Quantitative Assessment of GNN Counterfactual Explanation Robustness and Reproducibility under Adversarial Influence

Amir Reza Mohammadi[1,*], Michael M. Müller[1], Andreas Peintner[1], Beatriz Barroso Gstrein[1], Eva Zangerle[1] and Günther Specht[1]

[1]*Department of Computer Science, Universität Innsbruck, Austria*

Abstract

Within Machine Learning, Graph Neural Networks (GNNs) have emerged as prominent techniques, particularly excelling in tasks tailored for graph structures. Due to the intricate nature of GNNs and the essential role in conveying outcomes to users, there is a pressing demand to enhance the explainability of these approaches. Among state-of-the-art explanation strategies, counterfactual explanation provides intuitive and easily understandable insights into model predictions by showing how a small change in the input would lead to a different outcome. However, the absence of benchmarks and standardized tasks hampers the evaluation of such approaches. Moreover, there has not been an empirical comparison of counterfactuals and adversarial examples, both aiming to alter model outputs with minimal perturbations. This reproducibility study rigorously analyzes prominent GNN-based counterfactual explanation methods, contrasting them against established adversarial attack baselines. Our objective is to look into counterfactual methods through the lens of adversarials and thereby, explore the interconnectedness of these techniques and foster a deeper understanding of their combined utility and implications. We validate five selected GNN-based counterfactual explanation methods in two levels of local and model-level explanation and compare them to two well-established adversarial attack methods. Our findings reveal that adversarial methods can serve as a competitive baseline for counterfactual explanation on node classification, and in certain tasks, they may even outperform them.

Keywords

Graph Neural Networks, Counterfactual Explanation, Adversarial Examples, Reproducibility

1. Introduction

Graph Neural Networks (GNNs) [1] have witnessed a surge in prominence within the realm of Machine Learning (ML), showcasing remarkable effectiveness in tasks explicitly designed for graph-structured data [2, 3]. GNNs offer a powerful framework for capturing intricate relationships and patterns embedded in graph data. While the empirical success of GNNs is evident [4, 5], their black-box nature often hinders the interpretability of their decisions, limiting their broader adoption in critical domains.

Addressing the need for transparency and interpretability, Explainable Artificial Intelligence (XAI) has garnered substantial interest across various communities. Among these approaches, counterfactual explanation (CE) [6] is dedicated to advancing model explainability. CE not only provides intuitive and easily understandable insights into model predictions but also enables users to grasp how minor alterations in the input can lead to divergent outcomes. CE addresses a key question: "For a specific instance, how should the input features x be subtly perturbed for new features x' to yield a distinct predicted label (typically a desired label) from ML models?" CE promotes human interpretation through the comparison between x and x'. Departing from conventional CE studies centered on tabular or image data, there is a growing emphasis on CE within graphs [7, 8]. Despite the popularity of CE

36th GI-Workshop on Foundations of Databases (Grundlagen von Datenbanken), September 29 - October 01, 2025, Regensburg, Germany

[*] Corresponding author.

amir.reza@uibk.ac.at (A. R. Mohammadi)

0000-0003-3934-6941 (A. R. Mohammadi); 0009-0005-0918-8833 (M. M. Müller); 0000-0001-7337-524X (A. Peintner); 0000-0003-3195-8273 (E. Zangerle); 0000-0003-0978-7201 (G. Specht)

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073

published 2025-12-03

methods, the absence of established tasks, widely used metrics, and standardized benchmarks, has impeded comprehensive evaluations, hindering the establishment of robust and widely used baselines.

The exploration of adversarial attacks intersects with the study of counterfactuals (in fact, they have even been shown to be equivalent [9]). Adversarial Examples (AEs) are inputs that closely resemble authentic data but are misclassified by a trained ML model—for instance, an image of a turtle being classified as a rifle[1]. In this context, misclassified implies that the algorithm assigns the incorrect class or value compared to a predefined (usually human-provided) ground-truth [10]. The intriguing convergence of AE and CE in their shared goal of perturbing model outputs with minimal changes has ignited ongoing discussions within the research community [9, 11, 12, 13]. Surprisingly, there are no empirical comparisons between these two methodological paradigms, especially within the context of GNN counterfactuals [14].

Motivated by the lack of quantitative study and analysis of CE methods and also quantitative comparison of AE and CE, our study conducts a thorough reproducibility analysis of prominent GNN-based counterfactual explanation methods, juxtaposing their performance against established adversarial attack baselines. Our investigation systematically validates five selected GNN-based counterfactual explanation methods, namely CF^2 [15], CF-GNNExplainer [16], GCFExplainer [17], CLEAR [18], RCExplainer [19] which are the current SOTA methods in both model-level and instance level explanation (Figure 1). This examination aims to provide nuanced insights into the strengths, limitations, and contextual relevance of each approach. Moreover, to ensure a robust and meaningful comparison, two well-established adversarial attack methods are used as benchmarks against their counterfactual counterparts. By adopting a holistic perspective, we seek to foster a deeper understanding of their combined utility and potential applications within the dynamic landscape of GNNs.

We emphasize the significance of transparency and reproducibility in scientific research. Accordingly, we have made our code publicly accessible[2], providing comprehensive details of all comparative experiments.

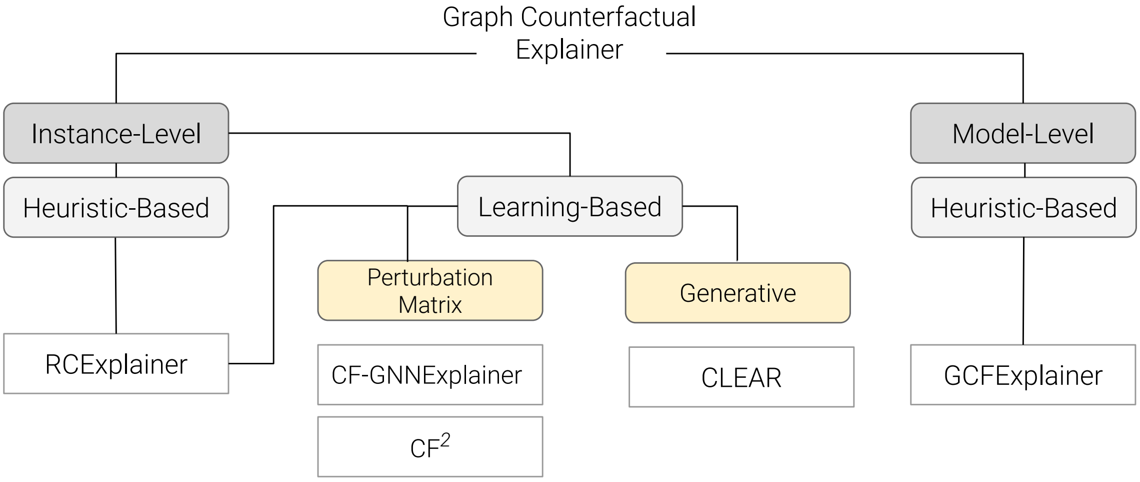

Figure 1: Taxonomy of Graph Counterfactual Explanation methods (adapted from [8]).

2. Background and Related Work

The field of interpretable ML has advanced significantly [20, 21, 22], particularly in local interpretability with early methods like LIME [23] and SHAP [22]. These works laid the foundation for post-hoc explainability, treating models as black boxes. Initial studies primarily focused on improving the

[1] https://www.theverge.com/2017/11/2/16597276/google-ai-image-attacks-adversarial-turtle-rifle-3d-printed
[2] https://github.com/amirreza-m95/CE_vs_AE

interpretability of the models themselves [24, 25]. In the context of GNNs, GNNExplainer [26] marked a breakthrough by identifying subgraphs responsible for node-level predictions, though it focused on factual explanations. The shift toward counterfactual explanations (CEs), as introduced by CF-GNNExplainer [16], enabled reasoning over “what-if” scenarios and inspired a range of new methods [18, 15, 17, 19]. While several surveys [7, 8, 27, 28] review the landscape, reproducibility studies specifically targeting GNN counterfactual explanations remain absent. To our knowledge, this is the first to empirically address that gap.

From a high-level view, CEs seek minimal perturbations that flip model predictions, closely resembling adversarial examples (AEs). While [9] highlight conceptual differences (e.g., “impossible worlds”), others argue for formal equivalence [29] or stress semantic and application-specific distinctions [11, 12, 13]. Freiesleben [14] emphasizes that AEs require misclassification, whereas CEs may retain the same class. This distinction becomes relevant in GNNs, where adversarial attacks (e.g., Nettack [30]) often degrade global performance rather than target local node predictions. CF-GNNExplainer also highlights this nuance. Despite these theoretical debates, empirical work comparing CEs and AEs in GNNs is scarce. This paper fills that gap through a systematic reproducibility and comparison study.

3. Problem Formulation and Definition

We denote a graph as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ represents the set of nodes and $\mathcal{E}$ denotes the set of edges. Each node $v_i \in \mathcal{V}$ is characterized by a feature vector $x_i \in \mathbb{R}^d$.

The existing body of literature on GNN explainability has predominantly concentrated on scenarios involving graph classification and node classification, with a focus on categorical output spaces (see [8] for a comprehensive survey of CE methods). In the context of graph classification, the input consists of a set of graphs, each associated with a specific class label. The objective for the GNN ϕ is to accurately predict these class labels. Conversely, in node classification, class labels are linked to individual nodes, and predictions are made at the node level.

In a message-passing GNN with l layers, the embedding of a node is intricately tied to its l-hop neighborhood. We introduce the term “neighborhood subgraph” to characterize this l-hop neighborhood. Henceforth, for the sake of clarity, we will employ the term “graph” to denote the neighborhood subgraph when referring to node classification.

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

$$\mathcal{G}^* = \arg\min \operatorname{dist}(\mathcal{G}, \mathcal{G}')\ s.t. \begin{cases} \mathcal{G}' \in \mathcal{G} \\ \Phi(\mathcal{G}) \neq \Phi(\mathcal{G}') \end{cases} \tag{1}$$

where $\operatorname{dist}(\mathcal{G}, \mathcal{G}')$ 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 preformed edge perturbations while keeping the node set fixed.

4. Experimental Setup

In the following, we detail the experimental setup for our studies.

4.1. Research Questions

Given our overall goals of (i) reproducing state-of-the-art counterfactual explanation methods, and (ii) comparing counterfactual explanation approaches to adversarial learning baselines, we address the following research questions in this paper:

(RQ1) How feasible is it to reproduce the outcomes of the state-of-the-art CE methods? To what degree do the underlying assumptions in these approaches withstand scrutiny? Which insights can be gained regarding the error modes associated with these methods?

(RQ2) What distinguishes counterfactual examples from adversarial examples within the framework of GNNs? How can these two directions mutually benefit each other?

(RQ3) Which CE performance evaluation metrics hold more promise? What are the respective advantages and drawbacks of each?

4.2. Datasets

We evaluate the algorithms on a diverse set of datasets, encompassing both synthetic and real-world scenarios. Specifically, we employ two synthetic datasets, BA-shapes and Tree-Cycles, introduced by [26], adhering to the same setup as outlined in their work. BA-Shapes and Tree-Cycles are employed for node classification, featuring predefined motifs ("house" and "cycle" structures) for interpretability. For real-world contexts, we utilize Mutagenicity [31, 15], NCI1 [32, 15] and Ogbg-molhiv [18]. The Mutagenicity dataset classifies molecules as either mutagenic or non-mutagenic, while the NCI1 dataset categorizes chemical compounds as positive or negative to cell lung cancer. Moreover, in Ogbg-molhiv, each graph represents a molecule, with each node denoting an atom and each edge symbolizing a chemical bond. Due to the unavailability of a ground-truth causal model, methods usually simulate both the label Y and the causal relations of interest R [18]. Additionally, we utilized Mutag0, a smaller subset of the Mutagenicity dataset. [34] made the assumption that the nitro group (NO2) and amino group (NH2) serve as the true contributors to mutagenicity. Consequently, they filtered out mutagens that did not contain these specific groups. However, NH2 has minimal impact on mutagenecity, with benzene-NO2 being the sole discriminative motif [35]. In response to this, a sub-dataset, Mutag0, has been crafted by [15], encompassing chemical compounds featuring benzene-NO2 that exhibit mutagenicity, or those lacking benzene-NO2 and displaying non-mutagenic properties and skipping other instances. An overview of the details of each dataset can be found in Table 1.

4.3. Methods

This study aims to replicate the most impactful and enduring methods from the CE era and time-tested approaches from the AE community. The selection of approaches is guided by the following considerations:

(a) The chosen methods must have a *significant influence* and highly known by community through being used as the SOTA baseline (see Table 2). (b) We prioritize methods with *diverse representation techniques* to enhance the generalizability of our research. This diversity is crucial for capturing a comprehensive understanding of GNN explanation methodologies, as illustrated in Figure 1. (c) The selected methods are specifically drawn from the GNN context to maintain *consistency* within the framework of this study.

Table 1
Dataset statistics

Dataset	# Graphs	# Nodes	# Edges	Features	# Classes
BA-Shapes [26]	1	700	4,100	10	4
Tree-Cycles [26]	1	871	1,950	10	2
Mutagenicity [31]	4,337	131,488	133,447	14	2
Mutag0 [15]	2,301	69,621	71,283	14	2
NCI1 [32]	4,110	122,747	132,753	37	2
Ogbg-molhiv [33]	41,127	1,049,163	2,259,376	9	2

Table 2
Methods overview. We list the counterfactual methods employed in our study, including the datasets, metrics, and baselines used in the original papers.

Method	Datasets	Metrics	Baselines
CF^2 [15] (TheWebConf22)	BA-shapes, Tree-Cycles, Mutag0, NCI1	Necessity, Sufficiency, #exp size	CF-GNNExplainer
CF-GNNExplainer [16] (AISTATS22)	BA-Shapes, Tree-Cycles, Tree-Grid	Fidelity, #exp size, Sparsity	Random, 1HOP, RM-1HOP
GCF-Explainer [17] (WSDM23)	NCI1, Mutagenicity, AIDS, Proteins	Coverage, Cost	CF^2, CF-GNNExp.
CLEAR [18] (NeurIPS22)	Community, Ogbg-molhiv, IMDB-M	Validity, Proximity, Causality	CF-GNNExp., MEG
RCExplainer [19] (NeurIPS22)	Mutagenicity, NCI1, BA-2motifs	Robustness, Efficiency, Fidelity, AUC	CF-GNNExplainer

4.3.1. Counterfactual Methods

We selected five representative counterfactual explanation methods (see Table 2) based on their influence and diversity.

CF^2 [15] solves a multi-objective optimization problem to balance factual and counterfactual reasoning, controlled by the parameter α. We consider both the optimized model ($\alpha = 0.6$) and the fully counterfactual variant ($\alpha = 0$).

GCFExplainer[17] is a model-level method for graph classification. It constructs a meta-graph of candidate counterfactuals and selects diverse explanations using vertex-reinforced random walks[36] and a greedy algorithm.

CF-GNNExplainer [16] is designed for node classification and learns a binary perturbation mask to sparsify the graph's adjacency matrix, minimizing changes needed to alter the prediction.

CLEAR [18] uses a variational autoencoder to generate counterfactuals in latent space. It outputs complete graphs with edge weights reflecting uncertainty, closely resembling the original input.

RCExplainer [19] identifies linear decision boundaries via an unsupervised strategy, enhancing robustness by generalizing across instances. It generates concise counterfactuals by selecting edge subsets guided by a boundary-based loss.

4.3.2. Adversarial Methods

We evaluate two widely used adversarial attack methods for graph-structured data [37]:

Nettack [30] targets node-level predictions by iteratively perturbing node features to deceive the GNN, while preserving the graph structure. It computes gradients to identify minimal changes that flip the model's output.

Meta Attack [38] uses a meta-learning approach to generate global adversarial attacks. Trained on various graph datasets and models, it can efficiently poison graph classifiers without requiring gradient access during inference.

4.3.3. Configuration

We employed the adversarial attack methods outlined earlier by utilizing the implementations available in the DeepRobust open-source project[3]. Subsequently, we seamlessly integrated these methods into our pipeline. We adhere to the recommended hyper-parameter settings provided by DeepRobust. It is important to highlight that certain modifications were necessary to harmonize these adversarial attack methods with counterfactual techniques, enabling a meaningful comparison of results on the same datasets. A noteworthy example is the Nettack method, which, by default, tends to both add and remove edges as a perturbation of the graph but to ensure a fair comparison with CE methods that primarily involve edge removal, we introduced constraints to the optimization method. These constraints guide the Nettack method to focus solely on edge removal, aligning it with the nature of CE methods. Further elaboration on these adjustments and their implications will be provided in Section 5.3. All the training is performed using an AMD Ryzen 2950X with 128GB RAM and a GeForce RTX 2070 with 8GB memory.

[3] https://github.com/DSE-MSU/DeepRobust

We repeat our experiments 3 times and report the average performance. We share both our dataset processing scripts, the source code, and the hyper-parameters using an anonymous repository[4].

4.4. Metrics

In this section, we discuss different evaluation metrics used in the community and compare them to clarify the rationale behind the metrics of our choice for this study.

Necessity [15, 28] measures the percentage of graphs in which the removal of the explanation subgraph induces a change in the GNN prediction, thereby establishing its necessity in influencing the model's output. Intuitively, Necessity quantifies the frequency with which removing subgraphs leads to prediction changes, divided by the total number of instances. This metric resembles with the metric called Validity or correctness introduced by [18]. In the context of explainable GNNs, Necessity refers to:

$$\text{Necessity}(\mathcal{N}) = \frac{\sum_{i=1}^{|\mathbb{G}|} \mathbb{1}\left(\Phi\left(\mathcal{R}^i\right) \neq \Phi\left(\mathcal{G}^i\right)\right)}{|\mathbb{G}|} \tag{2}$$

Where $\mathbb{G}$ is a graph set of $\mathcal{G}$, $\mathbb{R}$ is a residual graph set of $\mathcal{R}^i$, $\Phi\left(\mathbb{G}^i\right)$ is the prediction of the model on $\mathbb{G}^i$, $\mathcal{G}_S^i$ is explanation subgraph of $\mathcal{G}^i$ and $\mathcal{R}^i = \mathcal{G} - \mathcal{G}_S^i$. In the same setting of variables, Fidelity [16] is the exact opposite metric:

$$\text{Fidelity}(\mathcal{F}) = \frac{\sum_{i=1}^{|\mathbb{G}|} \mathbb{1}\left(\Phi\left(\mathcal{R}^i\right) = \Phi\left(\mathcal{G}^i\right)\right)}{|\mathbb{G}|} \tag{3}$$

As a result, in the context of counterfactual reasoning, we want lower values for Fidelity and higher values for Necessity and Validity.

Sufficiency is defined as the percentage of generated explanations that prove to be sufficient for an instance to achieve the same prediction as using the entire graph. In essence, Sufficiency intuitively quantifies the percentage of graphs where the explanation subgraph alone is capable of maintaining the GNN prediction unchanged.

$$\text{Sufficiency}(\mathcal{S}) = \frac{\sum_{i=1}^{|\mathbb{G}|} \mathbb{1}\left(\Phi\left(\mathcal{G}_S^i\right) = \Phi\left(\mathcal{G}^i\right)\right)}{|\mathbb{G}|} \tag{4}$$

Explanation size serves as a minimality evaluation metric which refers to the count of removed edges, representing the disparity between the original graph $\mathcal{G}$ and the counterfactual graph $\mathcal{G}'$. Given our aim to minimize explanations, a smaller value for this metric is preferable.

Coverage is a metric for evaluating recourse representation $\mathbb{C}$ for the graph classification task [17] which is the percentage of input graphs that possess nearby counterfactuals from C, within a specified distance threshold θ.

$$\text{Coverage}(\mathbb{C}) = \left|\left\{G \in \mathbb{G} \mid \min_{C \in \mathbb{C}}\{d(G, C)\} \leq \theta\right\}\right| / |\mathbb{G}| \tag{5}$$

In this context, [17] used the metric Cost which is recourse cost, representing the distance between each input graph and its respective counterfactual within the dataset.

$$\text{Cost}(\mathbb{C}) = \underset{G \in \mathbb{G}}{\text{agg}}\left\{\min_{C \in \mathbb{C}}\{d(G, C)\}\right\} \tag{6}$$

This metric also closely resembles with the explanation size from local CE.

[4] https://github.com/amirreza-m95/CE_vs_AE

5. Results

In this section, we present the results of our empirical study, shedding light on the performance and effectiveness of both adversarial examples and counterfactual explanations. Our experiments aim to provide insights into the comparative aspects of these methodological paradigms, addressing their impact.

5.1. CE Reproducibility Study (RQ1)

We assess the reproducibility of each counterfactual explanation method by replicating their experiments and evaluating consistency with reported results.

CF^2: All experiments ran smoothly except for the NCI1 dataset, due to missing code and data. We observed large fluctuations in the Necessity metric (0.58–0.90), depending on classifier performance and preprocessing. This suggests high sensitivity to the underlying model. Using the original Mutagenicity dataset (vs. Mutag0) also caused a drop in classification accuracy, though explanation performance remained stable.

CF-GNNExplainer: Reproduced results successfully despite minor code deprecations. Model performance was sensitive to hyperparameters, requiring re-optimization to match reported values.

GCFExplainer: As the only model-level method, it reproduced well using both pretrained and freshly trained models. However, its pretraining phase (VRRW) was resource-intensive, requiring a 256GB RAM machine due to memory constraints.

CLEAR: Results were partially reproducible. We matched the validity score on IMDB-M (0.91 vs. 0.96 reported) but could not evaluate the Community dataset due to unavailable data. On larger datasets (e.g., Mutagenicity), CLEAR failed due to memory issues, though it worked on the smaller Mutag0.

RCExplainer: The original code link was inactive, but we obtained a working version from the authors[5]. With that, we reproduced the results using both pretrained models and our own training.

Table 3
Explanation evaluation of graph classification methods based on the sufficiency metric (Suff.). Here, #exp represents the size of the generated explanations. The highest scores are indicated in bold font.

Methods	Mutagenicity		NCI1	
	Suff.	#exp	Suff.	#exp
RCExplainer	**0.64**	3.0	0.52	3.0
$CF^2(Opt.)$	0.47	3.18	**0.62**	17.70
CF^2 $(\alpha = 0)$	0.51	2.68	0.53	13.21
GCFExplainer	0.57	**1.01**	0.54	**1.21**

Methods	Mutag0		Ogbg-molhiv	
	Suff.	Size	Suff.	Size
$CF^2(Opt.)$	0.73	3.3	**0.92**	**17.06**
CF^2 $(\alpha = 0)$	**0.82**	**2.8**	0.90	18.04
CLEAR	0.61	19.32	OOM	OOM

5.2. Comparing Different CE-Methods (RQ1)

In this section, we undertake a comparative analysis of the methods within the two primary categories of node and graph classification tasks. The evaluation is based on metrics outlined in Section 4.4, specifically Necessity, Sufficiency, Explanation Size, Coverage and Cost. For node classification, we utilize the BA-shapes and Tree-Cycles datasets, while for graph classification, we employ the Mutagenicity and

[5] https://developer.huaweicloud.com/develop/aigallery/notebook/detail?id=e41f63d3-e346-4891-bf6a-40e64b4a3278

NCI1 datasets. These datasets were chosen due to their widespread adoption in the field, as indicated in Table 2. To ensure a thorough comparison, we attempted to use the most commonly adopted datasets and metrics. As a result, for some of the methods, we employed metrics and datasets that differ from those originally used in their respective research. We believe this approach opens up new possibilities for integrating various metrics and datasets, potentially enhancing the robustness of evaluations across different methodologies. The only exception to this approach is CLEAR, which utilized metrics and datasets not used in other papers. To ensure a fair comparison, we adapted the CLEAR method to the Mutag0 (non-original for CLEAR) dataset and also adapted CF^2 to the Ogbg-molhiv dataset (from CLEAR), facilitating a more comprehensive evaluation of the method. For an overview of this experiment, we refer to Tables 3 to 4.

Our comparison for the graph classification task is shown in Table 3. The evaluation is based on the sufficiency and explanation size metrics, with higher sufficiency and lower size values indicating better performance. RCExplainer demonstrates strong performance on the Mutagenicity dataset with a sufficiency score of 0.64. On the other hand, CF^2 (Opt.) displays a lower sufficiency score on both datasets compared to RCExplainer and GCFExplainer since it notably exhibits a much larger size, particularly on the NCI1 dataset with a size of 17.70, indicating a less concise explanation. Similarly, CF^2 ($\alpha = 0$) also shows lower sufficiency scores on both datasets as expected and reported by authors. While CF^2 ($\alpha = 0$) achieves competitive sufficiency scores, its larger explanation size makes it less concise than RCExplainer and GCFExplainer. In contrast, RCExplainer, which provides moderate sufficiency scores and slightly better performance on NCI1, consistently produces smaller explanations, indicating higher conciseness across both datasets.

In summary, RCExplainer stands out for its balance of high sufficiency and consistent size across datasets, making it a strong candidate for applications where interpretability and efficiency are paramount. Conversely, while CF^2 (Opt.) and CF^2 ($\alpha = 0$) offer competitive sufficiency scores, their larger sizes may indicate more complex explanations. GCFExplainer, with its moderate sufficiency scores and consistently low sizes, presents a viable alternative for scenarios where a balance between interpretability and complexity is desired. We also have to consider that GCFExplainer is the only model-level explanation method.

In Table 4, we include the Necessity metric to compare node classification methods. Sufficiency values for CF-GNNExplainer are not presented in the paper as this method does not incorporate sufficiency in its optimization process. CF-GNNExplainer primarily concentrates on minimizing perturbations to change the class label without explicitly optimizing for generating a concise summary of the graph. However, CF-Explainer demonstrates moderate performance with Necessity scores of 0.61 and 0.79 on BA-Shapes and Tree-Cycles datasets respectively, indicating its capability to identify essential features for classification. CF-Explainer exhibits relatively low sizes on both datasets, with values of 2.39 and 2.09, suggesting concise explanations.

CF^2 ($\alpha = 0$) and CF^2 ($\alpha = Opt.$) outperform CF-GNNExplainer in terms of Necessity on both datasets. However, both variants of CF^2 display larger sizes compared to CF-Explainer, with values ranging from 3.6 to 7.76, suggesting potentially more complex explanations. To facilitate a more comprehensive comparison between these two methods while maintaining a fixed explanation value, we evaluated their performance on the Necessity metric. This analysis revealed that CF-GNNExplainer exhibits superior performance at lower explanation size values. However, as we progress towards higher explanation size values, CF^2 outperforms CF-GNNExplainer. We refer to the results depicted in Figure 2 for further insights into the comparison between these two methods.

Overall, CF-Explainer offers concise explanations with moderate Necessity scores, but lacks at sufficiency. CF^2 variants provide more comprehensive explanations with higher Necessity and sufficiency scores, albeit at the cost of larger sizes, indicating potentially more complex explanations. Depending on the specific requirements of the application, practitioners may choose between CF-Explainer for its simplicity and CF^2 variants for their comprehensiveness. These findings are novel, as no prior studies have conducted such evaluation and comparison of these methods.

Table 3 compares CF^2, CLEAR, and others across Mutag0 and Ogbg-molhiv. CF^2 offers compact, high-sufficiency explanations, while CLEAR suffers from scalability issues, failing on larger datasets due to memory limitations. These results emphasize the need to balance accuracy, explanation size, and scalability in real-world graph classification tasks.

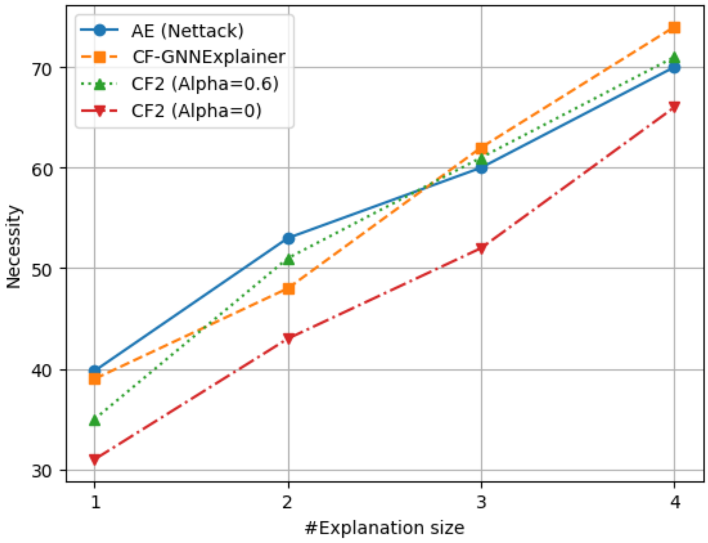

Figure 2: Comparative evaluation of AE and CE (CF2, CF-GNNExplainer) methods on node classification task on BA-Shapes dataset.

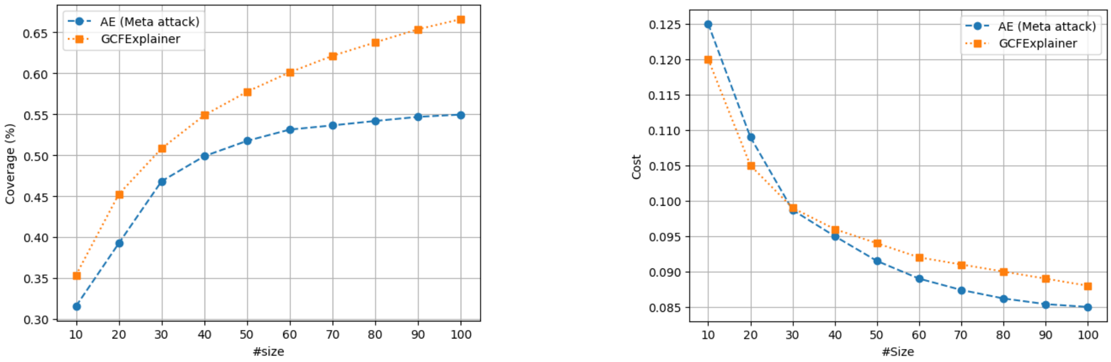

Figure 3: Coverage performance comparison between GCFExplainer and Meta attack based on different counterfactual summary sizes on Mutagenicity dataset.

5.3. GNN Counterfactuals Compared with Adversarial Examples (RQ2)

Building on the theoretical connections between counterfactual explanations (CEs) and adversarial examples (AEs), we compare two representative methods from each: Nettack vs. CF^2 on the BA-Shapes dataset (node classification) and Meta Attack vs. GCFExplainer on Mutagenicity (graph classification). Results are summarized in Figures 2 and 3.

To ensure fair comparison, we adapted Nettack to align with CF^2's constraints by limiting it to edge deletions within 3-hop neighborhoods and using the same evaluation pipeline. As shown in Figure 2, Nettack achieves higher Necessity scores at low perturbation levels, consistent with its goal of minimal changes. This supports its role as a strong baseline when minimal perturbations are desired.

In graph classification, Figure 3 shows that GCFExplainer outperforms Meta Attack in both coverage and cost, indicating that AEs are less effective for generating diverse, interpretable explanations. Meta Attack introduces larger changes while remaining less competitive, highlighting the trade-offs in global adversarial perturbations.

Table 4
Explanation evaluation of node classification methods based on sufficiency (Suff.) and Necessity metrics (Nec.). Here, #exp represents the size of the generated explanations. The highest scores are indicated in bold font.

Methods	BA-Shapes			Tree-Cycles		
	Nec.	Suff.	#exp	Nec.	Suff.	#exp
CF-Explainer	0.61	NA	**2.39**	0.79	NA	**2.09**
CF^2 ($\alpha = 0$)	0.72	0.63	7.76	**1.0**	0.74	3.6
CF^2 ($\alpha = Opt.$)	**0.73**	**0.67**	5.7	1.0	**0.87**	6.56

5.4. Insights and Observations on Evaluation Metrics (RQ3)

In this section, we delve into the metrics used for comparing and analyzing methods, as outlined in Table 2. It's noteworthy that none of the papers we studied used the same metrics for evaluation, posing challenges when comparing methods. Additionally, conflicts in the interpretation of these metrics further complicate matters. For example, regarding sufficiency, [15] advocates for higher values, while [28] argues that higher values indicate superior performance for factual explanations, yet lower values are preferred for counterfactual scenarios where the goal is to flip the class label.

Moreover, there are limitations in these metrics' ability to guide model improvement. For instance, in the case of the Fidelity metric, [16] showed that the Random algorithm outperforms all other methods with 0.0 percent accuracy, leaving no room for enhancement. This misconfiguration arises since the metric evaluates correctness based on ground truth labels rather than predictions, resulting in random perturbations failing to impact model performance, leading to a minimum fidelity score. Additionally, as observed in the Necessity metric for CF^2, results fluctuate depending on classifier performance, underscoring the need for benchmarked metrics within the CE community. These insights underscore the complexities in metric interpretation and stress the importance of standardized evaluation protocols in CE research.

6. Conclusion and Future Work

In this reproducibility paper, we conducted a comprehensive empirical study on prominent GNN-based counterfactual explanation methods, juxtaposing their performance against established adversarial attack baselines. Through our investigation, we systematically validated five selected GNN-based CE methods, namely CF^2, CF-GNNExplainer, GCFExplainer, CLEAR, and RCExplainer, shedding light on their strengths, limitations, and contextual relevance.

Our comparative analysis revealed nuanced insights into the performance of these methods across various datasets and tasks. Notably, RCExplainer emerged as a standout performer in graph classification tasks, exhibiting a balance of high sufficiency and consistent explanation size. Conversely, while CF^2 variants displayed competitive sufficiency scores, they often presented larger explanation sizes, potentially indicating more complex explanations.

Our study lays the groundwork for several avenues of future research aimed at advancing the field of XAI and GNNs. Here, we outline potential directions for further exploration: (a) The intersection of CE and adversarial attacks presents opportunities for developing hybrid approaches that leverage the strengths of both paradigms. Future work could explore the integration of CE and adversarial defense strategies to develop more robust and interpretable AI systems. (b) Future research could explore the development of context-aware counterfactual explanation methods tailored to specific application domains. Context-awareness involves considering additional contextual information such as user preferences, domain-specific constraints, and situational factors when generating explanations. (c) Future research could focus on unifying and standardizing metrics used for evaluating counterfactual explanation methods.

Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

References

[1] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, G. Monfardini, The graph neural network model, IEEE Trans. Neural Networks 20 (2009) 61–80. URL: https://doi.org/10.1109/TNN.2008.2005605.

[2] P. Yanardag, S. V. N. Vishwanathan, Deep graph kernels, in: L. Cao, C. Zhang, T. Joachims, G. I. Webb, D. D. Margineantu, G. Williams (Eds.), Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, August 10-13, 2015, ACM, 2015, pp. 1365–1374. URL: https://doi.org/10.1145/2783258.2783417.

[3] P. Sen, G. Namata, M. Bilgic, L. Getoor, B. Gallagher, T. Eliassi-Rad, Collective classification in network data, AI Mag. 29 (2008) 93–106. URL: https://doi.org/10.1609/aimag.v29i3.2157.

[4] T. N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, in: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings, OpenReview.net, 2017. URL: https://openreview.net/forum?id=SJU4ayYgl.

[5] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Liò, Y. Bengio, Graph attention networks, in: 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings, OpenReview.net, 2018. URL: https://openreview.net/forum?id=rJXMpikCZ.

[6] F. Doshi-Velez, B. Kim, Towards a rigorous science of interpretable machine learning, 2017. arXiv:1702.08608.

[7] H. Yuan, H. Yu, S. Gui, S. Ji, Explainability in graph neural networks: A taxonomic survey, CoRR abs/2012.15445 (2020). URL: https://arxiv.org/abs/2012.15445. arXiv:2012.15445.

[8] M. A. Prado-Romero, B. Prenkaj, G. Stilo, F. Giannotti, A survey on graph counterfactual explanations: Definitions, methods, evaluation, CoRR abs/2210.12089 (2022). URL: https://doi.org/10.48550/arXiv.2210.12089. arXiv:2210.12089.

[9] S. Wachter, B. D. Mittelstadt, C. Russell, Counterfactual explanations without opening the black box: Automated decisions and the GDPR, CoRR abs/1711.00399 (2017). URL: http://arxiv.org/abs/1711.00399. arXiv:1711.00399.

[10] P. Chen, H. Zhang, Y. Sharma, J. Yi, C. Hsieh, ZOO: zeroth order optimization based black-box attacks to deep neural networks without training substitute models, in: B. Thuraisingham, B. Biggio, D. M. Freeman, B. Miller, A. Sinha (Eds.), Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security, AISec@CCS 2017, Dallas, TX, USA, November 3, 2017, ACM, 2017, pp. 15–26. URL: https://doi.org/10.1145/3128572.3140448.

[11] S. Verma, J. P. Dickerson, K. Hines, Counterfactual explanations for machine learning: A review, CoRR abs/2010.10596 (2020). URL: https://arxiv.org/abs/2010.10596. arXiv:2010.10596.

[12] R. McGrath, L. Costabello, C. L. Van, P. Sweeney, F. Kamiab, Z. Shen, F. Lécué, Interpretable credit application predictions with counterfactual explanations, CoRR abs/1811.05245 (2018). URL: http://arxiv.org/abs/1811.05245. arXiv:1811.05245.

[13] T. Laugel, M. Lesot, C. Marsala, X. Renard, M. Detyniecki, Unjustified classification regions and counterfactual explanations in machine learning, in: U. Brefeld, É. Fromont, A. Hotho, A. J. Knobbe, M. H. Maathuis, C. Robardet (Eds.), Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2019, Würzburg, Germany, September 16-20, 2019, Proceedings, Part II, volume 11907 of *Lecture Notes in Computer Science*, Springer, 2019, pp. 37–54. URL: https://doi.org/10.1007/978-3-030-46147-8_3.

[14] T. Freiesleben, The intriguing relation between counterfactual explanations and adversarial examples, Minds Mach. 32 (2022) 77–109. URL: https://doi.org/10.1007/s11023-021-09580-9.

[15] J. Tan, S. Geng, Z. Fu, Y. Ge, S. Xu, Y. Li, Y. Zhang, Learning and evaluating graph neural

network explanations based on counterfactual and factual reasoning, in: F. Laforest, R. Troncy, E. Simperl, D. Agarwal, A. Gionis, I. Herman, L. Médini (Eds.), WWW '22: The ACM Web Conference 2022, Virtual Event, Lyon, France, April 25 - 29, 2022, ACM, 2022, pp. 1018–1027. URL: https://doi.org/10.1145/3485447.3511948.

[16] A. Lucic, M. A. ter Hoeve, G. Tolomei, M. de Rijke, F. Silvestri, Cf-gnnexplainer: Counterfactual explanations for graph neural networks, in: G. Camps-Valls, F. J. R. Ruiz, I. Valera (Eds.), International Conference on Artificial Intelligence and Statistics, AISTATS 2022, 28-30 March 2022, Virtual Event, volume 151 of *Proceedings of Machine Learning Research*, PMLR, 2022, pp. 4499–4511. URL: https://proceedings.mlr.press/v151/lucic22a.html.

[17] Z. Huang, M. Kosan, S. Medya, S. Ranu, A. K. Singh, Global counterfactual explainer for graph neural networks, in: T. Chua, H. W. Lauw, L. Si, E. Terzi, P. Tsaparas (Eds.), Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining, WSDM 2023, Singapore, 27 February 2023 - 3 March 2023, ACM, 2023, pp. 141–149. URL: https://doi.org/10.1145/3539597.3570376.

[18] J. Ma, R. Guo, S. Mishra, A. Zhang, J. Li, CLEAR: generative counterfactual explanations on graphs, in: S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, A. Oh (Eds.), Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022, 2022. URL: http://papers.nips.cc/paper_files/paper/2022/hash/a69d7f3a1340d55c720e572742439eaf-Abstract-Conference.html.

[19] M. Bajaj, L. Chu, Z. Y. Xue, J. Pei, L. Wang, P. C. Lam, Y. Zhang, Robust counterfactual explanations on graph neural networks, in: M. Ranzato, A. Beygelzimer, Y. N. Dauphin, P. Liang, J. W. Vaughan (Eds.), Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual, 2021, pp. 5644–5655. URL: https://proceedings.neurips.cc/paper/2021/hash/2c8c3a57383c63caef6724343eb62257-Abstract.html.

[20] A. Ghorbani, J. Wexler, J. Y. Zou, B. Kim, Towards automatic concept-based explanations, in: H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, R. Garnett (Eds.), Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada, 2019, pp. 9273–9282. URL: https://proceedings.neurips.cc/paper/2019/hash/77d2afcb31f6493e350fca61764efb9a-Abstract.html.

[21] A. R. Mohammadi, A. Peintner, M. Müller, E. Zangerle, Are we explaining the same recommenders? incorporating recommender performance for evaluating explainers, in: RecSys '24, ACM, 2024, p. 1113–1118.

[22] A. Ghorbani, J. Y. Zou, Neuron shapley: Discovering the responsible neurons, in: H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin (Eds.), Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual, 2020. URL: https://proceedings.neurips.cc/paper/2020/hash/41c542dfe6e4fc3deb251d64cf6ed2e4-Abstract.html.

[23] M. T. Ribeiro, S. Singh, C. Guestrin, "why should I trust you?": Explaining the predictions of any classifier, in: B. Krishnapuram, M. Shah, A. J. Smola, C. C. Aggarwal, D. Shen, R. Rastogi (Eds.), Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016, ACM, 2016, pp. 1135–1144. URL: https://doi.org/10.1145/2939672.2939778.

[24] A. Peintner, A. R. Mohammadi, E. Zangerle, SPARE: shortest path global item relations for efficient session-based recommendation, in: J. Zhang, L. Chen, S. Berkovsky, M. Zhang, T. D. Noia, J. Basilico, L. Pizzato, Y. Song (Eds.), Proceedings of the 17th ACM Conference on Recommender Systems, RecSys 2023, Singapore, Singapore, September 18-22, 2023, ACM, 2023, pp. 58–69. URL: https://doi.org/10.1145/3604915.3608768.

[25] A. Peintner, A. R. Mohammadi, E. Zangerle, Efficient session-based recommendation with contrastive graph-based shortest path search, ACM Trans. Recomm. Syst. 3 (2025). URL:

https://doi.org/10.1145/3701764. doi:10.1145/3701764.

[26] Z. Ying, D. Bourgeois, J. You, M. Zitnik, J. Leskovec, Gnnexplainer: Generating explanations for graph neural networks, in: H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, R. Garnett (Eds.), Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada, 2019, pp. 9240–9251. URL: https://proceedings.neurips.cc/paper/2019/hash/d80b7040b773199015de6d3b4293c8ff-Abstract.html.

[27] Z. Guo, T. Xiao, C. Aggarwal, H. Liu, S. Wang, Counterfactual learning on graphs: A survey, CoRR abs/2304.01391 (2023). URL: https://doi.org/10.48550/arXiv.2304.01391. arXiv:2304.01391.

[28] M. Kosan, S. Verma, B. Armgaan, K. Pahwa, A. K. Singh, S. Medya, S. Ranu, GNNX-BENCH: unravelling the utility of perturbation-based GNN explainers through in-depth benchmarking, CoRR abs/2310.01794 (2023). URL: https://doi.org/10.48550/arXiv.2310.01794. arXiv:2310.01794.

[29] K. Browne, B. Swift, Semantics and explanation: why counterfactual explanations produce adversarial examples in deep neural networks, CoRR abs/2012.10076 (2020). URL: https://arxiv.org/abs/2012.10076. arXiv:2012.10076.

[30] D. Zügner, A. Akbarnejad, S. Günnemann, Adversarial attacks on neural networks for graph data, in: Y. Guo, F. Farooq (Eds.), Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018, ACM, 2018, pp. 2847–2856. URL: https://doi.org/10.1145/3219819.3220078.

[31] A. K. Debnath, R. L. Lopez de Compadre, G. Debnath, A. J. Shusterman, C. Hansch, Structure-activity relationship of mutagenic aromatic and heteroaromatic nitro compounds. correlation with molecular orbital energies and hydrophobicity, Journal of Medicinal Chemistry 34 (1991) 786–797. URL: https://doi.org/10.1021/jm00106a046. arXiv:https://doi.org/10.1021/jm00106a046.

[32] N. Wale, G. Karypis, Comparison of descriptor spaces for chemical compound retrieval and classification, in: Proceedings of the 6th IEEE International Conference on Data Mining (ICDM 2006), 18-22 December 2006, Hong Kong, China, IEEE Computer Society, 2006, pp. 678–689. URL: https://doi.org/10.1109/ICDM.2006.39.

[33] W. Hu, M. Fey, M. Zitnik, Y. Dong, H. Ren, B. Liu, M. Catasta, J. Leskovec, Open graph benchmark: Datasets for machine learning on graphs, in: H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin (Eds.), Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual, 2020. URL: https://proceedings.neurips.cc/paper/2020/hash/fb60d411a5c5b72b2e7d3527cfc84fd0-Abstract.html.

[34] D. Luo, W. Cheng, D. Xu, W. Yu, B. Zong, H. Chen, X. Zhang, Parameterized explainer for graph neural network, in: H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin (Eds.), Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual, 2020. URL: https://proceedings.neurips.cc/paper/2020/hash/e37b08dd3015330dcbb5d6663667b8b8-Abstract.html.

[35] W. Lin, H. Lan, B. Li, Generative causal explanations for graph neural networks, in: M. Meila, T. Zhang (Eds.), Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event, volume 139 of *Proceedings of Machine Learning Research*, PMLR, 2021, pp. 6666–6679. URL: http://proceedings.mlr.press/v139/lin21d.html.

[36] W. Xiao, H. Zhao, V. W. Zheng, Y. Song, Vertex-reinforced random walk for network embedding, in: C. Demeniconi, N. V. Chawla (Eds.), Proceedings of the 2020 SIAM International Conference on Data Mining, SDM 2020, Cincinnati, Ohio, USA, May 7-9, 2020, SIAM, 2020, pp. 595–603. URL: https://doi.org/10.1137/1.9781611976236.67.

[37] L. Sun, Y. Dou, C. J. Yang, K. Zhang, J. Wang, P. S. Yu, L. He, B. Li, Adversarial attack and defense on graph data: A survey, IEEE Trans. Knowl. Data Eng. 35 (2023) 7693–7711. URL: https://doi.org/10.1109/TKDE.2022.3201243.

[38] D. Zügner, S. Günnemann, Adversarial attacks on graph neural networks via meta learning, in: 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019, OpenReview.net, 2019. URL: https://openreview.net/forum?id=Bylnx209YX.