GInX-Eval: Towards In-Distribution Evaluation of Graph Neural Network Explanations

Kenza Amara[1], **Mennatallah El-Assady**[1], **Rex Ying**[2]

[1]Department of Computer Science, ETH Zurich, Switzerland
[2]Department of Computer Science, Yale University, US
{kenza.amara@ai,mennatallah.elassady@inf}.ethz.ch
rex.ying@yale.edu

Abstract

Diverse explainability methods of graph neural networks (GNN) have recently been developed to highlight the edges and nodes in the graph that contribute the most to the model predictions. However, it is not clear yet how to evaluate the *correctness* of those explanations, whether it is from a human or a model perspective. One unaddressed bottleneck in the current evaluation procedure is the problem of out-of-distribution explanations, whose distribution differs from those of the training data. This important issue affects existing evaluation metrics such as the popular faithfulness or fidelity score. In this paper, we show the limitations of faithfulness metrics. We propose **GInX-Eval** (**G**raph **In**-distribution e**X**planation **Eval**uation), an evaluation procedure of graph explanations that overcomes the pitfalls of faithfulness and offers new insights on explainability methods. Using a fine-tuning strategy, the GInX score measures how informative removed edges are for the model and the HomophilicRank score evaluates if explanatory edges are correctly ordered by their importance and the explainer accounts for redundant information. GInX-Eval verifies if ground-truth explanations are instructive to the GNN model. In addition, it shows that many popular methods, including gradient-based methods, produce explanations that are not better than a random designation of edges as important subgraphs, challenging the findings of current works in the area. Results with GInX-Eval are consistent across multiple datasets and align with human evaluation.

1 Introduction

While people in the field of explainable AI have long argued about the nature of good explanations, the community has not yet agreed on a robust single measure of explanation *correctness*. Phenomenon-focused explanations should match the ground-truth defined by humans and are evaluated by the accuracy metric. Model-focused explanations contribute the most to the model predictions and are evaluated by the faithfulness metrics. Because ground-truth explanations are often unknown, faithfulness and its variants are the most common measures of quality. Faithfulness metrics remove or retrain only the important graph entities identified and observe the changes in model outputs. However, this edge masking strategy creates Out-Of-Distribution (OOD) graph inputs, so it is unclear if a high faithfulness score comes from the fact that the edge is important or from the distribution shift induced by the edge removal [11].

We propose **GInX-Eval**, an evaluation procedure of in-distribution explanations that brings new perspectives on GNN explainability methods. Testing two edge removal strategies, we evaluate the impact of removing a fraction t of edges in the graph on the GNN model performance. To

XAI in Action: Past, Present, and Future Applications @ NeurIPS 2023.

overcome the OOD problem of faithfulness metrics, the model is fine-tuned and tested on the reduced graphs at each degradation level. The best explainability methods can identify the graph entities whose removal triggers the sharpest model accuracy degradation. We compare generative and non-generative methods on their **GInX** score against a random baseline across four real-world graph datasets and two synthetic datasets, all used for graph classification tasks. With this strategy, we show that existing explainers are not better than random in most of the cases. In addition, we show the overall superiority of GNNExplainer, PGMExplainer, and most of the generative methods above gradient-based methods and Occlusion. Our results lead to diverging conclusions from recent studies [33, 1] and again question the use of faithfulness as a standard evaluation metric in explainable AI (xAI). Finally, GInX-Eval is a useful tool to validate ground-truth explanations provided with some datasets and discover both human- and model-based explanations. Because it relies on a fine-tuning strategy of black-box pre-trained models, GInX-Eval is also a useful evaluation procedure in real-world scenarios where models are not publicly available and can only be used via API calls. Due to the computational cost of fine-tuning, GInX-Eval is not proposed as a systematic evaluation metric but as a tool to throw light on the true informative power of explainers and validate ground-truth explanations. To summarize our contributions:

- We show that faithfulness evaluates OOD explanations. In addition, we observe that (1) it is inconsistent with the accuracy metric, (2) it leads to divergent conclusions across datasets, and (3) across edge removal strategies. Overcoming the OOD problem, we propose **GInX-Eval** (**G**raph **In**-distribution e**X**planation **Eval**uation), a new evaluation framework for GNN explainability methods. GInX-Eval helps to select explainability methods that capture informative graph entities for the model. This includes filtering out bad methods and choosing methods that can correctly rank edges by their importance.
- We propose a validation protocol of ground-truth explanations using the GInX score. This way, we can measure the degree of alignment between human-based and model-based explanations.

2 Related work

Evaluation in xAI. To measure the correctness of explanations, a few metrics have been developed. GNN explainability method should satisfy accuracy, faithfulness, stability [25, 33, 2, 1], consistency and contrastivity [33], usefulness [7]. The two most popular approaches are: (1) measuring accuracy using ground-truth annotations and (2) measuring faithfulness using objective metrics [5]. However, the accuracy metric also referred to as plausibility [17, 18, 22], needs ground-truth explanations and is therefore less universal and more subjective than faithfulness.

Solving the OOD problem. Recent works have proposed to adapt the GNN model or develop robust explainability methods to overcome the out-of-distribution (OOD) problem. [9] argue that explanations should stay in the training data distribution and propose CoGe to produce Distribution Compliant Explanation (DCE). [16] propose a novel out-of-distribution generalized graph neural network. [13] do not remove features but apply small adversarial changes to the feature values. Instead of developing robust methods, [12] evaluate interpretability methods by observing how the performance of a fine-tuned model degrades when removing the features estimated as important. While this fine-tuning strategy circumvents the OOD problem, it has only been developed for CNN models on images to evaluate feature importance. Building on this fine-tuning strategy, we propose a new evaluation procedure for GNN models and introduce an alternative to faithfulness metrics.

3 Method

3.1 Preliminaries

Given a well-trained GNN model f and an instance of the dataset, the objective of the explanation task is to identify concise graph substructures that contribute the most to the model's predictions. The given graph can be represented as a quadruplet $G(\mathcal{V}, \mathcal{E}, \mathbf{X}, \mathbf{E})$, where $\mathcal{V}$ is the node set, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the edge set. $\mathbf{X} \in \mathbb{R}^{|\mathcal{V}| \times d_n}$ and $\mathbf{E} \in \mathbb{R}^{|\mathcal{E}| \times d_e}$ denote the feature matrices for nodes and edges, respectively, where d_n and d_e are the dimensions of node features and edge features.

In this work, we focus on structural explanation, *i.e.,* we keep the dimensions of node and edge features unchanged. Given a well-trained GNN f and an instance represented as $G(\mathcal{V}, \mathcal{E}, \mathbf{X}, \mathbf{E})$, an explainability method generates an explanatory edge mask $M \in \mathbb{R}^{|\mathcal{E}|}$ that is normalized. Furthermore, to obtain a human-intelligible explanation, we transform the edge mask to a sparse matrix by forcing to keep only the fraction $t \in \mathcal{T}$ of the highest values and set the rest of the matrix values to zero. Each explainability method can be expressed as a function $h : \mathcal{G} \to \mathcal{G}$ that returns for each input graph G an explanatory subgraph $h(G)$.

Edge removal strategies There are two strategies to select edges in a graph: the *hard* selection and the *soft* selection. Hard selection picks edges from the graph so that the number of edges and nodes is reduced. This creates subgraphs that very likely do not lie in the initial data distribution. Soft selection sets edge weights to zero when edges are to be removed. Therefore it preserves the whole graph structure with all nodes and edge indices. Following these two definitions, we define *hard* and *soft* explanations. Note here that the hard removal strategy might break the connectivity of the input graphs, resulting in explanations represented by multiple disconnected subgraphs.

3.2 Faithfulness metrics

Definition The faithfulness or fidelity scores are the most general quality metrics in the field of GNN explainability. To evaluate the correctness of the explanation, the explanatory subgraph or weighted graph $h(G)$ produced by the explainer h is given as input to the model to compute the fidelity score on the probabilities:

$$fid = |p(f(h(G)) = y) - p(f(G) = y)| \in [0; 1] \tag{1}$$

where y is the true label for graph G and $f(G)$, $f(h(G))$ the predicted labels by the GNN given G and $h(G)$ respectively. The closer fid is to 0, the more faithful the explanation is. The faithfulness score is averaged over the N explanatory graphs $h(G_i), i \leq N$ as:

$$\text{Faithfulness} = 1 - \frac{1}{N} \sum_{i=1}^{N} |p(f(h(G_i)) = y_i) - p(f(G_i) = y_i)| \in [0; 1] \tag{2}$$

The metric is normalized and the closer it is to 1, the more faithful the evaluated N explanations are to the initial predictions. The above score corresponds to the $fid{-}^{prob}$, one of the four forms of the fidelity scores [33], described in Appendix A.1.

Prior work. While being the most popular, faithfulness metrics have been recently criticized. Based on a "removal" strategy, i.e., we remove or keep the graph entities estimated as important, faithfulness withdraws some entities by setting them to a baseline value either removing them from the graph or putting their weight to zero. [13] correctly observe that this evaluation procedure favors graph entities that are far away from the baseline. In addition, truncated graphs after edge removal can lie out of the data distribution used for training the GNN model [12]. In this case, model behavior might differ not because of removing important information but because of evaluating a sample outside the training distribution.

3.3 GInX-Eval

GinX-Eval is an evaluation procedure of explainability methods that overcomes the faithfulness metrics' OOD problem and assesses the informativeness of explanatory edges towards the GNN model. To evaluate the explainer h, GInX-Eval first gathers the explanations produced by h for all graph instances. The explanatory edges can be ranked according to their respective weight in the subgraph: the most important edges have a weight close to 1 in the mask, while the least important ones have weights close to 0. At each degradation level t, we remove the top t fraction of the ordered explanatory edge set from the input graphs. We generate new train and test sets at different degradation levels $t \in [0.1, 0.2, ..., 1]$. The pre-trained GNN model is then fine-tuned at each degradation level on the new train dataset and evaluated on the new test data. While being the most computationally expensive aspect of GInX-Eval, fine-tuning is scalable (see Appendix B.4) and we argue that it is a necessary step to decouple whether the model's degradation in performance is due to the loss of informative edges or due to the distribution shift. The pseudo-code to implement GInX-Eval is given in Appendix B.3.

A drop in test accuracy when removing edges indicates that those edges were important for the model to make correct predictions. These edges are therefore considered as important as they are the most informative to the model. It is worth noticing that edges might be correlated and those spurious correlations can lead to an absence of accuracy drop when removing the top important edges and then a sudden decrease in accuracy when all correlated edges are removed. Following this evaluation procedure, we define the GInX score at t. It captures how low the test accuracy is after removing the top t edges. Let $h(G)$ be the explanatory subgraph generated by the method h, y the true label for graph G and $\chi : \mathcal{G} \times \mathcal{T} \to \mathcal{G}$ the removal function that takes an explanation $h(G)$ and returns the hard or soft explanatory graph containing the top $t \in \mathcal{T}$ edges. We define GInX(t) as:

$$\text{GInX}(t) = 1 - \text{TestAcc}(t) = 1 - \frac{1}{N_{test}} \sum_{i=0}^{N_{test}} \mathbb{1}(f(G_i \setminus \chi(h(G_i), t)) = y_i) \tag{3}$$

The closer the GInX score is to one, the more informative the removed edges are to the model. Note here that the GInX score at t can be computed for hard or soft explanations; however, we show in Appendix A.2 that the GInX score computed on hard explanations has higher expressiveness.

4 Experimental results

In the following section, we propose to validate GInX-Eval and show its superiority over the widely used faithfulness metric. We support our claims with well-chosen experiments.

Experimental setting Explainability methods were evaluated on two synthetic datasets, BA-2Motifs and BA-HouseGrid, three molecular datasets MUTAG, Benzene and BBBP, and MNISTbin. They all have ground-truth explanations available except for the BBBP dataset. We test two GNN models: GIN [14] and GAT [28] because they score high on the selected real-world datasets, with a reasonable training time and fast convergence. For the two synthetic datasets, we only use GIN since the GAT model does not give good accuracy. Further details on the datasets, GNN training parameters, and time are given in Appendix B. We compare non-generative methods, including the heuristic Occlusion [35], gradient-based methods Saliency [4], Integrated Gradient [27], and Grad-CAM [24], and perturbation-based methods GNNExplainer [32], PGMExplainer [29] and SubgraphX [34]. We also consider generative methods: PGExplainer [19], GSAT [21], GraphCFE (CLEAR) [20], D4Explainer and RCExplainer [30]. For more details on the differences between generative and non-generative explainers, we refer the reader to Appendix B.5. We compare those explainability methods to base estimators: Random, Truth, and Inverse. Random assigns random importance to edges following a uniform distribution. Truth estimates edge importance as the pre-defined ground-truth explanations of the datasets. The Inverse estimator corresponds to the worst-case scenario where edges are assigned the inverted ground-truth weights. If $w_{i,j}$ is the ground-truth importance of the edge connecting nodes i and j, the weight assigned by the Inverse estimator is equal to $1 - w_{i,j}$.

4.1 The Out-Of-Distribution Faithfulness Evaluation

The biggest limitation of the faithfulness metrics is the so-called OOD problem. The generated explanations are out-of-distribution, i.e. they lie outside the data distribution and "fool" the underlying predictor to change the original class, i.e., $f(h(G)) \neq f(G)$. Whereas, in factual explainability scenarios, we expect the explanatory graph $h(G)$ to have the same class as the input graph G, i.e., $f(h(G)) = f(G)$. Figure 1 illustrates the OOD problem: the extracted model embeddings of explanations of toxic molecules are more similar to the ones of non-toxic molecules. In this case, the model predicts the explanatory subgraphs to be non-toxic while they are valid toxic molecular fragments. The model prediction is altered not necessarily because we keep only the important entities but also because the model lacks knowledge about these new explanatory graphs. Therefore, the faithfulness score which definition is based on the model predictions of explanations, does not entirely capture the quality of explanations and is ill-suited to evaluate explainability methods.

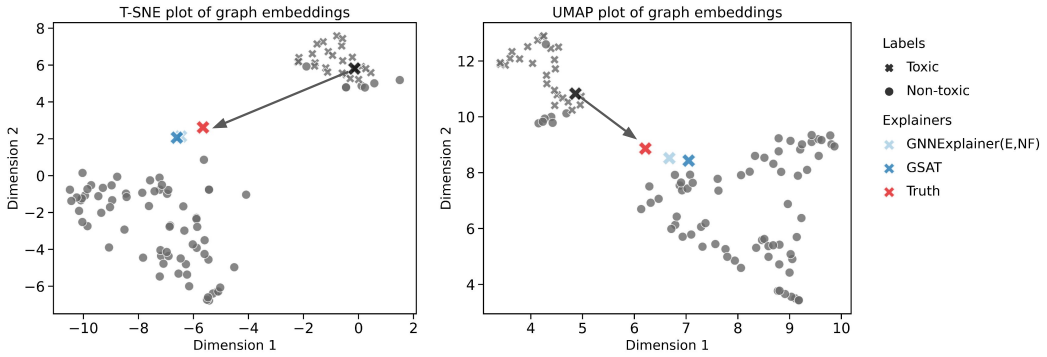

Figure 1: Illustration of the out-of-distribution problem: explanations of a toxic molecule lie closer to the non-toxic molecular representations. Graph embeddings were extracted after the readout layer of the pre-trained GIN model for the MUTAG dataset. We use both t-SNE and UMAP to project the embeddings into 2D representations. Both projection methods show the existence of out-of-distribution explanations.

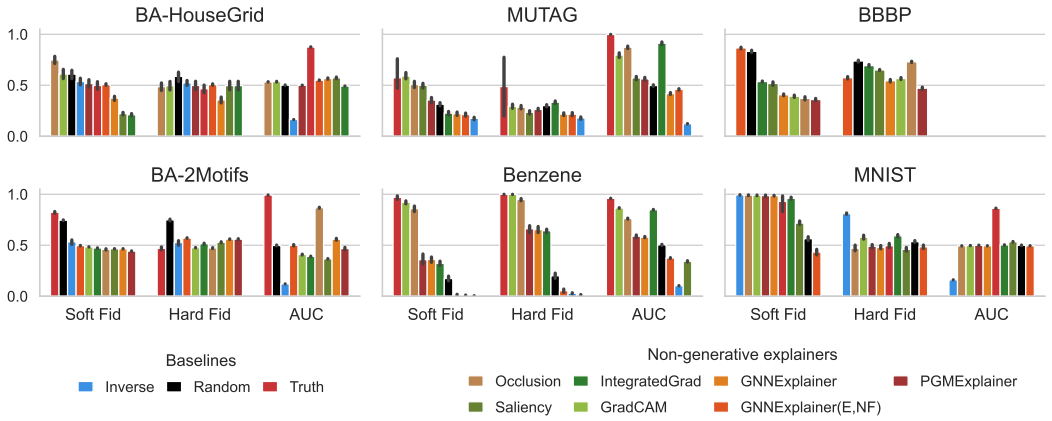

Figure 2: Rankings of base estimators and non-generative explainability methods according to the faithfulness score computed on soft explanations, the faithfulness score on hard explanations, and the AUC score. The AUC ranking is only reported for datasets with ground-truth explanations. Baselines were evaluated on the full explanatory masks, while explainability methods were evaluated on the truncated explanations, keeping the top 10 important undirected edges.

As a result, we cannot rely on the evaluation with faithfulness to draw general conclusions on the explainability methods. We compare the rankings of explainability methods according to the faithfulness evaluated on the two types of explanations, Hard Fidelity and Soft Fidelity respectively, and the accuracy score defined as the AUC score to stay consistent with previous work [18].

Observation 1 *The faithfulness metric is not consistent with the accuracy score.* In figure 2, there is a general misalignment in the rankings of explainers and base estimators on faithfulness or AUC score. For all datasets but Benzene, the Truth estimator, whose accuracy is maximal, has a small faithfulness score ~ 0.5. For MNISTbin, Inverse is by far the best according to the faithfulness score while being the worst explainer by definition on the AUC score. For BA-2Motifs, Random has the highest faithfulness score but can only be 50% accurate by definition. Due to the OOD problem of faithfulness, we cannot decide if the model is fooled by the subgraphs induced by the most informative edges or if human-based and model-based evaluations disagree. Therefore, we cannot quantify the alignment between human and model explainability.

Observation 2 *The evaluation of the explainability methods with the faithfulness metric is not consistent across datasets.* Figure 2 shows no consensus on the top-3 methods across datasets on the soft faithfulness or hard faithfulness score. For instance, we observe that GradCAM and Occlusion have the highest Soft Fid scores for BA-House-Grid, MUTAG, Benzene, and MNISTbin,

but not for BA-2Motifs and BBBP where Truth, Random, and GNNExplainer outperform. For Hard Fid, the results are also very heterogeneous among the six datasets. Due to the OOD problem, we cannot conclude that those inconsistencies across datasets are related to differences inherent to the graph data itself, e.g., differences in graph topology, size, or density among the datasets.

Observation 3 *The faithfulness metric is not consistent across edge removal strategies.* On figure 2, the top-3 ranking for Soft Fid and Hard Fid is always different except for Benzene dataset. This means that the edge removal strategy influences the model perception: the model does not predict labels only based on the information contained in the explanatory edges but also based on the structure of the given explanations. Because of the OOD problem, we cannot decide whether those inconsistencies come from the explainability methods themselves: methods that produce disconnected explanations are penalized by the hard removal strategy because the GNN model is not able to process the message passing.

4.2 Evaluating with GInX-Eval

4.2.1 Validating GInX-Eval on BA-HouseGrid dataset

We validate GInX-Eval on the BA-HouseGrid synthetic dataset. On Figure 3, GInX-Eval distinguishes the three base estimators because we observe a sharp decrease of the model performance after the removal of the top 10% edges considered important by the Truth explainer, while the Inverse estimator does not degrade the model performance, and the Random baseline starts decreasing after 20% of the edges are removed. Without fine-tuning, all base importance estimators lead to model performance degradation. Therefore, evaluating without fine-tuning the model cannot reflect the true explainability power of the methods.

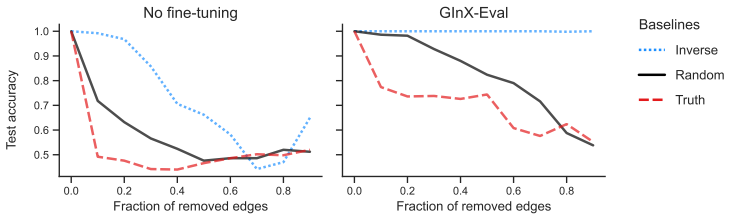

Figure 3: Comparison between not fine-tuning the GNN model and GInX-Eval on the BA-HouseGrid dataset. Without fine-tuning, the model's performance also decreases for the worst estimator Inverse where uninformative edges are removed first, preventing a correct evaluation of explainability methods. However, for GInX-Eval where the model is fine-tuned on modified datasets, we observe no test accuracy degradation for the worst-case estimator Inverse.

4.2.2 GInX-Eval of Baselines and Explainability Methods

GInX-Eval evaluates to what extent removing explanatory edges degrades the model accuracy. We adopt the hard selection strategy to remove edges. Even if conclusions are similar for both selection strategies (see Appendix A.3 and C.4), the degradation is of the order of 10^{-1} with hard selection versus 10^{-2} for soft selection. For visualization purposes, we prefer conveying here results with the hard selection. We refer the reader to Appendix C.4 for additional results with the soft selection. Figure 4 reports the GInX score when we remove a fraction $t \in [0.1, ..., 0.9]$ of the most important edges according to multiple explainability methods. For clarity, we choose to display a smaller set of explainability methods. For more methods, we refer the reader to Appendix C.1.

Truth, Inverse, and Random *Is the ground-truth explanation meaningful to the model?* The Truth and the Inverse edge importance estimators are evaluated on all datasets except BBBP which has no ground-truth available. We observe in figure 4 that the GInX score stays constant for Inverse and rises significantly for Truth. We conclude that the inverse explanations have only uninformative edges for the model while the ground-truth edges contain crucial information for making correct predictions. GInX-Eval is a useful tool to validate the quality of provided ground-truth explanations of published graph datasets. For all datasets except BA-HouseGrid, the Random baseline leads to a similar degradation as the Truth estimator. Random explanations

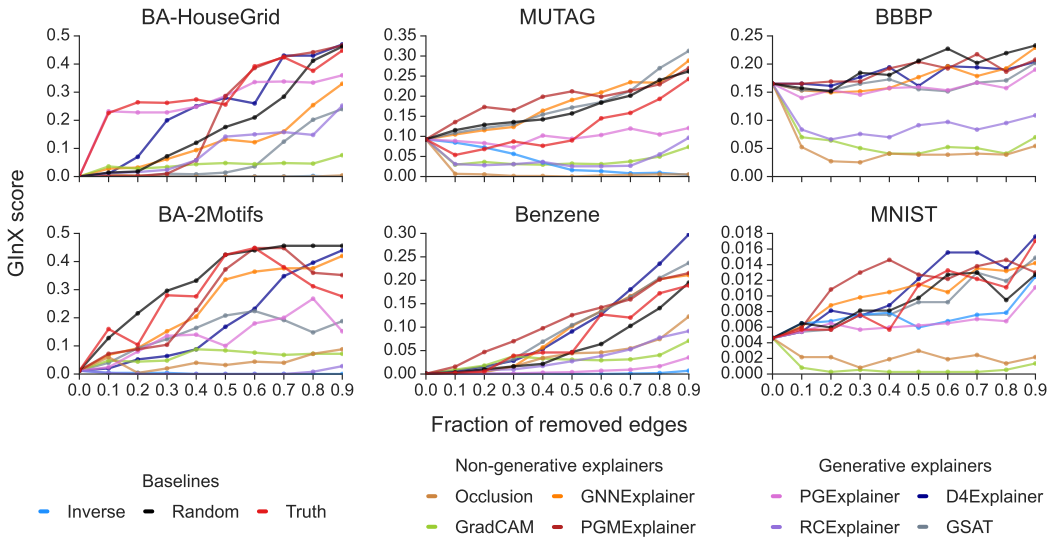

Figure 4: GInX scores of a fine-tuned GIN model on graphs with increasing fraction of removed edges. The removed edges are the most important based on explainability methods, and new input graphs are obtained with the *hard* selection, i.e., explanatory edges are strictly removed from the graph so that the number of edges and nodes is reduced. For more methods, see Appendix C.1.

contain enough informative edges so that removing just a few of them makes the GInX score increase sharply. This phenomenon is likely caused by strong correlations among informative edges.

Graph explainability methods *Are explanations obtained with graph explainability methods better than a random guess?* We observe that a random edge modification removes more informative edges than GradCAM, Occlusion, RCExplainer, and PGExplainer. GInX-Eval unveils new insights on gradient-based methods that go against recent studies [3, 33, 1]. On the other hand, D4Explainer, GNNExplainer, PGMExplainer, and GSAT for the non-synthetic datasets have competitive performance with Random and Truth baselines. This proves that generative methods are not necessarily better at capturing meaningful information than non-generative methods.

5 Discussion

This work discusses the pitfalls of faithfulness, one of the most popular metrics in xAI, and the problem of out-of-distribution explanations. To overcome these limitations, we propose a new evaluation procedure GInX-Eval that assesses the impact of removing the generated explanations from the graphs. Because GInX-Eval relies on a fine-tuning strategy of pre-trained black-box models, our method can easily be used for models only accessible via API calls, including large language models. Overall, GInX-Eval shows that gradient-based methods do not capture informative edges, contradicting the results of recent studies [3, 33, 1] reproduced in figure 2. More than evaluating methods, GInX-Eval can also validate the quality of provided ground-truth explanations and paves the way for developing approaches that conform with both human- and model-based explainability.

References

[1] Chirag Agarwal, Owen Queen, Himabindu Lakkaraju, and Marinka Zitnik. Evaluating explainability for graph neural networks. August 2022.

[2] Chirag Agarwal, Marinka Zitnik, and Himabindu Lakkaraju. Probing GNN explainers: A rigorous theoretical and empirical analysis of GNN explanation methods. June 2021.

[3] Kenza Amara, Rex Ying, Zitao Zhang, Zhihao Han, Yinan Shan, Ulrik Brandes, Sebastian Schemm, and Ce Zhang. Graphframex: Towards systematic evaluation of explainability methods for graph neural networks. *arXiv preprint arXiv:2206.09677*, 2022.

[4] Federico Baldassarre and Hossein Azizpour. Explainability techniques for graph convolutional networks. *CoRR*, abs/1905.13686, 2019.

[5] Chun Sik Chan, Huanqi Kong, and Liang Guanqing. A comparative study of faithfulness metrics for model interpretability methods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5029–5038, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[6] Jialin Chen, Kenza Amara, Junchi Yu, and Rex Ying. Generative explanation for graph neural network: Methods and evaluation. *extraction*, 54(12):47.

[7] Julien Colin, Thomas Fel, Remi Cadene, and Thomas Serre. What I cannot predict, I do not understand: A human-centered evaluation framework for explainability methods. December 2021.

[8] Ronan Collobert, Koray Kavukcuoglu, and Clément Farabet. Torch: A scientific computing framework for luajit. *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS)*, 24:237–245, 2011.

[9] Lukas Faber, Amin K Moghaddam, and Roger Wattenhofer. Contrastive graph neural network explanation. October 2020.

[10] Matthias Fey and Jan E. Lenssen. Fast graph representation learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019.

[11] Stephan Günnemann. Graph neural networks: Adversarial robustness. *Graph Neural Networks: Foundations, Frontiers, and Applications*, pages 149–176, 2022.

[12] Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. June 2018.

[13] Cheng-Yu Hsieh, Chih-Kuan Yeh, Xuanqing Liu, Pradeep Ravikumar, Seungyeon Kim, Sanjiv Kumar, and Cho-Jui Hsieh. Evaluation and methods for explanation through robustness analysis.

[14] Weihua Hu, Bowen Liu, Joseph Gomes, Marinka Zitnik, Percy Liang, Vijay Pande, and Jure Leskovec. Strategies for pre-training graph neural networks, 2020.

[15] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[16] Haoyang Li, Xin Wang, Ziwei Zhang, and Wenwu Zhu. OOD-GNN: Out-of-Distribution generalized graph neural network. December 2021.

[17] Peibo Li, Yixing Yang, Maurice Pagnucco, and Yang Song. Explainability in graph neural networks: An experimental survey. *arXiv preprint arXiv:2203.09258*, 2022.

[18] Antonio Longa, Steve Azzolin, Gabriele Santin, Giulia Cencetti, Pietro Liò, Bruno Lepri, and Andrea Passerini. Explaining the explainers in graph neural networks: a comparative study. *arXiv preprint arXiv:2210.15304*, 2022.

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

[20] Jing Ma, Ruocheng Guo, Saumitra Mishra, Aidong Zhang, and Jundong Li. Clear: Generative counterfactual explanations on graphs. *arXiv preprint arXiv:2210.08443*, 2022.

[21] Siqi Miao, Mia Liu, and Pan Li. Interpretable and generalizable graph learning via stochastic attention mechanism. In *International Conference on Machine Learning*, pages 15524–15543. PMLR, 2022.

[22] Meike Nauta, Jan Trienes, Shreyasi Pathak, Elisa Nguyen, Michelle Peters, Yasmin Schmitt, Jörg Schlötterer, Maurice van Keulen, and Christin Seifert. From anecdotal evidence to quantitative evaluation methods: A systematic review on evaluating explainable ai. *ACM Computing Surveys*, 2022.

[23] NVIDIA Corporation. NVIDIA CUDA Toolkit. `https://developer.nvidia.com/cuda-toolkit`, 2023.

[24] 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, 2019.

[25] Benjamin Sanchez-Lengeling, Jennifer Wei, Brian Lee, Emily Reif, Peter Y Wang, Wesley Wei Qian, Kevin Mc Closkey, Lucy Colwell, and Alexander Wiltschko. Evaluating attribution for graph neural networks. `https://papers.nips.cc/paper/2020/file/417fbbf2e9d5a28a855a11894b2e795a-Paper.pdf`. Accessed: 2021-11-22.

[26] Teague Sterling and John J Irwin. Zinc 15–ligand discovery for everyone. *Journal of chemical information and modeling*, 55(11):2324–2337, 2015.

[27] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *ICML*, volume 70, pages 3319–3328, 2017.

[28] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks, 2018.

[29] Minh N. Vu and My T. Thai. Pgm-explainer: Probabilistic graphical model explanations for graph neural networks. In *NeurIPS*, 2020.

[30] Xiang Wang, Yingxin Wu, An Zhang, Fuli Feng, Xiangnan He, and Tat-Seng Chua. Reinforced causal explainer for graph neural networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2022.

[31] Zhenqin Wu, Bharath Ramsundar, Evan N Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S Pappu, Karl Leswing, and Vijay Pande. Moleculenet: a benchmark for molecular machine learning. *Chemical science*, 9(2):513–530, 2018.

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

[33] 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, May 2023.

[34] Hao Yuan, Haiyang Yu, Jie Wang, Kang Li, and Shuiwang Ji. On explainability of graph neural networks via subgraph explorations. *ArXiv*, 2021.

[35] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13*, pages 818–833. Springer, 2014.

A Previous Evaluation Framework

A.1 Faithfulness metrics

Remove

$$fid-^{prob} = \frac{1}{N}\sum_{i=1}^{N} |p(f(G_i) = y_i) - p(f(h(G_i)) = y_i)|$$

$$fid-^{acc} = \frac{1}{N}\sum_{i=1}^{N} |\mathbb{1}(f(G_i) = y_i) - \mathbb{1}(f(h(G_i)) = y_i)|$$

Keep

$$fid+^{prob} = \frac{1}{N}\sum_{i=1}^{N} |p(f(G_i) = y_i) - p(f(G_i \setminus h(G_i)) = y_i)|$$

$$fid+^{acc} = \frac{1}{N}\sum_{i=1}^{N} |\mathbb{1}(f(G_i) = y_i) - \mathbb{1}(f(G_i \setminus h(G_i)) = y_i)|$$

Faithfulness or fidelity metrics [33] evaluate the contribution of the produced explanations to the initial prediction, either by giving only the explanation as input to the model (fidelity-) or by removing it from the entire graph and re-run the model (fidelity+). The keep/removing is done according to hard or soft selection (See section 3.1) and the explanatory edge mask. The fidelity scores capture how well an explainable model reproduces the natural phenomenon. The fidelity is measured with respect to the ground-truth label y. Equations A.1 detail the mathematical expressions of the different fidelity scores. We use the same notations as defined in section 3.1 where f represents the GNN model and h the explainability method. The fidelity scores (+/-) can be expressed either with probabilities ($fid^{prob}_{+/-}$) or indicator functions ($fid^{acc}_{+/-}$). While $fid^{prob}_{+/-}$ metrics are more appropriate for evaluating explanations in the context of regression tasks because they are only based on the predicted probabilities, $fid^{acc}_{+/-}$ metrics use the indicator function and are more suitable for classification problems.

A.2 The Out-Of-Distrbution Problem

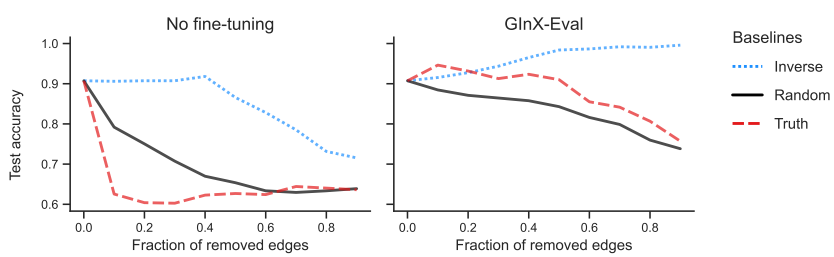

Figure 5: Comparison between not fine-tuning the GNN model and GInX-Eval on the MUTAG dataset when *hard* removing edges estimated informative by the three base estimators: Truth, Inverse and Random. New graphs are obtained with the *hard* selection strategy, i.e. edges are strictly removed from the initial graph structure.

The Inverse estimator is a good indicator of the out-of-distribution problem existence. When you remove uninformative edges and the model is not fine-tuned, the test accuracy should stay constant. But in the case of out-of-distribution explanations, the model has never seen these instances and therefore is not able to predict the correct labels anymore. A robust model toward OOD is a model in which test accuracy does not degrade with the Inverse estimator, i.e., as we remove uninformative edges.

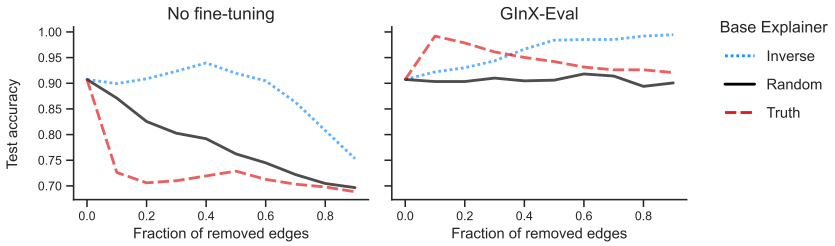

Figure 6: Comparison between not fine-tuning the GNN model and GInX-Eval on the MUTAG dataset when *soft* removing edges estimated informative by the three base estimators: Truth, Inverse, and Random. New graphs are obtained with the *soft* selection strategy, i.e. edges receive a zero weight if they must be removed.

In figures 5 and 6, we observe in both edge removal strategies a drop of model performance with the Inverse estimator when the model is not fine-tuned; but no degradation with GInX-Eval fine-tuning. Fine-tuning the GNN model at each degradation level overcomes the OOD problem and enables a robust evaluation.

Remark The degradation of the GNN test accuracy is smaller with fine-tuning than without. For MUTAG dataset we observe in figure 5 a drop in performance for the Truth estimator of 0.18 with GInX-Eval versus 0.25 with no fine-tuning at 90% edge removal. The model is remarkably robust to edge modification. After removing a large portion of all edges, the model still learns to make correct predictions and retain almost most of the original predictive power. For instance, for the MUTAG dataset, a random modification of 50% of the edges degrades the accuracy only to 85% accuracy. The model can extract meaningful representations from a small amount of remaining edges, which suggests that many edges are likely redundant or correlated.

A.3 Hard versus Soft Selection

In this section, we explore the effect of the selection strategy on GInX-Eval. With hard edge removal, the drop in test accuracy is significant, in the order of 10^{-1}. For the soft edge removal, the drop is very small in the order of 10^{-2} or 10^{-3} for some datasets. In figure 6 we observe for MUTAG dataset that, after removing 90% of the edges and when the model is fine-tuned, the accuracy drops by 10% for the Truth estimator in the case of soft selection, against 28% with the hard selection on figure 5. This suggests that removing informative edges by setting their weights to zeros does not prevent the model from learning from the masked information. For soft selection, the removed informative edges are actually not fully ignored. On the contrary, with hard selection, the model has no way of capturing the removed information since edges are fully removed from the structure. For this reason, we favor hard edge selection, even though similar effects are observed at different scales and similar conclusions are drawn with both types of selection strategies (See Appendix C.4).

A.4 Message Passing with edge weights

To understand why the soft selection enables the model to capture masked information and be robust to edge removal, we explain here how the soft selection affects the GNN training. In a weighted graph, each edge is associated with a semantically meaningful scalar weight. For soft explanations, for instance, the edge weights are importance scores. Graph neural networks integrate the graph topology information in forward computation by the message-passing mechanism. Edge weights modify the message from node i to node j in different ways according to the type of convolutional layer. For GAT convolutional layers, the attention scores are updated with additional edge-level attention coefficients that correspond to the edge weights, so that $\alpha = \alpha_{\text{node}} + \alpha_{\text{edge}}$. For GINE convolutional layers, the message is the node vector x_j followed by a ReLU activation. To account for edge importance, the message is modified by adding to the node vector the transformed edge weight $Lin(w_{ij})$. By modifying each message with its corresponding edge weight, the model keeps the whole graph connectivity but learns the importance of the node connections.

	Real-world				Synthetic	
Datasets	MUTAG	BBBP	Benzene	MNISTbin	BA-2Motifs	BA-HouseGrid
# Graphs	2,951	2,039	12,000	14,780	1,000	2,000
Avg # Nodes	30	24	20.6	63	25	101
Avg # Edges	61	51.9	43.7	486.6	51	817
# Node features	14	9	14	5	1	1
# Edge features	1	3	5	1	1	1
# Classes	2	2	2	2	2	2
GT Explanation	NH_2, NO_2	-	Benzene ring	Figure pixels	House, cycle	House, grid

Table 1: Datasets for graph classification

B Experimental details

B.1 Datasets

We evaluate the explainability methods on both synthetic and real-world datasets used for graph classification tasks.

BA-2Motifs [19] is a synthetic dataset with binary graph labels. The house motif and the cycle motif give class labels and thus are regarded as ground-truth explanations for the two classes.

BA-HouseGrid is our new synthetic dataset where we attach house or grid motifs to a Barabási base structure. We choose two distinct motifs, house and grid, where one is not the subgraph of the other, unlike BA-2Motifs with the cycle that is contained in the house motif. The datasets contain 2,000 graphs with balanced BA-Grid and BA-House graphs. The base Barabási graphs contain 80 nodes, and the number of motifs varies between 2 and 5 per graph.

MUTAG is a collection of $\sim$3,000 nitroaromatic compounds and it includes binary labels on their mutagenicity on Salmonella typhimurium. The chemical fragments -NO2 and -NH2 in mutagen graphs are labeled as ground-truth explanations [19].

BBBP includes binary labels for over 2,000 compounds on their permeability properties [31]. The task is to predict the target molecular properties. In molecular datasets, node features encode the atom type and edge features encode the type of bonds that connect atoms.

Benzene contains 12,000 molecular graphs extracted from the ZINC15 [26] database and labeled into two classes where the task is to identify whether a given molecule has a benzene ring or not. The ground-truth explanations are the nodes (atoms) comprising the benzene rings, and in the case of multiple benzenes, each of these benzene rings forms an explanation.

MNISTbin contains graphs that are converted from images in MNISTbin [15] using superpixels. In these graphs, the nodes represent the superpixels, and the edges are determined by the spatial proximity between the superpixels. The coordinates and intensity of the corresponding superpixel construct the node features. We reduce MNISTbin graph dataset for binary classification by selecting only the input of classes 0 and 1.

B.2 GNN Pre-Training

We test two GNN models: GIN [14] and GAT [28] because they score high on the selected real-world datasets, with a reasonable training time and fast convergence. The network structure of the GNN models for graph classification is a series of 3 layers with ReLU activation, followed by a max pooling layer to get graph representations before the final fully connected layer. We split the train/validation/test with 80/10/10% for all datasets and adopt the Adam optimizer with an initial learning rate of 0.001. Each model is initially trained for 200 epochs with an early stop. Each training is always repeated on five different seeds.

We report in Table 2 the test accuracy of both GIN and GAT models for all six datasets. We report the average and standard error of the experiments run on five different seeds. We observe high test accuracy, i.e., > 0.8, for all datasets for the GIN model and all real-world datasets for the GAT model. Only for BA-2Motifs and BA-HouseGrid is the test accuracy of GAT model not good enough to be tested in the explainability analysis and evaluated with GInX-Eval.

	BA-2Motifs	**BA-HouseGrid**	**BBBP**	**Benzene**	**MNISTbin**	**MUTAG**
GAT	$0.428_{\pm 0.068}$	$0.512_{\pm 0.034}$	$0.839_{\pm 0.024}$	$0.917_{\pm 0.044}$	$0.992_{\pm 0.001}$	$0.905_{\pm 0.022}$
GIN	$0.988_{\pm 0.016}$	$1.00_{\pm 0.000}$	$0.835_{\pm 0.035}$	$0.999_{\pm 0.001}$	$0.995_{\pm 0.002}$	$0.907_{\pm 0.016}$

Table 2: Test accuracy of the pre-trained GIN and GAT graph neural networks before fine-tuning for the six datasets on graph classification task.

B.3 GInX-Eval algorithm

Algorithm 1 GInX-Eval, an in-distribution evaluation procedure for explainability methods

Input: h: explainer function, f: pre-trained GNN model, $\mathcal{G}$: set of graph instances
Output: L: list of GInX scores at different degradation levels

function GInX-Eval $(h, f, \mathcal{G})$
Initialize an array to store evaluation results at different degradation levels
$L \leftarrow []$
for t in $[0.1, 0.2, ..., 1]$ with a step size of 0.1 **do**
    Gather explanations produced by the explainer for all graph instances
    $\mathcal{G}^* \leftarrow h(\mathcal{G})$
    Rank the explanatory edges based on their weights
    $ranked_edges \leftarrow$ sort_edges_by_weight$(\mathcal{G}^*)$
    Calculate the number of edges to remove based on the degradation level
    $N \leftarrow t \cdot |\mathcal{G}|$
    Remove the top t fraction of edges from the input graphs
    $\mathcal{G}_{\text{train}}, \mathcal{G}_{\text{test}} \leftarrow$ generate_train_test_datasets$(\mathcal{G}, ranked_edges, N)$
    Fine-tune the initial GNN model on the new train dataset
    $f \leftarrow$ fine_tune_GNN$(f, \mathcal{G}_{\text{train}})$
    Evaluate the fine-tuned model on the new test data
    $TestAcc \leftarrow$ evaluate_model$(f, \mathcal{G}_{\text{test}})$
    Calculate the GInX score
    $GInX \leftarrow 1 - TestAcc$
    Store the GInX score for this degradation level
    L.append$(GInX)$
end for
return L
end function

B.4 GInX-Eval computation time

Here we report the time for evaluating with GInX-Eval. GInX-Eval requires 10 GNN fine-tunings if we vary the threshold t from $[0, 0.1, ..., 0.9]$. In table 3, we report the average GNN fine-tuning time for GIN and GAT models for each dataset. The fine-tuned GNN models are trained in the same setting as the initial GNN model: the pre-trained model is fine-tuned for 200 epochs with an early stop. Each training is always repeated on five different seeds. In the case of the six datasets tested in this paper, the fine-tuning strategy has a low computational burden. In the case of large-scale datasets, selecting just a representative train/test subset can also speed up GInX-Eval.

	BA-2Motifs	BA-HouseGrid	BBBP	Benzene	MNISTbin	MUTAG
GAT	28.1	63.4	39.4	209.1	290.7	79.0
GIN	23.9	53.5	28.8	159.0	225.1	66.8

Table 3: Fine-tuning times (s) of GIN and GAT graph neural networks for the six datasets on graph classification task.

B.5 Generative Explainability Methods

Non-generative explainability methods like gradient-based or perturbation-based methods optimize individual explanations for a given instance. They lack a global understanding of the whole dataset as well as the ability to generalize to new unseen instances. To tackle this problem, non-generative methods have been developed. They learn the initial data distribution before generating individual explanations. Therefore, generative methods learn the underlying distributions of the explanatory graphs across the entire dataset, providing a more holistic approach to GNN explanations. The recent study of [6] shows the superiority of generative methods and in particular concerning their generalization capacity and faster inference time. GInX-Eval also proves that generative methods are the only type of methods that can do better than a random guess, and this is consistent across datasets and GNN models.

B.6 Code Implementation

Our code is implemented using torch-geometric 2.3.0 [10] and Torch 1.9.1 with CUDA version 11.1 [8, 23]. The generation of edge masks by the explainability methods and the training of the graph neural networks at each removal threshold are performed on a Linux machine with 1 GPUs NVIDIA RTX A6000 with 10 GB RAM memory. The code is available at `https://anonymous.4open.science/r/GInX-Eval`.

C Additional results

C.1 More explainability methods

Here, we show the GInX scores for all the tested methods: three baselines, eight non-generative methods, and five generative ones.

Why is the GInX score not increasing linearly? We found two possible reasons for the nonlinear rise of the GInX score when removing important edges. One reason is that edges might be redundant: some correlations exist between them and their effect is neutralized only if all are removed. Another reason is that important edges are not always correctly ordered or have no order of importance, e.g. Truth assigns a weight of 1 to all ground-truth edges. In these cases, the true least important edge among important edges according to the estimator might be removed first, not degrading the performance of the model that is still trained with the most informative edges.

C.2 Optimal threshold search

Figure 8 indicates the heterogeneity of masks generated by different explainability methods. While Truth, PGMExplainer, SubgraphX, and GraphCFE constrain their explanations to be sparse, the rest of the methods include most of the edges in the explanations, assigning a different importance weight to each edge. We define the critical threshold of a method as the fraction of non-zero values in the masks. After reaching the critical threshold, edges are randomly removed. Beyond the critical threshold, we are not evaluating the method anymore but a random importance attribution. Therefore, it is important to compare methods at a threshold t smaller than the minimum of the methods' critical thresholds.

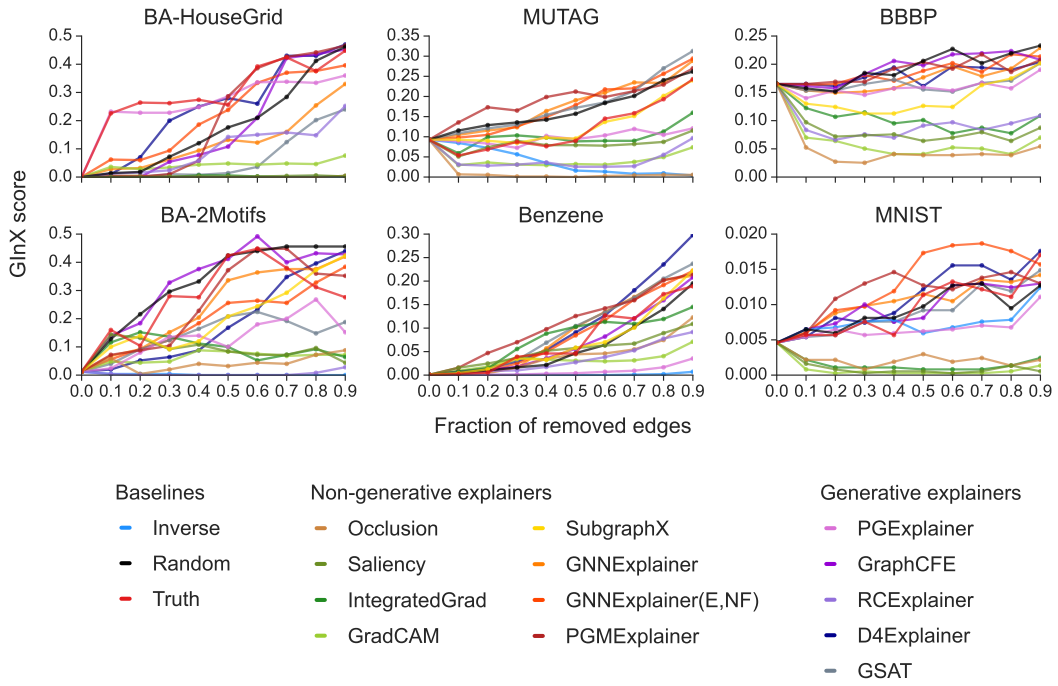

Figure 7: GInX scores of a fine-tuned GIN model on graphs with increasing fraction of removed edges. The removed edges are the most important based on explainability methods, and new input graphs are obtained with the *hard* selection, i.e. explanatory edges are strictly removed from the graph so that the number of edges and nodes is reduced.

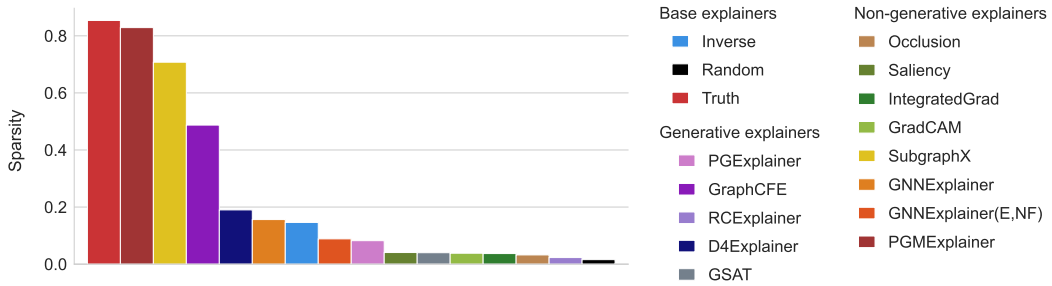

Figure 8: Mask sparsity of different explainability methods. A high sparsity indicates an explanatory mask with many zeros and small pre-processing explanatory subgraphs.

To compare methods fairly, we propose to define the dataset's optimal threshold t^*. This optimal threshold corresponds to the threshold closest to the average mask sparsity of ground-truth explanations. In other words, we take as reference the size of ground-truth explanations as the optimal number of informative edges in the graph and compare methods at this sparsity threshold. We compute the optimal thresholds for the six datasets and report them in table 4. Figure 9 displays the GInX scores of explainability methods at the optimal thresholds for each dataset. For the BBBP dataset, we have no ground-truth available and therefore no way of estimating the optimal threshold. We assume that the explanations contain a maximum of 30% of the bonds in the molecules because we expect human-intelligible sparse explanations. Except for Benzene dataset, we observe that gradient-based methods (in green colors) and Occlusion (in light brown) have the smallest GInX scores. This supports the previous conclusions drawn in the paper, namely that gradient-based methods contain less informative edges than GNNExplainer, PGMExplainer, and generative methods.

	BA-2Motifs	BA-HouseGrid	BBBP	Benzene	MNISTbin	MUTAG
Truth sparsity	0.216	0.065	-	0.175	0.235	0.039
Optimal threshold	0.3	0.1	-	0.2	0.3	0.1

Table 4: The truth mask sparsity is used to estimate the optimal threshold to compare GInX scores of diverse methods. The ground-truth explanatory mask sparsity is computed as the average sparsity of the masks generated by the truth edge importance estimator. We deduce the optimal threshold t^* at which explainability methods can be compared by rounding up the mask sparsity to the nearest multiple of 0.01.

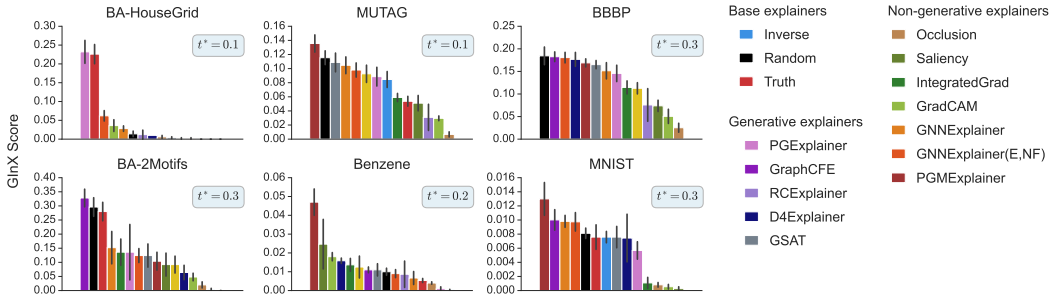

Figure 9: GInX scores at the optimal thresholds. For the BBBP dataset, we define an arbitrary optimal threshold $t = 0.3$. For the other datasets, the optimal threshold is estimated based on the explanatory mask sparsity generated by the Truth estimator.

C.3 HomophilicRank score

Based on the GInX score, we can compute the power of explainability methods to rank edges, i.e., to correctly order edges based on their importance and give identical importance to correlated edges. The edge homophilic ranking power can be evaluated with the HomophilicRank score defined as:

$$\text{HomophilicRank} = \sum_{t=0,0.1,\ldots,0.8} (1-t) \times (\text{GInX}(t+0.1) - \text{GInX}(t)) \tag{4}$$

The HomophilicRank score measures the capacity of a method to rank edges by their correct importance ordering while assigning similar importance weights to correlated edges. A high score characterizes methods that (1) assign the highest importance weights to the most genuinely informative edges for the model and (2) treat correlated edges on equal footing. It measures the capacity to uniformly assign importance to redundant information rather than only putting importance to a single representative edge and none to the correlated edges. This score penalizes methods that for instance only discover a subset of important edges and do not account for their correlated edges. This is especially important when you try to characterize an explanation and identify fundamental entities within the explanatory substructure.

We use the HomophilicRank score to evaluate the capacity of explainers to rank edges correctly according to their true informativeness for the model and identify redundant information. In figure 10, we observe that gradient-based methods and Occlusion are not good at correctly ordering edges by their importance. This is another reason why they should not be used to generate meaningful explanations. We also observe that RCExplainer and PGExplainer which perform well on the GInX score have a low edge homophilic ranking power, except for the BA-HouseGrid dataset. These two methods can capture the most informative edges but cannot decide what the relative importance of those important edges is. Finally, PGMExplainer, GNNexplainer, GraphCFE, and D4Explainer have both a high GInX score (see figure 4) and a high HomophilicRank score, making them the best choice for informative and edge-rank powerful explainers. With the HomophilicRank score, GInX-Eval indicates what method can better rank edges according to their

informative power, assigning all correlated edges a similar importance. This helps to compare methods that already perform well on GInX score.

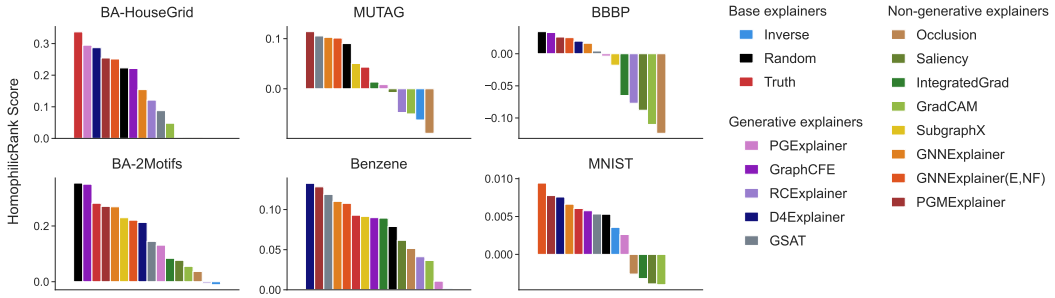

Figure 10: Edge homophilic ranking power of explainability methods

C.4 GInX-Eval with Soft Selection

In the paper, we describe the different selection strategies to remove edges from a graph (see section 3.1) and explain our preference for the *hard* selection to convey the results in section 4.2. Here, we add the results of GInX-Eval for *soft* selection. The experimental setting is the same. We display the results for the four real-world datasets.

Compared to the results with hard selection, the growth of GInX score here is way smaller, in the order of 0.01, as mentioned in the paper. Instead of observing a rise of the GInX score for the best methods, we observe that it stays constant. The impact of masking informative edges does not prevent the model from learning because the model can still pass the message on the node level. For more details, we refer the reader to the discussion in Appendix A.4. For the worse methods, the GInX score decreases instead of staying constant. Removing uninformative edges by setting their weight to zero brings the model to better learn the actual class of the graphs. Only for Benzene, the Random and Truth baselines as well as most of the generative methods produce a ~ 0.07 rise of the GInX score.

Results on MNISTbin dataset cannot be interpreted since the increase of the GInX score is of the order of 10^{-3} and Truth and Inverse estimators have the same behavior. For the three other datasets Benzene, MUTAG, and BBBP, figure 11 confirms the superiority of the non-generative methods GNNExplainer and PGMExplainer, and the generative methods GSAT, GraphCFE, and D4Explainer. Once again Occlusion and the gradient-based methods Saliency, Integrated Gradient, and GradCAM capture the less informative edges, contradicting the previous studies on explainability for GNN [33, 3, 1].

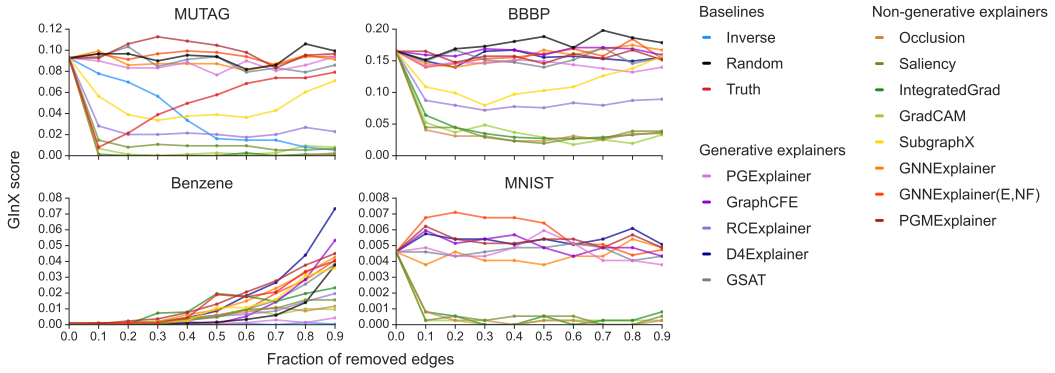

Figure 11: GInX scores of a fine-tuned GIN model on graphs with increasing fraction of removed edges. The removed edges are the most important based on explainability methods, and new input graphs are obtained with the *soft* selection, i.e. explanatory edges are masked so that their respective weight is put to zero.