Few-shot Learning over Graphs Using Topological Prompts

Jaidev Goel
jaidev@vt.edu
Virginia Tech
Blacksburg, Virginia, USA

Yuzhou Chen
yuzhou.chen@ucr.edu
University of California, Riverside
Riverside, California, USA

Yulia Gel
ygl@vt.edu
Virginia Tech
Blacksburg, Virginia, USA

Abstract

Prompt-based fine-tuning of pre-trained models has recently emerged as a promising trend for few-shot learning over graphs. Despite its significant potential, high variability and sensitivity to noise and perturbations remain the major challenges on the way of a wider adoption of prompt-based fine-tuning. We propose a new solution to these open problems by introducing the machinery of persistent homology to graph prompts. In particular, to better guide the fine-tuning process on downstream tasks, we extract intrinsic topological descriptors of the activation graphs of the pre-trained models in a form of Fréchet Means and incorporate this inherent topological information into the prompt-tuning process. Additionally, we implement bootstrapping over the topological summaries to mitigate the high variability, typically observed in prompt-based methods. Our extensive validation shows that the new *Topo-Prompt* tool results not only in relative gains in node classification accuracy up to 11% but also in up to 4 times reduction of variability with respect to the state-of-the-art prompt tuning methods. Furthermore, *Topo-Prompt* delivers superior robustness to perturbations, outperforming its competitors up to 25% under noisy conditions.

CCS Concepts

• **Computing methodologies** → **Machine learning**; **Artificial intelligence**; **Neural networks**; • **Mathematics of computing** → **Algebraic topology**; **Distribution functions**; **Graph theory**.

Keywords

Topological graph learning, Few-shot graph classification, Graph neural networks, Persistent homology, Graph pre-training, Transfer learning.

ACM Reference Format:

Jaidev Goel, Yuzhou Chen, and Yulia Gel. 2025. Few-shot Learning over Graphs Using Topological Prompts. In *Companion Proceedings of the ACM Web Conference 2025 (WWW Companion '25), April 28-May 2, 2025, Sydney, NSW, Australia.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3701716.3715549

WWW Companion '25, Sydney, NSW, Australia

ACM ISBN 979-8-4007-1331-6/2025/04
https://doi.org/10.1145/3701716.3715549

1 Introduction

Following the success of prompt learning in natural language processing, prompt-based methods for graph learning have recently emerged as a new promising direction. One of the key premises is that prompt learning potentially allows us to bridge the gap between pre-training and downstream tasks by reformulating input data to better align with pre-training objectives [10, 30, 31]. This emerging direction has shown notable effectiveness in leveraging transferable knowledge for various tasks while requiring a minimal re-tuning of the entire pre-trained network. However, designing effective prompts for graphs presents unique challenges due to the complex structure of the graph-structured data and the diverse nature of the pre-training models. To tackle these two inter-woven open problems and enhance the effectiveness of graph prompts, we propose a novel methodology that extracts the most essential latent higher-order *shape* structure present within the pre-trained model and incorporates it into the prompting process. By "shape" we broadly understand properties that are invariant under continuous transformations, e.g. stretching, twisting, and compressing. We approach this task by leveraging the machinery of topological data analysis (TDA) and, more specifically, persistent homology which has recently emerged as a powerful tool for capturing higher-order shape (sub)structures [1, 3, 8] and often results in superior performance and robustness of the PH-enhanced deep learning models [2, 4, 6, 28]. Using the PH tools, we can distill topological signatures of a learned neural network, for example, the intrinsic higher-order shape characteristics present within the space of activations of a neural network [16, 20]. Armed with such topological information about the underlying neural network model, we then develop a topologically aware prompt, or *Topo-prompt*. We posit that our method has the potential to capture more informative and more stable representations of the graph data. To the best of our knowledge, this is the first approach that guides the fine-tuning process on downstream tasks by incorporating higher-order structural information present within the pre-trained model itself.

Our key contributions can be summarized as follows:

- We introduce a novel prompt-tuning method, utilizing bootstrapped Fréchet means of activation graphs of pre-trained networks and topological loss functions, enabling more stable knowledge transfer in few-shot scenarios;
- we substantially outperform the state-of-the-art prompt-tuning methods, achieving up to 11% accuracy gains and major variance reduction across both homophilic and heterophilic graphs;

- we establish *Topo-Prompt*'s robustness under perturbations, yielding improvements in performance up to 25% under noisy conditions compared to existing approaches.

2 Background

Pre-Training and Fine-Tuning. Strategies for pre-training GNNs can be broadly categorized into three approaches. Node-level pre-training methods focus on understanding the underlying structural relationships within a graph, mapping nodes with similar properties to nearby regions in the embedding space to capture the graph's inherent mutual information [7, 32]. Edge-level pre-training methods aim to understand the regularities of edge attributes distributed across the graph, capturing how different interactions relate and correlate with each pair of nodes [12, 14, 15, 17, 19, 23]. Graph-level pre-training methods focus on generating meaningful graph embeddings composed of informative node embeddings, learning representations that capture the overall characteristics and patterns of the entire graph [13, 22, 24, 27, 29]. Prompt graphs typically involve: *prompt tokens* guide the pre-trained GNN towards a relevant and coherent response; *token structures* defines the structural relationships between prompt tokens, reflecting the graph's unique topology; *task tokens* represent class prototypes, providing additional context for the pre-trained GNN to effectively perform downstream tasks. Prompting methods differ in how they combine prompt graphs with input graphs and in their prompt tuning techniques.

3 Preliminaries and the Topo-prompt Idea in a Nutshell

Let $G = (V, E)$ be a graph, where $V = \{v_1, v_2, \ldots, v_N\}$ is the set of nodes (where $N = |V|$ is the number of nodes) and $E \subseteq V \times V$ is the set of edges. The node features are encoded in a matrix $\boldsymbol{X} \in \mathbb{R}^{N\times P}$, where $\boldsymbol{X} = [x_1, x_2, \ldots, x_N]^\top$ and each $x_i \in \mathbb{R}^P$ represents the feature vector of node v_i and P is the node feature dimension. Graph connectivity is described by the adjacency matrix $A \in \{0,1\}^{N\times N}$, where $A_{ij} = 1$ if $(v_i, v_j) \in E$, and $A_{ij} = 0$, otherwise.

Activation Graphs. Let f_θ be a pre-trained neural network parameterized by θ, consisting of n hidden layers. We denote the i-th hidden layer as $h_i : \mathbb{R}^{d_i} \to \mathbb{R}^{d_{i+1}}$ for $i = 1, \ldots, n$, where d_i is the dimensionality of the i-th layer. The trainable weight matrix connecting layers h_i and h_{i+1} is represented by $W_i \in \mathbb{R}^{d_i \times d_{i+1}}$. For an input $x \in \boldsymbol{X}$, we define the activation at layer i as $a_i(x) = h_i(h_{i-1} \ldots (h_1(x)))$. Activation graph $G_i(x) = (V_i \cup V_{i+1}, E_i)$ for the transition between layers i and $i+1$ is constructed as follows: $V_i = \{1, \ldots, d_i\}$ is the neurons in layer i, $V_{i+1} = \{1, \ldots, d_{i+1}\}$ is the neurons in layer $i+1$, and the edge set is defined as

$$E_i = \{(u, v, w) : u \in V_i, v \in V_{i+1}, w = a_i(x)[u] \cdot W_i[u, v]\},$$

where w is the edge weight computed as the dot product between the activation of neuron u in layer i and the corresponding weight connecting it to neuron v in layer $i+1$. This induces a sequence of activation graphs $\{G_i(x)\}_{i=1}^{n-1}$ for each input $x \in \boldsymbol{X}$.

Activation graphs encapsulate the essential response patterns of the trained neural network to specific input data. Shape properties of these graphs are known to shed important light on the hidden interplay between higher-order structures of the input data and organization of the fitted model [9]. As the pre-trained model aims to obtain more general data representations, analyzing how downstream data activates the pre-trained model can reveal critical latent information about how to tune the downstream model to our data.

Overview of Topological Prompt. We introduce a novel, topologically aware prompt-tuning approach that can be broken down into three key components which can be seen in Figure 1. In spirit of topological bootstrap [5], for each sample point $x \in \boldsymbol{X}$, we construct a bootstrapped sample set $\mathcal{B}$, i.e., randomly select data points for downstream processing. We then compute the set of activation graphs G_i and their corresponding topological descriptors, which we detail in subsequent sections. The next step is to infuse the pre-trained model embedding $\mathcal{E}$ with our obtained topological insights, which then serve as inputs to a learnable prompt module parameterized by ϕ. Following this construction, we compute the distance between the topology of pre-trained model and the prompt module to fine-tune the prompt in the direction of that the pre-trained model has learnt. Together these steps aim to achieve a balance between fine-tuning and prompting approaches. We posit that leveraging topological information of activation graphs in our proposed way encourages a *teacher-student* paradigm, where robust and more general representations of a learned model can be transferred to learner prompt modules during downstream tasks, guiding them to produce well-tuned data modifications. In the following sections, we highlight the details of our proposed method and then empirically validate our position.

Intuition behind Why and How do Topological Summaries Aid Knowledge Transfer? By analyzing how neural networks transform the underlying topology of data through their layers, we can investigate the fundamental mechanisms of knowledge transfer. Well-trained networks progressively reduce topological complexity (measured by persistent homology) of input data, transforming complicated, entangled manifolds into simpler, linearly separable components [20]. In this context, Fréchet means serve to capture aggregated representations of how data complexity evolves across activation patterns in different network layers. This topological summarization is particularly valuable for pre-trained models, which are specifically optimized for generalization tasks. By leveraging these summaries of data complexity transformation, we provide downstream fine-tuning with an informed initialization that preserves the pre-trained model's learned topological simplification strategy. The ability to effectively reshape topological structures, hence, emerges as a key factor for knowledge transfer and classification tasks, offering neq insights into how networks fundamentally transform data representations across different architectures and domains.

4 Empirical Validation

We evaluate *Topo-prompt* on 6 benchmark datasets in Table 1 for 1-, 5-, and 10-shot node classification tasks against state-of-the-art competitors: Gprompt [18], All-in-one [25], and GPF/GPF-plus [11]. Our methodology begins with Graph Contrastive Learning (GCL) pre-training on a two-layer base GNN, followed by our novel topological prompt-tuning approach. For K-shot classification, we compute Fréchet means (FMs) from bootstrapped activation graphs of the pre-trained network. These topological features enhance

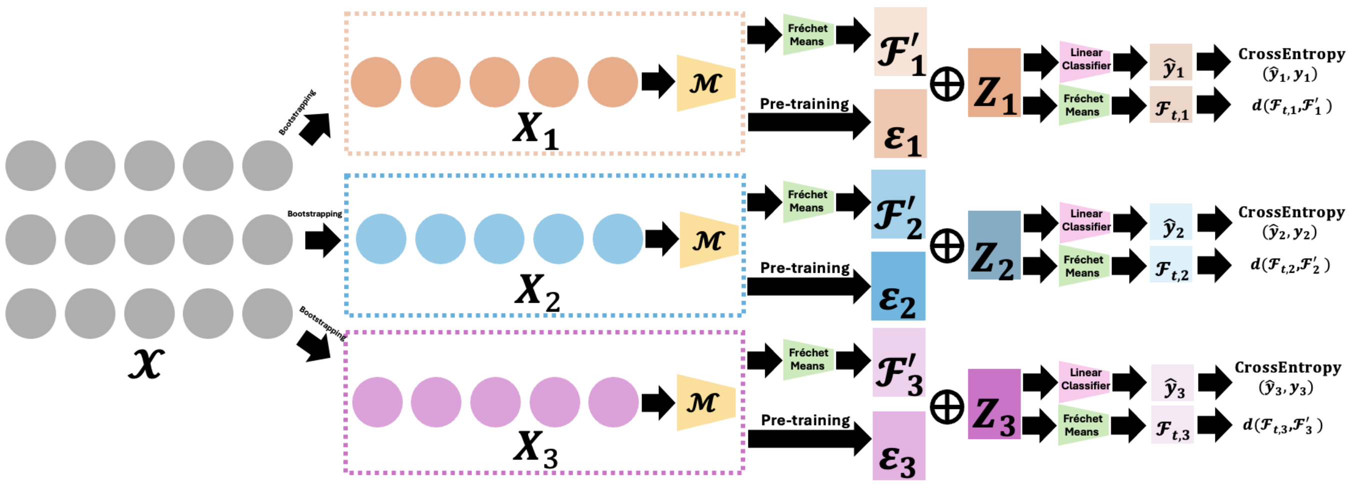

Figure 1: Overview of the *Topo-prompt* architecture. The approach consists of three key components: (1) bootstrapped sample generation, (2) computation of activation graphs and their topological summaries, and (3) infusion of topological insights into the pre-trained model embedding. These components work together to create a topologically-aware prompt-tuning method that balances fine-tuning and prompting approaches, leveraging a teacher-student paradigm for effective knowledge transfer in downstream tasks.

node embeddings through $\mathcal{E}' = \mathcal{E} + FM$. While sharing a similar token structure with GPF-plus, *Topo-prompt* introduces two key innovations: FM-enhanced embedding initialization and a topological loss function that aligns prompt structure with the pre-trained model's geometry. Following the ProG benchmark [33] protocols, all methods undergo 200 epochs of pre-training and 100 epochs of prompt-tuning using the Adam optimizer, with final classification performed via an MLP projection head.

Table 1: Summaries of the datasets.

Dataset	Nodes	Edges
Cora	2,708	10,556
Citeseer	3,327	9,104
PubMed	19,717	88,648
Wisconsin	251	515
Texas	183	325
OGBN-arxiv	169,343	1,166,243

4.1 Results

As shown in Table 2, *Topo-prompt* outperforms all baselines across datasets except Wisconsin, where it ranks second. Notably, our method achieves up to 4x reduction in standard deviation compared to baselines, addressing a critical challenge in graph prompt research. This improved stability stems from our bootstrapping approach and FM-based prompt construction, as revealed in our ablation study (Section 4.2). While *Topo-prompt* maintains superior performance in 10-shot settings, the gains are less pronounced compared to 1-shot and 5-shot scenarios, aligning with the general observation that topological approaches provide greater benefits under scarcer data conditions. The Wisconsin dataset exception likely stems from its activation patterns in our pre-trained GNN not exhibiting sufficiently rich topological structures.

4.2 Ablation Study and Robustness Analysis

Our ablation study on Cora and CiteSeer in figure 2 datasets reveals the significant contribution of each *Topo-prompt* component. Removing the topological loss resulted in accuracy drops of 3.22% and 5.80% respectively, while ablating bootstrapping led to larger decreases of 10%. The most dramatic impact came from removing both bootstrapping and FMs, causing accuracy drops of 40.38% (Cora) and 24.92% (CiteSeer), with significantly increased standard deviations. In robustness testing in figure 3, under node perturbations (2% to 30%), *Topo-prompt* consistently outperformed competitors, retaining 44.1% of its original accuracy at 30% perturbation compared to 33-38% retention for baselines. Further ablation under noise revealed that while removing the topological loss had minimal impact, ablating bootstrapping or FM features led to substantial performance drops, particularly at higher perturbation levels. This suggests that FM captures global structural properties while bootstrapping preserves local characteristics, with their synergy contributing to *Topo-prompt*'s superior robustness.

5 Conclusion

We have introduced *Topo-prompt*, a novel topologically-aware prompt-tuning method for few-shot node classification over graphs. As our experimental results have shown, incorporating bootstrapped Fréchet means of activation graphs and a topological loss function enable *Topo-prompt* to significantly boost the effectiveness of prompt-based fine-tuning. Empirical results on various homophilic and heterophilic graph datasets have demonstrated *Topo-prompt*'s

Table 2: Node-Classification performance with 1-shot, 5-shot and 10-shot setting.

K-Shot	Model	Cora	PubMed	CiteSeer	Wisconsin	Texas	OGBN-arxiv
1-Shot	GPF	0.489±0.060	0.397±0.070	0.477±0.031	0.493±0.062	**0.738±0.116**	0.113±0.034
	GPF-plus	**0.504±0.105**	**0.522±0.091**	**0.494±0.029**	***0.763±0.121***	0.729±0.088	0.162±0.031
	All-in-one	0.388±0.098	0.369±0.080	0.462±0.048	0.325±0.091	0.416±0.172	0.098±0.012
	Gprompt	0.492±0.043	0.481±0.080	0.452±0.049	0.686±0.087	0.718±0.126	**0.264±0.023**
	Topo-Prompt(ours)	***0.562±0.039***	***0.592±0.054***	***0.512±0.027***	**0.757±0.066**	***0.772±0.031***	***0.273±0.018***
5-Shot	GPF	0.588±0.045	0.553±0.076	0.623±0.034	0.842±0.054	0.844±0.072	0.149±0.024
	GPF-plus	**0.621±0.055**	0.595±0.058	**0.677±0.027**	***0.934±0.069***	**0.852±0.097**	0.242±0.069
	All-in-one	0.442±0.049	0.535±0.059	0.594±0.066	0.578±0.077	0.629±0.128	0.132±0.042
	Gprompt	0.584±0.031	**0.611±0.065**	0.663±0.036	0.907±0.094	0.774±0.084	**0.292±0.028**
	Topo-Prompt(ours)	***0.633±0.035***	***0.648±0.051***	***0.702±0.023***	**0.921±0.049**	***0.893±0.054***	***0.301±0.027***
10-Shot	GPF	0.609±0.037	0.572±0.048	0.638±0.029	0.873±0.033	0.856±0.069	0.269±0.014
	GPF-plus	0.668±0.028	0.667±0.022	0.776±0.025	**0.937±0.028**	**0.912±0.053**	0.282±0.036
	All-in-one	0.451±0.056	0.569±0.053	0.604±0.059	0.691±0.038	0.677±0.071	0.212±0.051
	Gprompt	**0.713±0.017**	**0.675±0.041**	**0.781±0.024**	0.931±0.044	0.838±0.047	**0.307±0.022**
	Topo-Prompt(ours)	***0.722±0.009***	***0.688±0.018***	***0.786±0.023***	***0.938±0.021***	***0.924±0.039***	***0.318±0.013***

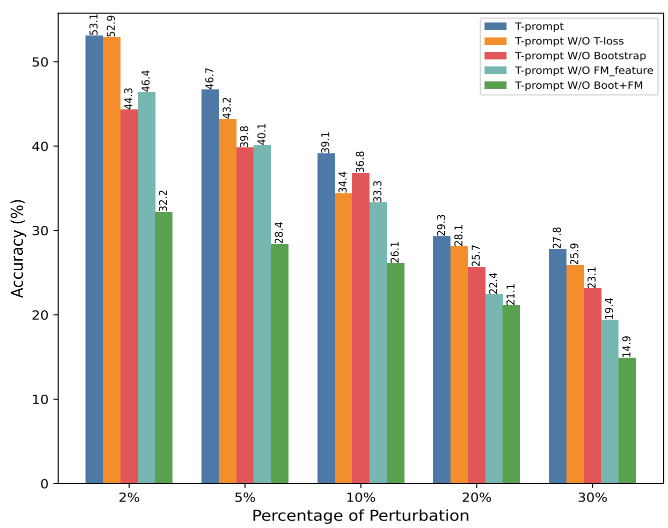

Figure 2: Ablation study of Topo-prompt's robustness under varying perturbation levels (from 2% to 30%) on the Cora dataset.

superiority, with accuracy improvements of up to 11% in few-shot node classification tasks and up to 4 times reduction in variability of graph learning. Moreover, *Topo-prompt* has been shown to exhibit the enhanced robustness to graph perturbations, yielding gains of up to 25% compared to the most powerful competitors. Our ablation studies also have indicated the synergistic importance of bootstrapping and Fréchet mean incorporation, contributing to both accuracy improvements and variance reduction.

As we view this paper as one of the very first steps toward improving our understanding of the utility of shape descriptors in few short graph learning and development of the associated topological prompts for graphs, we envision multiple future research directions. In particular, we plan to expand the ideas of *Topo-prompt* to learning multi-layer graphs and hyper-graphs, as well as to generalize *Topo-prompt* to dynamic networks. Furthermore, we will explore the utility of the joint hyper-graph representations for various pre-trained models and the associated graphical model-based LASSO techniques [21, 26] for more systematic quantification of uncertainties in transfer learning.

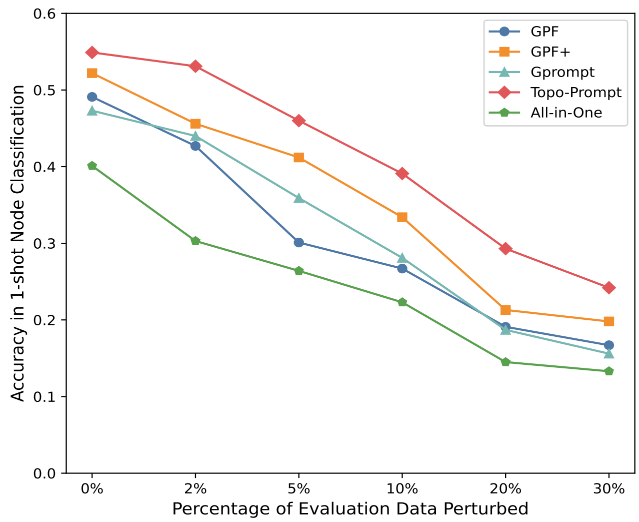

Figure 3: Robustness comparison of Topo-prompt with various graph prompting methods under different levels of node attribute perturbations (0% to 30%) on the Cora dataset.

6 Acknowledgments

This work was supported by the NSF grant TIP-2333703 and the ONR grant N00014-21-1-2530. The views expressed in the article do not necessarily represent the views of NSF or ONR.

References

[1] Gunnar Carlsson and Rickard Brüel Gabrielsson. 2020. Topological approaches to deep learning. In *Topological Data Analysis*. Springer, 119–146.
[2] Mathieu Carrière, Frédéric Chazal, Yuichi Ike, Théo Lacombe, Martin Royer, and Yuhei Umeda. 2020. Perslay: A neural network layer for persistence diagrams and new graph topological signatures. In *AISTATS*.
[3] Frédéric Chazal and Bertrand Michel. 2021. An introduction to topological data analysis: fundamental and practical aspects for data scientists. *Frontiers in artificial intelligence* 4 (2021), 667963.
[4] Yuzhou Chen, Baris Coskunuzer, and Yulia Gel. 2021. Topological relational learning on graphs. In *NeurIPS*.
[5] Yuzhou Chen, Miguel Heleno, Alexandre Moreira, and Yulia R Gel. 2023. Topological graph convolutional networks solutions for power distribution grid planning. In *PAKDD*.
[6] Yuzhou Chen, Elena Sizikova, and Yulia R Gel. 2022. TopoAttn-Nets: Topological Attention in Graph Representation Learning. In *ECML-PKDD*. 309–325.
[7] Jiashun Cheng, Man Li, Jia Li, and Fugee Tsung. 2023. Wiener graph deconvolutional network improves graph self-supervised learning. In *AAAI*, Vol. 37. 7131–7139.
[8] Herbert Edelsbrunner, John Harer, et al. 2008. Persistent homology-a survey. *Contemporary mathematics* 453, 26 (2008), 257–282.
[9] Ekin Ergen and Moritz Grillo. 2024. Topological Expressivity of ReLU Neural Networks. In *COLT*. 1599–1642.
[10] Taoran Fang, Yunchao Zhang, Yang Yang, Chunping Wang, and Lei Chen. 2024. Universal prompt tuning for graph neural networks. In *NeurIPS*.
[11] Taoran Fang, Yunchao Mercer Zhang, Yang Yang, Chunping Wang, and Lei CHEN. 2023. Universal Prompt Tuning for Graph Neural Networks. In *NeurIPS*.
[12] William L. Hamilton, Rex Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *NeurIPS*. 1025–1035.
[13] Weihua Hu, Bowen Liu, Joseph Gomes, Marinka Zitnik, Percy Liang, Vijay Pande, and Jure Leskovec. 2020. Strategies for Pre-training Graph Neural Networks. In *ICLR*.
[14] Ziniu Hu, Yuxiao Dong, Kuansan Wang, Kai-Wei Chang, and Yizhou Sun. 2020. GPT-GNN: Generative Pre-Training of Graph Neural Networks. In *SIGKDD*.
[15] Dongkwan Kim and Alice Oh. 2021. How to Find Your Friendly Neighborhood: Graph Attention Design with Self-Supervision. In *ICLR*.
[16] Théo Lacombe, Yuichi Ike, Mathieu Carriere, Frédéric Chazal, Marc Glisse, and Yuhei Umeda. 2021. Topological uncertainty: Monitoring trained neural networks through persistence of activation graphs. In *IJCAI*.
[17] Jintang Li, Ruofan Wu, Wangbin Sun, Liang Chen, Sheng Tian, Liang Zhu, Changhua Meng, Zibin Zheng, and Weiqiang Wang. 2023. What's Behind the Mask: Understanding Masked Graph Modeling for Graph Autoencoders. In *SIGKDD*. 1268–1279.
[18] Zemin Liu, Xingtong Yu, Yuan Fang, and Xinming Zhang. 2023. GraphPrompt: Unifying Pre-Training and Downstream Tasks for Graph Neural Networks. In *Web Conf.* 417–428.
[19] Yahui Long, Min Wu, Yong Liu, Yuan Fang, Chee Keong Kwoh, Jinmiao Chen, Jiawei Luo, and Xiaoli Li. 2022. Pre-training graph neural networks for link prediction in biomedical networks. *Bioinformatics* 38, 8 (02 2022), 2254–2262.
[20] Gregory Naitzat, Andrey Zhitnikov, and Lek-Heng Lim. 2020. Topology of deep neural networks. *Journal of Machine Learning Research* 21, 184 (2020), 1–40.
[21] Dorcas Ofori-Boateng, Jaidev Goel, Ivor Cribben, and Yulia R Gel. 2024. Graphical Model-Based Lasso for Weakly Dependent Time Series of Tensors. In *ECML-PKDD*.
[22] Jiezhong Qiu, Qibin Chen, Yuxiao Dong, Jing Zhang, Hongxia Yang, Ming Ding, Kuansan Wang, and Jie Tang. 2020. GCC: Graph Contrastive Coding for Graph Neural Network Pre-Training. In *SIGKDD*. 1150–1160.
[23] Arjun Subramonian. 2021. MOTIF-Driven Contrastive Learning of Graph Representations. In *AAAI*, Vol. 35. 15980–15981.
[24] Fan-Yun Sun, Jordan Hoffman, Vikas Verma, and Jian Tang. 2019. InfoGraph: Unsupervised and Semi-supervised Graph-Level Representation Learning via Mutual Information Maximization. In *ICLR*.
[25] Xiangguo Sun, Hong Cheng, Jia Li, Bo Liu, and Jihong Guan. 2023. All in One: Multi-Task Prompting for Graph Neural Networks. In *SIGKDD*. 2120–2131.
[26] Yu Wang, Byoungwook Jang, and Alfred Hero. 2020. The sylvester graphical lasso (syglasso). In *AISTATS*.
[27] Yaochen Xie, Zhao Xu, and Shuiwang Ji. 2022. Self-Supervised Representation Learning via Latent Graph Prediction. In *ICML*.
[28] Zuoyu Yan, Tengfei Ma, Liangcai Gao, Zhi Tang, and Chao Chen. 2021. Link prediction with persistent homology: An interactive view. In *ICML*.
[29] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. 2020. Graph contrastive learning with augmentations. In *NeurIPS*, Vol. 33. 5812–5823.
[30] Xingtong Yu, Zhenghao Liu, Yuan Fang, Zemin Liu, Sihong Chen, and Xinming Zhang. 2024. Generalized graph prompt: Toward a unification of pre-training and downstream tasks on graphs. *IEEE Trans. Knowl. Data Eng.* (2024).
[31] Haihong Zhao, Aochuan Chen, Xiangguo Sun, Hong Cheng, and Jia Li. 2024. All in one and one for all: A simple yet effective method towards cross-domain graph pretraining. In *SIGKDD*. 4443–4454.
[32] Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. 2021. Graph Contrastive Learning with Adaptive Augmentation. In *Web Conf.* 2069–2080.
[33] Chenyi Zi, Haihong Zhao, Xiangguo Sun, Yiqing Lin, Hong Cheng, and Jia Li. 2024. ProG: A Graph Prompt Learning Benchmark. *arXiv:2406.05346* (2024).