GNNX-BENCH: UNRAVELLING THE UTILITY OF PERTURBATION-BASED GNN EXPLAINERS THROUGH IN-DEPTH BENCHMARKING

Mert Kosan[1*†], **Samidha Verma**[2*], **Burouj Armgaan**[2], **Khushbu Pahwa**[3], **Ambuj Singh**[1]
Sourav Medya[4], **Sayan Ranu**[2]
University of California, Santa Barbara[1]
Indian Institute of Technology, Delhi[2]
Rice University[3]
University of Illinois, Chicago[4]
mertkosan@gmail.com, kp66@rice.edu, ambuj@cs.ucsb.edu, medya@uic.edu
{samidha.verma, burouj.armgaan, sayanranu}@cse.iitd.ac.in

ABSTRACT

Numerous explainability methods have been proposed to shed light on the inner workings of GNNs. Despite the inclusion of empirical evaluations in all the proposed algorithms, the interrogative aspects of these evaluations lack diversity. As a result, various facets of explainability pertaining to GNNs, such as a comparative analysis of counterfactual reasoners, their stability to variational factors such as different GNN architectures, noise, stochasticity in non-convex loss surfaces, feasibility amidst domain constraints, and so forth, have yet to be formally investigated. Motivated by this need, we present a benchmarking study on perturbation-based explainability methods for GNNs, aiming to systematically evaluate and compare a wide range of explainability techniques. Among the key findings of our study, we identify the Pareto-optimal methods that exhibit superior efficacy and stability in the presence of noise. Nonetheless, our study reveals that all algorithms are affected by stability issues when faced with noisy data. Furthermore, we have established that the current generation of counterfactual explainers often fails to provide feasible recourses due to violations of topological constraints encoded by domain-specific considerations. Overall, this benchmarking study empowers stakeholders in the field of GNNs with a comprehensive understanding of the state-of-the-art explainability methods, potential research problems for further enhancement, and the implications of their application in real-world scenarios.

1 INTRODUCTION AND RELATED WORK

GNNs have shown state-of-the-art performance in various domains including social networks Manchanda et al. (2020); Chakraborty et al. (2023), biological sciences Ying et al. (2021); Rampášek et al. (2022); Ranjan et al. (2022), modeling of physical systems Thangamuthu et al. (2022); Bhattoo et al. (2022; 2023); Bishnoi et al. (2023), event detection Cao et al. (2021); Kosan et al. (2021) and traffic modeling Gupta et al. (2023); Jain et al. (2021); Wu et al. (2017); Li et al. (2020). Unfortunately, like other deep-learning models, GNNs are black boxes due to lacking transparency and interpretability. This lack of interpretability is a significant barrier to their adoption in critical domains such as healthcare, finance, and law enforcement. In addition, the ability to explain predictions is critical towards understanding potential flaws in the model and generate insights for further refinement. To impart interpretability to GNNs, several algorithms to explain the inner workings of GNNs have been proposed. The diversified landscape of GNN explainability research is visualized in Fig. 1. We summarize each of the categories below:

- **Model-level:** Model-level or global explanations are concerned with the overall behavior of the model and search for patterns in the set of predictions made by the model. XGNN Yuan et al. (2020), GLG-Explainer Azzolin et al. (2023), Xuanyuan et al. Xuanyuan et al. (2023), GCFExplainer Huang et al. (2023).

[*] Both authors contributed equally to this research.
[†] Work done prior to joining Visa Inc.

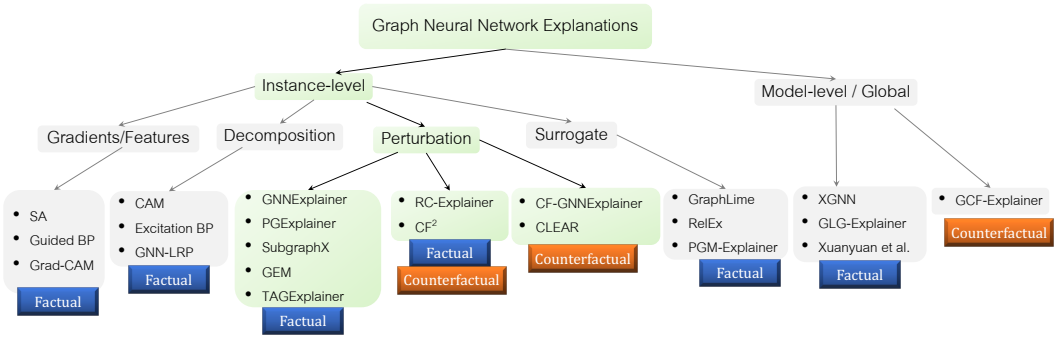

Figure 1: Structuring the space of the existing methods on GNN explainability.

- **Instance-level:** Instance-level or local explainers provide explanations for specific predictions made by a model. For instance, these explanations reason why a particular instance or input is classified or predicted in a certain way.
- **Gradient-based:** They follow the idea of the rate of change being represented by gradients. Additionally, the gradient of the prediction with respect to the input represents the prediction sensitivity to the input. This sensitivity gives the importance scores and helps in finding explanations. SA and Guided-BP Baldassarre & Azizpour (2019), Grad-CAM Pope et al. (2019).
- **Decomposition-based:** They consider the prediction of the model to be decomposed and distributed backward in a layer-by-layer fashion and the score of different parts of the input can be construed as its importance to the prediction. CAM and Excitation-BP Pope et al. (2019), GNN-LRP Schnake et al. (2021).
- **Perturbation-based:** They utilize input perturbations to identify important subgraphs serving as factual or counterfactual explanations. GNNExplainer Ying et al. (2019b), PGExplainer Luo et al. (2020), SubgraphX Yuan et al. (2021), GEM Lin et al. (2021a), TAGExplainer Xie et al. (2022), CF^2 Tan et al. (2022), RCExplainer Bajaj et al. (2021), CF-GNNexplainer Lucic et al. (2022), CLEAR Ma et al. (2022), Shan et al. (2021); Abrate & Bonchi (2021); Wellawatte et al. (2022)
- **Surrogate:** They use the generic intuition that in a smaller range of input values, the relationship between input and output can be approximated by interpretable functions. The methods fit a simple and interpretable surrogate model in the locality of the prediction. GraphLime Huang et al. (2022), Relex Zhang et al. (2021), PGM-Explainer Vu & Thai (2020).

The type of explanation offered represents a crucial component. Explanations can be broadly classified into two categories: *factual* reasoning and *counterfactual* reasoning.

- **Factual explanations** provide insights into the rationale behind a specific prediction by identifying the minimal subgraph that is sufficient to yield the same prediction as the entire input graph.
- **Counterfactual explanations** elucidate why a particular prediction was not made by presenting alternative scenarios that could have resulted in a different decision. In the context of graphs, this involves identifying the smallest perturbation to the input graph that alters the prediction of the GNN. Perturbations typically involve the removal of edges or modifications to node features.

1.1 CONTRIBUTIONS

In this benchmarking study, we systematically study perturbation-based factual and counterfactual explainers and identify their strengths and limitations in terms of their ability to provide accurate, meaningful, and actionable explanations for GNN predictions. The proposed study surfaces new insights that have not been studied in existing benchmarking literature Amara et al. (2022); Agarwal et al. (2023)(See. App. J for details). Overall, we make the following key contributions:

- **Comprehensive evaluation encompassing counterfactual explainers:** The benchmarking study encompasses seven factual explainers and four counterfactual explainers. The proposed work is the first benchmarking study on counterfactual explainers for GNNs.
- **Novel insights:** The findings of our benchmarking study unveil stability to noise and variational factors and generating feasible counterfactual recourses as two critical technical deficiencies that naturally lead us towards open research challenges.
- **Codebase:** As a by-product, a meticulously curated, publicly accessible code base is provided (https://github.com/Armagaan/gnn-x-bench/).

Table 1: Key highlights of the *perturbation-based* factual methods. The "NFE" column implies *Node Feature Explanation*. "GC" and "NC" indicate whether the dataset is used for graph classification and node classification respectively.

Method	Subgraph Extraction Strategy	Scoring function	Constraints	NFE	Task	Nature
GNNExplainer	Continuous relaxation	Mutual Information	Size	Yes	GC+NC	Transductive
PGExplainer	Parameterized edge selection	Mutual Information	Size, Connectivity	No	GC+NC	Inductive
TAGExplainer	Sampling	Mutual Information	Size, Entropy	No	GC+NC	Inductive
GEM	Granger Causality+Autoencoder	Causal Contribution	Size, Connectivity	No	GC+NC	Inductive
SubgraphX	Monte Carlo Tree Search	Shapley Value	Size, Connectivity	No	GC	Transductive

2 PRELIMINARIES AND BACKGROUND

We use the notation $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ to represent a graph, where $\mathcal{V}$ denotes the set of nodes and $\mathcal{E}$ denotes the set of edges. Each node $v_i \in \mathcal{V}$ is associated with a feature vector $x_i \in \mathbb{R}^d$. We assume there exists a GNN Φ that has been trained on $\mathcal{G}$ (or a set of graphs).

The literature on GNN explainability has primarily focused on *graph classification* and *node classification*, and hence the output space is assumed to be categorical. In graph classification, we are given a set of graphs as input, each associated with a class label. The task of the GNN Φ is to correctly predict this label. In the case of node classification, class labels are associated with each node and the predictions are performed on nodes. In a message passing GNN of ℓ layers, the embedding on a node is a function of its ℓ-hop neighborhood. We use the term *inference subgraph* to refer to this ℓ-hop neighborhood. Henceforth, we will assume that graph refers to the inference subgraph for node classification. Factual and counterfactual reasoning over GNNs are defined as follows.

Definition 1 (Perturbation-based Factual Reasoning) *Let $\mathcal{G}$ be the input graph and $\Phi(\mathcal{G})$ the prediction on $\mathcal{G}$. Our task is to identify the smallest subgraph $\mathcal{G}_S \subseteq \mathcal{G}$ such that $\Phi(\mathcal{G}) = \Phi(\mathcal{G}_S)$. Formally, the optimization problem is expressed as follows:*

$$\mathcal{G}_S = \arg \min_{\mathcal{G}' \subseteq \mathcal{G},\ \Phi(\mathcal{G}) = \Phi(\mathcal{G}')} ||\mathcal{A}(\mathcal{G}')|| \tag{1}$$

Here, $\mathcal{A}(\mathcal{G}_S)$ denotes the adjacency matrix of $\mathcal{G}_S$, and $||\mathcal{A}(\mathcal{G}_S)||$ is its L1 norm which is equivalent to the number of edges. Note that if the graph is undirected, the number of edges is half of the L1 norm. Nonetheless, the optimization problem remains the same.

While subgraph generally concerns only the topology of the graph, since graphs in our case may be annotated with features, some algorithms formulate the minimization problem in the joint space of topology and features. Specifically, in addition to identifying the smallest subgraph, we also want to minimize the number of features required to characterize the nodes in this subgraph.

Definition 2 (Counterfactual Reasoning) *Let $\mathcal{G}$ be the input graph and $\Phi(\mathcal{G})$ the prediction on $\mathcal{G}$. Our task is to introduce the minimal set of perturbations to form a new graph $\mathcal{G}^*$ such that $\Phi(\mathcal{G}) \neq \Phi(\mathcal{G}^*)$. Mathematically, this entails to solving the following optimization problem.*

$$\mathcal{G}^* = \arg \min_{\mathcal{G}' \in \mathbb{G},\ \Phi(\mathcal{G}) \neq \Phi(\mathcal{G}')} dist(\mathcal{G}, \mathcal{G}') \tag{2}$$

where $dist(\mathcal{G}, \mathcal{G}')$ quantifies the distance between graphs $\mathcal{G}$ and $\mathcal{G}'$ and $\mathbb{G}$ is the set of all graphs one may construct by perturbing $\mathcal{G}$. Typically, distance is measured as the number of edge perturbations while keeping the node set fixed. In case of multi-class classification, if one wishes to switch to a target class label(s), then the optimization objective is modified as $\mathcal{G}^* = \arg\min_{\mathcal{G}' \in \mathbb{G},\ \Phi(\mathcal{G}') = \mathbb{C}}\ dist(\mathcal{G}, \mathcal{G}')$, where $\mathbb{C}$ is the set of desired class labels.

2.1 REVIEW OF PERTURBATION-BASED GNN REASONING

Factual (Yuan et al. (2022); Kakkad et al. (2023)): The perturbation schema for factual reasoning usually consists of two crucial components: the subgraph extraction module and the scoring function module. Given an input graph $\mathcal{G}$, the subgraph extraction module extracts a subgraph $\mathcal{G}_s$; and the scoring function module evaluates the model predictions $\Phi(\mathcal{G}_s)$ for the subgraphs, comparing them with the actual predictions $\Phi(\mathcal{G})$. For instance, while GNNExplainer Ying et al. (2019a) identifies an explanation in the form of a subgraph that have the maximum influence on the prediction, PG-Explainer Luo et al. (2020) assumes the graph to be a random Gilbert graph. Unlike the existing explainers, TAGExplainer Xie et al. (2022) takes a two-step approach where the first step has an

Table 2: Key highlights of the counterfactuals methods."GC" and "NC" indicate whether the dataset is used for graph classification and node classification respectively.

Method	Explanation Type	Task	Target/Method	Nature
RCExplainer Bajaj et al. (2021)	Instance level	GC+NC	Neural Network	Inductive
CF^2 Tan et al. (2022)	Instance level	GC+NC	Original graph	Transductive
CF-GNNExplainer Lucic et al. (2022)	Instance level	NC	Inference subgraph	Transductive
CLEAR Ma et al. (2022)	Instance level	GC+NC	Variational Autoencoder	Inductive

embedding explainer trained using a self-supervised training framework without any information of the downstream task. On the other hand, GEM Lin et al. (2021a) uses Granger causality and an autoencoder for the subgraph extraction strategy where as SubgraphX Yuan et al. (2021) employes a monte carlo tree search. The scoring function module uses mutual information for GNNExplainer, PGExplainer, and TAGExplainer. This module is different for GEM and SubgraphX, and uses casual contribution and Shapley value respectively. Table 1 summarizes the key highlights.

Counterfactual (Yuan et al. (2022)): The four major counterfactual methods are CF-GNNExplainer Lucic et al. (2022), CF^2 Tan et al. (2022), RCExplainer Bajaj et al. (2021), and CLEAR Ma et al. (2022). They are instance-level explainers and apply to both graph and node classification tasks except for CF-GNNExplainer which is only applied to node classification. In terms of method, CF-GNNExplainer aims to perturb the computational graph by using a binary mask matrix. The corresponding loss function quantifies the accuracy of the produced counterfactual and captures the distance (or similarity) between the counterfactual graph and the original graph, whereas, CF^2 Tan et al. (2022) extends this method by including a contrastive loss that jointly optimizes the quality of both the factual and the counterfactual explanation. Both of the above methods are transductive. As an inductive method, RCExplainer Bajaj et al. (2021), aims to identify a resilient subset of edges to remove such that it alters the prediction of the remaining graph while CLEAR Ma et al. (2022) generates counterfactual graphs by using a graph variational autoencoder. Table 2 summarizes the key highlights.

3 Benchmarking Framework

In this section, we outline the investigations we aim to conduct and the rationale behind them. The mathematical formulation of the various metrics are summarized in Table 3.

Comparative Analysis: We evaluate algorithms for both factual and counterfactual reasoning and identify the pareto-optimal methods. The performance is quantified using *explanation size* and *sufficiency* Tan et al. (2022). Sufficiency encodes the ratio of graphs for which the prediction derived from the explanation matches the prediction obtained from the complete graph Tan et al. (2022). Its value spans between 0 and 1. For factual explanations, higher values indicate superior performance, while in counterfactual lower is better since the objective is to flip the class label.

Stability: Stability of explanations, when faced with minor variations in the evaluation framework, is a crucial aspect that ensures their reliability and trustworthiness. Stability is quantified by taking the *Jaccard similarity* between the set of edges in the original explanation vs. those obtained after introducing the variation (details in § 4). In order to evaluate this aspect, we consider the following perspectives:

- **Perturbations in topological space:** If we inject minor perturbations to the topology through a small number of edge deletions or additions, then that should not affect the explanations.
- **Model parameters:** The explainers are deep-learning models themselves and optimize a non-convex loss function. As a consequence of non-convexity, when two separate instances of the explainer starting from different seeds are applied to the same GNN model, they generate dissimilar

Table 3: The various metrics used to benchmark the performance of GNN explainers.

Sufficiency$(\mathcal{S}) = \frac{\sum_{i=1}^{\lvert\mathbb{G}\rvert} \mathbb{1}(\Phi(\mathcal{G}_S^i)=\Phi(\mathcal{G}^i))}{\lvert\mathbb{G}\rvert}$	• $\mathbb{G} = \{\mathcal{G}^1, \mathcal{G}^2, \ldots, \mathcal{G}^n\}$: graph set.
Necessity$(\mathcal{N}) = \frac{\sum_{i=1}^{\lvert\mathbb{G}\rvert} \mathbb{1}(\Phi(\mathcal{R}^i)\neq\Phi(\mathcal{G}^i))}{\lvert\mathbb{G}\rvert}$	• $\mathcal{G}_S^i$: explanation subgraph of $\mathcal{G}^i$ • $\mathbb{G}_S = \{\mathcal{G}_S^1, \mathcal{G}_S^2, \ldots, \mathcal{G}_S^n\}$: explanation set. • $\mathcal{R}^i = \mathcal{G} - \mathcal{G}_S^i$
Stability$(\mathcal{E}_X, \mathcal{E}'_X) = \frac{\lvert\mathcal{E}_X \cap \mathcal{E}'_X\rvert}{\lvert\mathcal{E}_X \cup \mathcal{E}'_X\rvert}$	• $\mathbb{R} = \{\mathcal{R}^1, \mathcal{R}^2, \ldots, \mathcal{R}^n\}$: residual graph set. • Φ, Φ_S, Φ_R: the models trained on $\mathbb{G}$, $\mathbb{G}_S$, $\mathbb{R}$. All models are trained on the same labels. • $\Phi(\mathbb{G}^i)$: the prediction of the model on $\mathbb{G}^i$. • $ACC(\Phi)$: the test accuracy of Φ.
Reproducibility$^+(\mathcal{R}^+) = \frac{ACC(\Phi_S)}{ACC(\Phi)}$	
Reproducibility$^-(\mathcal{R}^-) = \frac{ACC(\Phi_R)}{ACC(\Phi)}$	

explanations. Our benchmarking study investigates the impact of this stochasticity on the quality and consistency of the explanations produced.
- **Model architectures:** Message-passing GNNs follow a similar computation framework, differing mainly in their message aggregation functions. We explore the stability of explanations under variations in the model architecture.

Necessity: Factual explanations are *necessary* if the removal of the explanation subgraph from the graph results in counterfactual graph (i.e., flipping the label).
Reproducibility: We measure two different aspects related to how central the explanation is towards retaining the prediction outcomes. Specifically, Reproducibility^{+} measures if the GNN is retrained on the explanation graphs alone, can it still obtain the original predictions? On the other hand, Reproducibility^{-} measures if the GNN is retrained on the *residual* graph constructed by removing the explanation from the original graph, can it still predict the class label? The mathematical quantification of these metrics is presented in Fig. 3.
Feasibility: One notable characteristic of counterfactual reasoning is its ability to offer recourse options. Nonetheless, in order for these recourses to be effective, they must adhere to the specific domain constraints. For instance, in the context of molecular datasets, the explanation provided must correspond to a valid molecule. Likewise, if the domain involves consistently connected graphs, the recourse must maintain this property. The existing body of literature on counterfactual reasoning with GNNs has not adequately addressed this aspect, a gap we address in our benchmarking study.

Table 4: The statistics of the datasets. Here, "F" and "CF" in the column "'X-type" indicates whether the dataset is used for Factual or Counterfactual reasoning. "GC" and "NC" in the *Task* column indicates whether the dataset is used for graph classification and node classification respectively.

	#Graphs	#Nodes	#Edges	#Features	#Classes	Task	F/CF
MUTAGENICITY Riesen & Bunke (2008); Kazius et al. (2005)	4337	131488	133447	14	2	GC	F+CF
PROTEINS Borgwardt et al. (2005); Dobson & Doig (2003)	1113	43471	81044	32	2	GC	F+CF
IMDB-B Yanardag & Vishwanathan (2015)	1000	19773	96531	136	2	GC	F+CF
AIDS Ivanov et al. (2019)	2000	31385	32390	42	2	GC	F+CF
MUTAG Ivanov et al. (2019)	188	3371	3721	7	2	GC	F+CF
NCI1 Wale et al. (2008)	4110	122747	132753	37	2	GC	F
GRAPH-SST2 Yuan et al. (2022)	70042	714325	644283	768	2	GC	F
DD Dobson & Doig (2003)	1178	334925	843046	89	2	GC	F
REDDIT-B Yanardag & Vishwanathan (2015)	2000	859254	995508	3063	2	GC	F
OGBG-MOLHIV Allamanis et al. (2018)	41127	1049163	2259376	9	2	GC	CF
TREE-CYCLES Ying et al. (2019a)	1	871	1950	10	2	NC	CF
TREE-GRID Ying et al. (2019a)	1	1231	3410	10	2	NC	CF
BA-SHAPES Ying et al. (2019a)	1	700	4100	10	4	NC	CF

4 EMPIRICAL EVALUATION

In this section, we execute the investigation plan outlined in § 3. Unless mentioned specifically, the base black-box GNN is a GCN. Details of the set up (e.g., hardware) are provided in App. A.
Datasets: Table 4 showcases the principal statistical characteristics of each dataset employed in our experiments, along with the corresponding tasks evaluated on them. The TREE-CYCLES, TREE-GRID, and BA-SHAPES datasets serve as benchmark graph datasets for counterfactual analysis. These datasets incorporate ground-truth explanations Tan et al. (2022); Lin et al. (2021a); Lucic et al. (2022). Each dataset contains an undirected base graph to which predefined motifs are attached to random nodes, and additional edges are randomly added to the overall graph. The class label assigned to a node determines its membership in a motif.

4.1 COMPARATIVE ANALYSIS

Factual Explainers: Fig. 2 illustrates the sufficiency analysis of various factual reasoners in relation to size. Each algorithm assigns a score to edges, indicating their likelihood of being included in the factual explanation. To control the size, we adopt a greedy approach by selecting the highest-scoring edges. Both CF^2 and RCEXPLAINER necessitate a parameter to balance factual and counterfactual explanations. We set this parameter to 1, corresponding to solely factual explanations.

Insights: No single technique dominates across all datasets. For instance, while RCEXPLAINER performs exceptionally well in the MUTAG dataset, it exhibits subpar performance in IMDB-B and GRAPH-SST2. Similar observations are also made for GNNEXPLAINER in REDDIT-B vs. MUTAG and NCI1. Overall, we recommend using either RCEXPLAINER or GNNEXPLAINER as the preferred choices. The spider plot in Fig. Q more prominently substantiates this suggestion. GNNEXPLAINER is transductive, wherein it trains the parameters on the input graph itself. In contrast, inductive

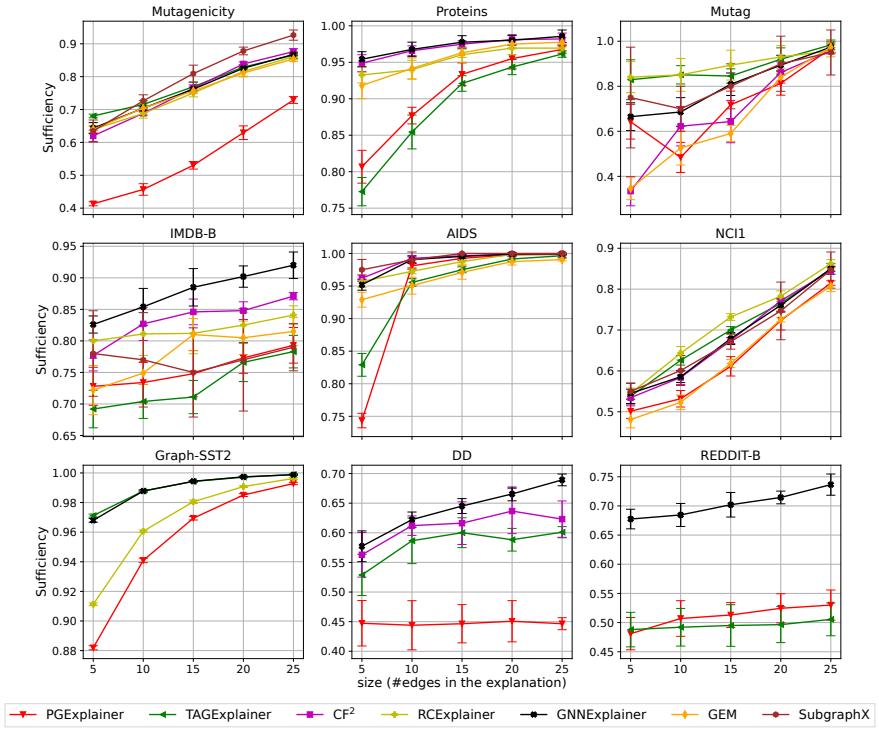

Figure 2: Sufficiency of the factual explainers against the explanation size. For factual explanations, higher is better. We omit those methods for a dataset that threw an out-of-memory (OOM) error.

methods use pre-trained weights to explain the input. Consequently, transductive methods, such as GNNEXPLAINER, at the expense of higher computation cost, has an inherent advantage in terms of optimizing sufficiency. Compared to other transductive methods, GNNEXPLAINER utilizes a loss function that aims to increase sufficiency directly. This makes the method a better candidate for sufficiency compared to other inductive and transductive explainers. On the other hand, for RCEXPLAINER, we believe that calculation of decision regions for classes helps to increase its generalizability as well as robustness.

In Fig. 2, the sufficiency does not always increase monotonically with explanation size (such as PGEXPLAINER in Mutag). This behavior arises due to the combinatorial nature of the problem. Specifically, the impact of adding an edge to an existing explanation on the GNN prediction is a function of both the edge being added and the edges already included in the explanation. An explainer seeks to learn a proxy function that mimics the true combinatorial output of a set of edges. When this proxy function fails to predict the marginal impact of adding an edge, it could potentially select an edge that exerts a detrimental influence on the explanation's quality.

Table 5: Sufficiency and size of counterfactual explainers on graph classification. Lower values are better for both metrics. OOM indicates that the technique ran out of memory.

	Mutag		**Mutagenicity**		**AIDS**		**Proteins**		**IMDB-B**		**ogbg-molhiv**	
Method / Metric	**Suffic.↓**	**Size↓**	**Suffic.↓**	**Size↓**	**Suffic.↓**	**Size↓**	**Suffic.↓**	**Size↓**	**Suffic.↓**	**Size↓**	**Suffic.↓**	**Size↓**
RCExplainer	0.4 ± 0.12	1.1 ± 0.22	0.4 ± 0.06	1.01 ± 0.19	0.91 ± 0.04	1.0 ± 0.0	0.96 ± 0.02	1.0 ± 0.0	0.72 ± 0.11	1.0 ± 0.0	0.90 ± 0.02	1 ± 0.0
CF²($\alpha = 0$)	0.90 ± 0.12	1.0 ± 0.0	0.50 ± 0.05	2.78 ± 0.98	0.98 ± 0.02	5.25 ± 0.35	1.0 ± 0.0	NA	0.81 ± 0.07	8.57 ± 4.99	0.96 ± 0.00	10.45 ± 4.43
CLEAR	0.55 ± 0.1	17.15 ± 1.62	OOM	OOM	0.84 ± 0.03	164.9 ± 47.9	OOM	OOM	0.96 ± 0.02	218.62 ± 0	OOM	OOM

Counterfactual Explainers: Tables 5 and 6 present the results on graph and node classification.

Insights on graph classification (Table 5): RCEXPLAINER is the best-performing explainer across the majority of the datasets and metrics. However, it is important to acknowledge that RCEXPLAINER's sufficiency, when objectively evaluated, consistently remains high, which is undesired. For instance, in the case of AIDS, the sufficiency of RCEXPLAINER reaches a value of 0.9, signifying its inability to generate counterfactual explanations for 90% of the graphs. This

Table 6: Performance of counterfactual explainers on node classification. Shaded cells indicate the best result in a column. Note that only CF-GNNEXPLAINER and CF^2 can explain node classification. In these datasets, ground truth explanations are provided. Hence, accuracy (Acc) represents the percentage of edges within the counterfactual that belong to the ground truth explanation.

	Tree-Cycles			Tree-Grid			BA-Shapes		
Method / Metric	**Suffic.** ↓	**Size** ↓	**Acc.(%)** ↑	**Suffic.** ↓	**Size** ↓	**Acc.(%)** ↑	**Suffic.** ↓	**Size** ↓	**Acc.(%)** ↑
CF-GNNEX	0.5 ± 0.08	1.03 ± 0.16	100.0 ± 0.0	0.09 ± 0.06	1.42 ± 0.55	92.70 ± 4.99	0.37 ± 0.05	1.37 ± 0.59	91.5 ± 4.36
CF^2 ($\alpha = 0$)	0.76 ± 0.06	4.55 ± 1.48	74.71 ± 18.70	0.99 ± 0.02	7.0 ± 0.0	14.29 ± 0.0	0.25 ± 0.88	4.24 ± 1.70	68.89 ± 12.28

observation suggests that there exists considerable potential for further enhancement. We also note that while CLEAR achieves the best (lowest) sufficiency in AIDS, the number of perturbations it requires (size) is exorbitantly high to be useful in practical use-cases.

Insights on node classification (Table 6): We observe that CF-GNNEXPLAINER consistently outperforms CF^2 ($\alpha = 0$ indicates the method to be entirely counterfactual). We note that our result contrasts with the reported results in CF^2 Tan et al. (2022), where CF^2 was shown to outperform CF-GNNEXPLAINER. A closer examination reveals that in Tan et al. (2022), the value of α was set to 0.6, placing a higher emphasis on factual reasoning. It was expected that with $\alpha = 0$, counterfactual reasoning would be enhanced. However, the results do not align with this hypothesis. We note that in CF^2, the optimization function is a combination of explanation complexity and explanation strength. The contribution of α is solely in the explanation strength component, based on its alignment with factual and counterfactual reasoning. The counterintuitive behavior observed with α is attributed to the domination of explanation complexity in the objective function, thereby diminishing the intended impact of α. Finally, when compared to graph classification, the sufficiency produced by the best methods in the node classification task is significantly lower indicating that it is an easier task. One possible reason might be the space of counterfactuals is smaller in node classification.

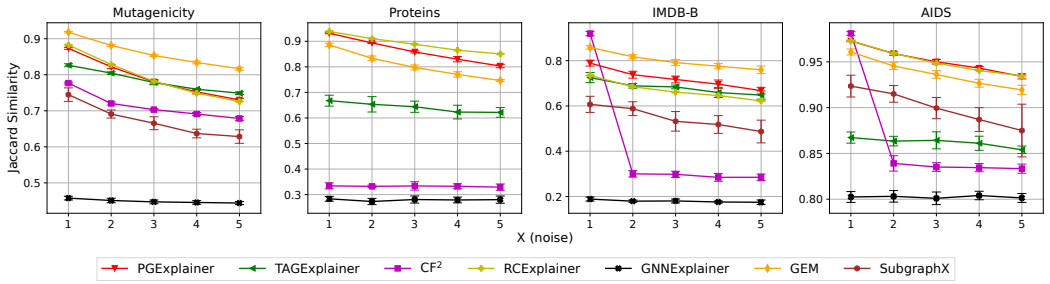

Figure 3: Stability of factual explainers in Jaccard similarity of explanations under topological noise. Here, the x-ticks (Noise) denote the number of perturbations made to the edge set of the original graph. Here, perturbations include randomly sampling x(denoted on x axis) negative edges and adding them to the original edge set (i.e., connect a pair of nodes that were previously unconnected).

4.2 STABILITY

We next examine the stability of the explanations against topological noise, model parameters, and the choice of GNN architecture. In App. C, we present the impact of the above mentioned factors on other metrics of interest such as sufficiency and explanation size. In addition, we also present impact of feature perturbation and topological adversarial attack in App. C.

Insights on factual-stability against topological noise: Fig. 3 illustrates the Jaccard coefficient as a function of the noise volume. Similar to Fig.2, edge selection for the explanation involves a greedy approach that prioritizes the highest score edges. A clear trend that emerges is that inductive methods consistently outperform the transductive methods (such as CF^2 and GNNEXPLAINER). This is expected since transductive methods lack generalizable capability to unseen data. Furthermore, the stability is worse on denser datasets of IMDB-B since due to the presence of more edges, the search space of explanation is larger. RCEXPLAINER (executed at $\alpha = 1$) and PGEXPLAINER consistently exhibit higher stability. This consistent performance reinforces the claim that RCEXPLAINER is the preferred factual explainer. The stability of RCEXPLAINER can be attributed to its strategy of selecting a subset of edges that is resistant to changes, such that the removal of these edges

Table 7: Stability in explanations provided by factual explainers across runs. We fix the size to 10 for all explainers. The most stable explainer for each dataset (row) corresponding to the three categories of $1vs2$, $1vs3$ and $2vs3$ are highlighted through gray, yellow and cyan shading respectively.

	PGExplainer			TAGExplainer			CF^2			RCExplainer			GNNExplainer		
Dataset / Seeds	$1vs2$	$1vs3$	$2vs3$	$1vs2$	$1vs3$	$2vs3$	$1vs2$	$1vs3$	$2vs3$	$1vs2$	$1vs3$	$2vs3$	$1vs2$	$1vs3$	$2vs3$
Mutagenicity	0.69	0.75	0.62	0.76	0.78	0.74	0.77	0.77	0.77	0.75	0.71	0.71	0.46	0.47	0.47
Proteins	0.38	0.51	0.38	0.55	0.48	0.46	0.34	0.34	0.35	0.88	0.85	0.91	0.28	0.28	0.28
Mutag	0.5	0.54	0.51	0.36	0.43	0.72	0.78	0.79	0.79	0.86	0.92	0.87	0.57	0.57	0.58
IMDB-B	0.67	0.76	0.67	0.67	0.60	0.56	0.32	0.32	0.32	0.75	0.73	0.70	0.18	0.19	0.18
AIDS	0.88	0.87	0.82	0.81	0.83	0.87	0.85	0.85	0.85	0.95	0.96	0.97	0.80	0.80	0.80
NCI1	0.58	0.55	0.64	0.69	0.81	0.65	0.60	0.60	0.60	0.71	0.71	0.94	0.44	0.44	0.44

Table 8: Stability of factual explainers against the GNN architecture. We fix the size to 10. We report the Jaccard coefficient of explanations obtained for each architecture against the explanation provided over GCN. The best explainers for each dataset (row) are highlighted in gray, yellow and cyan shading for GAT, GIN, and GRAPHSAGE, respectively. GRAPHSAGE is denoted by SAGE.

	PGExplainer			TAGExplainer			CF^2			RCExplainer			GNNExplainer		
Dataset / Architecture	GAT	GIN	SAGE	GAT	GIN	SAGE	GAT	GIN	SAGE	GAT	GIN	SAGE	GAT	GIN	SAGE
Mutagenicity	0.63	0.65	0.60	0.24	0.25	0.32	0.52	0.47	0.54	0.56	0.52	0.46	0.43	0.42	0.43
Proteins	0.22	0.47	0.38	0.45	0.41	0.18	0.28	0.28	0.28	0.37	0.41	0.42	0.28	0.28	0.28
Mutag	0.57	0.58	0.69	0.60	0.65	0.64	0.58	0.56	0.62	0.47	0.76	0.54	0.55	0.57	0.55
IMDB-B	0.48	0.45	0.56	0.44	0.35	0.47	0.17	0.23	0.17	0.30	0.33	0.26	0.17	0.17	0.17
AIDS	0.81	0.85	0.87	0.83	0.83	0.84	0.80	0.80	0.80	0.81	0.85	0.81	0.8	0.8	0.8
NCI1	0.39	0.41	0.37	0.45	0.17	0.58	0.37	0.38	0.38	0.49	0.53	0.52	0.37	0.38	0.39

significantly impacts the prediction made by the remaining graph Bajaj et al. (2021). PGEXPLAINER also incorporates a form of inherent stability within its framework. It builds upon the concept introduced in GNNEXPLAINER through the assumption that the explanatory graph can be modeled as a random Gilbert graph, where the probability distribution of edges is conditionally independent and can be parameterized. This generic assumption holds the potential to enhance the stability of the method. Conversely, TAGEXPLAINER exhibits the lower stability than RCEXPLAINER and PGEXPLAINER, likely due to its reliance solely on gradients in a task-agnostic manner Xie et al. (2022). The exclusive reliance on gradients makes it more susceptible to overfitting, resulting in reduced stability.

Insights on factual-stability against explainer instances: Table 7 presents the stability of explanations provided across three different explainer instances on the same black-box GNN. A similar trend is observed, with RCEXPLAINER remaining the most robust method, while GNNEXPLAINER exhibits the least stability. For GNNEXPLAINER, the Jaccard coefficient hovers around 0.5, indicating significant variance in explaining the same GNN. Although the explanations change, their quality remains stable (as evident from small standard deviation in Fig. 2). This result indicates that multiple explanations of similar quality exist and hence a single explanation fails to complete explain the data signals. This component is further emphasized when we delve into reproducibility (§ 4.3).

Insights on factual-stability against GNN architectures: Finally, we explore the stability of explainers across different GNN architectures in Table 8, which has not yet been investigated in the existing literature. For each combination of architectures, we assess the stability by computing the Jaccard coefficient between the explained predictions of the indicated GNN architecture and the default GCN model. One notable finding is that the stability of explainers exhibits a strong correlation with the dataset used. Specifically, in five out of six datasets, the best performing explainer across all architectures is unique. However, it is important to highlight that the Jaccard coefficients across architectures consistently remain low indicating stability against different architectures is the hardest objective due to the variations in their message aggregating schemes.

4.3 NECESSITY AND REPRODUCIBILITY

We aim to understand the quality of explanations in terms of necessity and reproducibility. The results are presented in App. D and E. Our findings suggest that necessity is low but increases with the removal of more explanations, while reproducibility experiments reveal that explanations do not provide a comprehensive explanation of the underlying data, and even removing them and retraining the model can produce a similar performance to the original GNN model.

4.4 FEASIBILITY

Counterfactual explanations serve as recourses and are expected to generate graphs that adhere to the feasibility constraints of the pertinent domain. We conduct the analysis of feasibility on molecular graphs. It is rare for molecules to be constituted of multiple connected components Vismara & Laurenço (2000). Hence, we study the distribution of molecules that are connected in the original dataset and its comparison to the distribution in counterfactual recourses. We measure the p-value of this deviation. App. A.7 presents the results.

4.5 VISUALIZATION-BASED ANALYSIS

We include visualizations of the explanations in App. F. Our analysis reveals that a statistically good performance does not always align with human judgment indicating an urgent need for datasets annotated with ground truth explanations. Furthermore, the visualization analysis reinforces the need to incorporate feasibility as a desirable component in counterfactual reasoning.

5 CONCLUDING INSIGHTS AND POTENTIAL SOLUTIONS

Our benchmarking study has yielded several insights that can streamline the development of explanation algorithms. We summarize the key findings below (please also see the App. K for our recommendations of explainer for various scenarios).

- **Performance and Stability:** Among the explainers evaluated, RCEXPLAINER consistently outperformed others in terms of efficacy and stability to noise and variational factors (§ 4.1 and § 4.2).
- **Stability Concerns:** Most factual explainers demonstrated significant deviations across explainer instances, vulnerability to topological perturbations and produced significantly different set of explanations across different GNN architectures. These stability notions should therefore be embraced as desirable factors along with other performance metrics.
- **Model Explanation vs. Data Explanation:** Reproducibility experiments (§ 4.3) revealed that retraining with only factual explanations cannot reproduce the predictions fully. Furthermore, even without the factual explanation, the GNN model predicted accurately on the residual graph. This suggests that explainers only capture specific signals learned by the GNN and do not encompass all underlying data signals.
- **Feasibility Issues:** Counterfactual explanations showed deviations in topological distribution from the original graphs, raising feasibility concerns (§ 4.4).

Potential Solutions: The aforementioned insights raise important shortcomings that require further investigation. Below, we explore potential avenues of research that could address these limitations.

- **Feasible recourses through counterfactual reasoning:** Current counterfactual explainers predominantly concentrate on identifying the shortest edit path that nudges the graph toward the decision boundary. This design inherently neglects the feasibility of the proposed edits. Therefore, it is imperative to explicitly address feasibility as an objective in the optimization function. One potential solution lies in the vibrant research field of generative modeling for graphs, which has yielded impressive results Goyal et al. (2020); You et al. (2018); Vignac et al. (2023). Generative models, when presented with an input graph, can predict its likelihood of occurrence within a domain defined by a set of training graphs. Integrating generative modeling into counterfactual reasoning by incorporating likelihood of occurrence as an additional objective in the loss function presents a potential remedy.
- **Ante-hoc explanations for stability and reproducibility:** We have observed that if the explanations are removed and the GNN is retrained on the residual graphs, the GNN is often able to recover the correct predictions from our reproducibilty experiments. Furthermore, the explanation exhibit significant instability in the face of minor noise injection. This incompleteness of explainers and instability is likely a manifestation of their *post-hoc* learning framework, wherein the explanations are generated post the completion of GNN training. In this pipeline, the explainers have no visibility to how the GNN would behave to perturbations on the input data, initialization seeds, etc. Potential solutions may lie on moving to an *ante-hoc* paradigm where the GNN and the explainer are jointly trained Kosan et al. (2023); Miao et al. (2022); Fang et al. (2023).

These insights, we believe, open new avenues for advancing GNN explainers, empowering researchers to overcome limitations and elevate the overall quality and interpretability of GNNs.

6 ACKNOWLEDGEMENTS

Samidha Verma acknowledges the generous grant received from Microsoft Research India to sponsor her travel to ICLR 2024. Additionally, this project was partially supported by funding from the National Science Foundation under grant #IIS-2229876 and the CSE Research Acceleration Fund of IIT Delhi.

REFERENCES

Carlo Abrate and Francesco Bonchi. Counterfactual graphs for explainable classification of brain networks. In *KDD*, pp. 2495–2504, 2021.

Chirag Agarwal, Owen Queen, Himabindu Lakkaraju, and Marinka Zitnik. Evaluating explainability for graph neural networks. 2023.

Miltiadis Allamanis, Earl T Barr, Premkumar Devanbu, and Charles Sutton. A survey of machine learning for big code and naturalness. *ACM Computing Surveys (CSUR)*, 51(4):1–37, 2018.

Kenza Amara, Zhitao Ying, Zitao Zhang, Zhichao Han, Yang Zhao, Yinan Shan, Ulrik Brandes, Sebastian Schemm, and Ce Zhang. Graphframex: Towards systematic evaluation of explainability methods for graph neural networks. In *The First Learning on Graphs Conference*, 2022. URL https://openreview.net/forum?id=rGVGf1T-dK.

Steve Azzolin, Antonio Longa, Pietro Barbiero, Pietro Lio, and Andrea Passerini. Global explainability of GNNs via logic combination of learned concepts. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=OTbRTIY4YS.

Mohit Bajaj, Lingyang Chu, Zi Yu Xue, Jian Pei, Lanjun Wang, Peter Cho-Ho Lam, and Yong Zhang. Robust counterfactual explanations on graph neural networks. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=Uq_tGs7N54M.

Federico Baldassarre and Hossein Azizpour. Explainability techniques for graph convolutional networks. *arXiv preprint arXiv:1905.13686*, 2019.

Ravinder Bhattoo, Sayan Ranu, and NM Krishnan. Learning articulated rigid body dynamics with lagrangian graph neural network. *Advances in Neural Information Processing Systems*, 35: 29789–29800, 2022.

Ravinder Bhattoo, Sayan Ranu, and NM Anoop Krishnan. Learning the dynamics of particle-based systems with lagrangian graph neural networks. *Machine Learning: Science and Technology*, 2023.

Suresh Bishnoi, Ravinder Bhattoo, Sayan Ranu, and NM Krishnan. Enhancing the inductive biases of graph neural ode for modeling dynamical systems. *ICLR*, 2023.

Karsten M Borgwardt, Cheng Soon Ong, Stefan Schönauer, SVN Vishwanathan, Alex J Smola, and Hans-Peter Kriegel. Protein function prediction via graph kernels. *Bioinformatics*, 21(suppl_1): i47–i56, 2005.

Yuwei Cao, Hao Peng, Jia Wu, Yingtong Dou, Jianxin Li, and Philip S Yu. Knowledge-preserving incremental social event detection via heterogeneous gnns. In *Proceedings of the Web Conference 2021*, pp. 3383–3395, 2021.

Pritish Chakraborty, Sayan Ranu, Krishna Sri Ipsit Mantri, and Abir De. Learning and maximizing influence in social networks under capacity constraints. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, pp. 733–741, 2023.

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.

Paul D Dobson and Andrew J Doig. Distinguishing enzyme structures from non-enzymes without alignments. *Journal of molecular biology*, 330(4):771–783, 2003.

Junfeng Fang, Xiang Wang, An Zhang, Zemin Liu, Xiangnan He, and Tat-Seng Chua. Cooperative explanations of graph neural networks. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, pp. 616–624, 2023.

Nikhil Goyal, Harsh Vardhan Jain, and Sayan Ranu. Graphgen: a scalable approach to domain-agnostic labeled graph generation. In *Proceedings of The Web Conference 2020*, pp. 1253–1263, 2020.

Mridul Gupta, Hariprasad Kodamana, and Sayan Ranu. FRIGATE: Frugal spatio-temporal forecasting on road networks. In *29th SIGKDD Conference on Knowledge Discovery and Data Mining*, 2023. URL https://openreview.net/forum?id=2cTw2M47L1.

William L. Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 1025–1035, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Qiang Huang, Makoto Yamada, Yuan Tian, Dinesh Singh, and Yi Chang. Graphlime: Local interpretable model explanations for graph neural networks. *IEEE Transactions on Knowledge and Data Engineering*, 2022.

Zexi Huang, Mert Kosan, Sourav Medya, Sayan Ranu, and Ambuj Singh. Global counterfactual explainer for graph neural networks. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, pp. 141–149, 2023.

Sergei Ivanov, Sergei Sviridov, and Evgeny Burnaev. Understanding isomorphism bias in graph data sets. *Geometric Learning and Graph Representations ICLR Workshop*, 2019.

Jayant Jain, Vrittika Bagadia, Sahil Manchanda, and Sayan Ranu. Neuromlr: Robust & reliable route recommendation on road networks. *Advances in Neural Information Processing Systems*, 34: 22070–22082, 2021.

Jaykumar Kakkad, Jaspal Jannu, Kartik Sharma, Charu Aggarwal, and Sourav Medya. A survey on explainability of graph neural networks. *arXiv preprint arXiv:2306.01958*, 2023.

Jeroen Kazius, Ross McGuire, and Roberta Bursi. Derivation and validation of toxicophores for mutagenicity prediction. *Journal of medicinal chemistry*, 48(1):312–320, 2005.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Mert Kosan, Arlei Silva, Sourav Medya, Brian Uzzi, and Ambuj Singh. Event detection on dynamic graphs. *arXiv preprint arXiv:2110.12148*, 2021.

Mert Kosan, Arlei Silva, and Ambuj Singh. Robust ante-hoc graph explainer using bilevel optimization. *arXiv preprint arXiv:2305.15745*, 2023.

Xiucheng Li, G. Cong, and Yun Cheng. Spatial transition learning on road networks with deep probabilistic models. *2020 IEEE 36th International Conference on Data Engineering (ICDE)*, pp. 349–360, 2020. URL https://api.semanticscholar.org/CorpusID:218906673.

Wanyu Lin, Hao Lan, and Baochun Li. Generative causal explanations for graph neural networks. In *International Conference on Machine Learning*, pp. 6666–6679. PMLR, 2021a.

Wanyu Lin, Hao Lan, and Baochun Li. Generative causal explanations for graph neural networks. In *ICML*, 2021b.

Ana Lucic, Maartje A Ter Hoeve, Gabriele Tolomei, Maarten De Rijke, and Fabrizio Silvestri. Cf-gnnexplainer: Counterfactual explanations for graph neural networks. In *AISTATS*, pp. 4499–4511, 2022.

Dongsheng Luo, Wei Cheng, Dongkuan Xu, Wenchao Yu, Bo Zong, Haifeng Chen, and Xiang Zhang. Parameterized explainer for graph neural network. *Advances in neural information processing systems*, 33:19620–19631, 2020.

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

Sahil Manchanda, Akash Mittal, Anuj Dhawan, Sourav Medya, Sayan Ranu, and Ambuj Singh. Gcomb: Learning budget-constrained combinatorial algorithms over billion-sized graphs. *Advances in Neural Information Processing Systems*, 33:20000–20011, 2020.

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

Phillip E Pope, Soheil Kolouri, Mohammad Rostami, Charles E Martin, and Heiko Hoffmann. Explainability methods for graph convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10772–10781, 2019.

Mario Alfonso Prado-Romero, Bardh Prenkaj, Giovanni Stilo, and Fosca Giannotti. A survey on graph counterfactual explanations: Definitions, methods, evaluation, and research challenges. *ACM Computing Surveys*, 2023.

Ladislav Rampášek, Mikhail Galkin, Vijay Prakash Dwivedi, Anh Tuan Luu, Guy Wolf, and Dominique Beaini. Recipe for a General, Powerful, Scalable Graph Transformer. *Advances in Neural Information Processing Systems*, 35, 2022.

Rishabh Ranjan, Siddharth Grover, Sourav Medya, Venkatesan Chakaravarthy, Yogish Sabharwal, and Sayan Ranu. Greed: A neural framework for learning graph distance functions. In *Advances in Neural Information Processing Systems*, 2022.

Kaspar Riesen and Horst Bunke. Iam graph database repository for graph based pattern recognition and machine learning. In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, pp. 287–297. Springer, 2008.

Thomas Schnake, Oliver Eberle, Jonas Lederer, Shinichi Nakajima, Kristof T Schütt, Klaus-Robert Müller, and Grégoire Montavon. Higher-order explanations of graph neural networks via relevant walks. *IEEE transactions on pattern analysis and machine intelligence*, 44(11):7581–7596, 2021.

Caihua Shan, Yifei Shen, Yao Zhang, Xiang Li, and Dongsheng Li. Reinforcement learning enhanced explainer for graph neural networks. In *NeurIPS 2021*, December 2021. URL https://www.microsoft.com/en-us/research/publication/reinforcement-learning-enhanced-explainer-for-graph-neural-networks/.

Richard Socher, A. Perelygin, J.Y. Wu, J. Chuang, C.D. Manning, A.Y. Ng, and C. Potts. Recursive deep models for semantic compositionality over a sentiment treebank. *EMNLP*, 1631:1631–1642, 01 2013.

Juntao Tan, Shijie Geng, Zuohui Fu, Yingqiang Ge, Shuyuan Xu, Yunqi Li, and Yongfeng Zhang. Learning and evaluating graph neural network explanations based on counterfactual and factual reasoning. In *Proceedings of the ACM Web Conference 2022*, WWW '22, pp. 1018–1027, 2022.

Abishek Thangamuthu, Gunjan Kumar, Suresh Bishnoi, Ravinder Bhattoo, NM Krishnan, and Sayan Ranu. Unravelling the performance of physics-informed graph neural networks for dynamical systems. In *Advances in Neural Information Processing Systems*, 2022.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=rJXMpikCZ.

Clement Vignac, Igor Krawczuk, Antoine Siraudin, Bohan Wang, Volkan Cevher, and Pascal Frossard. Digress: Discrete denoising diffusion for graph generation. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=UaAD-Nu86WX.

Philippe Vismara and Claude Laurenço. *An abstract representation for molecular graphs*, pp. 343–366. 04 2000. ISBN 9780821809877. doi: 10.1090/dimacs/051/26.

Minh Vu and My T Thai. Pgm-explainer: Probabilistic graphical model explanations for graph neural networks. *Advances in neural information processing systems*, 33:12225–12235, 2020.

Nikil Wale, Ian A Watson, and George Karypis. Comparison of descriptor spaces for chemical compound retrieval and classification. *Knowledge and Information Systems*, 14(3):347–375, 2008.

Xingchen Wan, Henry Kenlay, Binxin Ru, Arno Blaas, Michael Osborne, and Xiaowen Dong. Adversarial attacks on graph classifiers via bayesian optimisation. In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021.

Geemi P Wellawatte, Aditi Seshadri, and Andrew D White. Model agnostic generation of counterfactual explanations for molecules. *Chemical science*, 13(13):3697–3705, 2022.

Hao Wu, Ziyang Chen, Weiwei Sun, Baihua Zheng, and Wei Wang. Modeling trajectories with recurrent neural networks. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, IJCAI'17, pp. 3083–3090, 2017.

Yaochen Xie, Sumeet Katariya, Xianfeng Tang, Edward Huang, Nikhil Rao, Karthik Subbian, and Shuiwang Ji. Task-agnostic graph explanations. *NeurIPS*, 2022.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=ryGs6iA5Km.

Han Xuanyuan, Pietro Barbiero, Dobrik Georgiev, Lucie Charlotte Magister, and Pietro Lió. Global concept-based interpretability for graph neural networks via neuron analysis. 2023.

Pinar Yanardag and SVN Vishwanathan. Deep graph kernels. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1365–1374, 2015.

Chengxuan Ying, Tianle Cai, Shengjie Luo, Shuxin Zheng, Guolin Ke, Di He, Yanming Shen, and Tie-Yan Liu. Do transformers really perform badly for graph representation? In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=OeWooOxFwDa.

Rex Ying, Dylan Bourgeois, Jiaxuan You, Marinka Zitnik, and Jure Leskovec. Gnnexplainer: Generating explanations for graph neural networks. *Advances in neural information processing systems*, 32:9240, 2019a.

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, 2019b.

Jiaxuan You, Rex Ying, Xiang Ren, William L. Hamilton, and Jure Leskovec. Graphrnn: Generating realistic graphs with deep auto-regressive models. In Jennifer G. Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 5694–5703. PMLR, 2018. URL http://proceedings.mlr.press/v80/you18a.html.

Hao Yuan, Jiliang Tang, Xia Hu, and Shuiwang Ji. Xgnn: Towards model-level explanations of graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 430–438, 2020.

Hao Yuan, Haiyang Yu, Jie Wang, Kang Li, and Shuiwang Ji. On explainability of graph neural networks via subgraph explorations. In *ICML*, pp. 12241–12252. PMLR, 2021.

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*, 2022.

Yue Zhang, David Defazio, and Arti Ramesh. Relex: A model-agnostic relational model explainer. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 1042–1049, 2021.