Can Graph Neural Networks Learn Language with Extremely Weak Text Supervision?

Zihao Li[1], Lecheng Zheng[1], Bowen Jin[1], Dongqi Fu[2], Baoyu Jing[1], Yikun Ban[1], Jingrui He[1], Jiawei Han[1]

[1]University of Illinois Urbana-Champaign, [2]Meta AI

Correspondence: zihaoli5@illinois.edu

Abstract

While great success has been achieved in building vision models with Contrastive Language-Image Pre-training (CLIP) over internet-scale image-text pairs, building transferable Graph Neural Networks (GNNs) with CLIP pipeline is challenging because of the scarcity of labeled data and text supervision, different levels of downstream tasks, and the conceptual gaps between domains. In this work, to address these issues, we propose a multi-modal prompt learning paradigm to effectively adapt pre-trained GNN to downstream tasks and data, given only a few semantically labeled samples, each with extremely weak text supervision. Our new paradigm embeds the graphs directly in the same space as the Large Language Models (LLMs) by learning both graph prompts and text prompts simultaneously. We demonstrate the superior performance of our paradigm in few-shot, multi-task-level, and cross-domain settings. Moreover, we build the first CLIP-style zero-shot classification prototype that can generalize GNNs to unseen classes with extremely weak text supervision. The code is available at https://github.com/Violet24K/Morpher.

1 Introduction

Graphs are constructed from real scenarios, but GNNs, optimized according to numerical labels, still do not *understand* what a label represents in the real world. To solve the issue of predetermined numerical categories, CLIP (Radford et al., 2021) leverages natural language supervision by jointly training an image encoder and a text encoder in the same embedding space at scale. CLIP has demonstrated the ability to train high-quality, generalizable vision models (Radford et al., 2021; Jia et al., 2021; Li et al., 2022), which can adapt to diverse downstream tasks. Similar frameworks have been successfully extended to video (Xu et al., 2021), 3D images (Hess et al., 2024), speech (Shih et al., 2022) and audio (Guzhov et al., 2022), consistently demonstrating that alignment with text enhances the transferability of encoders. As for graphs, so far, such graph-text alignment has only been explored in the molecular domain (Luo et al., 2023; Liu et al., 2023e) and on text-attributed graphs (Wen and Fang, 2023; Li et al., 2023a; Jin et al., 2023b; Yan et al., 2023), where the paired graph-text data is relatively sufficient for joint pre-training.

However, extending this paradigm to more general graph data poses significant challenges due to three facts. First, compared with language or vision data, graph data is very scarce and the text supervision is extremely weak (Liu et al., 2023a; Chen et al., 2023b; Manchanda et al., 2023). Besides the number of samples being much smaller than images, many graph datasets are used for classification, where the label names consist of only a few tokens. Second, the task space of graph data could be on node-level, edge-level, and graph-level. Third, in general, language tokens and visual objects retain the same conceptual meaning across different distributions, but the same graph structure may have distinct interpretations in different domains.

Jointly pre-training graph and text encoders is impractical for graph data with extremely weak text supervision. Fortunately, we can deal with the two modalities separately for pre-training: large language models have already been extensively pre-trained, and tremendous efforts have been devoted to pre-train GNNs through self-supervision (Hu et al., 2020a; Liu et al., 2023c; Zheng et al., 2024c,d). However, even with a pre-trained graph model, effectively adapting it to both the semantic embedding space for text alignment and diverse downstream tasks remains non-trivial. This raises a critical question:

How to adapt pre-trained GNNs to the semantic embedding space given limited downstream data,

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 11138–11165

i.e., few samples and weak text supervision?

This paper aims to answer this question based on the following observations: (1) Semantic text embedding spaces do not necessarily result from joint pre-training. In fact, the embedding spaces of encoder LLMs are inherently semantic and high-quality, as LLMs are trained on massive text data and demonstrate strong performances. (2) When the downstream data are limited, prompt learning (Li and Liang, 2021; Houlsby et al., 2019; Zhang et al., 2022a; Lester et al., 2021) provides a better option than fine-tuning as much fewer parameters not only makes the optimization more efficient but also requires less resource than fine-tuning a large model. Notably, some works have explored prompt learning for better alignment and obtained improvement in vision prediction (Zhou et al., 2022b; Khattak et al., 2023). Inspired by these observations, we propose a prompting-based paradigm with an LLM that aligns the GNN representations in the semantic embedding space, while keeping the parameters of both GNN and LLM frozen.

When adapting the representation from one modality to another, solely prompting a single modality could be sub-optimal, as it limits the adjustment to downstream tasks in the other modality (Khattak et al., 2023). To this end, we propose Multi-modal Prompt Learning for Graph Neural Networks (Morpher). Given a pre-trained GNN and few-shot semantically labeled graph data with weak text supervision, we assume zeroth-order access to a pre-trained LLM. Then, to leverage its high-quality semantic embedding space, Morpher connects and aligns the graph embeddings to it through prompting on both modalities with a cross-modal projector. Nonetheless, designing such a paradigm is more challenging than vision-language models. First, we lack jointly pre-trained encoders for the two modalities; instead, we only have two encoders pre-trained independently in each modality. Second, determining how to prompt the graph modality is non-trivial and remains a trending research topic. Third, the downstream data for GNN usually have much fewer labeled classes and labeled samples than Vision-Language models, and the text supervision is extremely weak. Our contributions towards tackling these challenges are:

- We analyze that, state-of-the-art graph prompt (Sun et al., 2023a) is often unable to learn good representations of the downstream data. We further improve it to prevent unstable optimization.
- To connect and adapt the pre-trained GNN with LLM effectively with extremely weak text supervision, we propose Morpher, the first graph-text multi-modal prompt learning paradigm to align the representations of GNN and LLM without fine-tuning any of their parameters.
- With extremely weak text supervision, we demonstrate our improved graph prompt and Morpher under few-shot, multi-task, and cross-domain settings. To show that GNN learns language dependency through Morpher, we present the first CLIP-style zero-shot generalization prototype where the GNN can predict unseen classes.

2 Background

We use calligraphic letters (e.g., $\mathcal{A}$) for sets, and specifically $\mathcal{G}$ for graphs. We use bold capital letters for matrices (e.g., $\mathbf{A}$). For matrix indices, we use $\mathbf{A}(i,j)$ to denote the entry in the i^{th} row and the j^{th} column. $\mathbf{A}(i,:)$ is the i^{th} row in $\mathbf{A}$.

Graph Neural Networks. We use $\mathcal{G} = (\mathbf{A}, \mathbf{X})$ to denote a graph with node set $\mathcal{V}$ and edge set $\mathcal{E}$, where $\mathbf{A} \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$ is the adjacency matrix and $\mathbf{X} \in \mathbb{R}^{|\mathcal{V}|\times d}$ is the node feature matrix. $\mathbf{A}(u,v) = 1$ if there is an edge connecting u and v; otherwise $\mathbf{A}(u,v) = 0$. A graph neural network $f_\phi^g(\cdot)$ with hidden dimension d_g encodes $\mathcal{G}$ into the embedding space: $f_\phi^g(\mathcal{G}) \in \mathbb{R}^{|\mathcal{V}|\times d_g}$, which could preserve both feature and structure information of $\mathcal{G}$.

Few-shot Prompt Learning. Let $f_\phi^t(\cdot)$ denote the LLM encoder with embedding dimension d_t. For a series of input tokens $\{x_k\}_{k=1}^K$, the LLM encoder embeds it as a matrix $\mathbf{X}_t = f_\phi^t(\{x_k\}_{k=1}^K) \in \mathbb{R}^{K\times d_t}$. Prompt learning initializes a tunable matrix $\mathbf{P}_\theta^t \in \mathbb{R}^{n_t\times d_t}$, where n_t denotes the number of text prompt tokens. Then, this tunable matrix is concatenated with the input tokens' embeddings to form $[\mathbf{P}_\theta^t; \mathbf{X}_t]_{dim=0} \in \mathbb{R}^{(K+n_t)\times d_t}$.

Our Problem Set-up. Given a pre-trained GNN $f_\phi^g(\cdot)$ with embedding dimension d_g and a pre-trained LLM encoder $f_\phi^t(\cdot)$ with embedding dimension d_t. Without loss of generality, we assume the downstream task is graph-level classification, as node-level or edge-level GNN tasks can be reformulated as graph-level by inducing ego-graphs within neighbor distance γ. For L-shot graph classification, we are given limited text-labeled pairs $\{(\mathcal{G}_i, t_c)\}_{i=1}^L$ for each class c. Each text label t_c consists of only a few tokens. Assuming $\mathcal{T}$ is the

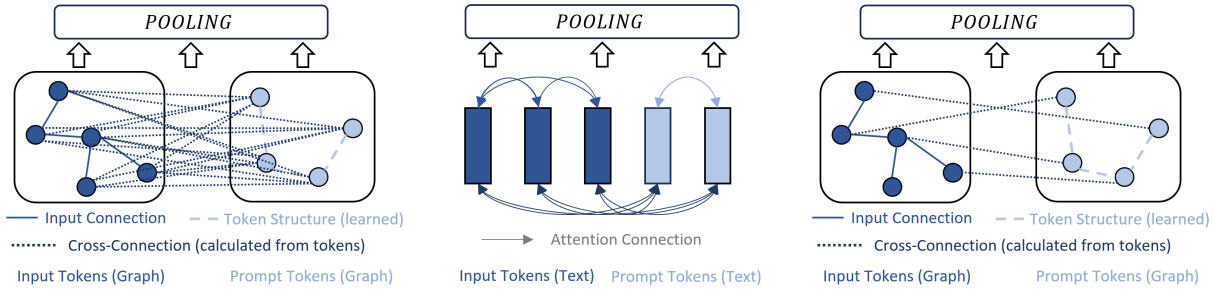


Figure 1: Cross-connections overwhelm inner-connections in current graph prompt design, which may be unstable during training (left); attention in NLP where $3 \times 2 = 6$ cross-connections and $3 + 1 = 4$ inner-connections are balanced (middle); and our balanced graph prompt design (right). **The cross-connections between input and prompt should have a consistent scale with the input connections.**

set of all text labels t_c, we are provided a set of test graphs $\{\mathcal{G}_j\}_{j=1}^{L_{test}}$ and want to predict the text label $t_j \in \mathcal{T}$ for each test graph $\mathcal{G}_j$.

3 Improving Single-modal Graph Prompt

Unlike prompting text data, prompting graph data presents a significant challenge due to the non-euclidean nature of graphs. The pioneering work (Sun et al., 2023a) designs the graph prompt still as a graph and then inserts it into the original graph.

Current Graph Prompt Design. To prompt a graph $\mathcal{G}$, each prompt token is a new node. Let n_g denote the number of prompt tokens and $\mathcal{P} = \{p_i\}_{i=1}^{n_g}$ denote the set of prompt tokens. The graph prompt is formulated by a tunable matrix $\mathbf{P}_\theta^g \in \mathbb{R}^{n_g \times d}$, where d is the node feature dimension. Each row vector $\mathbf{P}_\theta^g(i,:)$ is the feature of the prompt token p_i. Then, the mechanism to prompt a graph $\mathcal{G} = (\mathbf{A}, \mathbf{X})$ with n nodes and d feature dimension is (Sun et al., 2023a)

- Compute inner-connections to construct the prompt graph $\mathcal{G}_p = (\mathbf{A}_p, \mathbf{X}_p) = (\mathbf{A}_p, \mathbf{P}_\theta^g)$. For two prompt tokens p_i and p_j, $\mathbf{A}_p(i,j) = 1 \iff \sigma(\mathbf{P}_\theta^g(i,:)\mathbf{P}_\theta^g(j,:)^\top) > \delta_{inner}$, where $\sigma(\cdot)$ is the sigmoid function.

- Compute cross-connections to insert the prompt graph $\mathcal{G}_p$ into $\mathcal{G}$. Similarly, for $x_i \in \mathcal{G}$ and $p_j \in \mathcal{G}_p$, there is an edge between them if and only if $\sigma(\mathbf{X}(i,:)\mathbf{P}_\theta^g(j,:)^\top) > \delta_{cross}$.

- Construct the prompted graph (i.e., manipulated graph) $\mathcal{G}_m = (\mathbf{A}_m, \mathbf{X}_m)$. The overall adjacency matrix $\mathbf{A}_m \in \mathbb{R}^{(n+n_g)\times(n+n_g)}$ is constructed from the original adjacency matrix $\mathbf{A}$, the inner edges $\mathbf{A}_p$ and the cross edges. The overall node feature matrix is concatenated from the prompt token features and the original input node features: $\mathbf{X}_m = [\mathbf{P}_\theta^g; \mathbf{X}]_{dim=0} \in \mathbb{R}^{(n+n_g)\times d}$.

Issues associated with the current design. The input node features of most real-world datasets are sparse, resulting from the construction process (Yang et al., 2016; Morris et al., 2020; Dwivedi et al., 2023). As shown in Appendix Table 4, $||\mathbf{X}(i,:)||_1$ is typically 1. As the initialization of each token feature tensor $\mathbf{P}_\theta^g(i,:)$ is close to $\vec{\mathbf{0}}$ to stabilize gradients, for any node i and token p_j, the dot products $\mathbf{X}(i,:)\mathbf{P}_\theta^g(j,:)^\top$ is close to 0, and the sigmoid value is very close to 0.5. Consequently, if we want the graph prompt to have cross-connections, we must set $\delta_{cross} < 0.5$. Then, as the sigmoid values are close to 0.5, the cross-connections will be dense, i.e., almost every node in the original graph is connected with every node token in the prompt graph. For two different graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ in the same task, the prompt graph $\mathcal{G}_p$ is identical. Since the GNNs aggregate the node features, their embeddings $f_\phi^g(\mathcal{G}_1)$ and $f_\phi^g(\mathcal{G}_2)$ are approximately the same because the features in the prompt graph overwhelm the features in the original graphs due to the dense cross-connections. Then, even if $\mathcal{G}_1$ and $\mathcal{G}_2$ have different labels, the task head classifier cannot be trained to distinguish them [1].

In Appendix C.1, we show that initializing graph prompt token feature tensor with higher variance cannot effectively address this problem.

Improved Graph Prompt Design. The issue of the current graph prompt lies in the significant imbalance between original connections within the input graph and the input-prompt cross-connections, as illustrated in Figure 1 (left). We also visualize

[1] In fact, similar training instability problems have been observed by another work (Zhao et al., 2024).

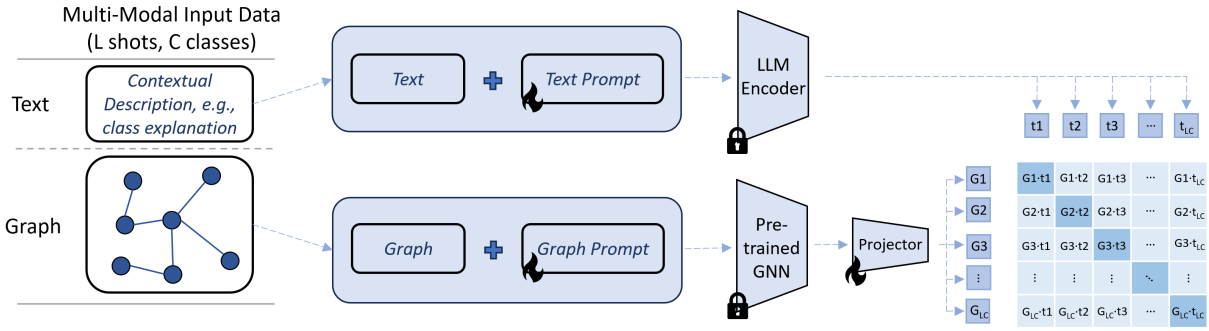


Figure 2: Similar to CLIP backbone, Morpher adapts the graph representations to semantic space through multi-modal prompt learning, even if the GNN and LLM are not jointly trained and are kept frozen.

the standard NLP attention mechanism (Vaswani et al., 2017) in Figure 1 (middle). When a sequence of text prompt tokens p_i^t is prepended to the input sequence, the features of the prompt tokens will be aggregated with those of the input tokens through a dense "cross-connection", i.e., attention. Simultaneously, the features within the input sequence are also densely aggregated via attention, maintaining a balance with the prompt-input aggregation to prevent overwhelming. Inspired by this, we deem that a balance could be achieved by approximately equalizing the number of cross-connections with that of input graph connections, i.e., n_e. Since the connection of a graph dataset is often sparse, we constrain the cross-connections to be sparse as well. Therefore, we set the number of cross-connections to at most n_e by connecting each node in the input graph with at most $\lfloor \frac{n_e}{a} \rfloor$ prompt tokens. Then, we can safely use a small δ_{cross} and cosine similarity $\frac{\mathbf{X}(i,:)\cdot\mathbf{P}_\theta^g(j,:)^\top}{\|\mathbf{X}(i,:)\|_2\|\mathbf{P}_\theta^g(j,:)\|_2}$ instead of $\sigma(\mathbf{X}(i,:)\mathbf{P}_\theta^g(j,:)^\top)$ to calculate the cross-connections.

4 GNN Multi-modal Prompt Learning

To adapt the GNN embeddings to the LLM's semantic embedding space and leverage the additional weak supervision provided by the text associated with graph labels, we explore the potential of multi-modal prompt learning for both graphs and language. This approach is motivated by the intuition that only prompting on the graph data may limit the flexibility to adjust the LLM representation space. The overall paradigm of Morpher is illustrated in Figure 2. Given the data $\{(\mathcal{G}_i, t_i)\}_{i=1}^{L\times C}$, we aim to align graph embedding $\text{readout}(f_\phi^g(\mathcal{G}_i))$ with $\text{readout}(f_\phi^t(\text{Tokenize}(t_i)))$. Yet one direct issue is that, $\text{readout}(f_\phi^g(\mathcal{G}_i)) \in \mathbb{R}^{1\times d_g}$ and $\text{readout}(f_\phi^t(\text{Tokenize}(t_i))) \in \mathbb{R}^{1\times d_t}$ may have distinct dimensions. To address this issue, we adopt a cross-modal projector that learns to map the graph embedding space to the text embedding space. For an input d_g-dimensional graph embedding $\mathbf{v}$, the projector maps it to a vector $\widetilde{\mathbf{v}}$ in the d_t-dimensional text embedding space:

$$\widetilde{\mathbf{v}} = \text{Proj}_\theta(\mathbf{v}) := \tanh(\mathbf{W}\mathbf{v} + \mathbf{b}) \in \mathbb{R}^{1\times d_t} \quad (1)$$

As discussed in Sections 2 and 3, we introduce the text prompt $\mathbf{P}_\theta^t \in \mathbb{R}^{n_t\times d_t}$ with n_t text prompt tokens and the graph prompt $\mathbf{P}_\theta^g \in \mathbb{R}^{n_g\times d}$ with n_g graph prompt tokens. The graph prompting function $\psi_g(\cdot, \mathbf{P}_\theta^g)$ modifies a given graph $\mathcal{G}$ into a manipulated graph $\mathcal{G}m = \psi_g(\mathcal{G}, \mathbf{P}_\theta^g)$.

Let $\omega_t(\cdot, \mathbf{P}_\theta^t)$ be the prompted text embedding given input text t. For the text prompt methods we choose, the prompted embedding is

$$\omega_t(t, \mathbf{P}_\theta^t) = [\mathbf{P}_\theta^t; f_\phi^t(\text{Tokenize}(t))]_{dim=0} \quad (2)$$

Let $\omega_g(\cdot, \mathbf{P}_\theta^g)$ be the prompted graph embedding given input graph $\mathcal{G}$, then we have:

$$\omega_g(\mathcal{G}, \mathbf{P}_\theta^g) = f_\phi^g(\mathcal{G}_m) = f_\phi^g(\psi_g(\mathcal{G}, \mathbf{P}_\theta^g)) \quad (3)$$

For the whole prompted text and the whole prompted graph of the sample $(\mathcal{G}_i, t_i)$, we apply readout (e.g., mean-pooling, max-pooling, etc.) to get their embedding:

$$\mathbf{h}_i^t = \text{readout}(\omega_t(t_i, \mathbf{P}_\theta^t)) \in \mathbb{R}^{1\times d_t} \quad (4)$$

$$\mathbf{h}_i^\mathcal{G} = \text{readout}(\omega_g(\mathcal{G}_i, \mathbf{P}_\theta^g)) \in \mathbb{R}^{1\times d_g} \quad (5)$$

For the given data $(\mathcal{G}_i, t_i)$, we compute the normalized embedding of prompted $\mathcal{G}_i$ and project it to the text embedding space through the projector:

$$\mathbf{z}_{norm,i}^\mathcal{G} = \frac{\mathbf{h}_i^\mathcal{G}}{||\mathbf{h}_i^\mathcal{G}||_2} = \frac{\text{readout}(\omega_g(\mathcal{G}_i, \mathbf{P}_\theta^g))}{||\text{readout}(\omega_g(\mathcal{G}_i, \mathbf{P}_\theta^g))||_2} \quad (6)$$

$$\mathbf{z}_i^\mathcal{G} = \text{Proj}_\theta(\mathbf{z}_{norm,i}^\mathcal{G}) \quad (7)$$

For the text embeddings, since for limited data the set $\mathcal{T} = \{t_i\}_{i=1}^{C}$ may contain texts that are semantically close as discussed in Appendix A.2, we extract a subspace in the text embedding space by normalizing the embedding as follows. We further normalize the text embeddings to the unit sphere.

$$\mu = \frac{1}{L}\sum_{i=1}^{L}\mathbf{h}_i^t, \quad \mathbf{h}_{norm,i}^t = \mathbf{h}_i^t - \mu \tag{8}$$

$$\mathbf{z}_i^t = \frac{\mathbf{h}_{norm,i}^t}{||\mathbf{h}_{norm,i}^t||_2} = \frac{\text{readout}(\omega_t(t_i, \mathbf{P}_\theta^t)) - \mu}{||\text{readout}(\omega_t(t_i, \mathbf{P}_\theta^t)) - \mu||_2} \tag{9}$$

Finally, we use the in-batch similarity-based contrastive loss with temperature τ to train text prompts, graph prompts, and the projector.

$$\mathcal{L}_{G\to T} = -\frac{1}{B}\sum_{i=1}^{B}\log\frac{\exp(\mathbf{z}_i^{\mathcal{G}}\cdot\mathbf{z}_i^t/\tau)}{\sum_{j=1}^{B}\exp(\mathbf{z}_i^{\mathcal{G}}\cdot\mathbf{z}_j^t/\tau)} \tag{10}$$

During inference stage, for an input graph $\mathcal{G}_i$ and text label candidates $\mathcal{T} = \{t_i\}_{i=1}^{C}$, we compute the embedding $\mathbf{z}_i^{\mathcal{G}} = \text{Proj}_\theta(\frac{\text{readout}(\omega_g(\mathcal{G}_i, \mathbf{P}_\theta^g))}{||\text{readout}(\omega_g(\mathcal{G}_i, \mathbf{P}_\theta^g))||_2})$ using trained $\mathbf{P}_\theta^g$ and $\text{Proj}_\theta(\cdot)$. Then, we compute $\mathbf{z}_i^t$ as Equations 9 and 10. Finally, $\mathcal{G}_i$ will be classified to associate with text label $\arg\max_{1\le i\le C}(\mathbf{z}_i^{\mathcal{G}}\cdot\mathbf{z}_i^t)$.

5 Experiments

We show that both Morpher and the improved graph prompt more effectively adapt pre-trained GNNs to the specific downstream classification task. We use RoBERTa (Liu et al., 2019) as the LLM encoder for Morpher in the main experiments. We also validate the performance of Morpher with ELECTRA (Clark et al., 2020) and DistilBERT (Sanh et al., 2019) in Appendix B.3.

Datasets. We use real-world graph datasets from PyTorch Geometric (Fey and Lenssen, 2019), including one molecular dataset MUTAG (Morris et al., 2020); two bioinformatic datasets ENZYMES and PROTEINS (Borgwardt et al., 2005); one computer vision dataset MSRC_21C (Neumann et al., 2016); three citation network datasets Cora, CiteSeer and PubMed (Yang et al., 2016). We use real-world class names as text labels. The text supervision is extremely weak, as each text label contains no more than five words. More details are summarized in Appendix A.

Pre-training algorithms and GNN backbones. To pretrain GNNs for evaluation, we adopt GraphCL (You et al., 2020) and SimGRACE (Xia et al., 2022) to pre-train three widely used GNN backbones: GCN (Kipf and Welling, 2016), GAT (Yun et al., 2019) and GraphTransformer (GT) (Khosla et al., 2020). Additionally, in Appendix B.4, we verify the effectiveness of our methods on GNNs pre-trained using GraphMAE (Hou et al., 2022) and MVGRL (Hassani and Ahmadi, 2020), two other representative GNN self-supervised learning algorithms. For each dataset, to pre-train GNNs, we leverage self-supervised learning methods on all the graphs without any label information.

Baselines. We compare our methods with the following baselines: (1) training a GNN from scratch supervised by few-shot data (*"supervised"*); (2) fine-tuning a task head together with pre-trained GNN (*"fine-tune"*). We allow GNNs to be tunable for "*supervised*" and "*fine-tune*"; (3) state-of-the-art graph prompting algorithms: All-in-one (*"AIO"*) (Sun et al., 2023a), which is the only graph prompting algorithm that supports multiple tasks in node-level, edge-level and graph-level to the best of our knowledge; GPF-plus (Fang et al., 2023) which prompt on graph features and Gprompt (Liu et al., 2023d) which is based on subgraph similarity.

5.1 Few-shot Learning

We investigate our improved graph prompt ("*ImprovedAIO*") and Multimodal prompt ("*Morpher*") to adapt frozen pre-trained GNNs using few-shot data. We focus on graph-level classification here and will further investigate the few-shot learning ability at other task levels in Section 5.2. Our few-shot learning setting is more challenging than existing works (Sun et al., 2023a, 2022) as we only allow no more than 10 labeled training and validation samples for each class. The results are shown in Table 1. By observations, given the same pre-trained GNN, our ImprovedAIO outperforms all the existing baseline methods. This improvement is attributed to its design, which restricts cross-connections, ensuring stable training and optimization. Moreover, Our Morpher can achieve an absolute further accuracy improvement over the baselines across all datasets. Its superior performance, even under extremely weak text supervision, stems from its ability to dynamically adapt and align the graph and language representation spaces with prompt learning. This flexibility enables Morpher to better leverage the semantic information from weakly-supervised text labels while preserving the structural integrity of the graph embeddings, resulting in more robust and accurate predictions.

Training schemes	GNN pretraining	MUTAG Acc	MUTAG F1	ENZYMES Acc	ENZYMES F1	PROTEINS Acc	PROTEINS F1	MSRC_21C Acc	MSRC_21C F1
Supervised	N/A + GCN	66.00	66.67	16.67	8.68	65.89	60.77	38.85	35.32
	N/A + GAT	66.00	65.69	16.45	4.65	64.75	64.08	41.14	39.86
	N/A + GT	66.66	66.26	15.62	4.22	62.81	57.12	38.28	41.62
Pre-train + Fine-tune	GraphCL+GCN	70.00	70.23	17.91	11.82	65.89	61.23	40.00	43.89
	GraphCL+GAT	70.00	69.73	17.91	10.46	65.16	63.92	44.57	45.74
	GraphCL+GT	68.00	67.81	17.70	8.99	63.28	56.41	41.71	43.73
	SimGRACE+GCN	66.67	67.27	17.29	8.78	66.82	64.70	40.57	43.84
	SimGRACE+GAT	70.67	69.10	16.87	7.18	65.42	63.65	42.85	42.37
	SimGRACE+GT	69.33	69.77	16.24	6.08	65.98	62.31	39.42	40.78
AIO (Sun et al., 2023a)	GraphCL+GCN	64.67	39.27	17.50	4.97	61.35	44.93	3.59	10.09
	GraphCL+GAT	64.67	39.27	17.50	4.97	59.21	37.19	14.37	3.11
	GraphCL+GT	73.33	72.06	18.33	9.09	40.79	28.97	17.96	8.30
	SimGRACE+GCN	64.67	39.27	16.04	4.61	67.42	60.87	34.73	18.16
	SimGRACE+GAT	64.67	39.27	16.04	4.61	59.21	37.19	7.78	1.79
	SimGRACE+GT	36.00	27.26	17.50	8.15	50.56	49.34	32.34	15.13
ImprovedAIO (Ours)	GraphCL+GCN	77.33	77.74	18.13	11.98	65.89	65.97	42.85	45.91
	GraphCL+GAT	74.67	75.51	18.33	11.26	65.76	66.05	46.85	51.39
	GraphCL+GT	74.67	74.67	19.16	9.04	68.12	68.18	42.85	43.54
	SimGRACE+GCN	68.00	69.01	17.91	9.02	66.82	66.40	44.57	49.24
	SimGRACE+GAT	77.33	77.20	18.75	9.39	66.91	65.49	45.14	42.31
	SimGRACE+GT	71.33	72.06	18.95	11.25	68.59	68.84	40.57	42.82
Morpher (Ours)	GraphCL+GCN	78.67	78.09	20.41	15.20	67.47	66.40	45.14	49.62
	GraphCL+GAT	**79.33**	**79.15**	**23.12**	**18.01**	70.89	70.30	**50.85**	**54.48**
	GraphCL+GT	76.00	76.51	19.58	13.28	**73.53**	**72.48**	45.71	48.41
	SimGRACE+GCN	69.33	70.27	19.79	14.94	67.10	66.15	45.71	51.24
	SimGRACE+GAT	78.00	77.65	20.21	16.27	68.12	67.26	45.71	51.13
	SimGRACE+GT	74.00	74.84	19.16	14.29	71.76	71.75	44.00	48.16
IMP of ImprovedAIO (%)		3.89 ↑	4.67 ↑	0.90 ↑	1.07 ↑	1.41 ↑	4.64 ↑	2.29 ↑	2.34 ↑
Further IMP of Morpher (%)		2.00 ↑	1.72 ↑	1.84 ↑	5.01 ↑	2.8 ↑	2.24 ↑	2.38 ↑	4.64 ↑

Table 1: Few-shot graph classification performance (%). *IMP (%)*: the average improvement (absolute value) compared to the **best result** among all the baseline methods. Best results are bolded and second-best results are underlined. We also compared with GPF-plus (Fang et al., 2023) and Gprompt (Liu et al., 2023d), in Appendix Table 14 due to space limit.

Tasks	Dataset / Methods	Cora Acc	Cora F1	CiteSeer Acc	CiteSeer F1
Node Level	Supervised	52.83	47.73	63.91	64.82
	Fine-tune	56.37	55.04	64.87	66.42
	AIO (Sun et al., 2023a)	14.69	7.10	18.93	6.92
	ImprovedAIO	58.46	55.10	66.44	66.53
	Morpher	**61.26**	**62.36**	**68.20**	**68.56**
Edge Level	Supervised	51.78	50.62	52.14	50.81
	Fine-tune	52.50	51.00	52.50	51.12
	AIO (Sun et al., 2023a)	50.00	33.33	50.00	33.33
	ImprovedAIO	54.64	54.57	53.92	53.55
	Morpher	**55.71**	**55.05**	**55.35**	**55.05**

Table 2: Node-level, edge-level performance. Best results are bolded and second-best results are underlined.

5.2 Morpher Supports Multiple-level Tasks

Inherited from AIO, our ImprovedAIO and Morpher also support adaptation to downstream tasks at node-level and edge-level, because they can be reformulated into graph-level tasks. After reformulating node classification and task classification as graph classifications by inducing ego-graphs, we use GraphCL+GCN to pre-train the GNN and report the performance in Table 2. The results are consistent with graph level, where ImprovedAIO and Morpher outperform existing methods, with Morpher achieving slightly better performance than ImprovedAIO. Notably, as analyzed in Section 3, AIO is unstable during training, for example in Table 2 (Node level) and in certain cases of Table 1 (e.g., on MSRC_21C with GraphCL or with SimGRACE+GAT). The performance fluctuations observed are consistent and reflect the same underlying issue of AIO. In contrast, our proposed ImprovedAIO effectively addresses this issue.

5.3 Domain Transfer

We explore the potential of using Morpher for domain adaptation. From the previous experiments, we have demonstrated that our ImprovedAIO outperforms the original AIO. Therefore, in the subsequent pages, we focus on comparing Morpher with ImprovedAIO to avoid redundancy. We pre-train GNNs on ENZYMES or CiteSeer datasets, then test the classification performance on MUTAG and PubMed and report the results in Table 3. We unify the pre-train feature dimension with the downstream feature dimension by padding zeros or SVD reduction. From the results, Morpher demonstrates the best transferability, followed by ImprovedAIO.

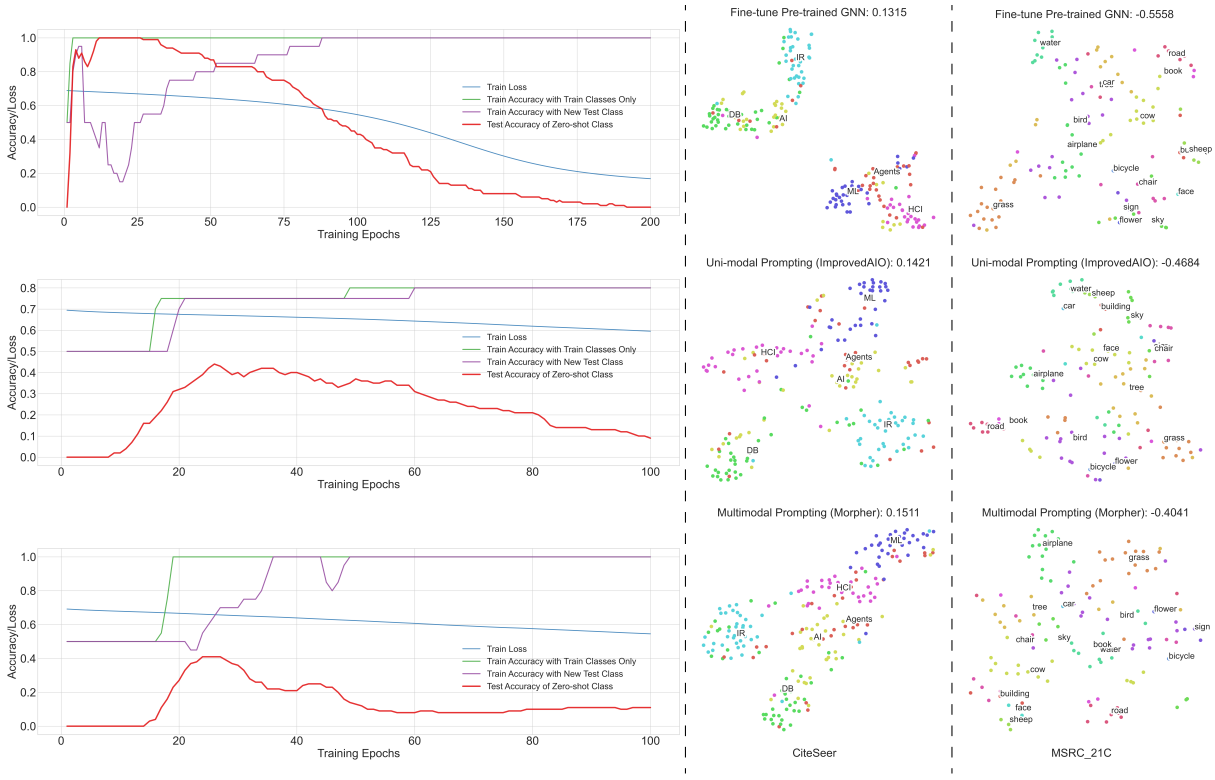


Figure 3: Results of novel class generalization (left); t-SNE embedding plots on CiteSeer, MSRC_21C (right). Train accuracy with train classes only is the accuracy of predicting the training graphs from the two training classes. Train accuracy with new test classes is the accuracy of predicting the training graphs from all three classes. Test Accuracy of zero-shot class is the accuracy of predicting the testing graphs from all three classes. Full-resolution figures can be found in Appendix D.

Target Domain		MUTAG		PubMed	
Target Task		graph-level		node-level	
Source	Methods	Acc	F1	Acc	F1
ENZYMES (graph-level)	Fine-tune	68.00	55.04	47.57	36.07
	ImprovedAIO	70.67	64.07	50.28	50.51
	Morpher	**72.67**	**73.29**	**54.42**	**53.96**
CiteSeer (node-level)	Fine-tune	71.33	62.19	48.71	40.66
	ImprovedAIO	74.00	73.76	52.57	51.29
	Morpher	**76.67**	**77.04**	**58.29**	**57.54**

Table 3: Domain Transfer Performance. Best results are bolded and second-best results are underlined.

5.4 Zero-shot Classification Prototype

An advantage of adapting pre-trained GNNs to the semantic embedding space is that GNNs might be empowered to "reasoning" in a CLIP style. Here, we conduct a novel experiment that generalizes GNN to an unseen class. Since no real-world data is available for this setting, we synthetically create three datasets, ZERO-Cora, ZERO-CiteSeer, and ZERO-PubMed, all from real-world connections. We aim to simulate a citation network with two research areas and an interdisciplinary research area in between. For each citation network, we randomly sample 120 nodes and induce their 2-hop ego-graphs and then replace the node features in 10 ego-graphs with $[1, 0]$ and another 10 ego-graphs with $[0, 1]$ to construct 20 training graph samples. For the remaining ego-graphs, we uniformly randomly replace the node features with $[1, 0]$ and $[0, 1]$ to construct 100 testing graph samples. We assign text labels of the first research area (e.g., "biology") to the $[1, 0]$ training graphs, the second research area (e.g., "informatics") to the $[0, 1]$ training graphs, and the interdisciplinary area (e.g., "bioinformatics") to the testing graphs. Intuitively, the nodes with feature $[1, 0]$ are papers in the first area, and nodes with feature $[0, 1]$ are in the second area, which makes the datasets rational.

For each dataset, using GraphCL+GCN, we pre-train GNNs on all graphs. Then, we train Morpher on the training graphs, only knowing the text labels of the two training classes. Since we do not have validation data in zero-shot learning, we report the results of each epoch in Figure 3 (left). We observe that, while Morpher quickly adapts the GNN to downstream training data, the CLIP-like framework can predict the graphs in the novel class

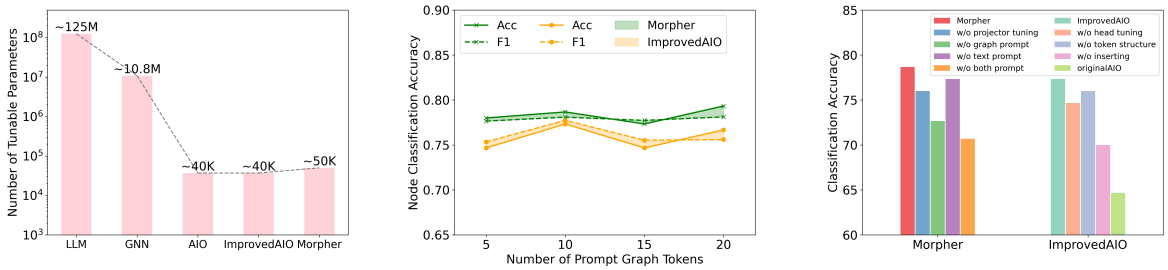


Figure 4: Efficiency comparison (left), parameter study (middle) and ablation study (right).

with good accuracy (red curve). Moreover, the training samples can be classified correctly from training and novel classes. Before the training overfits, there is a period when Morpher can distinguish all the graphs from the training and novel classes with high accuracy. Such zero-shot novel-class generalization ability validates Morpher's alignment between graph embeddings and text embeddings. When Morpher is trained on two classes of graphs with text labels of biology and informatics, a graph-in-the-middle will be classified as text-in-the-middle: bioinformatics, even if "bioinformatics" is an unseen label.

5.5 Efficiency and Embedding Analysis

Without fine-tuning the GNN or LLM, the prompt-based methods have better parameter efficiency. As shown in Figure 4 (left), our ImprovedAIO and Morpher require similar numbers of parameters with AIO (Sun et al., 2023a), which is 0.032% to 0.46% compared to either tune the LLM (RoBERTa) or GNN (GCN). Due to such parameter efficiency, our methods learn better graph representations given few-shot data. We visualize the graph embeddings of CiteSeer and MSRC_21C in Figure 3 and calculate the silhouette score, a metric for cluster quality ($\uparrow$) ranged in $[-1, 1]$. It turns out that our Morpher leads to better adaptation.

5.6 Hyperparameter and Ablation Study

We conduct the hyperparameter study by choosing and testing various numbers of graph prompt tokens for both ImprovedAIO and Morpher. The results are shown in Figure 4 (middle), from which we can observe that both methods are generally stable, and Morpher constantly outperforms ImprovedAIO under different choices. To verify the necessity of each component in our design, we compare Morpher and ImprovedAIO with multiple variants, respectively, and report the result in Figure 4 (right). We observe that removing any component would result in a performance drop.

We also conduct experiments to verify the effectiveness of our proposed Morpher with ELECTRA (Clark et al., 2020) and DistilBERT (Sanh et al., 2019) as the text encoder in Appendix B.3 due to space limitation. In general, Morpher is robust with respect to the language encoder. As for the robustness with respect to the pre-trained GNNs, we further conduct experiments using GNNs pre-trained from GraphMAE (Hou et al., 2022) and MVGRL (Hassani and Ahmadi, 2020). Due to the space limitation, the results are in Appendix B.4.

6 Related Work

GNN Pre-training. Recently, a surge of graph pre-training strategies have emerged (Hu et al., 2020a; Lu et al., 2021; Jing et al., 2021; Zhou et al., 2022a). The main idea of pre-trained graph models is to capture general graph information across different tasks and transfer this knowledge to the target task using techniques such as contrastive predictive coding (Khosla et al., 2020; Xia et al., 2022), context prediction (Hu et al., 2020b), and mutual information maximization (Sun et al., 2020). Different from these approaches, this paper aims to adapt pre-trained GNNs by leveraging multi-modal prompt learning techniques.

Graph Prompt Learning. Recent studies exploring prompt learning for GNNs mark a thriving research area (Sun et al., 2023b; Wu et al., 2023b). It is a promising way to adapt GNNs to downstream tasks through token-level (Fang et al., 2023; Tan et al., 2023; Chen et al., 2023a; Sun et al., 2022; Zhu et al., 2023) or graph-level (Sun et al., 2023a; Huang et al., 2023; Ge et al., 2023) prompting. Among all the existing methods, All-in-one (AIO) (Sun et al., 2023a) is the only algorithm to learn tunable graph prompts for multi-level downstream tasks (Table 6). Based on our improved AIO, we present a pioneer study to explore learning prompts

in multiple modalities simultaneously while keeping the pre-trained models frozen.

LLM on Graphs. LLMs' potential for graph-related tasks (Jin et al., 2023a) has been explored recently. The first category employs LLMs as pre-trained feature extractors to enhance GNNs (Duan et al., 2023; Chien et al., 2021; Zhu et al., 2021). The second category focuses on integrating graph structures directly into LLM architectures (Yang et al., 2021; Zhang et al., 2022b; Jin et al., 2023d,c). Despite these advancements, none of them have explored the collaboration between LLMs and GNNs with extremely weak text supervision and under graph prompt learning.

7 Conclusion

We present Morpher, the first multimodal prompt learning paradigm leveraging LLMs to semantically adapt pre-trained GNNs to downstream tasks with extremely weak text supervision. Addressing limitations of existing graph prompting techniques, we demonstrate through extensive experiments that Morpher excels in few-shot, multi-level tasks, and domain transfer. Notably, Morpher enables generalization to novel testing classes.

Acknowledgments

Research was supported in part by US DARPA INCAS Program No. HR0011-21-C0165 and BRIES Program No. HR0011-24-3-0325, National Science Foundation Award No. IIS-19-56151 and Award No. IIS-2117902, the Molecule Maker Lab Institute: An AI Research Institutes program supported by NSF under Award No. 2019897, the Institute for Geospatial Understanding through an Integrative Discovery Environment (I-GUIDE) by NSF under Award No. 2118329, and IBM-Illinois Discovery Accelerator Institute - a new model of an academic-industry partnership designed to increase access to technology education and skill development to spur breakthroughs in emerging areas of technology. The views and conclusions are those of the authors and should not be interpreted as representing the official policies of the funding agencies or the government.

8 Limitations

Our Multimodal graph prompt learning paradigm assumes the "pre-train + prompt" framework to learn transferable graph representations, yet there could be other paths to achieve graph-related foundation models. Also, graph prompt learning only works on the graph neural network architecture, and might not work for other architectures that are proposed in the future. Another limitation of this work is the requirement of language encoder. While RoBERTa is one of the most advanced encoder-only language models and can be considered an LLM with over 0.1B parameters, more recent LLMs such as Llama or Mistral cannot be used in Morpher because they are decoder-only LLMs and do not explicitly have an encoder.

References

Mengting Ai, Tianxin Wei, Yifan Chen, Zhichen Zeng, Ritchie Zhao, Girish Varatkar, Bita Darvish Rouhani, Xianfeng Tang, Hanghang Tong, and Jingrui He. 2025. Resmoe: Space-efficient compression of mixture of experts llms via residual restoration. *arXiv preprint arXiv:2503.06881*.

Yikun Ban, Jiaru Zou, Zihao Li, Yunzhe Qi, Dongqi Fu, Jian Kang, Hanghang Tong, and Jingrui He. 2024. Pagerank bandits for link prediction. In *NeurIPS*.

Wenxuan Bao, Zhichen Zeng, Zhining Liu, Hanghang Tong, and Jingrui He. Matcha: Mitigating graph structure shifts with test-time adaptation. In *The Thirteenth International Conference on Learning Representations*.

Karsten M. Borgwardt, Cheng Soon Ong, Stefan Schönauer, S. V. N. Vishwanathan, Alexander J. Smola, and Hans-Peter Kriegel. 2005. Protein function prediction via graph kernels. In *Proceedings Thirteenth International Conference on Intelligent Systems for Molecular Biology 2005, Detroit, MI, USA, 25-29 June 2005*, pages 47–56.

Giorgos Bouritsas, Fabrizio Frasca, Stefanos Zafeiriou, and Michael M. Bronstein. 2023. Improving graph neural network expressivity via subgraph isomorphism counting. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(1):657–668.

Mouxiang Chen, Zemin Liu, Chenghao Liu, Jundong Li, Qiheng Mao, and Jianling Sun. 2023a. ULTRA-DP: unifying graph pre-training with multi-task graph dual prompt. *CoRR*, abs/2310.14845.

Zefeng Chen, Wensheng Gan, Jiayang Wu, Kaixia Hu, and Hong Lin. 2023b. Data scarcity in recommendation systems: A survey. *CoRR*, abs/2312.10073.

Eli Chien, Wei-Cheng Chang, Cho-Jui Hsieh, Hsiang-Fu Yu, Jiong Zhang, Olgica Milenkovic, and Inderjit S Dhillon. 2021. Node feature extraction by self-supervised multi-scale neighborhood prediction. *arXiv preprint arXiv:2111.00064*.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: pre-training text encoders as discriminators rather than generators. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Keyu Duan, Qian Liu, Tat-Seng Chua, Shuicheng Yan, Wei Tsang Ooi, Qizhe Xie, and Junxian He. 2023. Simteg: A frustratingly simple approach improves textual graph learning. *arXiv preprint arXiv:2308.02565*.

Vijay Prakash Dwivedi, Chaitanya K. Joshi, Anh Tuan Luu, Thomas Laurent, Yoshua Bengio, and Xavier Bresson. 2023. Benchmarking graph neural networks. *J. Mach. Learn. Res.*, 24:43:1–43:48.

Taoran Fang, Yunchao Zhang, Yang Yang, Chunping Wang, and Lei Chen. 2023. Universal prompt tuning for graph neural networks. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Matthias Fey and Jan Eric Lenssen. 2019. Fast graph representation learning with pytorch geometric. *CoRR*, abs/1903.02428.

Fabrizio Frasca, Beatrice Bevilacqua, Michael M. Bronstein, and Haggai Maron. 2022. Understanding and extending subgraph gnns by rethinking their symmetries. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Dongqi Fu, Liri Fang, Zihao Li, Hanghang Tong, Vetle I. Torvik, and Jingrui He. 2024a. Parametric graph representations in the era of foundation models: A survey and position. *CoRR*, abs/2410.12126.

Dongqi Fu, Liri Fang, Ross Maciejewski, Vetle I. Torvik, and Jingrui He. 2022. Meta-learned metrics over multi-evolution temporal graphs. In *KDD*.

Dongqi Fu, Zhigang Hua, Yan Xie, Jin Fang, Si Zhang, Kaan Sancak, Hao Wu, Andrey Malevich, Jingrui He, and Bo Long. 2024b. Vcr-graphormer: A mini-batch graph transformer via virtual connections. In *ICLR*.

Dongqi Fu, Yada Zhu, Zhining Liu, Lecheng Zheng, Xiao Lin, Zihao Li, Liri Fang, Katherine Tieu, Onkar Bhardwaj, Kommy Weldemariam, Hanghang Tong, Hendrik F. Hamann, and Jingrui He. 2025. Climatebench-m: A multi-modal climate data benchmark with a simple generative method. *CoRR*, abs/2504.07394.

Dongqi Fu, Yada Zhu, Hanghang Tong, Kommy Weldemariam, Onkar Bhardwaj, and Jingrui He. 2024c. Generating fine-grained causality in climate time series data for forecasting and anomaly detection. *CoRR*.

Qingqing Ge, Zeyuan Zhao, Yiding Liu, Anfeng Cheng, Xiang Li, Shuaiqiang Wang, and Dawei Yin. 2023. Enhancing graph neural networks with structure-based prompt. *CoRR*, abs/2310.17394.

Lorenzo Giusti, Teodora Reu, Francesco Ceccarelli, Cristian Bodnar, and Pietro Liò. 2023. CIN++: enhancing topological message passing. *CoRR*, abs/2306.03561.

Gaoyang Guo, Chaokun Wang, Bencheng Yan, Yunkai Lou, Hao Feng, Junchao Zhu, Jun Chen, Fei He, and Philip S Yu. 2022. Learning adaptive node embeddings across graphs. *IEEE Transactions on Knowledge and Data Engineering*, 35(6):6028–6042.

Andrey Guzhov, Federico Raue, Jörn Hees, and Andreas Dengel. 2022. Audioclip: Extending clip to image, text and audio. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2022, Virtual and Singapore, 23-27 May 2022*, pages 976–980. IEEE.

Kaveh Hassani and Amir Hosein Khas Ahmadi. 2020. Contrastive multi-view representation learning on graphs. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 4116–4126. PMLR.

Xinrui He, Yikun Ban, Jiaru Zou, Tianxin Wei, Curtiss B. Cook, and Jingrui He. 2025a. Llm-forest: Ensemble learning of llms with graph-augmented prompts for data imputation. *Preprint*, arXiv:2410.21520.

Xinrui He, Shuo Liu, Jacky Keung, and Jingrui He. 2024. Co-clustering for federated recommender system. In *Proceedings of the ACM Web Conference 2024*, pages 3821–3832.

Xinyu He, Dongqi Fu, Hanghang Tong, Ross Maciejewski, and Jingrui He. 2025b. Temporal heterogeneous graph generation with privacy, utility, and efficiency. In *ICLR*.

Georg Hess, Adam Tonderski, Christoffer Petersson, Kalle Åström, and Lennart Svensson. 2024. Lidarclip or: How I learned to talk to point clouds. In *IEEE/CVF Winter Conference on Applications of Computer Vision, WACV 2024, Waikoloa, HI, USA, January 3-8, 2024*, pages 7423–7432. IEEE.

Zhenyu Hou, Xiao Liu, Yukuo Cen, Yuxiao Dong, Hongxia Yang, Chunjie Wang, and Jie Tang. 2022. Graphmae: Self-supervised masked graph autoencoders. In *KDD '22: The 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 14 - 18, 2022*, pages 594–604. ACM.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019.

Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR.

Weihua Hu, Bowen Liu, Joseph Gomes, Marinka Zitnik, Percy Liang, Vijay S. Pande, and Jure Leskovec. 2020a. Strategies for pre-training graph neural networks. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Ziniu Hu, Yuxiao Dong, Kuansan Wang, Kai-Wei Chang, and Yizhou Sun. 2020b. GPT-GNN: generative pre-training of graph neural networks. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 1857–1867. ACM.

Qian Huang, Hongyu Ren, Peng Chen, Gregor Krzmanc, Daniel Zeng, Percy Liang, and Jure Leskovec. 2023. PRODIGY: enabling in-context learning over graphs. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 4904–4916. PMLR.

Bowen Jin, Gang Liu, Chi Han, Meng Jiang, Heng Ji, and Jiawei Han. 2023a. Large language models on graphs: A comprehensive survey. *arXiv preprint arXiv:2312.02783*.

Bowen Jin, Wentao Zhang, Yu Zhang, Yu Meng, Xinyang Zhang, Qi Zhu, and Jiawei Han. 2023b. Patton: Language model pretraining on text-rich networks. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 7005–7020. Association for Computational Linguistics.

Bowen Jin, Wentao Zhang, Yu Zhang, Yu Meng, Xinyang Zhang, Qi Zhu, and Jiawei Han. 2023c. Patton: Language model pretraining on text-rich networks. *arXiv preprint arXiv:2305.12268*.

Bowen Jin, Yu Zhang, Qi Zhu, and Jiawei Han. 2023d. Heterformer: Transformer-based deep node representation learning on heterogeneous text-rich networks. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1020–1031.

Yihong Jin, Ze Yang, and Xinhe Xu. 2024. Scam detection for ethereum smart contracts: Leveraging graph representation learning for secure blockchain. *arXiv preprint arXiv:2412.12370*.

Baoyu Jing, Chanyoung Park, and Hanghang Tong. 2021. HDMI: high-order deep multiplex infomax. In *WWW '21: The Web Conference 2021, Virtual Event / Ljubljana, Slovenia, April 19-23, 2021*, pages 2414–2424. ACM / IW3C2.

Baoyu Jing, Yuchen Yan, Kaize Ding, Chanyoung Park, Yada Zhu, Huan Liu, and Hanghang Tong. 2024. Sterling: Synergistic representation learning on bipartite graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 12976–12984.

Baoyu Jing, Yuchen Yan, Yada Zhu, and Hanghang Tong. 2022. Coin: Co-cluster infomax for bipartite graphs. *arXiv preprint arXiv:2206.00006*.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *CoRR*, abs/2001.08361.

Muhammad Uzair Khattak, Hanoona Abdul Rasheed, Muhammad Maaz, Salman H. Khan, and Fahad Shahbaz Khan. 2023. Maple: Multi-modal prompt learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 19113–19122. IEEE.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Thomas N. Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *CoRR*, abs/1609.02907.

Lingkai Kong, Jiaming Cui, Haotian Sun, Yuchen Zhuang, B Aditya Prakash, and Chao Zhang. 2023. Autoregressive diffusion model for graph generation. In *International conference on machine learning*, pages 17391–17408. PMLR.

Devin Kreuzer, Dominique Beaini, William L. Hamilton, Vincent Létourneau, and Prudencio Tossou. 2021. Rethinking graph transformers with spectral attention. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 21618–21629.

Mengfei Lan, Lecheng Zheng, Shufan Ming, and Halil Kilicoglu. 2024. Multi-label sequential sentence

classification via large language model. In *Findings of the Association for Computational Linguistics: EMNLP 2024, Miami, Florida, USA, November 12-16, 2024*, pages 16086–16104. Association for Computational Linguistics.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 3045–3059. Association for Computational Linguistics.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven C. H. Hoi. 2022. BLIP: bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 12888–12900. PMLR.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 4582–4597. Association for Computational Linguistics.

Yichuan Li, Kaize Ding, and Kyumin Lee. 2023a. GRENADE: graph-centric language model for self-supervised representation learning on text-attributed graphs. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 2745–2757. Association for Computational Linguistics.

Zihao Li, Yuyi Ao, and Jingrui He. 2024a. Sphere: Expressive and interpretable knowledge graph embedding for set retrieval. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2024, Washington DC, USA, July 14-18, 2024*, pages 2629–2634. ACM.

Zihao Li, Dongqi Fu, Mengting Ai, and Jingrui He. 2025a. Apex^2: Adaptive and extreme summarization for personalized knowledge graphs. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining, V.1, KDD 2025, Toronto, ON, Canada, August 3-7, 2025*, pages 741–752. ACM.

Zihao Li, Dongqi Fu, and Jingrui He. 2023b. Everything evolves in personalized pagerank. In *Proceedings of the ACM Web Conference 2023, WWW 2023, Austin, TX, USA, 30 April 2023 - 4 May 2023*, pages 3342–3352. ACM.

Zihao Li, Xiao Lin, Zhining Liu, Jiaru Zou, Ziwei Wu, Lecheng Zheng, Dongqi Fu, Yada Zhu, Hendrik Hamann, Hanghang Tong, et al. 2025b. Language in the flow of time: Time-series-paired texts weaved into a unified temporal narrative. *arXiv preprint arXiv:2502.08942*.

Zilinghan Li, Shilan He, Ze Yang, Minseok Ryu, Kibaek Kim, and Ravi K. Madduri. 2024b. Advances in APPFL: A comprehensive and extensible federated learning framework. *CoRR*, abs/2409.11585.

Xiao Lin, Zhining Liu, Dongqi Fu, Ruizhong Qiu, and Hanghang Tong. 2024. Backtime: Backdoor attacks on multivariate time series forecasting. In *NeurIPS 2024*.

Xiao Lin, Zhining Liu, Ze Yang, Gaotang Li, Ruizhong Qiu, Shuke Wang, Hui Liu, Haotian Li, Sumit Keswani, Vishwa Pardeshi, et al. 2025a. Moralise: A structured benchmark for moral alignment in visual language models. *arXiv preprint arXiv:2505.14728*.

Xiao Lin, Zhichen Zeng, Tianxin Wei, Zhining Liu, Hanghang Tong, et al. 2025b. Cats: Mitigating correlation shift for multivariate time series classification. *arXiv preprint arXiv:2504.04283*.

Jiawei Liu, Cheng Yang, Zhiyuan Lu, Junze Chen, Yibo Li, Mengmei Zhang, Ting Bai, Yuan Fang, Lichao Sun, Philip S. Yu, and Chuan Shi. 2023a. Towards graph foundation models: A survey and beyond. *CoRR*, abs/2310.11829.

Shikun Liu, Tianchun Li, Yongbin Feng, Nhan Tran, Han Zhao, Qiang Qiu, and Pan Li. 2023b. Structural re-weighting improves graph domain adaptation. In *International Conference on Machine Learning*, pages 21778–21793. PMLR.

Xiaolong Liu, Zhichen Zeng, Xiaoyi Liu, Siyang Yuan, Weinan Song, Mengyue Hang, Yiqun Liu, Chaofei Yang, Donghyun Kim, Wen-Yen Chen, et al. 2024a. A collaborative ensemble framework for ctr prediction. *arXiv preprint arXiv:2411.13700*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Yixin Liu, Ming Jin, Shirui Pan, Chuan Zhou, Yu Zheng, Feng Xia, and Philip S. Yu. 2023c. Graph self-supervised learning: A survey. *IEEE Trans. Knowl. Data Eng.*, 35(6):5879–5900.

Zemin Liu, Xingtong Yu, Yuan Fang, and Xinming Zhang. 2023d. Graphprompt: Unifying pre-training and downstream tasks for graph neural networks. In *Proceedings of the ACM Web Conference 2023, WWW 2023, Austin, TX, USA, 30 April 2023 - 4 May 2023*, pages 417–428. ACM.

Zhining Liu, Rana Ali Amjad, Ravinarayana Adkathimar, Tianxin Wei, and Hanghang Tong. 2025. Selfelicit: Your language model secretly knows where is the relevant evidence. *arXiv preprint arXiv:2502.08767*.

Zhining Liu, Wei Cao, Zhifeng Gao, Jiang Bian, Hechang Chen, Yi Chang, and Tie-Yan Liu. 2020. Self-paced ensemble for highly imbalanced massive data classification. In *2020 IEEE 36th international conference on data engineering (ICDE)*, pages 841–852. IEEE.

Zhining Liu, Ruizhong Qiu, Zhichen Zeng, Hyunsik Yoo, David Zhou, Zhe Xu, Yada Zhu, Kommy Weldemariam, Jingrui He, and Hanghang Tong. 2024b. Class-imbalanced graph learning without class rebalancing. In *Forty-first International Conference on Machine Learning*.

Zhiyuan Liu, Sihang Li, Yanchen Luo, Hao Fei, Yixin Cao, Kenji Kawaguchi, Xiang Wang, and Tat-Seng Chua. 2023e. Molca: Molecular graph-language modeling with cross-modal projector and uni-modal adapter. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 15623–15638. Association for Computational Linguistics.

Yuanfu Lu, Xunqiang Jiang, Yuan Fang, and Chuan Shi. 2021. Learning to pre-train graph neural networks. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 4276–4284. AAAI Press.

Yizhen Luo, Kai Yang, Massimo Hong, Xing Yi Liu, and Zaiqing Nie. 2023. Molfm: A multimodal molecular foundation model. *CoRR*, abs/2307.09484.

Ali Madani, Bryan McCann, Nikhil Naik, Nitish Shirish Keskar, Namrata Anand, Raphael R. Eguchi, Po-Ssu Huang, and Richard Socher. 2020. Progen: Language modeling for protein generation. *CoRR*, abs/2004.03497.

Sahil Manchanda, Shubham Gupta, Sayan Ranu, and Srikanta J. Bedathur. 2023. Generative modeling of labeled graphs under data scarcity. In *Learning on Graphs Conference, 27-30 November 2023, Virtual Event*, volume 231 of *Proceedings of Machine Learning Research*, page 32. PMLR.

Christopher Morris, Nils M. Kriege, Franka Bause, Kristian Kersting, Petra Mutzel, and Marion Neumann. 2020. Tudataset: A collection of benchmark datasets for learning with graphs. *CoRR*, abs/2007.08663.

Marion Neumann, Roman Garnett, Christian Bauckhage, and Kristian Kersting. 2016. Propagation kernels: efficient graph kernels from propagated information. *Mach. Learn.*, 102(2):209–245.

Eric Nguyen, Michael Poli, Marjan Faizi, Armin W. Thomas, Michael Wornow, Callum Birch-Sykes, Stefano Massaroli, Aman Patel, Clayton M. Rabideau, Yoshua Bengio, Stefano Ermon, Christopher Ré, and Stephen Baccus. 2023. Hyenadna: Long-range genomic sequence modeling at single nucleotide resolution. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Sungjin Park, Seongsu Bae, Jiho Kim, Tackeun Kim, and Edward Choi. 2022. Graph-text multi-modal pretraining for medical representation learning. In *Conference on Health, Inference, and Learning, CHIL 2022, 7-8 April 2022, Virtual Event*, volume 174 of *Proceedings of Machine Learning Research*, pages 261–281. PMLR.

Jianing Qiu, Lin Li, Jiankai Sun, Jiachuan Peng, Peilun Shi, Ruiyang Zhang, Yinzhao Dong, Kyle Lam, Frank P.-W. Lo, Bo Xiao, Wu Yuan, Ningli Wang, Dong Xu, and Benny P. L. Lo. 2023. Large AI models in health informatics: Applications, challenges, and the future. *IEEE J. Biomed. Health Informatics*, 27(12):6074–6087.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

Shane Roach, Connie Ni, Alexei Kopylov, Tsai-Ching Lu, Jiejun Xu, Si Zhang, Boxin Du, Dawei Zhou, Jun Wu, Lihui Liu, et al. 2020. Canon: Complex analytics of network of networks for modeling adversarial activities. In *2020 IEEE International Conference on Big Data (Big Data)*, pages 1634–1643. IEEE.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108.

Yi-Jen Shih, Hsuan-Fu Wang, Heng-Jui Chang, Layne Berry, Hung-yi Lee, and David Harwath. 2022. Speechclip: Integrating speech with pre-trained vision and language model. In *IEEE Spoken Language Technology Workshop, SLT 2022, Doha, Qatar, January 9-12, 2023*, pages 715–722. IEEE.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S. Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Kumar Tanwani, Heather Cole-Lewis, Stephen Pfohl, Perry Payne, Martin Seneviratne, Paul Gamble, Chris Kelly, Nathaneal Schärli, Aakanksha Chowdhery, Philip Andrew Mansfield, Blaise Agüera y Arcas, Dale R. Webster, Gregory S. Corrado, Yossi Matias, Katherine Chou, Juraj Gottweis, Nenad Tomasev, Yun Liu, Alvin Rajkomar, Joelle K. Barral, Christopher Semturs, Alan Karthikesalingam, and Vivek Natarajan. 2022. Large language models encode clinical knowledge. *CoRR*, abs/2212.13138.

Geri Skenderi, Hang Li, Jiliang Tang, and Marco Cristani. 2023. Graph-level representation learning with joint-embedding predictive architectures. *CoRR*, abs/2309.16014.

Bing Su, Dazhao Du, Zhao Yang, Yujie Zhou, Jiangmeng Li, Anyi Rao, Hao Sun, Zhiwu Lu, and Ji-Rong Wen. 2022. A molecular multimodal foundation model associating molecule graphs with natural language. *CoRR*, abs/2209.05481.

Fan-Yun Sun, Jordan Hoffmann, Vikas Verma, and Jian Tang. 2020. Infograph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Mingchen Sun, Kaixiong Zhou, Xin He, Ying Wang, and Xin Wang. 2022. GPPT: graph pre-training and prompt tuning to generalize graph neural networks. In *KDD '22: The 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 14 - 18, 2022*, pages 1717–1727. ACM.

Xiangguo Sun, Hong Cheng, Jia Li, Bo Liu, and Jihong Guan. 2023a. All in one: Multi-task prompting for graph neural networks. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2023, Long Beach, CA, USA, August 6-10, 2023*, pages 2120–2131. ACM.

Xiangguo Sun, Jiawen Zhang, Xixi Wu, Hong Cheng, Yun Xiong, and Jia Li. 2023b. Graph prompt learning: A comprehensive survey and beyond. *CoRR*, abs/2311.16534.

Zhen Tan, Ruocheng Guo, Kaize Ding, and Huan Liu. 2023. Virtual node tuning for few-shot node classification. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2023, Long Beach, CA, USA, August 6-10, 2023*, pages 2177–2188. ACM.

Ross Taylor, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic. 2022. Galactica: A large language model for science. *CoRR*, abs/2211.09085.

Katherine Tieu, Dongqi Fu, Jun Wu, and Jingrui He. 2025. Invariant link selector for spatial-temporal out-of-distribution problem. In *The 28th International Conference on Artificial Intelligence and Statistics*.

Katherine Tieu, Dongqi Fu, Yada Zhu, Hendrik F. Hamann, and Jingrui He. 2024. Temporal graph neural tangent kernel with graphon-guaranteed. In *NeurIPS*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Clement Vignac, Igor Krawczuk, Antoine Siraudin, Bohan Wang, Volkan Cevher, and Pascal Frossard. 2022. Digress: Discrete denoising diffusion for graph generation. *arXiv preprint arXiv:2209.14734*.

Dingsu Wang, Yuchen Yan, Ruizhong Qiu, Yada Zhu, Kaiyu Guan, Andrew Margenot, and Hanghang Tong. 2023. Networked time series imputation via position-aware graph enhanced variational autoencoders. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2256–2268.

Limei Wang, Kaveh Hassani, Si Zhang, Dongqi Fu, Baichuan Yuan, Weilin Cong, Zhigang Hua, Hao Wu, Ning Yao, and Bo Long. 2025. Learning graph quantized tokenizers. In *ICLR*.

Tianxin Wei, Fuli Feng, Jiawei Chen, Ziwei Wu, Jinfeng Yi, and Xiangnan He. 2021. Model-agnostic counterfactual reasoning for eliminating popularity bias in recommender system. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pages 1791–1800.

Tianxin Wei, Bowen Jin, Ruirui Li, Hansi Zeng, Zhengyang Wang, Jianhui Sun, Qingyu Yin, Hanqing Lu, Suhang Wang, Jingrui He, et al. 2024. Towards unified multi-modal personalization: Large vision-language models for generative recommendation and beyond. *arXiv preprint arXiv:2403.10667*.

Tianxin Wei, Ziwei Wu, Ruirui Li, Ziniu Hu, Fuli Feng, Xiangnan He, Yizhou Sun, and Wei Wang. 2020. Fast adaptation for cold-start collaborative filtering with meta-learning. In *2020 IEEE International Conference on Data Mining (ICDM)*, pages 661–670. IEEE.

Tianxin Wei, Yuning You, Tianlong Chen, Yang Shen, Jingrui He, and Zhangyang Wang. 2022. Augmentations in hypergraph contrastive learning: Fabricated and generative. *Advances in neural information processing systems*, 35:1909–1922.

Zhihao Wen and Yuan Fang. 2023. Augmenting low-resource text classification with graph-grounded pre-training and prompting. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023*, pages 506–516. ACM.

Jun Wu, Jingrui He, and Elizabeth Ainsworth. 2023a. Non-iid transfer learning on graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 10342–10350.

Xuansheng Wu, Kaixiong Zhou, Mingchen Sun, Xin Wang, and Ninghao Liu. 2023b. A survey of graph prompting methods: Techniques, applications, and challenges. *CoRR*, abs/2303.07275.

Zhenqin Wu, Bharath Ramsundar, Evan N. Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S. Pappu, Karl Leswing, and Vijay S. Pande. 2017. Moleculenet: A benchmark for molecular machine learning. *CoRR*, abs/1703.00564.

Jun Xia, Lirong Wu, Jintao Chen, Bozhen Hu, and Stan Z. Li. 2022. Simgrace: A simple framework for graph contrastive learning without data augmentation. In *WWW '22: The ACM Web Conference 2022, Virtual Event, Lyon, France, April 25 - 29, 2022*, pages 1070–1079. ACM.

Haobo Xu, Yuchen Yan, Dingsu Wang, Zhe Xu, Zhichen Zeng, Tarek F Abdelzaher, Jiawei Han, and Hanghang Tong. Slog: An inductive spectral graph neural network beyond polynomial filter. In *Forty-first International Conference on Machine Learning*.

Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. 2021. Videoclip: Contrastive pre-training for zero-shot video-text understanding. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 6787–6800. Association for Computational Linguistics.

Zhe Xu, Ruizhong Qiu, Yuzhong Chen, Huiyuan Chen, Xiran Fan, Menghai Pan, Zhichen Zeng, Mahashweta Das, and Hanghang Tong. 2024. Discrete-state continuous-time diffusion for graph generation. *arXiv preprint arXiv:2405.11416*.

Hao Yan, Chaozhuo Li, Ruosong Long, Chao Yan, Jianan Zhao, Wenwen Zhuang, Jun Yin, Peiyan Zhang, Weihao Han, Hao Sun, Weiwei Deng, Qi Zhang, Lichao Sun, Xing Xie, and Senzhang Wang. 2023. A comprehensive study on text-attributed graphs: Benchmarking and rethinking. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Yuchen Yan, Yuzhong Chen, Huiyuan Chen, Xiaoting Li, Zhe Xu, Zhichen Zeng, Lihui Liu, Zhining Liu, and Hanghang Tong. 2024a. Thegcn: Temporal heterophilic graph convolutional network. *arXiv preprint arXiv:2412.16435*.

Yuchen Yan, Yongyi Hu, Qinghai Zhou, Lihui Liu, Zhichen Zeng, Yuzhong Chen, Menghai Pan, Huiyuan Chen, Mahashweta Das, and Hanghang Tong. 2024b. Pacer: Network embedding from positional to structural. In *Proceedings of the ACM Web Conference 2024*, pages 2485–2496.

Yuchen Yan, Yongyi Hu, Qinghai Zhou, Shurang Wu, Dingsu Wang, and Hanghang Tong. 2024c. Topological anonymous walk embedding: A new structural node embedding approach. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 2796–2806.

Yuchen Yan, Lihui Liu, Yikun Ban, Baoyu Jing, and Hanghang Tong. 2021. Dynamic knowledge graph alignment. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 4564–4572.

Yuchen Yan, Qinghai Zhou, Jinning Li, Tarek Abdelzaher, and Hanghang Tong. 2022. Dissecting cross-layer dependency inference on multi-layered inter-dependent networks. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 2341–2351.

Junhan Yang, Zheng Liu, Shitao Xiao, Chaozhuo Li, Defu Lian, Sanjay Agrawal, Amit Singh, Guangzhong Sun, and Xing Xie. 2021. Graphformers: Gnn-nested transformers for representation learning on textual graph. *Advances in Neural Information Processing Systems*, 34:28798–28810.

Zhilin Yang, William W. Cohen, and Ruslan Salakhutdinov. 2016. Revisiting semi-supervised learning with graph embeddings. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 40–48. JMLR.org.

Hyunsik Yoo, Zhichen Zeng, Jian Kang, Ruizhong Qiu, David Zhou, Zhining Liu, Fei Wang, Charlie Xu, Eunice Chan, and Hanghang Tong. 2024. Ensuring user-side fairness in dynamic recommender systems. In *Proceedings of the ACM Web Conference 2024*, pages 3667–3678.

Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. 2020. Graph contrastive learning with augmentations. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Qi Yu, Zhichen Zeng, Yuchen Yan, Lei Ying, R Srikant, and Hanghang Tong. 2025. Joint optimal transport and embedding for network alignment. In *Proceedings of the ACM on Web Conference 2025*, pages 2064–2075.

Seongjun Yun, Minbyul Jeong, Raehyun Kim, Jaewoo Kang, and Hyunwoo J. Kim. 2019. Graph transformer networks. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 11960–11970.

Zheni Zeng, Yuan Yao, Zhiyuan Liu, and Maosong Sun. 2022. A deep-learning system bridging molecule structure and biomedical text with comprehension comparable to human professionals. *Nature communications*, 13(1):862.

Zhichen Zeng, Boxin Du, Si Zhang, Yinglong Xia, Zhining Liu, and Hanghang Tong. 2024. Hierarchical multi-marginal optimal transport for network alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 16660–16668.

Zhichen Zeng, Ruizhong Qiu, Wenxuan Bao, Tianxin Wei, Xiao Lin, Yuchen Yan, Tarek F Abdelzaher, Jiawei Han, and Hanghang Tong. 2025. Pave your own path: Graph gradual domain adaptation on fused gromov-wasserstein geodesics. *arXiv preprint arXiv:2505.12709*.

Zhichen Zeng, Ruike Zhu, Yinglong Xia, Hanqing Zeng, and Hanghang Tong. 2023. Generative graph dictionary learning. In *International Conference on Machine Learning*, pages 40749–40769. PMLR.

Ningyu Zhang, Luoqiu Li, Xiang Chen, Shumin Deng, Zhen Bi, Chuanqi Tan, Fei Huang, and Huajun Chen. 2022a. Differentiable prompt makes pre-trained language models better few-shot learners. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Xikun Zhang, Antoine Bosselut, Michihiro Yasunaga, Hongyu Ren, Percy Liang, Christopher D Manning, and Jure Leskovec. 2022b. Greaselm: Graph reasoning enhanced language models for question answering. *arXiv preprint arXiv:2201.08860*.

Yuji Zhang, Sha Li, Cheng Qian, Jiateng Liu, Pengfei Yu, Chi Han, Yi R Fung, Kathleen McKeown, Chengxiang Zhai, Manling Li, et al. 2025. The law of knowledge overshadowing: Towards understanding, predicting, and preventing llm hallucination. *arXiv preprint arXiv:2502.16143*.

Haiteng Zhao, Shengchao Liu, Chang Ma, Hannan Xu, Jie Fu, Zhihong Deng, Lingpeng Kong, and Qi Liu. 2023. GIMLET: A unified graph-text model for instruction-based molecule zero-shot learning. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Huanjing Zhao, Beining Yang, Yukuo Cen, Junyu Ren, Chenhui Zhang, Yuxiao Dong, Evgeny Kharlamov, Shu Zhao, and Jie Tang. 2024. Pre-training and prompting for few-shot node classification on text-attributed graphs. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2024, Barcelona, Spain, August 25-29, 2024*, pages 4467–4478. ACM.

Lecheng Zheng, John R. Birge, Haiyue Wu, Yifang Zhang, and Jingrui He. 2025. Cluster aware graph anomaly detection. In *Proceedings of the ACM on Web Conference 2025, WWW 2025, Sydney, NSW, Australia, 28 April 2025- 2 May 2025*, pages 1771–1782. ACM.

Lecheng Zheng, Zhengzhang Chen, Jingrui He, and Haifeng Chen. 2024a. MULAN: multi-modal causal structure learning and root cause analysis for microservice systems. In *Proceedings of the ACM on Web Conference 2024, WWW 2024, Singapore, May 13-17, 2024*, pages 4107–4116. ACM.

Lecheng Zheng, Yu Cheng, Hongxia Yang, Nan Cao, and Jingrui He. 2021. Deep co-attention network for multi-view subspace learning. In *WWW '21: The Web Conference 2021, Virtual Event / Ljubljana, Slovenia, April 19-23, 2021*, pages 1528–1539. ACM / IW3C2.

Lecheng Zheng, Dongqi Fu, Ross Maciejewski, and Jingrui He. 2024b. Drgnn: Deep residual graph neural network with contrastive learning. *Transactions on Machine Learning Research*.

Lecheng Zheng, Baoyu Jing, Zihao Li, Hanghang Tong, and Jingrui He. 2024c. Heterogeneous contrastive learning for foundation models and beyond. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2024, Barcelona, Spain, August 25-29, 2024*, pages 6666–6676. ACM.

Lecheng Zheng, Baoyu Jing, Zihao Li, Zhichen Zeng, Tianxin Wei, Mengting Ai, Xinrui He, Lihui Liu, Dongqi Fu, Jiaxuan You, Hanghang Tong, and Jingrui He. 2024d. Pyg-ssl: A graph self-supervised learning toolkit. *CoRR*, abs/2412.21151.

Lecheng Zheng, Dawei Zhou, Hanghang Tong, Jiejun Xu, Yada Zhu, and Jingrui He. 2024e. Fairgen: Towards fair graph generation. In *40th IEEE International Conference on Data Engineering, ICDE 2024, Utrecht, The Netherlands, May 13-16, 2024*, pages 2285–2297. IEEE.

Lecheng Zheng, Yada Zhu, and Jingrui He. 2023. Fairness-aware multi-view clustering. In *Proceedings of the 2023 SIAM International Conference on Data Mining, SDM 2023, Minneapolis-St. Paul Twin Cities, MN, USA, April 27-29, 2023*, pages 856–864. SIAM.

Dawei Zhou, Lecheng Zheng, Dongqi Fu, Jiawei Han, and Jingrui He. 2022a. Mentorgnn: Deriving curriculum for pre-training gnns. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, October 17-21, 2022*, pages 2721–2731. ACM.

Dawei Zhou, Lecheng Zheng, Jiawei Han, and Jingrui He. 2020. A data-driven graph generative model for temporal interaction networks. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 401–411. ACM.

Dawei Zhou, Lecheng Zheng, Jiejun Xu, and Jingrui He. 2019. Misc-gan: A multi-scale generative model for graphs. *Frontiers Big Data*, 2:3.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. 2022b. Conditional prompt learning for vision-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 16795–16804. IEEE.

Jason Zhu, Yanling Cui, Yuming Liu, Hao Sun, Xue Li, Markus Pelger, Tianqi Yang, Liangjie Zhang, Ruofei Zhang, and Huasha Zhao. 2021. Textgnn: Improving text encoder via graph neural network in sponsored search. In *Proceedings of the Web Conference 2021*, pages 2848–2857.

Yun Zhu, Jianhao Guo, and Siliang Tang. 2023. SGL-PT: A strong graph learner with graph prompt tuning. *CoRR*, abs/2302.12449.

Jiaru Zou, Dongqi Fu, Sirui Chen, Xinrui He, Zihao Li, Yada Zhu, Jiawei Han, and Jingrui He. 2025. GTR: graph-table-rag for cross-table question answering. *CoRR*.

A Dataset Details

A.1 Dataset Statistics

Table 4 summarizes the statistics of the public real-world datasets, which we used in the few-shot experiments. For our synthetic datasets in the zero-shot prototype, we summarize their statistics in Table 5. As discussed in Section 5.4, the connections of our synthetic datasets are real, and we only replace the node feature by $[1, 0]$ and $[0, 1]$. The code to download the public data and the code to create synthetic data are provided in the supplementary materials.

A.2 Text Labels

When created, real-world graph datasets are usually coupled with textual meanings, but a common practice is to convert the textual meanings into numbers to create labels, which weakens the supervision of the graph data. For each real-world dataset, we convert the numerical labels back to text labels and feed into Morpher Language encoder through "[learnable text prompt] + [text label]". The mapping from the numbers to text labels for each dataset are provided as follows:

MUTAG. MUTAG is a dataset of nitroaromatic compounds, aiming to predict their mutagenicity on Salmonella typhimurium. Therefore, the mapping from numerical labels to text labels is: {0: non-mutagenic on Salmonella typhimurium, 1: mutagenic on Salmonella typhimurium}.

ENZYMES. ENZYMES aims to predict which subcategory each enzyme belongs to. The subcategories are: 0: oxidoreductases, 1: transferases, 2: hydrolases, 3: lyases, 4: isomerases, 5: ligases.

PROTEINS. PROTEINS is a dataset comprising proteins classified as either enzymes or non-enzymes. Therefore, the mapping is: 0: 'enzyme', 1: 'non-enzyme'.

MSRC_21C. Each graph in MSRC is constructed according to an image. The graph label is the image label. MSRC_21C contains 20 classes in MSRC, and "C" here means "Challenging" as the graphs(images) that are easy to classify has been filtered. The mapping from the numerical labels to text labels is: {0: building, 1: grass, 2: tree, 3: cow, 4: sheep, 5: sky, 6: airplane, 7: water, 8: face, 9: car, 10: bicycle, 11: flower, 12: sign, 13: bird, 14: book, 15: chair, 16: road}.

Cora. Cora is a citation network of papers in seven research areas. Each paper is labeled according to its corresponding research area. The mapping from the numerical labels to text labels is: {0: case based, 1: genetic algorithms, 2: neural networks, 3: probabilistic methods, 4: reinforcement learning, 5: rule learning, 6: theory}.

CiteSeer. CiteSeer is a citation network of papers, each labeled according to one of six research areas. The mapping from the numerical labels to text labels is: {0: Agents, 1: AI, 2: DB, 3: IR, 4: ML, 5: HCI}. We note that using abbreviations of the research area is not an issue because these abbreviations frequently appear, and the LLM tends to tokenize each of them as one token.

PubMed. PubMed is a collection of scientific publications from the PubMed database related to diabetes, classified into one of three categories. The mapping from the numerical labels to text labels is: {0: Diabetes Mellitus Experimental, 1: Diabetes Mellitus Type 1, 2: Diabetes Mellitus Type 2}.

Edge-level tasks. Cora, CiteSeer and PubMed can also be used as link prediction datasets. For link prediction, the mapping from the numerical labels to text labels is: {0: not connected, 1: connected}.

Synthetic Zero-shot Class Generalization Datasets. For ZERO-Cora, we synthetic three classes of ego-graph in a citation network. The first and second classes, respectively, have text labels "machine learning" and "theory", and the third (novel) class to generalize is "machine learning theory". For ZERO-CiteSeer, we synthetic three classes of ego-graph in a citation network. The first and second classes, respectively, have text labels "biology" and "informatics", and the third (novel) class to generalize is "bioinformatics". For ZERO-PubMed, we synthetic three classes of ego-graph in a citation network in the medical domain. The first and second classes, respectively, have text labels "cardiology" and "neurology", and the third (novel) class to generalize is "neurocardiology".

B Experiment Details

B.1 Reproducibility

Code. The code for the experiments is provided in the supplementary material with a well-written README file. We also provide the commands and instructions to run the code. The datasets used

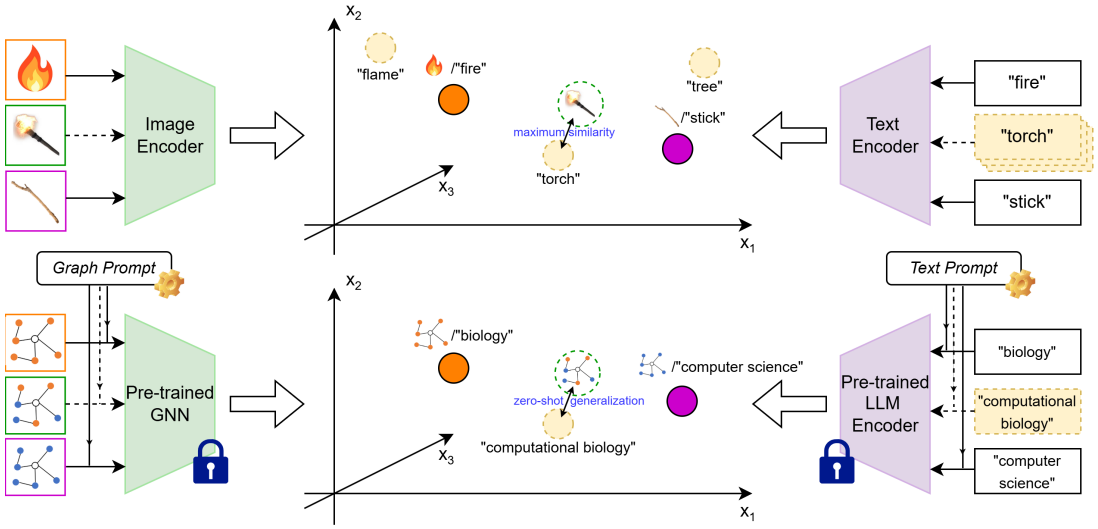


Figure 5: CLIP backbone (top) and this work (bottom). If a research paper cites many papers from biology and computer science, we realize this paper will likely be about computational biology, even if we do not know what exactly computational biology is. CLIP builds image encoders that learn such language dependency by Contrastive Language-Image Pre-training in the same embedding space according to Internet-scale data. However, text supervision is often extremely weak for graphs. This work leverages Multi-modal Prompt Learning for Graph Neural Networks that can effectively teach GNNs language dependency given few training samples with weak text supervision.

Table 4: Dataset statistics

Dataset	task level	# graphs	average # nodes	average # edges	# feature dimension	# classes	# shots per class	feature characteristic
MUTAG	graph	188	17.9	39.6	7	2	10	one-hot, sparse
ENZYMES	graph	600	32.6	124.3	3	6	10	one-hot, sparse
PROTEINS	graph	1113	39.1	145.6	3	2	10	one-hot, sparse
MSRC_21C	graph	209	40.28	96.60	22	17	1	one-hot, sparse
Cora	node, edge	1	2708	10556	1433	7	2 (node), 20 (edge)	sum 1, sparse
CiteSeer	node, edge	1	3327	9104	3703	6	2 (node), 20 (edge)	sum 1, sparse
PubMed	node	1	19,717	88648	500	3	10	TF-IDF value, dense

will be automatically downloaded when the code is executed.

Environment. We run all our experiments on a Windows 11 machine with a 13th Gen Intel(R) Core(TM) i9-13900H CPU, 64GB RAM, and an NVIDIA RTX A4500 GPU. We have also tested the code on a Linux machine with NVIDIA TITAN RTX GPU. All the code of our algorithms is written in Python. The Python version in our environment is 3.9.18. In order to run our code, one has to install some other common libraries, including PyTorch, PyTorch Geometric, pandas, numpy, scipy, etc. Please refer to our README in the code directory for downloading instructions.

We have optimized our code and tested that the space cost of the CPU memory is less than 16 GB, and the space cost of the graphics card is less than 6 GB. The execution time to run an experiment is less than 20 minutes on our machine.

B.2 Implementation Details

We provide the configuration files for the experiments to reproduce the results. We initialize the graph prompt using kaiming_initialization, and we initialize the text prompts through real token embeddings. We have tested multiple initializations, and they would not affect the overall results. Specifically, we initialize the text prompt for each dataset as follows.

MUTAG: “a graph with property”; ENZYMES: “this enzyme is”; PROTEINS: “this protein is”; MSRC_21C: “an image of”; Cora: “a paper of”; CiteSeer: “a paper of”; PubMed: “a paper of”; Edge tasks: “central nodes are”.

In our few-shot setting, we split the labeled data into training samples and validation samples at approximately 1:1. For all the parameters, we used the Adam optimizer, whose learning rate and weight decay are provided in the configuration files.

Table 5: Synthetic Zero-shot Class Generalization Dataset statistics

Dataset	# graphs	average # nodes	average # edges	#feature dimension	# classes	# shots per class
ZERO-Cora	120	8.41	10.38	2	2	10
ZERO-CiteSeer	120	10.03	21.31	2	2	10
ZERO-PubMed	120	20.33	41.75	2	2	10

Table 6: Comparison of graph prompts.

Method	prompt level	level of supported downstream tasks			learnable prompt	semantic
		node-level	edge-level	graph-level		
GPF-Plus (Fang et al., 2023)	token-level	√	×	×	√	×
Gprompt (Liu et al., 2023d)	token-level	√	×	√	√	×
VNT (Tan et al., 2023)	token-level	×	×	√	√	×
ULTRA-DP (Chen et al., 2023a)	token-level	√	×	×	√	×
GPPT (Sun et al., 2022)	token-level	√	×	×	√	×
SGL-PT (Zhu et al., 2023)	token-level	√	×	×	√	×
SAP (Ge et al., 2023)	graph-level	√	×	√	√	×
PRODIGY (Huang et al., 2023)	graph-level	√	√	√	×	×
All-in-one (AIO) (Sun et al., 2023a)	graph-level	√	√	√	√	×
ImprovedAIO (ours)	graph-level	√	√	√	√	×
Morpher (ours)	graph-level	√	√	√	√	√

B.3 Experiment with ELECTRA and DistilBERT

On the LLM pre-training side, RoBERTa is one of the most advanced encoder-only LLMs until now, and we have demonstrated the effectiveness with RoBERTa serving on the LLM side in the Morpher paradigm. Additionally, we conducted experiments with ELECTRA (Clark et al., 2020) and DistilBERT (Sanh et al., 2019). Using these two LLMs, Morpher can also achieve comparable performances to RoBERTa. The results are shown in Table 8 and in Table 7.

In general, using ELECTRA and DistilBERT results in similar performance compared to using RoBERTa, showing the robustness of Morpher with respect to the language encoder.

B.4 Experiment with GNNs trained using GraphMAE and MVGRL

In the main pages, we used GraphCL and SimGRACE to show that Morpher achieves better performance given a pre-trained GNN. Additionally, to further verify the robustness of Morpher over the pre-train method, we conducted experiments on the pre-trained GNNs using GraphMAE (Hou et al., 2022) and MVGRL (Hassani and Ahmadi, 2020). We use GCN as the GNN backbone and RoBERTa as the LLM encoder, and the results are reported in Table 9 and Table 10.

Using GraphMAE or MVGRL to pre-train the GNN, the trend of performance is similar to that when using GraphCL or SimGRACE. Also, ImprovedAIO and Morpher's performance is similar to that of pre-trained GNNs from GraphCL or SimGRACE and can still significantly outperform the pre-train + fine-tune baseline, showing the robustness of Morpher with respect to the pre-training strategy.

B.5 Morpher on MolecureNet with More Text Supervision

The relatively abundant labeled graph-text pairs in medical and biological domains have significantly accelerated research into training large models specifically tailored for these areas (Qiu et al., 2023). For example, (Park et al., 2022) pretrains the multimodal medical model MedGTX for Electronic Health Records (EHR) using the open-source EHR dataset MIMIC-III. Similarly, ProGen (Madani et al., 2020) trains a 1.2B-parameter language model on approximately 280M protein sequences. HyenaDNA (Nguyen et al., 2023) pretrains a genomic foundation model on the human reference genome with context lengths of up to 1 million tokens. Furthermore, (Singhal et al., 2022) introduces the MultiMedQA benchmark and per-

Table 7: Few-shot graph classification performance (%) of Morpher with ELECTRA (Clark et al., 2020) as language encoder. Other experiment settings are identical to the main experiment.

GNN pretraining	MUTAG		ENZYMES		PROTEINS		MSRC_21C	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1
GraphCL + GCN	78.00	78.17	20.41	15.79	67.38	65.66	43.42	47.19
GraphCL + GAT	76.67	75.75	20.41	11.37	66.26	65.66	44.57	49.01
GraphCL + GT	76.67	77.04	19.16	14.68	73.06	72.70	42.28	44.09
SimGRACE + GCN	70.00	70.99	19.79	12.41	68.96	67.77	45.71	48.44
SimGRACE + GAT	77.33	77.51	18.12	13.31	68.96	67.78	44.00	49.43
SimGRACE + GT	72.67	73.55	18.33	15.76	70.18	70.28	41.14	44.50

Table 8: Few-shot graph classification performance (%) of Morpher with DistilBERT (Sanh et al., 2019) as language encoder. Other experiment settings are identical to the main experiment.

GNN pretraining	MUTAG		ENZYMES		PROTEINS		MSRC_21C	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1
GraphCL + GCN	78.00	78.61	20.62	10.00	66.44	65.54	43.42	47.98
GraphCL + GAT	77.33	75.64	21.25	15.87	70.59	68.25	45.14	48.82
GraphCL + GT	74.67	75.20	19.58	14.96	70.27	70.55	44.57	47.28
SimGRACE + GCN	69.33	70.36	20.62	18.82	66.91	66.41	45.14	47.77
SimGRACE + GAT	77.33	76.90	18.54	14.44	67.56	65.08	45.71	44.36
SimGRACE + GT	72.67	73.52	17.91	11.06	70.55	70.36	45.14	44.01

forms instruction-tuning on the 540-billion parameter Flan-PaLM model within the clinical domain.

We demonstrate that, though not specifically designed for any downstream applications, the Morpher framework has the potential to be used in various tasks where there is more text supervision compared to previous experiments. As for a case study, We use bace (inhibitors of human beta-secretase), tox21 (toxicology in the 21st century) and hiv (inhibit HIV replication) from MolecureNet (Wu et al., 2017). These three datasets have 1513, 7831, and 41127 graphs to classify, respectively. In these datasets, each graph label is associated with a text description. The tasks on bace and hiv are bioactivity prediction and the task on tox21 is toxicity prediction. To adopt Morpher, we use GraphCL to pre-train the GAT model and initialize the text prompts and text labels using those from GIMLET (Zhao et al., 2023).

KVPLM (Zeng et al., 2022), MoMu (Su et al., 2022), Galactica-1.3B (Taylor et al., 2022) are zero-shot predictors for the three tasks; GIMLET-64M-50-shots is the GIMLET (Zhao et al., 2023) model fine-tuned on 50 additional training samples[2]; GAT-1M-fully-supervised uses all the training data to train a GAT. Our Morpher-k-shots uses only k training samples. From the results, first, using only 10 training samples, Morpher can outperform the zero-shot baselines KVPLM, MoMu, and Galactica-1.3B. Second, using only 50 shots, Morpher can achieve similar performance with the fully supervised GAT. Third, using the same amount of few-shot data (50 shots), Morpher-50 outperforms GIMLET-64M-50-shots on tox21 and hiv, the two largest datasets among the three. This means our graph-text multi-modal prompt learning, with much fewer learnable parameters ($\sim 50K$), is more sample-efficient than fine-tuning language model encoder.

[2]the performance of GIMLET and other baselines are directly from the GIMLET paper (Zhao et al., 2023).

B.6 Full Table for Few-shot Experiment

Due to the page limitation of the main pages, Table 14 shows the full table for the few-shot experiment.

B.7 Full Table for Domain Transfer

In the main pages, through the experiments presented in Tables 1 and 2, we have already demonstrated that fine-tune outperforms supervised, and ImprovedAIO significantly outperforms the original AIO. Therefore, in Table 3, we focus on comparing Morpher, ImprovedAIO and fine-tune methods to avoid redundancy. Here, we report the full comparison with the performance of supervised and AIO baselines. in Table 13. The result is consistent that fine-tune outperforms supervised, and our improvedAIO outperforms AIO.

We would like to acknowledge that AIO demonstrates state-of-the-art performance under the ex-

Table 9: Few-shot graph classification performance (%) of Morpher with the GNN pre-trained by GraphMAE (Hou et al., 2022). Other experiment settings are identical to the main experiment.

GNN pretraining	MUTAG		ENZYMES		PROTEINS		MSRC_21C	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1
Pre-train + Fine-tune	71.33	71.41	16.04	12.14	65.86	65.22	39.42	40.20
ImprovedAIO	76.67	76.95	19.58	12.59	66.36	65.30	42.28	46.81
Morpher	78.67	78.67	20.20	16.95	67.38	65.66	45.71	48.49

Table 10: Few-shot graph classification performance (%) of Morpher with the GNN pre-trained by MVGRL (Hassani and Ahmadi, 2020). Other experiment settings are identical to the main experiment.

GNN pretraining	MUTAG		ENZYMES		PROTEINS		MSRC_21C	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1
Pre-train + Fine-tune	68.67	69.46	16.45	10.16	65.15	64.71	38.85	40.56
ImprovedAIO	74.67	74.00	18.13	15.57	66.54	65.90	42.85	46.66
Morpher	78.00	77.81	18.96	14.97	67.56	66.79	44.57	48.67

perimental settings reported in its original paper (Sun et al., 2023a). However, our evaluation is conducted under more challenging conditions, particularly with fewer training and validation samples. In these settings, we observe that AIO's performance degrades, suggesting there is still room for improvement. To address this, we propose ImprovedAIO, which extends AIO's design to better handle these harder scenarios. Our goal is not to critique prior work, but to help advance the field of graph prompt learning and multimodal alignment by pushing toward more robust and generalizable solutions.

C Further Discussions

C.1 Unstable Training of Current Graph Prompt Design

As analyzed in Section 3, the current graph prompt design suffers from unstable training due to the imbalance of inner-connections and cross-connections because for any node i and token p_j, the dot products $\mathbf{X}(i,:)\mathbf{P}_\theta^g(j,:)^\top$ is close to 0, leading to ineffective prompt-token interactions. A potential solution to ensure $\mathbf{X}(i,:)\mathbf{P}_\theta^g(j,:)^\top$ has a larger nonzero value is to initialize prompt tokens with higher variance. Through further analysis and experimental validation, We tend to believe that it fails to address the root cause.

First, initializing parameters with high variance can introduce additional challenges during training, such as unstable gradients, over-reliance on the initial high-variance parameters, and ineffective weight regularization. Specifically, in the prompt learning setting, high-variance initialization of prompts may cause the training process to overly focus on the prompt embeddings, thereby overshadowing the information encoded in the input graph and hindering the model's ability to learn meaningful representations from the input graph structure.

Second, even with high-variance initialization, the computation of $\sigma(\mathbf{X}(i,:)\mathbf{P}_\theta^g(j,:)^\top)$ would still result in approximately half of the cross-connections being established. While this is an improvement over the original AIO, where nearly all cross-connections are formed, it would still lead to the prompted graph representations being overly similar. As a result, the task head cannot effectively learn to distinguish between different graphs.

To further validate this analysis, we conducted additional experiments using high-variance initialization in the original AIO method. These experiments were performed in the few-shot learning setting (Table 12) using GraphCL pretraining and a GAT encoder.

As shown in Table 12, increasing the initialization variance does not consistently improve performance. In some cases, it even leads to degradation, likely due to training instability. Furthermore, the overall performance of AIO remains significantly lower than that of our ImprovedAIO and Morpher, demonstrating that high-variance initialization is not a sufficient solution to the dense cross-connection issue.

Our ImprovedAIO effectively addresses the training issue of graph prompt learning without

Table 11: AUC-ROC (↑) on MolecureNet (bace, tox21, hiv). Morpher-K denotes K shots.

Dataset	KVPLM	MoMu	Galactica-1.3B	GIMLET-64M-50-shots	GAT-1M-supervised	Morpher-10	Morpher-20	Morpher-50
bace	0.5126	0.6656	0.5648	0.729	0.697	0.6231	0.6513	0.6858
tox21	0.4917	0.5757	0.4946	0.652	0.754	0.6769	0.7275	0.7459
hiv	0.6120	0.5026	0.3385	0.721	0.729	0.5742	0.7034	0.7283

	MUTAG	ENZYMES	PROTEINS	MSRC_21C
default variance	64.67	17.50	59.21	14.37
3× variance	64.67	17.50	61.79	13.17
5× variance	68.00	17.70	59.21	11.37
10× variance	67.33	16.45	58.65	17.96

Table 12: Accuracy results of high-variance initialization experiments in few-shot learning.

Target Domain		MUTAG		PubMed	
Target Task		graph-level		node-level	
Source	Methods	Acc	F1	Acc	F1
ENZYMES (graph-level)	Supervised	66.00	56.67	47.57	36.07
	Fine-tune	68.00	55.04	47.57	36.07
	AIO	64.00	54.50	44.85	34.13
	ImprovedAIO	<u>70.67</u>	<u>64.07</u>	<u>50.28</u>	<u>50.51</u>
	Morpher	**72.67**	**73.29**	**54.42**	**53.96**
CiteSeer (node-level)	Supervised	66.00	56.67	47.57	36.07
	Fine-tune	71.33	62.19	48.71	40.66
	AIO	65.33	57.20	45.71	34.39
	ImprovedAIO	<u>74.00</u>	<u>73.76</u>	<u>52.57</u>	<u>51.29</u>
	Morpher	**76.67**	**77.04**	**58.29**	**57.54**

Table 13: Domain Transfer Performance. Best results are bolded and second-best results are underlined.

introducing any new thresholding hyperparameters. Due to our pruning mechanism, ImprovedAIO is less sensitive to the choice of δ_{cross}. Specifically, a smaller δ_{cross} can safely be used in our framework, as it allows more cross edges to be initially introduced without affecting the final set of cross edges after pruning.

C.2 Scalability of GNNs and LLMs

The scalability of our proposed method across different sizes of GNNs and LLMs is an important consideration. However, the primary focus of this work is to introduce multimodal prompt learning for GNNs and validate the effectiveness of our novel paradigm, Morpher.

Regarding GNN scalability, the scale of a GNN is highly dependent on the total available samples for self-supervised pretraining. In our experiments, we employed GNNs with up to 10M parameters, which is already relatively large for the datasets we used. Notably, many recent works achieving state-of-the-art performance on similar classification tasks use GNNs of comparable or smaller scale (Skenderi et al., 2023; Giusti et al., 2023; Kreuzer et al., 2021; Frasca et al., 2022; Bouritsas et al., 2023).

For LLM scalability, as mentioned in the limitation section of our paper, our method requires a language encoder and has not yet been integrated with very large decoder-only LLMs. Nonetheless, our experiments with RoBERTa (0.1B parameters), ELECTRA (0.3B parameters), and DistilBERT (0.06B parameters) in our ablation study demonstrate that Morpher is robust across a variety of language encoders with different designs and scales.

Our contributions in this paper include improving the design of graph prompts and introducing a multimodal prompt learning paradigm for GNNs, tailored to real-world scenarios where text supervision is extremely weak. Through extensive experiments, we have demonstrated that our proposed methods outperform state-of-the-art baselines, establishing a robust and effective framework for multimodal learning. While scalability across larger GNNs and LLMs is an important direction, it is beyond the scope of this work and we highlight scalability as a potential future research to extend Morpher to even larger models.

C.3 Model and Data Scaling Laws

Scaling laws and emergence ability have attracted much research interest recently (Kaplan et al., 2020). To this end, we conduct additional experiments regarding the data scaling capability and model scaling capability using GraphCL+GCN on the MUTAG dataset.

Scaling law with respect to model size. Since our Morpher does not require the pretraining of large language models (LLMs), we report the performance using various LLMs of different sizes in Table 15. Next, we pre-train GNNs of varying sizes by adjusting parameters such as the hidden dimension, while keeping the language model fixed to RoBERTa, and report the results in Figure 16. Based on these results, we did not observe significant scaling laws with respect to the size of the LLM or GNN. We hypothesize that this may be because the sizes of the LLM and GNN have not yet reached the threshold for exhibiting emergent capa-

Training schemes	GNN pretraining	MUTAG		ENZYMES		PROTEINS		MSRC_21C	
		Acc	F1	Acc	F1	Acc	F1	Acc	F1
Supervised	N/A + GCN	66.00	66.67	16.67	8.68	65.89	60.77	38.85	35.32
	N/A + GAT	66.00	65.69	16.45	4.65	64.75	64.08	41.14	39.86
	N/A + GT	66.66	66.26	15.62	4.22	62.81	57.12	38.28	41.62
Pre-train + Fine-tune	GraphCL+GCN	70.00	70.23	17.91	11.82	65.89	61.23	40.00	43.89
	GraphCL+GAT	70.00	69.73	17.91	10.46	65.16	63.92	44.57	45.74
	GraphCL+GT	68.00	67.81	17.70	8.99	63.28	56.41	41.71	43.73
	SimGRACE+GCN	66.67	67.27	17.29	8.78	66.82	64.70	40.57	43.84
	SimGRACE+GAT	70.67	69.10	16.87	7.18	65.42	63.65	42.85	42.37
	SimGRACE+GT	69.33	69.77	16.24	6.08	65.98	62.31	39.42	40.78
AIO (Sun et al., 2023a)	GraphCL+GCN	64.67	39.27	17.50	4.97	61.35	44.93	3.59	10.09
	GraphCL+GAT	64.67	39.27	17.50	4.97	59.21	37.19	14.37	3.11
	GraphCL+GT	73.33	72.06	18.33	9.09	40.79	28.97	17.96	8.30
	SimGRACE+GCN	64.67	39.27	16.04	4.61	67.42	60.87	34.73	18.16
	SimGRACE+GAT	64.67	39.27	16.04	4.61	59.21	37.19	7.78	1.79
	SimGRACE+GT	36.00	27.26	17.50	8.15	50.56	49.34	32.34	15.13
GPF-plus (Fang et al., 2023)	GraphCL+GCN	68.67	67.27	16.88	15.48	64.75	61.45	47.42	29.02
	GraphCL+GAT	68.67	62.84	16.45	13.23	65.89	60.07	47.42	26.28
	GraphCL+GT	69.33	67.87	18.12	15.56	59.66	37.37	41.71	21.35
	SimGRACE+GCN	65.33	39.52	18.96	15.83	65.16	58.80	45.71	23.32
	SimGRACE+GAT	69.33	66.72	18.54	12.58	63.28	53.50	42.85	21.40
	SimGRACE+GT	70.00	67.31	17.91	14.69	64.83	52.97	34.13	20.13
Gprompt (Liu et al., 2023d)	GraphCL+GCN	73.33	66.93	17.91	8.44	61.01	60.01	1.80	0.21
	GraphCL+GAT	64.67	62.63	17.08	14.18	50.56	50.55	1.80	0.22
	GraphCL+GT	70.67	70.02	17.91	9.64	63.28	58.65	1.80	0.21
	SimGRACE+GCN	65.33	39.52	17.29	14.48	52.70	52.68	1.80	0.21
	SimGRACE+GAT	67.33	65.88	16.25	11.31	59.10	58.72	1.80	0.21
	SimGRACE+GT	73.33	67.84	16.87	13.54	64.75	62.37	1.80	0.223
Improved AIO (Ours)	GraphCL+GCN	77.33	77.74	18.13	11.98	65.89	65.97	42.85	45.91
	GraphCL+GAT	74.67	75.51	18.33	11.26	65.76	66.05	46.85	51.39
	GraphCL+GT	74.67	74.67	19.16	9.04	68.12	68.18	42.85	43.54
	SimGRACE+GCN	68.00	69.01	17.91	9.02	66.82	66.40	44.57	49.24
	SimGRACE+GAT	77.33	77.20	18.75	9.39	66.91	65.49	45.14	42.31
	SimGRACE+GT	71.33	72.06	18.95	11.25	68.59	68.84	40.57	42.82
Morpher (Ours)	GraphCL+GCN	78.67	78.09	20.41	15.20	67.47	66.40	45.14	49.62
	GraphCL+GAT	79.33	79.15	23.12	18.01	70.89	70.30	50.85	54.48
	GraphCL+GT	76.00	76.51	19.58	13.28	73.53	72.48	45.71	48.41
	SimGRACE+GCN	69.33	70.27	19.79	14.94	67.10	66.15	45.71	51.24
	SimGRACE+GAT	78.00	77.65	20.21	16.27	68.12	67.26	45.71	51.13
	SimGRACE+GT	74.00	74.84	19.16	14.29	71.76	71.75	44.00	48.16
IMP of ImprovedAIO		2.00 ↑	5.01 ↑	0.52 ↑	4.41 ↓	2.01 ↑	4.37 ↑	0.28 ↓	2.50 ↑
IMP of Morpher		4.00 ↑	6.73 ↑	2.36 ↑	0.60 ↑	4.81 ↑	6.61 ↑	2.66 ↑	7.14 ↑

Table 14: Few-shot graph classification performance (%). IMP (%): the average improvement (absolute value) compared to the **best result** among all the baseline methods.

bilities. Further investigation into this phenomenon is an interesting direction for future work.

Language Model Size	MUTAG Acc	MUTAG F1
DistilBERT 0.06B	78.00	78.61
RoBERTa 0.1B	78.67	78.09
ELECTRA 0.3B	78,00	78.17

Table 15: Scaling law with respect to language model size.

GCN Model Size	MUTAG Acc	MUTAG F1
1M	78.00	78.56
3M	78.67	78.97
10M	78.67	78.09

Table 16: Scaling law with respect to GCN model size.

Scaling law with respect to data size We conducted additional studies by adjusting either the pre-training data size or the downstream few-shot data size. To evaluate the effect of pre-training data size, we used GraphCL+GCN on the MUTAG dataset, randomly selecting k% of the samples for GNN pre-training. The results are presented in Table 17. The results suggest that increasing the size of either the pre-training data or the downstream few-shot data generally improves the performance of our Morpher. This observation is consistent with typical data scaling laws. While the current findings provide valuable insights, further investigation is required to explore the detailed effects of data scaling, which we leave for future work.

Pretrain Data Ratio	MUTAG Acc	MUTAG F1
10%	72.00	68.80
30%	75.33	75.11
100%	78.67	78.09

Table 17: Scaling law with respect to data size.

C.4 Tunable pre-trained GNN/LLM Scenario

Generally, in NLP, prompt or prefix tuning is often employed as a parameter-efficient alternative to fine-tuning, especially when fine-tuning the entire model is computationally expensive. While prompt tuning and fine-tuning are not technically incompatible, they are typically not used simultaneously, as the goal of prompt tuning is to achieve strong performance without the need to update the entire model.

That being said, making the pre-trained GNN tunable could potentially further enhance performance in specific scenarios. However, this would come at the cost of increased computational complexity and resource requirements, which goes against the motivation of our proposed method. We designed Morpher with efficiency in mind, ensuring that it achieves strong performance without requiring extensive updates to the pre-trained GNN. Furthermore, given the weak text supervision in our setting, increasing the parameter space by making the GNN tunable could reduce the sample efficiency of prompt tuning, potentially hindering the model's ability to learn effectively from limited supervision.

Therefore, while the scenario where the GNN—and potentially the language model—are also tunable is an interesting direction, it falls beyond the scope of this paper. Nonetheless, we acknowledge this as an open question that warrants further exploration in future research, particularly to better understand the trade-offs between parameter efficiency and model expressiveness.

C.5 Broader Impact and Future Directions

Learning on graphs has been a long-standing goal in the machine learning community, evolving from pattern-based mining (Li et al., 2025a) to modern graph neural network models (Zheng et al., 2024b; Wang et al., 2025), with broad applications in social network analysis (Li et al., 2023b; He et al., 2024), natural sciences (Fu et al., 2025), and beyond (Li et al., 2024a; Fu et al., 2024a; Li et al., 2024b; Lin et al., 2025a; Jin et al., 2024). In addition to advancing graph-language alignment and graph prompt learning, we hope this work can also inspire future research in the following directions.

Distribution Shift in Graph Data. Real-world graph data often undergoes distribution shifts in both node features and graph structures, which can severely degrade the performance of GNNs. Before foundational GNNs, to address this challenge, graph domain adaptation aims to adapt a pretrained GNN model to a target graph via either model adaptation (Bao et al.; Wu et al., 2023a; Guo et al., 2022; Tieu et al., 2025) or data adaptation (Liu et al., 2023b; Wei et al., 2022; Lin et al., 2024; Zeng et al., 2025; He et al., 2025b).

Graph Foundation Models versus Domain Specific GNNs. In the era of big data and AI (Liu et al., 2020; Wei et al., 2020; Liu et al., 2025; Fu et al., 2024b; Zou et al., 2025; Zhang et al., 2025),

graph foundation models play an important role in many applications, such as network alignment (Yan et al., 2021, 2022; Zeng et al., 2024; Yu et al., 2025), spectral graph signal processing (Xu et al.; Liu et al., 2024b), anomaly detection (Zheng et al., 2025, 2024a), multi-layered network embedding (Yan et al., 2024a,b,c; Jing et al., 2022, 2024), information retrieval (Wei et al., 2021; Yoo et al., 2024; Liu et al., 2024a) and time series analysis (Roach et al., 2020; Fu et al., 2022; Wang et al., 2023; Lin et al., 2025b; Ban et al., 2024; Tieu et al., 2024; Fu et al., 2024c). However, as noted earlier, graph foundation models still face challenges under distribution shifts. In practice, many real-world applications continue to rely on domain-specific GNN solutions when sufficient training data is available. While this work introduces a method for training graph foundation models under extremely weak text supervision, we still recommend evaluating domain-specific GNNs as a baseline when developing real-world systems.

Multimodal Learning with Graphs. Multimodality learning (Zheng et al., 2021, 2023; Li et al., 2025b; Wei et al., 2024) has been studied for decades. Inspired by the success of large language model (Radford et al., 2021; Lan et al., 2024; Ai et al., 2025), a new trend of multimodal learning with graphs (Zheng et al., 2024c; He et al., 2025a) is to align the text representation with graph structure for text-attributed graphs (Wen and Fang, 2023) or bridge molecular graphs and text data with semantic alignment for text-paired graphs (Liu et al., 2023e). Our work presents an alignment with graphs and languages, and similar frameworks can be expanded into graphs with other modalities.

Semantic-aware Graph Generative Modeling. Graph generation models (Kong et al., 2023; Vignac et al., 2022; Xu et al., 2024; Zeng et al., 2023) have a longstanding history with wide applications in many domains. These methods aim to capture and reproduce one or more important structure properties, such as community structure (Zhou et al., 2019), motif distribution (Zheng et al., 2024e), and densification in graph evolution (Zhou et al., 2020). In this work, we present a prototype of a semantic-aware graph predictive model, and semantic-aware graph generative models could also be a future direction, which may hold great potential in conditioned generation.

D Full-Resolution Figures

Due to space constraints in the main text, we resized the figures for a more compact presentation in Figure 3. We provide the full-resolution versions of the figures here for finer details.

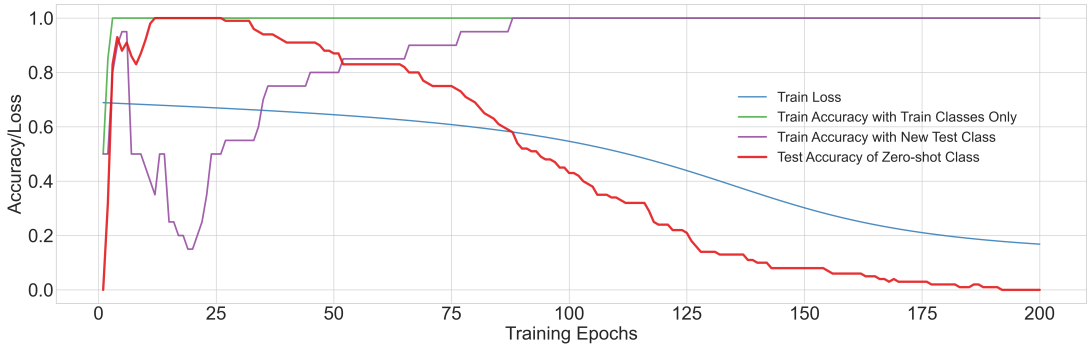


Figure 6: Novel class generalization result for our ZERO-Cora dataset.

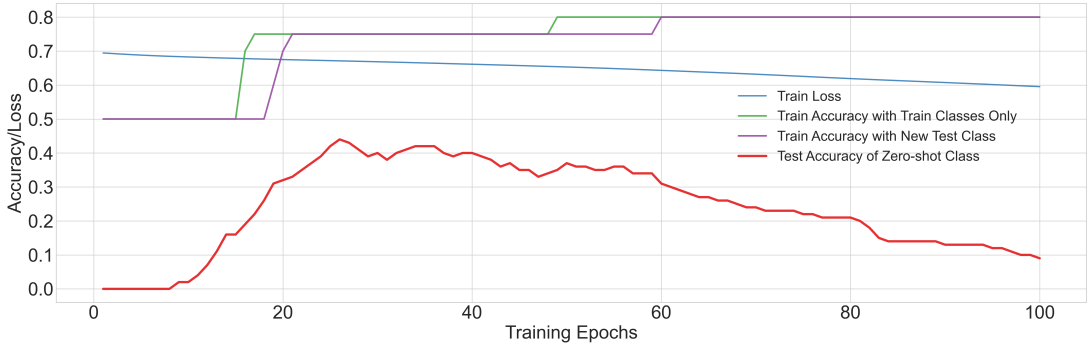


Figure 7: Novel class generalization result for our ZERO-CiteSeer dataset.

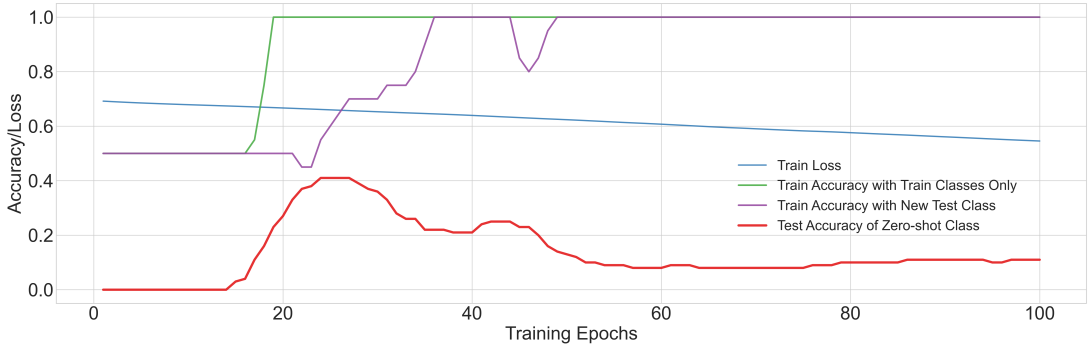
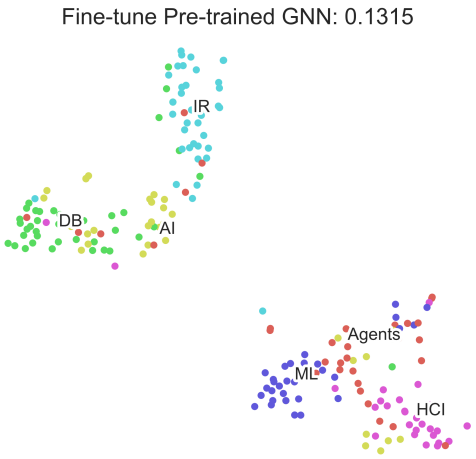
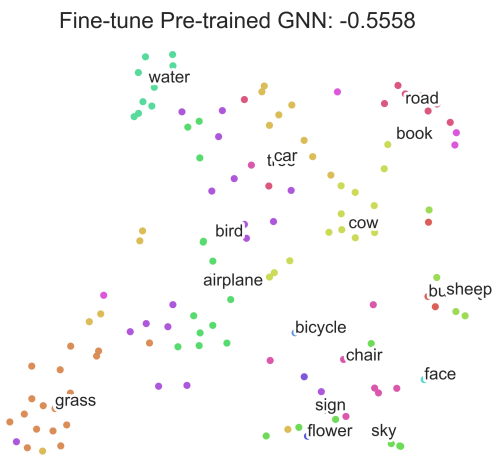


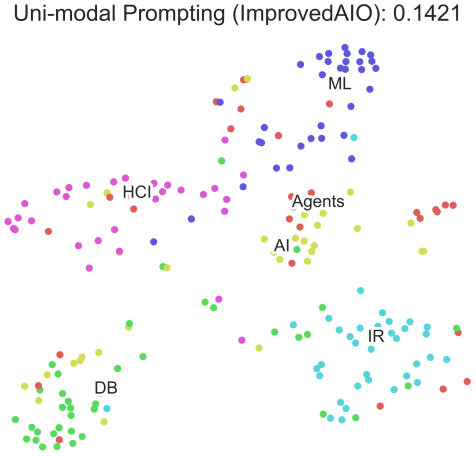
Figure 8: Novel class generalization result for our ZERO-PubMed dataset.



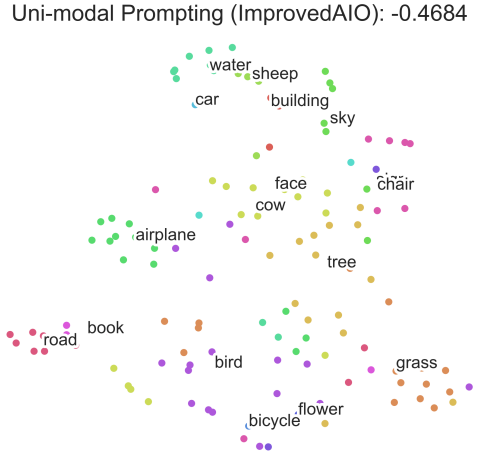
(a) Fine-tune on CiteSeer



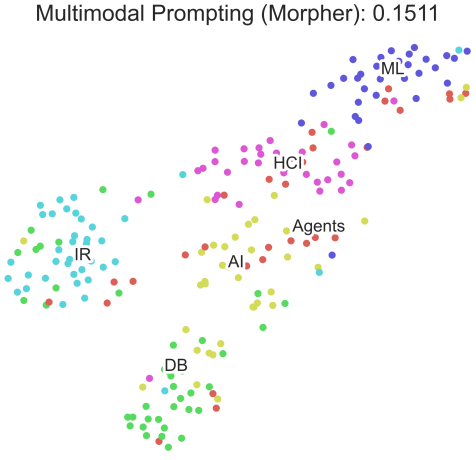
(b) Fine-tune on MSRC_21C



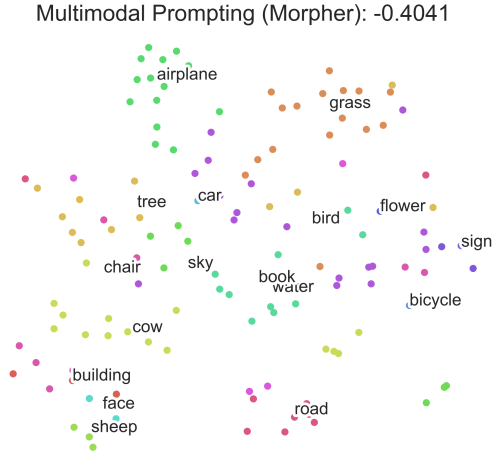
(c) ImprovedAIO on CiteSeer



(d) ImprovedAIO on MSRC_21C



(e) Morpher on CiteSeer



(f) Morpher on MSRC_21C

Figure 9: t-SNE embedding plots on CiteSeer (left) and MSRC_21C (right). We calculate the silhouette score, a metric for cluster quality (↑) ranged in $[-1, 1]$. It turns out that our Morpher leads to better adaptation.