Empowering Dual-Level Graph Self-Supervised Pretraining with Motif Discovery

Pengwei Yan[1,2], Kaisong Song[2, 3], Zhuoren Jiang[1*], Yangyang Kang[2*], Tianqianjin Lin[1,2], Changlong Sun[2], Xiaozhong Liu[4]

[1]Department of Information Resources Management, Zhejiang University, Hangzhou, 310058, China
[2]Alibaba Group, Hangzhou, 311121, China
[3]Northeastern University, Shenyang, 110819, China
[4]Computer Science Department, Worcester Polytechnic Institute, Worcester, 01609-2280, MA, USA
{12122093, jiangzhuoren, lintqj}@zju.edu.cn, {kaisong.sks, yangyang.kangyy}@alibaba-inc.com,
changlong.scl@taobao.com, xliu14@wpi.edu

Abstract

While self-supervised graph pretraining techniques have shown promising results in various domains, their application still experiences challenges of limited topology learning, human knowledge dependency, and incompetent multi-level interactions. To address these issues, we propose a novel solution, Dual-level Graph self-supervised Pretraining with Motif discovery (DGPM), which introduces a unique dual-level pretraining structure that orchestrates node-level and subgraph-level pretext tasks. Unlike prior approaches, DGPM autonomously uncovers significant graph motifs through an edge pooling module, aligning learned motif similarities with graph kernel-based similarities. A cross-matching task enables sophisticated node-motif interactions and novel representation learning. Extensive experiments on 15 datasets validate DGPM's effectiveness and generalizability, outperforming state-of-the-art methods in unsupervised representation learning and transfer learning settings. The autonomously discovered motifs demonstrate the potential of DGPM to enhance robustness and interpretability.

Introduction

As in the fields of natural language processing and computer vision (Devlin et al. 2018; He et al. 2020), pretraining with self-supervised learning (SSL) holds similar significance for models constructed on graph data, which aims to learn the informative graph representations from unlabeled data (Hu et al. 2019). Such approaches can effectively address the labeled data scarcity and improve the model's generalizability in graph domain (Xie et al. 2022). While numerous research endeavors (Veličković et al. 2018; You et al. 2020; Hou et al. 2022) have already yielded successful results, applying self-supervised pretraining techniques to graph data still faces the following challenges.

Limited Topology Learning. Defining appropriate pretext training tasks stands as a pivotal objective for graph pretraining models. However, most existing graph pretraining endeavors often restrict themselves to simple prediction tasks rooted in close neighborhood structures. For instance, utilizing k-hop neighborhood information (Rong et al. 2020) to predict node contextual property, and employing subgraphs to forecast the neighboring graph structures (Zhang et al. 2021b). However, these learning tasks might be confined to predicting lower-order structural attributes, potentially hampering the model's ability to comprehend graph topology (Chen et al. 2023) and also neglecting higher-order structural interpretability (Milo et al. 2002).

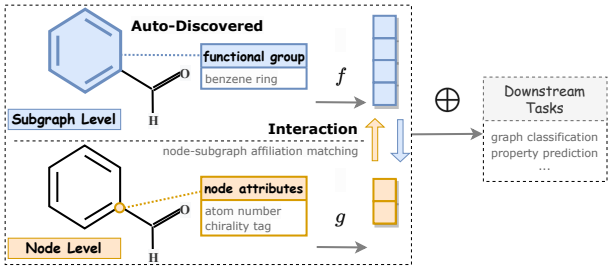


Figure 1: Toy example showing interactive dual-level graph pretraining.

Human Knowledge Dependency. Graph motif (Milo et al. 2002), the significant high-order graph pattern with semantic meaning, have shown their potential to enhance graph models (Chen et al. 2023; Rong et al. 2020; Zhang et al. 2021b). Despite their beneficial impact, existing motif-related methods (Rong et al. 2020; Zhang et al. 2021b) often rely on domain knowledge and manual motif pre-definition, limiting generalizability across domains for motif-driven self-supervised learning.

Incompetent Multi-Level Interactions. For effective pretraining, graph self-supervised learning models should capture knowledge from node attributes and structural topology (Ma and Tang 2021). But existing approaches often use step-by-step learning (Hu et al. 2019) or simple aggregation learning (Zhang et al. 2021b), missing intrinsic multi-level information interaction.

To address the above challenges, we propose Dual-level Graph self-supervised Pretraining with Motif discovery (**DGPM**). As shown in Figure 1, DGPM introduces a dual-level pretraining architecture. In addition to a node-level pretraining task, DGPM incorporates a novel and challenging task for subgraph-level pretraining. This task involves the autonomous discovery of motifs through an edge pooling

*Corresponding authors.

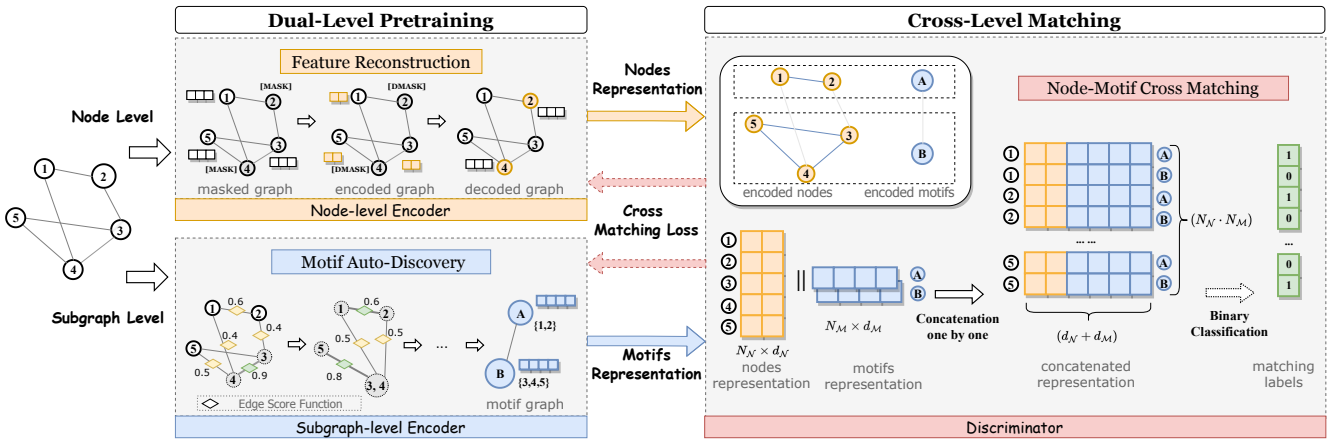


Figure 2: The DGPM framework comprises two main components: Dual-level pretraining and Cross-level matching. The Dual-level pretraining encompasses a node-level feature reconstruction task and a subgraph-level motif auto-discovery task.

module. The training objective is aligning two kinds of similarity: one derived from automatically learned motifs and the other is graph similarity computed by the Wasserstein Weisfeiler Lehman (WWL) graph kernel. Furthermore, a cross-matching task targeting node-motif affiliations is established to connect the node and subgraph-level pretraining.

Extensive experiments on 15 datasets, including validation of unsupervised representation learning and transfer learning, demonstrate the effectiveness and generalizability of DGPM. In unsupervised representation learning for graph classification, DPGM outperforms all SOTA methods and shows comparable performance with supervised ones. Besides, ablation studies, sensitive analyses, and auto-discovered motif analyses provide further validation of the reliability and robustness of our proposed framework.

The major contributions of this research can be summarized as follows:

1. We propose a novel dual-level graph pretraining architecture, DGPM [1] , to address the challenges of limited topology learning, human knowledge dependency, and incompetent multi-level interactions for graph self-supervised pretraining. To the best of our knowledge, this is the first graph pretraining framework that utilizes a motif auto-discovery mechanism to leverage subgraph structure information in self-supervised learning.

2. A novel and challenging pretext task for subgraph-level pretraining is introduced to comprehensively learn the vital graph structures. In which, the motif auto-discovery module can autonomously uncover the crucial patterns in graph structure to enhance the generalization of graph pretraining model and improve the interpretability by providing the visualized motifs in the pretraining process.

3. Through extensive experiments on 15 benchmark datasets, we demonstrate the superiority and generalizability of DGPM by comparing it with many strong baselines. The learned motif analysis further proves the potential of DGPM to enhance interpretability.

[1]The code is available at https://github.com/RocccYan/DGPM.

Methodology

Unlike prior research, the proposed DGPM aims to learn both node-level attributes and subgraph-level structures of graphs. The framework of DGPM is illustrated in Figure 2.

Problem Statement

Notations. Let $G = (V, E)$ denote a graph, where $V = \{v_1, ..., v_{N_\mathcal{N}}\}$ represents the set of nodes, and $E \subseteq V \times V$ represents the set of edges. G is associated with a feature matrix $\mathbf{X} \in \mathbb{R}^{N_\mathcal{N} \times d}$, and an adjacency matrix $\mathbf{A} \in \mathbb{R}^{N_\mathcal{N} \times N_\mathcal{N}}$ where $\mathbf{A}_{ij} = 1$ iff $(v_i, v_j) \in \mathcal{E}$ and $\mathbf{A}_{ij} = 0$ otherwise. Following (Milo et al. 2002), we define motifs $\mathcal{M} = (V_\mathcal{S}, E_\mathcal{S})$ as high frequency subgraphs of G, with $V_\mathcal{S} \subseteq V$ and $E_\mathcal{S} \subseteq E$.

Task: Dual-Level Graph Self-Supervised Pretraining In this study, we hypothesize that graph information should be hierarchically structured, involving both nodes and motifs, and motifs serve as important functional subgraph patterns (such as functional groups in chemical molecules). Given a graph G with $\mathbf{X}$ and $\mathbf{A}$, we aim to learn both an node level encoder $f_\mathcal{N}(\cdot)$ that produces node representations $\mathbf{H}^\mathcal{N} = f_\mathcal{N}(\mathbf{X}, \mathbf{A}) \in \mathbb{R}^{N_\mathcal{N} \times d_\mathcal{N}}$ and an subgraph level encoder $f_\mathcal{M}(\cdot)$ that generates motif representations $\mathbf{H}^\mathcal{M} = f_\mathcal{M}(\mathbf{X}, \mathbf{A}) \in \mathbb{R}^{N_\mathcal{M} \times d_\mathcal{M}}$. Specifically, our goal is to autonomously discover motifs and then interactively learn node and motif representations, amalgamating both for diverse downstream tasks.

The Design of DGPM

Dual-Level Pretrain

Node Feature Reconstruction Task For the node-level learning component, an encoder is designed primarily to capture the local node information. To this end, we employ a graph auto-encoder inspired by (Hou et al. 2022), which prioritizes node feature reconstruction. The encoder transforms masked input data into a representation, and the decoder reverses this process to reconstruct the input, guided by the node feature reconstruction criterion.

Given $f_{\mathcal{N}}$ and $f'_{\mathcal{N}}$ as the encoder and decoder for node feature reconstruction, $\mathbf{H}_{\mathcal{N}}$ denoting the node representations encoded by $f_{\mathcal{N}}$, the goal of node level learning task is to reconstruct the input as

$$\widetilde{\mathbf{H}}^{\mathcal{N}} = f_{\mathcal{N}}(\mathbf{A}, \widetilde{\mathbf{X}}) \tag{1}$$

$$\mathbf{Z} = f'_{\mathcal{N}}(\mathbf{A}, \widetilde{\mathbf{H}}^{\mathcal{N}}) \tag{2}$$

where $\widetilde{\mathbf{X}}$ and $\mathbf{Z}$ denote the masked and the reconstructed node features respectively. Thus the reconstruction loss is as

$$\mathcal{L}^{\mathcal{N}}_{rec} = \frac{1}{|\widetilde{V}|} \sum_{v_i \in \widetilde{V}} (1 - \frac{x_i^\top z_i}{||x_i|| \cdot ||z_i||}). \tag{3}$$

Subgraph Level Motif Auto-Discovery Task To effectively learn the subgraph structure information without human intervention, we propose a module for autonomous motif discovery, eliminating the need for domain-specific knowledge in motif predefinition. Taking inspiration from (Oliver et al. 2022), we customize EdgePool layers to merge nodes into subgraphs and enforce alignment between the cosine similarity of the merged subgraphs and the similarity generated by the graph kernel.

EdgePool Layers. For an input graph, EdgePool layers selectively collapse pairs of nodes connected by an edge into a single node to obtain a coarsened graph. To automatically discover essential structures (motif), we train an aggregator and edge score function during this process. The aggregator combines connected node features, with edge features (if exist). The edge score function generates a score per edge to decide node-merge suitability. Given $e_{u,v}$ as the edge to be merged, $f_{\mathcal{M}}(\cdot)$ as the aggregator, s as the edge score, the aggregated representations and edge score are as:

$$\begin{aligned} h^{\mathcal{M}}_{u,v} &= f_{\mathcal{M}}(u, v, e_{u,v}) \\ &= [x_u \mathbf{W_n}, x_v \mathbf{W_n}, x_{e_{u,v}} \mathbf{W_e}]\mathbf{W} \end{aligned} \tag{4}$$

$$s_{u,v} = \sigma(h^{\mathcal{M}}_{u,v}) \tag{5}$$

where $\mathbf{W_n}, \mathbf{W_e}$, as transformation weight matrix and $\mathbf{W}$ are aggregation parameters, x are corresponding node features and edge features, and $\sigma(\cdot)$ is the score function, a single-layer MLP is employed here. Utilizing the generated edge scores, edges are sorted in reverse order of scores and sequentially compared with a uniformly distributed probability p. Edges with scores exceeding p lead to the merging of the edge and connected nodes into a new node represented $h^{\mathcal{M}}_{u,v}$. As EdgePool layers stack, nodes in deep layers represent subgraphs derived from the input graph with informative structural features, thus completing the auto-discovery of the motif. Furthermore, as all pooling is edge-based, we can ensure the connectivity of the discovered motifs.

Graph Similarity Loss. With EdgePool layers, the input graph is pooled into a coarsened graph, whose nodes denote motifs from the original graph. Hence, encoded node representations serve as corresponding motif representations. From a motif (subgraph) perspective, we adopt graph similarity as the training objective. We employ the Wasserstein Weisfeiler Lehman (WWL) graph kernel(Togninalli et al. 2019) to measure motif similarity as ground truth, guiding EdgePool layer training. Graph kernels excel in addressing graph complexity and exhibit good predictive capabilities across diverse graph tasks (Shervashidze et al. 2011; Yanardag and Vishwanathan 2015). WWL graph kernel jointly models structural similarity and node feature agreement on graphs, effectively supervising graph topology properties. For node pairs within the coarsened graph, we compute cosine similarity with encoded representations, aiming to align it with WWL kernel-generated similarity. As shown in Fig 2, given motif pair (A, B) from motif graph $\mathcal{G}$,

$$\mathcal{L}^{\mathcal{M}}_{sim} = \sum_{\mathcal{G}} ||\Omega(h^{\mathcal{M}}_A, h^{\mathcal{M}}_B) - \mathbf{WWL}(\mathrm{S}\{A\}, \mathrm{S}\{B\})||^2_2. \tag{6}$$

where Ω refers to scaled cosine similarity and $\mathrm{S}\{\cdot\}$ as the corresponding motif of the node.

Cross-Level Matching Task Following training in node reconstruction and motif discovery, we obtain a node-level encoder and a subgraph-level encoder for corresponding representations. To exploit the inherent inter-relationship between nodes and motifs, we establish a node-motif matching task connecting node-level and subgraph-level training.

As shown in Figure 2, the learned motif comprises distinct nodes, serving as the learning objective for the node-motif relationship. With permutation and concatenation, we iteratively combine node and motif representations and train a discriminator to predict whether an affiliation exists. Given permutation $\mathcal{P}$, corresponding matching labels $\mathbf{y} = \{y_{1,1}, ..., y_{1,N^{\mathcal{M}}}, y_{2,1}, ..., y_{N^{\mathcal{N}},N^{\mathcal{M}}}\}$, and discriminator g, the matching loss is as

$$\begin{aligned} \mathcal{L}_{cross} = \frac{1}{|\mathcal{P}|} \sum_{(i,j)\in\mathcal{P}} -\Big(& y_{i,j} \cdot \log(g(h^{\mathcal{N}}_i, h^{\mathcal{M}}_j)) \\ & + (1 - y_{i,j}) \cdot \log(1 - g(h^{\mathcal{N}}_i, h^{\mathcal{M}}_j)) \Big) \end{aligned} \tag{7}$$

Different from prior joint learning or step-by-step learning approaches with simple loss stacking, the proposed cross-level matching learning task establishes an *interactive* connection between node-level encoder and subgraph-level encoder learning. In the training process, the node-level and subgraph-level representations can be iteratively enhanced based on the back-propagation of matching loss.

The overall training time complexity of DGPM is $\mathcal{O}(|V| \log(|V|))$, $|V|$ for the number of nodes.

Experiments

Performance Validation

To validate the performance of DGPM, we conducted experiments in two typical scenarios for downstream task applications: ***Unsupervised Representation Learning*** (direct utilization of trained representations for graph classification) and ***Transfer Learning*** (applying a pretrain-finetune approach for molecular property prediction). We followed the experimental setup employed in previous research work, such as data splits and evaluation metrics. Specifically, for unsupervised representation learning task, we adopted the

Category	Method	IMDB-B	IMDB-M	PROTEINS	COLLAB	MUTAG	REDDIT-B	NCI1
Supervised	GIN	75.1±5.1	52.3±2.8	76.2±2.8	80.2±1.9	89.4±5.6	92.4±2.5	82.7±1.7
	DiffPool	72.6±3.9	-	75.1±3.5	78.9±2.3	85.0±10.3	92.1±2.6	-
Unsupervised Graph Kernel	WL	72.30±3.44	46.95±0.46	72.92±0.56	78.9±1.9	80.72±3.00	68.82±0.41	80.31±0.46
	DGK	66.96±0.56	44.55±0.52	73.30±0.82	73.1±0.3	87.44±2.72	78.04±0.39	80.31±0.46
Self-Supervised	graph2vec	71.10±0.54	50.44±0.87	73.30±2.05	-	83.15±9.25	75.58±1.03	73.22±1.81
	Infograph	73.03±0.87	49.69±0.53	74.44±0.31	70.65±1.13	89.01±1.13	82.50±1.42	76.20±1.06
	GraphCL	71.14±0.44	48.58±0.67	74.39±0.45	71.36±1.15	86.80±1.34	87.53±0.84	77.87±0.41
	JOAO	70.21±3.08	49.20±0.77	74.55±0.41	69.50±0.36	87.35±1.02	85.29±1.35	78.07±0.47
	GCC	72.0	49.4	74.48±3.12	78.9	-	89.8	66.33±2.65
	MVGRL	74.20±0.70	51.20±0.50	71.50±0.30	76.01±1.20	89.70±1.10	84.50±0.60	-
	InfoGCL	75.10±0.90	51.40±0.80	-	80.00±1.30	91.20±1.30	-	80.20±0.60
	GraphMAE	75.52±0.66	51.63±0.52	75.30±0.39	80.32±0.46	88.19±1.26	88.01±0.19	80.40±0.30
	DGPM	**75.77±0.53**	**52.12±0.47**	**75.72±0.43**	**80.44±0.54**	**91.20±0.87**	**88.17±0.31**	**80.87±0.28**

Table 1: Experiment results in *unsupervised representation learning* for graph classification. We report accuracy (%) for all datasets. The reported results of baselines are from previous papers if available.

	BBBP	Tox21	ToxCast	SIDER	ClinTox	MUV	HIV	BACE	Avg.
No-pretrain	66.5±1.4	74.7±0.4	63.3±1.5	56.4±0.8	58.6±2.1	71.9±1.4	75.4±0.8	72.3±1.6	67.39
ContextPred	63.4±2.7	74.7±0.7	62.9±0.6	59.0±0.6	64.7±3.8	74.2±0.9	75.4±1.1	78.9±2.1	69.15
AttrMasking	63.5±2.7	**75.4±0.6**	63.2±0.5	59.1±0.7	70.7±2.1	73.1±2.2	75.2±1.0	78.7±1.7	69.86
Infomax	66.7±0.8	74.3±0.5	61.5±0.5	57.3±0.9	68.1±3.1	73.4±2.6	74.2±1.2	74.7±1.6	68.78
GrpahCL	67.8±0.8	72.7±0.9	60.4±0.6	58.6±0.8	74.2±2.1	68.9±3.2	77.1±0.9	74.1±0.7	69.23
JOAO	69.1±1.2	73.8±0.5	61.1±0.4	58.7±0.8	80.1±1.8	70.2±1.1	75.1±0.8	75.6±0.8	70.46
GraphLoG	70.4±0.8	74.9±0.5	61.8±0.7	59.5±1.1	75.7±2.6	74.8±1.1	74.9±0.9	81.0±1.1	71.63
GraphMAE	70.1±0.6	74.4±0.5	63.9±0.4	59.0±0.7	80.8±1.2	74.5±2.3	77.0±0.4	**81.4±0.9**	72.64
DGPM	**71.2±0.5**	75.3±0.4	**64.0±0.7**	**60.3±0.8**	**80.9±1.3**	**75.3±1.6**	**77.3±0.6**	81.1±0.7	**73.17**

Table 2: Experiment results in *transfer learning* on molecular property prediction benchmarks. The model is first pre-trained on ZINC15 and then finetuned on the following datasets. We report ROC-AUC scores (%).

experimental setup from (Zhang et al. 2021a; Hou et al. 2022); for transfer learning task, we followed the setup established in (Hu et al. 2019; You et al. 2020, 2021).

Datasets To validate *unsupervised representation learning*, we conducted experiments on 7 graph classification benchmarks (Hou et al. 2022) from four distinct domains: MUTAG, IMDB-B, IMDB-M, PROTEINS, COLLAB, REDDIT-B, and NCI1. Each of them is a collection of graphs where each graph is associated with a label. Specifically, node types serve as input features for MUTAG, PROTEINS, and NCI1 datasets, while node degrees are utilized in IMDB-B, IMDB-M, REDDIT-B, and COLLAB datasets.

To validate *transfer learning*, we conduct molecular property prediction experiments under a pretrain-finetune setting. 250k unlabeled molecules sampled from the ZINC15 (Sterling and Irwin 2015) are used for pretraining and 8 molecular benchmark datasets (Wu et al. 2018) are used for finetuning and testing: BBBP, Tox21, ToxCast, SIDER, ClinTox, MUV, HIV, and BACE. The downstream datasets are partitioned using scaffold-split to emulate real-world scenarios. The input node features include the atom number and chirality tag, while the edge features encompass the bond type and direction.

Baselines For *unsupervised representation learning* setting, we compared DGPM with two groups of strong unsupervised learning models. Graph kernel methods: Weisfeiler-Lehman sub-tree kernel (WL) (Shervashidze et al. 2011) and Deep Graph Kernel (DGK) (Yanardag and Vishwanathan 2015). SOTA self-supervised methods: graph2vec (Narayanan et al. 2017), Infograph (Sun et al. 2019), GraphCL (You et al. 2020), JOAO (You et al. 2021), GCC (Qiu et al. 2020), MVGRL (Hassani and Khasahmadi 2020), InfoGCL (Xu et al. 2021a), and GraphMAE (Hou et al. 2022). Additionally, we have included two supervised methods as references to evaluate graph classification performance: GIN (Xu et al. 2018) and DiffPool (Ying et al. 2018).

For *transfer learning* setting, we compared DGPM with 7 self-supervised methods: ContextPred, AttrMasking (Hu et al. 2019), Infomax (Sun et al. 2019), GraphCL (You et al. 2020), JOAO (You et al. 2021), GraphLoG (Xu et al. 2021b) and GraphMAE (Hou et al. 2022). Furthermore, we included

a no-pretrain model for comparison, in which DGPM is utilized for molecular property prediction without pretraining.

For each task, all baseline results were self-reported results in previous studies, under the same experimental setup.

Implementation Details Generally, we adopt a 5-layer GIN as encoder and a single-layer GIN as decoder for node-level pretraining module. The hidden dimension is set to 128 for both node and motif representations. The framework is trained using the AdamW optimizer for 100 epochs, with all implementations carried out using the PyTorch Geometric package. The learned node representations and motif representations are first pooled by mean-pooling readout function and then concatenated as dual-level representations.

For *unsupervised representation learning* setting, the generated representations are fed into a downstream LIBSVM (Chang and Lin 2011) classifier as graph features to predict the graph label. We report the mean 10-fold cross-validation accuracy with standard deviation after 5 runs. For *transfer learning* setting, we employ a 2-layer EdgePool as the encoder within the motif discovery module. Additionally, an MLP layer is incorporated to adapt the combined representations for molecular property prediction. We conduct experiments in 10 repetitions and provide the mean and standard deviation of ROC-AUC scores (%) as results.

Results For *unsupervised representation learning*, the results are shown in Table 1. DGPM outperforms all unsupervised baselines across all datasets, exhibiting an average improvement of 6.79% and a maximum improvement of 28.12%. DGPM also achieved comparable task performance with supervised learning on 3 datasets and even outperformed supervised learning on 4 benchmarks. The results show that dual-level pretraining is effective in learning informative representations and has the potential for unsupervised representation learning tasks.

For *transfer learning*, Table 2 shows that our model's performance on downstream tasks outperforms SOTA methods on 6 out of 8 tasks (with maximum 25% improvement), and achieves the best average performance (with average 4.2% improvement). These results further indicate the robust transferability of DGPM.

In summary, DGPM achieves remarkable performance in both unsupervised representation learning and transfer learning across 15 benchmarks. The consistent outcomes in these two task settings demonstrate DGPM's effectiveness and generalisability for a wide range of applications in various domains.

Ablation Study

Effect of Motif Auto-Discovery Task As shown in Table 3, by comparing task performance before and after removing the motif discovery module, we have the following observations. (1) Generally, the inclusion of motif discovery module proves beneficial for pretraining tasks. Both representation learning-oriented graph classification and pretrain-finetune-oriented transfer learning benefit from motif discovery, resulting in accuracy improvements ranging from 1.8% to 5.2%. These improvements could potentially be attributed to the fact that the motif discovery module effectively learns valuable information about subgraph structures. (2) Motif discovery can provide more substantial improvements for larger scale graph datasets as it may extract more informative structural information. For instance, in unsupervised representation learning, DGPM with the motif discovery module exhibits greater enhancements for REDDIT-B than MUTAG (REDDIT-B, with an average of 429.7 nodes per graph, comprises larger graphs than MUTAG). Similarly, in transfer learning, motif discovery module brings a greater improvement in BBBP than Tox21 (The graph scale in BBBP is larger than Tox21).

Dataset	MUTAG	REDDIT-B	BBBP	Tox21
DGPM	91.20	88.17	71.2	75.3
w/o cross matching	90.80	87.42	69.7	74.4
w/ edit distance	89.67	85.03	69.0	74.3
w/o motif discovery	88.19	84.29	67.4	72.7

Table 3: Ablation studies of the cross matching, motif discovery and measurement of motif similarity, with MUTAG and REDDIT-B for *unsupervised representation learning* and BBBP and Tox21 for *transfer learning*.

Effect of Motif Discovery Criterion By replacing WWL with edit distance, we examined the impact of graph similarity metric employed in motif discovery. As Table 3 shows, WWL offers advantages over edit distance. As WWL not only quantifies structural similarity but also incorporates node feature agreement in graph modeling, thereby effectively supervising graph topology properties.

Effect of Cross-Level Matching Task As shown in Table 3, we observe a significant drop in performance when the cross matching task is not included, amounting to an absolute drop of 0.4% - 1.5%. Interestingly, the datasets that benefit more from motif discovery also tend to benefit more from cross-level matching. These observations affirm that the cross-matching task indeed facilitates the learning of the inherent relationship between node-level information and subgraph-level structure, thereby contributing to a more comprehensive graph pretraining.

Sensitive Analysis

As nodes merge into subgraphs by the EdgePool layer in motif discovery module, the number of EdgePool layers l decides the allowed maximum size of the learned subgraphs, i.e. motifs. Figure 3 illustrates the impact of the number of EdgePool layers. Results for $l = 1$ show significant underperformance across all four datasets. This indicates that, in most scenarios, the motif learning task with a single EdgePool layer is insufficient for effective motif discovery. For REDDIT-B, the model's performance steadily improves with increasing l, reaching 88.17% at $l = 5$. In the case of BBBP, the model achieves its peak performance at $l = 2$, but its effectiveness decreases as l grows larger. The optimal number of EdgePool layers varies across different datasets. This phenomenon is closely related to the motif properties in different graphs. Taking motifs in molec-

	BBBP	Tox21	ToxCast	SIDER	ClinTox	MUV	HIV	BACE	Avg.
GROVER	68.03	76.31	63.39	60.66	76.92	75.78	77.81	79.51	72.30
GROVER(half #motifs)	67.11↓	74.67↓	62.11↓	60.04↓	74.25↓	74.87↓	75.35↓	76.87↓	70.66↓
MGSSL	69.68	76.36	64.12	61.81	79.98	78.68	78.72	79.12	73.56
MGSSL(BRCIS)	67.52↓	73.62↓	62.34↓	59.24↓	77.1↓	77.63↓	76.34↓	75.63↓	71.18↓
DGPM(auto-discovered)	71.15	75.28	64.02	60.3	80.91	75.28	77.26	81.13	73.17

Table 4: Comparison of DGPM and predefined-motif graph self-supervised methods. These models are first pre-trained on ZINC15 and then finetuned on the following datasets. We report ROC-AUC scores (%). For GROVER, we compare with the model pretrained with half number of motifs in designed motif prediction task. For MGSSL, we compare with the model trained with motifs generated by the molecule decomposition method BRCIS and without further processing.

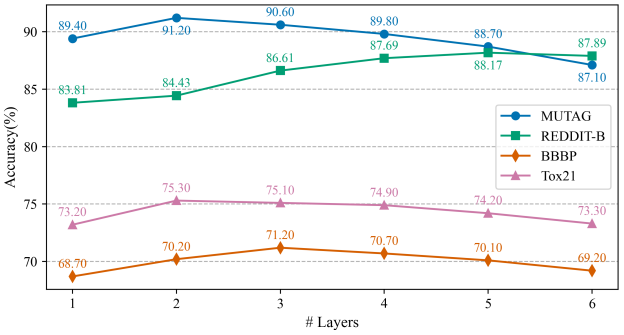
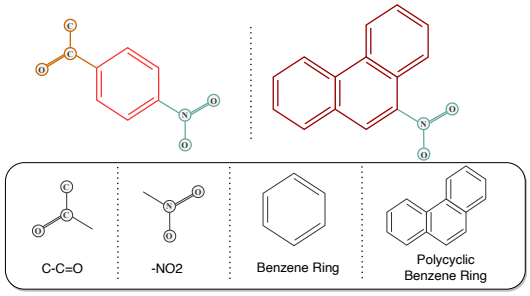
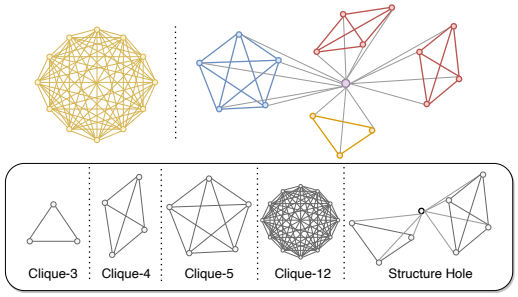


Figure 3: Sensitive analysis of the number of EdgePool layers in motif auto-discovery module.



(a) Molecule graph from MUTAG and discovered motifs.



(b) Movie collaboration graph from IMDB-B and discovered motifs.

Figure 4: Auto-discovered motifs by DGPM from dataset MUTAG and IMDB-BINARY.

ular graphs as an example, their sizes can be up to 6, like the benzene ring. With a limited number of EdgePool layers, the model struggles to learn more complex structures due to the constraint imposed by the maximum size of merged subgraphs. For larger graphs like those in REDDIT-B, intricate structures and large motifs are more prevalent, potentially leading the method to favor deeper networks. Meanwhile, an excessive number of EdgePool layers may adversely affect the framework's performance. This is due to the power law relationship between the number of motifs in a graph and their average size. When the average motif size doubles, the number of motifs decreases by half. In the context of the cross-matching module, a lower number of motifs in a graph leads to less challenging discrimination tasks, which in turn hinders DGPM's learning performance.

Analysis towards Motif Auto-Discovery

Comparison of Predefined Motif Methods To verify the superiority of the proposed motif auto-discovery module, we further simulate a practical scenario when the expert domain knowledge is insufficient and compare two graph pretrain methods using predefined motifs. As shown in Table 4, for GROVER (Rong et al. 2020), which uses a set of predefined motifs for pretrain, when we reduced the number of predefined motifs by half, its performance deteriorated across all datasets. Similarly, for MGSSL (Zhang et al. 2021b), which applies manual filtering to extracted motifs by a molecule decomposition method (BRCIS), if we remove the manual filtering and directly use BRCIS-extracted motifs for pretrain, its performance also decreases on all datasets. These results show that the performance of graph pre-training methods, which depend on predefined motifs, can be easily influenced by the quality of these motifs. In contrast, DGPM's autonomous motif discovery module eliminates the need for manual intervention, enhancing its stability and generalizability.

Case Study For an input graph, motifs and their representations are learned simultaneously by motif auto-discovery in DGPM. This unique feature allows for direct visualization of the underlying motif representations, enhancing the interpretability of the learned representations. To illustrate this, we check discovered graph motifs from diverse domains, encompassing both natural science and social sciences, as depicted in Figure 4.

In MUTAG, the graphs represent nitroaromatic com-

pounds, which are organic molecules containing at least one nitro group (-NO2) attached to a benzene ring. Figure 4 (a) illustrates the auto-discovered motifs of nitroaromatic compounds, including -NO2 and benzene rings, achieved by DGPM without incorporating chemical functional group information. Additionally, DGPM also identifies polycyclic aromatic structures that imply high mutagenic potency (Debnath et al. 1991).

IMDB-B graphs represent movie collaboration networks, with nodes representing actors/actresses and edges denoting co-appearances in movies. IMDB-B can provide an example of social network analysis. Figure 4 (b) displays original collaboration networks along with motifs autonomously discovered by DGPM. These motifs vary in size but share similar clique structures. Cliques, which are complete subgraphs where every member is directly connected to every other, indicate strong connections within a group. Furthermore, Figure 4 (b) identifies a structure hole—an interconnected node that links multiple cliques into a larger collaboration network. Structure holes play a crucial role in social networks by bridging disconnected groups and facilitating information flow between them, fostering collaborations and new ideas (Burt et al. 2001).

Additionally, a comparison between auto-discovered motifs and those decomposed by rules defined in (Zhang et al. 2021b) was conducted for the BBBP dataset. The results revealed that DPGM successfully discovered 64.28% of the motifs, highlighting its proficiency in motif discovery for chemical graphs.

Related Works

Self-Supervised Graph Pretraining

Self-supervised pretraining allows the model to acquire universal knowledge from large-scale unlabeled datasets and offers a superior starting point for downstream tasks (Hao et al. 2019; Zoph et al. 2020). Based on pretext task design, self-supervised graph pretraining methods can be categorized as contrastive or predictive learning.

Constrastive learning. Given training graphs, contrastive learning aims to learn graph encoders such that representations of similar graph instances exhibit concordance while representations of dissimilar instances manifest disagreement. DGI (Veličković et al. 2018) and InfoGraph (Sun et al. 2019) adopt the local-global mutual information maximization to learn node-level and graph-level representations. MVGRL (Hassani and Khasahmadi 2020) leverages graph diffusion to generate an additional view and contrasts node-graph representations of distinct views. GCC (Qiu et al. 2020) utilizes discrimination of subgraph-level instances generated by ego-graph for the pre-training. GRACE (Zhu et al. 2020), GraphCL (You et al. 2020), GCA (Zhu et al. 2021), D-SLA (Kim, Baek, and Hwang 2022) learn the node or graph representation by maximizing the agreement between different augmentations. GGD (Zheng et al. 2022) analyzes the defect of existing contrastive learning methods and introduces a group discrimination paradigm. **Predictive learning**. Compared with contrastive learning, predictive learning methods train the graph encoder f together with a prediction head f', guided by self-generated informative labels. Graph auto-encoders (GAEs) adhere to the essence of auto-encoders (Hinton and Zemel 1993), by reconstructing input graph to learn node representations. Most GAEs include reconstructing structural information (Pan et al. 2018; Wang et al. 2017; Park et al. 2019; Tang, Yang, and Li 2022) by link prediction. GraphMAE (Hou et al. 2022) focuses on reconstructing node features with masked graphs. Its successor, GraphMAE2 (Hou et al. 2023), further refines the framework via a multi-view random re-mask mechanism. In addition to graph auto-encoders, inspired by the success of autoregressive models in natural language processing, GPT-GNN (Hu et al. 2020) designs an attributed graph generation task, including attribute and edge generation, for pretraining GNN models.

Although numerous methods try to leverage graph structural information in self-supervised pretraining, the utilization of subgraph-level patterns, e.g., motif, remains relatively unexplored, both in terms of graph view augmentation and pretext task formation.

Motif-Enhanced Graph Pretraining

Graph motifs are patterns of interconnections occurring in complex networks at numbers that are significantly higher than those in randomized networks (Milo et al. 2002). Serving as fundamental units of graphs, motifs reveal interconnections of nodes and offer insight into the entire graph's characteristics. Hence, motifs hold the potential to enhance the performance of graph pre-training models. Several studies have initiated the design of motif-based self-supervised tasks that incorporate motif semantics to acquire more informative graph representations. GROVER (Rong et al. 2020) utilizes motifs (functional groups) in the molecule, which is extracted by RDKit, as property prediction labels for pretraining tasks. MMGSL (Zhang et al. 2021b) introduces a motif generation task for molecule graphs, decomposing motifs using the BRICS algorithm along with human-designed rules.

However, motifs in existing works are usually predefined based on domain-specific knowledge, thus these models have a high dependency on human intervention and can only be applied to specific domains, constraining the methods' generalizability.

Conclusion and Future Work

In this study, to address the challenges encountered in graph pretraining, we propose DGPM, a dual-level graph self-supervised pretraining with motif discovery. DGPM introduces a motif auto-discovery task to effectively learn subgraph-level topological information. A cross-matching learning module is proposed for better dual-level feature fusion. Comprehensive experiments conducted across diverse graph learning benchmarks demonstrate the effectiveness and generalizability of DGPM [2]. Future directions could involve exploring the extraction of heterogeneous graph's motifs and the automated learning of hyperparameters.

[2]Supplementary materials on the model and experiments can be found at https://github.com/RocccYan/DGPM.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (72104212, 72134007, 62106039), the Natural Science Foundation of Zhejiang Province (LY22G030002), the Fundamental Research Funds for the Central Universities, and Alibaba Group through Alibaba Innovative Research Program.

References

Burt, R. S.; et al. 2001. Structural holes versus network closure as social capital. *Social capital: Theory and research*, 31–56.

Chang, C.-C.; and Lin, C.-J. 2011. LIBSVM: a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3): 1–27.

Chen, X.; Cai, R.; Fang, Y.; Wu, M.; Li, Z.; and Hao, Z. 2023. Motif Graph Neural Network. *IEEE Transactions on Neural Networks and Learning Systems*.

Debnath, A. K.; Lopez de Compadre, R. L.; Debnath, G.; Shusterman, A. J.; and Hansch, C. 1991. Structure-activity relationship of mutagenic aromatic and heteroaromatic nitro compounds. correlation with molecular orbital energies and hydrophobicity. *Journal of medicinal chemistry*, 34(2): 786–797.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Hao, Y.; Dong, L.; Wei, F.; and Xu, K. 2019. Visualizing and understanding the effectiveness of BERT. *arXiv preprint arXiv:1908.05620*.

Hassani, K.; and Khasahmadi, A. H. 2020. Contrastive multi-view representation learning on graphs. In *International conference on machine learning*, 4116–4126. PMLR.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9729–9738.

Hinton, G. E.; and Zemel, R. 1993. Autoencoders, minimum description length and Helmholtz free energy. *Advances in neural information processing systems*, 6.

Hou, Z.; He, Y.; Cen, Y.; Liu, X.; Dong, Y.; Kharlamov, E.; and Tang, J. 2023. GraphMAE2: A Decoding-Enhanced Masked Self-Supervised Graph Learner. In *Proceedings of the ACM Web Conference 2023*, 737–746.

Hou, Z.; Liu, X.; Cen, Y.; Dong, Y.; Yang, H.; Wang, C.; and Tang, J. 2022. Graphmae: Self-supervised masked graph autoencoders. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 594–604.

Hu, W.; Liu, B.; Gomes, J.; Zitnik, M.; Liang, P.; Pande, V.; and Leskovec, J. 2019. Strategies for pre-training graph neural networks. *arXiv preprint arXiv:1905.12265*.

Hu, Z.; Dong, Y.; Wang, K.; Chang, K.-W.; and Sun, Y. 2020. Gpt-gnn: Generative pre-training of graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1857–1867.

Kim, D.; Baek, J.; and Hwang, S. J. 2022. Graph self-supervised learning with accurate discrepancy learning. *Advances in Neural Information Processing Systems*, 35: 14085–14098.

Ma, Y.; and Tang, J. 2021. *Deep learning on graphs*. Cambridge University Press.

Milo, R.; Shen-Orr, S.; Itzkovitz, S.; Kashtan, N.; Chklovskii, D.; and Alon, U. 2002. Network motifs: simple building blocks of complex networks. *Science*, 298(5594): 824–827.

Narayanan, A.; Chandramohan, M.; Venkatesan, R.; Chen, L.; Liu, Y.; and Jaiswal, S. 2017. graph2vec: Learning distributed representations of graphs. *arXiv preprint arXiv:1707.05005*.

Oliver, C.; Chen, D.; Mallet, V.; Philippopoulos, P.; and Borgwardt, K. 2022. Approximate network motif mining via graph learning. *arXiv preprint arXiv:2206.01008*.

Pan, S.; Hu, R.; Long, G.; Jiang, J.; Yao, L.; and Zhang, C. 2018. Adversarially regularized graph autoencoder for graph embedding. *arXiv preprint arXiv:1802.04407*.

Park, J.; Lee, M.; Chang, H. J.; Lee, K.; and Choi, J. Y. 2019. Symmetric graph convolutional autoencoder for unsupervised graph representation learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, 6519–6528.

Qiu, J.; Chen, Q.; Dong, Y.; Zhang, J.; Yang, H.; Ding, M.; Wang, K.; and Tang, J. 2020. Gcc: Graph contrastive coding for graph neural network pre-training. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, 1150–1160.

Rong, Y.; Bian, Y.; Xu, T.; Xie, W.; Wei, Y.; Huang, W.; and Huang, J. 2020. Self-supervised graph transformer on large-scale molecular data. *Advances in Neural Information Processing Systems*, 33: 12559–12571.

Shervashidze, N.; Schweitzer, P.; Van Leeuwen, E. J.; Mehlhorn, K.; and Borgwardt, K. M. 2011. Weisfeiler-lehman graph kernels. *Journal of Machine Learning Research*, 12(9).

Sterling, T.; and Irwin, J. J. 2015. ZINC 15–ligand discovery for everyone. *Journal of chemical information and modeling*, 55(11): 2324–2337.

Sun, F.-Y.; Hoffmann, J.; Verma, V.; and Tang, J. 2019. Infograph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization. *arXiv preprint arXiv:1908.01000*.

Tang, M.; Yang, C.; and Li, P. 2022. Graph auto-encoder via neighborhood wasserstein reconstruction. *arXiv preprint arXiv:2202.09025*.

Togninalli, M.; Ghisu, E.; Llinares-López, F.; Rieck, B.; and Borgwardt, K. 2019. Wasserstein weisfeiler-lehman graph kernels. *Advances in neural information processing systems*, 32.

Veličković, P.; Fedus, W.; Hamilton, W. L.; Liò, P.; Bengio, Y.; and Hjelm, R. D. 2018. Deep graph infomax. *arXiv preprint arXiv:1809.10341*.

Wang, C.; Pan, S.; Long, G.; Zhu, X.; and Jiang, J. 2017. Mgae: Marginalized graph autoencoder for graph clustering. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 889–898.

Wu, Z.; Ramsundar, B.; Feinberg, E. N.; Gomes, J.; Geniesse, C.; Pappu, A. S.; Leswing, K.; and Pande, V. 2018. MoleculeNet: a benchmark for molecular machine learning. *Chemical science*, 9(2): 513–530.

Xie, Y.; Xu, Z.; Zhang, J.; Wang, Z.; and Ji, S. 2022. Self-supervised learning of graph neural networks: A unified review. *IEEE transactions on pattern analysis and machine intelligence*, 45(2): 2412–2429.

Xu, D.; Cheng, W.; Luo, D.; Chen, H.; and Zhang, X. 2021a. Infogcl: Information-aware graph contrastive learning. *Advances in Neural Information Processing Systems*, 34: 30414–30425.

Xu, K.; Hu, W.; Leskovec, J.; and Jegelka, S. 2018. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*.

Xu, M.; Wang, H.; Ni, B.; Guo, H.; and Tang, J. 2021b. Self-supervised graph-level representation learning with local and global structure. In *International Conference on Machine Learning*, 11548–11558. PMLR.

Yanardag, P.; and Vishwanathan, S. 2015. Deep graph kernels. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, 1365–1374.

Ying, Z.; You, J.; Morris, C.; Ren, X.; Hamilton, W.; and Leskovec, J. 2018. Hierarchical graph representation learning with differentiable pooling. *Advances in neural information processing systems*, 31.

You, Y.; Chen, T.; Shen, Y.; and Wang, Z. 2021. Graph contrastive learning automated. In *International Conference on Machine Learning*, 12121–12132. PMLR.

You, Y.; Chen, T.; Sui, Y.; Chen, T.; Wang, Z.; and Shen, Y. 2020. Graph contrastive learning with augmentations. *Advances in neural information processing systems*, 33: 5812–5823.

Zhang, H.; Wu, Q.; Yan, J.; Wipf, D.; and Yu, P. S. 2021a. From canonical correlation analysis to self-supervised graph neural networks. *Advances in Neural Information Processing Systems*, 34: 76–89.

Zhang, Z.; Liu, Q.; Wang, H.; Lu, C.; and Lee, C.-K. 2021b. Motif-based graph self-supervised learning for molecular property prediction. *Advances in Neural Information Processing Systems*, 34: 15870–15882.

Zheng, Y.; Pan, S.; Lee, V.; Zheng, Y.; and Yu, P. S. 2022. Rethinking and scaling up graph contrastive learning: An extremely efficient approach with group discrimination. *Advances in Neural Information Processing Systems*, 35: 10809–10820.

Zhu, Y.; Xu, Y.; Yu, F.; Liu, Q.; Wu, S.; and Wang, L. 2020. Deep graph contrastive representation learning. *arXiv preprint arXiv:2006.04131*.

Zhu, Y.; Xu, Y.; Yu, F.; Liu, Q.; Wu, S.; and Wang, L. 2021. Graph contrastive learning with adaptive augmentation. In *Proceedings of the Web Conference 2021*, 2069–2080.

Zoph, B.; Ghiasi, G.; Lin, T.-Y.; Cui, Y.; Liu, H.; Cubuk, E. D.; and Le, Q. 2020. Rethinking pre-training and self-training. *Advances in neural information processing systems*, 33: 3833–3845.