Aggregation Mechanism Based Graph Heterogeneous Networks Distillation

Xiaobin Hong , Mingkai Lin* , Xiangkai Ma , Wenzhong Li* , Sanglu Lu
State Key Laboratory for Novel Software Technology, Nanjing University
{xiaobinhong, xiangkai.ma}@smail.nju.edu.cn, {mingkai, lwz, sanglu}@nju.edu.cn

Abstract

Graph Neural Networks (GNNs) have demonstrated remarkable effectiveness across various tasks but are often hindered by their high computational overhead. GNN-to-MLP distillation provides a promising remedy by transferring knowledge from complex GNNs to lightweight MLPs. However, existing methods largely overlook the differences in aggregation mechanisms and heterogeneous architectures. Simplifying such intricate information into MLP potentially causes information loss or distortion, ultimately resulting in suboptimal performance. This paper proposes an aggregation mechanism enhanced GNN distillation framework (AMEND). AMEND introduces multi-scope aggregation context preservation to replicate the teacher's broad aggregation scopes and an aggregation-enhanced centered kernel alignment method to match the teacher's aggregation patterns. To ensure efficient and robust knowledge transfer, we integrate a manifold mixup strategy, enabling the student to capture the teacher's insights into mixed data distributions. Experimental results on 8 standard and 4 large-scale datasets demonstrate that AMEND consistently outperforms state-of-the-art distillation methods.

1 Introduction

Graph is a universal language for modeling complex systems and is widely used to represent entities and their relations in a variety of domains [Dwivedi *et al.*, 2023; Hong *et al.*, 2024a], such as social networks [Xia *et al.*, 2022; Sharma *et al.*, 2024], protein-protein interaction networks [Liu *et al.*, 2020; Jha *et al.*, 2022], citation networks [Kipf and Welling, 2016; Yang *et al.*, 2021], etc. The success of GNNs lies in their aggregation mechanisms, which facilitate information propagation and capture complex relationships. Effective aggregation depends on the *scope* (how far the model looks) and the *pattern*(how information is combined). Early GNNs[Kipf and Welling, 2016; Jha *et al.*, 2022] used fixed, layer-wise schemes to aggregate local neighbor information. This was enhanced by Graph Attention Networks (GATs)[Velickovic *et al.*, 2017], which applied attention to assign dynamic weights to neighbors. More recently, Graph Transformers (GTs)[Yun *et al.*, 2019; Chen *et al.*, 2023] have introduced multi-head self-attention to capture global dependencies and richer interactions, overcoming limitations like over-smoothing and limited receptive fields. These developments greatly extend GNNs' capacity for complex graph mining tasks.

The complex aggregation mechanism and computational heft of GNNs can complicate their integration into latency-sensitive, large-scale applications. To address this, leveraging a Multi-layer Perceptron (MLP) for swift, streamlined deployment becomes appealing. A promising approach is to transfer insights from the GNN to an MLP, thus balancing potency with efficiency. GNN-to-MLP methods in graphs have recently received widespread attention and investigation. The GLNN [Zhang *et al.*, 2022] is a pioneering work advocating the distillation of a proficiently trained GNN into an efficient MLP, adhering to the traditional logit distillation approach and prediction mimicking. NOSMOG [Tian *et al.*, 2023] enhances the student MLP's capacity to grasp graph topology by appending structural encodings to the initial node attributes in its input layer. It also innovates by incorporating noise adversarial training as an additional module to bolster the MLP's robustness. VQGraph [Yang *et al.*, 2024] introduces the VQ-VAE technique in graph processing, condensing the teacher GNN's node embeddings into a compact codebook, and leveraging the ordering of query nodes relative to this codebook as a distillation signal. Currently, Graph Transformers are increasingly taking the place of GNNs in graph mining due to their superior global attention and scalability. Distilling the one-to-all attention aggregation pattern of the GT model into an efficient MLP has not been studied yet.

However, existing GNN-to-MLP distillation methods, mostly derived from classical knowledge distillation, fail to consider the unique role of aggregation mechanisms in graph learning. A toy experiment (Fig. 1) demonstrates this gap by comparing last-layer node embedding correlations between original GCN, GAT, GT models and their distilled MLP counterparts. The results show that differences in aggregation scope (e.g., GAT vs. GT) and pattern (e.g., GCN vs. GAT) significantly affect distillation outcomes. As aggregation becomes broader and more complex, the correlation between teacher and student embeddings declines, revealing a grow-

*Corresponding authors.

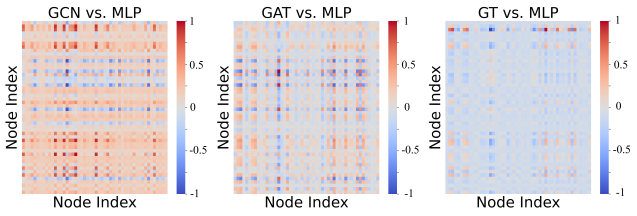


Figure 1: Pubmed dataset's hidden node embedding correlations for GCN, GAT, GT, and MLP models are shown, with warm colors indicating high correlation. MLP has a high correlation with GCN due to GCN's fixed weight and local aggregation, resembling MLP's simplicity. Correlation drops for GAT due to its attention-based local aggregation and is lowest for GT, which uses global attention aggregation, differing significantly from MLP.

ing representational gap that existing methods fail to bridge. These findings highlight the critical influence of aggregation mechanisms on representational alignment. Without explicitly addressing these differences, current approaches struggle to transfer structural knowledge essential for accurate student approximation. This leads to three key challenges: (1) **Aggregation scope mismatch**: GNNs operate over varying receptive fields, while MLPs lack such structural context, making knowledge transfer difficult. (2) **Pattern misalignment**: Diverse aggregation strategies yield different structural representations, complicating the learning of consistent node semantics. (3) **Rigid alignment strategies**: Many methods rely on direct feature or logit matching, ignoring model heterogeneity and limiting student model effectiveness.

As shown in Fig. 2, the persistent challenges highlight the necessity of our holistic framework AMEND (**A**ggregation **M**echanism **E**nhanced G**N**N **D**istillation), aiming to eliminate the impact of aggregation mechanism during GNN distillation through three key components. First, we propose *Multi-scope Aggregation Context Preservation* for preserving local and global dependencies, ensuring the student model captures both neighborhood information and broader structural contexts. Second, we design *Aggregation-enhanced Centered Kernel Alignment*, which aligns the aggregation patterns between the teacher and student models using kernelized similarities, incorporating graph structural information to transfer the teacher's aggregation behavior. Third, *Manifold Mixup Soft Matching*, which generates mixed embeddings through shuffling and interpolation, ensuring the student model mimics the teacher's structural knowledge while standardizing logits for efficient knowledge transfer. Together, these components form a unified approach that overcomes the challenges posed by varying aggregation mechanisms, guaranteeing effective distillation. To fully evaluate the proposed method, we conduct extensive experiments on 8 regular graph datasets and 4 large-scale graph datasets to compare with state-of-the-art methods. The experimental results demonstrate the effectiveness and superiority of AMEND. Our contributions can be summarized as:

- We are the first to systematically investigate the aggregation mechanisms in GNN distillation, introducing a new perspective on structure-aware knowledge transfer.
- We propose AMEND, a novel framework designed to enhance GNN-to-MLP distillation incorporating multi-scope aggregation context preservation, aggregation-enhanced kernel alignment, and manifold mixup soft matching, providing a comprehensive solution to bridge the gap between GNNs and MLPs.
- We perform extensive experiments on 8 regular and 4 large-scale graph datasets, and AMEND achieves superior performance compared to state-of-the-art methods.

2 Related Work

2.1 Graph Neural Networks

Graph Neural Networks (GNNs) have become crucial in graph representation learning due to their message passing paradigm. Vanilla GNNs aggregate node information from the local neighborhood [Kipf and Welling, 2016; Chen *et al.*, 2018; Hong *et al.*, 2021; Lin *et al.*, 2025], which limits the receptive field of node aggregation and the discrimination of information flows. To improve the discriminative of first-order neighbors, attention-based GNNs [Velickovic *et al.*, 2017; He *et al.*, 2023; Fountoulakis *et al.*, 2023] achieve the heterogeneous information filtering by adaptively adjusting the weights of node aggregation. Additionally, Graph Transformers (GTs) have emerged as a powerful GNN in graph representation learning, addressing limitations of traditional message-passing GNNs such as over-smoothing, over-squashing, and difficulty in modeling long-range dependencies and heterogeneous node types. GTs leverage the attention mechanism to capture global context and complex relational patterns for high-order node aggregation. The core innovation of GTs lies in their ability to apply attention across nodes in a graph, effectively modeling interactions without being constrained by locality. Graphormer [Ying *et al.*, 2021] extended the Transformer architecture to graph data by incorporating spatial encoding and structural encodings. GTN [Dwivedi and Bresson, 2020] expanded the node raw attributes with Laplacian eigenvectors and sent them to a vanilla transformer encoder for graph structure capturing. Integrating GCNs and GT to exploit neighborhood messages in global modeling has also received considerable attention recently. They can be categorized as combining GTs with GCNs to enhance the structure-awareness of GTs using incomplete message propagation [Wu *et al.*, 2021; Chen *et al.*, 2022; Rampášek *et al.*, 2022] and integrating the structural bias [Hussain *et al.*, 2022; Zhao *et al.*, 2021; Deng *et al.*, 2024] into the self-attention matrix to improve their expressiveness. Despite their strengths, GNNs are computationally intensive and require significant resources for both training and inference [Zhang *et al.*, 2022; Lin *et al.*, 2024], limiting their deployment in resource-constrained environments. This motivates us to distill the aggregation mechanism into more lightweight models such as MLP, which offers faster inference and lower resource consumption while aiming to retain the performance benefits of GNNs.

2.2 Knowledge Distillation on Graphs

Knowledge Distillation (KD) [Hao *et al.*, 2024; Hong *et al.*, 2024b; Wang *et al.*, 2025; Yang *et al.*, 2023] aims to transfer

the knowledge embedded in a cumbersome teacher to a simpler student. Traditional graph-to-graph KD focuses on transferring insights from larger, deeper GNNs to more compact student GNNs. Notable methods include LSP [Yang *et al.*, 2020] and TinyGNN [Yan *et al.*, 2020], which emphasize the preservation of localized structural patterns, and RDD [Zhang *et al.*, 2020] enhanced the reliability of node and edge representations to ensure the student GNN accurately mirrors the teacher GNN's essential characteristics. Distilling GNN knowledge into MLPs seeks faster reasoning, lightweight deployment, and scalability free from graph size constraints. GLNN [Zhang *et al.*, 2022] first introduced GNN-to-MLP following vanilla predictive mimicking with the soft label from a teacher GNN.. KRD [Wu *et al.*, 2023] employed a reliable sampling strategy to train MLPs with highly confident knowledge, ensuring robust performance despite the simplified architecture. NOSMOG [Tian *et al.*, 2023] integrated structural and attribute features into the MLP inputs, creating a structure-aware model enhanced by adversarial feature augmentation for noise robustness. Additionally, VQ-Graph [Yang *et al.*, 2024] introduced a code-based distillation method and performed sort alignment by leveraging quantization techniques. Despite the success of GNN-to-MLP, due to their similar parameter spaces and feature transformations, challenges persist when extending to different node aggregation GNNs (e.g., GT). The more complex structure of GTs, characterized by global attention mechanisms, complicates the direct application of conventional feature and predictive alignment strategies. Such methods are inadequate for effectively transferring the rich knowledge embedded in GNNs to MLPs, necessitating more sophisticated distillation techniques to bridge the heterogeneous models and ensure the student model benefits from the teacher's capabilities.

3 Methodology

3.1 Problem Definition

A graph can be represented by $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V}$ and $\mathcal{E}$ are the node set and edge set, respectively. The graph size can be dented as N, and the nodes attribute is formed with a feature matrix $\mathbf{X} \in \mathbb{R}^{N\times d}$, where d is the feature dimensions. The adjacency matrix $\mathbf{A} \in \mathbb{R}^{N\times N}$ indicates the graph topology, where $\mathbf{A}_{i,j} = 1$ denotes node v_i is connected with node v_j, otherwise $\mathbf{A}_{i,j} = 0$. D is a diagonal matrix representing the degrees of the nodes, $D_{ii} = \sum_j \mathbf{A}_{ij}$. For node classification, the most important graph mining task, the prediction targets are $\mathbf{Y} \in \mathbb{R}^{N\times c}$, where c is the number of node classes. Given the labels $\mathbf{Y}_L$, the goal is to predict the labels $\mathbf{Y}_U$ for unlabeled nodes. The GNN-to-MLP task aims to distill the aggregated node embedding from large GNNs to a lightweight MLP.

3.2 Aggregation Mechanism Enhanced Distillation

The overview framework of our proposed AMEND is shown in Figure 2. First, AMEND preserves both local and global dependencies by constructing a multi-scope context for node representations. This ensures that the student MLP can emulate the diverse aggregation ranges of the teacher model. To further enhance the transfer of aggregation patterns, AMEND explicitly aligns the aggregation patterns between the teacher and student models using a kernel-based similarity metric ACKA, ensuring that the MLP can replicate the nuanced aggregation patterns of the GNN, even without a native graph-aware mechanism. Lastly, AMEND facilitates knowledge transfer by manifold mixed embeddings, allowing the teacher's expressive capacity to be distilled into the student. This step smooths the differences in representational capabilities by combining soft matching strategies and embedding mixup to propagate knowledge effectively.

Multi-scope Aggregation Context Preservation. To tackle the challenge of aggregation scope in knowledge distillation, we propose strategies to preserve multi-scope neighborhood aggregation contexts in the GNN teacher model. This ensures effective knowledge transfer regardless of variations in the teacher model's aggregation scope. Thus, we extend the teacher model's outputs with additional embeddings that explicitly encode aggregation scope information, complementing the general embeddings generated by the teacher model.

Specifically, to effectively capture neighborhood information across multiple aggregation levels, we first construct a node propagation sequence through multi-scope message propagation. This sequence encodes aggregation information at different hops and is defined as:

$$\mathbf{H}^{(0)} = [\hat{\mathbf{A}}^0\mathbf{X}, \hat{\mathbf{A}}^1\mathbf{X}, \cdots, \hat{\mathbf{A}}^k\mathbf{X}], \tag{1}$$

where $\hat{\mathbf{A}} = \tilde{D}^{-1/2}\tilde{A}\tilde{D}^{-1/2}$ is the Laplace normalized adjacency matrix, and k is the number of hops. The resulting node embeddings are then processed by a global aggregation module to capture higher-level dependencies:

$$\mathbf{H}^{(l+1)} = \mathbf{H}^{(l)} + \text{Gloabl}(\mathbf{H}^{(l)}; \Theta), \tag{2}$$

where Global$(;\Theta)$ denotes the global aggregation function effectively capturing both local and global structural patterns. It is parametrized with Θ and can be implemented by a self-attention approach.

In this way, the general GNN teacher model's output for distillation can be combined with the scope information by a self-weighted readout function, allowing adaptive aggregation by assigning different importance to various scopes:

$$\mathbf{Z}_T = \text{GNN}(\hat{\mathbf{A}}^0, \mathbf{X}) + \sum_{i=0}^{k} \alpha_i Z_i, \tag{3}$$

where α_i are learnable weights and Z_i=$\mathbf{H}^{(L)}_{ik:(i+1)k}$ represents the embeddings corresponding to each scope in final $\mathbf{H}^{(L)}$.

Furthermore, inspired by graph transformers [Chen *et al.*, 2023; Chen *et al.*, 2022; Zhou *et al.*, 2024a], we also incorporate position encoding ($\mathbf{X} \leftarrow \mathbf{X}+\mathbf{X}_{pe}$) to capture the relative positions. Enhanced by random walk-based position encoding, this allows the model to consider a broader scope when performing aggregation. In the practical distillation process, position encoding is shared across teacher and student models, ensuring consistent knowledge transfer.

Aggregation-enhanced Centered Kernel Alignment. To effectively enable the student MLP to mimic the teacher GNN model's patterns to perform node aggregation, we propose Aggregation-enhanced Centered Kernel Alignment (ACKA)

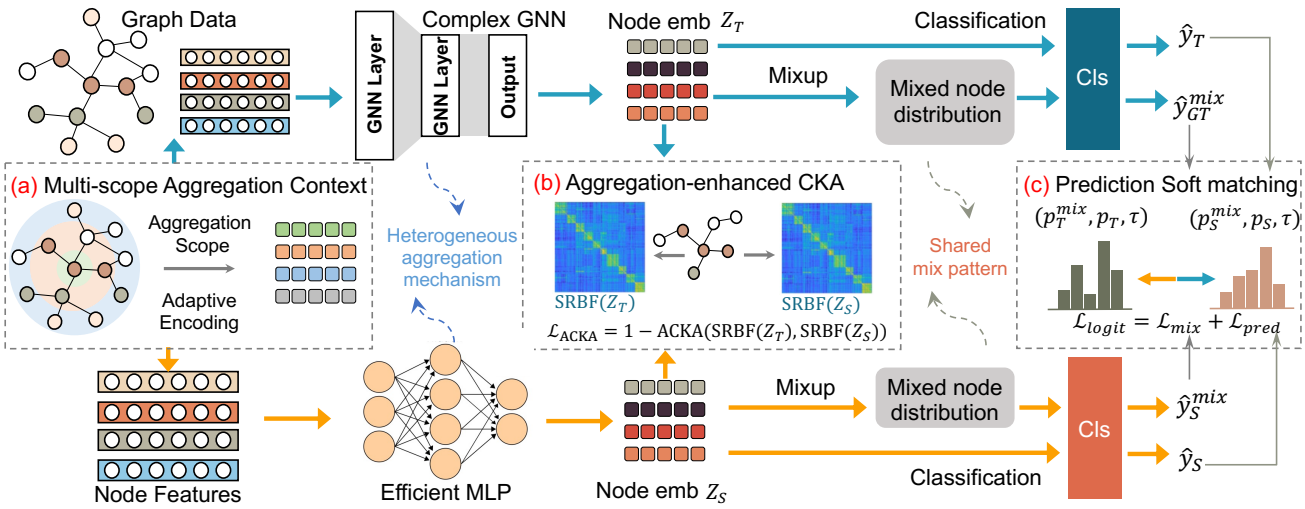


Figure 2: The overview framework of AMEND . (a) Multi-scope Aggregation Context Preservation; (b) Aggregation-enhanced Centered Kernel Alignment; and (c) Manifold Mixup Soft Matching Distillation.

as an intermediate supervision method. ACKA enhances the structure capture ability of student MLP by explicitly aligning aggregation patterns between the teacher and student models.

ACKA performs as a metric for representation similarity, which aligns aggregated representations by leveraging kernelized similarities between node embeddings. Its formulation is:

$$\text{ACKA}(\mathbf{K}, \mathbf{L}) = \frac{\text{HSIC}(\mathbf{K}, \mathbf{L})}{\sqrt{\text{HSIC}(\mathbf{K}, \mathbf{K})\text{HSIC}(\mathbf{L}, \mathbf{L})}}, \quad (4)$$

where the Hilbert-Schmidt Independence Criterion (HSIC) is empirically estimated by:

$$\text{HSIC}(\mathbf{K}, \mathbf{L}) = \frac{1}{(b-1)^2}\text{tr}(\mathbf{KCLC}), \quad (5)$$

and $\mathbf{C}$ is the centering matrix $\mathbf{C} = \mathbf{I} - \frac{1}{b}\mathbf{1}\mathbf{1}^\top$, $\mathbf{K}, \mathbf{L} \in \mathbb{R}^{b\times b}$ are kernel matrices derived from the teacher and student embeddings, representing their aggregated node dependencies.

To further enhance the alignment of aggregation patterns, ACKA chooses to integrate graph structural information into the kernel function through the Structure-Refined Gaussian Kernel (SRBF). SRBF activites the paired-wise node aggregation and informs the structure-aware similarity function determining how much influence neighboring nodes should have for knowledge distillation. SRBF kernel is defined as:

$$\mathcal{K}(\mathbf{Z}_i, \mathbf{Z}_j) = \exp\left(-\frac{1}{2\sigma^2} \parallel \hat{\mathbf{A}}_{ij}(\mathbf{Z}_i - \mathbf{Z}_j) \parallel_2^2\right), \quad (6)$$

where $\mathbf{Z}_i$ is the embedding of node i. The SRBF kernel ensures that node similarity computations are structure-aware, allowing the student to capture the adjacency-activated aggregation dynamics modeled by the teacher. To transfer the aggregation behavior of the teacher model to the student model, we introduce the ACKA loss function, which aligns the structure-enhanced kernelized representations between the teacher and student. The loss is formulated as:

$$\mathcal{L}_{ACKA} = 1 - \text{ACKA}(\text{SRBF}(\mathbf{Z}_T), \text{SRBF}(\mathbf{Z}_S)), \quad (7)$$

where $\mathbf{Z}_T \in \mathbb{R}^{b\times d_1}$ and $\mathbf{Z}_S \in \mathbb{R}^{b\times d_2}$ denote the node embeddings from GNN and MLP models, respectively, and $\text{SRBF}(\cdot)$ is the kernel function defined in Eq. 6.

From a theoretical perspective, ACKA can be interpreted as the upper bound of Maximum Mean Discrepancy (MMD) with an additional constant term [Zhou *et al.*, 2024b]. This implies that maximizing ACKA is equivalent to minimizing the upper bound of MMD between the teacher's aggregated embeddings and the student's transformed features. By transferring aggregation behavior rather than raw feature representations, ACKA provides an intuitive and effective distillation mechanism. Additionally, another advantage of ACKA is its dimension-independent design, which accommodates the different representation spaces of the teacher and student models. This is particularly important when the teacher (e.g., a Graph Transformer) and the student (e.g., an MLP) have significant architectural differences. By aligning kernelized aggregation patterns, ACKA ensures that the student model captures the teacher's structural aggregation patterns, even when their embeddings operate in different dimensions or scales.

Manifold Mixup Soft Matching Distillation. Considering the representative capacity differences between GNNs and MLP, we propose a novel manifold mixup soft matching distillation method in order to propagate the teacher GNN's insights of the augmented mixing node distributions into the MLP during the distillation process, improving the efficiency of knowledge transfer. The key components of this method include the generation of mixed representations through shared random shuffling and linear interpolation, and a temperature-shared KL divergence loss.

Suppose, the node embedding generated by the teacher GNN model is denoted as $\mathbf{Z}_T \in \mathbb{R}^{b\times d_1}$, where b and d_1 are the batch size and embedding dimensions of GNN, respectively, and MLP's node embedding is $\mathbf{Z}_S \in \mathbb{R}^{b\times d_2}$. To create mixed representations, we apply a random shuffle to the node embedding. Let $\mathbf{Z}'_T$ represent the shuffled version of $\mathbf{Z}_T$ obtained by randomly permuting the node indices. The mixed

embedding for the GT model is then computed as:

$$\mathbf{Z}_T^{mix} = \lambda \mathbf{Z}_T + (1-\lambda)\mathbf{Z}'_T, \tag{8}$$

where $\lambda \in [0,1]$ is a mixing coefficient drawn from a Beta distribution, $\lambda \sim \text{Beta}(\alpha, \alpha)$. The same shuffle indices are used to generate $\mathbf{Z}'_S$ from $\mathbf{Z}_S$, and the mixed embedding for the MLP model is:

$$\mathbf{Z}_S^{mix} = \lambda \mathbf{Z}_s + (1-\lambda)\mathbf{Z}'_S. \tag{9}$$

The mixed embeddings are fed into the classification heads of both models, yielding predicted logits $\hat{\mathcal{Y}}_T^{mix}$ and $\hat{\mathcal{Y}}_S^{mix}$ for the GNN and MLP models, respectively:

$$\hat{\mathcal{Y}}_T^{mix} = g_T(\mathbf{Z}_T^{mix}), \hat{\mathcal{Y}}_S^{mix} = g_S(\mathbf{Z}_S^{mix}) \tag{10}$$

In the prediction mimicking stage, we use the $\mathcal{Z}$-score logit standard distillation technique [Sun *et al.*, 2024] to alleviate the challenges of a lightweight student in predicting logits with a comparable range and variance as a cumbersome teacher, given the capacity gap between them. The objective function of our prediction soft alignment can presented by:

$$\begin{aligned}
\mathcal{L}_{mix} &= \mathcal{D}_{KL}(\phi(\mathcal{Z}(\hat{\mathcal{Y}}_T^{mix})/\tau), \phi(\mathcal{Z}(\hat{\mathcal{Y}}_S^{mix})/\tau)) \\
\mathcal{L}_{pred} &= \mathcal{D}_{KL}(\phi(\mathcal{Z}(\hat{\mathcal{Y}}_T)/\tau), \phi(\mathcal{Z}(\hat{\mathcal{Y}}_S)/\tau)) \\
\mathcal{L}_{logit} &= \mathcal{L}_{mix} + \mathcal{L}_{pred},
\end{aligned} \tag{11}$$

where $\mathcal{Z}(X) = \frac{X-\mu}{\sigma}$ is the $\mathcal{Z}$-score standardization, ϕ denotes the SoftMax function, τ is the shared-temperature (0.5 for all experiments), and $\hat{\mathcal{Y}}_T = g_T(\mathbf{Z}_T), \hat{\mathcal{Y}}_S = g_S(\mathbf{Z}_S)$ are the clean predict logits based their node embeddings, respectively. The mixup performs as graph rewiring augmentation which can improve the robustness of distillation. Furthermore, teaching the GNN's predictions of mixed data distributions to the MLP can somewhat mitigate logit collapses due to differences in model ability to make rigid predictive alignments (a problem that has been identified in the previous literature [Hao *et al.*, 2024; Lao *et al.*, 2023]).

The overall loss function for student MLP training is a weighted combination of classification task loss, ACKA loss, and logit distillation loss:

$$\mathcal{L}_S = \mathcal{L}_{task} + \beta \mathcal{L}_{ACKA} + \gamma \mathcal{L}_{logit}, \tag{12}$$

where $\mathcal{L}_{task}$ is the commonly used classification cross-entropy loss, β and γ are the tread-off hyper-parameters.

3.3 Algorithm Analysis

The pseudo-code of our proposed AMEND framework is illustrated in Algo. 1. The time complexity mainly depends on the self-attention module and for one layer is $\mathcal{O}(b(K+1)^2 d)$, where b, K, d denote the batch size, pre-aggregation hops, and hidden dimensions, respectively. In student training, the computational complexity primarily derives from ACKA and MLP, which can be formulated as: $\mathcal{O}(b^2 d + dL)$, where L is the student model depth. Manifold mixup is implemented through an efficient parameterization that requires only one additional forward process for the classification head, which has a negligible impact on the overall complexity. In addition, the teacher model and position encoding can be pre-trained and pre-computed offline, which improves the efficiency of our model training. In model inference, AMEND shares the same complexity with vanilla MLPs ($\mathcal{O}(dL)$), and its space complexity is $\mathcal{O}(d^2 L)$, enabling fast reasoning and lightweight deployment.

Algorithm 1 AMEND Algorithm

Input: graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, node feature matrix $\mathbf{X}$, and pre-computed position encoding $\mathbf{X}_{pe}$
Output: optimized parameters of the student MLP $\mathcal{S}$, predict node labels $\hat{\mathcal{Y}}$.

Model initialization and Dataset Partitioning.
Pretrain the teacher model $\mathcal{T}$ with cross-entropy loss.
#Student MLP Training
for Epochs **do**
  #Aggragation Context Preservation
  $\mathbf{Z}_T = \mathcal{T}(\mathbf{X}, \mathcal{E}, \mathbf{X}_{pe})$,
  $\mathbf{Z}_S = \mathcal{S}(\mathbf{X}, \mathbf{X}_{pe})$;
  #Aggregation-enhanced CKA
  $\mathcal{L}_{ACKA} \leftarrow \text{ACKA}(\mathbf{Z}_T, \mathbf{Z}_S)$ in Eq. 7;
  #Shared Manifold mixup
  $\mathbf{Z}_T^{mix} = \lambda \mathbf{Z}_T + (1-\lambda)\mathbf{Z}'_T$;
  $\mathbf{Z}_S^{mix} = \lambda \mathbf{Z}_S + (1-\lambda)\mathbf{Z}'_S$;
  $\hat{\mathcal{Y}}_T, \hat{\mathcal{Y}}_S \leftarrow g_T(\mathbf{Z}_T), g_S(\mathbf{Z}_S)$;
  $\hat{\mathcal{Y}}_T^{mix}, \hat{\mathcal{Y}}_S^{mix} \leftarrow g_T(\mathbf{Z}_T^{mix}), g_S(\mathbf{Z}_S^{mix})$;
  #Logit distillation
  $\mathcal{L}_{logit} = \mathcal{L}_{mix} + \mathcal{L}_{pred}$ in Eq. 11;
  #Overall loss compute
  $\mathcal{L}_S = \mathcal{L}_{task} + \beta \mathcal{L}_{ACKA} + \gamma \mathcal{L}_{logit}$ in Eq. 12;
  Gradient backward and model optimization.
end for
return $\mathcal{S}, \hat{\mathcal{Y}}$

4 Experiments

4.1 Experiments Setting

Datasets. To fully evaluate our proposed method, we use 8 public regular graph benchmarks [Yang *et al.*, 2021], i.e. Cora, Citeseer, Pubmed, Computer, Photo, Corafull, Coauthor-CS, Coauthor-Physics, and 4 large-scale graphs [Hu *et al.*, 2020], i.e., Ogbn-Arxiv, Aminer, Reddit, and Ogbn-Products. The details of these 12 datasets are in Appendix A. For each dataset, we follow the dataset protocol in [Chen *et al.*, 2023], where 6/2/2 of the nodes are used as training/validation/test sets, respectively.

We select GT as the teacher model for distillation because it presents a more challenging and representative case. GT leverages a global aggregation scope and an attention-based aggregation pattern, which demand that the student MLP replicate both its extensive receptive field and intricate aggregation mechanisms. These characteristics make GT an ideal testbed for evaluating the effectiveness of GNN distillation methods. In our experiments, we report the mean and standard deviation of ten separate runs. We employ accuracy to measure model performance, use validation data to select the optimal model, and report results for test data.

Baselines. Consistent with the comparative experimental setup of traditional knowledge distillation frameworks, we compare a variety of train-from-scratch baseline models, i.e., MLP, GNN-teacher (GCN [Kipf and Welling, 2016], GAT [Velickovic *et al.*, 2017], SAGE [Hamilton *et al.*, 2017], NAGphormer [Chen *et al.*, 2023]), and 3 state-of-the-art GNN-to-MLP methods, i.e., GLNN [Zhang *et al.*, 2022], NO-

Dataset	Cora	Citeseer	Pubmed	Computer	Photo	Corafull	CS	Physics
MLP	77.96±1.73	65.30±1.67	71.82±1.85	75.74±1.25	78.46±1.06	55.72±0.88	73.68±1.02	77.29±0.87
GCN	90.19±1.67	77.86±2.79	86.56±2.21	89.93±1.63	94.05±2.52	61.76±1.46	92.92±0.96	96.18±1.23
GAT	90.56±2.48	78.46±1.76	87.01±3.07	90.82±2.65	94.64±1.78	64.47±1.25	93.61±0.84	96.17±1.28
SAGE	90.78±2.46	78.61±3.02	88.31±2.17	90.04±1.28	94.77±2.10	67.24±1.73	93.87±0.97	96.58±1.49
NAGPhormer	91.01±2.30	78.31±2.18	89.83±0.96	91.35±1.62	95.68±2.47	70.51±1.59	95.75±0.94	96.68±1.25
GLNN	90.37±1.77	76.37±2.03	86.74±1.87	90.22±1.22	93.79±0.85	68.75±1.68	95.55±1.05	96.61±0.83
NOSMOG	90.49±1.57	76.58±2.34	88.23±0.96	90.40±3.02	94.97±1.25	69.28±1.06	95.56±0.94	96.45±1.21
VQGraph	90.19±0.97	76.30±1.15	88.13±0.57	91.17±0.88	93.07±1.21	69.33±1.06	95.66±0.96	96.88±0.78
Ours	**91.30±1.03**	**78.92±0.96**	**90.54±1.10**	**92.44±0.35**	**96.01±1.06**	**71.02±0.92**	**96.31±0.70**	**97.22±0.48**
Δ_{MLP}	↑13.34%	↑13.62%	↑18.72%	↑16.70%	↑17.55%	↑15.30%	↑22.63%	↑19.93%
Δ_{GNN}	↑0.52%	↑0.31%	↑2.23%	↑2.40%	↑1.24%	↑2.78%	↑2.44%	↑0.64%
Δ_{GT}	↑0.29%	↑0.61%	↑0.71%	↑1.20%	↑0.33%	↑0.51%	↑0.57%	↑0.54%
Δ_{GLNN}	↑0.93%	↑2.55%	↑3.80%	↑2.22%	↑2.22%	↑2.27%	↑0.76%	↑0.61%
Δ_{NOSMOG}	↑0.81%	↑2.34%	↑2.31%	↑2.04%	↑1.04%	↑1.74%	↑0.75%	↑0.77%
$\Delta_{VQGraph}$	↑1.11%	↑2.62%	↑2.41%	↑1.27%	↑2.94%	↑1.69%	↑0.65%	↑0.34%

Table 1: Performance on eight regular graphs. The top 5 rows report the performance of the teacher model with vanilla MLP, GNNs (i.e., GCN, GAT, SAGE), and GT. in the middle is the state-of-the-art distill-to-MLP methods and our AMEND results. Δ_X denotes the difference between the AMEND and others (SAGE for GNN), respectively. Results show accuracy(higher is better), best are highlighted in **bold**.

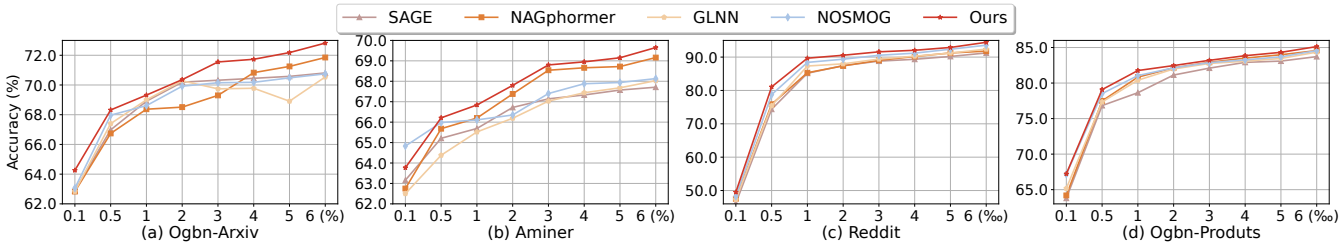


Figure 3: Results on four large-scale graphs with different training label rates. With increasing labeling rates, model performance improved in all cases. Among them, GT framework (NAGphormer) outperforms GNNs, and our AMEND (the red line) outperforms all teacher models and state-of-the-art distill-to-MLP methods overall.

SOMG [Tian *et al.*, 2023], and VQGraph [Yang *et al.*, 2024]. The details of baseline methods are in Appendix B.

4.2 Performance on Eight Regular Graphs

We compare the proposed AMEND to vanilla supervised MLP, GNN, GT, and state-of-the-art distill-to-MLP methods in the same experimental setting across eight datasets, with the results reported in Table 1. The results show that the vanilla MLP consistently exhibits the lowest performance, emphasizing the need for effective knowledge distillation from more complex models. GNN-based models (i.e., GCN, GAT, and SAGE) outperform the MLP thanks to the message passing for graph structure learning. The NAGphormer, representing the GT teacher, demonstrates superior performance compared to all GNNs, affirming the effectiveness of its global attention mechanisms and more sophisticated model architecture. Our proposed method consistently outperforms all other distill-to-MLP methods (GLNN, NOSMOG, VQGraph) across eight datasets. The Δ values indicate the performance gains of our method over the baseline models and other distill-to-MLP approaches, with significant improvements observed in several datasets: **1.79%** higher than NOSMOG on the Citeseer, **2.41%** higher than VQGraph on Pubmed, and **2.34%** higher than NOSMOG on CS, etc. These results confirm the effectiveness of our proposed AMEND method for GNN-to-MLP, which leverages the strengths of the graph transformer teacher and addresses the aggregation mechanism differences, resulting in substantial performance improvements and underscoring the potential for lightweight deployment and fast inference in graph data mining tasks.

4.3 Performance on Four Large-Scale Graphs

Figure 3 shows the performance comparison of our AMEND against SAGE, NAGphormer, GLNN, and NOSMOG across four large-scale graphs (i.e., Ogbn-Arxiv, Aminer, Reddit, Ogbn-Products.) in a few-shot setting with varying training label rates (For the first two datasets, we randomly selected two non-overlapping 10% nodes as the validation and test sets, respectively, and doubled 1% for the last two datasets.). The results indicate that the model performance steadily improves as the supervision increases, and our method almost surpasses all other distill-to-MLP approaches. For example, on Ogbn-Arxiv, our method achieves the highest accuracy at all label rates, with a notable improvement of approximately **5%** over NOSMOG at the 0.1% label rate. On Reddit and Ogbn-Products, our method maintains superior performance, especially at higher label rates (e.g., $\sim$**2%** higher than NOSMOG at 6‰ label rate on Ogbn-Products). These findings highlight the robustness and effectiveness of our approach, particularly in utilizing limited labeled data,

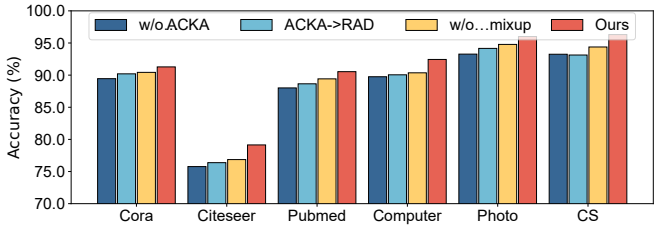
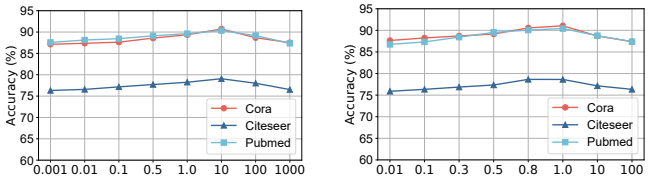
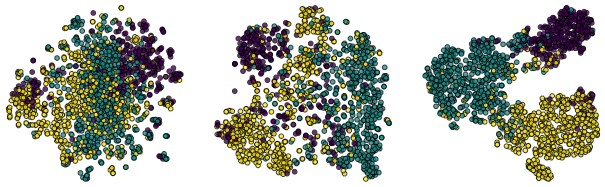


Figure 4: Ablation of ACKA and manifold mixup.



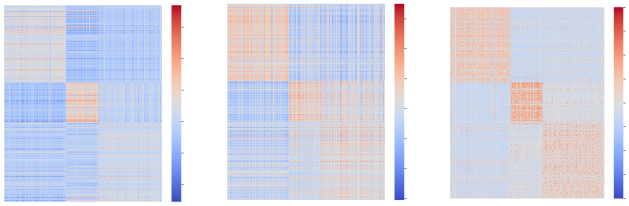
(a) Parameter β (b) Parameter γ

Figure 5: Hyper parameters sensitiveness.



(a) Raw Data (b) NOSMOG (c) Ours

Figure 6: Node embedding visualization on Pubmed.



(a) GCN→GLNN (b) SAGE→NOSMOG (c) GT→AMEND

Figure 7: The correlation heat map of node embedding between the teacher and student models.

underscoring its potential for scalable and efficient deployment in large-scale graph scenarios.

4.4 Ablation Study

In the ablation experiments, we investigate the impact of ACKA and manifold mixup modules within AMEND , and the results are reported in Figure 4, where ACKA→RAD represents we replace ACKA with the *Representational Similarity Distillation* (RAD) in NOSMOG [Tian *et al.*, 2023]. The results clearly demonstrate that removing either ACKA or manifold mixup significantly drops accuracy across all datasets. For example, on the Cora dataset, removing ACKA decreases accuracy from over 91% to approximately 89%. The impact of manifold mixup is also notable, as its removal causes a drop in performance across all datasets, such as from 96% to 93% on the Photo dataset. These results confirm that both ACKA and manifold mixup are critical components of our method, significantly contributing to its superior performance. Results underscored the importance of these modules in effectively transferring knowledge and enhancing model performance in graph data mining tasks.

4.5 Parameter Sensitive Analysis

In Figure 5, we explore the sensitivity of hyper parameters β and γ in overall objective function Eq. 12 on three citation graphs. β and γ represent the contributions of the ACKA and manifold mixup logit distillation, respectively. The results indicate that the optimal performance is achieved with $\beta = 10$ and $\gamma = 0.1$. According to the definition of $\mathcal{L}_{ACKA}$, its value range is $[0, 1]$. We monitored the values of each component of the loss function during training and found that, with $\beta = 10, \gamma = 0.1$, the scales of $\mathcal{L}_{ACKA}$ and $\mathcal{L}_{logit}$ were comparable to the task loss component $\mathcal{L}_{task}$, leading to optimal model convergence.

4.6 Visualization

To visually compare the node embeddings learned by different methods, we used the TSNE algorithm [Van der Maaten and Hinton, 2008] to create scatterplots of the test node representations from the Pubmed dataset, as shown in Figure 6. The figure demonstrates that the node embeddings extracted from our student model exhibit better class separability and intra-class compactness compared to those from the current state-of-the-art GNN-to-MLP approach. This indicates that AMEND effectively transfers the complex graph modeling knowledge from the GT teacher into an efficient MLP. Additionally, we visualize the node correlations between the local aggregation (GCN→GLNN and SAGE→NOSMOG) and global aggregation (GT→AMEND) in Fig. 7. Results demonstrate that our approach has a more regular and structured correlation with the teacher model than NOSMOG.

5 Conclusion

In this paper, we proposed the AMEND framework for effective and efficient knowledge transfer from GNNs to MLPs. The framework introduces a multi-scope aggregation context preservation strategy to enable the student MLP to preserve the teacher's broad and varying aggregation scopes effectively. Additionally, a pattern-guided alignment mechanism addresses aggregation pattern discrepancies so that the student MLP can accurately replicate the structural aggregation behaviors of the teacher GNN. We further incorporate a manifold mixup distillation approach to improve the efficiency and robustness of the student model by capturing the teacher's insights into mixed data distributions. Extensive experiments on 8 regular and 4 large-scale graph datasets, combined with ablation studies and visualization analyses, validate the superiority of the proposed method over existing baselines.

Acknowledgments

This work was partially supported by the Natural Science Foundation of Jiangsu Province (Grant No. BK20222003), the Collaborative Innovation Center of Novel Software Technology and Industrialization, and the Sino-German Institutes of Social Computing. The corresponding authors are Mingkai Lin (mingkai@nju.edu.cn) and Wenzhong Li (lwz@nju.edu.cn).

References

[Chen *et al.*, 2018] Jie Chen, Tengfei Ma, and Cao Xiao. Fastgcn: fast learning with graph convolutional networks via importance sampling. *arXiv preprint arXiv:1801.10247*, 2018.

[Chen *et al.*, 2022] Dexiong Chen, Leslie O'Bray, and Karsten Borgwardt. Structure-aware transformer for graph representation learning. In *International Conference on Machine Learning*, pages 3469–3489. PMLR, 2022.

[Chen *et al.*, 2023] Jinsong Chen, Kaiyuan Gao, Gaichao Li, and Kun He. Nagphormer: A tokenized graph transformer for node classification in large graphs. In *Proceedings of the International Conference on Learning Representations*, 2023.

[Deng *et al.*, 2024] Chenhui Deng, Zichao Yue, and Zhiru Zhang. Polynormer: Polynomial-expressive graph transformer in linear time. *arXiv preprint arXiv:2403.01232*, 2024.

[Dwivedi and Bresson, 2020] Vijay Prakash Dwivedi and Xavier Bresson. A generalization of transformer networks to graphs. *arXiv preprint arXiv:2012.09699*, 2020.

[Dwivedi *et al.*, 2023] Vijay Prakash Dwivedi, Chaitanya K Joshi, Anh Tuan Luu, Thomas Laurent, Yoshua Bengio, and Xavier Bresson. Benchmarking graph neural networks. *Journal of Machine Learning Research*, 24(43):1–48, 2023.

[Fountoulakis *et al.*, 2023] Kimon Fountoulakis, Amit Levi, Shenghao Yang, Aseem Baranwal, and Aukosh Jagannath. Graph attention retrospective. *Journal of Machine Learning Research*, 24(246):1–52, 2023.

[Hamilton *et al.*, 2017] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30, 2017.

[Hao *et al.*, 2024] Zhiwei Hao, Jianyuan Guo, Kai Han, Yehui Tang, Han Hu, Yunhe Wang, and Chang Xu. One-for-all: Bridge the gap between heterogeneous architectures in knowledge distillation. *Advances in Neural Information Processing Systems*, 36, 2024.

[He *et al.*, 2023] Liancheng He, Liang Bai, Xian Yang, Hangyuan Du, and Jiye Liang. High-order graph attention network. *Information Sciences*, 630:222–234, 2023.

[Hong *et al.*, 2021] Xiaobin Hong, Tong Zhang, Zhen Cui, Yuge Huang, Pengcheng Shen, Shaoxin Li, and Jian Yang. Graph game embedding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 7711–7720, 2021.

[Hong *et al.*, 2024a] Xiaobin Hong, Jiangyi Hu, Taishan Xu, Xiancheng Ren, Feng Wu, Xiangkai Ma, and Wenzhong Li. Magnet: Multilevel dynamic wavelet graph neural network for multivariate time series classification. *ACM Transactions on Knowledge Discovery from Data*, 19(1):1–22, 2024.

[Hong *et al.*, 2024b] Xiaobin Hong, Wenzhong Li, Chaoqun Wang, Mingkai Lin, and Sanglu Lu. Label attentive distillation for gnn-based graph classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 8499–8507, 2024.

[Hu *et al.*, 2020] Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: Datasets for machine learning on graphs. *Advances in neural information processing systems*, 33:22118–22133, 2020.

[Hussain *et al.*, 2022] Md Shamim Hussain, Mohammed J Zaki, and Dharmashankar Subramanian. Global self-attention as a replacement for graph convolution. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 655–665, 2022.

[Jha *et al.*, 2022] Kanchan Jha, Sriparna Saha, and Hiteshi Singh. Prediction of protein–protein interaction using graph neural networks. *Scientific Reports*, 12(1):8360, 2022.

[Kipf and Welling, 2016] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[Lao *et al.*, 2023] Shanshan Lao, Guanglu Song, Boxiao Liu, Yu Liu, and Yujiu Yang. Unikd: Universal knowledge distillation for mimicking homogeneous or heterogeneous object detectors. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6362–6372, 2023.

[Lin *et al.*, 2024] Mingkai Lin, Wenzhong Li, Xiaobin Hong, and Sanglu Lu. Scalable multi-source pre-training for graph neural networks. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 1292–1301, 2024.

[Lin *et al.*, 2025] Mingkai Lin, Xiaobin Hong, Wenzhong Li, and Sanglu Lu. Unified graph neural networks pre-training for multi-domain graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 12165–12173, 2025.

[Liu *et al.*, 2020] Yi Liu, Hao Yuan, Lei Cai, and Shuiwang Ji. Deep learning of high-order interactions for protein interface prediction. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 679–687, 2020.

[Rampášek *et al.*, 2022] Ladislav Rampášek, Michael Galkin, Vijay Prakash Dwivedi, Anh Tuan Luu, Guy Wolf, and Dominique Beaini. Recipe for a general, powerful, scalable graph transformer. *Advances in Neural Information Processing Systems*, 35:14501–14515, 2022.

[Sharma *et al.*, 2024] Kartik Sharma, Yeon-Chang Lee, Sivagami Nambi, Aditya Salian, Shlok Shah, Sang-Wook Kim, and Srijan Kumar. A survey of graph neural networks for social recommender systems. *ACM Computing Surveys*, 56(10):1–34, 2024.

[Sun *et al.*, 2024] Shangquan Sun, Wenqi Ren, Jingzhi Li, Rui Wang, and Xiaochun Cao. Logit standardization in knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15731–15740, 2024.

[Tian *et al.*, 2023] Yijun Tian, Chuxu Zhang, Zhichun Guo, Xiangliang Zhang, and Nitesh Chawla. Learning MLPs on graphs: A unified view of effectiveness, robustness, and efficiency. In *International Conference on Learning Representations*, 2023.

[Van der Maaten and Hinton, 2008] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[Velickovic *et al.*, 2017] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, Yoshua Bengio, et al. Graph attention networks. *stat*, 1050(20):10–48550, 2017.

[Wang *et al.*, 2025] Xiaoli Wang, Anqi Huang, Yongli Wang, Guanzhou Ke, Xiaobin Hong, and Jun Liu. Global-semantic alignment distillation for partial multi-view classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 21287–21295, 2025.

[Wu *et al.*, 2021] Zhanghao Wu, Paras Jain, Matthew Wright, Azalia Mirhoseini, Joseph E Gonzalez, and Ion Stoica. Representing long-range context for graph neural networks with global attention. *Advances in Neural Information Processing Systems*, 34:13266–13279, 2021.

[Wu *et al.*, 2023] Lirong Wu, Haitao Lin, Yufei Huang, Tianyu Fan, and Stan Z Li. Extracting low-/high-frequency knowledge from graph neural networks and injecting it into mlps: An effective gnn-to-mlp distillation framework. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 10351–10360, 2023.

[Xia *et al.*, 2022] Jun Xia, Lirong Wu, Jintao Chen, Bozhen Hu, and Stan Z Li. Simgrace: A simple framework for graph contrastive learning without data augmentation. In *Proceedings of the Web Conference*, pages 1070–1079, 2022.

[Yan *et al.*, 2020] Bencheng Yan, Chaokun Wang, Gaoyang Guo, and Yunkai Lou. Tinygnn: Learning efficient graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1848–1856, 2020.

[Yang *et al.*, 2020] Yiding Yang, Jiayan Qiu, Mingli Song, Dacheng Tao, and Xinchao Wang. Distilling knowledge from graph convolutional networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7074–7083, 2020.

[Yang *et al.*, 2021] Cheng Yang, Jiawei Liu, and Chuan Shi. Extract the knowledge of graph neural networks and go beyond it: An effective knowledge distillation framework. In *Proceedings of the web conference 2021*, pages 1227–1237, 2021.

[Yang *et al.*, 2023] Cheng Yang, Yuxin Guo, Yao Xu, Chuan Shi, Jiawei Liu, Chunchen Wang, Xin Li, Ning Guo, and Hongzhi Yin. Learning to distill graph neural networks. In *Proceedings of the sixteenth ACM international conference on web search and data mining*, pages 123–131, 2023.

[Yang *et al.*, 2024] Ling Yang, Ye Tian, Minkai Xu, Zhongyi Liu, Shenda Hong, Wei Qu, Wentao Zhang, CUI Bin, Muhan Zhang, and Jure Leskovec. Vqgraph: Rethinking graph representation space for bridging gnns and mlps. In *The Twelfth International Conference on Learning Representations*, 2024.

[Ying *et al.*, 2021] Chengxuan Ying, Tianle Cai, Shengjie Luo, Shuxin Zheng, Guolin Ke, Di He, Yanming Shen, and Tie-Yan Liu. Do transformers really perform badly for graph representation? *Advances in neural information processing systems*, 34:28877–28888, 2021.

[Yun *et al.*, 2019] Seongjun Yun, Minbyul Jeong, Raehyun Kim, Jaewoo Kang, and Hyunwoo J Kim. Graph transformer networks. *Advances in neural information processing systems*, 32, 2019.

[Zhang *et al.*, 2020] Wentao Zhang, Xupeng Miao, Yingxia Shao, Jiawei Jiang, Lei Chen, Olivier Ruas, and Bin Cui. Reliable data distillation on graph convolutional network. In *Proceedings of the 2020 ACM SIGMOD international conference on management of data*, pages 1399–1414, 2020.

[Zhang *et al.*, 2022] Shichang Zhang, Yozen Liu, Yizhou Sun, and Neil Shah. Graph-less neural networks: Teaching old mlps new tricks via distillation. In *International Conference on Learning Representations*, 2022.

[Zhao *et al.*, 2021] Jianan Zhao, Chaozhuo Li, Qianlong Wen, Yiqi Wang, Yuming Liu, Hao Sun, Xing Xie, and Yanfang Ye. Gophormer: Ego-graph transformer for node classification. *arXiv preprint arXiv:2110.13094*, 2021.

[Zhou *et al.*, 2024a] Zijie Zhou, Zhaoqi Lu, Xuekai Wei, Rongqin Chen, Shenghui Zhang, Pak Lon Ip, et al. Tokenphormer: Structure-aware multi-token graph transformer for node classification. *arXiv preprint arXiv:2412.15302*, 2024.

[Zhou *et al.*, 2024b] Zikai Zhou, Yunhang Shen, Shitong Shao, Huanran Chen, Linrui Gong, and Shaohui Lin. Rethinking centered kernel alignment in knowledge distillation. *arXiv preprint arXiv:2401.11824*, 2024.