Prototype-Guided Supervision for Graph Learning with Noisy and Sparse Labels

Qiyu Li[1,2,3], Xianxian Li[1,2,3*], De Li[1,4], Jinyan Wang[1,2,3*]

[1]Key Lab of Education Blockchain and Intelligent Technology, Ministry of Education, Guangxi Normal University, Guilin, 541004, China

[2]Guangxi Key Lab of Multi-Source Information Mining and Security, Guangxi Normal University, Guilin, 541004, China

[3]University Engineering Research Center of Educational Intelligent Technology, Guangxi Normal University, Guilin, 541004, China

[4]School of Physical Science and Technology, Guangxi Normal University, Guilin, 541004, China

qyl029@stu.gxnu.edu.cn, lixx@gxnu.edu.cn, lide@stu.gxnu.edu.cn, wangjy612@gxnu.edu.cn

Abstract

Graph learning faces major challenges under noisy and sparse supervision, where corrupted labels mislead representation learning and impair generalization. Prior work proposes robust training strategies such as correction, reweighting, and denoising to reduce the influence of noisy labels. However, most methods still optimize directly on training nodes using their possibly corrupted labels as supervision signals. In this work, we propose a prototype-guided framework that replaces direct label supervision over training nodes with semantic supervision derived from class-level prototypes. Each prototype is formed by aggregating representations of nodes sharing the same observed label and serves as a semantic anchor for guiding the classifier. To address the inherent supervision sparsity introduced by limited prototype instances, we introduce a dual-branch mixup strategy that integrates prototypes with high-confidence nodes through intra- and inter-class interpolation, which enhances supervision coverage and improves representation continuity. We further constrain the spatial variance of these samples to promote intra-class compactness. Theoretically, we demonstrate that the constructed prototypes remain aligned with true class semantics under bounded noise rates. Experiments on node classification tasks confirm the effectiveness of our approach under label noise and limited supervision.

Introduction

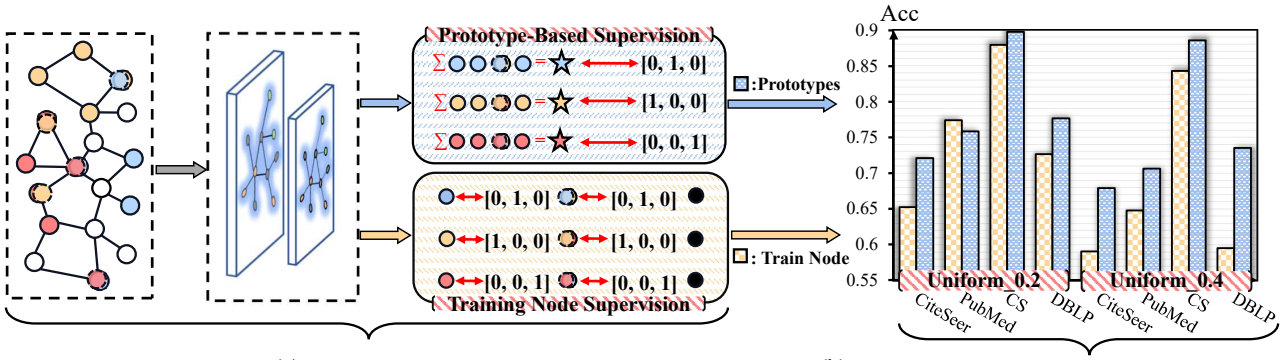
Graph-structured data pervade a wide range of real-world applications (Fan et al. 2019; He et al. 2025). These data encode both node-specific attributes and relational dependencies, which often serve as complementary sources of information. Effectively integrating node attributes with relational structures presents a central challenge in graph-based learning. Graph neural networks (GNNs) offer a principled solution by jointly modeling feature information and graph topology within a unified framework. Most GNNs adopt the message-passing paradigm, where each node iteratively updates its representation by aggregating information from its local neighborhood. This approach has demonstrated strong empirical performance across a wide range of graph learning tasks (Veličković et al. 2018; He et al. 2024; Zhang and Chen 2018; Xu et al. 2019; Cheng et al. 2025).

Despite their empirical success, most GNN architectures implicitly assume access to abundant and correct supervision. In practice, however, real-world graphs often suffer from label sparsity and noise, arising from annotation costs, subjective inconsistencies, or adversarial perturbations (Dai, Aggarwal, and Wang 2021). For instance, social networks typically label only a small fraction of users, and even those labels may be inconsistent or incorrect (Li, Wang, and Chen-Chuan 2012). These conditions result in weak supervision, where noisy labels mislead optimization and sparse annotations constrain semantic coverage. Label noise and sparsity jointly pose persistent challenges in graph learning.

To address the challenges of noisy and sparse supervision in node classification, a range of strategies has been developed. Some methods focus on loss correction by modifying the learning objective to account for noisy labels (Han et al. 2018; Patrini et al. 2017; Du et al. 2021), while others reduce the influence of label noise through denoising mechanisms or alternative training procedures (Chen et al. 2024; Ding et al. 2024). Structural augmentation approaches refine the graph topology to prevent noise amplification during message passing (Dai, Aggarwal, and Wang 2021; Qian et al. 2023). Some methods enhance representation robustness by applying consistency constraints and contrastive learning techniques. (Yuan et al. 2023; Li et al. 2024b). Curriculum-based training further mitigates noise by progressively selecting high-confidence samples based on structural cues (Wu et al. 2024). Although various strategies have been proposed to mitigate label noise, such as filtering unreliable nodes or correcting corrupted annotations, most methods still rely on direct supervision from training nodes and their potentially corrupted labels. Whether applied to the entire training set or a selected subset assumed to be clean, supervision remains closely coupled with signals of uncertain reliability. Under noisy conditions, this reliance may introduce misleading gradients that destabilize optimization and impair generalization.

To shift the supervision mechanism away from direct reliance on training labels, we introduce a prototype-guided learning framework that redefines the source of supervision. As illustrated in Fig. 1, our method discards direct supervision on training nodes and instead constructs semantic prototypes to guide representation learning. Each prototype aggregates representations from nodes sharing the same ob-

*Xianxian Li and Jinyan Wang are corresponding authors.



(a) Supervision Paradigms (b) Accuracy under Different Supervision Paradigms

Figure 1: Illustration and comparison of two supervision paradigms. (a) Training node supervision directly uses observed labels of training nodes as learning targets, whereas prototype-based supervision discards direct label supervision and instead uses prototypes as semantic guidance. (b) Empirical results under uniform label noise show that prototype-based supervision achieves superior performance in most cases across datasets and noise levels.

served label, thereby providing a stable and noise-tolerant semantic reference. To address the limited representational coverage caused by supervision sparsity, we further design a dual-branch interpolation strategy that integrates prototypes with high-confidence nodes via both intra-class and inter-class mixup. This design enriches the supervision signal and promotes smoother semantic transitions in the representation space. Extensive experiments on benchmark datasets under various noise settings validate the effectiveness of our framework in improving robustness and generalization.

In summary, we list our contributions as follows:

Supervision paradigm. We propose to replace direct label supervision on training nodes with prototype-based semantic guidance, reducing sensitivity to label noise and supervision sparsity.

Methodology. We propose a prototype-guided framework that constructs semantic prototypes and combines them with high-confidence nodes via dual-branch mixup. To improve class separability and compactness, we introduce a prototype orthogonality constraint and a radius regularization term.

Comprehensive validation. Experiments across multiple benchmarks and noise levels confirm the effectiveness and generalization of the proposed approach.

Related Work

Label Noise in Graphs

Label noise poses a critical challenge to graph learning, as unreliable supervision disrupts representation learning and degrades model generalization. Classical techniques such as loss correction (Patrini et al. 2017; Goldberger and Ben-Reuven 2017) and sample selection (Han et al. 2018; Jiang et al. 2018) perform well in computer vision (Han, Mao, and Dally 2016; Louizos, Welling, and Kingma 2018), but often struggle on graphs due to sparse and interdependent labels (Kudugunta and Ferrara 2018). Existing solutions for noisy graph learning vary by task setting. For graph classification, recent methods improve robustness through loss adjustment (Hoang, Jin, and Murata 2019), neighbor-aware denoising (Yin et al. 2023), and dual-space correction (Li et al. 2025). For node classification, approaches typically refine noisy supervision via local structural signals. Representative strategies include confident node filtering (Xia et al. 2023; Cheng et al. 2024), label propagation (Chen et al. 2024), propagation-free architectures to block noise amplification (Ding et al. 2024), reweighting and correction based on soft label aggregation (Li, Yin, and Chen 2021), pair consistency constraints (Du et al. 2021), and structural augmentation (Dai, Aggarwal, and Wang 2021; Qian et al. 2023). Contrastive methods (Yuan et al. 2023; Li et al. 2024b; Zhu et al. 2024; Li et al. 2024a) enhance representation robustness through consistency constraints, while curriculum-based sampling (Wu et al. 2024) progressively selects reliable nodes based on topological difficulty. Despite their success, most methods rely on observed training labels as direct supervision. Our framework instead introduces prototype-guided alignment to mitigate label noise and enhance semantic consistency.

Preliminaries

Notations

We consider node classification on a static attributed graph with sparse and noisy labels. Formally, the graph is defined as $G = (V, E, X, Y)$, where $V = \{v_1, \ldots, v_N\}$ is the set of nodes, $E \subseteq V \times V$ is the set of undirected edges, and $X \in \mathbb{R}^{N \times D}$ is the node feature matrix, where $\mathbf{x}_i \in \mathbb{R}^D$ denotes the feature vector of node v_i. The label matrix $Y \in \{0,1\}^{N \times C}$ provides one-hot representations of the ground-truth labels. The graph connectivity is represented by an adjacency matrix $A \in \{0,1\}^{N \times N}$, where $A_{ij} = 1$ indicates an edge between v_i and v_j. For simplicity, we denote the graph as $G = (A, X)$. To simulate the limited supervision setting, the node set V is partitioned into a labeled subset V^L and an unlabeled subset $V^U = V \setminus V^L$, where $|V^L| \ll N$. Each labeled node $v_i \in V^L$ is associated with a possibly corrupted label $\tilde{y}_i \in \{0, \ldots, C-1\}$, and clean labels y_i are reserved for evaluation only.

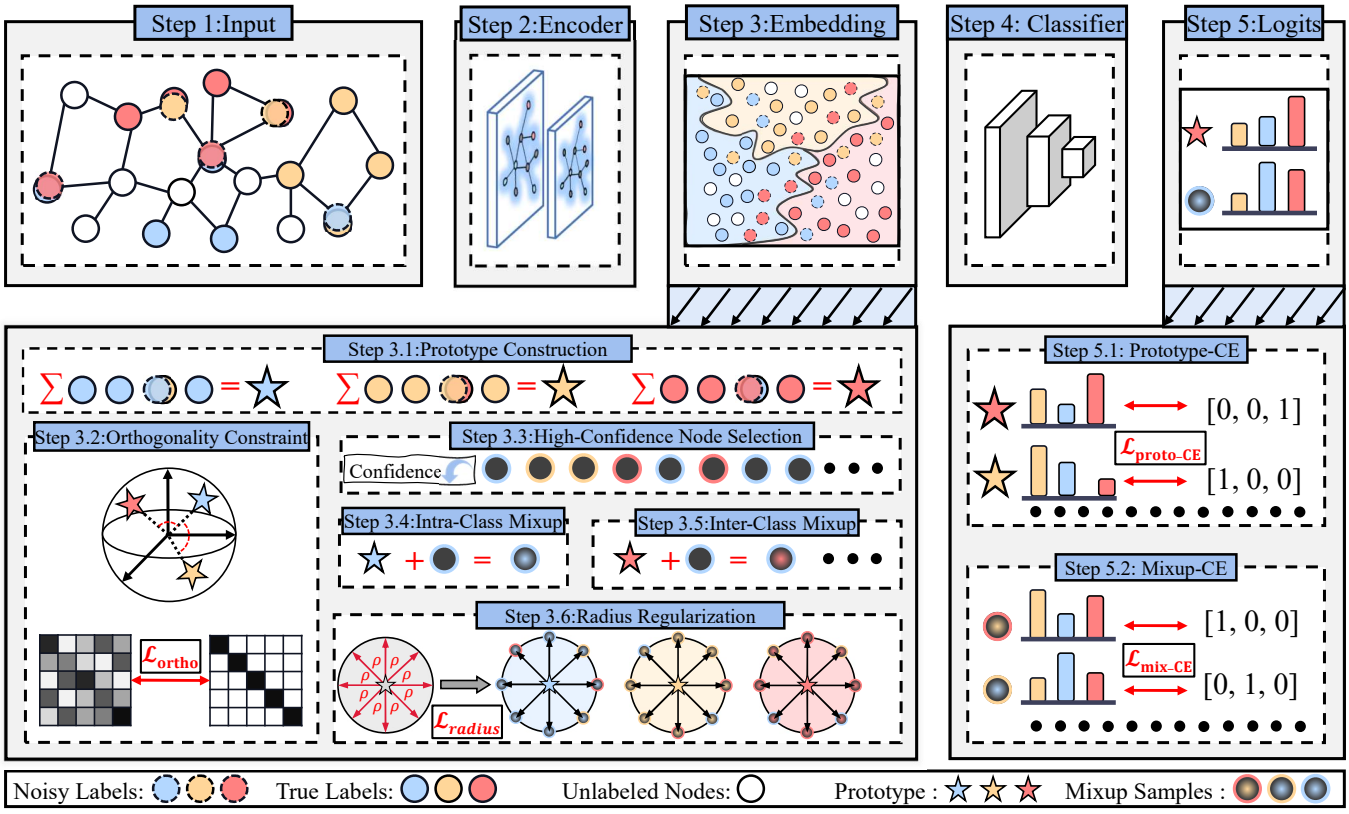


Figure 2: Illustration of the proposed framework. A GNN encoder extracts node embeddings, from which prototypes are constructed and constrained to be orthogonal. High-confidence nodes are selected and mixed with prototypes through intra-class and inter-class mixup. The classifier is trained using both prototype-based cross-entropy and mixup-based cross-entropy losses.

Problem Formulation

Given the attributed graph and noisy supervision setting defined above, the goal is to learn a node classification model that remains effective under label corruption and supervision sparsity. We adopt a decoupled architecture, where the encoder f_θ generates structure-aware node embeddings:

$$f_\theta : (A, X) \to Z = [z_1; \ldots; z_N] \in \mathbb{R}^{N \times d}, \tag{1}$$

and the classifier g_ϕ maps each embedding to a probability distribution over C classes:

$$p_i = \mathrm{softmax}(g_\phi(z_i)) \in \Delta^{C-1}. \tag{2}$$

The model is trained on the noisy labeled set $\mathcal{D}_{\text{train}} = \{(x_i, \tilde{y}_i) \mid v_i \in V^L\}$, and is expected to generalize to unlabeled nodes despite the presence of corrupted labels.

Methodology

Our framework is shown in Fig. 2. We next describe its main components.

Prototype Construction and Optimization

Motivation: Existing approaches often rely on denoising or reweighting strategies to alleviate the adverse impact of corrupted training labels. However, these methods remain inherently coupled to noisy annotations, leaving models susceptible to residual label errors. In this work, we depart from direct supervision based on training labels, and instead construct class-level prototypes as semantic anchors. These prototypes serve as the basis for generating high-quality virtual training signals, enabling instance-level learning without exposing the model to corrupted label supervision.

The encoder f_θ produces node representations $Z = f_\theta(A, X)$. For each class $c \in \{0, \ldots, C-1\}$, we define $\mathcal{I}_c = \{i \mid \tilde{y}_i = c\}$ as the index set of labeled nodes, and compute the class prototype by averaging their embeddings:

$$\mu_c = \frac{1}{|\mathcal{I}_c|} \sum_{i \in \mathcal{I}_c} z_i, \qquad \hat{\mu}_c = \frac{\mu_c}{\|\mu_c\|_2}. \tag{3}$$

Each prototype $\hat{\mu}_c$ serves as a semantic representative for class c, and is later used for class-level supervision. To avoid direct supervision from noisy node labels, we instead train the classifier to predict the correct class for each prototype. Specifically, each normalized prototype $\hat{\mu}_c$ is passed through the classifier g_ϕ to produce a class probability vector:

$$p_c^{\text{proto}} = \mathrm{softmax}(g_\phi(\hat{\mu}_c)). \tag{4}$$

We minimize the cross-entropy between the predicted distribution and its assigned class label:

$$\mathcal{L}_{\text{proto\_CE}} = -\frac{1}{C} \sum_{c=0}^{C-1} \log \left(p_c^{\text{proto}}[c] \right). \tag{5}$$

This loss drives the classifier to produce discriminative predictions based solely on prototype-level semantics, thereby avoiding direct reliance on noisy training labels.

Theorem 1. Prototype Correctness under Label Noise. *Let $C \geq 2$ be the number of classes, and $\eta \in [0, 1)$ the label noise rate. Under the uniform noise model, the prototype μ_c aligns with the true class semantics with high probability as the number of nodes increases, provided that $\eta < 1 - \frac{1}{C}$. Similarly, under the pair-flip noise model, correctness holds when $\eta < \frac{1}{2}$.*

Prototype Orthogonality Constraint

Motivation: Prototypes aggregated from noisy-labeled nodes may inherit semantic corruption and exhibit directional overlap, especially when mislabeled samples occur across classes. This weakens their discriminability and limits their supervisory value. To address this, we impose prototype orthogonality to maximize class separation, suppress cross-class interference, and strengthen the reliability of prototype supervision.

To regularize the geometry of the prototype space, we introduce an orthogonality constraint that encourages each class prototype to lie in a distinct semantic direction. Given the normalized prototype set $\{\hat{\mu}_c\}_{c=0}^{C-1}$, we compute the Gram matrix $G \in \mathbb{R}^{C \times C}$ as:

$$G_{ij} = \langle \hat{\mu}_i, \hat{\mu}_j \rangle. \tag{6}$$

Ideally, G should approximate the identity matrix, indicating that all prototypes are mutually orthogonal. We thus define the orthogonality loss as the squared Frobenius norm deviation:

$$\mathcal{L}_{\text{ortho}} = \|G - I_C\|_F^2. \tag{7}$$

This constraint encourages a well-separated prototype space, suppresses label-induced alignment noise, and promotes a stable geometric structure for effective classification.

Prototype Mixup

Motivation: While prototypes are established across all C classes, each is supervised through a single, fixed class target, resulting in an extremely sparse supervision signals. This one-to-one mapping between prototype and label severely limits the density and diversity of gradient signals, increasing the risk of overfitting and impairing the learning of discriminative decision boundaries. To alleviate this issue, we introduce mixup-based supervision (Zhang et al. 2018) by interpolating each prototype with high-confidence node embeddings. This construction yields a semantically meaningful spectrum of mixup-based training signals, enhancing both robustness and generalization.

Let $p_i = \text{softmax}(g_\phi(z_i))$ denote the predicted class probability for node v_i, and define its confidence score as $\text{conf}_i = \max_c p_i[c]$. We select the top-$M$ nodes with the highest confidence:

$$\mathcal{J} = \text{TopK}\left(\{\text{conf}_i\}_{i=1}^{N},\ M\right), \tag{8}$$

and assign pseudo-labels by $\hat{y}_j = \arg\max_c p_j[c]$ for each $j \in \mathcal{J}$.

To enrich the intra-class supervision manifold, each high-confidence embedding z_j is interpolated with its predicted class prototype $\mu_{\hat{y}_j}$. We define the distance-aware mixing coefficient as:

$$\lambda_j^{\text{intra}} = \text{clip}\left(\lambda_j^{\text{base}} \cdot \sigma(-\|z_j - \mu_{\hat{y}_j}\|_2/\tau),\ 0,\ 1\right), \tag{9}$$

where $\lambda_j^{\text{base}} \sim \text{Beta}(\alpha, \alpha)$ and $\sigma(\cdot)$ denotes the sigmoid function. We use $\text{clip}(x, a, b)$ to denote the element-wise truncation of x within the closed interval $[a, b]$.

The interpolated feature and its target label are computed as:

$$h_j^{\text{intra}} = \lambda_j^{\text{intra}} z_j + (1 - \lambda_j^{\text{intra}})\, \mu_{\hat{y}_j}, \quad \mathbf{y}_j^{\text{intra}} = \text{onehot}(\hat{y}_j). \tag{10}$$

To further enhance decision boundary sensitivity, we incorporate inter-class mixup by interpolating each high-confidence node with a semantically close non-target prototype. For each $j \in \mathcal{J}$, we compute the Euclidean distance between z_j and all class prototypes:

$$D_j(c) = \|z_j - \mu_c\|_2, \quad \forall\, c \in \{0, \ldots, C-1\},\ c \neq \hat{y}_j. \tag{11}$$

We identify the top-k nearest non-target classes and uniformly sample one as the competing class c_j^{comp}. Then, the mixed feature and its soft target are defined as:

$$\lambda_j^{\text{inter}} \sim \text{Beta}(\alpha, \alpha), \quad h_j^{\text{inter}} = \lambda_j^{\text{inter}} z_j + (1 - \lambda_j^{\text{inter}})\, \mu_{c_j^{\text{comp}}}, \tag{12}$$

$$\mathbf{y}_j^{\text{inter}} = \lambda_j^{\text{inter}} \,\text{onehot}(\hat{y}_j) + (1 - \lambda_j^{\text{inter}})\,\text{onehot}(c_j^{\text{comp}}). \tag{13}$$

We aggregate all mixup-based samples:

$$\mathcal{S}_{\text{mix}} = \left\{(h_j^{\text{intra}},\ \mathbf{y}_j^{\text{intra}}),\ (h_j^{\text{inter}},\ \mathbf{y}_j^{\text{inter}}) \mid j \in \mathcal{J}\right\}, \tag{14}$$

and define the overall mixup loss as:

$$\mathcal{L}_{\text{mix\_CE}} = \frac{1}{2M} \sum_{(h,\, \mathbf{y}) \in \mathcal{S}_{\text{mix}}} \text{CE}(\mathbf{y},\, \text{softmax}(g_\phi(h))). \tag{15}$$

By aligning interpolated representations with soft targets, this objective complements sparse prototype supervision and promotes smoother decision boundaries.

Radius Regularization

Motivation: While mixup augmentation improves representation diversity and alleviates prototype sparsity, it also introduces a risk of semantic deviation, particularly under noisy supervision. Without explicit structural constraints, mixup-based samples may populate the latent space in an uneven or overly dispersed manner, leading to degraded intra-class cohesion and ambiguous decision boundaries. To remedy this, we introduce a radius regularization term that encourages each class to maintain a consistent spread of interpolated samples around its prototype. This encourages intra-class compactness and promotes global geometric consistency in the representation space.

Each mixup sample $(h_j, \mathbf{y}_j) \in \mathcal{S}_{\text{mix}}$ is assigned to its dominant class $\widetilde{y}_j = \arg\max_c \mathbf{y}_j[c]$. For each class c, let $\mathcal{J}_c = \{j \in \mathcal{J} \mid \widetilde{y}_j = c\}$. The class-wise radius quantifies intra-class dispersion around prototype μ_c:

$$\sigma_c = \left(\frac{1}{|\mathcal{J}_c|} \sum_{j \in \mathcal{J}_c} \|h_j - \mu_c\|_2^2\right)^{1/2}. \tag{16}$$

Dataset	Methods	Clean	Uniform Noise 0.2	Uniform Noise 0.4	Uniform Noise 0.6	Pair Noise 0.2	Pair Noise 0.3	Pair Noise 0.4
	MLP	59.15±1.38	50.58±1.89	41.37±2.18	28.21±2.10	50.86±2.05	46.60±2.20	40.01±2.10
	GCN	83.84±1.23	77.58±1.94	73.37±3.33	51.82±3.08	82.40±2.00	76.16±2.54	67.33±2.99
	Coteaching	80.92±1.43	75.51±2.87	67.61±2.84	51.03±4.03	75.71±2.07	70.54±3.71	63.28±4.33
	Forward	83.72±1.39	75.92±2.20	71.30±2.90	48.43±3.32	81.06±1.32	73.84±4.31	65.56±3.11
	NRGNN	81.85±1.49	73.32±2.43	68.66±3.31	46.57±3.77	79.67±1.61	72.96±2.72	64.58±3.18
	RTGNN	81.09±2.09	78.96±2.79	71.19±3.50	48.59±4.84	81.79±1.33	76.06±2.68	66.87±4.14
Cora_ML	CGNN	83.30±1.59	76.59±2.61	66.79±3.41	46.12±4.17	77.76±2.08	72.10±2.97	63.92±3.92
	CRGNN	82.16±1.59	77.02±2.52	66.79±2.50	56.69±5.90	78.04±2.57	73.36±3.94	65.13±4.39
	PIGNN	82.46±1.48	80.03±1.39	76.30±2.86	59.70±4.35	80.50±1.94	76.74±2.03	66.83±3.46
	ERASE	81.63±1.14	80.01±1.51	78.00±2.45	65.49±6.07	82.04±1.42	80.44±1.73	72.92±3.95
	DNDNet	85.28±1.20	83.84±1.19	78.26±2.91	63.40±8.25	84.39±1.07	80.24±2.01	74.74±3.37
	OURS	**85.57±0.80**	**84.17±0.99**	**82.18±1.93**	**70.04±4.93**	**84.67±1.18**	**83.20±1.71**	**76.89±4.46**
	MLP	55.95±2.53	48.28±2.77	43.21±3.03	30.09±3.16	49.79±2.34	44.97±2.93	43.23±2.58
	GCN	71.46±1.21	65.22±2.15	59.03±3.31	41.14±4.47	66.10±2.88	61.44±3.42	59.42±3.63
	Coteaching	68.80±1.90	63.67±2.48	57.57±3.56	42.33±4.41	63.98±2.91	59.09±3.57	58.11±3.23
	Forward	72.25±1.34	66.36±2.16	60.17±3.66	42.20±4.47	67.68±1.70	63.48±2.66	62.03±3.12
	NRGNN	69.85±1.43	62.91±2.29	55.33±3.36	37.79±3.43	63.60±2.76	59.58±3.58	57.51±3.26
	RTGNN	74.28±1.41	68.89±2.50	61.36±4.02	44.69±4.26	68.45±2.48	64.23±3.31	62.89±3.87
CiteSeer	CGNN	69.64±3.74	64.59±3.82	58.28±3.32	41.27±4.54	64.93±3.66	59.32±4.86	58.33±6.57
	CRGNN	71.85±1.37	68.48±2.53	63.43±3.65	44.67±5.83	68.42±3.00	64.39±3.57	62.04±4.23
	PIGNN	70.63±1.67	67.55±2.43	60.75±3.94	44.67±6.90	65.66±2.85	61.74±3.01	59.30±4.39
	ERASE	72.73±1.66	71.88±2.36	69.24±5.33	55.10±7.01	72.20±2.15	70.65±2.78	67.70±5.38
	DNDNet	75.71±0.94	73.58±1.35	69.19±3.89	58.38±6.53	73.60±1.53	70.78±2.89	70.22±2.44
	OURS	**75.95±1.01**	**75.44±0.96**	**73.15±1.60**	**60.08±7.28**	**75.14±1.21**	**74.03±1.16**	**72.21±2.74**
	MLP	74.05±1.27	61.88±2.06	49.10±1.78	42.88±2.28	61.33±1.60	56.46±1.81	53.75±1.43
	GCN	81.81±0.79	77.41±1.86	64.76±3.68	52.60±5.45	75.50±2.14	71.73±3.36	67.51±3.31
	Coteaching	80.67±1.49	75.64±2.35	60.31±5.25	51.14±5.53	74.10±3.14	69.06±4.29	67.21±3.42
	Forward	82.37±0.74	77.58±1.52	65.49±3.25	53.31±5.79	75.44±2.34	71.56±3.26	67.49±2.97
	NRGNN	81.29±0.69	77.39±1.60	66.37±3.12	53.44±6.12	75.20±2.17	71.41±3.20	67.25±3.13
	RTGNN	76.55±1.61	77.80±1.57	70.57±2.20	58.46±5.32	76.70±1.88	74.21±2.61	72.29±2.94
PubMed	CGNN	81.40±0.58	75.00±8.36	59.64±7.06	49.62±5.53	73.22±9.52	69.38±7.86	66.39±6.87
	CRGNN	**83.23±0.66**	79.38±1.48	70.20±2.90	58.66±7.45	76.33±2.44	71.58±3.23	67.45±2.35
	PIGNN	81.59±0.81	79.39±1.51	72.16±2.70	57.72±8.38	77.06±1.86	73.36±2.47	69.78±2.94
	ERASE	78.51±1.79	76.42±3.34	73.43±3.34	56.59±6.47	74.97±1.48	72.75±2.56	70.46±2.53
	DNDNet	81.46±0.84	79.37±2.19	73.66±3.11	60.01±9.02	77.81±1.58	75.72±2.78	74.24±2.73
	OURS	82.32±0.94	**80.57±1.33**	**74.44±3.03**	**60.06±5.80**	**79.73±1.71**	**76.31±2.01**	**75.19±3.12**
	MLP	80.55±2.28	75.09±2.36	64.01±3.68	51.65±4.03	73.11±3.24	61.89±4.19	61.42±3.92
	GCN	91.48±0.45	88.00±1.20	84.34±1.59	73.38±3.34	86.53±0.98	79.73±3.08	77.19±2.01
	Coteaching	89.95±0.56	86.83±0.83	79.75±1.74	67.67±3.77	85.98±1.37	78.87±2.51	76.26±2.87
	Forward	86.11±6.56	74.91±3.49	65.87±9.84	58.55±5.83	75.07±8.73	63.02±8.91	66.18±7.71
	NRGNN	90.83±0.56	87.70±1.60	85.09±1.30	71.81±2.51	86.02±1.40	77.45±4.30	77.44±2.10
	RTGNN	89.54±0.78	87.70±1.47	82.83±2.13	74.92±3.51	84.48±1.98	75.55±4.19	74.05±3.22
CS	CGNN	89.24±2.57	86.08±2.29	78.78±3.26	64.39±5.34	83.82±3.39	77.14±4.52	74.77±2.90
	CRGNN	90.72±0.52	86.83±1.70	81.10±2.48	69.36±4.29	84.81±2.09	76.57±5.40	74.74±4.99
	PIGNN	89.86±0.79	88.02±1.33	85.09±1.30	73.12±3.12	86.07±1.11	79.08±4.14	76.63±3.01
	ERASE	90.15±0.78	89.94±0.64	84.53±1.92	84.14±4.23	88.91±1.14	88.60±1.57	84.42±2.10
	DNDNet	91.78±0.34	90.44±0.92	87.09±4.43	75.45±2.81	89.39±1.05	87.13±1.80	80.66±3.46
	OURS	**92.03±0.75**	**91.59±0.79**	**90.60±1.28**	**84.29±2.65**	**90.66±1.69**	**89.98±1.33**	**86.29±3.51**
	MLP	54.93±3.10	47.92±3.55	39.71±4.08	32.00±4.04	49.19±3.76	44.57±4.02	39.31±4.86
	GCN	78.88±1.20	72.65±1.62	59.49±3.44	34.67±3.93	74.04±2.44	68.57±2.30	55.25±4.62
	Coteaching	75.10±1.59	67.01±3.09	54.07±3.27	35.46±4.89	66.05±2.91	58.52±4.08	52.08±4.61
	Forward	78.45±1.36	72.10±1.64	59.14±3.20	34.14±3.44	73.18±2.86	67.62±2.50	54.42±4.25
	NRGNN	77.94±1.74	71.36±2.19	57.47±3.65	33.62±3.74	72.54±2.87	66.43±2.88	52.85±4.78
	RTGNN	80.75±1.02	76.34±1.19	62.59±6.35	35.44±5.32	75.49±2.75	70.44±3.23	55.74±9.08
DBLP	CGNN	78.50±1.25	73.03±1.64	60.72±4.36	41.24±4.25	71.90±2.87	67.22±3.65	56.36±6.40
	CRGNN	81.05±1.04	76.35±1.51	68.15±4.24	40.43±5.69	74.64±2.54	69.79±3.21	59.69±5.75
	PIGNN	79.10±1.33	77.44±1.15	71.48±2.97	40.61±4.46	76.25±1.81	72.76±2.12	61.47±4.71
	ERASE	81.22±0.59	80.61±0.75	79.12±1.38	52.23±8.62	80.10±0.94	79.09±1.68	74.59±3.10
	DNDNet	81.20±1.19	79.11±1.79	54.77±15.80	39.82±11.84	79.70±1.27	77.85±2.65	66.08±13.10
	OURS	**82.01±1.47**	**81.71±0.69**	**80.01±1.24**	**58.22 ± 8.28**	**80.81±0.65**	**79.88±0.99**	**75.87±1.62**

Table 1: Node classification performance w.r.t. accuracy. The highest value in each column is highlighted in **bold**.

We define the radius regularization loss as:

$$\mathcal{L}_{\text{radius}} = \frac{1}{C}\sum_{c=0}^{C-1} |\sigma_c - \rho| \,. \tag{17}$$

Here, ρ denotes a hyperparameter that controls the target intra-class dispersion. This loss constrains mixup dispersion, enhancing intra-class compactness and representation discriminability.

The final training objective is formulated as:

$$\mathcal{L} = \mathcal{L}_{\text{proto\_CE}} + \mathcal{L}_{\text{mix\_CE}} + \gamma_{\text{ortho}}\mathcal{L}_{\text{ortho}} + \gamma_{\text{rad}}\mathcal{L}_{\text{radius}}, \tag{18}$$

where γ_{ortho} and γ_{rad} are hyperparameters controlling the strength of the regularization terms.

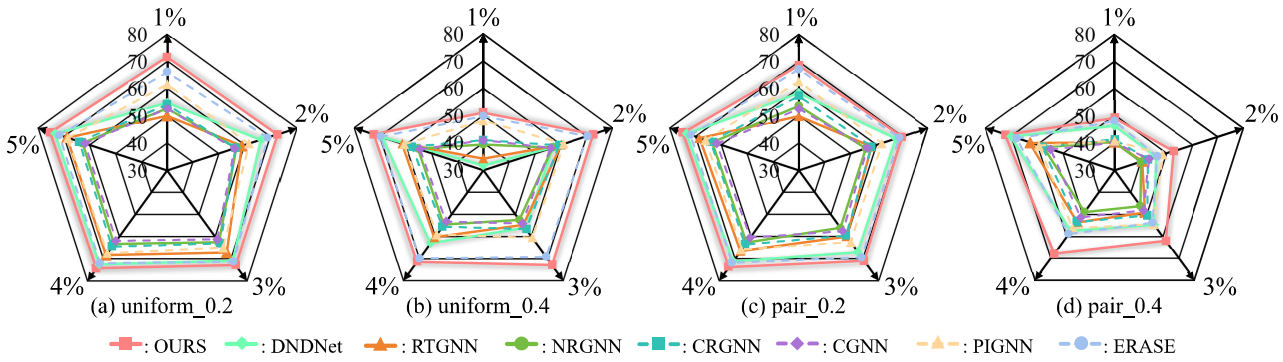


Figure 3: Node classification accuracy on the CiteSeer dataset under varying label rates (1%–5%) and different noise settings. Subfigures (a) and (b) present results under uniform noise with noise ratios 0.2 and 0.4, respectively, while (c) and (d) show performance under pair noise at the same corruption levels. Each pentagon represents the accuracy of a given method at different label rates.

Experiments

Experimental Setup

Datasets and Label Noise. We conduct experiments on six standard benchmarks: Cora_ML, CiteSeer, PubMed, CS, DBLP, and arxiv. Following (Dai, Aggarwal, and Wang 2021), we introduce two types of synthetic label noise: (1) *Uniform noise*, where labels are randomly flipped to any other class with probability η; (2) *Pair noise*, where each label is flipped to a fixed paired class with probability η. We evaluate model robustness under varying noise levels.

Baselines. To evaluate the effectiveness of our method, we compare it against a diverse set of baselines, including GCN (Kipf and Welling 2017), MLP, Coteaching (Han et al. 2018), Forward (Patrini et al. 2017), NRGNN (Dai, Aggarwal, and Wang 2021), RTGNN (Qian et al. 2023), PIGNN (Du et al. 2021), CGNN (Yuan et al. 2023), CRGNN (Li et al. 2024b), ERASE (Chen et al. 2024), and DNDNet (Ding et al. 2024).

Main Results

Methods	uniform		pair	
	0.2	0.4	0.2	0.4
GCN	54.55±1.10	49.28±1.95	55.74±1.02	49.10±2.46
PIGNN	51.24±1.87	45.78±3.51	51.24±2.77	42.60±5.14
DNDNet	38.87±0.82	34.89±2.76	41.54±0.93	39.47±0.40
OURS	**58.29±0.94**	**53.02±3.34**	**57.58±1.79**	**51.88±2.37**

Table 2: Performance comparison on arxiv under uniform and pair label noise

Overall Performance. We evaluate the proposed model under both clean and noisy supervision. As shown in Table 1, it achieves the highest accuracy on most datasets in the clean setting, indicating strong semantic discrimination without additional robustness modules. In noisy scenarios with uniform and pair corruption, the model consistently outperforms competitive baselines across all noise levels. This robustness is attributed to prototype-guided supervision, interpolation-based augmentation, and regularization, which together improve class separation and enhance generalization under corrupted supervision.

Low label rates Performance. Under both uniform and pair noise, the proposed method exhibits stable performance across varying label rates. As shown in Fig. 3, it remains robust under extremely sparse supervision (1%–3%), while baseline performance degrades significantly. This consistency across noise types and corruption levels suggests that the model can effectively leverage limited labeled data, highlighting its suitability for label-scarce graph scenarios.

Large-Scale dataset Performance. We evaluate GCN, PIGNN, DNDNet, and the proposed method on the arxiv dataset under uniform and pair label noise. As shown in Table 2, the proposed method yields relatively stable performance across noise types and levels, whereas other methods exhibit more pronounced variation under increasing corruption.

Parameter Analysis

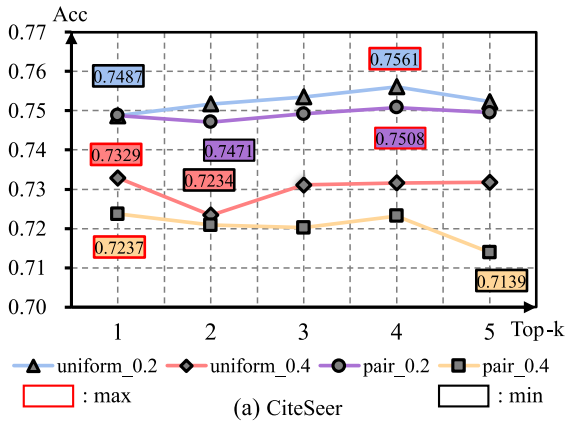


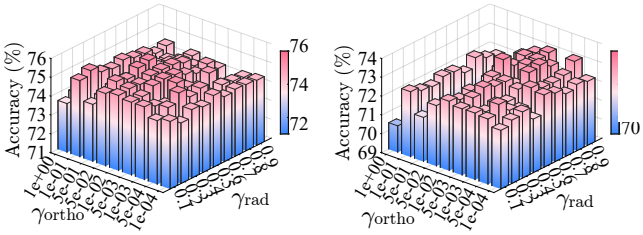
Figure 4: Accuracy variation under different k settings for top-k competing classes used in inter-class mixup.

top-k. We analyze the sensitivity of the inter-class mixup module to the hyperparameter k, which determines the num-

$\mathcal{L}_{\text{proto\_CE}}$	$\mathcal{L}_{\text{mix\_CE}}$	$\mathcal{L}_{\text{ortho}}$	$\mathcal{L}_{\text{radius}}$	Uniform Noise			Pair Noise		
				$\eta = 0.2$	$\eta = 0.4$	$\eta = 0.6$	$\eta = 0.2$	$\eta = 0.3$	$\eta = 0.4$
✓				72.23±1.39	68.94±2.24	49.34±5.70	71.46±2.10	67.58±2.12	64.31±3.46
✓	✓			74.47±1.74	71.94±3.10	54.16±7.03	74.98±1.56	70.53±3.50	68.54±3.33
✓		✓		72.72±1.44	68.72±2.06	53.01±4.78	72.91±1.31	69.45±1.87	67.53±3.13
✓	✓	✓		74.74±0.86	71.67±4.40	57.22±6.71	**75.35±0.86**	72.45±2.30	71.29±3.00
✓	✓		✓	74.57±1.21	72.99±2.56	54.69±7.19	74.03±1.63	71.54±5.33	68.91±5.33
✓	✓	✓	✓	**75.44±0.96**	**73.15±1.60**	**60.08±7.28**	75.14±1.21	**74.03±1.16**	**72.21±2.74**

Table 3: Ablation study of different loss components under varying noise settings on CiteSeer.

ber of competing classes. As shown in Fig. 4, the performance remains relatively stable across a range of k values, with peak accuracy typically observed around $k = 4$. Th suggests that introducing a moderate number of competi classes facilitates effective mixup generation, while exce sively small or large k may compromise alignment.



(a) uniform_0.2 (b) uniform_0.4

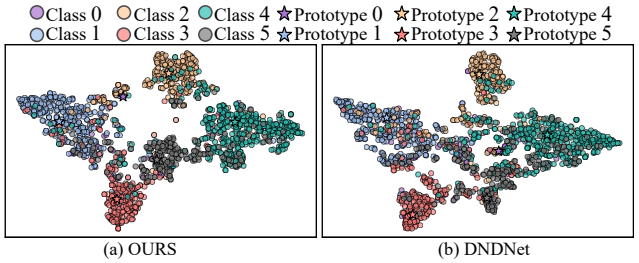
Figure 5: Accuracy variation under different γ_{ortho} and γ_{rad} across datasets.

γ_{ortho} and γ_{rad}. To assess the effect of structural regularization under label noise, we evaluate model performance across varying γ_{ortho} and γ_{rad} values, as illustrated in Fig. 5. Under a noise rate of 0.2, the model exhibits stable accuracy across a wide hyperparameter range, indicating low sensitivity to regularization strength. When the noise rate increases to 0.4, performance becomes more sensitive, with certain configurations leading to noticeable degradation. These results demonstrate the importance of appropriately balancing the two regularization terms to ensure robust learning under severe noise.

Ablation Study

We perform ablation studies to evaluate the contribution of each loss component under varying noise conditions. As shown in Table 3, removing individual terms from the overall objective consistently degrades performance, highlighting their complementary roles. The mixup-based supervision enhances the model's generalization under weak and corrupted labels, while the orthogonality constraint improves inter-class separability in the representation space. The radius regularization further promotes intra-class compactness, which is particularly beneficial under high noise. These results confirm that the proposed objective design improves the robustness and discriminability of graph representations in the presence of label noise.

Visualization Analysis



(a) OURS (b) DNDNet

Figure 6: t-SNE visualization of node representations on the CiteSeer dataset under 0.4 uniform label noise.

To qualitatively assess representation robustness under label noise, we visualize node embeddings using t-SNE (Van der Maaten and Hinton 2008) with 0.4 uniform noise. As shown in Fig. 6, our method forms more compact and well-separated clusters, with prototypes (stars) closely aligned with class centers. This suggests that prototype guidance effectively reduces the impact of noisy labels and sharpens class boundaries. In contrast, DNDNet produces dispersed and overlapping clusters, reflecting weakened discriminability. These results highlight the advantage of our method in learning clearer and more noise-resilient representations.

Conclusion

This work introduce a prototype-guided framework that shifts supervision away from loss on individual training node labels toward semantic anchors formed by aggregating representations of nodes with the same observed label. This design eliminates reliance on training node labels in the loss function. A dual-branch mixup strategy interpolates each prototype with high-confidence nodes both within the same label and across different labels, broadening supervision and smoothing the representation manifold. A variance regularizer further reduces dispersion among samples. We prove that prototypes remain faithful to true semantics under bounded noise rates and demonstrate consistent accuracy improvements on node classification benchmarks under diverse noise scenarios and scarce labels. Our results establish prototype-based semantic guidance as a robust alternative for graph learning in noisy, sparse environments and motivate future exploration of principled supervision methods.

Acknowledgments

This paper was supported by the National Natural Science Foundation of China (Nos. U21A20474 and 62162005), Guangxi "Bagui Scholar" Teams for Innovation and Research Project, Guangxi Bagui Youth Talent Training Program, Guangxi Academy of Artificial Intelligence, and Guangxi Collaborative Innovation Center of Multisource Information Integration and Intelligent Processing.

References

Chen, L.-H.; Zhang, Y.; Huang, T.; Su, L.; Lin, Z.; Xiao, X.; Xia, X.; and Liu, T. 2024. ERASE: Error-Resilient Representation Learning on Graphs for Label Noise Tolerance. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management, CIKM*, 270–280.

Cheng, J.; Tang, Y.; He, C.; Feng, P.; Han, K.; and Guan, Q. 2025. Rethinking Variational Bayes in Community Detection From Graph Signal Perspective. *IEEE Transactions on Knowledge and Data Engineering*, 37(5): 2903–2917.

Cheng, Y.; Shan, C.; Shen, Y.; Li, X.; Luo, S.; and Li, D. 2024. Resurrecting label propagation for graphs with heterophily and label noise. In *Proceedings of the 30th ACM SIGKDD conference on knowledge discovery and data mining*, 433–444.

Dai, E.; Aggarwal, C.; and Wang, S. 2021. NRGNN: Learning a Label Noise Resistant Graph Neural Network on Sparsely and Noisily Labeled Graphs. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD*, 227–236.

Ding, K.; Ma, X.; Liu, Y.; and Pan, S. 2024. Divide and Denoise: Empowering Simple Models for Robust Semi-Supervised Node Classification against Label Noise. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD*, 574–584.

Du, X.; Bian, T.; Rong, Y.; Han, B.; Liu, T.; Xu, T.; Huang, W.; and Huang, J. 2021. Pi-gnn: A novel perspective on semi-supervised node classification against noisy labels. In *arXiv preprint arXiv:2106.07451*.

Fan, W.; Ma, Y.; Li, Q.; He, Y.; Zhao, E.; Tang, J.; and Yin, D. 2019. Graph neural networks for social recommendation. In *The world wide web conference*, 417–426.

Goldberger, J.; and Ben-Reuven, E. 2017. Training Deep Neural Networks Using a Noise Adaptation Layer. In *Proceedings of the 5th International Conference on Learning Representations, ICLR*.

Han, B.; Yao, Q.; Yu, X.; Niu, G.; Xu, M.; Hu, W.; Tsang, I. W.; and Sugiyama, M. 2018. Co-teaching: Robust Training of Deep Neural Networks with Extremely Noisy Labels. In *Advances in Neural Information Processing Systems, NeurIPS*, 8536–8546.

Han, S.; Mao, H.; and Dally, W. J. 2016. Deep Compression: Compressing Deep Neural Network with Pruning, Trained Quantization and Huffman Coding. In *Proceedings of the 4th International Conference on Learning Representations, ICLR*.

He, C.; Cheng, J.; Chen, G.; Guan, Q.; Fei, X.; and Tang, Y. 2024. Detecting communities with multiple topics in attributed networks via self-supervised adaptive graph convolutional network. *Information Fusion*, 105: 102254.

He, C.; Luo, J.; Tang, Y.; Chen, G.; and Guan, Q. 2025. Graph Neural Network Empowers Intelligent Education: A Systematic Review from An Application Perspective. *IEEE Transactions on Learning Technologies*, 1–18.

Hoang, N.; Jin, C. J.; and Murata, T. 2019. Learning Graph Neural Networks with Noisy Labels. In *arXiv preprint arXiv:1905.01591*.

Jiang, L.; Zhou, Z.; Leung, T.; Li, L.-J.; and Fei-Fei, L. 2018. MentorNet: Learning Data-Driven Curriculum for Very Deep Neural Networks on Corrupted Labels. In *Proceedings of the 35th International Conference on Machine Learning, ICML*, volume 80 of *Proceedings of Machine Learning Research*, 2309–2318. PMLR.

Kipf, T. N.; and Welling, M. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *International Conference on Learning Representations*.

Kudugunta, S.; and Ferrara, E. 2018. Deep Neural Networks for Bot Detection. In *Information Sciences, IS*, volume 467, 312–322.

Li, D.; Qian, H.; Li, Q.; Tan, Z.; Gan, Z.; Wang, J.; and Li, X. 2024a. Fedrgl: Robust federated graph learning for label noise. *arXiv preprint arXiv:2411.18905*.

Li, R.; Wang, S.; and Chen-Chuan, K. 2012. Multiple Location Profiling for Users and Relationships from Social Network and Content. *Proceedings of the VLDB Endowment*, 5(11).

Li, X.; Gan, Z.; Li, Q.; Qu, B.; Wang, J.; et al. 2025. Rethinking the impact of noisy labels in graph classification: A utility and privacy perspective. *Neural Networks*, 182: 106919.

Li, X.; Li, Q.; Li, D.; Qian, H.; and Wang, J. 2024b. Contrastive Learning of Graphs under Label Noise. In *Neural Networks*, volume 172, 106113.

Li, Y.; Yin, J.; and Chen, L. 2021. Unified Robust Training for Graph Neural Networks Against Label Noise. In *Proceedings of the 25th Pacific-Asia Conference on Knowledge Discovery and Data Mining, PAKDD, Part I*, volume 12712 of *Lecture Notes in Computer Science*, 528–540. Springer.

Louizos, C.; Welling, M.; and Kingma, D. P. 2018. Learning Sparse Neural Networks through L_0 Regularization. In *Proceedings of the 6th International Conference on Learning Representations, ICLR*.

Patrini, G.; Rozza, A.; Menon, A. K.; Nock, R.; and Qu, L. 2017. Making Deep Neural Networks Robust to Label Noise: A Loss Correction Approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 2233–2241.

Qian, S.; Ying, H.; Hu, R.; Zhou, J.; Chen, J.; Chen, D. Z.; and Wu, J. 2023. Robust Training of Graph Neural Networks via Noise Governance. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining, WSDM*, 607–615.

Van der Maaten, L.; and Hinton, G. 2008. Visualizing data using t-SNE. volume 9.

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2018. Graph Attention Networks. In *International Conference on Learning Representations*.

Wu, Y.; Yao, J.; Xia, X.; Yu, J.; Wang, R.; Han, B.; and Liu, T. 2024. Mitigating Label Noise on Graphs via Topological Sample Selection. In *International Conference on Machine Learning*, 53944–53972. PMLR.

Xia, J.; Lin, H.; Xu, Y.; Tan, C.; Wu, L.; Li, S.; and Li, S. Z. 2023. Gnn cleaner: Label cleaner for graph structured data. *IEEE Transactions on Knowledge and Data Engineering*, 36(2): 640–651.

Xu, K.; Hu, W.; Leskovec, J.; and Jegelka, S. 2019. How Powerful are Graph Neural Networks? In *International Conference on Learning Representations*.

Yin, N.; Shen, L.; Wang, M.; Luo, X.; Luo, Z.; and Tao, D. 2023. OMG: Towards effective graph classification against label noise. *IEEE Transactions on Knowledge and Data Engineering*, 35(12): 12873–12886.

Yuan, J.; Luo, X.; Qin, Y.; Zhao, Y.; Ju, W.; and Zhang, M. 2023. Learning on Graphs under Label Noise. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, 1–5.

Zhang, H.; Cisse, M.; Dauphin, Y. N.; and Lopez-Paz, D. 2018. mixup: Beyond Empirical Risk Minimization. In *International Conference on Learning Representations*.

Zhang, M.; and Chen, Y. 2018. Link prediction based on graph neural networks. In *Advances in neural information processing systems*, volume 31.

Zhu, Y.; Feng, L.; Deng, Z.; Chen, Y.; Amor, R.; and Witbrock, M. 2024. Robust Node Classification on Graph Data with Graph and Label Noise. In *Proceedings of the AAAI Conference on Artificial Intelligence, AAAI*, volume 38, 17220–17227.