MDFS-GNN: MULTI-DILATION GRAPH PROPAGATION WITH FEATURE SHRINKAGE FOR SEMI-SUPERVISED NODE CLASSIFICATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Graph neural networks (GNNs) have been widely used for semi-supervised node classification by propagating structural and feature information across graph neighborhoods. However, when only a limited number of labeled nodes are available, insufficient supervision restricts effective information propagation between labeled and unlabeled nodes. Although high-order GNNs can enlarge the receptive field to capture long-range structural information, their insufficient modeling of local information and ineffective integration of local and global features may limit the learning of discriminative node representations. To address these challenges, we propose MDFS-GNN, a Multi-Dilation Graph Propagation with Feature Shrinkage framework for semi-supervised node classification. Specifically, MDFS-GNN employs graph convolution operations with different dilation factors to capture multi-order neighborhood information and incorporates the original node attributes through residual aggregation to preserve informative features during propagation. Furthermore, a feature shrinkage mechanism reconstructs the neighborhood structure according to node feature similarity and retains the Top-K most similar nodes, thereby strengthening informative local interactions while reducing redundant information. These components are integrated into a two-stage learning framework consisting of initialization and iterative refinement, in which node representations and pseudo-labels are progressively updated to facilitate information propagation across the graph. Experiments on three homophilic and four heterophilic graph datasets under different labeling rates demonstrate the effectiveness and robustness of MDFS-GNN, particularly in scenarios with extremely limited labeled nodes.

1 INTRODUCTION

With the rapid development of graph-based applications, such as social network analysis Wang et al. (2018) and protein-protein interaction prediction Huang et al. (2023), learning effective representations from non-Euclidean graph data has become an important research topic. Unlike conventional neural networks designed for regular Euclidean structures, graph neural networks (GNNs) exploit graph topology and node attributes through neighborhood aggregation, achieving remarkable performance in node classification Luo et al. (2025), graph classification Yao et al. (2025); Froehlich & Koeppl (2025), and link prediction Song et al. (2023). Among them, graph convolutional networks (GCNs) Kipf & Welling (2017) have become a fundamental framework for graph representation learning.

Existing GNN methods, such as GCN Kipf & Welling (2017), GAT Velickovic et al. (2018), and SGC Wu et al. (2019), mainly rely on neighborhood aggregation to update node representations. However, these approaches typically focus on local neighborhood information and have limited capability in capturing long-range structural dependencies. To overcome this limitation, high-order graph neural networks have been proposed to exploit multi-hop neighborhood information. For example, MixHop Abu-El-Haifa et al. (2019) aggregates multiple adjacency powers to capture diverse receptive fields, GraphSAGE Hamilton et al. (2017) adopts sampled multi-hop aggregation for scalable representation learning, APPNP Klicpera et al. (2018) introduces personalized PageRank

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

propagation, and GCNII Chen et al. (2020) preserves information from different propagation depths through residual connections and identity mapping.

GraphHop Xie et al. (2023) further combines label propagation and multi-hop graph aggregation through a two-stage propagation strategy, improving representation learning in large-scale and sparse-label scenarios. Despite these advances, existing high-order GNNs still face challenges in balancing local and global information propagation. Enlarging propagation ranges may introduce irrelevant neighbors and redundant information, while uniform aggregation strategies make it difficult to distinguish informative connections from noisy structures. These issues become more severe under limited-label conditions, where reliable information propagation from labeled nodes to unlabeled nodes is crucial.

To address the challenges of limited labeled data and imbalanced local-global information propagation, this paper proposes a semi-supervised node classification method, named **MDFS-GNN (Multi-Dilation Feature Shrinkage Graph Neural Network)**. The proposed model consists of three key components. First, a multi-dilation graph propagation module is developed to capture multi-order structural dependencies through different dilation factors, while incorporating initial node features via residual connections to alleviate over-smoothing. Second, a feature shrinkage module is introduced to reconstruct the adjacency matrix based on feature similarity and retain the top-K most relevant neighbors, thereby enhancing informative local interactions and reducing redundant propagation. Third, a two-stage information refinement strategy is designed. In the initialization stage, node representations generated by multi-dilation propagation are combined with pseudo-label information to train the classifier, while the labels of labeled nodes are reset to preserve supervision. During the iterative refinement stage, feature shrinkage and multi-dilation propagation are jointly employed to enhance local feature consistency and global structural information propagation. Experiments on homophilic and heterophilic graph benchmarks demonstrate that MDFS-GNN achieves superior performance over existing methods, particularly under limited-label scenarios.

The main contributions of this paper are summarized as follows:

- We introduce **MDFS-GNN**, a novel graph neural network framework for semi-supervised node classification, which integrates multi-dilation propagation and feature shrinkage to enhance representation learning under limited supervision.
- The proposed multi-dilation graph propagation mechanism enables adaptive exploration of diverse receptive fields by incorporating multiple propagation orders, allowing the model to capture both local patterns and long-range structural dependencies without increasing propagation depth.
- We develop a feature shrinkage strategy that leverages node feature similarity to refine graph structures, selectively preserving informative neighborhood relationships while reducing the influence of redundant and noisy connections.
- Comprehensive experiments on seven benchmark datasets covering both homophilic and heterophilic graphs validate the effectiveness of MDFS-GNN, demonstrating consistent improvements over competitive baselines, especially in sparse-label scenarios.

2 METHOD

We propose **MDFS-GNN (Multi-Dilation Feature Shrinkage Graph Neural Network)**, a semi-supervised graph neural network designed for node classification under limited-label scenarios. The key idea of MDFS-GNN is to enhance information propagation under sparse supervision by jointly modeling multi-order structural dependencies and informative local relationships.

Specifically, MDFS-GNN consists of two complementary modules: a multi-dilation graph propagation module and a feature shrinkage module. The former extends polynomial graph propagation with multiple dilation factors to capture neighborhood information from different orders, while the latter reconstructs the neighborhood structure based on feature similarity to preserve informative connections and reduce redundant interactions. Based on these modules, MDFS-GNN adopts a two-stage optimization strategy consisting of initialization and iterative refinement. In the initialization stage, multi-dilation graph propagation is first applied to aggregate multi-scale structural information, and the obtained representations are combined with initial node attributes through residual connections

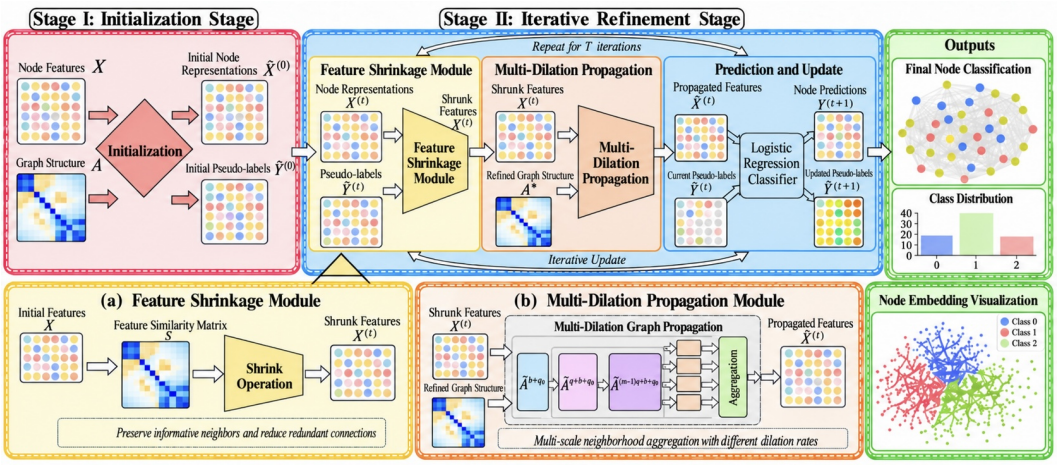


Figure 1: Overall framework of MDFS-GNN.

for classifier initialization and pseudo-label generation. During the iterative refinement stage, feature shrinkage is introduced to refine the neighborhood structure by retaining highly correlated neighbors. The refined features are subsequently propagated through the multi-dilation graph propagation module, and the classifier is updated with the generated pseudo-label information.

Through the interaction between multi-dilation propagation and feature shrinkage, MDFS-GNN effectively integrates global structural dependencies and local feature correlations, thereby improving node representation learning under limited-label conditions. The following subsections first formulate the semi-supervised node classification problem and then introduce the multi-dilation graph propagation module, feature shrinkage module, and optimization strategy in detail.

2.1 Problem Statement

Given an undirected graph $G = (V, E, X, A, Y)$, where V and E denote the node and edge sets, $X \in \mathbb{R}^{n \times d}$ represents the node feature matrix, and $A \in \{0, 1\}^{n \times n}$ denotes the adjacency matrix. The feature vector of node v_i is denoted as $x_i \in \mathbb{R}^d$.

In semi-supervised node classification, the node set V is divided into a labeled set L and an unlabeled set U, where $n = |L| + |U|$. The labels of labeled nodes are represented as one-hot vectors, while the labels of unlabeled nodes are unknown and need to be inferred.

Let $Y \in \mathbb{R}^{n \times c}$ denote the label matrix, where c is the number of classes. For labeled nodes, Y_L contains the ground-truth labels, whereas Y_U represents the predicted label distributions of unlabeled nodes. The final prediction of an unlabeled node is obtained by:

$$y_i = \arg\max_{j \in C} Y_{ij}. \tag{1}$$

The objective of semi-supervised graph learning is to infer the labels of unlabeled nodes by jointly exploiting node attributes, graph topology, and limited supervision information. When the labeling rate is extremely low, i.e., $|L| \ll |U|$, the major challenge is how to effectively propagate reliable information from labeled nodes to unlabeled nodes.

2.2 Multi-Dilation Graph Propagation Module

Conventional graph convolution networks usually aggregate information from fixed-order neighborhoods, which limits their capability to capture long-range dependencies. To overcome this limitation, MDFS-GNN introduces a multi-dilation graph propagation module, which constructs multiple receptive fields by employing different dilation factors.

Given the normalized adjacency matrix $\tilde{A}$, the dilation-based graph propagation operation is constructed by:

$$\mathcal{F}(\tilde{A}) = \left\{ \tilde{A}^{b+q_0}, \tilde{A}^{q+b+q_0}, \cdots, \tilde{A}^{(m-1)q+b+q_0} \right\}, \tag{2}$$

where q_0 represents the initial dilation order, q controls the interval between adjacent dilation orders, b determines the basic neighborhood coverage, and m denotes the number of dilation orders. By adjusting these parameters, multiple structural neighborhoods with different propagation ranges can be obtained.

Based on the generated dilation neighborhoods, the feature propagation at the t-th iteration is formulated as:

$$Z_i^{(t)} = \sigma\left(\tilde{A}^{iq+b+q_0} X^{(t)}\right), \quad i = 0, \cdots, m-1, \tag{3}$$

where $X^{(t)}$ denotes the feature matrix at the t-th iteration and $\sigma(\cdot)$ is the activation function.

To integrate information from different dilation orders while reducing redundant representations, a mean pooling operation is adopted:

$$\tilde{X}^{(t)} = \frac{1}{m} \sum_{i=0}^{m-1} \sigma\left(\tilde{A}^{iq+b+q_0} X^{(t)}\right). \tag{4}$$

Since repeated graph propagation may cause the loss of original node attributes, a residual connection is introduced by concatenating the initial features:

$$\hat{X}^{(t)} = \tilde{X}^{(t)} \| X^{(0)}, \tag{5}$$

where $\|$ denotes feature concatenation and $X^{(0)}$ represents the initial node feature matrix.

The obtained representation $\hat{X}^{(t)}$ preserves both multi-scale structural information and original attribute information, and is subsequently fed into the classifier for node prediction.

2.3 Feature Shrinkage Module

Although the multi-dilation graph propagation module can capture structural information from different neighborhood orders, directly aggregating all connected neighbors may introduce redundant or irrelevant information. Therefore, MDFS-GNN introduces a feature shrinkage module to refine the local neighborhood structure according to feature similarity.

The feature shrinkage module aims to preserve highly correlated neighbors and remove unnecessary connections. Specifically, the similarity between two nodes is calculated based on their initial feature representations. The cosine similarity between nodes v_i and v_j is defined as:

$$S_{ij} = \frac{x_i x_j^T}{\|x_i\| \|x_j\|}, \tag{6}$$

where x_i and x_j denote the initial feature vectors of nodes v_i and v_j, respectively. The obtained similarity matrix $S \in \mathbb{R}^{n \times n}$ describes the feature-level relationships between nodes.

Based on the similarity matrix, the original adjacency matrix is reconstructed by selecting the top-K most similar nodes for each node:

$$A_{ij}^* = \begin{cases} 1, & j \in TopK(S_{i,:}) \text{ or } i \in TopK(S_{j,:}), \\ 0, & otherwise, \end{cases} \tag{7}$$

where $TopK(S_{i,:})$ represents the set of K nodes with the highest similarity scores to node i. The bidirectional selection strategy guarantees that the reconstructed graph preserves reciprocal relationships between similar nodes.

After obtaining the refined adjacency matrix A^*, the feature shrinkage operation is performed as:

$$X^{(t)} = A^* X^{(t)}, \tag{8}$$

where $X^{(t)}$ denotes the feature matrix at the t-th iteration.

Through feature shrinkage, MDFS-GNN reduces the influence of irrelevant neighbors while enhancing information aggregation among highly similar nodes. The refined features are subsequently propagated through the multi-dilation graph propagation module to further capture global structural information.

2.4 Label Updating Based on Logistic Regression Classifier

After obtaining the refined feature representation through the multi-dilation graph propagation and feature shrinkage modules, MDFS-GNN employs a logistic regression classifier to predict node labels and update pseudo-labels.

At the t-th iteration, the classifier takes the propagated feature matrix $\hat{X}^{(t)}$ and the current label matrix $\hat{Y}^{(t)}$ as inputs, and generates the updated label embedding and feature representation:

$$Y^{(t+1)}, X^{(t+1)} = P_{\text{model}}\left(Y|\hat{X}^{(t)}, \hat{Y}^{(t)}, W\right), \tag{9}$$

where $P_{\text{model}}(\cdot)$ denotes the logistic regression classifier and W represents the learnable parameters.

For semi-supervised node classification, the classifier is optimized by jointly considering the labeled nodes and unlabeled nodes. The supervised loss of labeled nodes is defined as:

$$L_L = \frac{1}{|L|} \sum_{i \in L} \ell\left(Y_i, P_{\text{model}}(Y|\hat{x}^{(t)}, \hat{y}^{(t)}; W)\right), \tag{10}$$

where $\ell(\cdot)$ denotes the classification loss function.

Since unlabeled nodes do not contain ground-truth labels, the predicted labels obtained from previous iterations are utilized as pseudo-labels. The pseudo-label loss is formulated as:

$$L_U = \frac{1}{|U|} \sum_{i \in U} \ell\left(\hat{Y}_i^{(t)}, P_{\text{model}}(Y|\hat{x}^{(t)}, \hat{y}^{(t)}; W)\right). \tag{11}$$

The total loss is defined as:

$$L = L_L + \alpha L_U, \tag{12}$$

where α controls the contribution of unlabeled nodes during training.

After each iteration, the pseudo-label matrix is updated. To prevent the accumulation of incorrect pseudo-labels and preserve the original supervision information, the labels of labeled nodes are reset to their ground-truth values:

$$\hat{Y}_i^{(t+1)} = \begin{cases} Y_i^{(t+1)}, & i \in U, \\ Y_i^{(0)}, & i \in L. \end{cases} \tag{13}$$

Through the iterative interaction among feature shrinkage, multi-dilation propagation, and label updating, MDFS-GNN progressively enhances the information propagation from labeled nodes to unlabeled nodes, thereby improving classification performance under limited-label scenarios.

Table 1: Datasets Statistics.

Dataset	Nodes	Features	Edges	Classes	Hom. ratio
Cora	2708	1433	5429	7	0.81
PubMed	19717	500	44324	3	0.80
CiteSeer	3327	3703	4732	6	0.74
Cornell	183	1703	295	5	0.30
Texas	183	1703	309	5	0.11
Wisconsin	251	1703	499	5	0.21
Film	7600	932	26752	5	0.22

Table 2: Hyperparameter Settings.

Dataset	α	Dilation	K
Cora	0.1	2	13
PubMed	0.1	2	10
CiteSeer	0.1	2	7
Cornell	0	3	5
Texas	0	2	8
Wisconsin	0.1	3	4
Film	0	2	7

3 EXPERIMENTS

To evaluate MDFS-GNN under limited-label scenarios, we conduct experiments from three perspectives: (1) comparison with state-of-the-art methods; (2) robustness evaluation under different label rates; and (3) component analysis through ablation studies.Experiments are performed on both homophilic and heterophilic graph datasets. Specifically, Cora, PubMed, and CiteSeer are used as homophilic benchmarks, while Cornell, Texas, Wisconsin, and Film are selected as heterophilic datasets. Dataset statistics, including nodes, edges, features, classes, and homophily ratios, are summarized in Table 1 Choi et al. (2023). The homophilic datasets are citation networks, where nodes represent papers, edges denote citation relationships, features are bag-of-words representations, and labels indicate paper categories Pei et al. (2020). The heterophilic datasets include three WebKB subsets (Cornell, Texas, and Wisconsin) and the Film dataset Tang et al. (2009).

3.1 EXPERIMENT SETUP

All experiments were conducted on an NVIDIA GeForce RTX 3090 GPU using Python 3.7 and PyTorch. All models were optimized with the Adam optimizer, and the reported results are averaged over 10 independent runs. For fair comparison, baseline methods were implemented according to their original settings whenever possible. For MDFS-GNN, the maximum iteration number, learning rate, and weight decay were set to 100, 0.01, and 5×10^{-5}, respectively. Early stopping was applied based on validation loss. The hyperparameters α, dilation factors, and neighbor number K were determined by grid search. Specifically, m was set to 2 or 3, with $q_0 = 0$, $q = 2$, and $b = 1$. α was searched from $\{0, 0.1, 1\}$, and K was selected from $\{1, 2, \ldots, 15\}$. The optimal configurations under the highest labeling rate are reported in Table 2.

3.2 COMPARISON WITH STATE-OF-THE-ART METHODS

We evaluate MDFS-GNN on Cora, CiteSeer, and PubMed datasets. Following the fixed split protocol in Kipf & Welling (2017), 20 labeled nodes per class are selected for training, with 500 and 1000 nodes used for validation and testing, respectively. MDFS-GNN is compared with representative methods, including LPKothari & Jain (2002), GCN Kipf & Welling (2017), GAT Velickovic et al. (2018), SGC Wu et al. (2019), MixHop Abu-El-Haifa et al. (2019), GraphHop Xie et al. (2023), DGIVeličković et al. (2018), BootstrapEliav & Cohen (2018), Co-training Li et al. (2018), Self-training Li et al. (2018), AdaGNNDong et al. (2021), GCNIIChen et al. (2020), CMPGNNYan et al. (2025), and MSP-LRZhang et al. (2025). We consider both fixed and random split settings, and report the average accuracy over 10 runs. As shown in Table 3, MDFS-GNN achieves competitive or superior performance compared with existing methods.

3.3 THE EFFECT OF THE NUMBER OF LABELED DATA

Effect of Labeled Data Size: We investigate the impact of label availability on MDFS-GNN and baseline methods under different labeling rates. Experiments are conducted on both homophilic and heterophilic graphs, and the reported results are averaged over ten independent runs. Bold and underline indicate the best and top-three results, respectively, while values in parentheses denote the performance gain or degradation of MDFS-GNN compared with the best baseline.

Table 3: Node Classification Accuracy on Standard and Random Splits.

Method	**Standard**			**Random**		
	Cora	CiteSeer	PubMed	Cora	CiteSeer	PubMed
DeepWalk	43.2	67.2	65.3	70.2	47.2	72.0
LP	71.4	47.5	70.6	-	-	-
GCN	81.5	69.2	77.4	80.1	67.9	78.9
GAT	79.6	68.2	76.5	81.19	71.01	79.42
SGC	80.7	68.5	77.0	80.32	70.48	78.39
MixHop	77.0	61.9	74.8	-	-	-
N-GCN	81.8	71.0	79.4	-	-	-
GraphHop	81.0	70.3	77.2	-	-	-
DGI	82.3	71.8	76.8	81.84	**71.55**	78.25
Bootstrap	78.4	53.6	78.8	78.2	50.3	75.6
Co-training	80.0	68.7	76.9	-	-	-
Self-training	80.0	69.4	77.0	-	-	-
AdaGNN	79.9	69.2	78.9	80.4	66.1	75.9
GCNII	**85.7**	72.4	**79.8**	82.8	69.1	79.3
CMPGNN	81.9	72.3	76.8	82.45	70.39	74.11
MSP-LR	81.53	70.29	77.88	82.72	68.07	**79.79**
OURS	81.9	**73.2**	78.0	**83.5**	70.5	79.5

Table 4: Classification Accuracy (%) of Homophilic Graph Datasets under Different Label Rates

Method	**1 label**			**2 labels**			**4 labels**		
	Cora	**CiteSeer**	**PubMed**	**Cora**	**CiteSeer**	**PubMed**	**Cora**	**CiteSeer**	**PubMed**
GCN	42.4	26.0	48.1	52.0	34.7	65.0	65.0	52.0	71.0
GAT	41.8	26.8	57.6	51.8	42.0	70.0	66.4	52.4	69.8
SGC	40.6	28.7	66.7	51.2	31.6	70.1	65.2	46.1	69.7
MixHop	30.2	26.2	53.6	46.4	30.1	69.0	58.0	43.4	69.5
FA-GCN	41.3	32.0	64.6	57.1	38.7	71.9	67.2	48.4	**72.4**
AdaGNN	36.2	34.3	67.5	45.0	37.5	69.4	54.6	47.1	70.6
ACM-GCN	29.0	33.4	56.1	57.2	43.0	67.0	70.0	52.9	69.9
GLP	55.7	24.1	24.1	57.7	40.0	40.0	70.3	49.8	49.8
GCN-LP	46.7	47.0	39.2	54.6	48.8	39.2	62.5	54.4	69.6
Co-training	53.1	28.2	62.4	59.4	36.6	68.7	68.0	38.8	67.3
Self-training	40.6	30.5	54.1	52.3	36.3	70.5	67.5	43.6	67.4
GraphHop	59.1	48.4	67.1	59.2	53.3	68.8	69.3	54.3	69.6
GCNII	56.9	25.0	67.3	58.6	46.2	71.2	68.7	51.8	70.9
CMPGNN	64.3	50.0	63.1	73.8	52.3	65.2	78.9	57.3	69.2
MSP-LR	67.3	54.7	71.5	69.2	54.0	72.2	76.1	57.7	71.5
OURS	**72.5** (↑7.2)	**62.3** (↑7.6)	**72.0** (↑0.5)	**75.0** (↑1.2)	**64.5** (↑10.5)	**72.4** (↑0.2)	**81.5** (↑2.6)	**67.1** (↑9.4)	72.3 (↓0.1)

Homophilic graph datasets: For each class, 1, 2, and 4 labeled nodes are randomly selected for training, and 500 unlabeled nodes are used for validation Xie et al. (2023). MDFS-GNN is compared with representative methods, including GCN Kipf & Welling (2017), GAT Velickovic et al. (2018), SGC Wu et al. (2019), MixHop Abu-El-Haifa et al. (2019), FA-GCN Bo et al. (2021), AdaGNNDong et al. (2021), ACM-GCN Luan et al. (2022), GLP Li et al. (2019), GCN-LP Wang & Leskovec (2022), Co-training Li et al. (2018), Self-training Li et al. (2018), GraphHop Xie et al. (2023), GCNIIChen et al. (2020), CMPGNNYan et al. (2025), and MSP-LRZhang et al. (2025).

As shown in Table 4, increasing the number of labeled nodes improves the performance of all methods. MDFS-GNN achieves more significant gains under extremely limited-label settings, improving accuracy over the best baseline by 7.2% and 7.6% on Cora and CiteSeer with only one labeled node per class, respectively. It also maintains competitive performance with more labeled nodes, demonstrating its effectiveness in low-label scenarios.

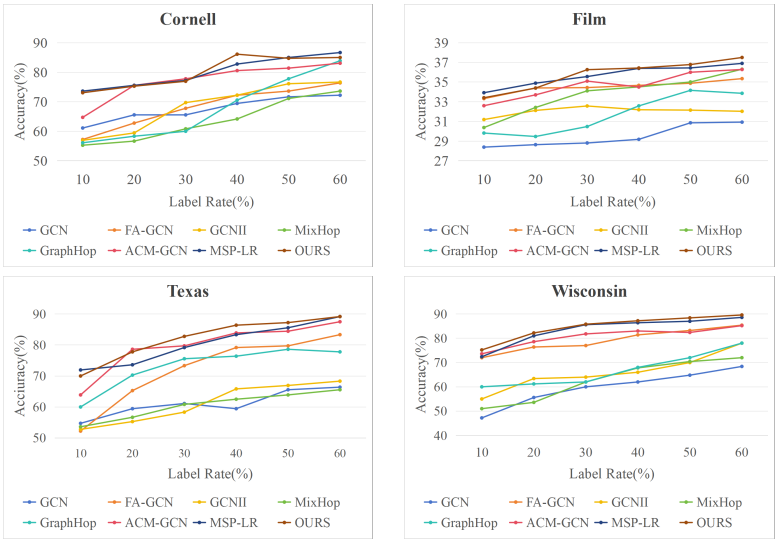


Figure 2: Accuracy Comparison under Different Label Rates on Heterophilic Graphs

Table 5: Ablation Results (20 Labels/Class)

Methods	Cora	CiteSeer	PubMed
MD-GNN	80.3±0.3	65.5±0.6	78.3±0.1
MDFS-GNN	83.5±0.2	70.5±0.3	79.1±0.4

Table 6: Ablation Results (1 Label/Class)

Methods	Cora	CiteSeer	PubMed
MD-GNN	55.7±3.0	37.2±0.8	70.0±0.1
MDFS-GNN	74.5±0.9	64.4±0.9	70.8±0.9

Heterophilic graph datasets: To evaluate the robustness of our model on heterogeneous graphs, we conduct experiments on four benchmark datasets. The datasets are divided into validation and test sets with a ratio of 20% each, while the training ratio is varied from 10% to 60% with an increment of 10%. The compared methods include GCN Kipf & Welling (2017), FA-GCN Bo et al. (2021), GCNII Chen et al. (2020), MixHop Abu-El-Haifa et al. (2019), GraphHop Xie et al. (2023), ACM-GCN Luan et al. (2022), and MSP-LR Zhang et al. (2025). Figure 3 presents the classification performance under different labeling rates. All methods benefit from increased supervision, while MDFS-GNN consistently achieves competitive performance compared with existing approaches, demonstrating its effectiveness on heterogeneous graph data.

3.4 Ablation Studies and Similarity Score Analysis

We conduct ablation studies and similarity analysis to evaluate the effectiveness of the feature shrinkage module in MDFS-GNN. Table 6 compares the base model (MD-GNN,Multi-Dilation Graph Neural Network) and MDFS-GNN with feature shrinkage under 20 labeled nodes per class. Feature shrinkage improves the accuracy by 3.2%, 5.1%, and 0.6% on Cora, CiteSeer, and PubMed, respectively. Under the extremely limited-label setting with one labeled node per class (Table 7), the improvements further increase to 18.8%, 27.2%, and 1.1%, demonstrating the effectiveness of feature shrinkage in enhancing information propagation. Similarity analysis is further conducted to investigate feature preservation. As shown in Figure 3, MDFS-GNN maintains higher similarity between propagated and initial features after multiple propagation steps, indicating that feature shrinkage alleviates feature degradation caused by redundant aggregation. Additional experiments on Cora with four labeled nodes per class further confirm that the proposed module improves feature consistency and representation quality.

3.5 Hyperparameter Analysis

To validate the impact of hyperparameter K on model performance, we conducted experiments on the Cora and CiteSeer datasets with 1, 2, and 4 labeled nodes per class. Results (Figure 4) show a significant interaction between label density and K: on Cora, performance follows a unimodal

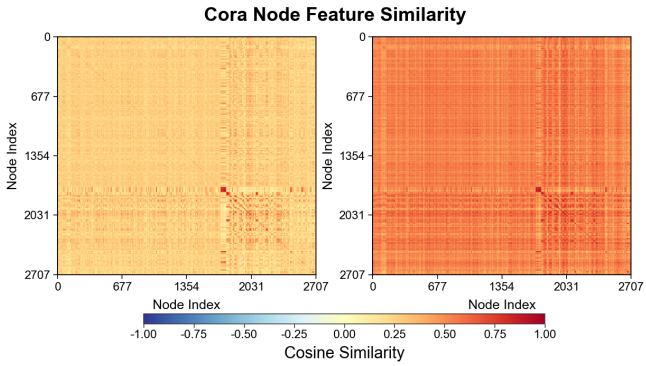


Figure 3: Feature Similarity Comparison Before and After Feature Shrinkage on Cora

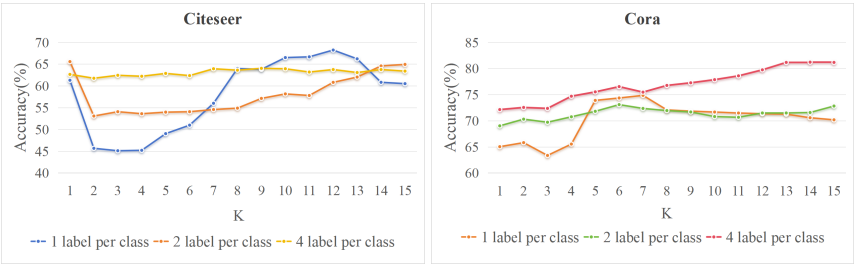


Figure 4: Hyperparameter Analysis on CiteSeer and Cora Datasets

distribution with K (peaking at K=7) under sparse labels, while robustness to larger K strengthens with more labels, indicating synergistic optimization between label density and graph structure; on CiteSeer, K exhibits a "Goldilocks effect" (peaking at K=11) under extreme sparsity, transitioning to monotonic improvement or stability as labels increase, reducing sensitivity to K. These findings highlight that sparser labels intensify the "diminishing returns" effect of K, requiring careful tuning to balance information gain and noise, whereas richer labels enhance adaptability to larger K, forming a dynamic equilibrium through complementary information. This provides empirical guidance for hyperparameter optimization in graph neural networks.

4 Conclusion

In this paper, we propose **MDFS-GNN (Multi-Dilation Feature Shrinkage Graph Neural Network)** for semi-supervised node classification under limited-label scenarios. By integrating multi-dilation graph propagation with feature shrinkage, MDFS-GNN addresses the limitations of fixed receptive fields and insufficient local-global information interaction in conventional GCNs. The multi-dilation propagation mechanism enables the model to capture structural dependencies from different neighborhood ranges, while feature shrinkage refines the neighborhood structure according to node feature similarity, preserving informative relationships and reducing redundant information. Based on these components, the two-stage learning framework progressively refines node representations and pseudo-labels, thereby improving information propagation between labeled and unlabeled nodes. Extensive experiments on both homophilic and heterophilic graph benchmarks demonstrate that MDFS-GNN achieves competitive or superior node classification performance, with particularly notable advantages under extremely limited-label settings. These results confirm the effectiveness of jointly exploiting multi-scale structural information and informative local relationships for label-efficient graph learning. Nevertheless, the model may still be affected by pseudo-label error accumulation on noisy graphs, while the selection of multiple hyperparameters introduces additional tuning costs. Future work will focus on more reliable pseudo-label refinement, adaptive hyperparameter learning, and lightweight feature shrinkage mechanisms to further improve the robustness and efficiency of MDFS-GNN.

REFERENCES

Sami Abu-El-Haifa, Bryan Perozzi, Amol Kapoor, Nazanin Alipourfard, Kristina Lerman, Hrayr Harutyunyan, Greg Ver Steeg, and Aram Galstyan. Mixhop: Higher-order graph convolutional architectures via sparsified neighborhood mixing. In *36th International Conference on Machine Learning (ICML)*, volume 97, 2019.

Deyu Bo, Xiao Wang, Chuan Shi, and Huawei Shen. Beyond low-frequency information in graph convolutional networks. In *35th AAAI Conference on Artificial Intelligence*, volume 35, pp. 3950–3957, 2021.

Ming Chen, Zhewei Wei, Zengfeng Huang, Bolin Ding, and Yaliang Li. Simple and deep graph convolutional networks. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pp. 1725–1735, 2020.

Jeongwhan Choi, Seoyoung Hong, Noseong Park, and Sung-Bae Cho. Gread: Graph neural reaction-diffusion networks. In *40th International Conference on Machine Learning*, volume 202, 2023.

Yushun Dong, Kaize Ding, Brian Jalaian, Shuiwang Ji, and Jundong Li. Adagnn: Graph neural networks with adaptive frequency response filter. In *30th ACM International Conference on Information and Knowledge Management (CIKM)*, pp. 392–401, 2021. doi: 10.1145/3459637.3482226.

Buchnik Eliav and Edith Cohen. Bootstrapped graph diffusions: Exposing the power of nonlinearity. *Proc. ACM Meas. Anal. Comput. Syst.*, 2(1), 2018. doi: 10.1145/3179413.

Philipp Froehlich and Heinz Koeppl. Graph structure inference with bam: neural dependency processing via bilinear attention. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, 2025.

William L. Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *31st Annual Conference on Neural Information Processing Systems (NIPS)*, volume 30, 2017.

Yan Huang, Stefan Wuchty, Yuan Zhou, and Ziding Zhang. Sgppi: structure-aware prediction of protein-protein interactions in rigorous conditions with graph convolutional network. *BRIEFINGS IN BIOINFORMATICS*, 24(2), 2023.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations*, 2017.

Johannes Klicpera, Aleksandar Bojchevski, and Stephan Günnemann. Predict then propagate: Graph neural networks meet personalized pagerank. In *International Conference on Learning Representations*, 2018.

R. Kothari and V. Jain. Learning from labeled and unlabeled data. In *Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02*, volume 3, pp. 2803–2808, 2002.

Qimai Li, Zhichao Han, and Xiao-Ming Wu. Deeper insights into graph convolutional networks for semi-supervised learning. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*, pp. 3538–3545, 2018.

Qimai Li, Xiao-Ming Wu, Han Liu, Xiaotong Zhang, and Zhichao Guan. Label efficient semi-supervised learning via graph filtering. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9574–9583, 2019.

Sitao Luan, Chenqing Hua, Qincheng Lu, Jiaqi Zhu, Mingde Zhao, Shuyuan Zhang, Xiao-Wen Chang, and Doina Precup. Revisiting heterophily for graph neural networks. In *36th Conference on Neural Information Processing Systems (NeurIPS)*, 2022.

Yuankai Luo, Lei Shi, and Xiao-Ming Wu. Classic gnns are strong baselines: reassessing gnns for node classification. NIPS '24, Red Hook, NY, USA, 2025. Curran Associates Inc. ISBN 9798331314385.

Hongbin Pei, Bingzhe Wei, Kevin Chen-Chuan Chang, Yu Lei, and Bo Yang. Geom-gcn: Geometric graph convolutional networks. *CoRR*, abs/2002.05287, 2020.

Xiran Song, Jianxun Lian, Hong Huang, Zihan Luo, Wei Zhou, Xue Lin, Mingqi Wu, Chaozhuo Li, Xing Xie, and Hai Jin. xgcn: An extreme graph convolutional network for large-scale social link prediction. In *Proceedings of the ACM Web Conference 2023*, pp. 349–359, 2023.

Jie Tang, Jimeng Sun, Chi Wang, and Zi Yang. Social influence analysis in large-scale networks. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 807–816, 2009. doi: 10.1145/1557019.1557108.

Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *6th International Conference on Learning Representations*, 2018.

Petar Veličković, William Fedus, William L. Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. Deep graph infomax, 2018. URL `https://arxiv.org/abs/1809.10341`.

Hongwei Wang and Jure Leskovec. Combining graph convolutional neural networks and label propagation. *ACM Trans. Inf. Syst.*, 40(4), 2022. doi: 10.1145/3490478.

Hongwei Wang, Jia Wang, Jialin Wang, Miao Zhao, Weinan Zhang, Fuzheng Zhang, Xing Xie, and Minyi Guo. Graphgan: Graph representation learning with generative adversarial nets. In *32nd AAAI Conference on Artificial Intelligence*, pp. 2508–2515, 2018.

Felix Wu, Amauri H. Souza Jr., Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Q. Weinberger. Simplifying graph convolutional networks. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pp. 6861–6871, 2019.

Tian Xie, Bin Wang, and C-C Jay Kuo. Graphhop: An enhanced label propagation method for node classification. *IEEE Transactions On Neural Networks and Learning Systems*, 34(11):9287–9301, 2023.

Hui Yan, Yuan Gao, Guoguo Ai, Huan Wang, and Xin Li. Contrastive message passing for robust graph neural networks with sparse labels. *Neural Networks*, 182:106912, 2025.

Ruijie Yao, Sheng Jin, Lumin Xu, Wang Zeng, Wentao Liu, Chen Qian, Ping Luo, and Ji Wu. Gkgnet: Group k-nearest neighbor based graph convolutional network for multi-label image recognition. In *COMPUTER VISION-ECCV 2024, PT XVIII*, volume 15076, pp. 91–107, 2025.

Jingxiao Zhang, Shifei Ding, Jian Zhang, Lili Guo, and Ling Ding. Multi-channel set polynomial based label regularized graph neural networks against extreme data scarcity. *Pattern Recognition*, 166:111754, 2025.