A Survey of Pretraining on Graphs: Taxonomy, Methods, and Applications

Jun Xia[1,2] , **Yanqiao Zhu**[3,4] , **Yuanqi Du**[5] and **Stan Z. Li**[1,2*]
[1]School of Engineering, Westlake University
[2]Institute of Advanced Technology, Westlake Institute for Advanced Study
[3]Center for Research on Intelligent Perception and Computing,
Institute of Automation, Chinese Academy of Sciences
[4]School of Artificial Intelligence, University of Chinese Academy of Sciences
[5]Department of Computer Science, George Mason University
{xiajun, stan.zq.li}@westlake.edu.cn, yanqiao.zhu@cripac.ia.ac.cn, ydu6@gmu.edu

Abstract

Pretrained Language Models (PLMs) such as BERT have revolutionized the landscape of Natural Language Processing (NLP). Inspired by their proliferation, tremendous efforts have been devoted to Pretrained Graph Models (PGMs). Owing to the powerful model architectures of PGMs, abundant knowledge from massive labeled and unlabeled graph data can be captured. The knowledge implicitly encoded in model parameters can benefit various downstream tasks and help to alleviate several fundamental issues of learning on graphs. In this paper, we provide the first comprehensive survey for PGMs. We firstly present the limitations of graph representation learning and thus introduce the motivation for graph pre-training. Then, we systematically categorize existing PGMs based on a taxonomy from four different perspectives. Next, we present the applications of PGMs in social recommendation and drug discovery. Finally, we outline several promising research directions that can serve as a guideline for future research.

1 Introduction

The developments of deep neural networks have revolutionized many machine learning tasks in recent years, ranging from image recognition to natural language processing. However, there are still many non-Euclidean graph datasets in real-world applications such as social networks and biochemical graphs which existing neural networks cannot handle with. Recent years have witnessed the prosperity of Graph Neural Networks (GNNs) [Wu *et al.*, 2020] that extend deep learning approaches for such graph-structured data. However, two fundamental challenges impede the wider usage of existing GNN models: (1) *Scarce labeled data:* Task-specific labeled data can be extremely scarce especially for biochemical graphs where high-quality data labeling often requires time-consuming and expensive wet-lab experiments [Xia *et al.*, 2021a]. (2) *Out-of-distribution generalization:* Existing GNNs lack out-of-distribution generalization abilities so that their performance substantially degrades when there exist distribution shifts between training and testing graph data. Indeed, nearly all of the deep learning domains are confronted with these challenges. To overcome these challenges, certain progress has been made. For example, the pretrain-then-finetune paradigm is thriving in NLP community. Specifically, they first pretrain the models on large-scale corpus and then fine-tune these models in various downstream tasks. It is widely recognized that this paradigm can provide a better initial point across various downstream tasks and leads to wider optima with better generalization than training from scratch [Hao *et al.*, 2019]. With the emergence of the Transformer architecture [Vaswani *et al.*, 2017], PLMs such as BERT [Devlin *et al.*, 2019] have emerged as a fundamental model for NLP and have established state-of-the-art results for a large variety of NLP tasks.

Inspired by the prosperity of PLMs, tremendous efforts have been devoted to Pretrained Graph Models (PGMs). In this paper, we present the first comprehensive survey for PGMs including their brief history, encoder architectures, pretraining strategies, various extensions, tuning strategies, and applications. Existing reviews related to this area have only partially focused on self-supervised learning on graphs [Liu *et al.*, 2021a; Xie *et al.*, 2021], but did not go broader to the other important ingredients of PGMs such as supervised pretraining, tuning strategies, various extensions, and etc. Overall, the contributions of this survey can be summarized as follows:

- *Comprehensive review.* To the best of our knowledge, our survey is the first work that presents a comprehensive review for PGMs, which provides graph researchers a synthesis and pointer to related works.

- *New taxonomy.* As revealed in Figure 1, we propose a new taxonomy, which categorizes existing PGMs from four different perspectives: (1) History; (2) encoder architectures; (3) Pretraining strategies; (4) Tuning strategies.

- *Abundant resources.* We collect abundant resources on PGMs, including open-sourced implementations, pre-

[*] Corresponding author

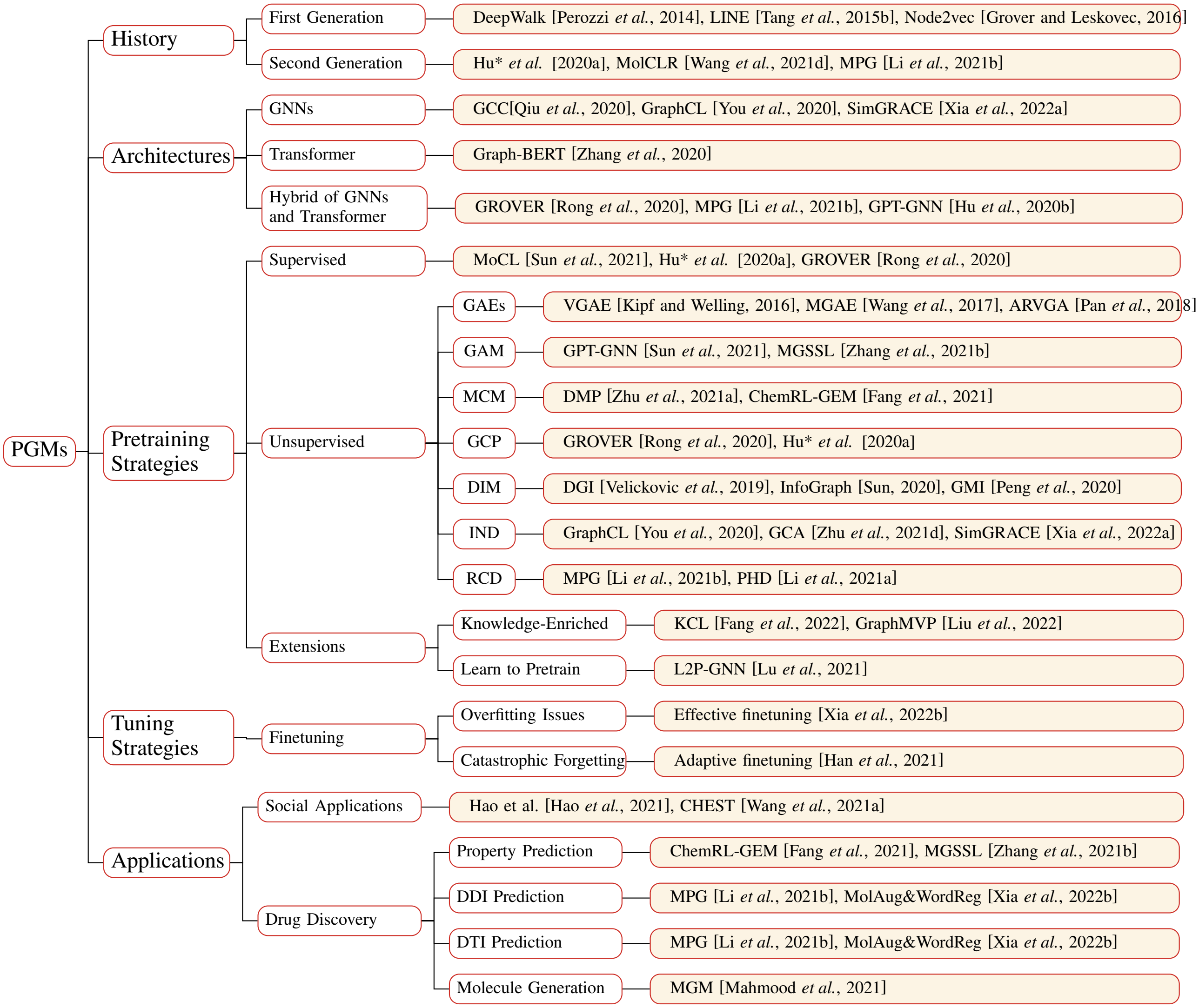
PGMs
History
First Generation
DeepWalk [Perozzi et al., 2014], LINE [Tang et al., 2015b], Node2vec [Grover and Leskovec, 2016]
Second Generation
Hu* et al. [2020a], MolCLR [Wang et al., 2021d], MPG [Li et al., 2021b]
Architectures
GNNs
GCC[Qiu et al., 2020], GraphCL [You et al., 2020], SimGRACE [Xia et al., 2022a]
Transformer
Graph-BERT [Zhang et al., 2020]
Hybrid of GNNs and Transformer
GROVER [Rong et al., 2020], MPG [Li et al., 2021b], GPT-GNN [Hu et al., 2020b]
Pretraining Strategies
Supervised
MoCL [Sun et al., 2021], Hu* et al. [2020a], GROVER [Rong et al., 2020]
Unsupervised
GAEs
VGAE [Kipf and Welling, 2016], MGAE [Wang et al., 2017], ARVGA [Pan et al., 2018]
GAM
GPT-GNN [Sun et al., 2021], MGSSL [Zhang et al., 2021b]
MCM
DMP [Zhu et al., 2021a], ChemRL-GEM [Fang et al., 2021]
GCP
GROVER [Rong et al., 2020], Hu* et al. [2020a]
DIM
DGI [Velickovic et al., 2019], InfoGraph [Sun, 2020], GMI [Peng et al., 2020]
IND
GraphCL [You et al., 2020], GCA [Zhu et al., 2021d], SimGRACE [Xia et al., 2022a]
RCD
MPG [Li et al., 2021b], PHD [Li et al., 2021a]
Extensions
Knowledge-Enriched
KCL [Fang et al., 2022], GraphMVP [Liu et al., 2022]
Learn to Pretrain
L2P-GNN [Lu et al., 2021]
Tuning Strategies
Finetuning
Overfitting Issues
Effective finetuning [Xia et al., 2022b]
Catastrophic Forgetting
Adaptive finetuning [Han et al., 2021]
Applications
Social Applications
Hao et al. [Hao et al., 2021], CHEST [Wang et al., 2021a]
Drug Discovery
Property Prediction
ChemRL-GEM [Fang et al., 2021], MGSSL [Zhang et al., 2021b]
DDI Prediction
MPG [Li et al., 2021b], MolAug&WordReg [Xia et al., 2022b]
DTI Prediction
MPG [Li et al., 2021b], MolAug&WordReg [Xia et al., 2022b]
Molecule Generation
MGM [Mahmood et al., 2021]

Figure 1: Taxonomy of PGMs with Representative Examples.

training database, pretrained models, and paper lists[1].

- *Future directions.* We discuss and analyze limitations of existing PGMs. Also, we suggest possible future research directions.

2 Brief History of PGMs

As early as 2006, the breakthrough of deep learning came with greedy layer-wise unsupervised pretraining followed by supervised finetuning [Hinton and Salakhutdinov, 2006]. With the development of computational power, the emergence of the deep models including GNNs and Transformer, and the constant enhancement of training strategies, PGMs have scored remarkable progress. The development of PGMs broadly falls into two generations according to their different usage, which we will elaborate below.

2.1 First-Generation PGMs: Pretrained Graph Embeddings

The first-generation PGMs aim to learn good graph embeddings for various tasks such as node clustering, link prediction, and visualization while these models themselves are no longer needed by downstream tasks. Initially, inspired by the Skip-Gram [Mikolov *et al.*, 2013] model for word embedding, DeepWalk [Perozzi *et al.*, 2014] pioneers graph embedding by considering the node paths traversed by random walks over graphs as sentences and leveraging Skip-Gram for learning latent node representations. LINE [Tang *et al.*, 2015b] learns embeddings by preserving the first- or second-order proximity separately and proposes an edge-sampling method for model inference. Following DeepWalk, Node2vec [Grover and Leskovec, 2016] defines a flexible notion of a node's network neighborhood and designs a biased random walk procedure, which can explore diverse neighborhoods efficiently. Besides, some researchers also try

[1] https://github.com/junxia97/awesome-pretrain-on-graphs

to learn embeddings for heterogeneous graphs, sub-graphs, and molecular graph such as PTE [Tang *et al.*, 2015a], subgraph2vec [Narayanan *et al.*, 2016] and N-gram Graph [Liu *et al.*, 2019]. Although pretrained graph embeddings have shown effective in graph-related tasks, the learned embeddings cannot be used to initialize other models for finetuning over other tasks and thus impede wider applications.

2.2 Second-Generation PGMs: Pretrained Encoders

With the emergence of GNNs and Transformer of high expressive ability, recent PGMs have embraced a transfer learning setting where the goal is to pretrain a generic encoder that can deal with various tasks. Apart from learning universal graph embeddings for downstream tasks as the first-generation PGMs, the second-generation PGMs can also provide better model initialization, which usually leads to better generalization performance and speeds up convergence on the target tasks [Hao *et al.*, 2019]. For example, Hu* *et al.* [2020a] initialize a 5-layer Graph Isomorphism Network (GIN) [Xu *et al.*, 2019] with the pretrained model obtained with both graph-level and node-level pretraining strategies. Also, GCC [Qiu *et al.*, 2020] utilizes a 5-layer GIN to extract representations for subgraphs and adopts subgraph discrimination in and across networks as the pretraining strategy to learn the intrinsic and transferable structural representations. Since these precursor PGMs, the modern PGMs are usually trained with larger scale database, more powerful architectures (e.g., hybrid of GNNs and Transformer), and more effective pretraining strategies. For example, the huge PGMs with ten millions of parameters trained on ten millions of molecular graphs have shown their powerful ability in learning universal molecular graph representations, such as GROVER [Rong *et al.*, 2020], MPG [Li *et al.*, 2021b], and DMP [Zhu *et al.*, 2021a].

3 Encoder Architectures

The encoder architectures of modern PGMs broadly fall into three categories: Graph Neural Networks, Transformer and the hybrid of GNNs and Transformer. We briefly introduce them below.

3.1 Graph Neural Networks (GNNs)

In GNNs, the structure of graph data guides the aggregation of local neighborhood information, which leads to a more contextual representation for each node. Also, we can adopt a graph pooling operation to get the representation for the whole graph. For PGMs, GIN is the most popular encoder due to its high expressive power [Hu* *et al.*, 2020a]. Besides, Heterogeneous Attention Network (HAN) [Wang *et al.*, 2019] is a suitable alternative for pretraining on heterogeneous graphs [Wang *et al.*, 2021c; Zhu *et al.*, 2022].

3.2 Transformer

Transformer architecture is composed of a stack of identical Transformer layers, which consists of a multi-head attention module followed by a feed-forward module, with a residual connection around each. To leverage the high expressiveness of Transformer, Graph-BERT [Zhang *et al.*, 2020] adopts 2-layer transformer as the encoder. Its input only consists of features and positional embeddings of the target node and the surrounding context, which is more efficient than existing GNNs relying on a complete input graph.

3.3 Hybrid of GNNs and Transformer

To leverage the advantages of GNNs and Transformer simultaneously, several recent works try to integrate GNNs into Transformer-style models. For PGMs, GROVER [Rong *et al.*, 2020] utilizes GNNs to capture local structural information of the graph data and then the outputs of the GNNs as queries, keys, and values for Transformer encoder. They claim that this bi-level information extraction strategy largely enhances the representation power. Analogously, MPG [Li *et al.*, 2021b] devises a neighbor attention (multi-head attention in essence) module to produce a message representation for each node and feed it to a fully connected feed-forward network. With the proper representations obtained, they adopt a GRU network [Cho *et al.*, 2014] to update node representations. For heterogeneous graphs, Heterogeneous Graph Transformer (HGT) [Hu *et al.*, 2020c] is a popular encoder for pretraining [Hu *et al.*, 2020b; Jiang *et al.*, 2021].

4 Pretraining Strategies

In this section, we elaborate on both supervised and unsupervised pretraining strategies, followed by the extensions of PGMs.

4.1 Supervised Tasks

Although the supervised labels are often time-consuming and expensive to collect, some cheaper annotations that may be less related to downstream tasks can also help pretrain on graphs, especially in biochemical domains. For example, Hu* *et al.* [2020a] propose to pretrain GNNs by predicting essentially all the properties of molecules that have been experimentally measured previously. Analogously, for protein function prediction, they pretrain GNNs to predict the existence of diverse protein functions that have been validated. Also, they leave a future work to take the structural similarities between two graphs as supervision. Inspired by this, MoCL [Sun *et al.*, 2021] calculates the Tanimoto coefficient [Bajusz *et al.*, 2015] between two molecules to measure their structural similarity, which serves as the pretraining objective. For molecular graphs, functional groups, one important class of motifs in molecules, encode rich domain knowledge of molecules and can be easily detected by the professional software such as RDkit[2]. In light of this, GROVER [Rong *et al.*, 2020] and MGSSL [Zhang *et al.*, 2021b] propose to predict the presence of the motifs or generate the motifs respectively. Although the supervised pretraining brings remarkable improvement, they often require domain-specific knowledge which significantly limits their application. Additionally, some supervised

[2] https://www.rdkit.org/

Table 1: Loss functions of various unsupervised pretraining strategies.

Task	Loss Function	Description
GAEs	$\mathcal{L}_{\text{GAEs}} = -\log p\left(\mathcal{X}, \mathcal{E} \mid \mathcal{G}\right)$	$\mathcal{X}, \mathcal{E}$ and $\mathcal{G}$ are the attributes, edges and input graph respectively.
GAM	$\mathcal{L}_{\text{GAM}} = -\sum_{i=1}^{\lvert\mathcal{V}\rvert} \log p\left(\mathcal{X}_i, \mathcal{E}_i \mid \mathcal{X}_{<i}, \mathcal{E}_{<i}\right)$	$\mathcal{X}_{<i}, \mathcal{E}_{<i}$ are the attributes and edges generated before node i respectively.
MCM	$\mathcal{L}_{\text{MCM}} = -\sum_{\widehat{\mathcal{G}} \in m(\mathcal{G})} \log p\left(\widehat{\mathcal{G}} \mid \mathcal{G}_{\setminus m(\mathcal{G})}\right)$	$m(\mathcal{G})$ are the masked components from $\mathcal{G}$ and $\mathcal{G}_{\setminus m(\mathcal{G})}$ are the rest.
GCP	$\mathcal{L}_{\text{GCP}} = -\log p(t \mid \mathcal{G}_1, \mathcal{G}_2)$	$t = 1$ if neighborhood graph $\mathcal{G}_1$ and contexts $\mathcal{G}_2$ belong to the same node.
IND	$\mathcal{L}_{\text{IND}} = -s\left(\mathcal{G}, \mathcal{G}^{+}\right) + \log \sum_{\mathcal{G}^{-} \in \mathcal{N}} s\left(\mathcal{G}, \mathcal{G}^{-}\right)$	$\mathcal{N}$ is a set of negatives; $\mathcal{G}^{+}$ is a positive sample; $s(\cdot, \cdot)$: similarity measure.
DIM	$\mathcal{L}_{\text{IND}} = -s\left(\mathcal{G}, \mathcal{C}\right) + \log \sum_{\mathcal{C}^{-} \in \mathcal{N}} s\left(\mathcal{G}, \mathcal{C}^{-}\right)$	$\mathcal{N}$ is a set of negatives; $\mathcal{C}$ is a substructure of $\mathcal{G}$.
RCD	$\mathcal{L}_{\text{RCD}} = -\log p(t \mid \mathcal{G}_1, \mathcal{G}_2)$	$t = 1$ if two half graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ are homologous couples.

pretraining strategies might be unrelated to the downstream task of interest and can even be harmful to them.

4.2 Unsupervised Tasks

4.2.1 Graph Reconstruction

Graph reconstruction serves as a natural supervision for pretraining on graphs. The prediction targets in graph reconstruction are certain parts of the given graphs such as the attribute of a subset of nodes or the existence of edge between a pair of nodes [Hu *et al.*, 2019]. The graph reconstruction-based pretraining strategies broadly fall into two categories: *Graph AutoEncoders* and *Graph Autoregressive Modeling*.

Graph AutoEncoders (GAEs) Inspired by the success of AutoEncoders in CV and NLP, various GAEs have been proposed recently. Among them, GAE [Kipf and Welling, 2016] is the simplest model, which reconstructs the adjacency matrix. Also, there exist multiple variants of GAEs that utilize graph reconstruction to pretrain the GNNs. Representative examples include VGAE [Kipf and Welling, 2016], MGAE [Wang *et al.*, 2017], ARVGA [Pan *et al.*, 2018], SIG-VAE [Hasanzadeh *et al.*, 2019], and many others.

Graph Autoregressive Modeling (GAM) Following the idea of GPT [Brown *et al.*, 2020] that conducts generative language model pretraining, GPT-GNN [Hu *et al.*, 2020b] proposes an autoregressive framework to perform reconstruction on given graphs iteratively, which is different from GAEs that reconstruct the graph all at once. In particular, given a graph with its nodes and edges randomly masked, GPT-GNN generates one masked node and its edges at a time and optimizes the parameterized models via maximizing the likelihood of the node and edges generated in the current iteration. Similarly, MGSSL [Zhang *et al.*, 2021b] generates molecular graph motifs in an autoregressive way based on existing motifs and connections.

4.2.2 Masked Components Modeling (MCM)

Similar to masked language modeling in NLP, MCM first masks out some components from the graphs and then trains the GNNs model to predict them. For example, Hu et.al [Hu* *et al.*, 2020a] propose attribute masking where the input node/edge attributes are randomly masked, and the GNN is asked to predict them. Also, GROVER [Rong *et al.*, 2020] tries to predict the masked subgraphs to capture the contextual information in molecular graphs. These masking methods are especially beneficial for graphs with rich annotations from scientific domains. For example, masking nodes attributes (atom type) in molecular graphs enables GNNs to learn simple chemistry rules such as valency, as well as potentially more complex chemistry phenomenon.

4.2.3 Graph Context Prediction (GCP)

GCP is proposed to explore the distribution of graph structure in graph data. For example, Hu* *et al.* [2020a] use subgraphs to predict their surrounding graph structures. They pretrain a GNN so that it maps nodes appearing in similar structural contexts to nearby embeddings. GROVER tries to predict the context-aware properties of the target node/edge within the same local subgraph. Here, the properties refer to some node-edge counts terms around the target node/edge.

4.2.4 Graph Contrastive Learning (GCL)

The pretraining strategies of GCL can be broadly categorized into two types: *Deep InfoMax* and *Instance Discrimination*. The former performs cross-scale contrastive learning, while the latter contrasts between the instances at the same scale.

Deep InfoMax (DIM) Deep InfoMax is originally proposed for images, which trains the model by maximizing the mutual information between an image representation and its local regions. For graphs, initially, DGI [Velickovic *et al.*, 2019] and InfoGraph [Sun, 2020] are proposed to obtain expressive representations for graphs or nodes via maximizing the mutual information between graph- and substructure-level representations. Similarly, GMI [Peng *et al.*, 2020] adopts two discriminators to directly measure mutual information between input and representations of both nodes and edges. Besides, MV-GRL [Hassani and Khasahmadi, 2020] performs node diffusion to generate an augmented view and then maximizes the mutual information between original and augmented views. More recently, EGI [Zhu *et al.*, 2021b] maximizes the mutual information between the ego-graphs to obtain a transferable GNN.

Instance Discrimination (IND) IND is one of the most popular pretraining strategies which embeds aug-

Table 2: List of representative PGMs. Here KG of KCL is Chemical Element Knowledge Graph.

PGMs	Input	Architecture	Pretraining Task	Pretraining Database	# Params.
Hu* *et al.* [2020a]	Graph	5-layer GIN	GCP + MCM	ZINC15 (2M) + ChEMBL (456K)	$\sim$ 2M
Graph-BERT [Zhang *et al.*, 2020]	Graph	Graph Transformer [Zhang *et al.*, 2020]	GAEs	Cora + CiteSeer + PubMed	N/A
GraphCL [You *et al.*, 2020]	Graph	5-layer GIN	IND	ZINC15 (2M) + ChEMBL (456K)	$\sim$ 2M
GPT-GNN [Hu *et al.*, 2020b]	Graph	HGT [Hu *et al.*, 2020c]	GAM	OAG + Amazon	N/A
GCC [Qiu *et al.*, 2020]	Graph	5-layer GIN	IND	Academia + DBLP + IMDB + Facebook + LiveJournal	<1M
JOAO [You *et al.*, 2021]	Graph	5-layer GIN	IND	ZINC15 (2M) + ChEMBL (456K)	$\sim$ 2M
AD-GCL [Suresh *et al.*, 2021]	Graph	5-layer GIN	IND	ZINC15 (2M) + ChEMBL (456K)	$\sim$ 2M
GraphLog [Xu *et al.*, 2021]	Graph	5-layer GIN	IND	ZINC15 (2M) + ChEMBL (456K)	$\sim$ 2M
GROVER [Rong *et al.*, 2020]	Graph	GTransformer [Rong *et al.*, 2020]	GCP + MCM	ZINC + ChEMBL (10M)	48M$\sim$100M
MGSSL [Zhang *et al.*, 2021b]	Graph	5-layer GIN	MCM + GAM	ZINC15 (250K)	$\sim$ 2M
CPT-HG [Jiang *et al.*, 2021]	Graph	HGT [Hu *et al.*, 2020c]	IND	DBLP + YELP + Aminer	N/A
PGM [Li *et al.*, 2021b]	Graph	MolGNet [Li *et al.*, 2021b]	RCD + MCM	ZINC + ChEMBL (11M)	53M
LP-Info [You *et al.*, 2022]	Graph	5-layer GIN	IND	ZINC15 (2M) + ChEMBL (456K)	$\sim$ 2M
SimGRACE [Xia *et al.*, 2022a]	Graph	5-layer GIN	IND	ZINC15 (2M) + ChEMBL (456K)	$\sim$ 2M
MolCLR [Wang *et al.*, 2021d]	Graph + SMILES	GCN + GIN	IND	PubChem (10M)	N/A
DMP [Zhu *et al.*, 2021a]	Graph + SMILES	DeeperGCN + Transformer	MCM + IND	PubChem (110M)	104.1 M
ChemRL-GEM [Fang *et al.*, 2021]	Graph + Geometry	GeoGNN [Fang *et al.*, 2021]	MCM+GCP	ZINC15 (20M)	N/A
KCL [Fang *et al.*, 2022]	Graph + KG	GCN + KMPNN [Fang *et al.*, 2022]	IND	ZINC15 (250K)	<1M
3D Infomax [Stärk *et al.*, 2021]	2D and 3D graph	PNA [Corso *et al.*, 2020]	IND	QM9(50K) + GEOM-drugs(140K) + QMugs(620K)	N/A
GraphMVP [Liu *et al.*, 2022]	2D and 3D graph	GIN + SchNet [Schütt *et al.*, 2017]	IND + GAEs	GEOM (50k)	$\sim$ 2M

mented versions of the same sample close to each other (positive samples) and pushes the embeddings of other samples (negatives) apart. For node-level representations, GRACE [Zhu *et al.*, 2020] and its variants [Zhu *et al.*, 2021d; Jin *et al.*, 2021; Xia *et al.*, 2021b] maximize the agreement of node embeddings across two corrupted views of the graph. For graph-level pretraining, GraphCL [You *et al.*, 2020] and its variants [You *et al.*, 2021; Sun *et al.*, 2021; Suresh *et al.*, 2021] propose advanced graph augmentation strategies to synthesize positive samples. Besides, some works such as BGRL [T. *et al.*, 2021], CCA-SSG [Zhang *et al.*, 2021a], LP-Info [You *et al.*, 2022], and SimGRACE [Xia *et al.*, 2022a] try to simplify graph contrastive learning via discarding the negatives, mutual information estimator or even data augmentations respectively. We develop an open-source graph contrastive learning (GCL) library[3] for PyTorch [Zhu *et al.*, 2021c].

4.2.5 Replaced Component Detection (RCD)

To capture the global information of graphs, RCD is proposed as a pretraining strategy on a random permutation of input graphs. For example, PHD [Li *et al.*, 2021a] first decomposes each molecular graph in the database into two half-graphs and replace one of them with a half-graph from the other half randomly. The GNN encoder is pretrained to detect whether two half-graphs in reorganized graphs are homologous pairs.

4.3 Extensions

4.3.1 Knowledge-Enriched Pretraining

PGMs usually learn universal graph representations from general-purpose graphs databases. However, they often lack domain-specific knowledge. To enhance their performance, several recent works try to inject external knowledge during pretaining. For example, GraphCL [You *et al.*, 2020] first pointed out that edge perturbation is conceptually incompatible with domain knowledge and empirically unhelpful for downstream performance for chemical compounds. Hence, they avoid adopting edge perturbation to augment molecular graphs. To explicitly incorporate the domain knowledge into pretraining, MoCL [Sun *et al.*, 2021] proposed a new augmentation operator called substructure substitution, in which a valid substructure in a molecule is replaced by a bioisostere [Meanwell, 2011] which produces a new molecule with similar physical or chemical properties as the original one. They compile 230 substitution rules in total. Recently, to capture the correlations between atoms that have common attributes but are not directly connected by bonds, KCL [Fang *et al.*, 2022] constructs a Chemical Element Knowledge Graph (KG) to summarize microscopic associations between elements and proposes a novel Knowledge-enhanced Contrastive Learning framework for molecular representation learning. Considering that 3D geometric information of molecule also plays a vital role in predicting molecular functionalities, 3DInfoMax [Stärk *et al.*, 2021] proposes pretraining a model to reason about the geometry of molecules given only their 2D molecular graphs while GraphMVP [Liu *et al.*, 2022] performs self-supervised pretraining via maximizing the correspondence and consistency between 2D topological structures and 3D geometric views.

4.3.2 Learning to Pretrain

There exists a gap between objectives of pretraining and finetuning steps, which will significantly hurt the generalization ability of PGMs. To narrow this gap, L2P-GNN [Lu *et al.*, 2021] simulates the finetuning via creating new tasks during pretraining. This setup enables PGMs to adapt to new tasks quickly and lead to better generalization on downstream tasks.

5 Tuning Strategies

As shown in Table 2, we list representative PGMs with their input, encoder architectures, pretraining tasks, pretraining database, and number of parameters. Although PGMs can capture abundant knowledge useful for downstream tasks, the process of vanilla finetuning suffers two harmful issues: *Overfitting* and *Catastrophic Forgetting*. Specifically, Xia et al. [Xia *et al.*, 2022b] observe that PGMs are prone to overfitting on insufficient labeled data for downstream tasks due to their high complexity. In particular, unlike image or text data,

[3] https://github.com/GraphCL/PyGCL

getting labels for biochemical graph data often requires laborious wet-lab experiments. To enrich the labeled molecular graphs, they propose to augment molecular graph data with chemical enantiomers and homologies, which share similar properties with original molecules. Besides, they introduce a new regularization built on dropout which encourages the output of PGMs not to change much when injecting a small perturbation and thus effectively controls PGMs' capacity. Also, catastrophic forgetting often happens when adapting PGMs to downstream tasks. In other words, PGMs often forget their learned general knowledge when finetuning. To alleviate this issue, Han et al. [Han *et al.*, 2021] utilize meta learning to adaptively select and combine various pretraining tasks with the target task in finetuning stage to achieve better adaptation. This strategy preserves sufficient knowledge captured by pretraining tasks while improving the effectiveness of transfer learning on GNNs. However, it takes the pretraining tasks as prerequisite, which impedes the usage of their methods in practice where the pretraining tasks are often unknown.

6 Applications

6.1 Social Recommendation

Owing to the outstanding performance in graph data learning, GNNs have been widely applied to social recommendation [Wang *et al.*, 2021b; Wu *et al.*, 2019]. Although remarkable progress has been made, they are still fraught with issues. For example, cold-start problem impedes their wider applications because GNNs fail to learn high-quality embeddings for the cold-start users/items with sparse interactions. To alleviate this critical issue, Hao et al. [Hao *et al.*, 2021] propose to pretrain the GNN model via predicting the ground-truth embeddings of users/item. In this way, they can enhance the embeddings of the cold-start users or items with the PGMs. Besides, PMGT [Liu *et al.*, 2021b] adopts graph reconstruction and MCM as the pretraining strategies to make full use of the side information of items for accurate recommendation. S^2-MHCN [Yu *et al.*, 2021] and DHCN [Xia *et al.*, 2021c] utilize DIM and IND as the pretraining strategies for social- and session-based recommendation respectively.

6.2 Drug Discovery

A molecule is naturally treated as a graph, where nodes refer to atoms and edges correspond to chemical bonds. Recently, the advancements in graph pretraining provide opportunities to expedite drug discovery and development pipeline. For example, the oral bioavailability of a drug is related to many properties, such as solubility in gastrointestinal tract, intestinal membrane permeability and intestinal/hepatic first-pass metabolism [Hou *et al.*, 2007]. However, it is often time-consuming and unsafe to conduct such experiments on human bodies. *Molecular property prediction* with PGMs serves as an alternative for such experiments. PGMs can be directly applied as a drug encoder to obtain expressive representations [Wang *et al.*, 2021d; Rong *et al.*, 2020]. Besides, *Drug-Drug Interaction (DDI) prediction* is also of vital importance in drug discovery because DDIs may lead to adverse drug reactions, which will damage people's health or even cause death. DDI prediction tasks can be regarded as a task that classifies the influence of combining drugs into three categories: synergistic, additive, and antagonistic. Works on molecular graph pretraining, such as MPG [Li *et al.*, 2021b] and WordReg & MolAug [Xia *et al.*, 2022b], have applied DDI prediction as a downstream task to reveal the effectiveness of the PGMs. Also, *Drug-Target Interaction (DTI) prediction* is another crucial step in drug discovery and repositioning as it reduces experimental validation costs if done right. Thus, developing methods to predict DTI has become a competitive research niche. In DTI, PGMs can be directly applied as a drug encoder and the well-pretrained model weights can be regarded as the initial weights of drug encoder. The drug encoder and target encoder are then trained with the DTI prediction task. Related works including MPG and WordReg & MolAug have followed this idea for DTI prediction. For *molecule generation*, MGM [Mahmood *et al.*, 2021] introduces a masked graph model, which learns a distribution over graphs by capturing conditional distributions over unobserved nodes (atoms) and edges (bonds) given observed ones.

7 Conclusions and Future Outlooks

This paper is the first survey paper focusing on pretraining on graphs, which is one of the most popular research trend in GNNs community. Due to the appealing effectiveness, PGMs show a promising potential in a range of applications.

Despite the fruitful progress of PGMs in the past, the development of pretraining on graphs still remains widely open. In this section, we suggest several promising research directions for the future.

7.1 More Effective and Efficient Knowledge Transfer

Currently, most researches focus on designing pretraining strategies. However, how to leverage these PGMs more effective and effectively is still under-explored compared to vivid development in pretrained language models. Finetuning is a dominant technique to adapt the knowledge to various downstream tasks, but there are several non-negligible deficiencies to be solved. The first one is poor generalization because of the high complexity of PGMs and scrace labeled data for downstream tasks. The second issue is parameter inefficiency. The finetuned parameters vary across both datasets and tasks, which are often huge in scale and thus being inconvenient in special scenarios such as low-capacity devices. Furthermore, there are some promising alternatives to mine the knowledge from PGMs. For example, PGMs can extract expressive representations as adopted in graph self-supervised learning. Distilling the knowledge from PGMs as adopted in NLP is also another promising direction [Yang *et al.*, 2020].

7.2 Improving Interpretability of PGMs

A major limitation of PGMs is that they are not amenable to interpretability. Unlike CNNs for images, interpreting PGMs is more difficult due to the complexities of both the Transformer-style architecture and non-Euclidean property of graph data. However, for safety-critical scenarios like molecular toxicity prediction, it is of vital importance for the PGMs to explain the reason why a molecule is non-toxic. Also, interpretability can accelerate scientific findings such as identifying biomarks. Overall, as a key component in graph-related applications, the interpretability of PGMs remain to be explored further in many respects, which helps us understand how PGMs work and provides a guide for better usage.

7.3 Broader Scope of Applications

PGMs can serve as generic encoders for multiple graph-related applications. However, for a specific downstream task, it is still under-explored which pretraining strategy is more suitable. Besides, PGMs have been applied in various sub-tasks in drug discovery such as molecular property prediction, DDI, DTI and molecule generation. However, it remains to be explored how PGMs can benefit other tasks of drug discovery such as chemical reaction prediction, retrosynthesis, molecule design and optimization. Also, for macromolecules such as proteins, recent works demonstrate that GNNs can help learn expressive representations for them [Xia and Ku, 2021]. More endeavors are still expected to study whether PGMs are conducive to protein representation learning.

References

[Bajusz *et al.*, 2015] D. Bajusz, A. Rácz, and K. Héberger. Why is tanimoto index an appropriate choice for fingerprint-based similarity calculations? *Journal of Cheminformatics*, 2015.

[Brown *et al.*, 2020] T. Brown, B. Mann, et al. Language models are few-shot learners. *NeurIPS*, 2020.

[Cho *et al.*, 2014] K. Cho, B. van Merrienboer, and others. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *EMNLP*, 2014.

[Corso *et al.*, 2020] G. Corso, L. Cavalleri, et al. Principal neighbourhood aggregation for graph nets. *NeurIPS*, 2020.

[Devlin *et al.*, 2019] J. Devlin, M. Chang, and others. Bert: Pre-training of deep bidirectional transformers for language understanding. *NACCL*, 2019.

[Fang *et al.*, 2021] X. Fang, L. Liu, and others. Geometry-enhanced molecular representation learning for property prediction. *Nature Machine Intelligence*, 2021.

[Fang *et al.*, 2022] Y. Fang, Q. Zhang, and others. Molecular contrastive learning with chemical element knowledge graph. *AAAI*, 2022.

[Grover and Leskovec, 2016] A. Grover and J. Leskovec. node2vec: Scalable feature learning for networks. In *KDD*, 2016.

[Han *et al.*, 2021] X. Han, Z. Huang, and others. Adaptive transfer learning on graph neural networks. In *KDD*, 2021.

[Hao *et al.*, 2019] Yaru Hao, Li Dong, and others. Visualizing and understanding the effectiveness of bert. *EMNLP/IJCNLP*, 2019.

[Hao *et al.*, 2021] B. Hao, J. Zhang, and others. Pre-training graph neural networks for cold-start users and items representation. In *WSDM*, 2021.

[Hasanzadeh *et al.*, 2019] A. Hasanzadeh, E. Hajiramezanali, and others. Semi-implicit graph variational auto-encoders. *NeurIPS*, 2019.

[Hassani and Khasahmadi, 2020] K. Hassani and A. Khasahmadi. Contrastive multi-view representation learning on graphs. In *ICML*, 2020.

[Hinton and Salakhutdinov, 2006] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 2006.

[Hou *et al.*, 2007] T. Hou, J. Wang, W. Zhang, and X. Xu. Adme evaluation in drug discovery. 6. can oral bioavailability in humans be effectively predicted by simple molecular property-based rules? *Journal of Chemical Information and Modeling*, 2007.

[Hu *et al.*, 2019] Z. Hu, C. Fan, and et al. Pre-training graph neural networks for generic structural feature extraction. *arXiv preprint arXiv:1905.13728*, 2019.

[Hu* *et al.*, 2020a] Weihua Hu*, Bowen Liu*, and others. Strategies for pre-training graph neural networks. In *ICLR*, 2020.

[Hu *et al.*, 2020b] Z. Hu, Y. Dong, and others. Gpt-gnn: Generative pre-training of graph neural networks. In *KDD*, 2020.

[Hu *et al.*, 2020c] Z. Hu, Y. Dong, K. Wang, and Y. Sun. Heterogeneous graph transformer. In *WWW*, 2020.

[Jiang *et al.*, 2021] Xunqiang Jiang, Yuanfu Lu, and others. Contrastive pre-training of gnns on heterogeneous graphs. In *CIKM*, 2021.

[Jin *et al.*, 2021] M. Jin, Y. Zheng, and others. Multi-scale contrastive siamese networks for self-supervised graph representation learning. In *IJCAI*, 2021.

[Kipf and Welling, 2016] Thomas N Kipf and Max Welling. Variational graph auto-encoders. *arXiv:1611.07308*, 2016.

[Li *et al.*, 2021a] P. Li, J. Wang, and others. Pairwise half-graph discrimination: A simple graph-level self-supervised strategy for pre-training graph neural networks. In *IJCAI*, 2021.

[Li *et al.*, 2021b] Pengyong Li, Jun Wang, and others. An effective self-supervised framework for learning expressive molecular global representations to drug discovery. *BIB*, 2021.

[Liu *et al.*, 2019] S. Liu, M. Demirel, and Y. Liang. N-gram graph: Simple unsupervised representation for graphs, with applications to molecules. *NeurIPS*, 2019.

[Liu *et al.*, 2021a] Y. Liu, S. Pan, and others. Graph self-supervised learning: A survey. *arXiv:2103.00111*, 2021.

[Liu *et al.*, 2021b] Y. Liu, S. Yang, and others. Pre-training graph transformer with multimodal side information for recommendation. In *ACM MM*, 2021.

[Liu *et al.*, 2022] Shengchao Liu, Hanchen Wang, and others. Pre-training molecular graph representation with 3d geometry. In *ICLR*, 2022.

[Lu *et al.*, 2021] Y. Lu, X. Jiang, Y. Fang, and C. Shi. Learning to pre-train graph neural networks. In *AAAI*, 2021.

[Mahmood *et al.*, 2021] O. Mahmood, E. Mansimov, and others. Masked graph modeling for molecule generation. *Nature communications*, 2021.

[Meanwell, 2011] Nicholas A Meanwell. Synopsis of some recent tactical application of bioisosteres in drug design. *Journal of medicinal chemistry*, 2011.

[Mikolov *et al.*, 2013] Tomas Mikolov, Ilya Sutskever, and others. Distributed representations of words and phrases and their compositionality. *NeurIPS*, 2013.

[Narayanan *et al.*, 2016] Annamalai Narayanan, Mahinthan Chandramohan, and others. subgraph2vec: Learning distributed representations of rooted sub-graphs from large graphs. *arXiv:1606.08928*, 2016.

[Pan *et al.*, 2018] S. Pan, R. Hu, and others. Adversarially regularized graph autoencoder for graph embedding. In *IJCAI*, 2018.

[Peng *et al.*, 2020] Z. Peng, W. Huang, and others. Graph representation learning via graphical mutual information maximization. In *WWW*, 2020.

[Perozzi *et al.*, 2014] B. Perozzi, R. Al-Rfou, and S. Skiena. Deepwalk: Online learning of social representations. In *KDD*, 2014.

[Qiu *et al.*, 2020] Jiezhong Qiu, Qibin Chen, and others. Gcc: Graph contrastive coding for graph neural network pre-training. In *KDD*, 2020.

[Rong *et al.*, 2020] Y. Rong, Y. Bian, and others. Self-supervised graph transformer on large-scale molecular data. *NeurIPS*, 2020.

[Schütt *et al.*, 2017] K. Schütt, P. Kindermans, et al. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. In *NIPS*, 2017.

[Stärk *et al.*, 2021] H. Stärk, D. Beaini, and others. 3d infomax improves gnns for molecular property prediction. *arXiv:2110.04126*, 2021.

[Sun *et al.*, 2021] Mengying Sun, Jing Xing, and others. Mocl: Contrastive learning on molecular graphs with multi-level domain knowledge. *KDD*, 2021.

[Sun, 2020] Infograph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization. In *ICLR*, 2020.

[Suresh *et al.*, 2021] Susheel Suresh, Pan Li, and others. Adversarial graph augmentation to improve graph contrastive learning. In *NeurIPS*, 2021.

[T. *et al.*, 2021] Shantanu T., Corentin T., and others. Bootstrapped representation learning on graphs. In *ICLR Workshop*, 2021.

[Tang *et al.*, 2015a] J. Tang, M. Qu, and Q. Mei. Pte: Predictive text embedding through large-scale heterogeneous text networks. In *KDD*, 2015.

[Tang *et al.*, 2015b] J. Tang, M. Qu, and others. Line: Large-scale information network embedding. In *WWW*, 2015.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, and others. Attention is all you need. *NIPS*, 2017.

[Velickovic *et al.*, 2019] P. Velickovic, W. Fedus, and others. Deep graph infomax. *ICLR*, 2019.

[Wang *et al.*, 2017] C. Wang, S. Pan, and others. Mgae: Marginalized graph autoencoder for graph clustering. In *CIKM*, 2017.

[Wang *et al.*, 2019] Xiao Wang, Houye Ji, and others. Heterogeneous graph attention network. In *WWW*, 2019.

[Wang *et al.*, 2021a] Hui Wang, Kun Zhou, and others. Curriculum pre-training heterogeneous subgraph transformer for top-n recommendation. *arXiv:2106.06722*, 2021.

[Wang *et al.*, 2021b] S. Wang, L. Hu, and others. Graph learning based recommender systems: A review. *IJCAI*, 2021.

[Wang *et al.*, 2021c] X. Wang, N. Liu, and others. Self-supervised heterogeneous graph neural network with co-contrastive learning. In *KDD*, 2021.

[Wang *et al.*, 2021d] Y. Wang, J. Wang, and others. Molclr: Molecular contrastive learning of representations via graph neural networks. *ArXiv*, abs/2102.10056, 2021.

[Wu *et al.*, 2019] Shu Wu, Yuyuan Tang, Yanqiao Zhu, Liang Wang, Xing Xie, and Tieniu Tan. Session-based Recommendation with Graph Neural Networks. In *AAAI*, 2019.

[Wu *et al.*, 2020] Z. Wu, S. Pan, and others. A comprehensive survey on graph neural networks. *TNNLS*, 2020.

[Xia and Ku, 2021] T. Xia and W. Ku. Geometric graph representation learning on protein structure prediction. In *KDD*, 2021.

[Xia *et al.*, 2021a] J. Xia, H. Lin, and others. Towards robust graph neural networks against label noise. *Openreview*, 2021.

[Xia *et al.*, 2021b] J. Xia, L. Wu, J. Chen, G. Wang, and S. Li. Debiased graph contrastive learning. *arXiv:2110.02027*, 2021.

[Xia *et al.*, 2021c] X. Xia, H. Yin, and others. Self-supervised hypergraph convolutional networks for session-based recommendation. In *AAAI*, 2021.

[Xia *et al.*, 2022a] J. Xia, L. Wu, , J. Chen, B. Hu, and S. Li. SimGRACE: A Simple Framework for Graph Contrastive Learning without Data Augmentation. In *WWW*, 2022.

[Xia *et al.*, 2022b] J. Xia, J. Zheng, C. Tan, G. Wang, and S. Li. Towards effective and generalizable fine-tuning for pre-trained molecular graph models. *bioRxiv*, 2022.

[Xie *et al.*, 2021] Yaochen Xie, Zhao Xu, and others. Self-supervised learning of graph neural networks: A unified review. *arXiv:2102.10757*, 2021.

[Xu *et al.*, 2019] Keyulu Xu, Weihua Hu, and others. How powerful are graph neural networks? In *ICLR*, 2019.

[Xu *et al.*, 2021] M. Xu, H. Wang, and others. Self-supervised graph-level representation learning with local and global structure. In *ICML*, 2021.

[Yang *et al.*, 2020] Z. Yang, Y. Cui, et al. TextBrewer: An Open-Source Knowledge Distillation Toolkit for Natural Language Processing. In *ACL: System Demonstrations*, 2020.

[You *et al.*, 2020] Y. You, T. Chen, and others. Graph contrastive learning with augmentations. In *NeurIPS*, 2020.

[You *et al.*, 2021] Y. You, T. Chen, and others. Graph contrastive learning automated. *ICML*, 2021.

[You *et al.*, 2022] Y. You, T. Chen, Z. Wang, and Y. Shen. Bringing your own view: Graph contrastive learning without prefabricated data augmentations. In *WSDM*, 2022.

[Yu *et al.*, 2021] J. Yu, H. Yin, and others. Self-supervised multi-channel hypergraph convolutional network for social recommendation. In *WWW*, 2021.

[Zhang *et al.*, 2020] J. Zhang, H. Zhang, C. Xia, and L. Sun. Graph-bert: Only attention is needed for learning graph representations. *arXiv preprint arXiv:2001.05140*, 2020.

[Zhang *et al.*, 2021a] H. Zhang, Q. Wu, and others. From canonical correlation analysis to self-supervised graph neural networks. In *NeurIPS*, 2021.

[Zhang *et al.*, 2021b] Zaixi Zhang, Qi Liu, and others. Motif-based graph self-supervised learning for molecular property prediction. *NeurIPS*, 2021.

[Zhu *et al.*, 2020] Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. Deep Graph Contrastive Representation Learning. In *ICML Workshop*, 2020.

[Zhu *et al.*, 2021a] J. Zhu, Y. Xia, and others. Dual-view molecule pre-training. *ArXiv*, abs/2106.10234, 2021.

[Zhu *et al.*, 2021b] Q. Zhu, C. Yang, and others. Transfer learning of graph neural networks with ego-graph information maximization. *NeurIPS*, 2021.

[Zhu *et al.*, 2021c] Yanqiao Zhu, Yichen Xu, Qiang Liu, and Shu Wu. An empirical study of graph contrastive learning. In *NeurIPS Datasets and Benchmarks Track*, 2021.

[Zhu *et al.*, 2021d] Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. Graph Contrastive Learning with Adaptive Augmentation. In *WWW*, 2021.

[Zhu *et al.*, 2022] Yanqiao Zhu, Yichen Xu, Hejie Cui, Carl Yang, Qiang Liu, and Shu Wu. Structure-enhanced heterogeneous graph contrastive learning. In *SDM*, 2022.