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 a graph neural network (GNN), *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 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 model for better efficiency. We propose **F**idelity$^-$-**i**nspired **P**runing (**FiP**), a straightforward yet effective method to construct global edge masks from local explanations. Our empirical observations using 7 edge attribution methods demonstrate that, surprisingly, general XAI 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 resulted in an increasing demand in developing eXplainable AI (XAI) methods for GNN models. While early works focused on extending various edge attribution methods into GNNs [Baldassarre and Azizpour, 2019; Pope *et al.*, 2019] for explaining the 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], where it relies 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.

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 has been vastly underexplored. Specific 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 some XAI method. If such utilization of XAI are shown to be successful, then we naturally result in 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].

Specifically, our work attempts to make 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. 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 for 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. 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 edges that frequently gets removed 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 metric itself in the context of pruning the edges of the input graph. In other words, we attempt to use the aggregate of given local explanations for each node for pruning the edges in the input graph. To the best of our knowledge, the only work that shares a similar objective to ours is [Naik *et al.*, 2024]; however, it focuses on providing additional node fea-

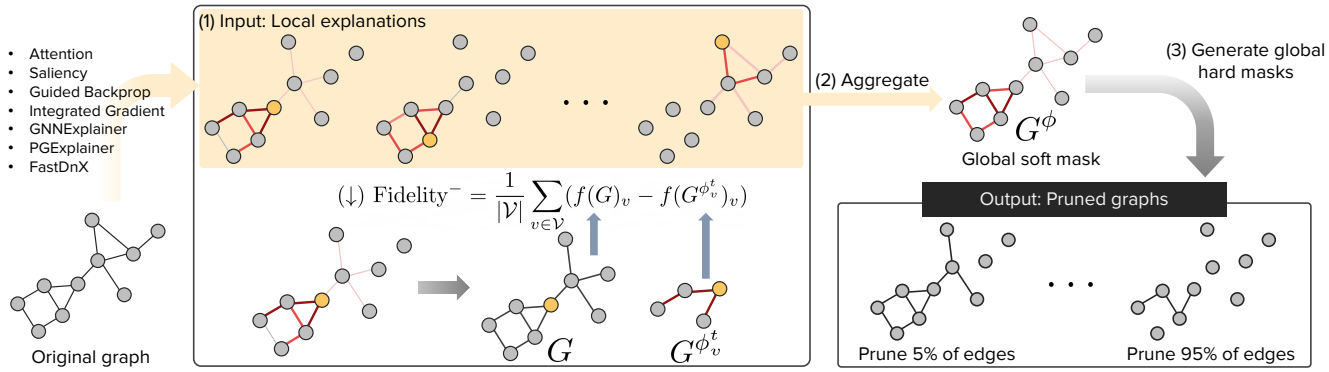

Figure 1: Overview of the FiP framework.

ture information as a result. Toward this end, we first provide a simple yet effective framework, which can be 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 shows 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, even if they have great fidelity$^-$ performances. Our analysis further proves this point by explicitly visualizing the pruned graph, and we provide theoretical discussions that shows fidelity$^-$ does not always translate to graph pruning performance, necessitating the development of a more sophisticated aggregation method.

2 Methodology

2.1 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 $\mathcal{A} : \mathcal{E} \to \mathbb{R}$ 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 denote a GNN explanation method Φ 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., $\Phi(e_{i,j}; v, t) = \phi^t_v(i, j) \in \mathbb{R}$. We set t as the predicted class of v otherwise stated.[1] By collecting the edge attribution values $\phi^t_v(i, j)$ for node v, we can construct a soft mask over the input graph, which denote as $G^{\phi^t_v}$.

2.2 Fidelity$^-$-inspired Pruning Framework

To utilize the local explanations for graph pruning, we propose **F**idelity$^-$-inspired **P**runing (denoted as **FiP**), a simple and straightforward process that aggregates the edge attribution scores and creating a global edge mask (see Figure 1). The flow of the framework is as follows:

1. Explanations (i.e., local soft masks $G^{\phi^t_v}$) for each target node (yellow nodes in figure) from a specific edge attribution method are taken as input.
2. The local soft masks $G^{\phi^t_v}$ are aggregated over all $v \in \mathcal{V}$ via summation or averaging the edge attributions to generate G^{ϕ} (i.e., turning $\phi^t_v(i, j)$ into a global soft mask $\phi(i, j)$).
3. Hard masks (i.e., edge pruning) 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 (as it eventually results in loss of input information), 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 discards unimportant edges in $G^{\phi^t_v}$ or G^{ϕ}. Although there may be more sophisticated methods to aggregate local edge attributions aside summation and averaging, we leave those investigations as future work.

3 Empirical Observations

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

3.1 Basic Settings

For 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 we aggregate local edge attributions in FiP.

3.2 Explanation Methods

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

[1] Although explanation methods often provide feature masks, we focus on edge-wise explanations in this work.

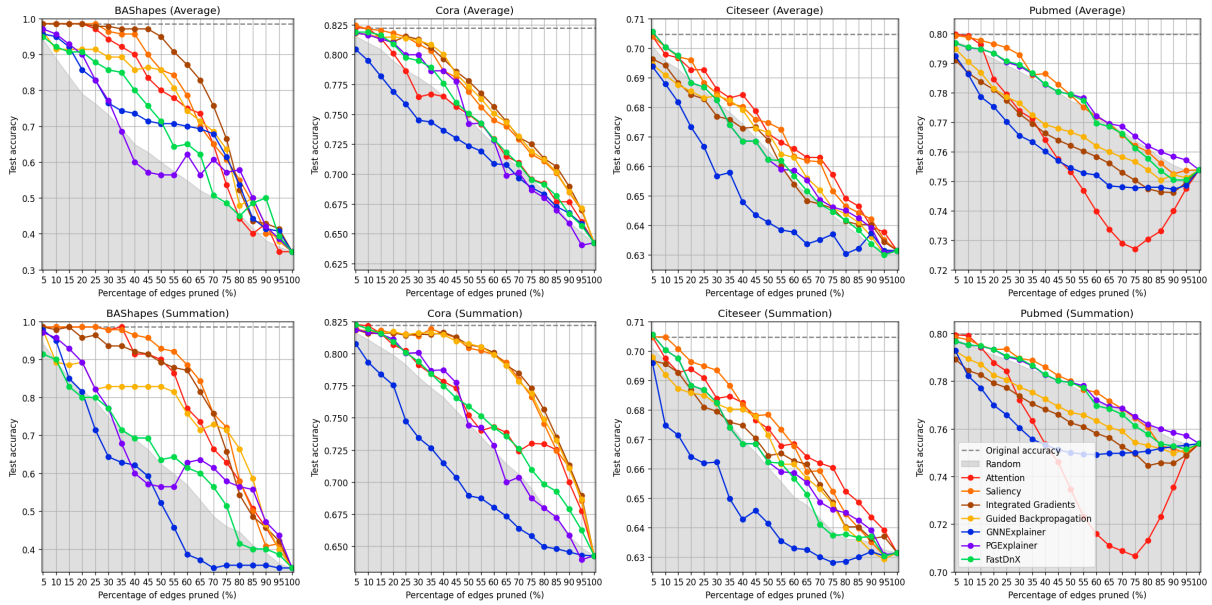

Figure 2: Graph pruning performance of FiP for 7 edge attribution methods (with a random baseline) for 4 benchmark datasets. The grey area indicates the performance of random attributions, and the dashed line indicate the test performance without any pruning.

- **Attention (Att)** denotes the edge attention weights which we use as a proxy of edge attribution. We average the attention weights over the layers, similar to [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 inputs along the path from a baseline to the inputs.
- **Guided Backpropagation (GB)** [Springenberg *et al.*, 2015] is similar to Saliency, 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 specifically designed for 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.

3.3 Experimental Results on Graph Pruning

In this section, we show the experimental results of using various explanation methods in FiP and discuss our findings. As our main result, Figure 2 shows the test performance when we remove $p\%$ of the edges with the lowest global soft mask G^{ϕ}. In all cases, $p \in \{5, 10, \cdots, 100\}$. As a baseline, we also show the performance when we use random attribution (averaged over 10 independent trials), depicted as the grey area. We make the following observations:

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: Average rank in performance over different pruning percentages (average/summation).

- Overall, edge attribution methods do show its potential in edge pruning. For example, we observe that we can delete half of the edges using Integrated Gradients and is able to achieve performance drop of less than 4% in the BAShapes dataset.
- Although vary rare, there are cases where the accuracy after pruning outperforms the original test accuracy (e.g., 5% pruning with FastDnX by summation in Citeseer dataset).
- Observing the average rank for each method in Table 1, the best edge attribution methods for graph pruning tends be one of Att, SA, or IG. This is quite unexpected, as SA and IG are 'general' XAI methods (i.e., not specifically tailored to GNNs). In a similar regard, GNNExplainer tends to exhibit the worst performance over most of the datasets.

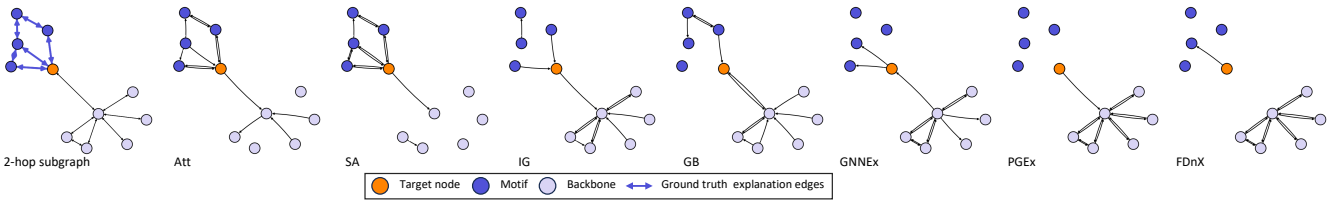

Figure 3: Visualizations of graph pruning using different edge attribution methods by removing 50% of the edges from the original graph.

Method	BAShapes	Cora	Citeseer	Pubmed
Att	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^{-}.

- There are no significant differences between using summation and averaging for aggregation for FiP in Table 1.

In summary, general XAI methods (e.g., Att, SA, IG) tend to result in better test performance after graph pruning.

3.4 Pruning Visualizations

We directly observe the effect of using different edge attributions for graph pruning by visualizing the resulting graph. Figure 3 shows the visualization results for BA-Shapes for a specific target node (yellow nodes) using FiP with summation. We have chosen to visualize BA-Shapes since it is the only dataset that contains ground-truth explanation edges (see blue arrows). By construction, only ground-truth explanation edges are meaningful in the sense that they construct house-shapes structures (i.e., motifs), which directly influences the ground-truth node labels included in the motif (dark blue and yellow nodes). The remaining edges does not have any semantic meanings and merely serves 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 that shows superior performance in 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.

3.5 Relationship with Fidelity Scores

We perform further analysis by first measuring the average fidelity^{-} scores over all nodes in the graph for each edge attribution methods. 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^{-} suggests that a higher quality explanation for each node. The fidelity^{-} scores is summarized in Table 2 for all edge attribution methods and datasets. Here, we find that the fidelity^{-} scores does not necessarily translate to graph pruning performances. As an example, GNNExplainer shows the best performance in fidelity^{-} for the Cora dataset, however the average ranking when using GNNExplainer in FiP is 5.84 and 5.58 for average and summation aggregations, respectively.

Theoretical discussions. To illustrate how this might happen, we attempt to analyze the relationship between the local soft mask $G^{\phi_v^t}$ and the resulting global soft mask G^{ϕ} in FiP.

Let us consider a simple case with a 1-layer GNN model where for a target node $v \in \mathcal{V}$, the explanation produces a set of edge attributions for all edges connected to v, i.e., $\phi_v^t(v,u)$ for all edges $e_{v,u}$ where $u \in \mathcal{N}_v$. Denote $\hat{\mathbf{y}}_v$ as the model output of node v with the original graph, and $\hat{\mathbf{y}}_v^{\phi_v^t}$ as the model output using the same graph with $\phi_v^t(v,u)$ as edge weights. Then, let us compare three different values for node v: the original prediction $\hat{\mathbf{y}}_v = \hat{Y}_{v,:}$, the prediction after applying the soft mask produced by the explanation $\hat{\mathbf{y}}_v^{\Phi_v^t}$, and the prediction after applying a global soft mask $\hat{\mathbf{y}}_v^{\phi}$. Then, we have the following theorem:

Theorem 1. *Given a soft mask $G^{\phi_v^t}$ from an explanation and a global soft mask G^{ϕ} obtained from the aggregation step from FiP, the following holds:*

$$\|\hat{\mathbf{y}}_v - \hat{\mathbf{y}}_v^{\phi}\|_2 \leq \|\hat{\mathbf{y}}_v - \hat{\mathbf{y}}_v^{\phi_v^t}\|_2 + \|\hat{\mathbf{y}}_v^{\phi_v^t} - \hat{\mathbf{y}}_v^{\phi}\|_2 \quad (1)$$

$$\leq \gamma(D_1 + D_2), \quad (2)$$

where $D_1 = \sum_{u \in \mathcal{N}_v} |1 - \phi_v^t(v,u)| \|\mathbf{x}_u\|_2$ *and* $D_2 = \sum_{u \in \mathcal{N}_v} |\phi(v,u) - \phi_v^t(v,u)| \|\mathbf{x}_u\|_2$, *and γ is a constant.*

The two terms C_1 and C_2 in Theorem 1 directly stems from the two terms in the right hand side in Eq. (1), respectively. The first term C_1 directly calculates fidelity^{-}, i.e., the quality of the local explanation ϕ_v^t. However, the bound also has a second term, C_2, which calculates the overall difference of the resulting global soft scores from the local edge attribution values. This implies that the distribution of $\phi(v,w)$ also plays a decisive role, suggesting that the performance is not solely dependent on the local fidelity scores.

4 Conclusion

In this work. we have performed empirical observations on the feasibility of using local edge attribution methods for edge pruning. Our observations show that local edge attributions can be effectively used for graph pruning with our FiP method, and general XAI methods tends to outperform XAI methods tailored to GNN models. Our analysis shows that the construction of global edge masks also play a crucial role, and potential avenues of future work include development of more sophisticated aggregation methods.

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.

[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.*, 2019] 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.

[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.

[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.

A Proof for Theorem 1

The proof mainly follows a similar logic to the proof for Theorem 1 of [Agarwal *et al.*, 2022]. Without loss of generality, we assume that the feed-forward process of our 1-layer GNN for a node v is as follows:

1. $\mathbf{m}_v = \text{softplus}\left(\mathbf{W}^1 \sum_{u \in \mathcal{N}_v} \mathbf{x}_u\right)$
2. $\mathbf{n}_v = \mathbf{W}_{\text{fc}}\mathbf{m}_v + \mathbf{b}$
3. $\hat{\mathbf{y}}_v = \text{softmax}(\mathbf{n}_v)$,

where $\mathbf{W}^1$, $\mathbf{W}_{\text{fc}}$, $\mathbf{b}$ are the weight matrix for the first message-passing layer, the weight matrix for the fully connected layer, the bias vector for the fully connected layer, respectively. We can also denote similar notations for $\hat{\mathbf{y}}_v^{\phi_v^t}$ and $\hat{\mathbf{y}}_v^{\phi}$ as follows:

1. $\mathbf{m}_v^{\phi_v^t} = \text{softplus}\left(\mathbf{W}^1 \sum_{u \in \mathcal{N}_v} \phi_v^t(v,u)\mathbf{x}_u\right)$
2. $\mathbf{n}_v^{\phi_v^t} = \mathbf{W}_{\text{fc}}\mathbf{m}_v^{\phi_v^t} + \mathbf{b}$
3. $\hat{\mathbf{y}}_v^{\phi_v^t} = \text{softmax}(\mathbf{n}_v^{\phi_v^t})$

and

1. $\mathbf{m}_v^{\phi} = \text{softplus}\left(\mathbf{W}^1 \sum_{u \in \mathcal{N}_v} \phi(v,u)\mathbf{x}_u\right)$
2. $\mathbf{n}_v^{\phi} = \mathbf{W}_{\text{fc}}\mathbf{m}_v^{\phi} + \mathbf{b}$
3. $\hat{\mathbf{y}}_v^{\phi} = \text{softmax}(\mathbf{n}_v^{\phi})$.

Now we are able to prove the theorem. We directly arrive at our conclusion by starting with the triangle inequality:

$$\|\hat{\mathbf{y}}_v - \hat{\mathbf{y}}_v^{\phi}\|_2 \leq \|\hat{\mathbf{y}}_v - \hat{\mathbf{y}}_v^{\phi_v^t}\|_2 + \|\hat{\mathbf{y}}_v^{\phi_v^t} - \hat{\mathbf{y}}_v^{\phi}\|_2. \tag{3}$$

We apply the following manipulations to the first term of the right hand side in Eq. (3).

$$\begin{aligned}
\|\hat{\mathbf{y}}_v - \hat{\mathbf{y}}_v^{\phi_v^t}\|_2 &\leq C_{\text{fc}}\|\mathbf{W}_{\text{fc}}\mathbf{m}_v + \mathbf{b} - \mathbf{W}_{\text{fc}}\mathbf{m}_v^{\phi_v^t} - \mathbf{b}\|_2 \\
&= C_{\text{fc}}\|\mathbf{W}_{\text{fc}}\mathbf{m}_v - \mathbf{W}_{\text{fc}}\mathbf{m}_v^{\phi_v^t}\|_2 \\
&\leq C_{\text{fc}}\|\mathbf{W}_{\text{fc}}\|_2\|\mathbf{m}_v - \mathbf{m}_v^{\phi_v^t}\|_2 \qquad (4) \\
&= C_{\text{fc}}\|\mathbf{W}_{\text{fc}}\|_2\|\text{softplus}\left(\mathbf{W}^1 \sum_{u \in \mathcal{N}_v} \mathbf{x}_u\right) \\
&\quad - \text{softplus}\left(\mathbf{W}^1 \sum_{u \in \mathcal{N}_v} \phi_v^t(v,u)\mathbf{x}_u\right)\|_2 \\
&\leq C_{\text{fc}}C_1\|\mathbf{W}_{\text{fc}}\|_2 \\
&\quad \times \|\mathbf{W}^1 \sum_{u \in \mathcal{N}_v} \mathbf{x}_u - \mathbf{W}^1 \sum_{u \in \mathcal{N}_v} \phi_v^t(v,u)\mathbf{x}_u\|_2 \\
&\leq C_{\text{fc}}C_1\|\mathbf{W}_{\text{fc}}\|_2\|\mathbf{W}_1\|_2 \\
&\quad \times \|\sum_{u \in \mathcal{N}_v} \mathbf{x}_u - \sum_{u \in \mathcal{N}_v} \phi_v^t(v,u)\mathbf{x}_u\|_2 \qquad (5) \\
&= \gamma\|\sum_{u \in \mathcal{N}_v} (1 - \phi_v^t(v,u))\mathbf{x}_u\|_2 \\
&\leq \gamma \sum_{u \in \mathcal{N}_v} |1 - \phi_v^t(v,u)|\|\mathbf{x}_u\|_2, \qquad (6)
\end{aligned}$$

where C_{fc}, C_1 denotes the Lipschitz constant for the softmax function and the softplus function, respectively. The Cauchy-Schwartz inequality is used in Eq. (4) and Eq. (5).

We also arrive at a similar conclusion for the second term:

$$\begin{aligned}
\|\hat{\mathbf{y}}_v^{\phi} - \hat{\mathbf{y}}_v^{\phi_v^t}\|_2 &\leq C_{\text{fc}}\|\mathbf{W}_{\text{fc}}\mathbf{m}_v^{\phi} + \mathbf{b} - \mathbf{W}_{\text{fc}}\mathbf{m}_v^{\phi_v^t} - \mathbf{b}\|_2 \\
&= C_{\text{fc}}\|\mathbf{W}_{\text{fc}}\mathbf{m}_v^{\phi} - \mathbf{W}_{\text{fc}}\mathbf{m}_v^{\phi_v^t}\|_2 \\
&\leq C_{\text{fc}}\|\mathbf{W}_{\text{fc}}\|_2\|\mathbf{m}_v^{\phi} - \mathbf{m}_v^{\phi_v^t}\|_2 \\
&= C_{\text{fc}}\|\mathbf{W}_{\text{fc}}\|_2\|\text{softplus}\left(\mathbf{W}^1 \sum_{u \in \mathcal{N}_v} \phi(v,u)\mathbf{x}_u\right) \\
&\quad - \text{softplus}\left(\mathbf{W}^1 \sum_{u \in \mathcal{N}_v} \phi_v^t(v,u)\mathbf{x}_u\right)\|_2 \\
&\leq C_{\text{fc}}C_1\|\mathbf{W}_{\text{fc}}\|_2 \\
&\quad \times \|\mathbf{W}^1 \sum_{u \in \mathcal{N}_v} \phi(v,u)\mathbf{x}_u - \mathbf{W}^1 \sum_{u \in \mathcal{N}_v} \phi_v^t(v,u)\mathbf{x}_u\|_2 \\
&\leq C_{\text{fc}}C_1\|\mathbf{W}_{\text{fc}}\|_2\|\mathbf{W}_1\|_2 \\
&\quad \times \|\sum_{u \in \mathcal{N}_v} \phi(v,u)\mathbf{x}_u - \sum_{u \in \mathcal{N}_v} \phi_v^t(v,u)\mathbf{x}_u\|_2 \\
&= \gamma\|\sum_{u \in \mathcal{N}_v} (\phi(v,u) - \phi_v^t(v,u))\mathbf{x}_u\|_2 \\
&\leq \gamma \sum_{u \in \mathcal{N}_v} |\phi(v,u) - \phi_v^t(v,u)|\|\mathbf{x}_u\|_2. \qquad (7)
\end{aligned}$$

Combining Eq. (6) and (7) and denoting $D_1 = \sum_{u \in \mathcal{N}_v} |1 - \phi_v^t(v,u)|\|\mathbf{x}_u\|_2$ and $D_2 = \sum_{u \in \mathcal{N}_v} |\phi(v,u) - \phi_v^t(v,u)|\|\mathbf{x}_u\|_2$, we arrive at Theorem 1. □