NAAM: Node-Aware Attention Mechanism for Distilling GNNs-to-MLP (Student Abstract)

Itsuki Nakayama, Makoto Onizuka

Osaka University, 1–5 Yamadaoka, Suita, Osaka, Japan
{nakayama.itsuki, onizuka}@ist.osaka-u.ac.jp

Abstract

Recently, researchers have focused on methods that not only distill knowledge from a Graph Neural Network (GNN) into a Multi-Layer Perceptron (MLP) but also leverage multiple teacher GNNs. However, existing methods assign a single attention weight to each teacher GNN. We propose a Node-Aware Attention Mechanism (NAAM) that flexibly adjusts the attention weight for each node to leverage multiple GNNs fully. Experimental results show that NAAM outperforms existing GNN-to-MLP methods. our source code is available at: https://github.com/NakayamaItsuki/NAAM.

Introduction

Graph Neural Networks (GNNs) have achieved significant success in various graph-based machine learning tasks. However, in real-world applications, the slow inference speed of GNNs poses a major challenge. This issue stems from the need for GNNs to fetch from memory not only the information of the target node but also that of its neighboring nodes during inference. Several approaches employ knowledge distillation from a GNN to a Multi-Layer Perceptron (MLP), achieving comparable or even better accuracy than the GNN. Recent research, such as MTAAM (Yang et al. 2024), leverages multiple teacher GNNs to improve the accuracy of student MLP. However, MTAAM assigns a single attention weight to each teacher GNN, assuming that all GNNs have the same importance for all nodes in the graph. We propose a Node-Aware Attention Mechanism (NAAM) that leverages the strengths of multiple GNNs at a node level by flexibly adjusting the attention weight for each node.

Methodology

An overview of the proposed model is shown in Figure 1.

Notations. We focus on a node classification task with the number of classes denoted as C. The set of nodes is represented by $\mathcal{V}$ and the total number of nodes is denoted as N. There are K pre-trained teacher GNNs, each labeled as t_k where $k = 1, 2, \ldots, K$. Its output is denoted by $\mathbf{Z}^{t_k} \in \mathbb{R}^{N \times C}$, and the output of the student MLP is represented as $\mathbf{Z}^{s} \in \mathbb{R}^{N \times C}$.

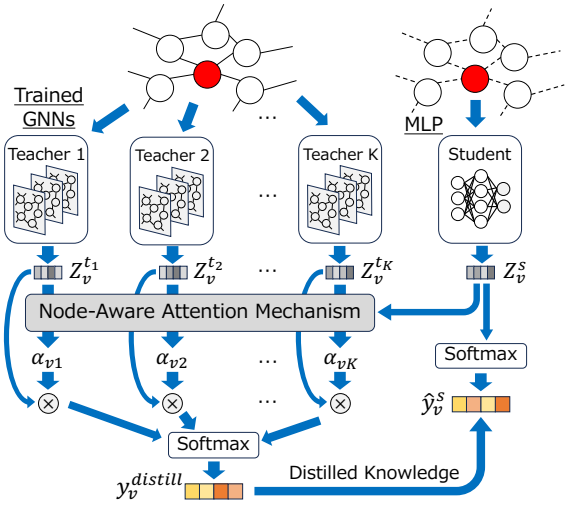

Figure 1: Overview of the proposed method.

Formulation of Knowledge Distillation Loss. The training of the student MLP through knowledge distillation is achieved by minimizing the loss $\mathcal{L}_{\text{MLP}}$ in Equation (1). Specifically, the first term, denoted as the Cross-Entropy loss $\mathcal{L}_{\text{CE}}$, makes predicted labels $\hat{\mathbf{y}}_v^s$ of the student MLP closer to true labels $\mathbf{y}_v$, while the second term, represented by the Kullback-Leibler divergence loss $\mathcal{L}_{\text{KL}}$, aligns the estimated label distribution $\hat{\mathbf{y}}^s$ of the student MLP with the soft label distribution $\mathbf{y}^{\text{distill}}$, which integrates the outputs from multiple GNNs (to be described later). The temperature parameter T and the balancing parameter λ_{kd} are used to appropriately scale and balance these two terms.

$$\begin{aligned}\mathcal{L}_{\text{MLP}} = (1-\lambda_{\text{kd}}) \textstyle\sum_{v \in \mathcal{V}_{\text{train}}} \mathcal{L}_{\text{CE}}(\hat{\mathbf{y}}_v^s, \mathbf{y}_v) \\ + \lambda_{\text{kd}} \textstyle\sum_{v \in \mathcal{V}} \mathcal{L}_{\text{KL}}(\hat{\mathbf{y}}_v^s/T, \mathbf{y}_v^{\text{distill}}/T)\end{aligned} \tag{1}$$

Formulation of the Node-Aware Attention Mechanism. We introduce node-aware attention α_{vk} by multiplying it with the embeddings $\mathbf{Z}_v^{t_k}$ obtained from each of the K teacher GNNs. This process generates the soft labels $\mathbf{y}^{\text{distill}}$ for each node v, which are then used to train the student MLP. This attention mechanism allows for flexible adjustment of the importance of each teacher GNN at node level on the learning of the student MLP.

$$\mathbf{y}_v^{\text{distill}} = \text{softmax}\left(\textstyle\sum_{k=1}^{K} \alpha_{vk} \mathbf{Z}_v^{t_k}\right) \tag{2}$$

Datasets	Eval	Non KD methods		KD methods			Δ_{GLNN*}	Δ_{MTAAM}
		GNN*	MLP	GLNN*	MTAAM	NAAM		
Cora	prod	**79.7**	58.7	74.0	73.5	75.0	↑ 1.35%	↑ 2.04%
	ind	**81.4 ± 2.0**	58.6 ± 1.6	70.3 ± 1.8	68.2 ± 1.7	71.2 ± 1.8	↑ 1.28%	↑ 4.40%
	tran	78.1 ± 1.6	58.7 ± 1.8	77.8 ± 2.5	78.7 ± 2.2	**78.8 ± 2.2**	↑ 1.29%	↑ 0.13%
Citeseer	prod	70.3	58.0	70.3	70.0	**71.4**	↑ 1.56%	↑ 2.00%
	ind	**72.4 ± 1.4**	57.6 ± 1.6	70.1 ± 2.0	69.3 ± 1.9	70.8 ± 2.3	↑ 1.00%	↑ 2.16%
	tran	68.1 ± 1.4	58.3 ± 1.9	70.5 ± 1.6	70.6 ± 2.0	**71.9 ± 2.3**	↑ 1.99%	↑ 1.84%
Pubmed	prod	75.3	68.9	75.5	75.2	**76.4**	↑ 1.19%	↑ 1.60%
	ind	76.1 ± 2.1	68.8 ± 1.6	75.2 ± 2.1	75.1 ± 2.1	**76.3 ± 2.2**	↑ 1.46%	↑ 1.60%
	tran	74.5 ± 2.0	68.9 ± 1.5	75.8 ± 2.2	75.4 ± 2.4	**76.5 ± 2.4**	↑ 0.92%	↑ 1.46%
A-computer	prod	**83.3**	66.9	82.0	81.9	82.9	↑ 1.10%	↑ 1.22%
	ind	**83.5 ± 1.2**	67.1 ± 2.5	80.4 ± 1.3	80.6 ± 1.6	81.5 ± 1.4	↑ 1.37%	↑ 1.12%
	tran	83.0 ± 1.3	66.7 ± 2.7	83.5 ± 1.7	83.3 ± 2.0	**84.3 ± 1.6**	↑ 0.96%	↑ 1.20%
A-photo	prod	90.5	76.9	90.4	90.7	**90.8**	↑ 0.44%	↑ 0.11%
	ind	**90.7 ± 0.8**	77.2 ± 1.5	89.2 ± 1.4	89.4 ± 0.8	89.4 ± 0.9	↑ 0.22%	0.00%
	tran	90.2 ± 0.8	76.7 ± 1.8	91.6 ± 1.4	92.1 ± 1.0	**92.3 ± 0.9**	↑ 0.76%	↑ 0.22%

Table 1: Comparison of knowledge distillation methods. The "GNN*" represents the highest test accuracy among the GNNs, and the "GLNN*" represents the highest test accuracy when each GNN is used as a teacher. Bolded text indicates the best results, and underlined text indicates the second-best results.

Learnable Attention Computation Algorithm. Negative Kullback-Leibler divergence is used as the scoring metric, and softmax is applied to calculate the Attention $\boldsymbol{\alpha}_v$. To achieve more effective Attention weights, learnable weights $\mathbf{W}_s, \mathbf{W}_t \in \mathbb{R}^{C' \times C}$ and biases $\mathbf{b}_s, \mathbf{b}_t \in \mathbb{R}^{C'}$ are employed to linearly transform the outputs of both the GNNs and the MLP.

$$s_{vk} = -D_{\mathrm{KL}}(\mathbf{W}_s \mathbf{Z}_v^s + \mathbf{b}_s \parallel \mathbf{W}_t \mathbf{Z}_v^{t_k} + \mathbf{b}_t) \quad (3)$$

$$\boldsymbol{\alpha}_v = \mathrm{softmax}\left([s_{v1}, s_{v2}, \cdots, s_{vK}]\right) \quad (4)$$

Optimization of the Attention Mechanism. In addition to the cross-entropy loss $\mathcal{L}_{\text{val}}$ on validation data, a smoothing loss $\mathcal{L}_{\text{smooth}}$ is introduced (in Equation (7)) to mitigate extreme fluctuations in Attention weights. Furthermore, the optimization of the Attention Mechanism is performed at the end of every training epoch. This allows for dynamic adaptation to the evolving learning landscape and better integration of knowledge from the teacher models.

$$\mathcal{L}_{\text{val}} = \textstyle\sum_{v \in \mathcal{V}_{\text{val}}} \mathcal{L}_{\text{CE}}(\mathbf{y}_v^{\text{distill}}, \mathbf{y}_v) \quad (5)$$

$$\mathcal{L}_{\text{smooth}} = \textstyle\sum_{v \in \mathcal{V}} \|\boldsymbol{\alpha}_v^{\text{cur}} - \boldsymbol{\alpha}_v^{\text{prev}}\|^2 \quad (6)$$

$$\mathcal{L}_{\text{att}} = \mathcal{L}_{\text{val}} + \lambda_{\text{smooth}} \mathcal{L}_{\text{smooth}} \quad (7)$$

Experimental Evaluation

Datasets. By following the MTAAM study (Yang et al. 2024), we utilize five publicly available benchmark datasets: Cora, Citeseer, Pubmed, A-computer, and A-photo.

Settings. The student MLP is configured with two layers and a hidden dimension of 64. Parameters for the Attention Mechanism are set as follows: $T = 1$, $\lambda_{\text{kd}} = 0.0$, $\lambda_{\text{smooth}} = [0, 3]$, and $C' = 64$. We employ five well-known teacher GNN models: GCN (Kipf and Welling 2017), GAT (Veličković et al. 2018), SGC (Wu et al. 2019), APPNP (Gasteiger, Bojchevski, and Günnemann 2019), GCNII (Chen et al. 2020). We conducted experiments in an inductive setting (50% of test nodes were unseen during training) to evaluate the accuracy in real-world scenarios.

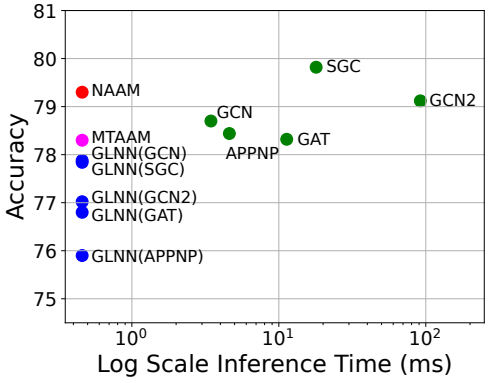

Figure 2: Accuracy vs. Inference Time.

Compared Baselines. To demonstrate the effectiveness of our method, we compare it with GLNN (Zhang et al. 2022) and MTAAM. GLNN employs a single teacher GNN for knowledge distillation, while MTAAM utilizes multiple teacher GNNs for the distillation process. In addition, all KD methods use the above five GNN models as teachers

Experimental Results. Table 1 shows that NAAM achieves higher accuracy than baseline methods. By leveraging multiple GNNs, our approach captures structural information in graph data without explicitly including the graph structure in the input, producing a structurally informed MLP. This demonstrates the effectiveness of our node-aware attention mechanism in enhancing GNN-to-MLP distillation.

Comparison of Inference Speed and Accuracy. Figure 2 compares average inference speed and accuracy across all datasets, showing that NAAM is more accurate than other KD methods and faster than GNN methods, making it a viable option for real-world applications balancing performance and computational demands.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Number JP20H00583.

References

Chen, M.; Wei, Z.; Huang, Z.; Ding, B.; and Li, Y. 2020. Simple and Deep Graph Convolutional Networks. In *Proceedings of the ICML*.

Gasteiger, J.; Bojchevski, A.; and Günnemann, S. 2019. Combining Neural Networks with Personalized PageRank for Classification on Graphs. In *Proceedings of the ICLR*.

Kipf, T. N.; and Welling, M. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *Proceedings of the ICLR*.

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2018. Graph Attention Networks. In *Proceedings of the ICLR*.

Wu, F.; Souza, A.; Zhang, T.; Fifty, C.; Yu, T.; and Weinberger, K. 2019. Simplifying Graph Convolutional Networks. In *Proceedings of the ICML*.

Yang, B.-W.; Chang, M.-Y.; Lu, C.-H.; and Shen, C.-Y. 2024. Two Heads are Better than One: Teaching MLPs with Multiple Graph Neural Networks via Knowledge Distillation. In *Proceedings of DASFAA*.

Zhang, S.; Liu, Y.; Sun, Y.; and Shah, N. 2022. Graph-less Neural Networks: Teaching Old MLPs New Tricks via Distillation. In *Proceedings of the ICLR*.