Can Pseudo-Label Be More Reliable? A Simple yet Effective Topology-Aware Graph Self-Training Method

Gen Liu[1], Zhongying Zhao[1*], Hui Zhou[1], Chao Li[2], Qingtian Zeng[1*]

[1]College of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao 266590, China
[2]College of Electronic and Information Engineering, Shandong University of Science and Technology, Qingdao 266590, China
lg97@sdust.edu.cn, zzysuin@163.com, zhouhui1026@foxmail.com, lichao@sdust.edu.cn, qtzeng@163.com

Abstract

Graph Neural Networks (GNNs) have demonstrated impressive success across a range of graph-based tasks. However, their performance in node classification typically relies on abundant high-quality labeled data that are difficult to obtain in practice. Self-training emerges as a promising solution to tackle the issue of label scarcity. Most existing studies in this direction mainly rely on classification scores to explore high-confidence unlabeled samples. Nevertheless, these methods often lead to false positive samples, which hinders the capability of GNNs. To this end, we propose a simple yet effective **T**opology-**A**ware **G**raph **S**elf-**T**raining (**TA-GST**) method. Specifically, we first explore the origin of false positives in pseudo-labeled samples. We then design a topology-aware scoring method, which considers both the classification score and connectivity pattern to enhance the reliability of pseudo-labeled samples. Besides, we depart TA-GST from the traditional teacher-student pattern and simplify it in an end-to-end manner. Extensive experiments on seven real-world datasets demonstrate the effectiveness of our method.

Code —
https://github.com/ZZY-GraphMiningLab/TA-GST

Introduction

Graphs are widely used to model various real-world applications, such as fraud detection (Peng et al. 2024), recommendation systems (Luo, Liu, and Pan 2024), and traffic network forecasting (Luo et al. 2023a). Graph Neural Networks, also known as deep learning on graphs, have achieved remarkable success in various graph analysis tasks (Yang et al. 2023; Xu et al. 2023). However, their performance typically relies on large amounts of labeled data, especially in the task of semi-supervised node classification (Liu et al. 2022). Once confronted with the decrease of labeled data, the effectiveness of GNNs notably diminishes (Singh et al. 2023).

As one of the promising approaches, self-training leverages vast amounts of unlabeled data to alleviate the scarcity of labeled data. Its core idea is to pseudo-label reliable unlabeled samples (Zhu and Goldberg 2022). Specifically, it first trains the teacher model on the available labeled samples. Subsequently, it leverages the predictions of the teacher model to pseudo-label unlabeled samples. Finally, it trains the student model on the enlarged data until convergence (Zoph et al. 2020). In the sample-independent scenarios (e.g., image classification and text classification), self-training assumes that the higher the classification score, the more reliable the prediction. In essence, higher-classification-score samples authentically exhibit analogous characteristics to the labeled samples (Yang et al. 2022; Mukherjee and Awadallah 2020). However, this mechanism yields less satisfactory results in the task of node classification (Juan, Peng, and Wang 2021). That is because it introduces false positive samples, which means that some samples are wrongly labeled. Thus, it raises a fundamental question: "***What is the origin of false positive samples in graph self-training?***"

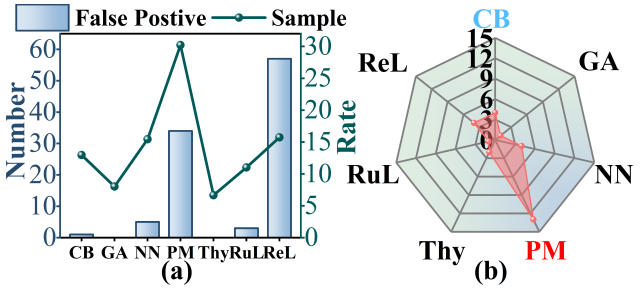

Figure 1: (a) The distribution of false positives and sample rate. (b) The label distribution of neighbors for a false positive sample (v_{733}), where blue category denotes its ground-truth label and the red category represents its pseudo-label.

As a motivating example, we adopt GCN as a baseline model and investigate the false positive samples on Cora dataset. We take a deep analysis of the experimental results from the following two perspectives.

(1) Distribution of false positive samples (Figure 1 (a)). It can be observed that the distribution of false positive samples is not linearly correlated with the proportions of various categories. Furthermore, the experimental results exhibit that false positives exist in both majority and minority categories, and their quantity remains uncorrelated with the proportion of a category.

(2) Connectivity pattern of the false positive sample

*Corresponding Authors.

(Figure 1 (b)). We define the connectivity pattern of a node as the label distribution of its neighbors. It becomes evident that the neighbors of false positives typically belong to a different, or alien category. GNNs excel in learning node representations by message passing and adhere to the homophily assumption (Li et al. 2023), wherein connected nodes are inclined to share the same label. However, intricate connectivity patterns inevitably introduce noise, resulting in false positive samples.

In light of the above observations, we propose a novel **T**opology-**A**ware **G**raph **S**elf-**T**raining (**TA-GST**) method, which identifies reliable pseudo-labels by considering both the classification score and label distribution of neighbors. Specifically, it begins by determining the maximum term of normalized probability as the classification score for each unlabeled node. In the semi-supervised learning paradigm, the labeled samples are limited, making it difficult to estimate the label distribution of each node. Thus, the proposed TA-GST leverages the ***soft label*** to address the above problem. Subsequently, a statistical analysis of these distributions is performed to evaluate the connectivity pattern for the unlabeled nodes, facilitating the scoring of individual nodes based on their connectivity patterns. Following this, the method assesses the importance of both the classification score and the score of the connectivity pattern to establish the confidence score for each unlabeled node. These confidence scores are then sorted, and the top-k scored samples are selected as high-reliability samples to enlarge the training set. In particular, the proposed TA-GST does not explicitly employ the teacher-student pattern. Instead, the model training involves two phases. It first trains the model only on the available labeled samples and then directly leverages the model to label reliable pseudo-labels for successive training. This variant enables graph self-training in an end-to-end manner.

The main contributions of this work are as follows.

- We propose a simple yet effective graph self-training method (TA-GST). It considers both the classification score and connectivity pattern, thereby enhancing the reliability of the pseudo-labels.
- Different from the traditional teacher-student paradigm, we devise a self-pseudo-labeling method. It not only facilitates an end-to-end graph self-training but also simplifies the training process.
- We conduct comprehensive experiments on seven real-world datasets. The experimental results fully demonstrate that TA-GST significantly outperforms state-of-the-art baselines.

Related Work

Graph Neural Networks

Graph Neural Networks characterize the structural semantics via the message-passing mechanism. Bruna et al. (Bruna et al. 2014) first introduced the convolution operation from Euclidean data to graph-structured data. However, subject to computational complexity, it cannot be generalized to large-scale application scenarios. Hence, Defferrard et al. (Defferrard, Bresson, and Vandergheynst 2016) leveraged Chebyshev polynomials to approximate the convolution kernel, simplifying the convolution operation on the graphs. Kipf et al. (Kipf and Welling 2016) applied the first-order approximation of the convolution kernel in the frequency domain to further simplify the graph convolution operation. Duvenaud et al. (Duvenaud et al. 2015) generalized the graph convolution operation to an arbitrary-shaped graph by learning various parameters from nodes of varied degrees. Hamilton et al. (Hamilton, Ying, and Leskovec 2017) proposed GraphSAGE, a graph neural network based on neighbor sampling, which investigated the fusion of neighboring information through various aggregation functions. Veličković et al. (Veličković et al. 2018) introduced the attention mechanism into the graph neural network and proposed the Graph Attention Network (GAT), which learns node representations by performing attention-weighted aggregation on neighbors. Recently, researchers also explored various techniques, including graph diffusion convolution (Li et al. 2025), graph foundation model (Liu et al. 2023), etc.

Self-Training on Graphs

Self-training is an effective semi-supervised method. The researchers have tried to generalize self-training to GNN-based models, with the goal of leveraging abundant unlabeled data on graphs. Wang et al. (Wang et al. 2021) considered that vanilla self-training on graphs ignored the low-confidence but high-accuracy predictions, resulting in inferior performance. Hence, they proposed a confidence-calibrated graph self-training method CaGCN-st. Zhou et al. (Zhou et al. 2023) devised a stabilized pseudo-labeling method and a negative sampling regularizer to consequently boost the performance of self-training on graphs. Liu et al. (Liu et al. 2022) explored the issue of distribution shift from the perspective of information gain and proposed a creative loss to improve the quality of pseudo-labels. Wang et al. (Wang, Zhao, and Wang 2024) proposed a distribution-consistent graph self-training method to identify the pseudo-labeled samples that are capable of redeeming the distribution discrepancy. Juan et al. (Juan, Peng, and Wang 2025) identified the reliability of pseudo-labels from the perspective of uncertainty.

Proposed Method

Preliminary

Let $\mathcal{G} = (\boldsymbol{\mathcal{V}}, \boldsymbol{\mathcal{E}}, \boldsymbol{X})$ be an attributed graph, where $\boldsymbol{\mathcal{V}}$ and $\boldsymbol{\mathcal{E}}$ represent the sets of nodes and edges, respectively. $\boldsymbol{X} \in \mathbb{R}^{|\boldsymbol{\mathcal{V}}| \times d}$ is the feature matrix, where d represents the dimension of node feature. $\boldsymbol{A} \in \mathbb{R}^{|\boldsymbol{\mathcal{V}}| \times |\boldsymbol{\mathcal{V}}|}$ is the adjacency matrix, where $\boldsymbol{A}_{ij} = 1$ indicates that there is an edge between node v_i and v_j, while $\boldsymbol{A}_{ij} = 0$ otherwise. In the setting of semi-supervised node classification, we only have access to scarce labeled nodes $\boldsymbol{\mathcal{V}}^L$ with potential labels $\boldsymbol{\mathcal{Y}}$ and massive unlabeled nodes $\boldsymbol{\mathcal{V}}^U$, where $|\boldsymbol{\mathcal{V}}^L| << |\boldsymbol{\mathcal{V}}^U|$.

Framework

We propose the **T**opology-**A**ware **G**raph **S**elf-**T**raining (**TA-GST**) for semi-supervised node classification. As illustrated

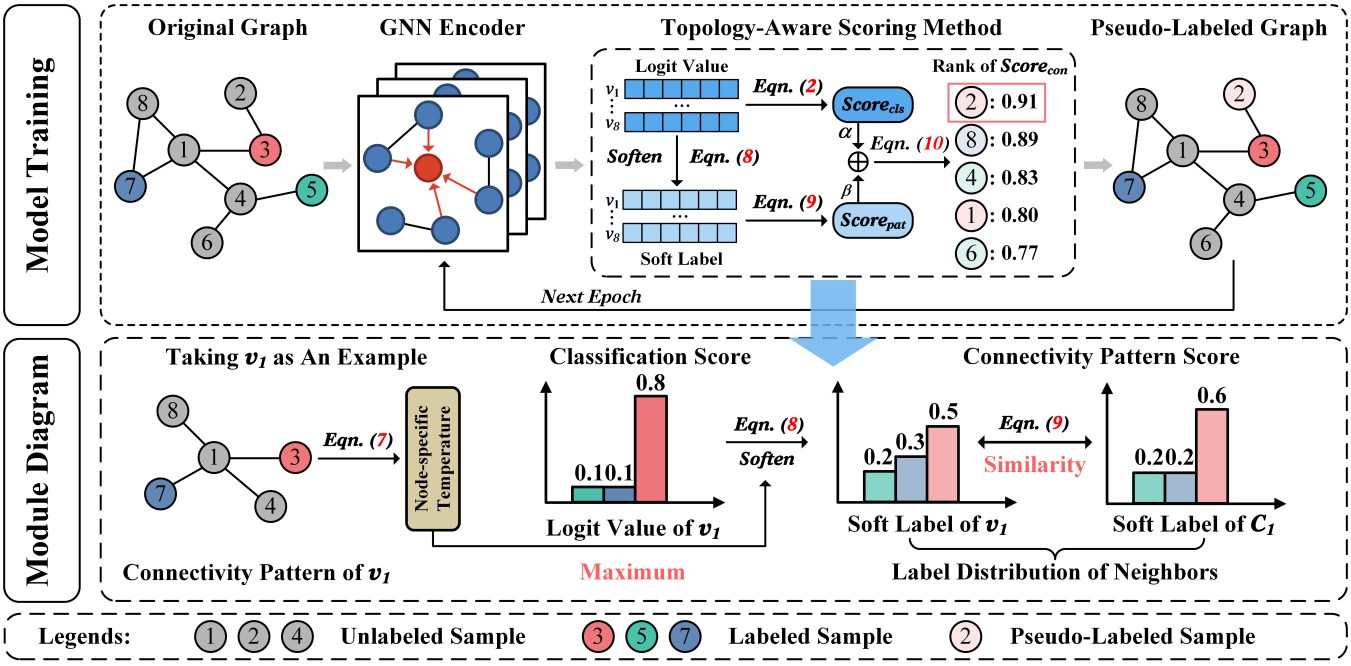

Figure 2: The framework of our proposed TA-GST.

in Figure 2, the framework of the proposed method is composed of four parts. Initially, it feeds the topological structure and feature matrix into the GNN-based model. Subsequently, the GNN-based model proceeds with information aggregation. Then, it leverages the learned node representations to evaluate the scoring criterion. In particular, both the classification score and the label distribution of neighbors are considered to identify reliable unlabeled samples, differing from vanilla graph self-training methods. Afterward, it sorts the confidence scores and labels the top-k scored unlabeled samples. Finally, the available label information is updated, and the follow-up model training is performed on the enlarged dataset. Notably, the proposed TA-GST does not explicitly identify the teacher and student model but segments the complete training process into two phases. In the first phase, it trains the model only on the available labeled samples. In the second phase, it directly leverages the model to label reliable pseudo-labels for successive training. This variant simplifies the training process and performs the self-training in an end-to-end manner. The details of our proposed TA-GST are introduced in the following subsections.

Topology-Aware Scoring Method

We devise a topology-aware scoring method to identify reliable pseudo-labels. In detail, it measures the confidence of an unlabeled node from two perspectives, including the classification probability and the connectivity pattern. This enables the full utilization of topological information while ensuring that the pseudo-labels remain highly reliable. Specifically, it first estimates the classification score and the predicted label of a node via Eqn. (1) to Eqn. (3).

$$\boldsymbol{Z} = f_\theta(\boldsymbol{X}, \boldsymbol{A}), \tag{1}$$

$$\mathcal{S}_v^{cls} = max(\boldsymbol{Z}_v = [p_1, p_2, ..., p_n]), \tag{2}$$

$$y_v = argmax(\boldsymbol{Z}_v = [p_1, p_2, ..., p_n]), \tag{3}$$

where f_θ is the GNN-based model, θ are the trainable parameters and $\boldsymbol{Z}$ is the output embedding matrix. For the task of node classification, $\boldsymbol{Z}_v$ represents the logit vector of node v, p_i denotes the probability of node v being classified into category $\boldsymbol{\mathcal{Y}}_i$, n represents the number of categories, $\mathcal{S}_v^{cls}$ indicates the classification score of node v, and y_v represents the classification result of node v.

Subsequently, the proposed method incorporates the connectivity pattern as the criterion for measuring the confidence of the pseudo-label. It defines the connectivity pattern of a node as the label distribution of its neighbors. Meanwhile, two definitions are introduced, i.e., *neighboring label distribution* and *class-wise connectivity pattern*, to evaluate the node-level and class-level connectivity pattern, respectively.

Definition 1. **Neighboring Label Distribution Matrix.** Let $\boldsymbol{\mathcal{D}} \in \mathbb{R}^{|\boldsymbol{\mathcal{V}}| \times |\boldsymbol{\mathcal{Y}}|}$ describe the node-wise label distribution of neighbors. Then, it is defined as Eqn. (4).

$$\boldsymbol{\mathcal{D}}_{i,j} = \frac{|\{v \in \boldsymbol{\mathcal{N}}(v_i) \cup \{v_i\} \mid y_v = \boldsymbol{\mathcal{Y}}_j\}|}{d_i + 1}, \tag{4}$$

where $\boldsymbol{\mathcal{N}}(v_i)$ is the set of neighbors for node v_i, d_i indicates the degree of node v_i, and the i-th row represents the label distribution of neighbors for node v_i.

Definition 2. **Class-wise Connectivity Pattern Matrix.** Let $\boldsymbol{\mathcal{C}} \in \mathbb{R}^{|\boldsymbol{\mathcal{Y}}| \times |\boldsymbol{\mathcal{Y}}|}$ represent the class-wise connectivity pattern. Then, it is defined by Eqn. (5).

$$\boldsymbol{\mathcal{C}}_{i,j} = \frac{1}{\left|\left\{v \in \boldsymbol{\mathcal{V}}^L \mid y_v = \boldsymbol{\mathcal{Y}}_i\right\}\right|} \sum_{u \in \left\{v \in \boldsymbol{\mathcal{V}}^L \mid y_v = \boldsymbol{\mathcal{Y}}_i\right\}} \boldsymbol{\mathcal{D}}_{u,j}. \tag{5}$$

The proposed TA-GST scores the connectivity pattern of an unlabeled node by measuring the disparity between the label distribution of its node-wise connectivity pattern and the corresponding class-wise one. This assumes that the label distribution of neighbors for all nodes is accessible. However, it is unreachable in the semi-supervised learning paradigm. In addition, the classification probability cannot unerringly reflect the label distribution of neighbors. Hence, it employs the ***soft label*** to approximate the connectivity pattern of the unlabeled nodes. This is an idea of knowledge distillation. The probability transformation involved in soft label is highly associated with the label distribution of neighbors. Therefore, the soft label can approximate the label distribution of neighbors (Joshi et al. 2022).

In detail, the proposed method customizes the distillation temperature $\mathcal{T}$ for various unlabeled nodes, aligning the softened logit vector with the label distribution of the neighbors. The node with a high degree exhibits a miscellaneous label distribution of neighbors. Consequently, a higher distillation temperature makes the softened logit vector deliberate the contributions of each neighboring node more evenly, reflecting the comprehensive label distribution of neighbors. Correspondingly, the connectivity pattern of the node with a low degree is relatively simple. Thus, a lower distillation temperature mitigates the softening effect, placing greater emphasis on the node's own label information, thereby reducing the possibility of overfitting the connectivity pattern.

Specifically, the proposed method first evaluates the customized distillation temperatures for various nodes, as expressed in Eqn. (6) and Eqn. (7).

$$c_i = \frac{\sum_{u \in \left\{v \in \boldsymbol{\mathcal{V}}^L | y_v = i\right\}} |\boldsymbol{\mathcal{N}}(u)|}{\left|\left\{v \in \boldsymbol{\mathcal{V}}^L \mid y_v = \boldsymbol{\mathcal{Y}}_i\right\}\right|}, \tag{6}$$

$$\mathcal{T}_v = e^{\frac{|N(v)|}{c_i}}, \tag{7}$$

where c_i is the average degree of class $\boldsymbol{\mathcal{Y}}_i$ and $\mathcal{T}_v$ represents the customized distillation temperature of node v.

Then, it softens the classification probability into the label distribution of neighbors according to Eqn. (8).

$$\hat{p}_i^s = \frac{exp\left(p_i/\mathcal{T}_v\right)}{\sum_{j=1}^{n} exp\left(p_j/\mathcal{T}_v\right)}, \tag{8}$$

where $\hat{p}_i^s$ represents the softened classification probability of p_i.

Consequently, it uses $\boldsymbol{\mathcal{Z}}_v^s = [\hat{p}_1^s, \hat{p}_2^s, ..., \hat{p}_n^s]$ to approximate $\boldsymbol{\mathcal{D}}_v$ and measures $\boldsymbol{\mathcal{C}}_{\boldsymbol{\mathcal{Y}}_v}$ by averaging $\boldsymbol{\mathcal{D}}_v$ of the labeled nodes belonging to the same class $\boldsymbol{\mathcal{Y}}_v$. Subsequently, the score of connectivity pattern is estimated by calculating the cosine similarity between $\boldsymbol{\mathcal{D}}_v$ and $\boldsymbol{\mathcal{C}}_{\boldsymbol{\mathcal{Y}}_v}$, as specified in Eqn. (9).

$$\mathcal{S}_v^{pat} = cos\left(\boldsymbol{\mathcal{D}}_v, \boldsymbol{\mathcal{C}}_{\boldsymbol{\mathcal{Y}}_v}\right), \tag{9}$$

where $\mathcal{S}_v^{pat}$ represents the score of the connectivity pattern of node v.

Finally, the confidence score of a certain node is evaluated by integrating the classification score and the connectivity pattern score, as defined in Eqn. (10).

$$\mathcal{S}_v^{con} = \alpha \mathcal{S}_v^{cls} + \beta \mathcal{S}_v^{pat}, \tag{10}$$

where α controls the relative importance of the classification score and β controls the relative importance of the connectivity pattern score. The proposed method sorts the confidence scores and incorporates the top-k scored samples into the training set.

In total, the model first identifies the logit vector of each node and takes the maximum value of the logical components as the classification score. Then, it evaluates $\boldsymbol{\mathcal{D}}$ and $\boldsymbol{\mathcal{C}}$ by softening the logit vector. Subsequently, it measures the cosine similarity between the node-wise connectivity pattern and the corresponding class-wise one to identify the score of the connectivity pattern of an unlabeled node. Finally, it labels the top-k scored unlabeled samples and adds them to the training set.

Objective of Optimization

The proposed TA-GST is implemented in an end-to-end manner. It does not explicitly employ the teacher-student pattern, but segments the complete model training into two phases. In the first phase, it achieves an undertrained GNN-based model on the labeled samples. Thus, the objective of optimization can be defined by Eqn. (11).

$$\mathcal{L} = -\sum_{v \in \boldsymbol{\mathcal{V}}^L} y_v \log\left(\hat{y}_v\right), \tag{11}$$

where y_v indicates the ground-truth label of node v and $\hat{y}_v$ represents its prediction label.

In the second phase, it leverages the topology-aware scoring method to pseudo-label reliable unlabeled samples for itself. Therefore, the objective of optimization incorporates both accessible labeled samples and reliable pseudo-labeled samples, as formulated in Eqn. (12).

$$\mathcal{L} = -\sum_{v \in \boldsymbol{\mathcal{V}}^L} y_v \log\left(\hat{y}_v\right) - \gamma \sum_{u \in \boldsymbol{\mathcal{V}}^P} y_u \log\left(\hat{y}_u\right), \tag{12}$$

where $\boldsymbol{\mathcal{V}}^P$ represents the set of pseudo-labeled samples and γ controls the relative importance of them.

In particular, it sets two hyper-parameters t and k to control the epoch of triggering pseudo-labeling and the number of pseudo-labels added per epoch, respectively. Generally, TA-GST is a common framework and can be easily generalized to arbitrary GNN-based models.

Algorithm and Complexity Analyses

The algorithm of TA-GST is summarized in Algorithm 1. The proposed method introduced $\boldsymbol{\mathcal{D}}$ and $\boldsymbol{\mathcal{C}}$ to evaluate the score of the connectivity pattern. In the semi-supervised learning paradigm, it is unreachable to ascertain the label distribution of neighbors of each node. Therefore, it softens the logit vector to approximate the guideline, and the time complexity of evaluating $\boldsymbol{\mathcal{D}}$ is $\mathcal{O}(|\boldsymbol{\mathcal{V}}|)$. Then, it sums $\boldsymbol{\mathcal{D}}_v$ of the labeled nodes that belong to the same class and takes the average to estimate $\boldsymbol{\mathcal{C}}$, where the corresponding time complexity is $\mathcal{O}(|\boldsymbol{\mathcal{V}}^L|)$. Based on $\boldsymbol{\mathcal{D}}$ and $\boldsymbol{\mathcal{C}}$, it measures the score of the connectivity pattern for each unlabeled node. Consequently, the time complexity is $\mathcal{O}(|\boldsymbol{\mathcal{V}}^U|)$. In conclusion, the time complexity of our proposed TA-GST is $\mathcal{O}(|\boldsymbol{\mathcal{V}}|)$.

Algorithm 1: The proposed TA-GST method.

Input: Graph $\mathcal{G} = (\boldsymbol{\mathcal{V}}, \boldsymbol{\mathcal{E}}, \boldsymbol{X})$, the adjacency matrix $\boldsymbol{A}$, the GNN-based model f_θ, the loss value $\mathcal{L}$, the epoch of triggering pseudo-labeling t, the number of training epochs N.

Output: The matrix of node representations.

/ Phase 1: Train the GNN-based model only on the available labeled samples */*

for $i = 1, 2, ..., t-1$ **do**
  Get the output embedding according to Eqn. (1);
  Calculate $\mathcal{L}$ according to Eqn. (11);
  Backward propagate $\mathcal{L}$ and optimize the model parameters θ;
end for

/ Phase 2: Train the GNN-based model on the available labeled samples and reliable pseudo-labeled samples */*

for $i = t, t+1, ..., N$ **do**
  Get the output embedding according to Eqn. (1);
  Calculate the confidence score of unlabeled sample and pseudo-label reliable unlabeled samples;
  Calculate $\mathcal{L}$ according to Eqn. (12);
  Backward propagate $\mathcal{L}$ and optimize the parameters of model θ;
end for

Experiments

In this section, we conduct experiments to demonstrate the effectiveness of our proposed TA-GST. We first briefly describe the datasets and baselines. Then, we conduct a comprehensive and in-depth analysis of the experimental results. This section aims to address the following four essential research questions.

RQ1: To what extent does TA-GST improve the performance on the task of semi-supervised node classification?

RQ2: Can the proposed TA-GST perform reliably and stably?

RQ3: How do the hyper-parameters take effect on the performance of the proposed TA-GST?

RQ4: How do the visualization results provide evidence for the superiority of our proposed TA-GST?

Datasets

We employ widely-used citation datasets (Cora, Citeseer, Pubmed, and ogbn-arXiv) (Yang, Cohen, and Salakhudinov 2016; Hu et al. 2020), social network dataset (BlogCatalog) (Tang and Liu 2009), co-purchase dataset (Amazon-Computers), and co-author dataset (Coauthor-CS) (Shchur et al. 2018) to evaluate the effectiveness of our method. These datasets are split following the prior work (Liu et al. 2022). They are summarized in Table 1.

Baselines

We compare our method with competitive baselines, including STs (Li, Han, and Wu 2018), M3S (Sun, Lin, and Zhu 2020), SST (Zhou et al. 2023), DR-GST (Liu et al. 2022), CPL (Wang et al. 2023), HCPL (Luo et al. 2023b), RNCGLN (Zhu et al. 2024), and KD-FSNC (Wu et al. 2024).

Datasets	Nodes	Features	Edges	Classes	Train	Validation	Test
Cora	2,708	1,433	5,429	7	140	500	1,000
Citeseer	3,327	3,703	4,732	6	120	500	1,000
Pubmed	19,717	500	44,338	3	60	500	1,000
BlogCatalog	5,196	8,189	171,743	6	519	1,039	3,638
Amazon-Computers	13,752	767	245,861	10	1,375	2,750	9,627
Coauthor-CS	18,333	6,805	81,894	15	1,833	3,666	12,834
ogbn-arXiv	169,343	128	1,166,243	40	8,467	29,799	48,603

Table 1: The brief description of datasets used in the paper.

Experimental Results and Analyses (RQ1)

The experimental results of node classification are reported in Table 2. The key observations are as follows.

(1) The proposed TA-GST is compatible with various GNN-based models and outperforms the vanilla training paradigm in most cases. Specifically, on Cora dataset, TA-GST improves the GNN-based models (GCN, GAT, GraphSAGE, and ChebyNet) by 5.25%, 5.17%, 5.50%, and 5.05% in terms of Micro-F1, respectively. The proposed TA-GST also performs well on other datasets. In particular, similar improvements are also observed in the Macro-F1 metrics. These experimental results demonstrate that TA-GST effectively improves the classification performance of GNN models on diverse graph types and scales.

(2) The proposed TA-GST exhibits an overwhelming superiority in improving the performance of GNN-based models compared to existing graph self-training methods. Among these competitive methods, STs and M3S perform well on Citeseer dataset with an improvement of 3.70% and 3.76% compared to GCN. SST yields a significant performance improvement on ogbn-arXiv dataset. DR-GST and CPL exhibit obvious superiority on Cora dataset. RNCGLN and KD-FSNC demonstrate superiority on Amazon-Computers and Coauthor-CS datasets. HCPL achieves the best performance on Pubmed dataset. However, the proposed TA-GST outperforms all other competitive methods by taking the label distribution of neighbors into graph self-training to ensure the reliability of the pseudo-label.

(3) The proposed TA-GST yields varied performance improvement across various datasets. Among the citation datasets, the performance improvement is more significant on Cora and Citeseer datasets than on Pubmed dataset. We attribute this phenomenon to the gap in the number of categories. Cora, Citeseer, and Pubmed datasets contain 7, 6, and 3 subcategories, respectively. The dataset with a large number of categories benefits the model by learning a smoother label distribution. Moreover, the proposed TA-GST performs better on a sparse dataset in topology (e.g., Cora) than on a dense one (e.g., BlogCatalog). We argue that the complex connectivity confuses the accurate propagation of pseudo-labels.

Dataset	Cora		Citeseer		Pubmed		BlogCatalog		Amazon-Computers		Coauthor-CS		ogbn-arXiv	
Methods	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1
GCN	79.62	80.02	66.87	70.66	76.21	77.34	70.86	71.28	87.30	88.81	89.92	91.57	42.72	58.86
GAT	79.45	80.35	68.03	71.55	75.79	76.83	68.77	69.56	87.26	88.15	90.33	92.23	42.71	58.77
GraphSAGE	78.33	79.03	66.59	70.61	74.83	76.03	68.73	69.66	86.72	87.62	90.08	91.85	41.68	57.54
ChebyNet	77.88	78.91	65.92	69.36	75.90	77.09	67.92	68.86	86.75	87.89	88.80	90.44	41.59	57.29
STs	81.77	82.89	71.32	74.36	80.05	81.02	72.86	73.95	88.32	89.75	92.15	93.45	42.89	58.73
M3S	81.65	82.78	71.25	74.42	80.56	81.34	72.58	73.27	88.81	90.87	91.84	93.01	43.66	59.64
SST	81.25	81.79	69.13	72.81	78.23	79.09	72.07	72.86	89.04	90.75	91.39	92.86	43.87	59.72
DR-GST	83.29	84.03	72.49	75.78	80.27	81.08	73.77	74.63	90.33	91.52	92.17	93.56	45.02	60.71
CPL	83.01	83.94	69.72	72.96	79.22	79.98	72.99	73.26	90.28	91.77	91.93	93.19	43.95	60.07
HCPL	83.42	84.20	71.25	74.40	**81.56**	**82.40**	73.17	73.55	91.35	92.61	92.04	93.25	44.67	59.99
RNCGLN	84.07	84.46	73.57	74.02	80.55	80.98	73.39	73.73	92.50	93.37	93.85	94.28	46.11	61.24
KD-FSNC	84.23	84.77	73.66	74.19	80.27	81.33	73.51	73.93	92.30	93.25	93.77	94.36	45.89	60.38
TA-GST$_{GCN}$	84.78	85.27	**75.23**	**77.88**	81.25	82.07	**74.53**	**75.12**	92.74	93.63	94.22	95.07	47.87	**64.27**
TA-GST$_{GAT}$	**84.87**	**85.52**	74.93	77.04	80.31	81.56	74.39	75.03	**93.12**	**94.05**	**94.37**	**95.22**	**47.93**	64.12
TA-GST$_{SAGE}$	83.67	84.53	72.37	75.13	80.53	81.44	73.40	74.13	92.51	93.15	92.92	94.15	47.26	63.88
TA-GST$_{Cheby}$	83.08	83.96	71.87	74.66	79.41	80.60	73.31	73.76	92.37	92.71	92.70	93.91	47.09	63.29

Table 2: The experimental results of semi-supervised node classification, where the best results are highlighted in **bold** and the runner-ups are underlined.

The Stability and Reliability of TA-GST (RQ2)

We employ GCN as the backbone and verify the stability and reliability of our proposed TA-GST on Cora, BlogCatalog, Amonzon-Computers, and Coauthor-CS datasets.

Stability. The experimental results of the stability study are shown in Figure 3. It is obvious that the vanilla self-training performs unsteadily and even deteriorates the performance of the GNN-based model. The analogous cases appear simultaneously on various types of datasets, demonstrating that vanilla self-training is inapplicable to the GNN-based model. In contrast, the proposed TA-GST achieves remarkable and stable improvements. Moreover, it consistently yields better performance than the training mode of '*w/o* ST'. In addition, the improvement of TA-GST is more significant and steady than the training mode of 'Vanilla ST'. In conclusion, the experimental results demonstrate that our proposed TA-GST is a stable graph self-training method for GNN-based models.

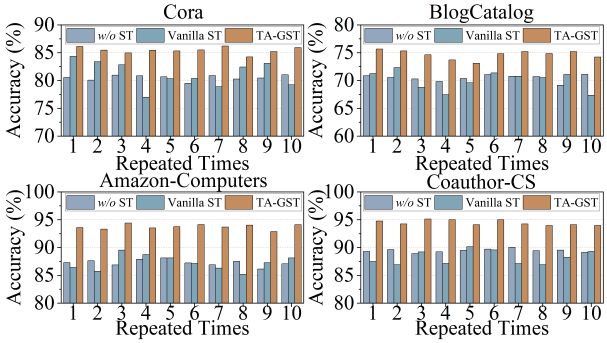

Figure 3: The experimental results of stability study.

Reliability. The experimental results of the reliability study are shown in Figure 4. It can be observed that the accuracy of the pseudo-labels for TA-GST and vanilla self-training is almost equal at the beginning. That's because the GNN-based model is under-fitted, and the quantity of pseudo-labels is scarce. However, with the advancement of training, the superiority of TA-GST becomes evident. It incorporates the classification probability and connectivity pattern in parallel to identify the reliability of unlabeled samples. Hence, the pseudo-labels are reliable and remain confident. In summary, the experimental results demonstrate that TA-GST leverages reliable pseudo-labels.

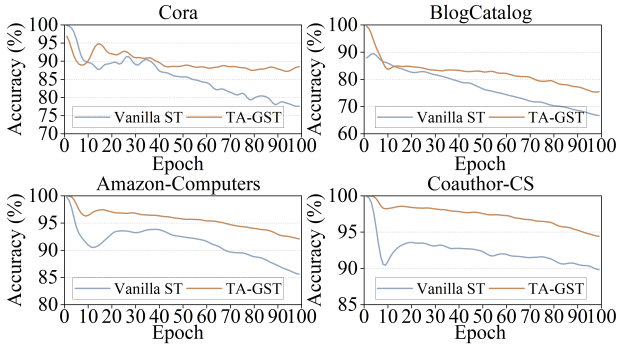

Figure 4: The experimental results of reliability study.

Parameter Sensitivity Analysis (RQ3)

In this section, we explore the performance of the proposed TA-GST under various hyper-parameter settings. We employ GCN as the backbone and conduct the parameter analysis experiment on Cora, BlogCatalog, Amazon-Computers,

and Coauthor-CS datasets.

Impact of the Hyper-parameters α and β. α controls the relative importance of the classification score, and β controls the relative importance of the connectivity pattern score. We vary α and β from $\{0.5, 0.6, ...1.5\}$. The experimental results are shown in Figure 5. It can be seen that when $\alpha = 1.4$, $\beta = 1$ and $\alpha = 1$, $\beta = 1.2$, the model achieves optimal performance on Cora dataset. On BlogCatalog dataset, the model realizes ideal performance when $\alpha = 1.4$, $\beta = 1$ and $\alpha = 1$, $\beta = 1.1$. On Amazon-Computers dataset, the model performs well when $\alpha = 1.1$, $\beta = 1$ and $\alpha = 1$, $\beta = 0.8$. Similarly, the model achieves peak performance when $\alpha = 1.1$, $\beta = 1$ and $\alpha = 1$, $\beta = 0.9$ on Coauthor-CS dataset. In conclusion, both the classification score and connectivity pattern score are essential for identifying the reliability of the pseudo-labels. The classification score is slightly larger than the connectivity pattern score, which benefits the performance of the proposed method.

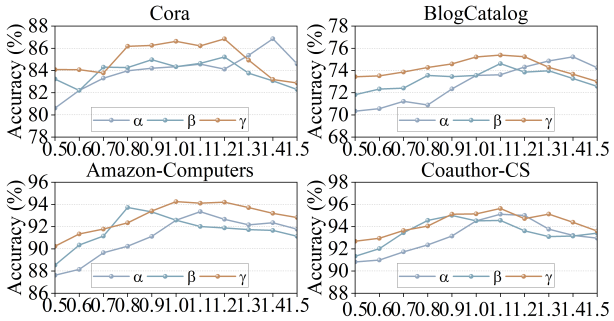

Figure 5: The impact of hyper-parameters α, β, and γ.

Impact of Hyper-parameter γ. γ controls the relative importance of pseudo-labeled samples in optimization. We search γ from $\{0.5, 0.6..., 1.5\}$. The experimental results are shown in Figure 5. It can be observed that the model performs well when $\gamma = 1.2$, $\gamma = 1.1$, $\gamma = 1.0$, and $\gamma = 1.1$ on Cora, BlogCatalog, Amazon-Computers, and Coauthor-CS datasets, respectively. The proposed TA-GST integrates both the classification score and connectivity pattern score, thereby improving the reliability of the pseudo-labels. As a result, the pseudo-labels can be considered as the real labels and assigned the same training weights as the real ones.

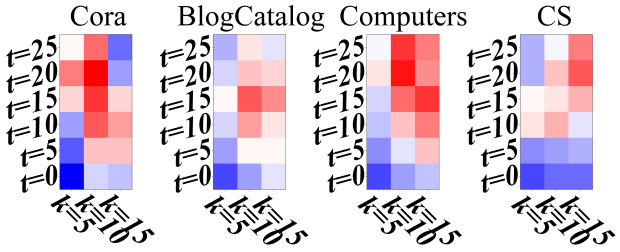

Figure 6: The investigation of hyper-parameters t and k.

Impact of Hyper-parameter t and k. t specifies the epoch of triggering pseudo-labeling, and k identifies the number of pseudo-labels added per epoch. We select t and k from $\{0, 5, 10, 15, 20, 25\}$ and $\{5, 10, 15\}$, respectively. The experimental results are shown in Figure 6. It is obvious that the performance of TA-GST is sensitive to the hyper-parameters t and k. Specifically, it achieves the best performance on Cora dataset when $t = 20$, $k = 10$. On BlogCatalog dataset, the optimal results are achieved when $t = 15$, $k = 10$. Similarly, on Amazon-Computers and Coauthor-CS datasets, the proposed TA-GST excels when $t = 20$, $k = 10$ and $t = 20$, $k = 15$, respectively. Furthermore, an indiscriminate increase in the number of pseudo-labels does not guarantee continuous improvement. We attribute this phenomenon to the fact that the excessive pseudo-labels lead to overfitting and potentially degrade model performance. Additionally, the timing of triggering pseudo-labeling is critical. The proposed TA-GST achieves superior performance when the GNN-based models are first trained on the available labeled samples for an appropriate number of epochs.

Visualization (RQ4)

In this section, we visualize the learned node embedding on Cora dataset. Specifically, we employ the t-distributed stochastic method (Van der Maaten and Hinton 2008) to map the node embedding into a 2-dimensional space. The visualization results are shown in Figure 7, where each dot represents a node and the diverse colors discriminate various categories. It is clear that the visualization results of TA-GST are more compact and take on explicit category boundaries. The experimental results verify the capability of TA-GST in learning distinguishable node representations.

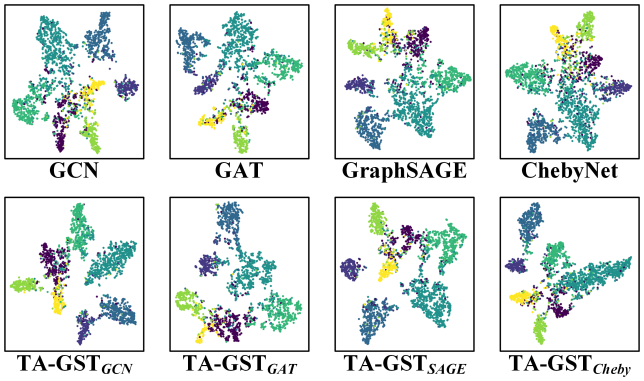

Figure 7: The t-SNE visualization on Cora dataset.

Conclusion

In this paper, we first conduct an empirical study for self-training on graphs. The empirical findings reveal that it may be confused by the label distribution of neighbors and even lead to a false positive issue, resulting in unpromising performance. To address this challenging issue, we propose a simple yet effective graph self-training method, which incorporates classification probabilities and label distribution of neighbors to measure the reliability of the pseudo-labels. The experimental results demonstrate the superiority of our proposed TA-GST.

Acknowledgments

This research is supported by the National Key R&D Program of China (Grant No. 2022ZD0119501), the National Natural Science Foundation of China (Grant No. 62472263, 52374221, 52574256, 62502288), the Taishan Scholar Program of Shandong Province (tstp20250506, tsqn202211154), Shandong Youth Innovation Team, the Natural Science Foundation of Shandong Province (Grant No. ZR2024MF034, ZR2025QC624), the Ministry of Education in China Foundation for Humanities and Social Sciences (Grant No. 24YJAZH058, 24YJCZH461), the Guangxi Key Laboratory of Trusted Software (KX202305).

References

Bruna, J.; Zaremba, W.; Szlam, A.; and LeCun, Y. 2014. Spectral Networks and Deep Locally Connected Networks on Graphs. In *International Conference on Learning Representations*, 1–14.

Defferrard, M.; Bresson, X.; and Vandergheynst, P. 2016. Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering. *Advances in Neural Information Processing Systems*, 29: 1–9.

Duvenaud, D. K.; Maclaurin, D.; Iparraguirre, J.; Bombarell, R.; Hirzel, T.; Aspuru-Guzik, A.; and Adams, R. P. 2015. Convolutional Networks on Graphs for Learning Molecular Fingerprints. *Advances in Neural Information Processing Systems*, 28: 1–9.

Hamilton, W.; Ying, Z.; and Leskovec, J. 2017. Inductive Representation Learning on Large Graphs. *Advances in Neural Information Processing Systems*, 30: 1–11.

Hu, W.; Fey, M.; Zitnik, M.; Dong, Y.; Ren, H.; Liu, B.; Catasta, M.; and Leskovec, J. 2020. Open Graph Benchmark: Datasets for Machine Learning on Graphs. *Advances in Neural Information Processing Systems*, 33: 22118–22133.

Joshi, C. K.; Liu, F.; Xun, X.; Lin, J.; and Foo, C. S. 2022. On Representation Knowledge Distillation for Graph Neural Networks. *IEEE Transactions on Neural Networks and Learning Systems*, 35(4): 4656–4667.

Juan, X.; Peng, M.; and Wang, X. 2021. Exploring Self-Training for Imbalanced Node Classification. In *International Conference on Neural Information Processing*, 28–36.

Juan, X.; Peng, M.; and Wang, X. 2025. Dynamic Self-Training with Less Uncertainty for Graph Imbalance Learning. *Expert Systems with Applications*, 271: 126643.

Kipf, T. N.; and Welling, M. 2016. Semi-Supervised Classification with Graph Convolutional Networks. In *International Conference on Learning Representations*, 1–14.

Li, Q.; Han, Z.; and Wu, X. 2018. Deeper Insights into Graph Convolutional Networks for Semi-Supervised Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 1–8.

Li, W.; Wang, C.; Xiong, H.; and Lai, J. 2023. HomoGCL: Rethinking Homophily in Graph Contrastive Learning. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 1341–1352.

Li, Z.; Xia, L.; Hua, H.; Zhang, S.; Wang, S.; and Huang, C. 2025. DiffGraph: Heterogeneous Graph Diffusion Model. In *Proceedings of the 18th ACM International Conference on Web Search and Data Mining*, 40–49.

Liu, H.; Hu, B.; Wang, X.; Shi, C.; Zhang, Z.; and Zhou, J. 2022. Confidence May Cheat: Self-Training on Graph Neural Networks Under Distribution Shift. In *Proceedings of the ACM Web Conference 2022*, 1248–1258.

Liu, J.; Yang, C.; Lu, Z.; Chen, J.; Li, Y.; Zhang, M.; Bai, T.; Fang, Y.; Sun, L.; Yu, P. S.; et al. 2023. Towards Graph Foundation Models: A Survey and Beyond. *arXiv preprint arXiv:2310.11829*, 1–23.

Luo, G.; Zhang, H.; Yuan, Q.; Li, J.; Wang, W.; and Wang, F.-Y. 2023a. One Size Fits All: A Unified Traffic Predictor for Capturing the Essential Spatial–Temporal Dependency. *IEEE Transactions on Neural Networks and Learning Systems*, 35(8): 11317–11331.

Luo, T.; Liu, Y.; and Pan, S. J. 2024. Collaborative Sequential Recommendations via Multi-View GNN-Transformers. *ACM Transactions on Information Systems*, 42(6): 1–27.

Luo, X.; Ju, W.; Gu, Y.; Qin, Y.; Yi, S.; Wu, D.; Liu, L.; and Zhang, M. 2023b. Toward Effective Semi-Supervised Node Classification with Hybrid Curriculum Pseudo-labeling. *ACM Transactions on Multimedia Computing, Communications and Applications*, 20(3): 1–19.

Mukherjee, S.; and Awadallah, A. 2020. Uncertainty-Aware Self-Training for Few-Shot Text Classification. *Advances in Neural Information Processing Systems*, 33: 21199–21212.

Peng, H.; Zhang, J.; Huang, X.; Hao, Z.; Li, A.; Yu, Z.; and Yu, P. S. 2024. Unsupervised Social Bot Detection via Structural Information Theory. *ACM Transactions on Information Systems*, 42(6): 1–42.

Shchur, O.; Mumme, M.; Bojchevski, A.; and Günnemann, S. 2018. Pitfalls of Graph Neural Network Evaluation. *arXiv preprint arXiv:1811.05868*, 1–11.

Singh, K.; Tsai, Y.; Li, C.; Cha, M.; and Lin, S. 2023. GraphFC: Customs Fraud Detection with Label Scarcity. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, 4829–4835.

Sun, K.; Lin, Z.; and Zhu, Z. 2020. Multi-Stage Self-Supervised Learning for Graph Convolutional Networks on Graphs with Few Labeled Nodes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 5892–5899.

Tang, L.; and Liu, H. 2009. Relational Learning via Latent Social Dimensions. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 817–826.

Van der Maaten, L.; and Hinton, G. 2008. Visualizing Data Using t-SNE. *Journal of Machine Learning Research*, 9(11): 1–27.

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2018. Graph Attention Networks. In *International Conference on Learning Representations*, 1–12.

Wang, B.; Li, J.; Liu, Y.; Cheng, J.; Rong, Y.; Wang, W.; and Tsung, F. 2023. Deep Insights into Noisy Pseudo Labeling on Graph Data. *Advances in Neural Information Processing Systems*, 36: 76214–76228.

Wang, F.; Zhao, T.; and Wang, S. 2024. Distribution Consistency based Self-Training for Graph Neural Networks With Sparse Labels. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, 712–720.

Wang, X.; Liu, H.; Shi, C.; and Yang, C. 2021. Be Confident! Towards Trustworthy Graph Neural Networks via Confidence Calibration. *Advances in Neural Information Processing Systems*, 34: 23768–23779.

Wu, Z.; Mo, Y.; Zhou, P.; Yuan, S.; and Zhu, X. 2024. Self-Training Based Few-Shot Node Classification by Knowledge Distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 15988–15995.

Xu, C.; Xu, J.; Dong, Z.; and Wen, J.-R. 2023. Syntactic-Informed Graph Networks for Sentence Matching. *ACM Transactions on Information Systems*, 42(2): 1–29.

Yang, L.; Zhuo, W.; Qi, L.; Shi, Y.; and Gao, Y. 2022. ST++: Make Self-Training Work Better for Semi-Supervised Semantic Segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4268–4277.

Yang, Y.; Zhang, C.; Song, X.; Dong, Z.; Zhu, H.; and Li, W. 2023. Contextualized Knowledge Graph Embedding for Explainable Talent Training Course Recommendation. *ACM Transactions on Information Systems*, 42(2): 1–27.

Yang, Z.; Cohen, W.; and Salakhudinov, R. 2016. Revisiting Semi-Supervised Learning with Graph Embeddings. In *International Conference on Machine Learning*, 40–48.

Zhou, Z.; Shi, J.; Zhang, S.; Huang, Z.; and Li, Q. 2023. Effective Stabilized Self-Training on Few-labeled Graph Data. *Information Sciences*, 631: 369–384.

Zhu, X.; and Goldberg, A. B. 2022. *Introduction to Semi-Supervised Learning*.

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

Zoph, B.; Ghiasi, G.; Lin, T.-Y.; Cui, Y.; Liu, H.; Cubuk, E. D.; and Le, Q. 2020. Rethinking Pre-Training and Self-Training. *Advances in Neural Information Processing Systems*, 33: 3833–3845.