On the Feasibility of Fidelity$^-$ for Graph Pruning

Yong-Min Shin[1] , **Won-Yong Shin**[1,*]

[1]Yonsei University, Seoul, South Korea

{jordan3414, wy.shin}@yonsei.ac.kr

Abstract

As one of popular quantitative metrics to assess the quality of explanation of graph neural networks (GNNs), *fidelity* measures the output difference after removing unimportant parts of the input graph. Fidelity has been widely used due to its straightforward interpretation that the underlying model should produce similar predictions when features deemed unimportant from the explanation are removed. This raises a natural question: "Does fidelity induce a global (soft) mask for graph pruning?" To solve this, we aim to explore the potential of the fidelity measure to be used for graph pruning, eventually enhancing the GNN models for better efficiency. To this end, we propose **F**idelity$^-$-**i**nspired **P**runing (**FiP**), an effective framework to construct global edge masks from local explanations. Our empirical observations using 7 edge attribution methods demonstrate that, surprisingly, general eXplainable AI methods outperform methods tailored to GNNs in terms of graph pruning performance.

1 Introduction

Alongside the recent popularity of graph neural networks (GNNs) for graph-related tasks spanning across domains from social network recommendations [Wu *et al.*, 2023] to molecular property predictions [Reiser *et al.*, 2022], such developments have resulted in an increasing demand in developing eXplainable AI (XAI) methods for GNN models. While early work focused on extending various edge attribution methods into GNNs [Baldassarre and Azizpour, 2019; Pope *et al.*, 2019a] for explaining the underlying model's behavior, many XAI methods specifically designed with GNN models in mind have been since proposed [Yuan *et al.*, 2023], e.g., GNNExplainer [Ying *et al.*, 2019]. More recent studies include FastDnX [Pereira *et al.*, 2023], relying on training SGC [Wu *et al.*, 2019] as a simpler surrogate model to the original GNN to extract relevant subgraphs. Although there are alternative forms of explanations for GNN models, the most prevalent one lies in the form of locally identifying the most relevant subgraph structure to the GNN's output for a given node (i.e., the predicted node class).

One of the broader objectives of XAI is to ultimately enhance the performance based on the knowledge gained from the explanation [Samek and Müller, 2019; Ali *et al.*, 2023]. In this regard, even though the majority of XAI methods of GNNs have successfully developed effective explanations, studies on utilizing such explanations to *improve* the underlying GNN model have been vastly underexplored. In the context of to graph datasets and GNN models, we focus on the problem of *graph pruning*, which is related to increasing the GNN model's efficiency by removing unimportant edges from the underlying graph. In other words, we are interested in removing edges from the input graph altogether, guided by edge attributions from XAI methods. If such utilization of XAI is shown to be successful, then we are capable of naturally boosting the efficiency of the underlying GNN model, since the time complexity of most GNN models is directly determined by the number of input edges [Wu *et al.*, 2021].

Our work aims at making a connection between graph pruning and *fidelity*, a quantitative metric that is often used to assess the quality of (graph) explanations [Ancona *et al.*, 2017; Yeh *et al.*, 2019; Yuan *et al.*, 2023]. In the context of GNN explanations, two variants of fidelity are commonly used. First, fidelity$^-$ measures the output difference between two instances when the original input graph is used and when the 'unimportant' parts (i.e., edges) are removed from the input graph. The intuition behind this metric is quite straightforward: if the explanation is valid, then structures deemed less important (i.e., assigned low edge attribution scores) should have less impact to the model's output after removal from the input. Second, for fidelity$^+$, the definition and interpretations are vice versa (i.e., removing 'important' parts). Revisiting on the intuition of the fidelity$^-$ measure, we hypothesize that frequently removed edges, when measuring fidelity$^-$, may potentially be simply removed from the original graph, provided that the quality of the given explanation is good enough.

In this work, we investigate the feasibility of this hypothesis, i.e., we are interested in using the intuition of fidelity itself in the context of pruning the edges from the input graph. In other words, we attempt to use the aggregate of given local explanations for each node for pruning edges from the input graph. Towards this end, we first provide a simple yet effec-
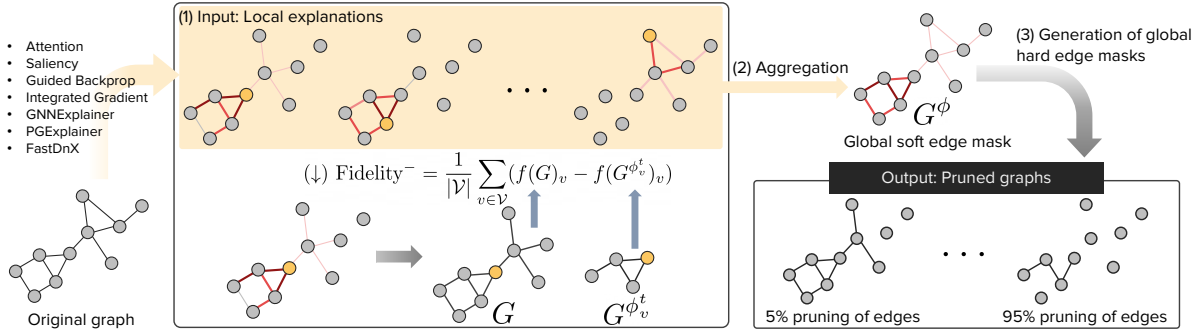

Figure 1: The overview of the FiP framework.

tive framework, built upon any edge attribution methods, to prune the input graph based on local explanations. Our empirical results from 7 different edge attribution methods comprehensively demonstrate the feasibility of using XAI methods for graph pruning. Surprisingly, we find that explanation methods that are specifically designed for GNN methods does not perform well in graph pruning, although they have superb performance in fidelity$^-$. Our analysis further validates this finding by explicitly visualizing pruned graphs, while emphasizing the necessity of developing a more sophisticated aggregation method.

2 Related Work

Fidelity. The fidelity metric, i.e., the measurement on the subset of input features highlighted by the explanation for the actual relevance to the model, has been acknowledged as one of the core properties for explanation [Yeh *et al.*, 2019]. In GNN explanations, two variants, i.e., fidelity$^+$ and fidelity$^-$, are widely used depending on the unimportant/important part to be removed from the input [Yuan *et al.*, 2023]. Although the fidelity measures are empirical measurements, theoretical analysis on its robustness [Agarwal *et al.*, 2022; Zheng *et al.*, 2023] has also been performed.

Graph pruning. Removing irrelevant edges from the underlying graph is a common strategy for reducing the computation complexity of the GNN models, as the complexity is known to be proportional to the number of edges [Chiang *et al.*, 2019]. Different ways to identify such irrelevant edges, including training a separate neural network [Zheng *et al.*, 2020], using effective resistance measures [Srinivasa *et al.*, 2020; Liu *et al.*, 2023b] and using graph lottery ticket hypothesis [Liu *et al.*, 2023a], were presented.

Our work focuses on the unique task of utilizing the intuition of the fidelity measurement on graph explanations to be used to prune edges from the input graph. To the best of our knowledge, [Naik *et al.*, 2024] lies in a similar objective to our study; however, it focuses on providing additional node features as a result.

3 Methodology

3.1 Basic Notations and Problem Settings

We denote an undirected and unweighted graph as $G = (\mathcal{V}, \mathcal{E}, X, \mathcal{A})$, where $\mathcal{V}$ is the set of nodes, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is

the set of edges, $X \in \mathbb{R}^{|\mathcal{V}| \times d}$ is the node feature matrix, and $\mathcal{A} : \mathcal{E} \rightarrow \mathbb{R}$ maps each edge in $\mathcal{E}$ to a real number (representing the set of edge weights or attributions). Also, we denote the set of neighbors for node v as $\mathcal{N}_v$. We focus on node classification, where a set of classes $C = \{1, ..., c\}$ are given. Then, we denote the one-hot label matrix $Y \in \{0, 1\}^{|\mathcal{V}| \times |\mathcal{C}|}$, where $Y_{v,:} = \mathbf{y}_v$ is the ground-truth label for node v. We assume that we are given a pre-trained L-layered GNN model f, which produces a prediction $\hat{Y} \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{C}|}$.

We consider a GNN explanation method Φ that takes a target node v, a target output $t \in \mathcal{C}$ as input and assigns a non-negative edge attribution value to a given edge $e_{i,j}$ on the GNN model, i.e., $\Phi(e_{i,j}; v, t) \coloneqq \phi_v^t(i, j) \in \mathbb{R}$. We set t as the predicted class of v unless otherwise stated.[1] By collecting the edge attribution values $\phi_v^t(i, j)$ for node v, we can construct a soft mask over the input graph, denoted as $G^{\phi_v^t}$.

3.2 Fidelity$^-$-inspired Pruning Framework

To utilize the local explanations for graph pruning, we propose **F**idelity$^-$-**i**nspired **P**runing, **FiP**, a simple yet effective framework that aggregates the edge attribution scores and creates a global edge mask (see Figure 1). The step-by-step description of FiP is as follows:

1. Explanations (i.e., local soft edge masks $G^{\phi_v^t}$) for each target node (yellow nodes in the figure) obtained by a specific edge attribution method are taken as input.

2. The local soft edge masks $G^{\phi_v^t}$ are aggregated over all $v \in \mathcal{V}$ via summing or averaging the edge attributions to generate G^ϕ (i.e., turning $\phi_v^t(i, j)$ into a global soft edge mask $\phi(i, j)$).

3. Hard edge masks are generated via discarding edges with the lowest aggregated edge attribution scores $\phi(i, j)$.

We can expect that the performance gradually decreases when we prune more edges (eventually resulting in information loss of the input), but a *good* global soft edge mask G^ϕ will assign a low score to noisy edges and a higher score to edges that severely hurt the performance when removed. Note that the process of FiP can be interpreted as a global version of fidelity$^-$, since both FiP and fidelity$^-$ discard unim-

---

[1]Although explanation methods often provide feature masks, we focus on edge-wise explanations in this work.
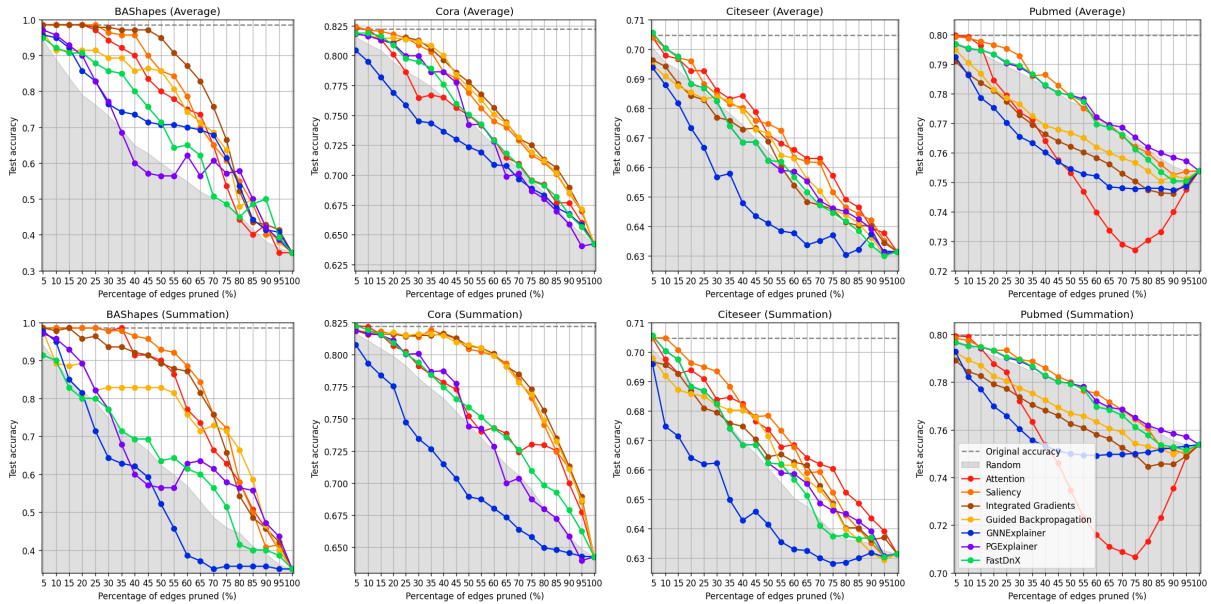
Figure 2: Graph pruning performance of FiP for 7 edge attribution methods as well as a random baseline on 4 benchmark datasets. The grey area indicates the performance of random attributions, and the dashed line indicate the test performance without any pruning.

portant edges in $G^{\phi_v^t}$ or G^ϕ. Although there may be more sophisticated methods to aggregate local edge attributions aside summation and averaging them, we leave the design of such methods as future work.

4 Empirical Observations

In this section, we evaluate the graph pruning performance of FiP using various GNN explanation methods.

4.1 Basic Settings

In our experiments, we train a 2-layer GAT model [Velickovic *et al.*, 2018] on 4 benchmark datasets, BA-Shapes [Ying *et al.*, 2019], Cora, Citeseer, and Pubmed [Yang *et al.*, 2016], where the model achieves test performance of 0.9857, 0.8531, 0.7389, and 0.8056, respectively. As mentioned, we only consider average or summation when aggregating local edge attributions in FiP.

4.2 Explanation Methods

We adopt the following seven edge attribution methods commonly used in the literature.

Attention (Att). The edge attention weights are treated as a proxy of edge attribution. We average the attention weights over all layers, similarly as in [Ying *et al.*, 2019; Sánchez-Lengeling *et al.*, 2020].

Saliency (SA). [Simonyan *et al.*, 2014] is the absolute value of the gradient with respect to the input.

Integrated Gradient (IG). [Sundararajan *et al.*, 2017] calculates an edge attribution score via approximating the integral of gradients of the model's output with respect to the input along the path from a baseline to the input.

Guided Backpropagation (GB). [Springenberg *et al.*, 2015] is similar to SA, except that the negative gradients are clipped during backpropagation, basically focusing on features with an excitation effect.

GNNExplainer (GNNEx). [Ying *et al.*, 2019] is the most widely used explanation method tailored to GNNs, where it identifies a local subgraph most relevant to the model's predictions by maximizing the mutual information.

PGExplainer (PGEx). [Luo *et al.*, 2020] trains a separate parameterized mask predictor to generate edge masks that identify edges important to the prediction.

FastDnX (FDnX). [Pereira *et al.*, 2023] is a recently proposed method for explaining GNNs, where it basically relies on a surrogate SGC model [Wu *et al.*, 2019] to explain the model's behavior.

4.3 Experimental Results on Graph Pruning

We show the experimental results of using various explanation methods in FiP with our findings. As our main result, Figure 2 shows the test performance when we remove $p\%$ of the edges with the lowest global soft edge mask G^ϕ, where $p \in \{5, 10, \cdots, 100\}$. For comparison, we also show the performance when we use random attributions (averaged over 10 independent trials), depicted as the grey area. From Figure 2, we make the following observations:

- Overall, edge attribution methods reveal their potential in edge pruning. For example, on BAShapes, we can delete half of the edges using IG only with the performance degradation of less than 4%.

- Despite being rare, there are cases where the accuracy after pruning outperforms the original test accuracy (e.g., on Citeseer, 5% pruning with FastDnX by summation).
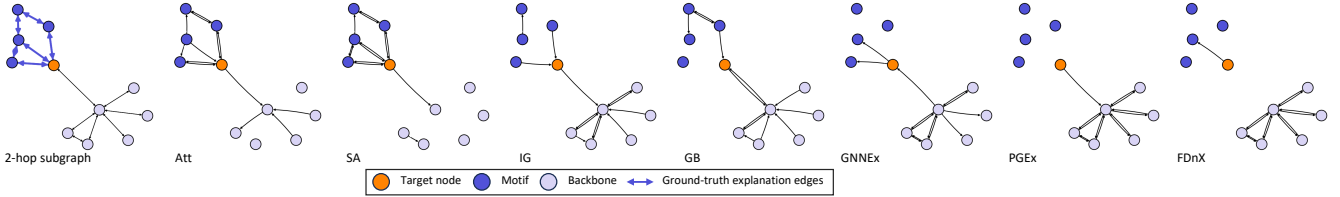
2-hop subgraph   **Att**   **SA**   **IG**   **GB**   **GNNEx**   **PGEx**   **FDnX**

● Target node   ● Motif   ○ Backbone   ↔ Ground-truth explanation edges

Figure 3: Visualizations of graph pruning using different edge attribution methods by removing 50% of the edges from the original graph.

Method	BAShapes	Cora	Citeseer	Pubmed
Att	3.63/2.26	4.89/4.11	**1.74/1.89**	6.32/6.58
SA	2.42/**1.58**	2.58/2.47	1.89/2.11	**2.21/2.11**
IG	**1.53**/2.58	**1.95/1.84**	5.32/4.26	6.58/6.58
GB	3.84/3.68	2.16/2.26	4.42/5.05	5.16/5.42
GNNEx	5.11/7.42	7.42/7.95	7.68/7.79	7.26/6.74
PGEx	5.58/4.84	5.84/5.58	4.00/3.84	**2.21**/2.42
FDnX	5.32/6.16	4.53/4.42	4.68/4.53	3.16/3.05
Random	7.53/6.47	6.42/6.68	5.47/5.68	3.05/3.00

Table 1: Rank in performance averaged over 20 different pruning percentages (average/summation).

Table 1 summarizes the rank in performance averaged over 20 different pruning cases for 7 edge attribution methods (as well as a random baseline) on 4 benchmark datasets. Our findings are as follows.

- The best edge attribution method for graph pruning tends be one of Att, SA, and IG. This is quite unexpected, as SA and IG are 'general' XAI methods (i.e., the ones not tailored to GNNs). In a similar regard, GNNExplainer tends to exhibit the worst performance for most of the datasets.

- There are no significant differences between using summation and average for aggregating local edge attributions in FiP.

4.4 Visualizations of Graph Pruning

We observe the effect of using different edge attributions for graph pruning by visualizing the resulting graphs. Figure 3 visualizes results on BA-Shapes for a specific target node (yellow nodes) using FiP with summation. We choose BA-Shapes since it is the only dataset that contains ground-truth explanation edges (see blue arrows). Note that only ground-truth explanation edges are meaningful in the sense that they construct house-shapes (i.e., motifs), influencing the ground-truth node labels included in the motif (dark blue and yellow nodes). The remaining edges do not have any semantic meanings and merely serve as a backbone structure of the graph. By setting $p = 50$ (i.e., removing half of the edges), we observe the following:

- Edge attribution methods showing superior performance on BA-Shapes (i.e., Att, SA, IG, and GB) tend to prune edges that are not included in the ground-truth explanation edges compared to GNNex, PGEx, and FDnX.

- Especially for Att and SA, the resulting graphs after pruning tend to be less noisy and explainable.

Method	BAShapes	Cora	Citeseer	Pubmed
Att	4.06×10^{-2}	3.67×10^{-2}	2.23×10^{-2}	2.46×10^{0}
SA	3.54×10^{-7}	2.21×10^{-7}	$\mathbf{8.90 \times 10^{-8}}$	2.46×10^{0}
IG	6.25×10^{0}	1.26×10^{0}	5.68×10^{-1}	$\mathbf{2.25 \times 10^{0}}$
GB	3.77×10^{0}	1.42×10^{0}	7.04×10^{-1}	2.40×10^{0}
GNNEx	$\mathbf{3.44 \times 10^{-7}}$	$\mathbf{2.14 \times 10^{-7}}$	3.52×10^{-1}	2.46×10^{0}
PGEx	3.83×10^{-7}	2.04×10^{-2}	7.11×10^{-3}	2.46×10^{0}
FDnX	1.41×10^{-1}	1.77×10^{-2}	7.05×10^{-3}	2.46×10^{0}

Table 2: Measurement of fidelity$^-$.

4.5 Relationship with Fidelity Scores

We perform further analysis by measuring the average fidelity$^-$ scores over all nodes in the graph for each edge attribution method. In our setting, we measure fidelity$^-$ as the average output (logit) difference between using the original graph and a sparser graph as input, where the sparse graph is generated by removing 50% of the edges with the lowest edge attribution scores in G^ϕ. Lower fidelity$^-$ indicates a higher quality explanation for each node. Table 2 summarizes the measurement of fidelity$^-$ scores for all edge attribution methods and datasets. Here, we find that the fidelity$^-$ does not necessarily translate to graph pruning performance. As an example, GNNEx shows the best performance in fidelity$^-$ for the Cora dataset; however, the average rank when using GNNEx in FiP is 5.84 and 5.58 for average and summation aggregation cases, respectively.

5 Discussion and Conclusion

In this work, we have empirically validated the feasibility of using local edge attribution methods for edge pruning. Our empirical analysis demonstrated that 1) local edge attributions can be effectively used for graph pruning with our FiP framework and 2) general XAI methods outperform XAI methods tailored to GNN models. We believe that FiP will not only improve efficiency but also eventually result in sparser explanations, which makes manual inspection feasible [Pope *et al.*, 2019b], as sparse explanations are generally considered to be more human-comprehensible [Yuan *et al.*, 2023; Funke *et al.*, 2023]. Potential avenues of future work include development of more sophisticated aggregation methods as well as investigation of the relationship between sparsity levels and human comprehensiveness on the explanations.

References

[Agarwal *et al.*, 2022] Chirag Agarwal, Marinka Zitnik, and Himabindu Lakkaraju. Probing GNN explainers: A rigorous theoretical and empirical analysis of GNN explanation methods. In *AISTATS*, Virtual event, Mar. 2022.

[Ali *et al.*, 2023] Sajid Ali, Tamer Abuhmed, Shaker H. Ali El-Sappagh, Khan Muhammad, Jose Maria Alonso-Moral, Roberto Confalonieri, Riccardo Guidotti, Javier Del Ser, Natalia Díaz Rodríguez, and Francisco Herrera. Explainable artificial intelligence (XAI): what we know and what is left to attain trustworthy artificial intelligence. *Inf. Fusion*, 99:101805, 2023.

[Ancona *et al.*, 2017] Marco Ancona, Enea Ceolini, A. Cengiz Öztireli, and Markus H. Gross. A unified view of gradient-based attribution methods for deep neural networks. In *NeurIPS*, Long Beach, CA, Dec. 2017.

[Baldassarre and Azizpour, 2019] Federico Baldassarre and Hossein Azizpour. Explainability techniques for graph convolutional networks. In *ICLR Workshop on Learning and Reasoning with Graph-Structured Data*, Long Beach, CA, Jun. 2019.

[Chiang *et al.*, 2019] Wei-Lin Chiang, Xuanqing Liu, Si Si, Yang Li, Samy Bengio, and Cho-Jui Hsieh. Cluster-gcn: An efficient algorithm for training deep and large graph convolutional networks. In *KDD*, Anchorage, AK, Aug. 2019.

[Funke *et al.*, 2023] Thorben Funke, Megha Khosla, Mandeep Rathee, and Avishek Anand. Zorro: Valid, sparse, and stable explanations in graph neural networks. *IEEE Trans. Knowl. Data Eng.*, 35(8):8687–8698, 2023.

[Liu *et al.*, 2023a] Chuang Liu, Xueqi Ma, Yibing Zhan, Liang Ding, Dapeng Tao, Bo Du, Wenbin Hu, and Danilo P. Mandic. Comprehensive graph gradual pruning for sparse training in graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–15, 2023.

[Liu *et al.*, 2023b] Zirui Liu, Kaixiong Zhou, Zhimeng Jiang, Li Li, Rui Chen, Soo-Hyun Choi, and Xia Hu. Dspar: An embarrassingly simple strategy for efficient gnn training and inference via degree-based sparsification. *Transactions on Machine Learning Research*, 2023.

[Luo *et al.*, 2020] Dongsheng Luo, Wei Cheng, Dongkuan Xu, Wenchao Yu, Bo Zong, Haifeng Chen, and Xiang Zhang. Parameterized explainer for graph neural network. In *NeurIPS*, Virtual event, Dec. 2020.

[Naik *et al.*, 2024] Harish Naik, Jan Polster, Raj Shekhar, Tamás Horváth, and György Turán. Iterative graph neural network enhancement via frequent subgraph mining of explanations. *CoRR*, abs/2403.07849, 2024.

[Pereira *et al.*, 2023] Tamara A. Pereira, Erik Nascimento, Lucas E. Resck, Diego Mesquita, and Amauri H. Souza. Distill n' explain: explaining graph neural networks using simple surrogates. In *AISTATS*, Palau de Congressos, Spain, Apr. 2023.

[Pope *et al.*, 2019a] Phillip E. Pope, Soheil Kolouri, Mohammad Rostami, Charles E. Martin, and Heiko Hoffmann. Explainability methods for graph convolutional neural networks. In *CVPR*, pages 10772–10781, Long Beach, CA, Jun. 2019.

[Pope *et al.*, 2019b] Phillip E. Pope, Soheil Kolouri, Mohammad Rostami, Charles E. Martin, and Heiko Hoffmann. Explainability methods for graph convolutional neural networks. In *CVPR*, Long Beach, CA, CVPR 2019.

[Reiser *et al.*, 2022] Patrick Reiser, Marlen Neubert, André Eberhard, Luca Torresi, Chen Zhou, Chen Shao, Houssam Metni, Clint van Hoesel, Henrik Schopmans, Timo Sommer, and Pascal Friederich. Graph neural networks for materials science and chemistry. *Commun. Mater.*, 3(93), 2022.

[Samek and Müller, 2019] Wojciech Samek and Klaus-Robert Müller. *Towards Explainable Artificial Intelligence*, pages 5–22. Springer International Publishing, Cham, 2019.

[Sánchez-Lengeling *et al.*, 2020] Benjamín Sánchez-Lengeling, Jennifer N. Wei, Brian K. Lee, Emily Reif, Peter Wang, Wesley Wei Qian, Kevin McCloskey, Lucy J. Colwell, and Alexander B. Wiltschko. Evaluating attribution for graph neural networks. In *NeurIPS*, Virtual event, Dec. 2020.

[Simonyan *et al.*, 2014] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *ICLR Workshop Track Proceedings*, Banff, Canada, 4 2014.

[Springenberg *et al.*, 2015] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin A. Riedmiller. Striving for simplicity: The all convolutional net. In *ICLR Workshop Track Proceedings*, San Diego, CA, 5 2015.

[Srinivasa *et al.*, 2020] Rakshith Sharma Srinivasa, Cao Xiao, Lucas Glass, Justin Romberg, and Jimeng Sun. Fast graph attention networks using effective resistance based graph sparsification. *CoRR*, abs/2006.08796, 2020.

[Sundararajan *et al.*, 2017] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *ICML*, Sydney, Australia, 8 2017.

[Velickovic *et al.*, 2018] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, 2018.

[Wu *et al.*, 2019] Felix Wu, Amauri H. Souza Jr., Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Q. Weinberger. Simplifying graph convolutional networks. In *ICML*, Long Beach, CA, Jun. 2019.

[Wu *et al.*, 2021] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S. Yu. A comprehensive survey on graph neural networks. *IEEE Trans. Neural Networks Learn. Syst.*, 32(1):4–24, 2021.

[Wu *et al.*, 2023] Shiwen Wu, Fei Sun, Wentao Zhang, Xu Xie, and Bin Cui. Graph neural networks in recommender systems: A survey. *ACM Comput. Surv.*, 55(5):97:1–97:37, 2023.

[Yang *et al.*, 2016] Zhilin Yang, William W. Cohen, and Ruslan Salakhutdinov. Revisiting semi-supervised learning with graph embeddings. In *ICML*, New York City, NY, Jun. 2016.

[Yeh *et al.*, 2019] Chih-Kuan Yeh, Cheng-Yu Hsieh, Arun Sai Suggala, David I. Inouye, and Pradeep Ravikumar. On the (in)fidelity and sensitivity of explanations. In *NeurIPS*, Vancouver, Canada, Dec. 2019.

[Ying *et al.*, 2019] Zhitao Ying, Dylan Bourgeois, Jiaxuan You, Marinka Zitnik, and Jure Leskovec. Gnnexplainer: Generating explanations for graph neural networks. In *NeurIPS*, pages 9240–9251, Vancouver, Canada, Dec. 2019.

[Yuan *et al.*, 2023] Hao Yuan, Haiyang Yu, Shurui Gui, and Shuiwang Ji. Explainability in graph neural networks: A taxonomic survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(5):5782–5799, 2023.

[Zheng *et al.*, 2020] Cheng Zheng, Bo Zong, Wei Cheng, Dongjin Song, Jingchao Ni, Wenchao Yu, Haifeng Chen, and Wei Wang. Robust graph representation learning via neural sparsification. In *ICML*, Virtual Event, Jul. 2020.

[Zheng *et al.*, 2023] Xu Zheng, Farhad Shirani, Tianchun Wang, Wei Cheng, Zhuomin Chen, Haifeng Chen, Hua Wei, and Dongsheng Luo. Towards robust fidelity for evaluating explainability of graph neural networks. Vienna, Austria, Jul. 2023.