ATEX-CF: ATTACK-INFORMED COUNTERFACTUAL EXPLANATIONS FOR GRAPH NEURAL NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Counterfactual explanations offer an intuitive way to interpret graph neural networks (GNNs) by identifying minimal changes that alter a model's prediction, thereby answering "*what must differ for a different outcome?*". In this work, we propose a novel framework, ATEX-CF that unifies adversarial attack techniques with counterfactual explanation generation—a connection made feasible by their shared goal of flipping a node's prediction, yet differing in perturbation strategy: adversarial attacks often rely on edge additions, while counterfactual methods typically use deletions. Unlike traditional approaches that treat explanation and attack separately, our method efficiently integrates both edge additions and deletions, grounded in theory, leveraging adversarial insights to explore impactful counterfactuals. In addition, by jointly optimizing fidelity, sparsity, and plausibility under a constrained perturbation budget, our method produces instance-level explanations that are both informative and realistic. Experiments on synthetic and real-world node classification benchmarks demonstrate that ATEX-CF generates faithful, concise, and plausible explanations, highlighting the effectiveness of integrating adversarial insights into counterfactual reasoning for GNNs.

1 INTRODUCTION

Graph neural networks excel at node classification by recursively aggregating neighbor features and graph topology, yet their opaque inference undermines trust in critical applications such as healthcare, finance, and scientific discovery (Chen et al., 2024; Zhong et al., 2025). This limitation has spurred research into GNN explainability, with *counterfactual methods* (Yuan et al., 2022; Qiu et al., 2025; Prado-Romero et al., 2024) in particular aiming to determine the smallest modifications to node features or graph structure that cause a model's prediction to change.

Meanwhile, *adversarial attacks* (Zhang et al., 2024; Zhu et al., 2024; Sun et al., 2023) on GNNs have become an equally important line of research, as GNNs can be undermined by minimal, strategically crafted graph-structure perturbations, highlighting the need for robustness analysis. Consequently, robustness against adversarial attacks has become a key priority in GNN research.

Traditional counterfactual graph generation methods, e.g., CF^2 (Tan et al., 2022b), GCFExplainer (Huang et al., 2023), primarily rely on *edge deletion* to identify crucial substructures that support a particular prediction. While effective, this deletion-centric perspective overlooks the role of *missing relations* in the original graph whose addition could substantially influence predictions. In parallel, extensive studies in graph adversarial learning have demonstrated that adding a small (e.g., 2) number of carefully selected edges can effectively flip the prediction of a target node (Chen et al., 2025; Zhu et al., 2024). Such added edges—though absent in the input graph—often correspond to semantically plausible and structurally coherent relations.

Despite their importance, current approaches address these two directions largely in isolation. From a counterfactual reasoning perspective, adversarially added edges naturally serve as *actionable candidates* for counterfactual generation: *They represent the minimal structural additions required to alter the model's decision.* However, existing counterfactual methods, which predominantly rely on edge deletion, have largely overlooked the potential of incorporating edge-addition information derived from adversarial attack strategies.

Motivated by these insights, *we design a unified framework, ATEX-CF*[1] *that incorporates attack semantics into counterfactual generation in a controlled and interpretable manner.* Extending counterfactual generation to include *edge addition* has significant benefits. From a **quantitative** perspective, we demonstrate that edge-addition counterfactuals can (1) increase the likelihood of flipping predictions and (2) achieve this with a smaller perturbation budget. From a **qualitative** perspective, they provide practical advantages: (1) **Complementary explanatory coverage** — while edge-deletion counterfactual identifies which existing relations are crucial for a prediction, edge-addition candidates reveal which missing relations could have altered the outcome. For example, in healthcare, a GNN may classify a patient as low-risk for heart disease due to the lack of an edge representing "symptom–drug correlation", while introducing an edge "patient medication record → cardiac side effects" can flip the prediction and reveal hidden reasoning paths. (2) **Uncovering model bias and data deficiencies** — adding certain edges can divulge over-reliance on specific nodes or structural biases. For example, a paper may be misclassified as "theoretical mathematics" due to missing citation edges to authoritative AI conferences. Introducing an edge "paper → ICLR Best Paper Award" corrects the prediction, highlighting dataset limitations and model vulnerabilities.

Case Study. To illustrate the limitations of existing counterfactual methods, consider a scenario from the *Loan-Decision* dataset (Ma et al., 2025). Loan approval is granted when both conditions are met: `income` > 5 and `degree` > 3. Applicant `Alice` has income 6 (satisfies condition) but degree 3 (fails). The model predicts rejection. Classical **deletion-based** counterfactual methods fail here–removing edges further reduces degree. Unconstrained **edge additions** (e.g., linking to a billionaire) succeed but can be implausible. Our method ATEX-CF identifies a feasible peer connection that serves as an actionable update and flips the prediction.

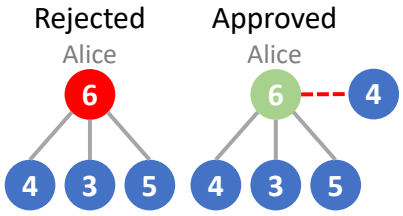

Figure 1: Illustration of counterfactual limitations in the Loan Decision dataset.

While this fusion is promising, combining adversarial attacks with counterfactual explanations is non-trivial. Adversarial edges are optimized for misclassifications rather than interpretability, raising challenges in ensuring the qualities of a good counterfactual explanation, such as *high impact*, *sparsity*, and *plausibility* (Longa et al., 2025). Furthermore, when considering missing edge additions to the input graph, the search space of possible perturbations remains combinatorially large, requiring principled mechanisms to balance effectiveness with efficiency.

Our contributions can be summarized as follows:

- **Unified perspective.** We establish, for the first time, a theoretical bridge between adversarial attacks and counterfactual explanations in GNNs, showing that adversarial edge additions can be repurposed as counterfactual candidates. This connection provides a principled foundation for unifying attack and explanation.
- **Hybrid counterfactual framework.** We design a novel solution, ATEX-CF, that simultaneously leverages *edge deletions* (traditional counterfactual explanations) and *attack-informed edge additions* (from adversarial strategies), thereby offering a more comprehensive and actionable counterfactual than deletion-only approaches.
- **Enhanced explanatory coverage.** By incorporating edge-addition counterfactual, ATEX-CF uncovers missing but semantically plausible relations, complements deletion-based explanations, and enables proactive optimization (e.g., suggesting constructive graph modifications rather than only indicating critical existing edges).
- **Efficiency and controllability.** We exploit adversarial attack logistics to form a focused candidate space, significantly reducing the combinatorial complexity of our counterfactual search. In addition, ATEX-CF integrates sparsity and plausibility constraints to ensure interpretable and realistic explanations.
- **Empirical validation.** Through experiments on benchmark datasets, we demonstrate that ATEX-CF improves explanatory power, maintains semantic plausibility, and reduces computational burden compared with state-of-the-art counterfactual generation and adversarial attack methods.

[1] abbreviation for **At**tack **Ex**planation **C**ounter**f**actual

2 PRELIMINARIES

2.1 NODE CLASSIFICATION AND GRAPH NEURAL NETWORKS

Node Classification in a Graph. We consider the task of node classification in a graph, denoted as $G = (V, E, \mathbf{X})$, where V is a set of nodes, $E \subseteq \{(v, w) \mid v, w \in V\}$ is a set of undirected, unweighted edges, and $\mathbf{X} = \{\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_{N-1}\}$ comprises node feature vectors with $\mathbf{x}_i \in \mathbb{R}^d$ for each node v_i. The adjacency matrix $\mathbf{A} \in \{0, 1\}^{N \times N}$ has entries $\mathbf{A}_{vw} = 1$ if $(v, w) \in E$ and 0 otherwise. A subset $V_L \subseteq V$ is labeled, forming training data; each labeled node has a class $y_v \in \mathcal{C} = \{1, \ldots, c\}$. The goal is to predict the label of a target node $v \in V$ in a supervised manner given $\mathbf{A}$ and $\mathbf{X}$. Key mathematical symbols are summarized in Table 5 in the Appendix.

Graph Neural Networks. Graph Neural Networks classify nodes through a message-passing scheme (Kipf & Welling, 2017). Each node representation is iteratively updated by aggregating and transforming information from its neighbors. For the Graph Convolutional Network (GCN), a prominent GNN, the hidden representation at layer $l + 1$ is $\mathbf{H}^{(l+1)} = \sigma\Big(\hat{\mathbf{A}}\mathbf{H}^{(l)}\mathbf{W}^{(l)}\Big)$, where $\mathbf{H}^{(0)} = \mathbf{X}$, σ is a nonlinear activation, $\mathbf{A}_{\text{self}} = \mathbf{A} + \mathbf{I}_N$ augments the adjacency with self-loops, $\mathbf{D}_{ii} = \sum_j (\mathbf{A}_{\text{self}})_{ij}$ is the degree matrix, and $\hat{\mathbf{A}} = \mathbf{D}^{-\frac{1}{2}}\mathbf{A}_{\text{self}}\mathbf{D}^{-\frac{1}{2}}$. The trainable weights at layer l are $\mathbf{W}^{(l)}$. The final output is obtained by applying a softmax to the last hidden layer $\mathbf{Z} = \text{softmax}\Big(\hat{\mathbf{A}}\mathbf{H}^{(K)}\mathbf{W}^{(K)}\Big)$, with $\mathbf{Z} \in \mathbb{R}^{N \times c}$ giving class probability distributions. Row $\mathbf{Z}_v$ is the distribution for node v, and the predicted class is $\hat{y}_v = \arg\max \mathbf{Z}_v$.

2.2 GNN EXPLANATIONS

GNN explanation methods (Yuan et al., 2022; Longa et al., 2025) reveal the structural and feature-based evidence that plays a key role in predictions. We categorize them into two paradigms:

Factual explanations identify subgraphs or features *supporting* the original prediction. For a target node v, an explanation subgraph $G_v \subseteq G$ satisfies $f(G_v, \mathbf{X_v}) = f(G, \mathbf{X_v})$, where f is the GNN model and $\mathbf{X_v}$ denotes the features of v. The GNNExplainer method (Ying et al., 2019) optimizes G_v to maximize mutual information with the prediction.

Counterfactual explanations identify *minimal perturbations* $\Delta\mathbf{A}$ to alter target node v's prediction $f(\mathbf{A}, \mathbf{X}, v) \neq f(\mathbf{A} \odot \Delta\mathbf{A}, \mathbf{X}, v), \quad \text{s.t.} \quad \|\Delta\mathbf{A}\|_0 \leq \kappa$, where κ is a perturbation budget.

2.3 ADVERSARIAL ATTACKS ON GNNS

Adversarial attacks deliberately perturb graphs (including edge-based and feature-based perturbation) to mislead predictions. Key categories include i) evasion and ii) poisoning attacks.

Evasion attacks modify the graph *during inference* without retraining. For target node v, edge-based attackers solve $\max_{\Delta\mathbf{A}} \mathcal{L}(f(\mathbf{A} \odot \Delta\mathbf{A}, \mathbf{X}, v), y_v) \quad \text{s.t.} \quad \|\Delta\mathbf{A}\|_0 \leq \kappa$, where $\mathcal{L}$ is the loss function which quantifies prediction error.

Poisoning attacks corrupt the *training graph* to degrade retrained models. For target node v, edge-based attackers optimize $\max_{\Delta\mathbf{A}} \mathcal{L}(f_{\theta^*}(\mathbf{A}, \mathbf{X}, v), y_v) \quad \text{s.t.} \quad \theta^* = \arg\min_\theta \mathcal{L}(f_\theta(\mathbf{A} \odot \Delta\mathbf{A}, \mathbf{X})), \quad \|\Delta\mathbf{A}\|_0 \leq \kappa$.

Table 1 summarizes GNN explanations and adversarial attacks according to edge-based perturbation methods by their core characteristics.

Key Insight. While counterfactual explanations have historically emphasized E^- to reveal model fragility, adversarial attacks often exploit E^+ by introducing new connections. More importantly, the attack literature has developed efficient methods to select which edges to add/delete under small perturbation budgets (e.g., $\kappa = 1, \ldots, 5$), despite the combinatorially large number of possible additions in graphs, making naïve counterfactual search impractical. This potential synergy between counterfactual reasoning and attack strategies motivates our problem formulation (§2.4) and the unified framework we propose in (§4).

Table 1: Comparison of GNN explanation and attack paradigms. We use E^- to denote edge deletions (removing existing edges) and E^+ to denote edge additions (introducing new edges).

Category	Goal	Primary Operation	Example
Factual Expl.	Explain prediction	Identify key subgraph	GNNExplainer (Ying et al., 2019)
Counterfactual Expl.	Alter prediction	Mainly E^- (edge deletions), though some recent work includes E^+	CF-GNNExplainer (Lucic et al., 2022)
Evasion Attack	Misclassify node	E^+/E^- in inference, often E^+ dominant	TDGIA (Zou et al., 2021)
Poisoning Attack	Degrade model	E^+/E^- in training, often E^+ dominant	Nettack (Zügner et al., 2018)

2.4 Problem Formulation

Given a graph $G = (V, E, \mathbf{X})$ with adjacency matrix $\mathbf{A}$ and node features $\mathbf{X}$, and a pre-trained GNN classifier f, our goal for a target node $v \in V$ is to find a small set of edge perturbations $\Delta\mathbf{E} = \Delta\mathbf{E}^+ \cup \Delta\mathbf{E}^-$, corresponding to additions ($\Delta\mathbf{E}^+$) and deletions ($\Delta\mathbf{E}^-$), such that the prediction for v flips while the resulting counterfactual graph remains *interpretable* and *plausible*. This problem combines two perspectives: from the attack literature, where efficient methods have been developed to select high-impact edge additions under small budgets, and from counterfactual explanations, where minimal and semantically meaningful deletions expose decision-supporting edges. We formalize this hybrid objective in §4.

3 A Dual Approach of Explanations and Attacks for GNNs

We develop a theoretical framework that links targeted structural evasion attacks on graph neural networks with instance-level counterfactual explanation subgraphs. The core objective is to formalize when and why adversarial perturbations can serve as building blocks for counterfactual explanations. To this end, we introduce a hypothesis to capture the relationship between the attack subgraph and the counterfactual explanation of a target node. **More importantly, we provide empirical support for this hypothesis in Appendix A.11**. To the best of our knowledge, the hypothesis and evidence are presented for the first time to formally connect adversarial attacks and interpretability in graph learning.

To compare explanation and attack subgraphs, we consider two forms of graph similarity: i) structural similarity (Doan et al., 2021): overlap in nodes or edges, measurable via graph edit distance, and maximum common subgraph metrics. ii) semantic similarity (Bai et al., 2020): closeness in learned graph-level embeddings, indicating similar functional or predictive roles even if the structures differ.

Hypothesis 1 states that the added edges in a successful evasion attack overlap with the most influential edges in a pre-attack counterfactual explanation subgraph.

Hypothesis 1. *For a target node v, let $\Delta G(E^+)$ denote the set of added edges in an evasion attack that flips the prediction of f, and let $CFEx(G)$ denote the pre-attack counterfactual explanation subgraph of the graph G. Then, there exists a high graph similarity between $\Delta G(E^+)$ and $CFEx(G)$. The proof is provided in the Appendix A.11.1.*

Building on the hypothesis, in Appendix A.11, we also present two propositions and two corollaries that formalize when attack-based additions outperform deletions in flipping GNN predictions. These results characterize conditions under which deletions provably fail, yet targeted additions succeed, focusing on the functional advantage of attack-informed counterfactuals.

4 ATEX-CF: Methodology for Counterfactual Generation

Our objective is to design a counterfactual explainer that simultaneously incorporates **edge addition** (E^+) and **edge deletion** (E^-), combining GNN adversarial attacks with counterfactual explanation concepts. This explainer should generate high-impact perturbations while maintaining interpretability and realism. In particular, we jointly optimize three core objectives: **Impact** — efficacy in

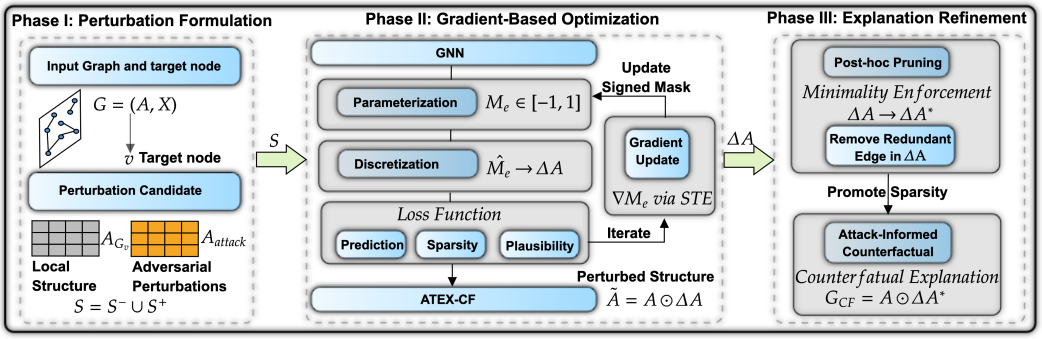

Figure 2: An overview of the ATEX-CF framework.

altering model predictions; **Sparsity** — minimal edits for interpretability; **Plausibility** — semantic validity of graph modifications. Figure 2 illustrates the end-to-end architecture of our ATEX-CF framework, which unifies adversarial edge perturbations with counterfactual explanation generation through a joint optimization of impact, sparsity, and plausibility.

To operationalize these objectives, we cast counterfactual generation as an optimization problem over edge perturbations. Given a candidate set $\mathcal{S}$ of feasible edge edits, we search for $\Delta\mathbf{A} \in \mathcal{S}$ that flips the prediction of the target node while balancing sparsity and plausibility. This is achieved by defining a composite loss with three components, corresponding to our objectives.

Loss Function. We formulate counterfactual generation as

$$\min_{\Delta\mathbf{A}\in\mathcal{S}} \mathcal{L}(\Delta\mathbf{A}) = \lambda_1\mathcal{L}_{pred}(\Delta\mathbf{A}) + \lambda_2\mathcal{L}_{dist}(\Delta\mathbf{A}) + \lambda_3\mathcal{L}_{plau}(\Delta\mathbf{A}), \tag{1}$$

where $\mathcal{S}$ is the candidate search space. Here $\mathcal{L}_{pred}$ enforces label flipping, $\mathcal{L}_{dist}$ penalizes the number of edge edits, and $\mathcal{L}_{plau}$ enforces plausibility constraints. Weights $\lambda_i \geq 0$ balance these terms. Next, we will define each loss function.

Prediction Loss. We denote by $f(\mathbf{A}_v, \mathbf{X}_v; \mathbf{W})$ the prediction for node v under the original adjacency $\mathbf{A}_v$, and by $g(\mathbf{A}_v, \mathbf{X}_v, \mathbf{W}; \Delta\mathbf{A})$ the prediction under a perturbed adjacency $\mathbf{A}_v \odot \Delta\mathbf{A}$. Both share the same weights $\mathbf{W}$; the difference lies only in the perturbation $\Delta\mathbf{A}$.

To encourage prediction flips, we define the loss as

$$\mathcal{L}_{pred}(\Delta\mathbf{A}) = -\mathbb{I}[f(\mathbf{A}_v, \mathbf{X}_v; \mathbf{W}) = f(\mathbf{A}_v \odot \Delta\mathbf{A}, \mathbf{X}_v; \mathbf{W})] \cdot \mathcal{L}_{\text{NLL}}(f(\mathbf{A}_v, \mathbf{X}_v; \mathbf{W}),\, g(\mathbf{A}_v, \mathbf{X}_v, \mathbf{W}; \Delta\mathbf{A}))\,. \tag{2}$$

The indicator ensures that the loss is active only when the perturbed graph yields the same prediction as the original. In that case, the negative log-likelihood term penalizes the perturbed prediction, pushing it away from the original class. Once a flip occurs, the loss becomes zero. Although this objective is non-differentiable due to the discrete nature of the indicator function, we employ the straight-through estimator (STE) to enable gradient-based optimization, as detailed in §4.2.

Sparsity Loss. To encourage concise and interpretable modifications, we impose a sparsity penalty on the number of structural edits. Specifically, we minimize the ℓ_0 norm of the adjacency change $\Delta\mathbf{A} = \Delta\mathbf{E}^+ \cup \Delta\mathbf{E}^-$, where $\Delta\mathbf{E}^+$ and $\Delta\mathbf{E}^-$ denote the sets of added and removed edges, respectively. $\mathcal{L}_{dist}(\Delta\mathbf{A}) = \|\Delta\mathbf{A}\|_0$.

The objective $\mathcal{L}_{dist}(\Delta\mathbf{A})$ measures the total number of edits. By requiring $\|\Delta\mathbf{A}\|_0$ to be small, we keep the modified graph close to the original, curb unnecessary complexity, and reduce overfitting.

Plausibility Loss. When generating counterfactual graphs by adding/removing edges, we must control the plausibility of the produced structure. For example, in a citation graph, an old article cannot cite a more recent article. The plausibility penalty discourages unnatural degree/motif changes: $\mathcal{L}_{plau}(\Delta\mathbf{A}) = \mathcal{C}(\Delta\mathbf{A}) = \alpha_{deg} \cdot \text{DegAnom}(\Delta\mathbf{A}) + \alpha_{motif} \cdot \text{MotifViol}(\Delta\mathbf{A})$. We tune α_{deg} and α_{motif} to enforce realism; larger α_{deg} avoids implausible degree jumps, larger α_{motif} avoids implausible clustering jumps.

$$\mathrm{DegAnom}(\Delta\mathbf{A}) = \sum_{v_i \in V_{\mathrm{sub}}} \frac{\left|\deg_{\tilde{\mathbf{A}}_{v_i}}(v_i) - \deg_{\mathbf{A}_{v_i}}(v_i)\right|}{1 + \deg_{\mathbf{A}_{v_i}}(v_i)}, \tag{3}$$

$$\mathrm{MotifViol}(\Delta\mathbf{A}) = \sum_{v_i \in V_{\mathrm{sub}}} |c_{\tilde{\mathbf{A}}_{v_i}}(v_i) - c_{\mathbf{A}_{v_i}}(v_i)|. \tag{4}$$

$\mathrm{DegAnom}$ penalizes large relative changes in node degree to prevent structural anomalies, where $\deg_{\mathbf{A}}(v)$ and $\deg_{\tilde{\mathbf{A}}}(v)$ are degrees of node v before and after modification. $\mathrm{MotifViol}$ penalizes drastic changes in local motifs, measured via clustering coefficients $c_{\mathbf{A}_v}(v)$ and $c_{\tilde{\mathbf{A}}_v}(v)$.

4.1 Candidate Selection

As a key aspect in ATEX-CF, we constrain the search space of possible perturbations $\Delta\mathbf{A}$ to a pre-selected candidate set $\mathcal{S}$. This tractable set is constructed through a dual mechanism that incorporates both **local neighborhood structures** and **non-local, attack-informed candidates**, balancing interpretability with the ability to discover impactful counterfactuals.

Edge Deletion Candidates ($\mathcal{S}^-$): We follow the principle of *actionability* and *plausibility* (Wachter et al., 2017); counterfactual explanations should suggest meaningful changes within an entity's sphere of influence (e.g., local graph neighborhood), rather than involving arbitrary, distant entities. As a result, candidate edges for removal are restricted to the existing edges within the $(l+1)$-hop neighborhood $\mathcal{N}^{l+1}(v)$ of the target node v, i.e., $\mathcal{S}^- = \{e \mid e \in E, e \in \mathcal{N}^{l+1}(v)\}$.

Edge Addition Candidates ($\mathcal{S}^+$): To overcome the limitation of deletion-only approaches and incorporate insights from adversarial attacks, our key innovation is to draw candidate edges for addition from adversarial attack subgraphs. Specifically, we employ the latest GOttack method (Alom et al., 2025) to generate a set of candidate edges ΔA_{attack} for the target node v. GOttack identifies influential nodes for edge addition by learning the **graph orbit characteristics** of nodes that, when connected to v, maximally increase the probability of misclassification. An orbit in graph theory represents the role of a node within its local substructure (e.g., a central node in a star graph). The underlying Hypothesis 1 of GOttack, validated by our experiments in Table 14, is that nodes occupying similar structural roles (orbits) often have similar predictive influences on the target node. Therefore, edges suggested by GOttack (connecting v to nodes in specific, influential orbits) are both highly impactful and structurally coherent.

Final Candidate Set and Local Graph Formation: The complete candidate set is the union $\mathcal{S} = \mathcal{S}^- \cup \mathcal{S}^+$. The adjacency matrix $\mathbf{A}_v$ for the local subgraph used in subsequent optimization (Eq. 2) is then formed by combining the original $(\ell+1)$-hop neighborhood structure of v and the adversarial perturbation candidates:

$$\mathbf{A}_v = \underbrace{\mathbf{A}_{G_v}}_{\text{local structure}} + \underbrace{\Delta\mathbf{A}_{\text{attack}}}_{\text{adversarial perturbations}} \tag{5}$$

This formulation provides a focused and principled search space $\mathcal{S}$ that is crucial for the efficiency and effectiveness of our counterfactual search algorithm. We use the $(l+1)$-hop neighborhood because an l-layer GCN aggregates information from nodes up to l hops away; including the $(l+1)$-hop ensures that all nodes and edges within the target's effective receptive field—including those that can indirectly influence its representation—are considered as candidates.

4.2 Signed-Mask Perturbation and Forward Discretization

After candidate edges are selected, the challenge is to optimize over the discrete choices of additions and deletions. Since direct optimization of binary graph structures is non-differentiable, we employ a continuous signed mask relaxation. In the forward pass, the mask is discretized into $\{-1, 0, +1\}$ to yield concrete perturbations, while in backpropagation, the straight-through estimator treats this step as identity, allowing gradients to propagate through discrete edge decisions. This process is carried out as follows.

Each candidate edge $e \in \mathcal{S}$ (where $\mathcal{S}$ is the candidate set defined in §4.1) is associated with a continuous signed parameter $M_e \in [-1, 1]$. This parameter encodes both the directionality and the

magnitude of the proposed modification; a signed mask variable M_e encodes perturbations, with $M_e > 0$ denoting an edge addition ($e \in \Delta\mathbf{E}^+$), $M_e < 0$ denoting an edge deletion ($e \in \Delta\mathbf{E}^-$), and $M_e \approx 0$ no modification. Here, the sign of M_e indicates the type of operation (addition or deletion), while the magnitude $|M_e|$ reflects the proposed strength or importance of the perturbation. This continuous representation facilitates gradient-based learning.

During the forward pass, we discretize these continuous parameters to obtain a binary perturbation matrix. This process involves two steps: thresholding and sparsity enforcement. First, we apply thresholding to convert M_e into a ternary value. The discretized mask is obtained by thresholding, $\widehat{M}_e = +1$ if $M_e > \tau^+$, $\widehat{M}_e = -1$ if $M_e < -\tau^-$, and $\widehat{M}_e = 0$ otherwise.

where τ^+ and τ^- are positive thresholds that control the sensitivity for edge addition and deletion, respectively. Typically, we set $\tau^+ = \tau^- = 0.5$ to ensure symmetry.

Next, to enforce the perturbation budget constraint $\|\Delta\mathbf{A}\|_0 \leq \kappa$, we retain only the κ edges with the largest magnitudes $|M_e|$ and assign their discretized values $\widehat{M}_e \in \{-1, 0, +1\}$ to the corresponding entries in the adjacency matrix. The perturbation matrix is defined as $\Delta\mathbf{A}_{i,j} = \widehat{M}_e$ if edge (i, j) is among the top-κ candidates ranked by $|M_e|$, and 0 otherwise. This ensures that at most κ edges are modified, producing sparse and interpretable counterfactuals.

Algorithm 1: ATEX-CF: Counterfactual Generator

Require: Graph $G = (\mathbf{A}, X)$, model f, target node v, candidate set $\mathcal{S}$

Initialize mask $M_e \leftarrow \mathbf{0}$ over $\mathcal{S}$
for $t = 1$ to $T_{\max}$ **do**
  $\widehat{M}_e \leftarrow \textsc{Threshold}(M_e, \tau^+, \tau^-)$ ▷ Discretize
  $\Delta\mathbf{A} \leftarrow \textsc{Top-}\kappa(|M_e|)$ ▷ Sparsify
  Evaluate $\mathcal{L}(M_e)$ on $\mathbf{A} \odot \Delta\mathbf{A}$
  $M \leftarrow M - \eta\nabla_M\mathcal{L}(M)$ ▷ Update via STE
  if flipped(v) **and** $\|\Delta\mathbf{A}\|_0$ stable **then**
    break
  end if
end for
return $\textsc{Prune}(\Delta\mathbf{A}, G, f, v)$ ▷ See Alg. 2

The resulting perturbed adjacency matrix is then computed as $\widetilde{\mathbf{A}} = \mathbf{A} \odot \Delta\mathbf{A}$, where the operator $\odot$ applies the signed edge modifications encoded in $\widehat{M}_e \in \{-1, 0, +1\}$. To maintain differentiability through this discretization step, we employ the straight-through gradient estimator (STE) during backpropagation $\frac{\partial \widehat{M}_e}{\partial M_e} \approx 1$.

This approximation allows gradients to flow directly through the binarization operation, treating the discretization as if it were an identity function in the backward pass (Bengio et al., 2013). Consequently, the continuous parameters M_e can be updated using gradient descent, even though the forward pass involves non-differentiable operations. This approach is widely used in training binary neural networks and has been shown to be effective in practice.

Minimality-Aware Post-Hoc Pruning While the training loss promotes sparsity and plausibility in expectation, the discrete relaxation can leave redundant edges active in $\Delta\mathbf{A}$. This occurs mostly due to noisy or approximate gradient updates that over-compensate. To enforce the minimality of counterfactual explanations, we adopt a simple yet effective greedy algorithm (Algorithm 2 in Appendix A.4). Edges in the candidate set are ranked by their importance score $\psi_e \propto |\partial\mathcal{L}/\partial M_e|$ (approximated gradient magnitude). The algorithm then iteratively removes the least important edge, checking if the prediction flip persists. This continues until no more edges can be removed without reverting the prediction, and it attains final perturbation $\Delta\mathbf{A}^*$. The complete ATEX-CF framework is given in Algorithm 1.

5 Experiments

5.1 Experimental Setup

Datasets. We evaluate ATEX-CF on both synthetic and real-world benchmarks. Synthetic datasets include **BA-SHAPES** and **TREE-CYCLES** (Ying et al., 2019), widely used in GNN explainability, and the **Loan-Decision** social graph (Ma et al., 2025). For real-world evaluation, we use the **Cora** citation network (Sen et al., 2008) and the large-scale **ogbn-arxiv** dataset from OGB (Hu et al., 2020). Dataset statistics are summarized in Table 2.

Table 2: Dataset statistics.

Dataset	Homophily Ratio	#Nodes	#Edges	#Features	#Classes	Type
BA-SHAPES (Ying et al., 2019)	0.80	700	3958	–	4	Synthetic
TREE-CYCLES (Ying et al., 2019)	0.90	871	1,940	–	2	Synthetic
Loan-Decision (Ma et al., 2025)	0.47	1000	3950	2	2	Synthetic
Cora (Sen et al., 2008)	0.81	2,708	5,429	1,433	7	Real
Ogbn-Arxiv (Hu et al., 2020)	0.66	169,343	1,166,243	128	40	Real

Table 3: Meta Results. Average ranks (↓) across five datasets (lower is better). Ranks are computed per metric per dataset (best=1; ties get the same rank), then averaged across datasets equally. "Wins" counts how many times a method achieved rank one across all metric–dataset cells (5 datasets $\times$5 metrics = 25 cells, ties allowed).

Method	Misclass.	Fidelity	ΔE	Plausibility	Time (sec)	Overall Avg.	Wins
CF-GNNExplainer	3.6	4.0	2.0	2.0	6.0	3.52	0
GNNExplainer	4.0	5.0	3.6	3.6	2.6	3.76	0
PGExplainer	5.0	5.8	3.4	3.8	**1.0**	3.80	5
Nettack	2.8	2.2	5.0	5.2	3.8	3.80	2
GOttack	4.0	3.6	5.0	5.4	2.4	4.08	0
ATEX-CF (ours)	**1.0**	**1.4**	**1.0**	**1.0**	5.0	**1.88**	**18**

GNNs. We evaluate our approach on three standard GNN architectures: **GCN** (Kipf & Welling, 2017), **GAT** (Velickovic et al., 2018), and **Graph Transformer** (Shi et al., 2021).

Baselines: We compare our method against a comprehensive set of baseline approaches, which we categorize into two groups. The first group comprises **explanation-based baselines**: **CF-GNNExplainer** (Lucic et al., 2022), a counterfactual method that optimizes for edge deletions using a perturbation mask; **GNNExplainer** (Ying et al., 2019), a factual explainer adapted for counterfactual analysis by removing edges in descending order of importance until prediction flips; and **PGExplainer** (Luo et al., 2020), another factual method adapted similarly to GNNExplainer. The second group consists of **attack-based baselines** repurposed for counterfactual generation: **Nettack** (Zügner et al., 2018), a white-box adversarial attack method adapted by using its edge perturbation capability such that the target class is different from the original prediction; and **GOttack** (Alom et al., 2025), a recent adversarial method that leverages graph orbital theory to identify critical nodes for edge additions, making it naturally suited for generating addition-based counterfactuals. For fair comparison, all methods are constrained to a default perturbation budget (i.e., maximum possible number of edge flips) of $\kappa = 5$ edges. We vary κ for ablation study in Figure 3. Explanation-based methods (CF-GNNExplainer, GNNExplainer, PGExplainer) are restricted to edge deletions only, while attack-based methods (Nettack, GOttack) and our ATEX-CF can use both edge additions and deletions within the same budget.

In our experiments, we set the random seed (102, 103, 104) for reproducibility. For the attack model, we employed evasion attacks using the GOttack method. For the ATEX-CF, we used a learning rate of 0.001, trained for 200 epochs, and adopted the SGD optimizer to generate counterfactual explanation with a maximal perturbed budget of 5 edges. The default loss weights were configured as follows: $\lambda_1 = 1.5$, $\lambda_2 = 0.5$, $\lambda_3 = 0.5$, $\alpha_{deg} = 1.5$, and $\alpha_{motif} = 1.0$. These hyperparameters were chosen to balance prediction flipping, sparsity, and plausibility in counterfactual generation. **Our code is available at `https://anonymous.4open.science/r/GNN_graph_analysis-D90A/README.md`.**

Evaluation Metrics: We evaluate the performance of counterfactual explainers in misclassification rate, fidelity, explanation size, plausibility, and time costs. Definitions are given in Appendix A.5.

5.2 Results and Analysis

We evaluate ATEX-CF against all baselines under the same budget constraints ($\kappa = \{1, \ldots, 5\}$). Table 3 summarizes average rankings across datasets and metrics. Our method achieves the best overall rank (1.88 vs. 3.52 for the next best) and wins 18/25 metric–dataset combinations, far exceeding competitors. This confirms that ATEX-CF consistently finds more effective counterfactuals.

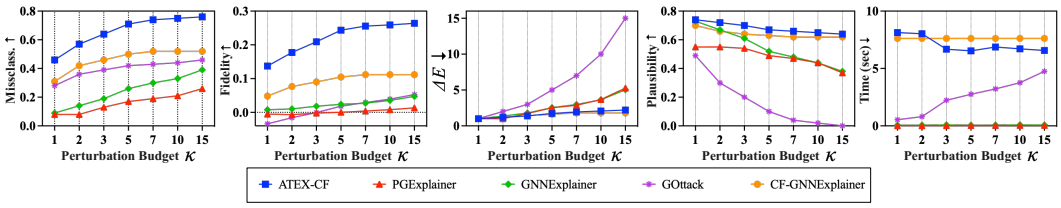

Figure 3: Counterfactual explanations on **Cora** and GCN under varying perturbation budgets κ

In particular, counterfactual explainers (CF-GNNExplainer, GNNExplainer, PGExplainer) are limited to edge deletions, while adversarial attack methods (Nettack, GOttack) and ATEX-CF can also add edges. Crucially, CF-GNNExplainer explicitly seeks minimal edge deletions, and GOttack systematically manipulates graph orbits to induce errors, yet neither matches ATEX-CF on combined effectiveness and realism. **Our empirical results on individual datasets and with other GNNs (GAT and Graph Transformer) are given in Appendix A.6-A.7.**

Figure 3 plots performance vs. perturbation budget on Cora. As the budget grows, all methods improve: ATEX-CF quickly raises misclassification (e.g., from 0.46 at κ=1 to 0.76 at κ=15) far above others, and maintains the highest fidelity and plausibility. Notably, ATEX-CF 's edit size increases only mildly with κ, whereas attack baselines must exhaust all allowed edits ($\Delta E \rightarrow 5$). This trend illustrates that our objective effectively exploits additional budget to find better counterfactuals without excessive edits.

Ablation Study. Table 13 in Appendix A.8 shows the effect of removing each loss. Our findings demonstrate that $\mathcal{L}_{dist}$ enforces concise edits, $\mathcal{L}_{plau}$ preserves semantic plausibility, and their combination in ATEX-CF achieves the best overall balance across all metrics.

Sensitivity Analysis. We next analyze key hyperparameters. **Search depth (l):** Figure 4 in Appendix A.9 shows that l = 2 captures sufficient local structure surrounding the target node for effective counterfactuals. **Hyperparameters ($\alpha_{deg}, \alpha_{motif}$):** Figure 5 in Appendix A.9 demonstrates that ATEX-CF is robust across a range of hyperparameter values (e.g., $\alpha = 0.5$–1.5); while moderate α maximizes fidelity and plausibility together.

Impact of Pruning Strategy. We also evaluate the impact of our candidate-edge pruning strategy (Algorithm 2 in Appendix A.4). As shown in Figure 6, pruning yields more concise explanations by reducing redundant edge edits ($\Delta\mathbf{A} = 1.71 \rightarrow 1.62$), but as intended, its real utility is the reduced runtime (6.12s $\rightarrow$ 3.00s), while preserving predictive accuracy (misclass.=0.71), plausibility (0.76 vs. 0.75), making it an effective and efficient enhancement of our framework.

6 Conclusions

We presented ATEX-CF, a theoretically grounded framework that unifies adversarial attacks and counterfactual explanations for graph neural networks. By incorporating both edge additions and deletions under a constrained budget, ATEX-CF generates explanations that are not only faithful but also informative. Our joint optimization of fidelity, sparsity, and plausibility ensures instance-level counterfactuals that balance interpretability with realism. Experiments on synthetic and real-world benchmarks confirm the effectiveness of this integration, highlighting how adversarial insights can substantially improve the quality of counterfactual explanations, compared with state-of-the-art counterfactual generation and adversarial attack methods.

References

Carlo Abrate and Francesco Bonchi. Counterfactual graphs for explainable classification of brain networks. In *ACM SIGKDD conference on knowledge discovery & data mining*, pp. 2495–2504, 2021.

Carlo Abrate, Giulia Preti, and Francesco Bonchi. Counterfactual explanations for graph classification through the lenses of density. In *World Conference on Explainable Artificial Intelligence*, pp. 324–348. Springer, 2023.

Zulfikar Alom, Tran Gia Bao Ngo, Murat Kantarcioglu, and Cuneyt Gurcan Akcora. Gottack: Universal adversarial attacks on graph neural networks via graph orbits learning. In *International Conference on Learning Representations (ICLR)*, 2025.

Yunsheng Bai, Hao Ding, Ken Gu, Yizhou Sun, and Wei Wang. Learning-based efficient graph similarity computation via multi-scale convolutional set matching. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 3219–3226, 2020.

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. *Advances in neural information processing systems (NeurIPS)*, 34:5644–5655, 2021.

Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.

Christian Borgs, Michael Brautbar, Jennifer Chayes, and Brendan Lucier. Maximizing social influence in nearly optimal time. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pp. 946–957. SIAM, 2014.

Ruichu Cai, Yuxuan Zhu, Xuexin Chen, Yuan Fang, Min Wu, Jie Qiao, and Zhifeng Hao. On the probability of necessity and sufficiency of explaining graph neural networks: A lower bound optimization approach. *Neural Networks*, 184:107065, 2025.

Dibaloke Chanda, Saba Heidari Gheshlaghi, and Nasim Yahya Soltani. Explainability-based adversarial attack on graphs through edge perturbation. *Knowl. Based Syst.*, 310:112895, 2025.

Heng Chang, Yu Rong, Tingyang Xu, Wenbing Huang, Honglei Zhang, Peng Cui, Wenwu Zhu, and Junzhou Huang. A restricted black-box adversarial framework towards attacking graph embedding models. In *AAAI conference on artificial intelligence*, volume 34, pp. 3389–3396, 2020.

Jinyin Chen, Yangyang Wu, Xuanheng Xu, Yixian Chen, Haibin Zheng, and Qi Xuan. Fast gradient attack on network embedding. *arXiv preprint arXiv:1809.02797*, 2018.

Tingyang Chen, Dazhuo Qiu, Yinghui Wu, Arijit Khan, Xiangyu Ke, and Yunjun Gao. View-based explanations for graph neural networks. *Proc. ACM Manag. Data*, 2(1):40:1–40:27, 2024.

Zhaoliang Chen, Zhihao Wu, Ylli Sadikaj, Claudia Plant, Hong-Ning Dai, Shiping Wang, Yiu-Ming Cheung, and Wenzhong Guo. Adedgedrop: Adversarial edge dropping for robust graph neural networks. *IEEE Trans. Knowl. Data Eng.*, 37(9):4948–4961, 2025.

Chirag Chhablani, Sarthak Jain, Akshay Channesh, Ian A Kash, and Sourav Medya. Game-theoretic counterfactual explanation for graph neural networks. In *ACM Web Conference*, pp. 503–514, 2024.

Susanne Dandl, Christoph Molnar, Martin Binder, and Bernd Bischl. Multi-objective counterfactual explanations. In *International conference on parallel problem solving from nature*, pp. 448–469. Springer, 2020.

Khoa D Doan, Saurav Manchanda, Suchismit Mahapatra, and Chandan K Reddy. Interpretable graph similarity computation via differentiable optimal alignment of node embeddings. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, pp. 665–674, 2021.

Wenqi Fan, Han Xu, Wei Jin, Xiaorui Liu, Xianfeng Tang, Suhang Wang, Qing Li, Jiliang Tang, Jianping Wang, and Charu Aggarwal. Jointly attacking graph neural network and its explanations. In *IEEE International Conference on Data Engineering (ICDE)*, pp. 654–667, 2023.

Dimitri Galli, Andrea Venturi, Isabella Marasco, and Mirco Marchetti. Evaluating explainability of graph neural networks for network intrusion detection with structural attacks. In *SERICS*, volume 3962, 2025.

Simon Geisler, Tobias Schmidt, Hakan Şirin, Daniel Zügner, Aleksandar Bojchevski, and Stephan Günnemann. Robustness of graph neural networks at scale. *Advances in Neural Information Processing Systems (NeurIPS)*, 34:7637–7649, 2021.

Kangjia He, Li Liu, Youmin Zhang, Ye Wang, Qun Liu, and Guoyin Wang. Learning counterfactual explanation of graph neural networks via generative flow network. *IEEE Transactions on Artificial Intelligence*, 5(9):4607–4619, 2024.

Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: Datasets for machine learning on graphs. *Advances in neural information processing systems*, 33:22118–22133, 2020.

Zexi Huang, Mert Kosan, Sourav Medya, Sayan Ranu, and Ambuj Singh. Global counterfactual explainer for graph neural networks. In *ACM international conference on web search and data mining (WSDM)*, pp. 141–149, 2023.

Arijit Khan and Ehsan Bonabi Mobaraki. Interpretability methods for graph neural networks. In *IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pp. 1–4, 2023.

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

Mert Kosan, Zexi Huang, Sourav Medya, Sayan Ranu, and Ambuj Singh. Gcfexplainer: Global counterfactual explainer for graph neural networks. *ACM Transactions on Intelligent Systems and Technology*, 2024.

Andreas Krause and Carlos Guestrin. Near-optimal observation selection using submodular functions. In *AAAI*, volume 7, pp. 1650–1654, 2007.

Jiate Li, Meng Pang, Yun Dong, Jinyuan Jia, and Binghui Wang. Graph neural network explanations are fragile. In *International Conference on Machine Learning (ICML)*, 2024.

Jiate Li, Meng Pang, Yun Dong, Jinyuan Jia, and Binghui Wang. Provably robust explainable graph neural networks against graph perturbation attacks. In *International Conference on Learning Representations (ICLR)*, 2025.

Jintang Li, Tao Xie, Liang Chen, Fenfang Xie, Xiangnan He, and Zibin Zheng. Adversarial attack on large scale graph. *IEEE Transactions on Knowledge and Data Engineering*, 35(1):82–95, 2021.

Yifei Liu, Chao Chen, Yazheng Liu, Xi Zhang, and Sihong Xie. Multi-objective explanations of gnn predictions. In *2021 IEEE International Conference on Data Mining (ICDM)*, pp. 409–418. IEEE, 2021.

Antonio Longa, Steve Azzolin, Gabriele Santin, Giulia Cencetti, Pietro Lio, Bruno Lepri, and Andrea Passerini. Explaining the explainers in graph neural networks: a comparative study. *ACM Comput. Surv.*, 57(5):120:1–120:37, 2025.

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

Kirill Lukyanov, Georgii Sazonov, Serafim Boyarsky, and Ilya Makarov. Robustness questions the interpretability of graph neural networks: what to do? *CoRR*, abs/2505.02566, 2025.

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 (NeurIPS)*, 33:19620–19631, 2020.

Jiali Ma, Ichigaku Takigawa, and Akihiro Yamamoto. C2explainer: Customizable mask-based counterfactual explanation for graph neural networks. In *ACM Conference on Fairness, Accountability, and Transparency (FAccT)*, pp. 137–149, 2025.

Jiaqi Ma, Shuangrui Ding, and Qiaozhu Mei. Towards more practical adversarial attacks on graph neural networks. *Advances in neural information processing systems (NeurIPS)*, 33:4756–4766, 2020.

Jing Ma, Ruocheng Guo, Saumitra Mishra, Aidong Zhang, and Jundong Li. Clear: Generative counterfactual explanations on graphs. *Advances in neural information processing systems (NeurIPS)*, 35:25895–25907, 2022.

Mario Alfonso Prado-Romero, Bardh Prenkaj, Giovanni Stilo, and Fosca Giannotti. A survey on graph counterfactual explanations: Definitions, methods, evaluation, and research challenges. *ACM Comput. Surv.*, 56(7):171:1–171:37, 2024.

Dazhuo Qiu, Mengying Wang, Arijit Khan, and Yinghui Wu. Generating robust counterfactual witnesses for graph neural networks. In *IEEE International Conference on Data Engineering (ICDE)*, pp. 3351–3363, 2024.

Dazhuo Qiu, Jinwen Chen, Arijit Khan, Yan Zhao, and Francesco Bonchi. Finding counterfactual evidences for node classification. In *ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2025.

Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. Collective classification in network data. *AI magazine*, 29(3):93–93, 2008.

Yunsheng Shi, Zhengjie Huang, Shikun Feng, Hui Zhong, Wenjing Wang, and Yu Sun. Masked label prediction: Unified message passing model for semi-supervised classification. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1548–1554, 2021.

Lichao Sun, Yingtong Dou, Carl Yang, Kai Zhang, Ji Wang, Philip S. Yu, Lifang He, and Bo Li. Adversarial attack and defense on graph data: A survey. *IEEE Trans. Knowl. Data Eng.*, 35(8): 7693–7711, 2023.

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 *WWW*, pp. 1018–1027, 2022a.

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 *ACM web conference*, pp. 1018–1027, 2022b.

Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. 2018.

Samidha Verma, Burouj Armgaan, Sourav Medya, and Sayan Ranu. Induce: Inductive counterfactual explanations for graph neural networks. *Transactions on Machine Learning Research*, 2024.

Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harv. JL & Tech.*, 31:841, 2017.

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

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*, 45(5): 5782–5799, 2022.

Yi Zhang, Yuying Zhao, Zhaoqing Li, Xueqi Cheng, Yu Wang, Olivera Kotevska, Philip S. Yu, and Tyler Derr. A survey on privacy in graph neural networks: Attacks, preservation, and applications. *IEEE Trans. Knowl. Data Eng.*, 36(12):7497–7515, 2024.

Zhiqiang Zhong, Anastasia Barkova, and Davide Mottin. Knowledge-augmented graph machine learning for drug discovery: A survey. *ACM Comput. Surv.*, 57(12):302:1–302:38, 2025.

Guanghui Zhu, Mengyu Chen, Chunfeng Yuan, and Yihua Huang. Simple and efficient partial graph adversarial attack: A new perspective. *IEEE Trans. Knowl. Data Eng.*, 36(8):4245–4259, 2024.

Xu Zou, Qinkai Zheng, Yuxiao Dong, Xinyu Guan, Evgeny Kharlamov, Jialiang Lu, and Jie Tang. Tdgia: Effective injection attacks on graph neural networks. In *ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pp. 2461–2471, 2021.

Daniel Zügner, Amir Akbarnejad, and Stephan Günnemann. Adversarial attacks on neural networks for graph data. In *ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 2847–2856, 2018.

REPRODUCIBILITY STATEMENT

We provide the full implementation of our models and experimental setup to ensure reproducibility. Experimental results are reported as the mean and standard deviation across different random seeds, and the hyperparameters used are detailed in Section 5.1. **Our code and data are available at `https://anonymous.4open.science/r/GNN_graph_analysis-D90A/README.md`.**

A APPENDIX

A.1 LIMITATION

A key limitation of this study is the assumption that edge additions and deletions are equally feasible, which may not hold in domains where graph modifications are inherently constrained. Future work could incorporate domain-specific constraints and node-feature perturbations to enhance the practical relevance of ATEX-CF while preserving its theoretical contributions. The central premise of our approach is that unifying adversarial attack strategies with counterfactual reasoning strengthens both the fidelity and plausibility of explanations. Unlike methods that treat these perspectives independently, ATEX-CF provides a principled integration that balances model sensitivity with explanation realism in a computationally tractable way.

A.2 RELATED WORK

GNN Explanations. Different categories of GNN explanation methods have been developed to offer diverse perspectives and improve the interpretability of GNN models (Khan & Mobaraki, 2023; Yuan et al., 2022). Two main categories of explanations persist: factual and counterfactual. **Counterfactual explanations**, which are the focus of this work, provide explanations by identifying the minimum perturbation or change to the input graph that leads to a different prediction from the model (Bajaj et al., 2021; Huang et al., 2023; Tan et al., 2022b), thereby revealing the most critical structures underlying the decision. Existing methods are predominantly based on edge deletions. For instance, CF-GNNExplainer (Lucic et al., 2022), RCExplainer (Bajaj et al., 2021), GNN-MOExp (Dandl et al., 2020), CF^2 (Tan et al., 2022b), NSEG (Cai et al., 2025), Banzhaf (Chhablani et al., 2024), and CF-GFNExplainer (He et al., 2024) all design deletion-oriented mechanisms, such as gradient-based mask optimization, decision boundary constraints, multi-objective optimization, or probabilistic sampling. These approaches emphasize faithfulness, sparsity, or necessity/sufficiency guarantees, but rely mainly on removing salient substructures.

More recently, several works on node classification have extended counterfactual explanations to include edge additions, or the joint use of both addition and deletion. INDUCE (Verma et al., 2024) treats counterfactual search as a Markov decision process, allowing the model to learn edge modifications (both additions and deletions) that lead to flips. C2Explainer (Ma et al., 2025) further integrates hypergraph representations with straight-through optimization to balance reliability and fidelity, and explicitly models the potential risks of false evidence from edge additions. In the context of graph classification, approaches such as counterfactual graphs (Abrate & Bonchi, 2021), CLEAR (Ma et al., 2022), GCFExplainer (Kosan et al., 2024), and density-based counterfactual graphs (Abrate et al., 2023) adopt generative or global search strategies that combine edge addition and deletion to ensure causally consistent and semantically coherent explanations.

Overall, while edge-deletion-based methods dominate current counterfactual explanation research, the emerging edge-addition or mixed approaches demonstrate that edge addition can serve as a complementary mechanism, especially in cases where deletion-based explanations fail to capture counterfactual reasoning. This motivates our design of hybrid counterfactual explainers that leverage both deletion and addition. Unlike prior counterfactual methods that may include edge additions, our approach is the first to integrate adversarial attack strategies—systematically leveraging their capacity to identify high-impact edge additions—with traditional deletion-based reasoning, thereby unifying two separately studied domains to generate more effective and actionable explanations. Table 4 summarizes important GNN explanation methods, including both factual and counterfactual approaches, along with their explanation type, candidate modification, and target task.

Table 4: Characteristics important GNN explainers including ours. "E , F, N" denote removing/adding edges, node feature modification, removing/adding nodes, respectively. GC and NC denote graph classification and node classification, respectively.

Method	Type	Candidate	Task
GNNExplainer (Ying et al., 2019)	factual/instance-level	E, N	GC/NC
PGExplainer (Luo et al., 2020)	factual/instance-level	E	GC/NC
MOO (Liu et al., 2021)	counterfactual/instance-level	E(-), N(-)	NC
CF-GNNExplainer (Lucic et al., 2022)	counterfactual/instance-level	E(-)	NC
RCExplainer (Bajaj et al., 2021)	counterfactual/instance-level	E(-)	GC/NC
CF^2 (Tan et al., 2022b)	counterfactual/instance-level	E(-), F	GC/NC
INDUCE (Verma et al., 2024)	counterfactual/instance-level	E(+,-)	NC
NSEG (Cai et al., 2025)	counterfactual/instance-level	E(-), F	GC/NC
Banzhaf (Chhablani et al., 2024)	counterfactual/instance-level	E(-)	NC
C2Explainer (Ma et al., 2025)	counterfactual/instance-level	E(+,-), F	GC/NC
ATEX-CF (ours)	counterfactual/instance-level	E(+,-)	NC

GNN Adversarial Attacks. Graph adversarial attacks investigate structural perturbations but from a different perspective: their objective is to reduce model performance rather than to improve interpretability. These attacks can be divided into two main categories: evasion attacks and poisoning attacks (Yuan et al., 2022; Longa et al., 2025). In evasion attacks, the GNN parameters are fixed and the adversary perturbs the test graph to flip predictions without retraining. Examples include targeted edge modifications during inference (Zou et al., 2021; Chang et al., 2020; Ma et al., 2020; Fan et al., 2023). Poisoning attacks, in contrast, manipulate the training data by injecting adversarial samples, forcing the retrained model to internalize the perturbations and degrade performance (Alom et al., 2025; Zügner et al., 2018; Li et al., 2021; Chen et al., 2018; Geisler et al., 2021).

Empirical studies show that adversarial evasion attacks on GNNs — particularly those based on strategically adding edges — exploit data biases and model weaknesses to induce misclassifications, in stark contrast to counterfactual explanations, which predominantly rely on edge deletions. Integrating these attack-inspired edge-addition perturbations into counterfactual frameworks can enrich explanation graphs and forge a novel link between adversarial robustness and interpretability.

Fusing GNN Explanations and Robustness against Attacks. Recent efforts on robust explainable graph neural networks combine explainability with adversarial defense to preserve explanation quality under worst-case perturbations. GNNEF (Li et al., 2024) reveals that perturbation-based explainers (e.g., GNNExplainer, PGExplainer) are highly fragile, as minor structural changes can drastically alter explanations without affecting predictions, and proposes loss- and deduction-based attacks exposing this vulnerability across both graph- and node/edge-level tasks. Fan et al. (2023) develop GEAttack that can attack both a GNN model and its explanations by simultaneously exploiting their vulnerabilities. Chanda et al. (2025) exploit explainability-based strategy to devise adversarial attacks on GNNs. Complementarily, Lukyanov et al. (2025) introduce a benchmark analyzing the interplay between robustness and interpretability under poisoning and evasion attacks, showing that most defenses improve interpretability but with architecture-dependent trade-offs and limitations in existing metrics. Building on these insights, XGNNCert (Li et al., 2025) provides the first certifiable robustness guarantee for graph-level tasks, ensuring stable explanations without sacrificing predictive performance. At the node/edge level, k-RCW (Qiu et al., 2024) proposes robust counterfactual witnesses (RCWs) that remain factual, counterfactual, and resilient to structural disturbances, while GNNNIDS (Galli et al., 2025) introduces an evaluation framework for intrusion detection via structural adversarial attacks, demonstrating that Integrated Gradients produces precise yet exploitable explanations. While these works improve the robustness of explanations, to the best of our knowledge, we are the first to unify adversarial attack techniques such as both edge additions and deletions for better counterfactual explanation generation.

A.3 Summary of notations used in this paper

Table 5 provides a concise summary of the key notations used in this paper, covering graph structure, node features, GNN models, optimization terms, and theoretical concepts.

Table 5: Summary of notations used in this paper

Symbol Group	Description
	Graph Structure
G, V, E	Input graph, node set, edge set
N, m	Number of nodes and edges ($N = \|V\|$, $m = \|E\|$)
$\mathbf{A}, \mathbf{A}_{self}$	Adjacency matrix $\mathbf{A} \in \{0,1\}^{n\times n}$, adjacency matrix with self-loops ($\mathbf{A} + \mathbf{I}_N$)
$\mathbf{D}, \hat{\mathbf{A}}$	Degree matrix, normalized adjacency matrix ($\mathbf{D}^{-\frac{1}{2}}\mathbf{A}_{self}\mathbf{D}^{-\frac{1}{2}}$)
$\widetilde{\mathbf{A}}, \Delta\mathbf{A}$	Perturbed adjacency matrix ($\mathbf{A} \odot \Delta\mathbf{A}$), edge modifications ($\in \{-1,0,1\}^{n\times n}$)
$\Delta\mathbf{E}^+, \Delta\mathbf{E}^-$	Added/deleted edge sets
	Node Features, Neighborhood, & Labels
$\mathcal{N}^l(v)$	l-hop neighborhood of node v
$\mathbf{X}$	Node feature matrix ($\in \mathbb{R}^{n\times d}$)
$v, y_v, \hat{y}_v$	Target node, ground-truth label, predicted label
	GNN Model
$\mathbf{W}^{(l)}, \mathbf{H}^{(l)}$	Weight matrix and hidden representations at GNN layer l
$\mathbf{Z}$	Output logits
$f(\mathbf{A}, \mathbf{X}, v)$	GNN prediction for node v
	Optimization & Loss
$\mathcal{L}(\bullet)$	Loss objective function
$\mathcal{L}_{pred}, \mathcal{L}_{dist}$	Prediction loss (flipping), sparsity loss (minimal edits)
$\mathcal{L}_{plau}, \mathcal{C}(\Delta\mathbf{A})$	Plausibility loss, plausibility penalty
$\|\Delta\mathbf{A}\|_0, \kappa$	Number of changed edges, perturbation budget
$M_e, \widehat{M}_e$	Continuous signed mask ($\in [-1,1]$), discretized mask ($\in \{-1,0,1\}$)
τ^+, τ^-	Positive/negative thresholds for discretization
$\psi_e, \mathcal{S}$	Edge importance score, candidate modification set
$\lambda_1, \lambda_2, \lambda_3$	Loss trade-off weights
$\alpha_{deg}, \alpha_{motif}$	Realism penalty weights
η	Learning rate
	Theoretical Concepts
m_v	Prediction margin
g_e	Gradient influence: $\frac{\partial m_v}{\partial A_e}$
$\mathbf{A}_v, \mathbf{D}_v, \tilde{\mathbf{D}}_v$	Local adjacency, degree matrix, perturbed degree matrix for node v
$c_{\mathbf{A}}(v), c_{\tilde{\mathbf{A}}}(v)$	Clustering coefficient (original/perturbed) for node v

A.4 Minimality-Aware Post-Hoc Pruning: Algorithm 2

Algorithm 2 removes redundant edges left after training by greedily pruning the least important ones while preserving the prediction flip. This yields minimal perturbations $\Delta\mathbf{A}^*$ that enhance conciseness without extra cost. Empirically, pruning reduces edits ($1.71 \rightarrow 1.62$) while maintaining fidelity, plausibility, and runtime (Figure 6).

A.5 Evaluation Metrics

- **Misclassification Rate**: It measures the fraction of predictions flipped by perturbations. Higher values indicate stronger disruption, consistent with the *attack success rate* widely used in GNN adversarial attacks, such as Nettack (Zügner et al., 2018) and GOttack (Alom et al., 2025).

$$\text{Misclassification Rate} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}(\hat{y}_i^{1-m_i} \neq c_i), \tag{6}$$

Algorithm 2 Minimality Pruning

Input: Perturbation $\Delta\mathbf{A}$, graph G, model f, target node v
Output: Minimal perturbation $\Delta\mathbf{A}^*$

1. Initialize: $\Delta\mathbf{A}^* \leftarrow \Delta\mathbf{A}$
2. Rank edges in $\Delta\mathbf{A}^*$ by importance score ψ_e (descending)
3. **for each** edge e_i in ascending order of ψ_e:
   (a) $\Delta\mathbf{A}' \leftarrow \Delta\mathbf{A}^* \setminus \{e_i\}$ (tentatively remove)
   (b) **if** $f(\mathbf{A} \odot \Delta\mathbf{A}', v) \neq f(\mathbf{A}, v)$:
      i. $\Delta\mathbf{A}^* \leftarrow \Delta\mathbf{A}'$ (keep the smaller perturbation)
4. **return** $\Delta\mathbf{A}^*$

where N is the number of evaluated target nodes, $c_i = f(\mathbf{A}, \mathbf{X}, v_i)$ denotes the model-predicted class of target node v_i. $\hat{y}_i^{1-m_i} = f(\tilde{\mathbf{A}}, \mathbf{X}, v_i)$, $\tilde{\mathbf{A}}$ is the perturbed adjacency, m_i is the explanation mask (edges added/removed), $\mathbb{I}$ is the indicator function.

- **Fidelity**: This metric measures the prediction confidence drop on the model's predicted class c_i (Bajaj et al., 2021). Formally:

$$\text{Fidelity} = \frac{1}{N}\sum_{i=1}^{N}\big(f(\mathbf{A}, \mathbf{X}, v_i)_{c_i} - f(\tilde{\mathbf{A}}, \mathbf{X}, v_i)_{c_i}\big), \tag{7}$$

where $f(\mathbf{A}, \mathbf{X}, v)_c$ denotes the softmax probability assigned to class c. Unlike the binary Misclassification Rate, which captures label flips, Fidelity provides a finer-grained sensitivity analysis by quantifying how perturbations reduce the model's confidence in its own prediction.

- **Explanation Size** $\Delta\mathbf{E}$: It represents the average number of structural modifications (including both edge additions and deletions) made per counterfactual explanation, calculated as:

$$\Delta\mathbf{E} = \frac{1}{n}\sum_{i=1}^{n}\Delta\mathbf{E_i} = \frac{1}{n}\sum_{i=1}^{n}(\Delta\mathbf{E_i}^{+} + \Delta\mathbf{E_i}^{-}), \tag{8}$$

We report the average over successful counterfactuals n since $\Delta\mathbf{E}_i$ is well-defined only when a valid counterfactual is generated, ensuring that the metric reflects the true complexity of feasible explanations rather than being diluted by failed cases (Lucic et al., 2022; Tan et al., 2022a). Here, $\Delta\mathbf{E_i}$ represents the set of perturbed edges for node v_i. Smaller values indicate more compact and interpretable explanations.

- **Plausibility**: This evaluates the human-interpretable quality of counterfactual explanations by assessing their realism and coherence with domain knowledge. The plausibility score is averaged across n successful counterfactuals:

$$\text{Plausibility} = \frac{1}{n}\sum_{i=1}^{n} S_{plau}^{(i)}, \quad S_{plau}^{(i)} = 2 \cdot \Big(1 - \frac{1}{1+\exp(-k \cdot L_{plau}^{(i)})}\Big), \tag{9}$$

where $S_{plau}^{(i)} \in (0, 1)$ is the plausibility score for target node v_i, k is a scaling factor (default $k = 1$), and $L_{plau}^{(i)} \in (0, \infty)$ encodes domain-specific constraints quantifying the realism of the counterfactual. Higher values indicate more plausible explanations. In our experiments, $L_{plau}^{(i)}$ is instantiated using the definition in Eq.3 and Eq.4 in §4, ensuring consistency with our evaluation setup. More generally, $L_{plau}^{(i)}$ serves as a flexible placeholder that can incorporate task-specific structural and semantic constraints to assess the realism of counterfactuals in diverse domains.

- **Time Cost**: We record the average running time required in seconds to generate a counterfactual explanation for a single node, providing insights into the computational efficiency of different methods.

A.6 Experimental Results on Individual Datasets

Across all datasets, ATEX-CF flips the most target nodes while using very few edits. On Cora (Table 6), ATEX-CF achieves a misclassification rate of 0.72 with only 1.63 average edge changes,

Table 6: Performance of counterfactual explanations on **Cora** and GCN.

Method	Base GNN	Misclass. ↑	Fidelity ↑	$\Delta\mathbf{E}(\mathbf{E}^+, \mathbf{E}^-)$ ↓	Plausibility ↑	Time (sec) ↓
CF-GNNExplainer	GCN	0.49±0.013	0.1060±0.0034	1.70±0.08 (0.00, 1.70)	0.64±0.008	10.21±2.88
GNNExplainer	GCN	0.22±0.016	0.0197±0.0150	2.58±0.13 (0.00, 2.58)	0.53±0.021	0.44±0.52
PGExplainer	GCN	0.14±0.009	-0.0010±0.0017	2.38±0.03 (0.00, 2.38)	0.53±0.005	**0.04±0.02**
Attack Models						
Nettack	GCN	0.53±0.005	0.1484±0.0057	5.00±0.00 (3.86, 1.14)	0.13±0.005	3.36±0.85
GOttack	GCN	0.53±0.005	0.1466±0.0043	5.00±0.00 (4.70, 0.30)	0.10±0.000	2.24±0.81
ATEX-CF (Ours)	GCN	**0.72±0.008**	**0.2336±0.0003**	**1.63±0.01 (0.90, 0.73)**	**0.75±0.008**	7.26±2.5

Table 7: Performance of counterfactual explanations on **BA-SHAPES** and GCN.

Method	Base GNN	Misclass. ↑	Fidelity ↑	$\Delta\mathbf{E}(\mathbf{E}^+, \mathbf{E}^-)$ ↓	Plausibility ↑	Time (sec) ↓
Explainers						
CF-GNNExplainer	GCN	0.64±0.017	0.3383±0.0079	1.33±0.20 (0.00, 1.33)	0.57±0.012	11.30±3.72
GNNExplainer	GCN	0.65±0.022	0.3055±0.0019	1.83±0.15 (0.00, 1.83)	0.34±0.009	0.81±1.03
PGExplainer	GCN	0.73±0.031	0.3672±0.0015	1.45±0.05 (0.00, 1.45)	0.41±0.075	**0.03±0.01**
Attack Models						
Nettack	GCN	0.64±0.005	0.3526±0.0063	5.00±0.00 (4.08, 0.92)	0.22±0.0036	0.89±0.38
GOttack	GCN	0.63±0.012	0.3399±0.0081	5.00±0.00 (4.30, 0.70)	0.32±0.008	0.73±0.22
ATEX-CF (Ours)	GCN	**0.83±0.009**	**0.4237±0.0118**	**1.24±0.02 (1.21, 0.03)**	**0.71±0.000**	8.96±0.43

compared to only 0.53 for both Nettack and GOttack (each being forced to flip 5 edges). Our fidelity (0.2336) and plausibility (0.75) are also the highest. In contrast, PGExplainer is extremely fast (0.04s) but flips almost no nodes, and attack methods (Nettack/GOttack) flip all 5 edges but yield very low plausibility (≈ 0.1–0.13). A similar pattern holds on BA-Shapes (Table 7) and Tree-Cycles (Table 8), which are motif-based synthetic graphs. On BA-Shapes, ATEX-CF attains 0.83 misclassification with ΔE=1.24 and plausibility 0.71, clearly outperforming others; on Tree-Cycles, it achieves 0.58 misclassification vs. 0.58 for Nettack, but with far higher plausibility (0.64 vs. 0.27) and much smaller edits (ΔE=1.29 vs. 5.00). These synthetic benchmarks have no node features and explicit motif structures, and ATEX-CF reliably discovers the minimal motif changes needed.

On the Loan-Decision social graph (Table 9), ATEX-CF again dominates: 0.68 misclassification (vs. ≤ 0.35 for others) and highest fidelity (0.3658) with only ΔE=1.27. Finally, on the large real ogbn-arxiv network (Table 10), ATEX-CF flips 0.90 fraction of nodes vs. 0.85–0.86 for attacks, yet uses just ΔE=1.20 edges (attacks use 5) and achieves plausibility 0.73 (vs. 0.58–0.66). The ogbn-arxiv dataset is a citation graph of CS papers with 128-dimensional features and 40 classes, confirming ATEX-CF scales to large, feature-rich graphs. In summary, our method consistently finds compact counterfactual edits that flip more predictions than baselines, yielding higher fidelity while preserving realistic graph structure.

A.7 Experimental results with Graph Transformer and GAT

Tables 11 and 12 further demonstrate that ATEX-CF remains consistently superior on both Graph Transformer (Shi et al., 2021) and GAT (Velickovic et al., 2018) backbones. On Graph Transformer (Table 11), our method achieves the highest misclassification rate (0.44) and plausibility (0.50), while also maintaining competitive edit compactness ($\Delta\mathbf{E} = 1.66$). On GAT (Table 12), ATEX-CF shows an even clearer margin, boosting misclassification to 0.47 and plausibility to 0.65, outperforming all baselines by a large gap. These results confirm that our mask optimization generalizes beyond GCNs, remaining stable and effective across different architectures, including both attention-based and transformer-based GNNs.

Moreover, Tables 11 and 12 show that applying CF-GNNExplainer to attention-based models such as GAT and Graph Transformer often results in unstable mask optimization. This instability arises because, unlike GCN where the normalized adjacency enters linearly into the convolution allowing effective gradient flow from the loss to the mask, attention-based architectures compute edge attention coefficients via nonlinear transformations (LeakyReLU, softmax). Any mask applied to edge weights is absorbed and scaled by $\alpha_{ij}(1 - \alpha_{ij}) \ll 1$, leading to vanishing gradient signals and preventing the identification of meaningful counterfactual edges.

In CF-GNNExplainer, the adjacency mask P is treated as a continuous parameter (after a sigmoid), which scales the edge weight or serves as an edge attribute. In attention-based models, these edge

Table 8: Performance of counterfactual explanations on **TREE-CYCLES** and GCN.

Method	Base GNN	Misclass. ↑	Fidelity ↑	$\Delta\mathbf{E}(\mathbf{E}^+, \mathbf{E}^-)$ ↓	Plausibility ↑	Time (sec) ↓
Explainers						
CF-GNNExplainer	GCN	0.49±0.054	0.3437±0.0422	1.95±0.03 (0.00, 1.95)	0.34±0.005	6.16±2.09
GNNExplainer	GCN	0.53±0.085	0.3608±0.0637	2.57±0.31 (0.00, 2.57)	0.26±0.041	0.70±0.93
PGExplainer	GCN	0.41±0.033	0.2733±0.0288	2.52±0.12 (0.00, 2.52)	0.31±0.022	**0.01±0.00**
Attack Models						
Nettack	GCN	0.58±0.022	**0.4508±0.0217**	5.00±0.00 (4.34, 0.66)	0.27±0.099	0.58±0.17
GOttack	GCN	0.18±0.005	0.1083±0.0033	5.00±0.00 (4.91, 0.09)	0.21±0.016	0.41±0.09
ATEX-CF (Ours)	GCN	**0.58±0.009**	0.4052±0.0221	**1.29±0.07 (0.69, 0.60)**	**0.64±0.009**	2.98±1.41

Table 9: Performance of counterfactual explanations on **Loan-Decision** and GCN.

Method	Base GNN	Misclass. ↑	Fidelity ↑	$\Delta\mathbf{E}(\mathbf{E}^+, \mathbf{E}^-)$ ↓	Plausibility ↑	Time (sec) ↓
Explainers						
CF-GNNExplainer	GCN	0.45±0.092	0.2520±0.0490	1.35±0.20 (0.00, 1.35)	0.53±0.038	56.00±7.04
GNNExplainer	GCN	0.16±0.048	0.0438±0.0497	2.56±0.29 (0.00, 2.56)	0.42±0.017	3.33±4.36
PGExplainer	GCN	0.10±0.008	0.0281±0.0105	2.80±0.34 (0.00, 2.80)	0.21±0.024	**0.32±0.04**
Attack Models						
Nettack	GCN	0.34±0.005	0.1685±0.0075	5.00±0.00 (3.01, 1.99)	0.24±0.017	1.16±0.43
GOttack	GCN	0.35±0.005	0.1742±0.0053	5.00±0.00 (4.25, 0.75)	0.15±0.005	0.52±0.16
ATEX-CF (Ours)	GCN	**0.68±0.024**	**0.3658±0.0171**	**1.27±0.02 (0.38, 0.89)**	**0.67±0.026**	20.327±0.58

attributes enter the computation of attention logits z_{ij}:

$$z_{ij} = s_{ij} + b \cdot e_{ij}, \quad e_{ij} = \sigma(P_{ij}), \tag{10}$$

where s_{ij} is a feature-derived score, b is a scalar, and σ is the sigmoid. The normalized attention coefficient is

$$\alpha_{ij} = \frac{\exp(z_{ij})}{\sum_{k\in\mathcal{N}(i)} \exp(z_{ik})}. \tag{11}$$

The gradient of the loss L with respect to P_{ij} is then

$$\frac{\partial L}{\partial P_{ij}} = \frac{\partial L}{\partial \alpha_{ij}} \cdot \underbrace{\frac{\partial \alpha_{ij}}{\partial z_{ij}}}_{\alpha_{ij}(1-\alpha_{ij})} \cdot \underbrace{\frac{\partial z_{ij}}{\partial e_{ij}}}_{b} \cdot \underbrace{\frac{\partial e_{ij}}{\partial P_{ij}}}_{\sigma'(P_{ij})}. \tag{12}$$

The critical term is the Jacobian of the softmax:

$$\frac{\partial \alpha_{ij}}{\partial z_{ij}} = \alpha_{ij}(1 - \alpha_{ij}).$$

When the degree of node i is N and neighbors are similar, $\alpha_{ij} \approx 1/N$, thus $\alpha_{ij}(1-\alpha_{ij}) \approx \mathcal{O}(1/N)$. This means that the mask gradient is strongly diluted by $1/N$, which is further multiplied by the sigmoid derivative $\sigma'(P_{ij})$. As a result, the gradient magnitude quickly vanishes, especially for high-degree nodes. This explains why CF-GNNExplainer struggles to optimize adjacency masks in attention-based models.

In contrast, ATEX-CF uses a *signed, discrete mask* $M_{ij} \in \{-1, 0, +1\}$ combined with a straight-through estimator (STE) for backpropagation:

$$\tilde{A}_{ij} = \tilde{A}_{ij}^{\text{discrete}} + \big(M_{ij}^{\text{cont}} - \text{sg}(M_{ij}^{\text{cont}})\big), \quad \frac{\partial L}{\partial M_{ij}^{\text{cont}}} \approx \frac{\partial L}{\partial \tilde{A}_{ij}}, \tag{13}$$

where $\tilde{A}_{ij}$ is the perturbed adjacency entry for edge (i, j), $\tilde{A}_{ij}^{\text{discrete}}$ is the discrete (binary) adjacency entry, M_{ij}^{cont} is the continuous mask, $\text{sg}(\cdot)$ denotes the stop-gradient operator that blocks gradients, and L is the model loss. This allows gradients to capture the *finite-difference effect* of adding or deleting an edge on the loss, without attenuation from softmax or nonlinearities. As a result, the signed mask optimization remains stable and effective across both GCNs and attention-based models, enabling reliable counterfactual explanations.

Table 10: Performance of counterfactual explanations on **ogbn-arxiv** and GCN.

Method	Base GNN	Misclass. ↑	Fidelity ↑	$\Delta\mathbf{E}(\mathbf{E}^+, \mathbf{E}^-)$ ↓	Plausibility ↑	Time (sec) ↓
Explainers						
CF-GNNExplainer	GCN	0.45±0.033	0.0791±0.0072	1.56±0.06 (0.00, 1.56)	0.66±0.005	7.16±1.71
GNNExplainer	GCN	0.33±0.014	0.0136±0.0040	2.22±0.07 (0.00, 2.22)	0.63±0.008	0.10±0.01
PGExplainer	GCN	0.26±0.012	0.0206±0.0056	2.25±0.13 (0.00, 2.25)	0.59±0.022	**0.10±0.05**
Attack Models						
Nettack	GCN	0.86±0.009	**0.3366±0.0059**	5.00±0.00 (4.38, 0.62)	0.58±0.029	2.14±0.69
GOttack	GCN	0.85±0.022	0.3251±0.0107	5.00±0.00 (5.00, 0.00)	0.62±0.012	0.63±0.08
ATEX-CF (Ours)	GCN	**0.90±0.012**	0.3251±0.0023	**1.20±0.05 (0.92, 0.28)**	**0.73±0.017**	3.35±0.17

Table 11: Performance of counterfactual explanations on **Loan-Decision** and Graph Transformer.

Method	Base GNN	Misclass. ↑	Fidelity ↑	$\Delta\mathbf{E}(\mathbf{E}^+, \mathbf{E}^-)$ ↓	Plausibility ↑	Time (sec) ↓
CF-GNNExplainer	Graph Trans.	–	–	–	–	–
GNNExplainer	Graph Trans.	0.30±0.039	0.2452±0.0513	2.99±0.27 (0.00, 2.00)	0.36±0.012	4.77±3.45
PGExplainer	Graph Trans.	0.39±0.007	0.3187±0.0141	1.66±0.27 (0.00, 1.66)	0.45±0.031	**0.05±0.03**
Nettack	Graph Trans.	0.32±0.004	0.2509±0.0101	5.00±0.00 (4.03, 0.97)	0.22±0.016	1.03±0.48
GOttack	Graph Trans.	0.31±0.006	0.2420±0.0072	5.00±0.00 (4.81, 0.19)	0.30±0.005	0.66±0.21
ATEX-CF (Ours)	Graph Trans.	**0.44±0.035**	**0.3563±0.0120**	**1.66±0.01 (0.85, 0.81)**	**0.50±0.024**	8.07±2.41

A.8 Ablation Study

Table 13 shows the effect of removing each loss on Cora. Removing $\mathcal{L}_{dist}$ reduces misclassification slightly (0.71 → 0.70), leading to larger edit sets (1.62 → 1.66) and lower plausibility (0.75 → 0.71), indicating that edit minimality is compromised. Omitting the plausibility loss ($\mathcal{L}_{plau}$) yields the smallest edit size (1.57), but severely hurts misclassification, dropping the rate to 0.68, as edits no longer respect semantic structure. Removing both losses reduces misclassification and plausibility. These findings demonstrate that $\mathcal{L}_{dist}$ enforces concise edits, $\mathcal{L}_{plau}$ preserves semantic plausibility, and their combination in ATEX-CF achieves the best overall balance across all metrics.

A.9 Sensitivity Analysis

We analyze the key hyperparameters using Cora. **Search depth (l):** Varying the number of hops for local structure surrounding the target node shows diminishing returns beyond local context. Figure 4 depicts that going from $l = 2$ to $l = 3$ yields only marginal improvements in fidelity, edits, and plausibility (e.g., +0.06 fidelity, -0.38 $\Delta\mathbf{E}$, +0.03 plausibility), while increasing computation time and dropping misclassification. This indicates that depth-2 captures sufficient structure for effective counterfactuals. **Hyperparameters ($\alpha_{deg}, \alpha_{motif}$):** We vary the weights of degree-anomaly and motif-anomaly terms in plausibility loss. Figure 5 demonstrates that ATEX-CF is robust across a range of values (e.g. $\alpha = 0.5$–1.5); misclassification and fidelity remain high. Very low α removes the corresponding regularizer and slightly degrades plausibility, while very high α yields negligible gains but more aggressive edits. In practice, moderate α maximizes fidelity and plausibility together.

A.10 Impact of Pruning Strategy

We also evaluate the impact of our candidate-edge pruning on GCN with the Cora dataset. As shown in Figure 6, pruning yields more concise explanations by reducing redundant edits ($\Delta\mathbf{A} = 1.71 \rightarrow 1.62$), while maintaining nearly identical predictive accuracy (misclassification $= 0.71$) and plausibility (0.76 vs. 0.75). Runtime is significantly reduced (6.12s vs. 3.00s), confirming that pruning improves explanatory minimality and efficiency without sacrificing fidelity or plausibility.

A.11 Proof and Evidence for the Hypotheses

Throughout this section, let v be the target node under analysis. We use $s_G(v)$ to denote the logit of the target class for v, $f_G(v)$ for the predicted label of v, and m_v for the margin between the logit of v's true class and the highest competing class.

By definition, $CFEx(G)$ is an inclusion-minimal set of edge modifications (additions or deletions) such that applying them flips f's prediction for node v. Minimal means that no proper subset of

Table 12: Performance of counterfactual explanations on **Loan-Decision** and GAT.

Method	Base GNN	Misclass. ↑	Fidelity ↑	$\Delta\mathbf{E}(\mathbf{E}^+, \mathbf{E}^-)$ ↓	Plausibility ↑	Time (sec) ↓
CF-GNNExplainer	GAT	–	–	–	–	–
GNNExplainer	GAT	0.01±0.005	0.0002±0.0315	3.00±0.00 (0.00, 3.00)	0.34±0.015	3.40±1.67
PGExplainer	GAT	0.08±0.032	0.0057±0.0012	3.39±0.00 (0.00, 3.39)	0.46±0.092	**0.06±0.01**
Nettack	GAT	0.41±0.006	0.0781±0.0078	5.00±0.12 (3.93, 1.07)	0.20±0.014	1.01±0.51
GOttack	GAT	0.32±0.007	0.0689±0.0043	5.00±0.04 (4.80, 0.20)	0.18±0.005	0.60±0.18
ATEX-CF (Ours)	GAT	**0.47±0.021**	**0.0892±0.0193**	**1.58±0.03 (0.67, 0.91)**	**0.65±0.019**	4.32±1.95

Table 13: Ablation Study on **Cora** and GCN.

Method	Base GNN	Misclass. ↑	Fidelity ↑	$\Delta\mathbf{E}(\mathbf{E}^+, \mathbf{E}^-)$ ↓	Plausibility ↑	Time (sec) ↓
w/o $\mathcal{L}_{dist}$	GCN	0.70	0.2360	1.66 (0.76, 0.90)	0.71	7.42
w/o $\mathcal{L}_{plau}$	GCN	0.68	0.2225	**1.57 (0.72, 0.85)**	0.71	6.82
w/o $\mathcal{L}_{dist}$ and $\mathcal{L}_{plau}$	GCN	0.69	**0.2469**	1.60 (0.59, 1.01)	0.68	6.22
ATEX-CF (Ours)	GCN	**0.71**	0.2336	1.62 (0.92, 0.70)	**0.75**	**3.80**

those modifications is sufficient to flip the prediction. Let us denote this modification set by $F := CFEx(G)$.

$$f_{G\oplus F}(v) \neq f_G(v),$$

but for any strict subset $F' \subsetneq F$,

$$f_{G\oplus F'}(v) = f_G(v).$$

We assume that the influence function of f over edge sets is submodular, so the marginal effect of adding or removing an edge diminishes as more modifications are applied (influence functions on graphs are often modeled as submodular (Krause & Guestrin, 2007; Borgs et al., 2014)). This submodularity assumption implies that the minimal counterfactual explanation set F is unique, which ensures that alignment between the attack-selected edges and the explanation subgraph is well defined, i.e., the top-k edges chosen by the attack coincide with the uniquely defined set F rather than an arbitrary minimal set.

When multiple such minimal sets exist, we fix a canonical choice by breaking ties, for example, by selecting the lexicographically smallest edge set. Intuitively, F captures the single most crucial evidence subgraph in G supporting the original prediction.

For any edge e and set of edges S, define the conditioned marginal effect as $\Delta_e f(G \cup S;\, v) := f_{G\cup S\cup\{e\}}(v) - f_{G\cup S}(v)$.

A.11.1 Hypothesis H1: Edge Gradient Attack Alignment

Hypothesis 1 (Restated)**.** *Let $G = (A, X)$ be an input graph and f a pre-trained GNN classifier. For a target node v, let $\Delta G(E^+)$ denote the set of added edges in an evasion attack that flips the prediction of f, and let $CFEx(G)$ denote the counterfactual explanation graph of the graph G. Then, the graph similarity between $\Delta G(E^+)$ and $CFEx(G)$:*

$$\textit{Sim}(\Delta G(E^+), CFEx(G)) \approx c, \quad 0 << c < 1,$$

where Sim$(\cdot,\cdot)$ denotes a graph similarity measure by graph edit distance, maximum common subgraph, and graph embedding vectors, and c is a positive score, indicating non-trivial overlap between the attack edges and the explanation graph.

Proof Sketch: Edges with the largest gradient influence on the target node's logit margin are the most potent for adversarial attacks. Formally, for a target node v with margin $m_v(A)$ and edge gradients $g_e = \partial m_v / \partial A_e$, suppose e_1 and e_2 are two candidate edges (with e_1 either currently present or absent depending on the attack type, and similarly for e_2). If $|g_{e_1}| > |g_{e_2}|$, then flipping e_1 (adding it if $g_{e_1} < 0$ or removing it if $g_{e_1} > 0$) yields a larger drop in m_v than flipping e_2. In particular, a Projected Gradient Descent (PGD) attack will primarily select edges from among those with the highest $|g_e|$, aligning adversarial modifications with the gradient-based explanation subgraph. Intuitively, the gradient g_e indicates how sensitively the margin m_v changes with respect to edge e. A large-magnitude gradient $|g_e|$ means that a small change in A_e has a big effect on m_v. In a 2-layer GCN with ReLU, the model is piecewise linear, so locally m_v changes approximately linearly with A_e. Thus, the edge with the largest $|g_e|$ produces the steepest change in m_v when

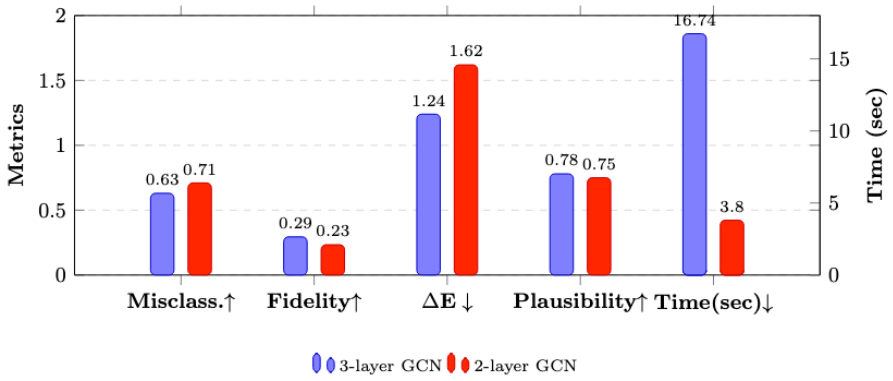

Figure 4: Performance of counterfactual explanations vs. the number of GNN layers: The results demonstrate sensitivity w.r.t. the number of hops for the local structure surrounding the target node.

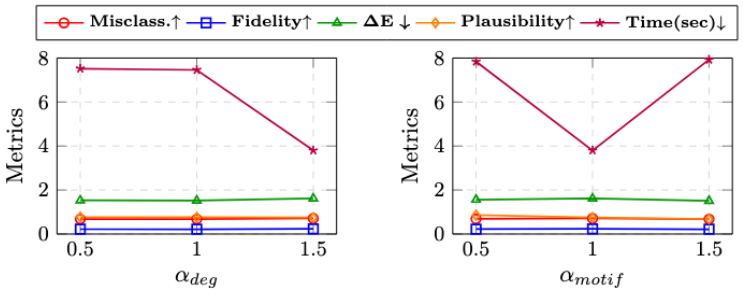

Figure 5: Sensitivity w.r.t. Hyperparameters α_{deg} and α_{motif}.

perturbed. A PGD adversarial attack, which follows the gradient of the loss (or negative margin), will therefore choose the edge with the most negative gradient (for additions) or the most positive gradient (for deletions) to maximally decrease the margin. In essence, explanation methods pick out these high-$|g_e|$ edges as important, and the attacker targets the very same edges to flip the prediction.

Proof. Consider the target node v with true class y_v and margin $m_v(A) = z_{y_v}(A, v) - \max_{c \neq y_v} z_c(A, v)$. Let $g_e = \frac{\partial m_v}{\partial A_e}$ be the gradient influence of edge e on the margin. We analyze edge addition, and show that larger $|g_e|$ implies a greater reduction in margin when e is perturbed:

Edge addition (E+ attack). Suppose $e = (i, j)$ is a non-existent edge ($A_e = 0$). If $g_e < 0$, then e is a detrimental or counterfactual edge for the current prediction: increasing A_e (adding this edge) will lower the margin m_v. In a small continuous relaxation of A_e, m_v would decrease by about $|g_e| \cdot \Delta A_e$. For the actual discrete addition ($A_e : 0 \to 1$), the change $m_v(A_{+e}) - m_v(A)$ will be approximately g_e (since g_e is negative, this is a drop in margin). Because our GCN is piecewise linear (ReLU activation), adding e causes a margin change on the order of g_e. If $|g_{e_1}| > |g_{e_2}|$ for two absent edges with negative gradients, adding e_1 produces a larger margin drop than adding e_2. Thus, an adversary performing PGD will add the edge with the most negative gradient first, which is precisely the top edge identified by a counterfactual explanation method.

The attacker's choice of edge corresponds to the edge with the largest $|g_e|$ that reduces the margin (negative g_e for addition). By repeating this argument iteratively (considering the next most influential edge after the first, and so on), one can see that an attack adding/removing k edges will choose the k edges with highest gradient magnitudes that contribute to lowering m_v. Therefore, the set of edges targeted by the PGD attack aligns with the gradient-based counterfactual explanation subgraph (which consists of edges with the largest $|g_e|$). This establishes that ranking edges by $|g_e|$ is equivalent to ranking them by adversarial effectiveness, proving the hypothesis. □

Empirical Evidence for Hypothesis 1

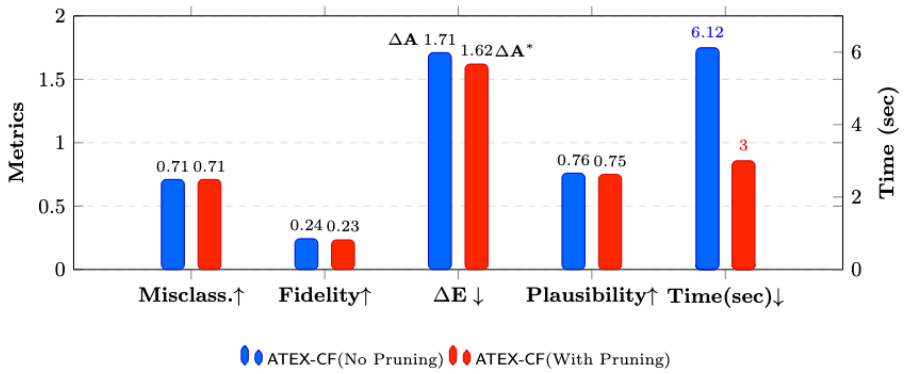

Figure 6: Effectiveness of Post-Hoc Pruning.

Table 14: **Attacks and Counterfactuals.** The structural similarity between evasion attack edges $\Delta\mathbf{G}$ (mainly additions $\Delta\mathbf{E}^+$ from GOttack) and instance-level factual explanations $Ex(G')$ from GNNExplainer on post-attack graph G'. 280 target nodes are correctly classified in the original graph G. Budget = 5. GCN (2-layer), **Cora** dataset.

Metric	All (280)	Attack Success (225)	Attack Fail (55)
GED↓	0.38	0.37	0.41
MCS↑	0.31	0.33	0.24
GEV↑	0.72	0.80	0.39

The results in Table 14 support Hypothesis 1, which posits a high structural overlap between the attacker's perturbation ΔG and the counterfactual explanation $CFEx(G)$ produced by pre-attack explanation methods. Notice that here we consider the instance-level factual explanations $Ex(G')$ from GNNExplainer (Ying et al., 2019) on the post-attack graph G' as a proxy for the counterfactual explanation $CFEx(G)$ produced by pre-attack explanation methods. This is because the state-of-the-art counterfactual explainers generally do not support edge addition.

In both correctly and incorrectly predicted instances, the Graph Edit Distance (GED) remains moderate (≈ 0.38), and the Maximum Common Subgraph (MCS) similarity is non-negligible, particularly for successful attacks. Notably, Graph Embedding Vector (GEV) similarity reaches 0.88 for misclassified nodes and 0.80 for successful attacks on correctly predicted nodes, indicating substantial alignment in the embedded subgraph structure. In other words, Table 14 shows that similarity between attack perturbations ΔG and counterfactual explanations $CFEx(G)$ depends strongly on attack outcome. For successful attacks, distances such as GED are lower (lower is better) and similarities such as MCS and GEV are higher (higher is better), while for failed attacks, the opposite holds. In other words, when the attack succeeds, the perturbations align closely with counterfactual explanations, whereas in failed cases the overlap weakens. This pattern offers evidence that effective adversarial edits not only cause misclassification but also resemble the explanatory structures that counterfactual methods would identify.

A.11.2 Propositions on Counterfactual Completeness via Attack-Informed Additions

In principle, for the completeness of our hypothesis, one would like to prove that edge additions "always" yield a successful counterfactual attack, which would strengthen our claim that unifying attacks and counterfactuals is universally beneficial, even when counterfactuals alone fail. Unfortunately, this cannot be guaranteed, since the data may lack any node whose connection to the target would flip its label. Instead, we establish a next-best guarantee: When sufficiently informative opposite-class nodes exist, additions can flip the label while deletions cannot. State-of-the-art counterfactual explanations may overlook such opportunities, but attack algorithms are designed to exploit them.

Let f be a GNN classifier and let $v \in V$ be a target node with $f_G(v) = y$. Throughout, $s_G(v)$ denotes a real valued class y score for v, $f_G(v)$ denotes the predicted label, and m_v denotes the margin for class y at v. w_{vu} is the weight assigned by the model to the contribution of neighbor u when aggregating into the score of node v. Fix a one versus rest view for class y and use the decision rule $f_G(v) = y$ if and only if $s_G(v) > 0$. Assume an additive, degree-independent neighborhood model

$$s_G(v) = bias_v + \sum_{u \in \mathcal{N}(v)} w_{vu}\, r_u,$$

with $w_{vu} \geq 0$, where r_u is the contribution aligned with class y. This additive influence model abstracts away normalization and attention redistribution, but shows the monotonic nature of homophilic neighborhoods. While GNNs are more complex, we observe empirically that their behavior is consistent with the model's prediction: deletion of a few homophilic neighbors rarely flips predictions, whereas a small number of targeted additions frequently does (as evidenced in the addition attacks of Gottack Alom et al. (2025) and Nettack Zügner et al. (2018)).

We assume homophily in the immediate neighborhood so that $f_G(u) = y$ for all $u \in \mathcal{N}(v)$, hence $r_u \geq 0$ for all incident neighbors. No term in $s_G(v)$ is rescaled by $|\mathcal{N}(v)|$.

In this setting, deletion and addition have asymmetric effects. Deleting any number of incident edges can only remove nonnegative summands, while adding edges to informative opposite class nodes can introduce negative summands. The next two propositions formalize this.

Proposition A.1 (Deletion Infeasibility)**.** *Let $G' = G \setminus S$ for some strict subset $S \subsetneq (v, u) : u \in \mathcal{N}(v)$. If*

$$bias_v + \min_{u \in \mathcal{N}(v)} w_{vu}\, r_u \;>\; 0,$$

where $bias_v$ is a bias term for node v's own features, r_u is the contribution from neighbor u's features, aligned with class y, and $w_{vu} \geq 0$ is the scalar weight that measures how strongly neighbor u influences v's score. Then $f_{G'}(v) = y$. In words, as long as at least one incident neighbor remains, the score stays positive, and the label does not change.

Argument. The smallest possible post-deletion score over all strict subsets occurs when only the least contributing neighbor of v remains. This score equals $b_v + \min_u w_{vu} r_u$, which is positive by assumption, hence $f_{G'}(v) = y$.

Proposition A.2 (Addition Sufficiency)**.** *Suppose there exists a set of candidate nodes C with $f_G(u) \neq y$ such that for each $u \in C$, adding the edge (v, u) decreases the score by at least a fixed amount $\gamma > 0$:*

$$s_{G \cup \{(v,u)\}}(v) \;\leq\; s_G(v) - \gamma.$$

Let $m_v = s_G(v) > 0$. Then there exists a set $E^+ \subseteq (v, u) : u \in C$ with

$$|E^+| \;\leq\; \lceil m_v/\gamma \rceil$$

such that $f_{G \cup E^+}(v) \neq y$. Thus, a small number of informative additions flips the prediction.

Argument. Each addition reduces the score by at least γ. After $k = \lceil m_v/\gamma \rceil$ additions, the score is nonpositive, which changes the predicted label.

Corollary A.3 (Budgeted reachability and strict advantage of additions)**.** *Let $\mathcal{R}_{\mathrm{del}}(k) = \{G \setminus S : S \subseteq \{(v, u) : u \in \mathcal{N}(v)\}, |S| \leq k\}$ and $\mathcal{R}_{\mathrm{add}}(k) = \{G \cup E^+ : E^+ \subseteq \{(v, u) : u \in C\}, |E^+| \leq k\}$. Under the assumptions above, if*

$$bias_v + \min_{u \in \mathcal{N}(v)} w_{vu} r_u > 0,$$

then for every $k < |\mathcal{N}(v)|$ there is no graph in $\mathcal{R}_{\mathrm{del}}(k)$ that flips v's label. If, in addition, there exists $\gamma > 0$ such that each (v, u) with $u \in C$ decreases $s_G(v)$ by at least γ, then with $k_+ = \lceil m_v/\gamma \rceil$

Table 15: Failure rate of deletion-based counterfactual explanations for correctly predicted target nodes (**Cora**, 2-layer GCN).

Method	Total Nodes	Has CF Explanation	No CF Explanation
CF-GNNExplainer	280	54	226

Table 16: Failure rate of deletion-based counterfactual explanations for incorrectly predicted target nodes (**Cora**, 2-layer GCN).

Method	Total Nodes	Has CF Explanation	No CF Explanation
CF-GNNExplainer	220	76	144

there exists $G^+ \in \mathcal{R}_{\text{add}}(k_+)$ that flips v's label. Consequently, whenever $k_+ < |\mathcal{N}(v)|$, the set of counterfactuals reachable by at most k_+ additions is nonempty while the set reachable by at most k_+ deletions is empty, hence additions strictly dominate deletions under equal edit budgets.

Argument. The deletion claim follows from the deletion infeasibility proposition. The addition claim follows from the addition sufficiency proposition with $k_+ = \lceil m_v/\gamma \rceil$. If $k_+ < |\mathcal{N}(v)|$, then $\mathcal{R}_{\text{add}}(k_+)$ contains a prediction flipping graph while $\mathcal{R}_{\text{del}}(k_+)$ does not.

Corollary A.4 (Edit cost and latent stability)**.** *Let d_{edit} be the edge edit distance. Any witnessing addition set E^+ has $d_{\text{edit}}(G, G \cup E^+) = |E^+| \leq \lceil m_v/\gamma \rceil$. If a node-level embedding map $\psi(v; G)$ is L-Lipschitz with respect to incident edge edits at v, then*

$$\|\psi(v; G) - \psi(v; G \cup E^+)\|_2 \leq L\,|E^+| \leq L\,\lceil m_v/\gamma \rceil.$$

Thus the latent perturbation can be bounded linearly by the required number of additions.

Remark. The strict advantage condition $k_+ < |\mathcal{N}(v)|$ is testable from estimates of m_v and per edge gains. If $k_+ \geq |\mathcal{N}(v)|$, the theory is agnostic about dominance, but the separation holds whenever the margin-to-gain ratio is small relative to the neighborhood size.

Empirical Evidence for the Counterfactual Completeness

Tables 15 and 16 show how often CF-GNNExplainer (Lucic et al., 2022) fails to generate deletion-only counterfactual explanations under two conditions.

In Table 15 (correctly predicted target nodes), out of 280 test nodes, only the "HAS CF EXPLANATION" column reports 54 nodes ($\approx 19\%$) for which a deletion-based counterfactual exists; the remaining 226 nodes ($\approx 81\%$) are in the "NO CF EXPLANATION" column. Similarly, in Table 16 (misclassified nodes), 76 out of 220 nodes ($\approx 35\%$) have a deletion-only counterfactual, while 144 nodes ($\approx 65\%$) do not.

These high failure rates support our theoretical propositions and corollaries: namely, that there are many nodes for which deletion-based counterfactuals are infeasible. These empirical gaps justify the necessity of incorporating attack-informed edge additions to recover explanations for those nodes.