On the Consistency of GNN Explainability Methods

Ehsan Hajiramezanali*, Sepideh Maleki, Alex Tseng, Aïcha Bentaieb,

Gabriele Scalia, Tommaso Biancalani

Genentech
hajiramezanali.ehsan@gene.com

Abstract

Despite the widespread utilization of *post-hoc* explanation methods for graph neural networks (GNNs) in high-stakes settings, there has been a lack of comprehensive evaluation regarding their quality and reliability. This evaluation is challenging primarily due to the data's non-Euclidean nature, arbitrary size, and complex topological structure. In this context, we argue that the *consistency* of GNN explanations, denoting the ability to produce similar explanations for input graphs with minor structural changes that do not alter their output predictions, is a key requirement for effective *post-hoc* GNN explanations. To fulfill this gap, we introduce a novel metric based on Fused Gromov–Wasserstein distance to quantify consistency. Finally, we demonstrate that current methods do not perform well according to this metric, underscoring the need for further research on reliable GNN explainability methods.

1 Introduction

Graph neural networks (GNNs) [19, 17, 13] have experienced growing success across various domains, including molecular chemistry, biological networks, and recommendation systems [12, 14, 23, 16]. However, their ability to learn intricate functions of structured inputs often comes at the price of limited comprehensibility of the resulting model. GNNs frequently require millions to billions of operations to transform input graphs into predictions. In critical fields like healthcare, where deploying these models has significant implications, their lack of interpretability poses a significant challenge [30]. To address this issue, many *post-hoc explainability methods* have been developed to explain GNNs predictions[35, 37].

As the number of proposed methods for explaining GNNs continues to increase, it is crucial to ensure their reliability. However, the field of explainability in graph machine learning is still in its early stages, lacking standardized evaluation strategies and dependable data resources to assess, test, and compare GNN explanations [3, 2, 1]. Commonly used metrics such as explanation fidelity heavily rely on implementation details of the explanation method (e.g., the perturbation function) and do not provide a true picture of the explanation quality [3].

Although a few studies—e.g., Agarwal et al. [3]—have recognized this challenge, they often rely on synthetic datasets with limited ground-truth explanations for their analyses. However, depending solely on synthetic data and associated ground-truth explanations is inadequate, as they do not represent the diverse range of real-world graph datasets [20]. Furthermore, it is crucial to acknowledge that multiple underlying justifications can result in the same correct class labels, leading to redundant or non-unique explanations [20]. When a GNN model is trained, it may only capture one of these justifications or even rely on an entirely different rationale

*Corresponding author.

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

altogether, similar to how different humans may have different valid and correct justifications for the same decision. Consequently, evaluating the explanation produced by a state-of-the-art method using "the ground-truth explanation" is incorrect since the underlying GNN model may not depend on that specific ground-truth explanation.

Given this, we argue that *consistency to small structural perturbations* is an essential property that interpretability methods for GNNs should satisfy to generate meaningful explanations. At its most intuitive level, this requirement suggests that similar graphs with *identical GNN predictions* should not lead to significantly different explanations. There are two key arguments supporting the importance of consistency as a crucial property for *post-hoc* GNN explainability methods. Firstly, for an explanation to be considered valid in a particular context, it should remain relatively stable in its immediate surroundings, regardless of how it is represented (e.g., as saliency, a score function, or a linear model). On the other hand, if we aim to obtain an explanation that holds predictive power, then the consistency of the simplified model suggests that it can serve as an approximate substitute for the full complex GNN, at least within a small neighborhood.

In this context, the objective of this study is to examine whether widely used GNN explainability methods adhere to the principle of consistency. To achieve this, we initially formalize the concept of consistency that aligns with our intuitive understanding, as detailed in the following sections. Then, we evaluate the performance of different popular GNN explainers based on this criterion both quantitatively and qualitatively. Furthermore, we introduce a novel consistency benchmark dataset sourced from the real-world MUTAG dataset. This dataset encompasses 18 molecules exhibiting nearly identical structures, carefully chosen to assess the consistency of GNN explanations. Lastly, we summarize our findings and explore potential approaches to enhance consistency in GNN explainability methods.

2 Preliminaries

Capturing the similarity between structured data is a challenging endeavor that involves harnessing both feature and structural information [11]. This task requires finding ways to associate these two types of information to effectively leverage their combined power. Titouan et al. [27] addresses the problem of computing distances between structured objects, specifically undirected graphs, by treating them as probability distributions within a specific metric space. They introduce a transportation distance that minimizes the overall cost of transporting probability masses, thereby revealing the underlying geometric nature of the space where structured objects reside. This approach, known as Fused Gromov–Wasserstein (FGW) distance, stands out from metrics like Wasserstein or Gromov–Wasserstein, which solely focus on either node attributes (by employing a metric in the feature space) or structure (by considering structure as a metric space). Instead, FGW distance jointly exploits both feature and structural information [14, 7, 15].

2.1 Graphs as probability measures

Let us define undirected graphs as tuples in the form of $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \ell_f, \ell_s)$, where $(\mathcal{V}, \mathcal{E})$ represents the set of vertices and edges of the graph. The node-attribute function $\ell_f : \mathcal{V} \to \Omega_f$ associates each vertex $v_i \in \mathcal{V}$ with a feature $a_i \stackrel{\text{def}}{=} \ell_f(v_i)$ in a feature metric space (Ω_f, d). We will refer to the collection of all the features $(a_i)_i$ of the graph as the *feature information*. Similarly, the function $\ell_s : \mathcal{V} \to \Omega_s$ maps each vertex v_i in the graph to its structure representation $x_i \stackrel{\text{def}}{=} \ell_s(v_i)$ in a structure space (Ω_s, C) that is specific to each graph. Here, $C : \Omega_s \times \Omega_s \to \mathbb{R}_+$ is a symmetric function that measures the similarity between nodes in the graph. However, unlike the feature space, Ω_s is implicit, and in practical terms, knowing the similarity measure C is sufficient. For simplicity, we will use C to denote both the structure similarity measure and the matrix that encodes this similarity between pairs of nodes in the graph $(C(i,k) = C(x_i, x_k))_{i,k}$, allowing for a slight abuse of notation.

The nature of the similarity measure C depends on the context. It can represent various aspects such as the neighborhood information of the nodes, the edge information of the graph, or, more generally, it can capture the distance between nodes, such as the shortest-path distance or the harmonic distance [29, 27, 15]. In cases where C functions as a metric, like the shortest-path distance, we generally consider the structure equipped with the metric space (Ω_s, C). The

collection of all structure embeddings $(x_i)_i$ of the graph will be referred to as the *structure information*.

Titouan et al. [27] and Chen et al. [7] propose an enrichment to the previously described graph by introducing a histogram that indicates the relative importance of the vertices in the graph. To achieve this, assuming the graph has n vertices, each vertex is assigned a weight $(h_i)i \in \Sigma n$. This leads to the concept of *structured data*, represented by a tuple $\mathcal{S} = (\mathcal{G}, h_\mathcal{G})$, where $\mathcal{G}$ is the graph as previously defined, and $h_\mathcal{G}$ is a function that assigns a weight to each vertex. With this definition, the entire structured data can be described by a fully supported probability measure over the product space feature/structure, denoted by $\mu = \sum_{i=1}^{n} h_i \delta_{(x_i,a_i)}$. This measure captures the complete information of the structured data. When all the weights are equal (i.e., $h_i = \frac{1}{n}$), indicating that all vertices have the same relative importance, the structured data retains the same information as the original graph. However, assigning different weights can encode *a priori* information or biases.

Titouan et al. [27] and Chen et al. [7] propose an enrichment to the previously described graph by introducing a histogram that indicates the relative importance of the vertices in the graph. To achieve this, assuming the graph has n vertices, each vertex is assigned a weight $(h_i)i \in \Sigma n$. This leads to the concept of *structured data*, represented by a tuple $\mathcal{S} = (\mathcal{G}, h_\mathcal{G})$, where $\mathcal{G}$ is the graph as previously defined, and $h_\mathcal{G}$ is a function that assigns a weight to each vertex. With this definition, the entire structured data can be described by a fully supported probability measure over the product space feature/structure, denoted by $\mu = \sum_{i=1}^{n} h_i \delta_{(x_i,a_i)}$. This measure captures the complete information of the structured data. When all the weights are equal (i.e., $h_i = \frac{1}{n}$), indicating that all vertices have the same relative importance, the structured data retains the same information as the original graph. However, assigning different weights can encode *a priori* information or biases.

2.2 Fused Gromov–Wasserstein distance

In the context of measuring the distance between two graphs $\mathcal{G}_1$ and $\mathcal{G}_2$, we describe them by their respective probability measures, $\mu = \sum_{i=1}^{n} h_i \delta_{(x_i,a_i)}$ and $\nu = \sum_{i=1}^{m} g_j \delta_{(z_j,b_j)}$, where $h \in \Sigma_n$ and $g \in \Sigma_m$ are distributions. We assume, without loss of generality, that $(x_i, a_i) \neq (x_j, a_j)$ for $i \neq j$ (similarly for z_j and b_j).

Let $\Pi(h, g)$ be the set of all valid couplings between h and g, defined as:

$$\Pi(h,g) = \{\pi \in \mathbb{R}_+^{n \times m} \text{ s.t. } \sum_{i=1}^{n} \pi_{i,j} = h_j \ , \sum_{j=1}^{m} \pi_{i,j} = g_i\}, \tag{1}$$

where $\pi_{i,j}$ represents the amount of mass transferred from bin h_i to g_j for a particular coupling π. The matrix π represents a probabilistic matching of the nodes between the two graphs. $M_{AB} = (d(a_i, b_j))_{i,j}$ is a $n \times m$ matrix that represents the distances between the features a_i and b_j. The structure matrices are represented by C_1 and C_2. The marginals of μ and ν with respect to the structure and feature are denoted as μ_X, μ_A (respectively, ν_Z, ν_B). To quantify the similarity between the structures, we utilize the 4-dimensional tensor $L(C_1, C_2)$. This tensor compares pairwise distances within each graph and is defined as follows:

$$L_{i,j,k,l}(C_1, C_2) = |C_1(i,k) - C_2(j,l)|. \tag{2}$$

The **FGW distance**, regarded as an Optimal Transport discrepancy [27], is defined with a trade-off parameter $\alpha \in [0, 1]$ as follows:

$$\mathrm{FGW}_{q,\alpha}(\mu,\nu) = \min_{\pi \in \Pi(h,g)} E_q(M_{AB}, C_1, C_2, \pi) \tag{3}$$

where

$$\begin{aligned} E_q(M_{AB}, C_1, C_2, \pi) &= \left\langle (1-\alpha) M_{AB}^q + \alpha L(C_1, C_2)^q \otimes \pi, \pi \right\rangle \\ &= \sum_{i,j,k,l} (1-\alpha) d(a_i, b_j)^q + \alpha |C_1(i,k) - C_2(j,l)|^q \pi_{i,j} \pi_{k,l}. \end{aligned} \tag{4}$$

The FGW distance aims to find a coupling π between the vertices of the graphs that minimizes the cost function E_q. This cost function is a linear combination of the cost $d(a_i, b_j)$ associated

with transporting a feature a_i to a feature b_j and the cost $|C_1(i,k) - C_2(j,l)|$ associated with transporting pairs of nodes within each structure.

The optimal coupling in FGW tends to associate pairs of features and structure points that have similar distances within each structure pair and similar features [27]. This property allows FGW to handle structured data with continuous-attributed or discrete-labeled nodes, thanks to the definition of the cost function d. Additionally, FGW can be computed even when the graphs have a different number of nodes, making it a versatile measure for comparing subgraph outputs of GNN explainers [15].

3 Consistency of GNN explanation methods

We are interested in the concept of consistency regarding the variations in the explanation of a GNN prediction when there are changes in the structure of the input graph leading to that prediction. In simpler terms, if we make slight modifications to the input graph's structure without *significantly altering the model's prediction*, we expect the subgraph explanation given by GNN explainers for the modified graph to remain relatively unchanged. However, the crucial finding and primary motivation of this study indicate that most current *post-hoc* GNN explanation methods do not exhibit this desired consistency.

Figure 1 shows the explanations provided by three popular perturbation-based methods, SubgraphX [36], GStarX [37], and PGExplainer [22], for the predictions of a Graph Convolutional Network (GCN) classifier on a real-world MUTAG [35, 33] dataset. As expected, their generated subgraphs are fairly stable when explaining non-mutagenic molecules with simpler structures, but for more complex molecules, they yield explanations that are considerably different for very similar molecules and are often inconsistent with each other.

The inconsistency demonstrated in Figure 1 is the phenomenon we seek to investigate. Visual inspection of different graphs, although illustrative, is subjective and infeasible for more complex graphs. To conclusively gauge this (lack of) consistency, we need objective tools to quantify it. In this paper, we turn the task of evaluating the consistency of GNN explainers into a graph-matching problem [31]. We do this by assessing the distance between subgraph outputs relative to the distance of their input graphs. Specifically, we propose to incorporate *FGW distance* as a parametric notion of consistency that measures relative changes in the generated subgraphs with respect to the input graphs.

Definition 3.1. Let $f(\cdot) : \mathcal{G} \longrightarrow \mathcal{Y}$ be a trained GNN model that maps an input graph $G \in \mathcal{G}$ to its predicted class $y \in \mathcal{Y}$. The explanation model, denoted as ψ_{exp}, generates the subgraph $G^* = \psi_{\mathsf{exp}}(f(\cdot), G, y)$ in order to explain the GNN prediction y for the input graph G.

We propose to quantify the consistency of an explanation model ψ_{exp} for each graph $G_i \in \mathcal{G}$ as

$$L_{\psi_{\mathsf{exp}}}(G_i) = \frac{1}{|\mathcal{N}(G_i)|} \sum_{G_j \in \mathcal{N}_\varepsilon(G_i)} \left(\frac{\mathsf{FGW}(G_j^{(*)}, G_i^{(*)})}{\mathsf{FGW}(G_j, G_i) + \epsilon} \right), \tag{5}$$

where $\mathcal{N}_\varepsilon(G_i) = \{G_j \in \mathcal{G} \,|\, \mathsf{FGW}(G_j, G_i) \leq \varepsilon \wedge p(y_j) \approx p(y_i)\}$.

Remark 3.2. Please note that $L_{\psi_{\mathsf{exp}}}$ in Definition 3.1 is a sample-dependent quantity, and there is no unique *ideal* quantity that is universally desirable. The desirability of this quantity depends on the specific application and the objective of explainability. In this context, we compare different methods by making relative assessments.

Remark 3.3. Please note that in 5, we have the flexibility to utilize a different distance metric instead of FGW. For instance, we also employed the Euclidean distance on the latent representations of two graphs as an alternative baseline. However, there are several advantages to using FGW: i) It does not introduce additional sources of bias, such as the need for additional unsupervised/self-supervised graph learning to obtain graph representations. ii) FGW does not necessitate equal graph sizes for calculating distances. iii) FGW can simultaneously account for changes in both graph structure and node attributes.

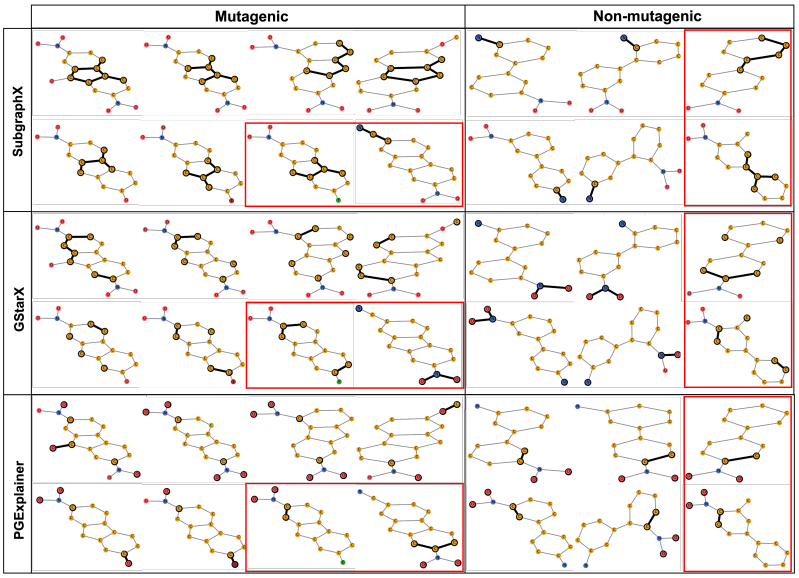

Figure 1: The GCN explainability for 14 molecules with minor structural changes from the MUTAG dataset. Carbon, oxygen, and nitrogen atoms are highlighted in yellow, red, and blue, respectively. The top, middle, and bottom rows display the SubgraphX, GstarX, and PGExplainer explanations. The molecules enclosed in red boxes represent instances misclassified by the GCN.

4 Experiments

In addition to the previously mentioned SubgraphX, GStarX, and PGExplainer, we also assess the consistency of perturbation-based GNN explanation methods, namely GNNExplainer and GraphSVX. These evaluations utilize implementations of these methods from the DIG package [21] and GStarX [37].

To establish a new consistency benchmark dataset, we leverage the real-world MUTAG molecule dataset [8]. We identify 18 molecules with nearly identical structures, with 10 of them being labeled as Mutagenic. This dataset forms the basis for our assessment. Furthermore, we extend our analysis to evaluate the consistency of various GNN explainers using the proposed metric on Graph-Twitter [35]. This non-molecular dataset demonstrates the generality of our evaluation method beyond molecular data.

As a baseline to demonstrate the effectiveness of the proposed FGW-based consistency metric, we also employ the Euclidean distance (ED) between the graph representations of the MUTAG dataset. To achieve this, we first trained a graph contrastive learning model (graphCL) [34] on the MUTAG dataset. Subsequently, we embed the identified subgraphs by passing them through the pretrained graphCL encoder. Finally, we compute the Euclidean distance between these two embedding vectors.

Note. The $L_{\psi_{\text{exp}}}$ metric captures the consistency based on the FGW distance between generated subgraphs. We also calculate $L^{(o)}_{\psi_{\text{exp}}}$ based on FGW distance between $G^{(o)} = G \setminus G^{(*)}$ which have been extracted by removing identified subgraphs from the input graphs.

Table 1 shows a consistency comparison among various GNN explainers for both $L_{\psi_{\text{exp}}}$ and $L^{(o)}\psi\text{exp}$. In the context of FGW, lower values are preferred as they indicate that the generated subgraphs are highly similar for graphs with similar input structures and the same GNN predictions. According to this metric, SubgraphX outperforms the other baselines, possibly because it excels in identifying connected subgraphs. Table 2 shows a similar comparison for the Graph-Twitter dataset. These results demonstrate the effectiveness of the proposed consistency metric across various types of graphs.

FGW is a robust metric. When comparing the ED metric to FGW, we observe that the reliability of ED can be suboptimal in certain cases, such as non-mutagenic samples in SubgraphX. Specifically, Figure 1 shows that SubgraphX has identified carbon–nitrogen bonds as the important

Table 1: Consistency comparison of different perturbation-based GNN explanability methods over the proposed MUTAG benchmark dataset based on FGW and Euclidean distance (ED).

Metric	Method	Mutagenic		Non-mutagenic		Both classes	
		$L_{\psi_{\text{exp}}}$	$L^{(o)}_{\psi_{\text{exp}}}$	$L_{\psi_{\text{exp}}}$	$L^{(o)}_{\psi_{\text{exp}}}$	$L_{\psi_{\text{exp}}}$	$L^{(o)}_{\psi_{\text{exp}}}$
FGW (↓)	SubgraphX	**0.21± 0.15**	**82.62.93± 49.77**	**0.37± 0.07**	**99.10±18.03**	**0.24± 0.15**	**85.92± 45.72**
	GstarX	54.35± 23.41	261.17± 112.58	20.32± 9.59	407.83± 204.74	47.55± 25.34	290.50± 148.21
	PGExplainer	15.39± 11.33	112.10± 74.91	14.55± 9.49	115.13± 69.95	15.09± 10.72	113.17± 73.2
	GNNexpainer	57.45± 28.57	172.17± 86.87	63.03± 28.74	169.28± 75.10	59.43± 28.76	171.15± 82.90
	GraphSVX	34.22± 21.23	164.02± 58.38	31.12± 18.34	176.66± 55.05	33.12± 20.30	168.50± 57.54
ED (↑)	SubgraphX	**0.99± 0.05**	0.94± 0.05	0.25± 0.20	0.77±0.10	0.84± 0.31	0.91± 0.09
	GstarX	0.94± 0.07	0.98± 0.07	0.67± 0.17	0.55± 0.35	0.88± 0.14	0.89± 0.23
	PGExplainer	**0.99± 0.02**	**0.99± 0.03**	**0.99± 0.02**	**0.99± 0.02**	**0.99± 0.03**	**0.99± 0.02**
	GNNexpainer	0.90± 0.08	0.83± 0.15	0.92± 0.06	0.89± 0.09	0.91± 0.07	0.86± 0.14
	GraphSVX	0.85± 0.12	0.92± 0.07	0.87± 0.13	0.95± 0.05	0.86± 0.13	0.93± 0.06

subgraphs for all non-mutagenic samples with correct GNN predictions. Therefore, we expect to obtain high values for their similarities. However, the ED-based similarities of subgraph embeddings for non-mutagenic samples, as identified by SubgraphX, are quite low (Table 1). We posit that this discrepancy may be attributed to a distribution shift in the identified subgraphs. In such cases, the identified subgraphs are quite small, and the GraphCL encoder may struggle to correctly capture their representations due to generalizability issues. This raises questions about the effectiveness of the ED-based metric for consistency evaluation. On the other hand, FGW operates within the primary domain and does not suffer from out-of-distribution generalizability issues, making it a *robust metric* for evaluating consistency.

Subgraph discrepancies between misclassified and correctly classified samples. We then investigate the similarity between subgraph explanations of similar graphs which were correctly classified versus misclassified. As described above, GNN explainers should be *consistent*, in that similar graphs which were correctly classified should have similar subgraph explanations. In a related vein, we also believe that if two graphs are similar, but one is correctly predicted, and the other is misclassified, then the explanations should be *different*. To explore this, we calculate the percentage increase in FGW distance for misclassified subgraphs compared to correctly classified ones. Figure 2 shows that SubgraphX outperforms other GNN explainers in this regard. However, the identified subgraphs for misclassified samples based on PGExplainer, GNNexplainer, and GstarX are almost identical to their correctly classified counterparts.

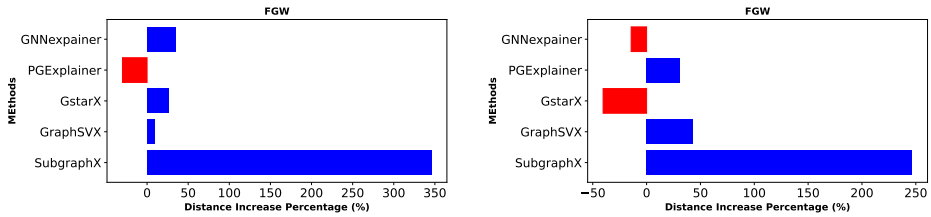

Figure 2: The percentage increase in FGW distance for misclassified subgraphs with respect to correctly classified ones for **(Left)** $L_{\psi_{\text{exp}}}$ and **(Right)** $L^{(o)}_{\psi_{\text{exp}}}$. Higher percentages are desirable as they indicate that the explanations for two similar inputs but different GNN predictions are different.

5 Discussion

In this study, our primary objective was to assess the consistency of widely used perturbation-based GNN explanation frameworks when subjected to minor alterations in the input graph (assuming that the GNN predictions remained unchanged). To evaluate this consistency, we introduced a novel metric based on Fused Gromov–Wasserstein. Our findings demonstrate that this metric exhibits fewer bias-related issues in comparison to distance computation methods relying on pre-trained encoders. Our experiments have revealed that, in general, existing frameworks lack consistency. SubgraphX demonstrates (surprisingly) greater consistency compared to its structure-aware counterpart, GStarX. This advantage may be due to the fact that SubgraphX explanations primarily consist of *connected subgraphs*, whereas the other explanation methods tend to identify disconnected subgraphs. However, it is worth noting that this advantage of

SubgraphX comes at the cost of higher computational complexity. Specifically, SubgraphX is 2.5 times slower than GStarX and 25 times slower than GraphSVX.

Table 2: Consistency comparison of different perturbation-based GNN explanability methods on Graph-Twitter dataset.

Method	$L_{\psi_{\text{exp}}}$	$L^{(o)}_{\psi_{\text{exp}}}$
SubgraphX	**0.00± 0.00**	1.15±0.43
GstarX	0.10± 0.13	0.92± 0.39
PGExplainer	0.13± 0.12	1.18± 0.49
GNNexpainer	0.31± 0.21	0.19± 0.22
GraphSVX	0.33± 0.35	0.96± 0.41

Our results suggested that GNN explainers that generate connected subgraphs—such as SubgraphX—can contribute to improved consistency. Several other directions may also be explored in the future to produce more consistent GNN explanations. Firstly, incorporating uncertainty quantification [24] for generated subgraphs could enhance consistency. Secondly, explanations based on the probability output of all classes, rather than just the predicted class, could be beneficial, especially when the model exhibits uncertainty. As more GNN explanation methods are developed, incorporating techniques such as the aforementioned ones to improve their consistency may lead to better-quality explanations, toward more trustable models overall.

References

[1] Julius Adebayo, Michael Muelly, Ilaria Liccardi, and Been Kim. Debugging tests for model explanations. *arXiv preprint arXiv:2011.05429*, 2020.

[2] Julius Adebayo, Michael Muelly, Harold Abelson, and Been Kim. Post hoc explanations may be ineffective for detecting unknown spurious correlation. In *International conference on learning representations*, 2021.

[3] Chirag Agarwal, Owen Queen, Himabindu Lakkaraju, and Marinka Zitnik. Evaluating explainability for graph neural networks. *Scientific Data*, 10(1):144, 2023.

[4] David Alvarez-Melis and Tommi S Jaakkola. On the robustness of interpretability methods. *arXiv preprint arXiv:1806.08049*, 2018.

[5] Hugh Chen, Ian C Covert, Scott M Lundberg, and Su-In Lee. Algorithms to estimate shapley value feature attributions. *Nature Machine Intelligence*, pages 1–12, 2023.

[6] Jianbo Chen, Le Song, Martin J Wainwright, and Michael I Jordan. Learning to explain: An information-theoretic perspective on model interpretation. In *International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 882–891. PMLR, 2018.

[7] Liqun Chen, Zhe Gan, Yu Cheng, Linjie Li, Lawrence Carin, and Jingjing Liu. Graph optimal transport for cross-domain alignment. In *International Conference on Machine Learning*, pages 1542–1553. PMLR, 2020.

[8] Asim Kumar Debnath, Rosa L Lopez de Compadre, Gargi Debnath, Alan J Shusterman, and Corwin Hansch. Structure-activity relationship of mutagenic aromatic and heteroaromatic nitro compounds. correlation with molecular orbital energies and hydrophobicity. *Journal of medicinal chemistry*, 34(2):786–797, 1991.

[9] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*, 2017.

[10] Alexandre Duval and Fragkiskos D Malliaros. Graphsvx: Shapley value explanations for graph neural networks. In *Machine Learning and Knowledge Discovery in Databases. Research Track: European Conference, ECML PKDD 2021, Bilbao, Spain, September 13–17, 2021, Proceedings, Part II 21*, pages 302–318. Springer, 2021.

[11] Ji Gao, Xiao Huang, and Jundong Li. Unsupervised graph alignment with wasserstein distance discriminator. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 426–435, 2021.

[12] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International Conference on Machine Learning*, pages 1263–1272. PMLR, 2017.

[13] Ehsan Hajiramezanali, Arman Hasanzadeh, Krishna Narayanan, Nick Duffield, Mingyuan Zhou, and Xiaoning Qian. Variational graph recurrent neural networks. *Advances in neural information processing systems*, 32, 2019.

[14] Ehsan Hajiramezanali, Arman Hasanzadeh, Nick Duffield, Krishna Narayanan, and Xiaoning Qian. Bayrel: Bayesian relational learning for multi-omics data integration. *Advances in Neural Information Processing Systems*, 33:19251–19263, 2020.

[15] Arman Hasanzadeh, Ehsan Hajiramezanali, Nick Duffield, and Xiaoning Qian. Morel: Multi-omics relational learning. In *International Conference on Learning Representations*.

[16] Arman Hasanzadeh, Ehsan Hajiramezanali, Krishna Narayanan, Nick Duffield, Mingyuan Zhou, and Xiaoning Qian. Semi-implicit graph variational auto-encoders. *Advances in neural information processing systems*, 32, 2019.

[17] Arman Hasanzadeh, Ehsan Hajiramezanali, Shahin Boluki, Mingyuan Zhou, Nick Duffield, Krishna Narayanan, and Xiaoning Qian. Bayesian graph neural networks with adaptive connection sampling. In *International conference on machine learning*, pages 4094–4104. PMLR, 2020.

[18] Pieter-Jan Kindermans, Sara Hooker, Julius Adebayo, Maximilian Alber, Kristof T Schütt, Sven Dähne, Dumitru Erhan, and Been Kim. The (un) reliability of saliency methods. *Explainable AI: Interpreting, explaining and visualizing deep learning*, pages 267–280, 2019.

[19] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017.

[20] Chris Lin, Hugh Chen, Chanwoo Kim, and Su-In Lee. Contrastive corpus attribution for explaining representations. *arXiv preprint arXiv:2210.00107*, 2022.

[21] Meng Liu, Youzhi Luo, Limei Wang, Yaochen Xie, Hao Yuan, Shurui Gui, Haiyang Yu, Zhao Xu, Jingtun Zhang, Yi Liu, Keqiang Yan, Haoran Liu, Cong Fu, Bora Oztekin, Xuan Zhang, and Shuiwang Ji. DIG: A turnkey library for diving into graph deep learning research. *arXiv preprint arXiv:2103.12608*, 2021.

[22] Dongsheng Luo, Wei Cheng, Dongkuan Xu, Wenchao Yu, Bo Zong, Haifeng Chen, and Xiang Zhang. Parameterized explainer for graph neural network, 2020. URL `https://arxiv.org/abs/2011.04573`.

[23] Youzhi Luo, Keqiang Yan, and Shuiwang Ji. Graphdf: A discrete flow model for molecular graph generation. In *International Conference on Machine Learning*, pages 7192–7203. PMLR, 2021.

[24] Dylan Slack, Anna Hilgard, Sameer Singh, and Himabindu Lakkaraju. Reliable post hoc explanations: Modeling uncertainty in explainability. *Advances in neural information processing systems*, 34:9391–9404, 2021.

[25] Suraj Srinivas, Sebastian Bordt, and Hima Lakkaraju. Which models have perceptually-aligned gradients? an explanation via off-manifold robustness. *arXiv preprint arXiv:2305.19101*, 2023.

[26] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International conference on machine learning*, pages 3319–3328. PMLR, 2017.

[27] Vayer Titouan, Nicolas Courty, Romain Tavenard, and Rémi Flamary. Optimal transport for structured data with application on graphs. In *International Conference on Machine Learning*, pages 6275–6284. PMLR, 2019.

[28] Talip Ucar and Ehsan Hajiramezanali. Parameter averaging for robust explainability. *arXiv preprint arXiv:2208.03249*, 2022.

[29] Saurabh Verma and Zhi-Li Zhang. Hunt for the unique, stable, sparse and fast feature learning on graphs. *Advances in Neural Information Processing Systems*, 30, 2017.

[30] Yaochen Xie, Sumeet Katariya, Xianfeng Tang, Edward Huang, Nikhil Rao, Karthik Subbian, and Shuiwang Ji. Task-agnostic graph explanations. In *Advances in Neural Information Processing Systems*, volume 35, pages 12027–12039. Curran Associates, Inc., 2022.

[31] Hongteng Xu, Dixin Luo, Hongyuan Zha, and Lawrence Carin Duke. Gromov-wasserstein learning for graph matching and node embedding. In *International conference on machine learning*, pages 6932–6941. PMLR, 2019.

[32] Zhitao Ying, Dylan Bourgeois, Jiaxuan You, Marinka Zitnik, and Jure Leskovec. Gnnexplainer: Generating explanations for graph neural networks. *Advances in neural information processing systems*, 32, 2019.

[33] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. Graph contrastive learning with augmentations. *Advances in Neural Information Processing Systems*, 33:5812–5823, 2020.

[34] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. Graph contrastive learning with augmentations. *Advances in neural information processing systems*, 33:5812–5823, 2020.

[35] Hao Yuan, Haiyang Yu, Shurui Gui, and Shuiwang Ji. Explainability in graph neural networks: A taxonomic survey. *IEEE transactions on pattern analysis and machine intelligence*, PP, 2020.

[36] Hao Yuan, Haiyang Yu, Jie Wang, Kang Li, and Shuiwang Ji. On explainability of graph neural networks via subgraph explorations. In *International Conference on Machine Learning*, pages 12241–12252. PMLR, 2021.

[37] Shichang Zhang, Yozen Liu, Neil Shah, and Yizhou Sun. Gstarx: Explaining graph neural networks with structure-aware cooperative games. In *Advances in Neural Information Processing Systems*, 2022.

A Related works

GNN explanation methods. These methods aim to produce an explanation for a GNN prediction on a given graph, usually as a subgraph induced by important nodes or edges [32, 35]. The majority of existing methods are prediction-based. This means that they operate at the level of a single graph/prediction pair, generating explanations to shed light on why the model made a specific prediction for that particular input graph [37]. These methods can be roughly divided into two categories: gradient- and perturbation-based approaches[35]. Methods in the former category use signal from gradients or output decomposition to infer salient node features [26]. Conversely, perturbation-based methods involve querying the model around the desired prediction to determine the relevance of input features in relation to the output [35, 30, 22]. These methods are based on different scoring functions to identify important nodes or edges. For example, the scoring function of GNNExplainer [32] is the mutual information between a masked graph and the prediction on the original graph, where soft masks on edges and node features are generated by direct parameter learning. PGExplainer [22] uses the same scoring function as GNNExplainer but generates a discrete mask on edges by training an edge mask predictor.

SubgraphX [36] uses the Shapley value as its scoring function on subgraphs selected by Monte Carlo Tree Search (MCTS), and GraphSVX [10] uses a least-square approximation to the Shapley value to score nodes and their features. While SubgraphX was shown to perform better than prior alternatives, the Shapley value they try to approximate is non-ideal as it is non-structure-aware. To address this issue, Zhang et al. [37] propose a graph structure-aware explanation (GStarX) to leverage the critical graph structure information to improve the GNN explanation. Specifically, GStarX defines a scoring function based on Hamiache and Navarro (HN) value that can utilize graph structures to attribute cooperation surplus between neighbor nodes, resembling message passing in GNNs, so that node importance scores reflect not only the node feature importance but also the node structural roles [37].

The (un)reliability of interpretability methods. The ability to offer explanations has emerged as a key focus in machine learning [6]. It serves not only to enhance our understanding of a model's underlying reasoning but also to adhere to regulatory obligations, facilitate control, and aid in model debugging [20, 25, 5]. Despite the numerous interpretability tools developed by machine learning researchers, they have faced significant criticism [9, 28, 28]. These criticisms often emphasize the need for caution when using explanations generated from these tools, highlighting concerns such as computational limitations or the reliance on qualitative user studies as evidence [18, 4]. Given the need for a quantitative method of comparison, several properties such as completeness, implementation invariance, and sensitivity have been articulated as desirable to ensure that saliency methods are reliable [18]. However, the reliability of explainability methods has not been well studied in the graph domain and requires GNN-specific considerations to handle the non-Euclidean nature of the data and their complex topological structures.