co-regularization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 13726–13735.

[47] X. Ma, H. Huang, Y. Wang, S. Romano, S. Erfani, and J. Bailey, "Normalized loss functions for deep learning with noisy labels," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 6543–6553.

[48] X. Yu, B. Han, J. Yao, G. Niu, I. Tsang, and M. Sugiyama, "How does disagreement help generalization against label corruption?" in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 7164–7173.

[49] Q. Li, Z. Han, and X.-M. Wu, "Deeper insights into graph convolutional networks for semi-supervised learning," in *Proc. AAAI Conf. Artif. Intell.*, 2018, pp. 3538–3545.

[50] M. Zhang, L. Hu, C. Shi, and X. Wang, "Adversarial label-flipping attack and defense for graph neural networks," in *Proc. IEEE Int. Conf. Data Min.*, 2020, pp. 791–800.