Investigating Task Equivariant Graph Performance on Diverse Tasks

Sahil Ashish Nimsarkar[a], Priyanshu Kumawat[a], Mrinal Bhan[a], Avantika Singh[a]

[a] *Dr. SPM International Institute of Information Technolgy, , Naya Raipur, Raipur, 493661, Chhattisgarh, India*

Abstract

Few-shot node classification is an emerging challenge, where labelled data is scarce for each class and poses a major hurdle for current Graph Neural Networks. For real-world applications, traditional meta-learning techniques frequently call for enormous volumes of diverse training data, which is unfeasible. This work investigates the Task-Equivariant Graph (TEG) framework for few-shot node categorization. Our aim is to identify and explain TEG's limitations on a diverse set of datasets, thus building upon its already strong generalization capabilities. TEG is extremely suitable for real-world scenarios with minimal labeled data, achieving state-of-the-art performance even with limited training data. We extensively investigate TEG's generalizability and robustness on a wide variety of benchmark datasets to investigate its effectiveness in a wide range of applications.

Keywords: node classification, meta learning, graph neural networks, few-shot learning, task-equivariant graph framework, task embedder

1. Introduction

Graph Neural Networks (GNNs) have shown outrageous success in applications involving node classification on attributed graphs [22, 30, 25, 23], like social[16], biological[17], and recommendation service networks[21]. However, GNNs' performance heavily relies on having abundant labeled nodes for each class. In practice, many classes lack sufficient labeled nodes, and there are often novel classes where even manual labeling is difficult. An example of one of such insufficiently labeled node dataset is demonstrated in Figure 1. This

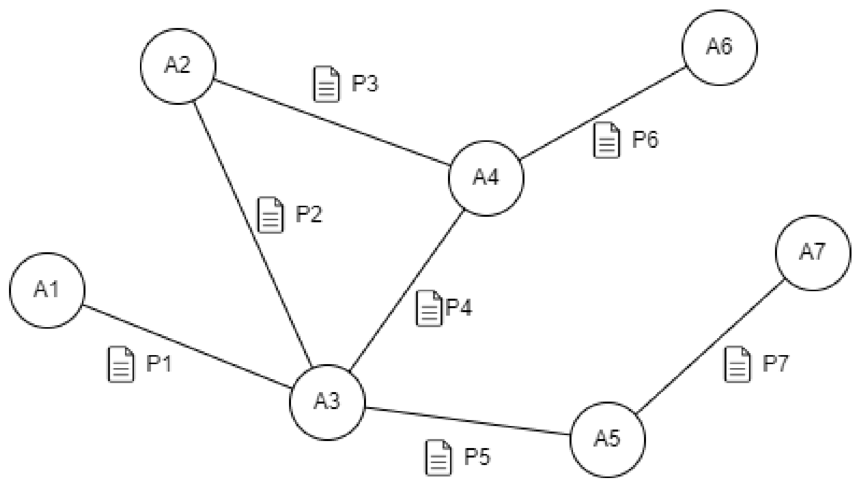

Figure 1: **Sample Visualization of the coauthorCS Dataset.** The diagram depicts a graphical representation of a subset of the coauthorCS dataset. The nodes, represented by circles - A_i (where $i \in [1, 7]$), correspond to authors or researchers in computer science domain. Lines connecting the nodes (edges) represent co-authorship between researchers on at least one publication. This visualization highlights the collaborative nature of research in computer science, where researchers frequently co-author papers with colleagues who share similar research interests.

necessitates few-shot node classification, where GNNs must classify nodes given limited labeled examples.

Existing methods for few-shot node classification typically rely on episodic meta-learning [29, 24]. This approach involves training the model on a series of mini-tasks (meta-tasks) with limited labeled data per class. While effective, these methods require a vast collection of diverse training meta-tasks that encompass a wide range of classes and nodes. Unfortunately, when the real-world scenario presents limited labeled data, the diversity of training meta-tasks is often restricted. This lack of diversity significantly hinders the performance of such methods.

This paper aims at investigating the effectiveness of the Task-Equivariant Graph (TEG) framework [1] one of the latest work in few-shot node classification with limited labeled data. While TEG demonstrates success in learning transferable task-adaptation strategies for similar training tasks (meta-tasks), as demonstrated in Figure 2. We aim to test its generalizability by applying it to a broader range of datasets.

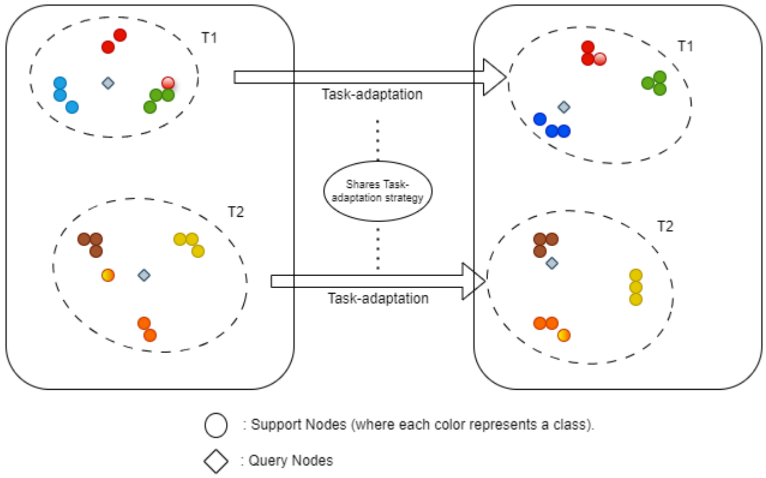

Figure 2: **Task-Adaptation Strategy Visualization.** The figure depicts a task-adaptation strategy in machine learning. Tasks (T1 & T2) share knowledge (represented by connections) between corresponding nodes (circles) based on their relative positions within the tasks. Squares denote support nodes, and triangles denote query nodes. This strategy allows efficient learning by leveraging knowledge from related tasks.

In this work, we make the following contributions:

- **Enhanced Understanding of TEG Framework:** We conduct a comprehensive investigation of Graph Neural Networks (GNNs) and delve deeper into the Task-Equivariant Graph (TEG) framework [1]. This investigation analyzed the reasoning behind TEG's potential for improvement, particularly in its capabilities for few-shot node classification with limited labeled data.

- **Verification Through Rigorous Testing:** We extend the evaluation of the TEG framework by applying it to a broader range of datasets. This includes specifically focusing on more edge-case and constrained datasets compared to those used in the original paper. This rigorous testing allowed us to verify the generalizability and effectiveness of TEG across diverse scenarios.

- **Insights from Underperformance:** By analyzing the performance of TEG on these various datasets, we identified the reasoning behind the low scores encountered. It further highlights areas where the TEG framework can be further improved for broader applicability and robustness in few-shot node classification tasks.

2. Literature Review

One of the early challenges in GNN development was ensuring their ability to handle variations in how data is presented[11, 18]. Even if the underlying structure remains the same, the order of nodes or the specific labeling of edges can change. This is where task-equivariant GNNs (TEGs) come in. TEGs are a specific type of GNN architecture designed to be equivariant, meaning their predictions remain consistent even when the data presentation changes.

2.1. TEG's Advantages and Areas of Exploration

Recent research has explored TEG's capabilities in various areas, highlighting their strengths and limitations. Here's a closer look:

- **Resilience to Limited Data:** Studies by Suo et al. [19] demonstrate the struggle of traditional meta-learning algorithms with limited training data. TEGs, however, showcase exceptional performance in these scenarios, effectively learning from limited data [1]. This makes them ideal for real-world applications where extensive labeled data might be scarce.

- **Applications in Dynamical Systems and Graph Autoencoders:** Sattoras et al. [20] explore TEG's effectiveness in two specific areas. Firstly, TEGs outperform existing methods in predicting particle positions during simulations, potentially impacting various scientific fields. Secondly, they address the symmetry problem in graph autoencoders, leading to superior reconstruction accuracy. However, further research is needed to explore TEG's potential in a wider range of applications.

- **Theoretical Underpinnings and Comparison with Other GNNs:** Research by [2] delves into the theoretical foundation of GNNs, emphasizing the importance of equivariance for graph data processing. It compares different GNN architectures, highlighting the strong separating and approximation power of k-order Folklore GNNs (k-FGNNs), a type of TEG architecture[10, 3, 4]. This analysis suggests TEG's suitability for tasks requiring high accuracy and efficiency. However, further exploration is needed to compare TEGs with other emerging GNN architectures.

2.2. Current Challenges for TEGs

While TEGs show promise, there's room for further exploration:

- **Performance on Diverse Real-World Problems:** Testing TEGs on a wider range of real-world problems will solidify their practical value [14, 6]
- **Scalability and Generalizability:** Investigating TEG's performance with larger and more intricate datasets is crucial for broader applicability [5] Additionally, exploring techniques like data augmentation that can help in improving generalizability with limited training data [12, 27] need to be explored.

In a nutshell in recent years TEGs framework has revolutionized the graph neural networks. The key comparison of TEG model with two of the latest contemporary techniques is depicted in Table 1.

Author	Methodology Used	Limitation	Scope of Improvement
Junseok Lee et al. [1]	Investigated the influence of task variety on meta-learning models employing an episodic approach and TEG.	The study was conducted in a controlled environment with limited classes and data instances.	Further research needed to explore TEG' performance in more complex real-world meta-learning settings with varying task structures and noise levels.
Emiel Hoogeboom et al. [20]	Introduced a novel E(n) equivariant GNN architecture.	Relied on introducing noise to node features, which might not be ideal for all scenarios.	Explore TEG in various applications outside of dynamical systems and graph autoencoders.
Miltiadis Kofinas et al. [2]	Explored Message Passing GNNs (MGNNs), Order-k Linear GNNs (k-LGNNs), and Order-k Folklore GNNs (k-FGNNs).	Focused on a limited set of GNN architectures and acknowledged limitations in generalizability.	Compare TEG with other emerging GNN architectures such as Gated Graph Neural Networks (GGNNs) or Attentive GNNs (AGNNs), and incorporate data augmentation techniques.

Table 1: Brief overview of related contemporary approaches

3. The TEG Approach

This section dives into the working of Task-Equivalent Graph model [1], the novel approach designed for efficient node classification. TEG leverages the power of episodic training, a technique commonly employed in meta-learning, to excel in scenarios with limited labeled data. Unlike traditional methods that struggle with data variations, TEG prioritizes learning robust node representations. These representations remain stable even when the data's underlying structure undergoes minor transformations, such as rotations or flips. This critical capability allows TEG to rapidly adapt to new tasks that exhibit similar patterns, significantly boosting classification performance.

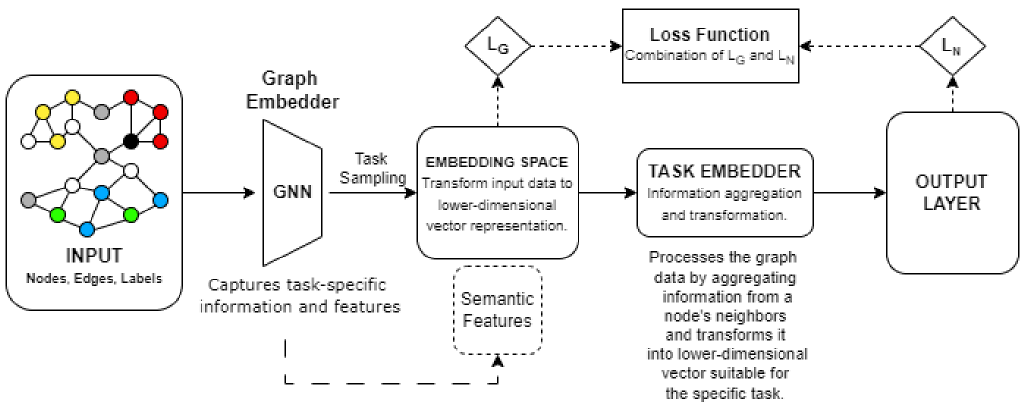

Figure 3: **Overview of Task-Equivalent Graph (TEG).** The TEG model begins with input layers for task-specific structures and structural features, followed by task-embedder which consists of multiple EGNN layers for graph data processing. Next, it transforms the output into a task-specific vector, leading to a task-dependent output layer.

To achieve this robustness, TEG employs a two-pronged feature extraction strategy. As shown in Figure 3, first, it extracts semantic features that capture the inherent properties of each node. This is achieved by leveraging Graph Convolutional Networks (GCNs), a powerful tool for learning node representations within graph-structured data. GCNs effectively analyze the interconnectedness of nodes, allowing TEG to understand the relationships between nodes and their attributes.

Second, TEG extracts structural features that encode the relative position of each node within the graph structure. If the graph is imagined as a map, then TEG calculates the shortest paths between each node and pre-defined anchor nodes strategically positioned throughout the map. This process allows TEG to understand a node's location in the broader context of the graph, providing valuable spatial information.

With both semantic and structural features in hand, TEG feeds them into a crucial component: the task embedder. This embedder utilizes Equivariant Graph Neural Networks (EGNNs), a specialized type of GNN that ensures task-equivariance. In simpler terms, task-equivariance guarantees that the model considers the inherent relationship between the training data (support set) and the testing data (query set) within a specific task pattern. This allows TEG to effectively transfer knowledge learned from similar tasks, further enhancing its adaptation capabilities.

Finally, TEG optimizes its performance by employing a combination of two loss functions. One loss function encourages the model to develop a fast adaptation strategy, allowing it to quickly adjust to new tasks. The other loss function prioritizes class separability, ensuring that the model can effectively distinguish between different node classes. By combining these loss functions with a tunable hyperparameter, TEG achieves a balance between adaptation speed and classification accuracy.

4. Results & Discussion

4.1. Datasets

This work leverages a comparative analysis of four graph network datasets: (CoauthorCS[13], OGBN-arxiv[9], Squirrel[28], CiteSeer[8]). The key properties investigated encompass the number of nodes, edges, features, and classes within each dataset, as detailed in the Table 2 below.

Nodes represent the individual entities within the network; the number of nodes varies considerably across the datasets. CoauthorCS exhibits the smallest entity set (18,333 nodes), while OGBN-arxiv boasts the largest (169,343 nodes). **Edges** characterize the relationships between nodes; the edge count demonstrates significant disparities. CoauthorCS possesses the fewest edges (163,788), whereas OGBN-arxiv again presents the most (1,166,243). **Features** capture the attributes associated with each node; the feature dimension demonstrates variations in data complexity. CoauthorCS offers the richest feature set (6,805), whereas Squirrel provides the most constrained one (2,089 features). **Classes** represent the distinct categories to which nodes belong; the class count highlights the inherent classification complexity. OGBN-arxiv offers the most fine-grained classification task with 40 classes, while Squirrel presents the coarsest with only 5 classes.

This selection of an appropriate set of datasets is to demonstrate the the performance of the TEG model on a much-diverse data.

	CoauthorCS	OGBN-arxiv	Squirrel	CiteSeer
# Nodes	18,333	169,343	5201	3327
# Edges	163,788	1,166,243	217,073	9228
# Features	6,805	128	2089	1620
# Classes	15	40	5	6

Table 2: **Statistics of the experimental datasets.** Summary of dataset characteristics including the count of nodes, edges, features, and classes for four different datasets: CoauthorCS, OGBN-arxiv, Squirrel, and CiteSeer.

4.2. Results

This section assesses the Task-Equivariant Graph few-shot learning (TEG) framework's performance for node categorization tasks on new and diverse benchmark datasets. We compare TEG's performance against state-of-the-art[7] baselines, particularly when dealing with challenging datasets. The scores achieved are shown in Table 3. The accuracy comparison is is visualized in Figure 4 for 3-way 3-shot setting and Figure 5 for 5-way 3-shot setting.

Dataset	**CoauthorCS**		**OGBN-arxiv**		**Squirrel**		**CiteSeer**	
Method	3-way 3-shot	5-way 3-shot	3-way 3-shot	5-way 3-shot	3-way 3-shot	5-way 3-shot	3-way 3-shot	5-way 3-shot
MAML	45.30 ± 0.19	25.7 ± 0.14	39.60 ± 0.57	33.19 ± 0.57	23.42 ± 0.9	19.42 ± 0.9	54.13 ± 2.18	49.36 ± 1.88
Meta-GCN	55.44 ± 0.69	32.94 ± 0.46	38.35 ± 0.41	35.40 ± 0.39	28.60 ± 0.9	25.60 ± 0.9	63.68 ± 2.65	58.68 ± 1.82
G-Meta	57.37 ± 0.63	42.35 ± 0.59	46.07 ± 0.25	37.47 ± 0.20	34.89 ± 1.34	29.89 ± 1.34	54.92 ± 2.26	49.92 ± 1.96
TENT	58.29 ± 1.21	53.52 ± 1.70	58.60 ± 0.80	51.72 ± 1.12	39.61 ± 0.74	34.61 ± 0.74	64.55 ± 2.63	62.55 ± 1.63
TAGM(w&g)	69.31 ± 0.59	58.82 ± 0.35	70.10 ± 0.43	64.84 ± 0.34	45.23 ± 1.73	39.23 ± 1.73	74.06 ± 1.40	70.64 ± 1.52
GPRGNN	82.84 ± 0.40	79.54 ± 1.38	69.45 ± 1.32	64.04 ± 0.80	$\mathbf{49.03 \pm 1.28}$	$\mathbf{44.08 \pm 0.62}$	72.30 ± 1.22	64.73 ± 0.78
TEG	$\mathbf{87.06 \pm 1.2}$	$\mathbf{82.18 \pm 0.88}$	$\mathbf{73.6 \pm 0.71}$	$\mathbf{66.27 \pm 1.83}$	46.85 ± 0.78	42.50 ± 0.70	$\mathbf{78.38 \pm 1.12}$	$\mathbf{72.18 \pm 1.82}$

Table 3: This table compares the performance of various few-shot learning methods on benchmark datasets (CoauthorCS[13], OGBN-arxiv[9], Squirrel[28], CiteSeer[8]). The table shows accuracy results for different numbers of classes (3-way, 5-way) and training examples per class (3-shot). TEG achieves the highest accuracy across most datasets and configurations, demonstrating its effectiveness in few-shot learning tasks.

We leverage the best base scores reported in [26, 15] that trained for various methods (MAML, TENT, GPRGNN, etc.) on CoauthorCS, OGB-arxiv, Squirrel, and CiteSeer datasets. These scores provide a common baseline for evaluating different methods across these diverse datasets. To ensure consistent testing, we train the TEG model for both 3-way 3-shot and 5-way 3-shot settings on all four datasets.

Across CoauthorCS, OGBN-arxiv, and CiteSeer datasets, several models

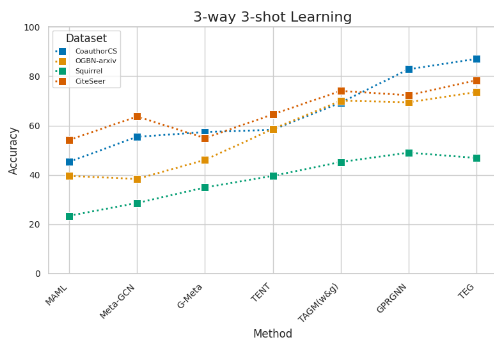

Figure 4: Accuracy comparison for various few-shot learning methods on benchmark datasets performed in 3-way 3-shot setting.

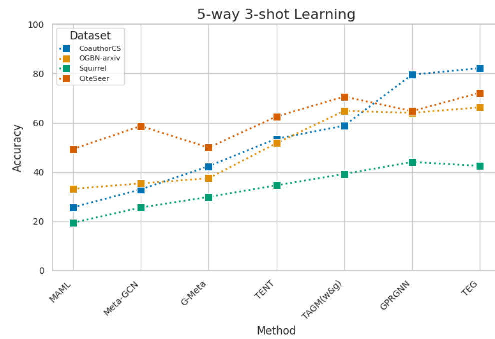

Figure 5: Accuracy comparison for various few-shot learning methods on benchmark datasets performed in 5-way 3-shot setting.

exhibited consistent and commendable performance, achieving high accuracies despite the scarcity of labeled data. Models like TEG demonstrated remarkable adaptability, leveraging equivariant neural networks to generalize effectively from minimal training samples.

Conversely, the Squirrel dataset presented a considerable challenge to the models, resulting in lower accuracies compared to other datasets. The characteristics of the Squirrel dataset, closely aligned with heterophily, where node features hold more predictive value than those of its neighbors. However, these features, particularly the node's own and self-looped features, showcased low correlation with node labels, indicating potential inadequacy in capturing pertinent patterns. Additionally, the dataset's low-quality node features, coupled with the presence of noisy features, imposed significant hurdles for accurate predictions. The choice of aggregation operation in graph neural network models further compounded these challenges, impeding the models' ability to effectively leverage information from the dataset's features.

4.3. Discussion

The disparities observed in model performance across datasets underscore the critical importance of understanding dataset characteristics and tailoring model architectures and training strategies accordingly. While certain models exhibited robust performance on CoauthorCS, OGBN-arxiv, and CiteSeer datasets, the unique challenges posed by the Squirrel dataset highlight the need for continued research and development efforts.

The lower accuracies observed for the Squirrel dataset compared to other

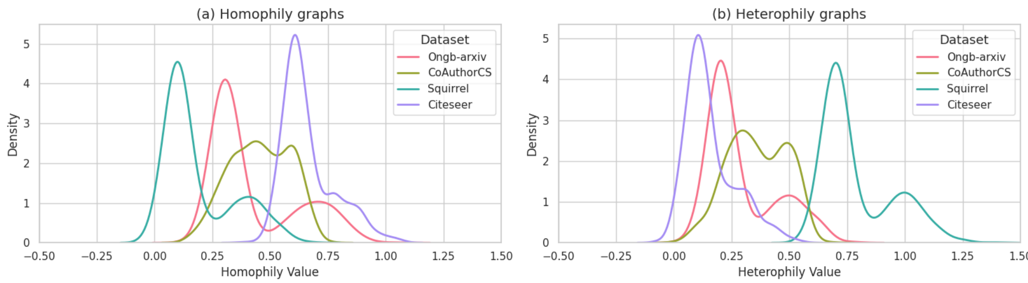

Figure 6: Weighted homophily and heterophily density distribution on different datasets

datasets necessitate a deeper exploration of the underlying factors contributing to this discrepancy. Potential avenues for investigation include the refinement of feature engineering techniques to enhance the representation of low-quality features and the development of novel aggregation strategies tailored to datasets with heterophilic characteristics. The distinct low homophily (h_{edge} and h_{adj}), which is evident from Figure 6, and the complex connectivity patterns, as highlighted in the Squirrel dataset, pose unique challenges that standard GNNs struggle to address effectively. Moreover, the integration of domain-specific knowledge and contextual information may offer insights into the inherent complexities of the Squirrel dataset, enabling more effective model adaptation and performance improvement.

Furthermore, the findings underscore the significance of leveraging insights gleaned from dataset analysis to inform model design and training methodologies. By iteratively refining model architectures and learning algorithms based on empirical observations, such as those illustrated in the weighted homophily and heterophily density distribution graphs as demonstrated in Figure 6, researchers can enhance the adaptability and effectiveness of few-shot learning models in diverse and challenging environments. Ultimately, these efforts contribute to the advancement of the field of few-shot learning, facilitating its broader applicability and efficacy in real-world scenarios.

5. Conclusion & Future Work

This work investigated Task-Equivariant Graph (TEG) framework for few-shot node classification. TEG demonstrates exceptional performance in scenarios with limited labeled data, overcoming a significant challenge for existing methods. Its core strength lies in its transferable task-adaptation strategy, allowing it to learn meta-knowledge applicable across diverse real-world applications. TEG's potential for breakthroughs in various domains, such as recommendation systems, social network analysis, and biological network analysis, has been well-established.

TEG's potential is vast, but further exploration is necessary. Scaling efficiently to handle massive graphs is crucial. Additionally, uncovering how TEG adapts to new tasks would provide valuable insights into its inner workings. Finally, ensuring TEG can generalize to graphs with mixed node types (heterophily) is essential for broader applicability. Addressing these limitations will solidify TEG as a powerful tool for tasks requiring accurate node classification with limited data, particularly in domains where real-world graphs exhibit heterophilic characteristics.

This research paves the way for future investigations aimed at optimizing TEG's performance on massive graphs. Hyperparameter tuning strategies specifically tailored to TEG's architecture and graph characteristics hold promise for significant efficiency gains. Additionally, unraveling the intricacies of TEG's decision-making process during adaptation would provide valuable insights for further improvements. Furthermore, exploring mechanisms to enhance TEG's ability to handle graphs with mixed node types, encompassing both categorical and numerical attributes, would broaden its applicability to real-world scenarios. Finally, incorporating techniques to address label noise, a ubiquitous challenge in real-world datasets, will be crucial for ensuring robustness and generalizability. By addressing these key areas, future research can refine TEG and unlock its full potential for efficient and accurate node classification on complex graph-structured data.

References

[1] Task-equivariant graph few-shot learning. *KDD 2023*, 2023.

[2] Graph neural networks for learning equivariant representations of neural networks. *ICLR*, 2024.

[3] Waiss Azizian and Marc Lelarge. Expressive power of invariant and

equivariant graph neural networks. *arXiv preprint arXiv:2006.15646*, 2020.

[4] Shaofei Cai, Liang Li, Jincan Deng, Beichen Zhang, Zheng-Jun Zha, Li Su, and Qingming Huang. Rethinking graph neural architecture search from message-passing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6657–6666, 2021.

[5] Yu Chen, Lingfei Wu, and Mohammed Zaki. Iterative deep graph learning for graph neural networks: Better and robust node embeddings. *Advances in neural information processing systems*, 33:19314–19326, 2020.

[6] Hao Cheng, Joey Tianyi Zhou, Wee Peng Tay, and Bihan Wen. Attentive graph neural networks for few-shot learning. In *2022 IEEE 5th International Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 152–157. IEEE, 2022.

[7] Stavros Georgousis, Michael P Kenning, and Xianghua Xie. Graph deep learning: State of the art and challenges. *IEEE Access*, 9:22106–22140, 2021.

[8] C Lee Giles, Kurt D Bollacker, and Steve Lawrence. Citeseer: An automatic citation indexing system. In *Proceedings of the third ACM conference on Digital libraries*, pages 89–98, 1998.

[9] Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: Datasets for machine learning on graphs. *Advances in neural information processing systems*, 33:22118–22133, 2020.

[10] Chenqing Hua, Guillaume Rabusseau, and Jian Tang. High-order pooling for graph neural networks with tensor decomposition. *Advances in Neural Information Processing Systems*, 35:6021–6033, 2022.

[11] Yilun Jin, Guojie Song, and Chuan Shi. Gralsp: Graph neural networks with local structural patterns. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 4361–4368, 2020.

[12] Jaekoo Lee, Hyunjae Kim, Jongsun Lee, and Sungroh Yoon. Transfer learning for deep learning on graph-structured data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

[13] Mina Lee, Percy Liang, and Qian Yang. Coauthor: Designing a human-ai collaborative writing dataset for exploring language model capabilities. In *Proceedings of the 2022 CHI conference on human factors in computing systems*, pages 1–19, 2022.

[14] Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard Zemel. Gated graph sequence neural networks. *arXiv preprint arXiv:1511.05493*, 2015.

[15] Sunil Kumar Maurya, Xin Liu, and Tsuyoshi Murata. Simplifying approach to node classification in graph neural networks. *Journal of Computational Science*, 62:101695, 2022.

[16] Bonaventure C Molokwu and Ziad Kobti. Social network analysis using rlvecn: Representation learning via knowledge-graph embeddings and convolutional neural-network. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 5198–5199, 2021.

[17] Giulia Muzio, Leslie O'Bray, and Karsten Borgwardt. Biological network analysis with deep learning. *Briefings in bioinformatics*, 22(2):1515–1530, 2021.

[18] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4938–4947, 2020.

[19] Qiuling Suo, Jingyuan Chou, Weida Zhong, and Aidong Zhang. Tadanet: Task-adaptive network for graph-enriched meta-learning. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1789–1799, 2020.

[20] Max Welling Victor Garcia Satorras, Emiel Hoogeboom. E(n) equivariant graph neural networks. 2022.

[21] Shoujin Wang, Liang Hu, Yan Wang, Xiangnan He, Quan Z Sheng, Mehmet A Orgun, Longbing Cao, Francesco Ricci, and Philip S Yu.

Graph learning based recommender systems: A review. *arXiv preprint arXiv:2105.06339*, 2021.

[22] Feng Xia, Ke Sun, Shuo Yu, Abdul Aziz, Liangtian Wan, Shirui Pan, and Huan Liu. Graph learning: A survey. *IEEE Transactions on Artificial Intelligence*, 2(2):109–127, 2021.

[23] Hao Yuan, Haiyang Yu, Shurui Gui, and Shuiwang Ji. Explainability in graph neural networks: A taxonomic survey. *IEEE transactions on pattern analysis and machine intelligence*, 45(5):5782–5799, 2022.

[24] Qiannan Zhang, Xiaodong Wu, Qiang Yang, Chuxu Zhang, and Xiangliang Zhang. Hg-meta: Graph meta-learning over heterogeneous graphs. In *Proceedings of the 2022 SIAM International Conference on Data Mining (SDM)*, pages 397–405. SIAM, 2022.

[25] Wentao Zhang, Zeang Sheng, Yuezihan Jiang, Yikuan Xia, Jun Gao, Zhi Yang, and Bin Cui. Evaluating deep graph neural networks. *arXiv preprint arXiv:2108.00955*, 2021.

[26] Feng Zhao, Min Zhang, Tiancheng Huang, and Donglin Wang. Tagm: Task-aware graph model for few-shot node classification. In *Proceedings of the 2023 ACM International Conference on Multimedia Retrieval*, pages 462–471, 2023.

[27] Tong Zhao, Yozen Liu, Leonardo Neves, Oliver Woodford, Meng Jiang, and Neil Shah. Data augmentation for graph neural networks. In *Proceedings of the aaai conference on artificial intelligence*, volume 35, pages 11015–11023, 2021.

[28] Rui Zhong, Yongheng Chen, Hong Hu, Hangfan Zhang, Wenke Lee, and Dinghao Wu. Squirrel: Testing database management systems with language validity and coverage feedback. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, pages 955–970, 2020.

[29] Fan Zhou, Chengtai Cao, Kunpeng Zhang, Goce Trajcevski, Ting Zhong, and Ji Geng. Meta-gnn: On few-shot node classification in graph meta-learning. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 2357–2360, 2019.

[30] Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. Graph neural networks: A review of methods and applications. *AI open*, 1:57–81, 2020.