# A Survey of Data-Efficient Graph Learning

**Wei Ju**[1] , **Siyu Yi**[2*] , **Yifan Wang**[3] , **Qingqing Long**[1] , **Junyu Luo**[1] , **Zhiping Xiao**[4] , **Ming Zhang**[1*]

[1]School of Computer Science, Peking University ,
[2]School of Statistics and Data Science, Nankai University ,
[3]School of Information Technology & Management, University of International Business and Economics ,
[4]Department of Computer Science, University of California Los Angeles

{juwei, qingqinglong, mzhang_cs}@pku.edu.cn, siyuyi@mail.nankai.edu.cn, yifanwang@uibe.edu.cn,
luojunyu@stu.pku.edu.cn, patricia.xiao@cs.ucla.edu

## Abstract

Graph-structured data, prevalent in domains ranging from social networks to biochemical analysis, serve as the foundation for diverse real-world systems. While graph neural networks demonstrate proficiency in modeling this type of data, their success is often reliant on significant amounts of labeled data, posing a challenge in practical scenarios with limited annotation resources. To tackle this problem, tremendous efforts have been devoted to enhancing graph machine learning performance under low-resource settings by exploring various approaches to minimal supervision. In this paper, we introduce a novel concept of Data-Efficient Graph Learning (DEGL) as a research frontier, and present the first survey that summarizes the current progress of DEGL. We initiate by highlighting the challenges inherent in training models with large labeled data, paving the way for our exploration into DEGL. Next, we systematically review recent advances on this topic from several key aspects, including self-supervised graph learning, semi-supervised graph learning, and few-shot graph learning. Also, we state promising directions for future research, contributing to the evolution of graph machine learning.

## 1 Introduction

Graph learning has emerged as a pivotal field at the intersection of machine learning and graph theory, offering a versatile framework for modeling and analyzing complex relationships in various domains [Jin *et al.*, 2020; Ju *et al.*, 2024a]. With the increasing prevalence of graph-structured data, ranging from social networks to biological interactions [Sun *et al.*, 2020], the demand for effective methods to extract meaningful insights from graphs has grown substantially.

With the rising popularity of graph neural networks, particularly notable for their effective message-passing mechanisms [Kipf and Welling, 2017], a myriad of graph-related challenges have witnessed outstanding performance. However, despite the promising advancements in graph learning, the current landscape predominantly relies on extensive labeled data,

---

*Corresponding authors.

introducing challenges associated with high annotation costs, time-intensive processes, and resource demands [Hao *et al.*, 2020; Luo *et al.*, 2023b]. This limitation becomes particularly evident in practical applications where obtaining substantial labeled data is impractical. To illustrate this imperative, in molecular domain, annotating intricate molecular structures or interactions could involve sophisticated experiments and the engagement of specialized scientists, leading to elevated annotation costs [Ju *et al.*, 2023b]; in the genomics research, acquiring precisely annotated gene functions or interactions might demand the expertise and time of professional biologists for understanding complex relationships [Yu *et al.*, 2023]. These instances highlight the urgent need for graph learning methodologies tailored to low-resource settings, emphasizing the critical necessity of developing approaches that can operate effectively with limited labeled data.

For effective graph learning, researchers have conducted extensive and specialized studies focusing on the exploration of graph machine learning under low-resource settings, aiming at mitigating the costs and time associated with annotation. Nevertheless, although there have been increasingly applied in effective graph learning, this rapidly expanding field still lacks a systematic review. To fill this gap, in this paper, we develop a novel concept of Data-Efficient Graph Learning (DEGL) to summarize the existing works of DEGL and provide a promising research frontier to aid researchers in reviewing, summarizing, and strategizing for the future.

## 2 Taxonomy

To enhance our understanding of the dynamic evolution in DEGL, we identify pivotal research endeavors, analyze their motivations, and succinctly encapsulate their primary technical contributions. As illustrated in Figure 1, this survey establishes a new taxonomy, which divides these works into three different categories, i.e., self-supervised graph learning, semi-supervised graph learning, and few-shot graph learning. These groups can be further summarized in different scenarios. Then, we briefly introduce these three categories as follows:

- *Self-supervised Graph Learning* is a paradigm that leverages the inherent structure and relationships within graph data to train models without relying on external labeled information. At its core, the key idea is to design tasks that encourage the model to learn meaningful representa-
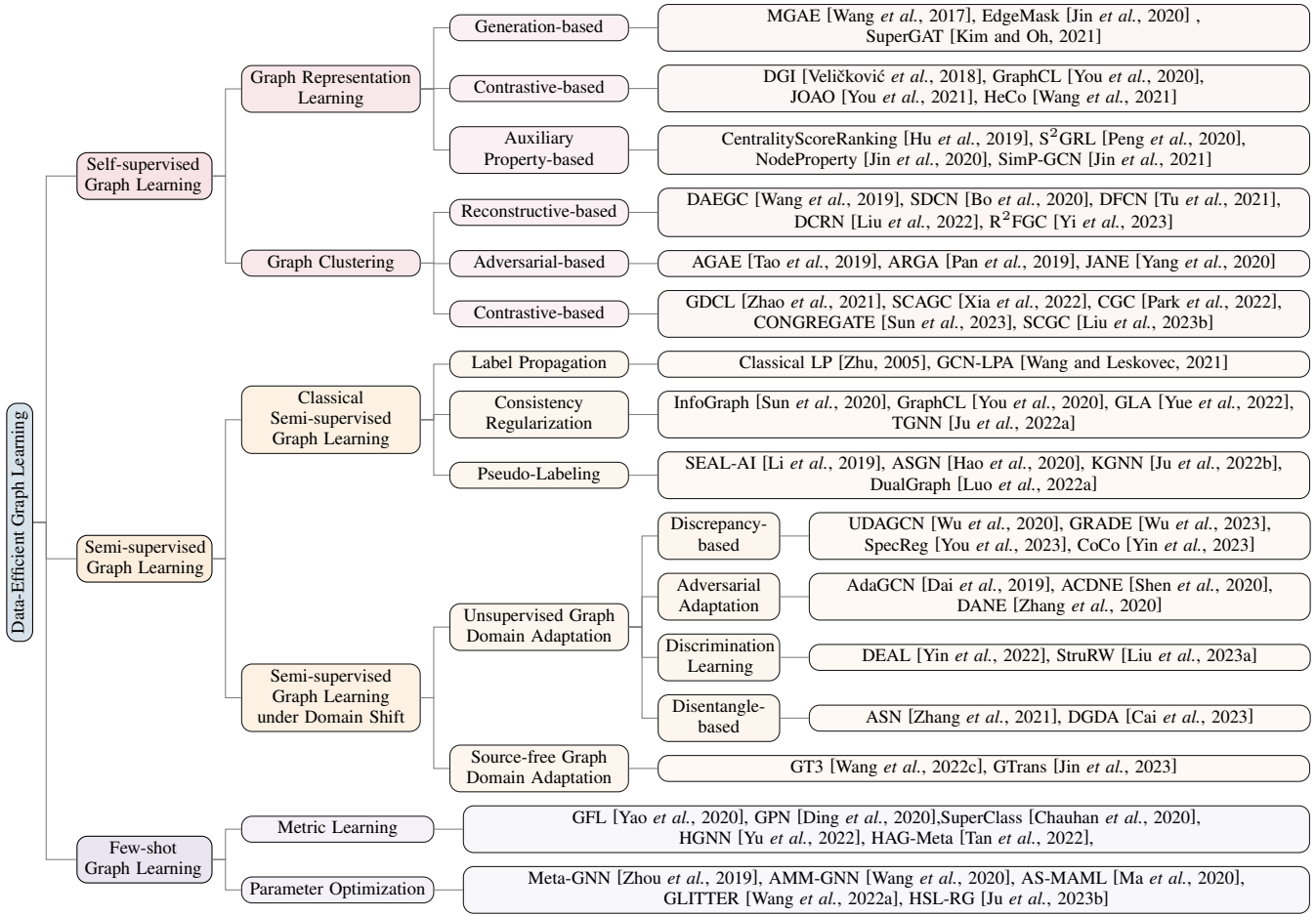
Figure 1: A taxonomy of data-efficient graph learning (DEGL).

tions from the graph data itself. Based on the presence of a specific downstream task, self-supervised graph learning can be further categorized into non-end-to-end graph representation learning [Jin *et al.*, 2020] and end-to-end graph clustering [Bo *et al.*, 2020; Liu *et al.*, 2022].

- **Semi-supervised Graph Learning** involves using both labeled and unlabeled samples to train models, capitalizing on the available information while handling scenarios with limited labeled samples. The fundamental idea is to leverage the relationships within the labeled and unlabeled instances to guide the model in learning representations for unlabeled nodes or graphs. Depending on whether there is a distribution shift during inference, we further categorize semi-supervised graph learning into classical semi-supervised graph learning [Wang and Leskovec, 2021; Ju *et al.*, 2022a] and semi-supervised graph learning under domain shift [Yin *et al.*, 2023; Jin *et al.*, 2023].

- **Few-shot Graph Learning** is a specialized area designed to enable models to effectively generalize and make accurate predictions when presented with only a limited number of labeled examples. The core idea is to equip the model with the ability to learn from a few annotated instances and then

apply this acquired knowledge to make predictions on new, unseen data. Thus researchers either adopt metric learning to encourage each query node to approach its prototypes [Tan *et al.*, 2022] or parameter optimization to generate node representations using meta-learning [Ju *et al.*, 2023b].

The three research directions are interconnected rather than mutually exclusive, contributing collectively to the rapid advancement of the data-efficient graph learning field. Table 1 further analyzes the representative DEGL works according to the different properties. In the following sections, we delve deeper into these three research directions, exploring key challenges, representative solutions, and emerging trends.

## 3 Self-supervised Graph Learning

### 3.1 Graph Representation Learning

Graph Representation Learning (GRL) [Jin *et al.*, 2020] allows the model to capture nuanced patterns and dependencies among nodes without relying on manual labels, making it particularly valuable in scenarios where labels are expensive or hard to acquire. In general, the framework of GRL can be summarized as:

$$\min_{\theta, \psi} \mathcal{L}\left(\mathcal{D}, f_\theta, g_\psi\right), \qquad (1)$$

| Branch | | Methods | Data Type | Objective Function |
|---|---|---|---|---|
| Self | GRL | MGAE | Node | Reconstruction Loss |
| | | GraphCL | Graph | InfoNCE Loss |
| | | S$^2$GRL | Graph | Reconstruction Loss |
| | GC | DFCN | Node | KL Loss, Reconstruction Loss |
| | | ARGA | Node | Adversarial Loss, Reconstruction Loss |
| | | GLCC | Graph | InfoNCE Loss |
| Semi | CSemi | LP | Node | Cross-Entropy Loss |
| | | TGNN | Graph | Consistency Loss, Cross-Entropy Loss |
| | | DualGraph | Graph | InfoNCE Loss, Cross-Entropy Loss |
| | Semi w DS | UDAGCN | Node | Cross-Entropy Loss |
| | | GTrans | Node | Surrogate Loss |
| | | DEAL | Graph | Adversarial Loss, Cross-Entropy Loss |
| Few-shot | | GFL | Node | Reconstruction Loss Cross-Entropy Loss |
| | | SuperClass | Graph | Cross-Entropy Loss |
| | | HSL-RG | Graph | InfoNCE Loss Cross-Entropy Loss |

Table 1: Analysis for representative DEGL works according to the data type and objective function. Self-supervised Graph Learning (Self); Semi-supervised Graph Learning (Semi); Few-shot Graph Learning (Few-shot); Graph Representation Learning (GRL); Graph Clustering (GC); Classical Semi-supervised Graph Learning (CSemi); Semi-supervised Graph Learning under Domain Shift (Semi w DS).

where $\mathcal{D}$ denotes the data distribution of the unlabeled graph, $f_\theta$ denotes the node or graph encoder, aiming at learning a low-dimensional representation $h_i \in \mathcal{H}$ for node $v_i$. The decoder $g_\theta$ takes the node representation $\mathcal{H}$ as its input and is generally well-designed on specific downstream tasks.

Existing GRL methods can be roughly divided into three categories, i.e., generation-based, contrastive-based, and auxiliary property-based methods.

**Generation-based GRL**
Generation-based GRL [Kim and Oh, 2021; Wang *et al.*, 2022b] aims at reconstructing the input full graph or sampled subgraphs. These methods contribute to understanding the underlying structure and node dependencies within the graph, which encourages the model to encode representations that preserve the inherent information. The self-supervised loss is generally designed to quantify the difference between the reconstructed and the original graph. In such case, Eq. (1) is derived as:

$$\min_{\theta,\psi} \mathcal{L}\left(g_\psi(f_\theta(\hat{\mathcal{G}})), \mathcal{G}\right), \tag{2}$$

where $\mathcal{G}$ denotes the perturbed graph data, $f_\theta(\cdot)$ and $g_\psi(\cdot)$ denote graph encoder and decoder respectively. MGAE [Wang

*et al.*, 2017], estimates raw features from the noisy input node features in each GNN layer, i.e., $\hat{\mathcal{G}} = (A, \hat{X})$, where $\hat{X}$ denotes the corrupted random noise. EdgeMask [Jin *et al.*, 2020] and SuperGAT [Jin *et al.*, 2020] are representative works that recover the graph structure. Specifically, EdgeMask recovers the similarity between the embeddings of two connected nodes, and SuperGAT adopts a hierarchical variational inference strategy to reconstruct the global topological structure.

**Contrastive-based GRL**
Contrastive-based GRL [You *et al.*, 2020; Luo *et al.*, 2023a; Ju *et al.*, 2024b] mainly incorporates contrastive learning on learning node representations by contrasting similar instances (positive pairs) against dissimilar instances (negative pairs). Contrastive-based models provide a powerful mechanism for exploiting the inherent structure and relationships within the graph. For this objective, Eq. (1) can be derived as:

$$\min_{\theta,\psi} \mathcal{L}\left(g_\psi(f_\theta(\hat{\mathcal{G}}_{\text{view1}}), f_\theta(\hat{\mathcal{G}}_{\text{view2}}))\right), \tag{3}$$

where $\hat{\mathcal{G}}_{\text{view1}}$ and $\hat{\mathcal{G}}_{\text{view2}}$ denote the augmented instances, which can be node pairs, graph pairs, and task pairs. $g_\psi$ denotes the similarity discriminator, which estimates the distances of contrastive instances. GraphCL [You *et al.*, 2020] contrasts two generated views of node feature masking and edge reconstruction, and then the Mutal Information is maximized between two target nodes from different views. JOAO [You *et al.*, 2021] proposes a bi-level optimization problem, which optimizes the augmentation selection module and contrastive objectives jointly. DGI [Veličković *et al.*, 2018] contrasts node-level representations with graph-level ones for better local-global interactions. For heterogeneous graphs, HeCo [Wang *et al.*, 2021] is designed to contrast two generated views of network schema and meta-paths.

**Auxiliary Property-based GRL**
Auxiliary property-based GRL [Zhu *et al.*, 2020; Luo *et al.*, 2022b] generates pseudo labels by leveraging classical attributive and graph algorithms, to enrich the self-supervision signals. Compared with the human-annotated labels in supervised learning, the pseudo labels in auxiliary property-based GRL do not need additional cost. The utilization of auxiliary properties in GRL enhances the robustness and generalization capability of the representations. Under this setting, Eq. (1) can be derived as:

$$\min_{\theta,\psi} \mathcal{L}\left(g_\psi(f_\theta(\mathcal{G})), c(\mathcal{G})\right), \tag{4}$$

where $c(\mathcal{G})$ denotes the annotated auxiliary properties, such as the shortest path and graph centrality. S$^2$GRL [Peng *et al.*, 2020] is designed to estimate the shortest paths among different nodes for better structural representations. CentralityScoreRanking [Hu *et al.*, 2019] predicts the ranking of centrality scores among node pairs, and compares the ranking with pseudo labels. NodeProperty [Jin *et al.*, 2020] adopts a node-level property construction task aiming at exploring the properties of node features. SimP-GCN [Jin *et al.*, 2021] designs a $k$-nearest-neighbor graph based on the node features to enhance the neighbor aggregation performance.

**Discussion**
Generation-based GRL is easier to implement as the reconstruction task is easy to build. But it is memory-consuming in certain cases such as dealing with large-scale graphs. Auxiliary property-based methods benefit from the uncomplicated decoders, while the selection of effective auxiliary properties is challenging. Compared with the above categories, contrastive-based ones are more flexible.

## 3.2 Graph Clustering

In self-supervised graph learning, an end-to-end learning paradigm that possesses a specific downstream task mainly boils down to graph clustering. Graph clustering aims to divide the nodes into $C$ disjoint clusters without label signals as reliable guidance. Recently, the graph clustering task has attracted widespread attention and extensive approaches have focused on this task with promising performance. In general, given an attribute graph $\mathcal{G}$, the framework of graph clustering includes a self-supervised neural network $F$ that outputs the node representations and a clustering network/traditional clustering $\Omega$ that exports the clustering result $\hat{\mathbf{Y}}$, i.e.,

$$\hat{\mathbf{Y}} = \Omega(F(\mathcal{G}), C).$$

Existing methods can be roughly partitioned into three main groups: reconstructive-based, adversarial-based, and contrastive-based approaches. In the following, we present a comprehensive overview of these algorithms.

**Reconstructive-based Methods**
Reconstructive-based methods typically encode either the attribute information or the structural information of the graph, aiming to reconstruct the input information to supervise the network training and achieve meaningful node representations. DAEGC [Wang *et al.*, 2019] performs representation learning through an attention network based on node features and structural information under the GAT paradigm. The network training is supervised by reconstructing the graph structure and a self-supervised clustering module. SDCN [Bo *et al.*, 2020] leverages auto-encoder (AE) to assist GCN, effectively alleviating the over-smoothing issue. It obtains semantically rich node representations by optimizing the reconstruction loss of the AE and a self-supervised clustering loss based on GCN. DFCN [Tu *et al.*, 2021] introduces a dynamic information fusion technique based on AE and graph auto-encoder (GAE) to explore attributive and structural information finely. It also designs a triplet self-supervision mechanism for self-supervised clustering. Based on [Tu *et al.*, 2021], DCRN [Liu *et al.*, 2022] reduces information correlation through a dual approach to prevent representation collapse and obtain discriminative node representations. R$^2$FGC [Yi *et al.*, 2023] builds upon the relational learning at the attributive and structural levels from both global and local views based on AE and GAE. It preserves relationships among positive samples and reduces redundant relationships among negative samples, thereby acquiring effective and discriminative node representations.

**Adversarial-based Methods**
Adversarial-based methods engage in a game between the generator and the discriminator to achieve robust node representations. AGAE [Tao *et al.*, 2019] combines adversarial learning and auto-encoder to perform representation learning, which introduces an adversarial regularization term and adaptive partition-dependent prior to guide the network training. ARGA [Pan *et al.*, 2019] incorporates an adversarial training scheme into the graph auto-encoder architecture to regularize the latent codes for learning a robust graph representation. The adversarial training module is designed to discern whether the latent codes originate from a real prior distribution or the graph encoder. Further, JANE [Yang *et al.*, 2020] designs this prior distribution by incorporating the node embeddings to capture the semantic variations.

**Contrastive-based Methods**
Contrastive-based methods enhance the discriminative power of learned representations by pulling positive samples closer and pushing negative samples farther apart to explore the semantic information. Under the graph contrastive learning framework, GDCL [Zhao *et al.*, 2021] utilizes pseudo-labels to randomly select samples from classes different from positive samples to form negative samples, which corrects the sampling bias and thus decreases the false-negative samples in graph clustering. SCAGC [Xia *et al.*, 2022] further constructs the self-consistent contrastive loss by treating all samples from the same class in both graph views as positive samples and all non-matching samples as negative samples. CGC [Park *et al.*, 2022] utilizes a multi-level scheme for the selection of positive and negative samples, ensuring their ability to reflect hierarchical community structures and network homophily. Moreover, CGC extends its applicability to temporal graph clustering, which is capable of detecting change points. Additionally, CONGREGATE [Sun *et al.*, 2023] reexamines graph clustering through a geometric lens. It constructs a novel heterogeneous curvature space for generating representations and introduces an augmentation-free reweighted contrastive method to focus more on both hard negatives and hard positives in the curvature space. SCGC [Liu *et al.*, 2023b] incorporates a low-pass denoising operation in pre-processing, employs un-shared siamese encoders to eliminate the need for graph augmentation in contrastive learning, and introduces a cross-view structural consistency objective function to boost the discriminative capability of the learned network and avert direct clustering-guided loss.

**Discussion**
In addition to attribute graphs, numerous studies have explored clustering tasks on other graph types, such as heterogeneous graphs and dynamic graphs. Furthermore, graph-level clustering is another research-worthy issue [Ju *et al.*, 2023a], but it has been relatively underexplored. Besides, graph clustering can be applied to practical applications, such as guiding recommendation services, analyzing protein-protein interaction networks, and uncovering cellular heterogeneity in single-cell RNA-Seq analyses, e.g. [Yu *et al.*, 2023].

# 4 Semi-supervised Graph Learning

## 4.1 Classical Semi-supervised Graph Learning

Semi-supervised learning is one of the most important tasks in machine learning, which manages to leverage an extensive corpus of unlabeled data to enhance the learning models

trained using comparatively limited labeled examples. Compared with other semi-supervised learning methods, classical semi-supervised graph learning emphasizes the structural information of the graph. This focus enables a detailed exploration of the intrinsic relationships and dependencies among data points within the graph.

Based on the different data types within the graphs, classical semi-supervised graph learning can be categorized into node-level semi-supervised graph learning and graph-level semi-supervised graph learning. For the node-level task, the most representative methods are based on label propagation on the graph. For the graph-level task, the methods can be divided into consistency regularization and pseudo-labeling.

### Label Propagation Methods

Label propagation (LP) is the most representative method for label inference on semi-supervised graph learning. The framework can be formulated as a propagation process in which some of the nodes with labels, also referred to as seeds, propagate their labels to unlabeled nodes based on the similarity of the connected nodes, which can be formulated as:

$$\mathcal{L} = \mathcal{L}_{cls}(f(X), \mathcal{V}_L) + \lambda f(X)^\top \Delta f(X), \tag{5}$$

where the first term $\mathcal{L}_{cls}$ is the classification loss which trains the model $f : \mathbb{R}^{N \times m}$ to predict the known labels $\mathcal{V}_L$, $m$ is the feature dimension. The second term is graph-based regularization, ensuring the connected nodes have a similar model output, $\Delta$ is the graph Laplacian, $\lambda \in \mathbb{R}$ is the regularization coefficient. For example, classical LP methods iteratively propagate the label of each data to its neighbors based on the constructed graph [Zhu, 2005]. GCN [Kipf and Welling, 2017] utilizes the power of graph neural networks and performs message passing on the graph to constrain neighborhood nodes with similar representations. GCN-LPA [Wang and Leskovec, 2021] further proposes to combine GCN and LP with learnable edge weights. The model views LP as regularization to assist the GCN in learning proper edge weights.

### Consistency Regularization Methods

There are also some endeavors using consistency regularization for semi-supervised graph learning. These methods are based on the manifold or the smoothness assumption which posits that realistic perturbations of the graph data should not change the output of the model, and can be formulated as:

$$\mathcal{L} = \mathcal{L}_s(f(X), \mathcal{G}_L) + \alpha \mathcal{L}_u(f(X)) + \beta \mathcal{R}(f(X)), \tag{6}$$

where $\mathcal{L}_s$ and $\mathcal{L}_u$ denote the supervised and unsupervised loss, $R$ denotes the consistency regularization loss. Typically, InfoGraph [Sun *et al.*, 2020] learns a supervised and an unsupervised model respectively and maximizes the mutual information between the two models. GraphCL [You *et al.*, 2020] and GLA [Yue *et al.*, 2022] leverage contrastive learning between a graph and its augmented views to learn graph representation for semi-supervised graph learning. TGNN [Ju *et al.*, 2022a] incorporates both graph neural network and graph kernels with consistency regularization loss to implicitly and explicitly explore graph structural knowledge from unlabeled data for semi-supervised graph classification.

### Pseudo-Labeling Methods

Pseudo-labeling is another type of popular method for semi-supervised graph learning, which predicts the label distribution of unlabeled data and selects confident samples to the training data set as labeled data, which can be defined as:

$$\mathcal{L} = \mathcal{L}_s(f(X), \mathcal{G}_L) + \alpha \mathcal{L}_s(f(X'), \mathcal{G}'_U), \tag{7}$$

where $X$ and $X'$ denote the supervised and selected unsupervised graph data feature, $\mathcal{G}_L$ and $\mathcal{G}'_U$ denote the label of the original labeled graph data and pseudo label of the unlabeled graph data respectively. For graph-structured data, SEAL-AI [Li *et al.*, 2019] and ASGN [Hao *et al.*, 2020] leverage active learning techniques to select the most representative graph samples from the unlabeled data. KGNN [Ju *et al.*, 2022b] adopts the posterior regularization to incorporate graph kernels as structured constraints, generating pseudo labels and guiding the training process of graph neural networks under the EM-style framework. DualGraph [Luo *et al.*, 2022a] further jointly learn a graph prediction and a graph retrieval module via posterior regularization during the pseudo-labeling process for semi-supervised graph learning.

### Discussion

Label propagation methods represent nodes as training samples, and each edge denotes some similarity measurement of the node pair. In contrast, consistency regularization methods usually rely on the consistency constraint of rich data transformations and pseudo-labeling methods rely on the high confidence of pseudo-labels, which can be added to the training data set as labeled data. Besides, some hybrid works integrating different types of methods into one unified framework [Sohn *et al.*, 2020] can be adapted to graph data.

## 4.2 Semi-supervised Graph Learning under Domain Shift

In real-world scenarios, graph applications often face out-of-distribution (OOD) challenges, which arise when the data distribution during inference differs from the data on which the model was trained. Furthermore, the discrepancy between training and inference is complicated by the lack of labeled data in new domains, making supervised adaptation impractical. To address these issues, the concept of Graph Domain Adaptation (GDA) has been introduced to account for these distribution shifts and facilitate effective knowledge transfer. To achieve data efficiency, GDA must operate with minimal reliance on large amounts of labeled data, which are often scarce or expensive to obtain. It emphasizes the value of leveraging prior knowledge and discriminative features from the available data. Efficient GDA methods work to extend model adaptability with limited target domain data while maintaining robust performance despite rapid changes in data distribution.

Based on the data dependency, GDA can be categorized into unsupervised GDA, which does not require labeled data from the target domain, and source-free GDA, which performs adaptation without accessing the source domain data.

### Unsupervised Graph Domain Adaptation

Unsupervised Graph Domain Adaptation (UGDA) uses both source and target graphs for training, and tests primarily on

the target graphs. UGDA methods can be broadly categorized into the following four types.

***Discrepancy-based methods.*** These methods usually apply discrepancy measurement, including MMD, Jensen–Shannon divergence, and Wasserstein distance, to measure the domain distribution shift. Formally, give source graphs $\mathcal{G}^s$, target graphs $\mathcal{G}^t$, and discrepancy measurement $d$, we have

$$\mathcal{L}_{dis} = d(\mathcal{G}^s, \mathcal{G}^t), \qquad (8)$$

When the domain discrepancy loss $\mathcal{L}_{dis}$ is minimized, the knowledge can be transferred from the source domain to the target domain. Among them, UDAGCN [Wu *et al.*, 2020] exploits local and global consistency within graphs and integrates an attention mechanism to fuse these consistencies into node representations. GRADE [Wu *et al.*, 2023] introduces the graph subtree discrepancy measure to capture the distribution shift. SpecReg [You *et al.*, 2023] proposes a tighter generalization bounded by spectral regularization. More recently, CoCo [Yin *et al.*, 2023] has achieved superior performance by synergizing two branches, including the graph message passing branch and the graph kernel branch.

***Adversarial adaptation methods.*** These methods typically utilize adversarial components to minimize domain discrepancies, where the graph encoder and domain classifier compete. Among them, AdaGCN [Dai *et al.*, 2019] employs adversarial domain adaptation techniques to acquire node representations that are invariant across varying domains, enabling effective knowledge transfer. ACDNE [Shen *et al.*, 2020] introduces adversarial learning across networks, in order to learn invariant node representations while preserving network structural information. DANE [Zhang *et al.*, 2020] is an adaptation method that attains domain-adaptive embedding through the utilization of a shared weight message passing network, complemented by adversarial learning regularization.

***Discrimination learning methods.*** Researchers propose to utilize self-training to improve the performance in the target domain, where pseudo-labeling is one of the popular techniques. These methods obtain the pseudo-labels by the network itself on unlabeled target data to expand the training set. Among them, DEAL [Yin *et al.*, 2022] uses augmented views to distill reliable pseudo-labels for better graph-level classification in the target domain. StruRW [Liu *et al.*, 2023a] introduces a structural reweighting approach that uses estimated edge probabilities based on the pseudo-labels to adjust the neighborhood bootstrapping of the graph neural network, counteracting conditional shifts between domains.

***Disentangle-based methods.*** These methods generally disentangle graph representation into domain-invariant and domain-relevant parts, and conduct domain adaptation with domain-invariant embeddings. Among them, ASN [Zhang *et al.*, 2021] utilize both graph model embedding and adversarial adaptation to generate network-invariant node representations. DGDA [Cai *et al.*, 2023] employs a variational graph autoencoder approach to separate semantic, domain, and random latent variables for graph domain adaptation.

### Source-free Graph Domain Adaptation
Source-free Graph Domain Adaptation (SFGDA) eliminates the dependence on source data. The assumption that source

and target data can be used for adaptation is not always true in real-world scenarios. On the one hand, privacy, confidentiality, and copyright issues may prevent access to the source data. On the other hand, the requirement to store the complete source dataset on devices is often impractical. Among SFGDA methods, GT3 [Wang *et al.*, 2022c] utilizes a self-supervised test-time training framework, with a unique balancing constraint to prevent distribution bias. GTrans [Jin *et al.*, 2023] introduces a data-centric method for target data transformation to improve generalization and robustness.

### Discussion
Graph domain adaptation aims at achieving data-efficient learning under domain shift. UGDA methods learn by modeling between domains, using labeled source domain data and unlabeled target domain data. SFGDA focuses on an even more data-efficient and more challenging problem, while the research in SFGDA is not comprehensive, and ongoing research continues to advance the field.

## 5 Few-shot Graph Learning
Few-shot graph learning aims to learn graph models to make accurate predictions with a small amount of labeled data, which usually adopts a meta-learning paradigm. Compared with semi-supervised graph learning concerning incomplete supervision, few-shot graph learning focuses on transferring prior knowledge across different tasks with high generalization capacity. Specifically, we aim to build a graph model that is readily tuneable to fit future meta-test tasks given a variety of meta-training tasks. In this section, we primarily concentrate on the most popular few-shot node classification and cover numerous few-shot graph classification works as well. Current few-shot graph learning approaches can be roughly divided into metric learning approaches and parameter optimization approaches, which we will elaborate on as below.

### 5.1 Metric Learning Methods
Metric learning methods typically incorporate ProNet [Snell *et al.*, 2017] into graph few-shot learning, which generates prototypes by averaging the node representations of GNNs in the support set for each class. Then, they encourage each query node to approach its corresponding prototypes. Formally, given each graph $\mathcal{G}$ with $N$ classes and a query set $\mathcal{V}^q$ of $K$ query nodes for each class, we have

$$\mathcal{L}_{ML} = - \sum_{(v_i, y_i) \in \mathcal{V}^q} \log \frac{\exp\left(-d\left(\mathbf{z}_i, \mathbf{p}_{y_i}\right)\right)}{\sum_c \exp\left(-d\left(\mathbf{z}_i, \mathbf{p}_c\right)\right)}, \qquad (9)$$

where $\mathbf{p}_c$ denotes the representations for the $c$-th class, $\mathbf{z}_i$ is the node representation of $v_i$. $d(\cdot, \cdot)$ is a distance metric, e.g., cosine distance. The episodic training would then be executed iteratively until convergence is achieved. In particular, GFL [Yao *et al.*, 2020] integrates hierarchical graph representation gates to enhance node representations with structure-specific knowledge and employs a reconstruction loss to ensure stability during training. To improve the generalization capacity, GPN [Ding *et al.*, 2020] estimates the importance of nodes for prototype reconstruction using a second GNN with additional adjustments based on centrality. HAG-Meta

[Tan *et al.*, 2022] additionally takes into account the scenarios involving newly encountered classes and introduces a regularization term based on task-level and novel-level attention to tackle the potential class imbalance issue. HGNN [Yu *et al.*, 2022] combines prototype GNN with an instance GNN, which consists of all support nodes and the query, and encourages the consistency between these complementary GNNs. SuperClass [Chauhan *et al.*, 2020] further computes prototype graphs for few-shot graph classification by minimizing the average distance to other similar graphs. Subsequently, it constructs a graph-of-graph to represent inter-class correlations for generalized graph learning.

## 5.2 Parameter Optimization Methods

Parameter optimization methods usually adhere to the paradigm of model-agnostic meta-learning (MAML) [Finn *et al.*, 2017; Liu *et al.*, 2020], which generates node representations using GNNs for meta-learning. They construct meta-tasks using random nodes and then conduct multiple gradient descent for each meta-task. Formally, for each sampled meta-task $\mathcal{T}_i$, given a GNN $f_\theta(\cdot)$ and learning rate $\lambda$, we have:

$$\theta'_i = \theta - \lambda \frac{\partial \mathcal{L}_{\mathcal{T}_i}(f_\theta)}{\partial \theta}. \tag{10}$$

Finally, these tasks would be aggregated to optimize the task-agnostic parameters as a whole:

$$\theta \leftarrow \theta - \beta \nabla_\theta \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}(f_{\theta'_i}), \tag{11}$$

where $\beta$ is the learning rate for meta-optimization. On the basis of the above optimization framework, numerous GNN approaches are developed. In particular, Meta-GNN [Zhou *et al.*, 2019] is the first work to combine meta-learning with GNNs and validate the effectiveness of GNNs in few-shot learning scenarios. AMM-GNN [Wang *et al.*, 2020] further incorporates the attention mechanism to capture the relationships among meta-tasks for more generalized knowledge. GLITTER [Wang *et al.*, 2022a] extracts relevant nodes according to their influence to get rid of potential interference and then constructs a task-specific graph by leveraging mutual information and nodal influence, which makes the best of meta-tasks for effective meta-learning. In regard to few-shot graph classification, AS-MAML [Ma *et al.*, 2020] analyzes the difficulty of applying MAML in graph domains and thus introduces an adaptive step controller that takes into account both training loss and the quality of graph embeddings to regulate the optimization procedure. HSL-RG [Ju *et al.*, 2023b] investigates the correlation between graph samples using graph kernels from a global perspective and introduces various augmentation strategies for self-supervised learning from a local perspective. The attention mechanism further integrates both complementary views into a task-adaptive meta-learning framework.

## 5.3 Discussion

In addition to the above works, several advanced techniques including prompt learning and contrastive learning have also been applied in few-shot graph learning [Tan *et al.*, 2023]. Besides, these few-shot learning works have also been extended to tackle zero-shot graph learning [Ju *et al.*, 2023c], which focuses on nodes from unseen classes.

## 6 Conclusion and Future Work

In summary, this paper provides a comprehensive overview of Data-Efficient Graph Learning. We initiate by discussing the current challenges in graph machine learning. Subsequently, we categorize existing works into three parts: self-supervised graph learning, semi-supervised graph learning, and few-shot graph learning. In each section, we present representative strategies and introduce their key technical contributions. Despite the progress made, there still remain several challenges in the field that warrant further research in the future.

**Inherent Challenges.** Though being efficient in general, the data-efficient graph learning models are inherently facing challenges such as robustness and generalizability.We might also face extra challenges when running data-efficient models on out-of-distribution data. Some real-world applications prefer explainable results, so that the model can be trusted. Drug discovery is a good example. All these considerations not only enhance the capabilities of existing data-efficient graph learning models but also steer the trajectory of research towards more versatile and reliable solutions for real-world scenarios.

**Combined with Large Models.** Recently, combining LLM (i.e., Large Language Models) with graph learning approaches is one of the trending topics [Yang *et al.*, 2023]. For example, one might use GNN components to learn the underlying knowledge graphs to improve LLMs for question-answering [Yasunaga *et al.*, 2021] or use LLMs to generate node embeddings in social network graphs [Liu *et al.*, 2021]. For data-efficient graph learning, we care about all components. Both the LLM-based components and GNN-based components are crucial for achieving effectiveness and efficiency. It is theoretically possible to achieve high performance with only few-shot or zero-shot learning on these relatively-complicated models.

**Towards Different Convolutional Algorithms.** There are some GNN models that are fundamentally different from most of the others. Therefore, data-efficient learning approaches that work elsewhere might no longer be applicable. For example, some GNNs are designed in non-Euclidean space. The non-Euclidean-space models (e.g., spherical or hyperbolic) are well-known for being good at handling certain substructures, such as trees or cycles. Researchers are still on their way to extend GNN models to non-Euclidean spaces, while believing it to be a promising direction. The data-efficient design in Euclidean space might no longer work for non-Euclidean space, and there might be data-efficient non-Euclidean GNN models that can not work in Euclidean space, highlighting the continuous evolution and diversification of GNN research.

**Proof of Efficiency.** In theory, data-efficient learning has the potential to be rigorously justified by mathematical proofs. So far, with the help of learnability theory, researchers have made strides in proving some theoretical bound under certain problem settings [Grohs and Voigtlaender, 2021]. However, so far, the majority of works in this field focus on fully-connected neural networks, together with fully-labeled data. If mathematical analysis could be provided in the data-efficient graph learning setting, it would undoubtedly provide valuable insights for future researchers, paving the way for more robust and informed advancements in this evolving domain.

## Acknowledgments

## References

[Bo et al., 2020] Deyu Bo, Xiao Wang, Chuan Shi, Meiqi Zhu, Emiao Lu, and Peng Cui. Structural deep clustering network. In WWW, 2020.

[Cai et al., 2023] Ruichu Cai, Fengzhu Wu, Zijian Li, Pengfei Wei, Lingling Yi, and Kun Zhang. Graph domain adaptation: A generative view. TKDD, 2023.

[Chauhan et al., 2020] Jatin Chauhan, Deepak Nathani, and Manohar Kaul. Few-shot learning on graphs via superclasses based on graph spectral measures. In ICLR, 2020.

[Dai et al., 2019] Quanyu Dai, Xiao-Ming Wu, Jiaren Xiao, Xiao Shen, and Dan Wang. Graph transfer learning via adversarial domain adaptation with graph convolution. TKDE, 2019.

[Ding et al., 2020] Kaize Ding, Jianling Wang, Jundong Li, Kai Shu, Chenghao Liu, and Huan Liu. Graph prototypical networks for few-shot learning on attributed networks. In CIKM, 2020.

[Finn et al., 2017] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In ICML, 2017.

[Grohs and Voigtlaender, 2021] Philipp Grohs and Felix Voigtlaender. Proof of the theory-to-practice gap in deep learning via sampling complexity bounds for neural network approximation spaces. arXiv preprint arXiv:2104.02746, 2021.

[Hao et al., 2020] Zhongkai Hao, Chengqiang Lu, Zhenya Huang, Hao Wang, Zheyuan Hu, Qi Liu, Enhong Chen, and Cheekong Lee. Asgn: An active semi-supervised graph neural network for molecular property prediction. In KDD, 2020.

[Hu et al., 2019] Ziniu Hu, Changjun Fan, Ting Chen, Kai-Wei Chang, and Yizhou Sun. Pre-training graph neural networks for generic structural feature extraction. arXiv preprint arXiv:1905.13728, 2019.

[Jin et al., 2020] Wei Jin, Tyler Derr, Haochen Liu, Yiqi Wang, Suhang Wang, Zitao Liu, and Jiliang Tang. Self-supervised learning on graphs: Deep insights and new direction. arXiv preprint arXiv:2006.10141, 2020.

[Jin et al., 2021] Wei Jin, Tyler Derr, Yiqi Wang, Yao Ma, Zitao Liu, and Jiliang Tang. Node similarity preserving graph convolutional networks. In WSDM, 2021.

[Jin et al., 2023] Wei Jin, Tong Zhao, Jiayuan Ding, Yozen Liu, Jiliang Tang, and Neil Shah. Empowering graph representation learning with test-time graph transformation. In ICLR, 2023.

[Ju et al., 2022a] Wei Ju, Xiao Luo, Meng Qu, Yifan Wang, Chong Chen, Minghua Deng, Xian-Sheng Hua, and Ming Zhang. Tgnn: A joint semi-supervised framework for graph-level classification. In IJCAI, 2022.

[Ju et al., 2022b] Wei Ju, Junwei Yang, Meng Qu, Weiping Song, Jianhao Shen, and Ming Zhang. Kgnn: Harnessing kernel-based networks for semi-supervised graph classification. In WSDM, 2022.

[Ju et al., 2023a] Wei Ju, Yiyang Gu, Binqi Chen, Gongbo Sun, Yifang Qin, Xingyuming Liu, Xiao Luo, and Ming Zhang. Glcc: A general framework for graph-level clustering. In AAAI, 2023.

[Ju et al., 2023b] Wei Ju, Zequn Liu, Yifang Qin, Bin Feng, Chen Wang, Zhihui Guo, Xiao Luo, and Ming Zhang. Few-shot molecular property prediction via hierarchically structured learning on relation graphs. Neural Networks, 2023.

[Ju et al., 2023c] Wei Ju, Yifang Qin, Siyu Yi, Zhengyang Mao, Kangjie Zheng, Luchen Liu, Xiao Luo, and Ming Zhang. Zero-shot node classification with graph contrastive embedding network. TMLR, 2023.

[Ju et al., 2024a] Wei Ju, Zheng Fang, Yiyang Gu, Zequn Liu, Qingqing Long, Ziyue Qiao, Yifang Qin, Jianhao Shen, Fang Sun, Zhiping Xiao, et al. A comprehensive survey on deep graph representation learning. Neural Networks, 2024.

[Ju et al., 2024b] Wei Ju, Yifan Wang, Yifang Qin, Zhengyang Mao, Zhiping Xiao, Junyu Luo, Junwei Yang, Yiyang Gu, Dongjie Wang, Qingqing Long, et al. Towards graph contrastive learning: A survey and beyond. arXiv preprint arXiv:2405.11868, 2024.

[Kim and Oh, 2021] Dongkwan Kim and Alice Oh. How to find your friendly neighborhood: Graph attention design with self-supervision. ICLR, 2021.

[Kipf and Welling, 2017] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In ICLR, 2017.

[Li et al., 2019] Jia Li, Yu Rong, Hong Cheng, Helen Meng, Wenbing Huang, and Junzhou Huang. Semi-supervised graph classification: A hierarchical graph perspective. In WWW, 2019.

[Liu et al., 2020] Zequn Liu, Ruiyi Zhang, Yiping Song, and Ming Zhang. When does maml work the best? an empirical study on model-agnostic meta-learning in nlp applications. arXiv preprint arXiv:2005.11700, 2020.

[Liu et al., 2021] Yujia Liu, Kang Zeng, Haiyang Wang, Xin Song, and Bin Zhou. Content matters: A gnn-based model combined with text semantics for social network cascade prediction. In PAKDD, 2021.

[Liu et al., 2022] Yue Liu, Wenxuan Tu, Sihang Zhou, Xinwang Liu, Linxuan Song, Xihong Yang, and En Zhu. Deep graph clustering via dual correlation reduction. In AAAI, 2022.

[Liu et al., 2023a] Shikun Liu, Tianchun Li, Yongbin Feng, Nhan Tran, Han Zhao, Qiang Qiu, and Pan Li. Structural

re-weighting improves graph domain adaptation. In *ICML*, 2023.

[Liu *et al.*, 2023b] Yue Liu, Xihong Yang, Sihang Zhou, Xinwang Liu, Siwei Wang, Ke Liang, Wenxuan Tu, and Liang Li. Simple contrastive graph clustering. *TNNLS*, 2023.

[Luo *et al.*, 2022a] Xiao Luo, Wei Ju, Meng Qu, Chong Chen, Minghua Deng, Xian-Sheng Hua, and Ming Zhang. Dualgraph: Improving semi-supervised graph classification via dual contrastive learning. In *ICDE*, 2022.

[Luo *et al.*, 2022b] Xiao Luo, Wei Ju, Meng Qu, Yiyang Gu, Chong Chen, Minghua Deng, Xian-Sheng Hua, and Ming Zhang. Clear: Cluster-enhanced contrast for self-supervised graph representation learning. *TNNLS*, 2022.

[Luo *et al.*, 2023a] Xiao Luo, Wei Ju, Yiyang Gu, Zhengyang Mao, Luchen Liu, Yuhui Yuan, and Ming Zhang. Self-supervised graph-level representation learning with adversarial contrastive learning. *TKDD*, 2023.

[Luo *et al.*, 2023b] Xiao Luo, Yusheng Zhao, Yifang Qin, Wei Ju, and Ming Zhang. Towards semi-supervised universal graph classification. *TKDE*, 2023.

[Ma *et al.*, 2020] Ning Ma, Jiajun Bu, Jieyu Yang, Zhen Zhang, Chengwei Yao, Zhi Yu, Sheng Zhou, and Xifeng Yan. Adaptive-step graph meta-learner for few-shot graph classification. In *CIKM*, 2020.

[Pan *et al.*, 2019] Shirui Pan, Ruiqi Hu, Sai-fu Fung, Guodong Long, Jing Jiang, and Chengqi Zhang. Learning graph embedding with adversarial training methods. *TCYB*, 2019.

[Park *et al.*, 2022] Namyong Park, Ryan Rossi, Eunyee Koh, Iftikhar Ahamath Burhanuddin, Sungchul Kim, Fan Du, Nesreen Ahmed, and Christos Faloutsos. Cgc: Contrastive graph clustering forcommunity detection and tracking. In *WWW*, 2022.

[Peng *et al.*, 2020] Zhen Peng, Yixiang Dong, Minnan Luo, Xiao-Ming Wu, and Qinghua Zheng. Self-supervised graph representation learning via global context prediction. *arXiv preprint arXiv:2003.01604*, 2020.

[Shen *et al.*, 2020] Xiao Shen, Quanyu Dai, Fu-lai Chung, Wei Lu, and Kup-Sze Choi. Adversarial deep network embedding for cross-network node classification. In *AAAI*, 2020.

[Snell *et al.*, 2017] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *NeurIPS*, 2017.

[Sohn *et al.*, 2020] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *NeurIPS*, 2020.

[Sun *et al.*, 2020] Fan-Yun Sun, Jordan Hoffmann, Vikas Verma, and Jian Tang. Infograph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization. In *ICLR*, 2020.

[Sun *et al.*, 2023] Li Sun, Feiyang Wang, Junda Ye, Hao Peng, and Philip S Yu. Congregate: contrastive graph clustering in curvature spaces. In *IJCAI*, 2023.

[Tan *et al.*, 2022] Zhen Tan, Kaize Ding, Ruocheng Guo, and Huan Liu. Graph few-shot class-incremental learning. In *WSDM*, 2022.

[Tan *et al.*, 2023] Zhen Tan, Ruocheng Guo, Kaize Ding, and Huan Liu. Virtual node tuning for few-shot node classification. In *KDD*, 2023.

[Tao *et al.*, 2019] Zhiqiang Tao, Hongfu Liu, Jun Li, Zhaowen Wang, and Yun Fu. Adversarial graph embedding for ensemble clustering. In *IJCAI*, 2019.

[Tu *et al.*, 2021] Wenxuan Tu, Sihang Zhou, Xinwang Liu, Xifeng Guo, Zhiping Cai, En Zhu, and Jieren Cheng. Deep fusion clustering network. In *AAAI*, 2021.

[Veličković *et al.*, 2018] Petar Veličković, William Fedus, William L Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. Deep graph infomax. *arXiv preprint arXiv:1809.10341*, 2018.

[Wang and Leskovec, 2021] Hongwei Wang and Jure Leskovec. Combining graph convolutional neural networks and label propagation. *TOIS*, 2021.

[Wang *et al.*, 2017] Chun Wang, Shirui Pan, Guodong Long, Xingquan Zhu, and Jing Jiang. Mgae: Marginalized graph autoencoder for graph clustering. In *CIKM*, 2017.

[Wang *et al.*, 2019] Chun Wang, Shirui Pan, Ruiqi Hu, Guodong Long, Jing Jiang, and Chengqi Zhang. Attributed graph clustering: A deep attentional embedding approach. *arXiv preprint arXiv:1906.06532*, 2019.

[Wang *et al.*, 2020] Ning Wang, Minnan Luo, Kaize Ding, Lingling Zhang, Jundong Li, and Qinghua Zheng. Graph few-shot learning with attribute matching. In *CIKM*, 2020.

[Wang *et al.*, 2021] Xiao Wang, Nian Liu, Hui Han, and Chuan Shi. Self-supervised heterogeneous graph neural network with co-contrastive learning. In *KDD*, 2021.

[Wang *et al.*, 2022a] Song Wang, Chen Chen, and Jundong Li. Graph few-shot learning with task-specific structures. In *NeurIPS*, 2022.

[Wang *et al.*, 2022b] Yifan Wang, Yiping Song, Shuai Li, Chaoran Cheng, Wei Ju, Ming Zhang, and Sheng Wang. Disencite: Graph-based disentangled representation learning for context-specific citation generation. In *AAAI*, 2022.

[Wang *et al.*, 2022c] Yiqi Wang, Chaozhuo Li, Wei Jin, Rui Li, Jianan Zhao, Jiliang Tang, and Xing Xie. Test-time training for graph neural networks. *arXiv preprint arXiv:2210.08813*, 2022.

[Wu *et al.*, 2020] Man Wu, Shirui Pan, Chuan Zhou, Xiaojun Chang, and Xingquan Zhu. Unsupervised domain adaptive graph convolutional networks. In *WWW*, 2020.

[Wu *et al.*, 2023] Jun Wu, Jingrui He, and Elizabeth Ainsworth. Non-iid transfer learning on graphs. In *AAAI*, 2023.

[Xia *et al.*, 2022] Wei Xia, Qianqian Wang, Quanxue Gao, Ming Yang, and Xinbo Gao. Self-consistent contrastive attributed graph clustering with pseudo-label prompt. *TMM*, 2022.

[Yang *et al.*, 2020] Liang Yang, Yuexue Wang, Junhua Gu, Chuan Wang, Xiaochun Cao, and Yuanfang Guo. Jane: Jointly adversarial network embedding. In *IJCAI*, 2020.

[Yang *et al.*, 2023] Junwei Yang, Hanwen Xu, Srbuhi Mirzoyan, Tong Chen, Zixuan Liu, Wei Ju, Luchen Liu, Ming Zhang, and Sheng Wang. Poisoning scientific knowledge using large language models. *bioRxiv*, 2023.

[Yao *et al.*, 2020] Huaxiu Yao, Chuxu Zhang, Ying Wei, Meng Jiang, Suhang Wang, Junzhou Huang, Nitesh Chawla, and Zhenhui Li. Graph few-shot learning via knowledge transfer. In *AAAI*, 2020.

[Yasunaga *et al.*, 2021] Michihiro Yasunaga, Hongyu Ren, Antoine Bosselut, Percy Liang, and Jure Leskovec. Qa-gnn: Reasoning with language models and knowledge graphs for question answering. *arXiv preprint arXiv:2104.06378*, 2021.

[Yi *et al.*, 2023] Siyu Yi, Wei Ju, Yifang Qin, Xiao Luo, Luchen Liu, Yongdao Zhou, and Ming Zhang. Redundancy-free self-supervised relational learning for graph clustering. *TNNLS*, 2023.

[Yin *et al.*, 2022] Nan Yin, Li Shen, Baopu Li, Mengzhu Wang, Xiao Luo, Chong Chen, Zhigang Luo, and Xian-Sheng Hua. Deal: An unsupervised domain adaptive framework for graph-level classification. In *MM*, 2022.

[Yin *et al.*, 2023] Nan Yin, Li Shen, Mengzhu Wang, Long Lan, Zeyu Ma, Chong Chen, Xian-Sheng Hua, and Xiao Luo. Coco: A coupled contrastive framework for unsupervised domain adaptive graph classification. In *ICML*, 2023.

[You *et al.*, 2020] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. Graph contrastive learning with augmentations. In *NeurIPS*, 2020.

[You *et al.*, 2021] Yuning You, Tianlong Chen, Yang Shen, and Zhangyang Wang. Graph contrastive learning automated. In *International Conference on Machine Learning*, 2021.

[You *et al.*, 2023] Yuning You, Tianlong Chen, Zhangyang Wang, and Yang Shen. Graph domain adaptation via theory-grounded spectral regularization. In *ICLR*, 2023.

[Yu *et al.*, 2022] Tianyuan Yu, Sen He, Yi-Zhe Song, and Tao Xiang. Hybrid graph neural networks for few-shot learning. In *AAAI*, 2022.

[Yu *et al.*, 2023] Zhuohan Yu, Yanchi Su, Yifu Lu, Yuning Yang, Fuzhou Wang, Shixiong Zhang, Yi Chang, Ka-Chun Wong, and Xiangtao Li. Topological identification and interpretation for single-cell gene regulation elucidation across multiple platforms using scmgca. *Nature Communications*, 2023.

[Yue *et al.*, 2022] Han Yue, Chunhui Zhang, Chuxu Zhang, and Hongfu Liu. Label-invariant augmentation for semi-supervised graph classification. In *NeurIPS*, 2022.

[Zhang *et al.*, 2020] Yizhou Zhang, Guojie Song, Lun Du, Shuwen Yang, and Yilun Jin. Domain adaptive network embedding. *arXiv preprint arXiv:1906.00684*, 2020.

[Zhang *et al.*, 2021] Xiaowen Zhang, Yuntao Du, Rongbiao Xie, and Chongjun Wang. Adversarial separation network for cross-network node classification. In *CIKM*, 2021.

[Zhao *et al.*, 2021] Han Zhao, Xu Yang, Zhenru Wang, Erkun Yang, and Cheng Deng. Graph debiased contrastive learning with joint representation clustering. In *IJCAI*, 2021.

[Zhou *et al.*, 2019] Fan Zhou, Chengtai Cao, Kunpeng Zhang, Goce Trajcevski, Ting Zhong, and Ji Geng. Meta-gnn: On few-shot node classification in graph meta-learning. In *CIKM*, 2019.

[Zhu *et al.*, 2020] Yanqiao Zhu, Yichen Xu, Feng Yu, Shu Wu, and Liang Wang. Cagnn: Cluster-aware graph neural networks for unsupervised graph representation learning. *arXiv preprint arXiv:2009.01674*, 2020.

[Zhu, 2005] Xiaojin Zhu. *Semi-supervised learning with graphs*. Carnegie Mellon University, 2005.