Collateral Damage Constrained Backdoor Attacks on Graph Neural Networks

Di jin[1,2], **Zechuan Zhang**[1], **Bingdao Feng**[1,*], **Xiaobao Wang**[3], **Dongxiao He**[1] and **Zhen Wang**[4]
[1]School of Computer Science and Technology, Tianjin University, Tianjin, China
[2]Key Laboratory of Artificial Intelligence Application Technology, Qinghai Minzu University, Xining, 810007, China
[3]School of Artificial Intelligence, Tianjin University, Tianjin, China
[4]School of Cybersecurity, Northwestern Polytechnical University, Xi'an, China
{jindi, zechuanzhang, fengbingdao, wangxiaobao, hedongxiao}@tju.edu.cn, w-zhen@nwpu.edu.cn

Abstract

Graph Neural Networks (GNNs) are vulnerable to backdoor attacks, where models behave normally on clean data but exhibit targeted misclassifications once specific triggers are activated. Existing backdoor attacks on GNNs mainly focus on enhancing trigger stealthiness or diversifying attack paradigms. However, these methods overlook a fundamental property of GNNs: trigger-induced malicious signals inevitably propagate through graph neighborhoods, causing unintended mispredictions on clean nodes, i.e., Collateral Damage. To address this issue, we propose the Collateral Damage Constrained Graph Backdoor Attack (CDCA), a novel framework that explicitly controls malicious diffusion. Specifically, the proposed method combines neighborhood-aware target node selection with a self-constrained trigger generation strategy to suppress trigger-induced propagation by enforcing prediction consistency on clean K-hop neighboring nodes. Extensive experiments on real-world datasets demonstrate that the proposed method remains effective while significantly reducing collateral damage.

1 Introduction

Graphs, mathematical structures used to model pairwise relationships between objects, arise in a wide range of real-world applications, including social networks [Hamilton *et al.*, 2017], recommendation systems [Fan *et al.*, 2019], financial applications [Liu *et al.*, 2021], and biological networks [Li *et al.*, 2023]. To effectively model such relational data, Graph Neural Networks (GNNs) have emerged as a powerful learning paradigm [Kipf and Welling, 2017; Veličković *et al.*, 2018; Xu *et al.*, 2019; Zhang and Zitnik, 2020; Wu *et al.*, 2020]. By iteratively aggregating information from neighboring nodes through the message-passing mechanism [Gilmer *et al.*, 2017], GNNs have achieved state-of-the-art performance on node classification [Liu *et al.*, 2022; Yang *et al.*, 2023; Yan *et al.*, 2025].

*Corresponding author

Despite achieving remarkable success, recent studies [Lyu *et al.*, 2024; Li *et al.*, 2025; Jin *et al.*, 2025a] have shown that GNNs are vulnerable to backdoor attacks and even small perturbations can effectively manipulate model predictions. Typically, graph backdoor attack involves injecting triggers into target nodes to create a shortcut between the target and an adversary-chosen label. During inference phase, once these backdoors are activated by the trigger, the target nodes are misclassified into the malicious class. Following pioneering works such as SBA [Zhang *et al.*, 2021] and GTA [Xi *et al.*, 2021], methods like UGBA [Dai *et al.*, 2023] and DPGBA [Zhang *et al.*, 2024] focus on optimizing attack strategies to bypass specific defense mechanisms and enhance stealthiness. However, these methods overlook a critical phenomenon inherent to graph neural networks: due to the message-passing mechanism, malicious messages propagate during inference and affect nodes beyond the intended targets based on their receptive field, thereby amplifying the risk of backdoor exposure. We term this phenomenon Graph Backdoor **Collateral Damage**. As illustrated in Fig. 1, once the trigger is embedded into a target node, numerous non-target nodes within K-hop neighborhoods are also affected by the trigger, where K denotes the number of model layers.

The above observations are inseparable consequences of the message-passing mechanism. To successfully activate the backdoor, the trigger must induce a strong and transferable representation change through neighborhood aggregation. However, this propagation process inevitably contaminates clean nodes within the K-hop range, amplifying the scale of mispredictions and increasing the risk of exposure, thus posing a critical challenge:

How can a backdoor be reliably activated on target nodes while preventing malicious signals from poisoning surrounding clean nodes under the message-passing mechanism?

To address these challenges, we propose a novel framework, the Collateral Damage Constrained Backdoor Attack (CDCA), which introduces a joint training strategy from both structural and representational perspectives. First, CDCA explicitly introduces neighborhood-level constraints during the trigger generation process to localize backdoor effects under message passing mechanism. This design enables the model to manipulate unintended malicious diffusion to surrounding clean nodes while maintaining reliable activation on targets. Second, in the node selection scheme, we prioritize low-

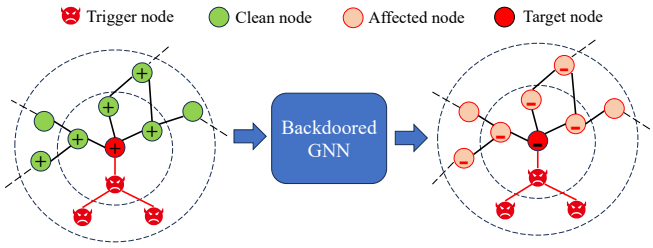


Figure 1: Large-scale mispredictions from collateral damage in a 3-layer GNN. We use "+" and "-" to denote the labels.

degree, structurally less influential nodes to minimize structural perturbation. Additionally, we identify locally inconsistent nodes that are more susceptible to poisoning. This strategy allows for subtle attack perturbations that are effectively dissipated during propagation, thereby enhancing the attack success rate while mitigating the adverse impact on neighboring nodes. Our contributions are summarized as follows:

- We identify the Collateral Damage phenomenon and verify that due to the message-passing mechanism, existing attacks suffer from severe unintended malicious signal diffusion.
- We propose the CDCA, which jointly optimizes trigger generation with collateral damage constraints and employs a neighborhood-aware node selection strategy. This approach effectively localizes the backdoor impact, solving the trade-off between attack success and collateral damage.
- Extensive experiments on multiple benchmark datasets demonstrate that the proposed method significantly preserves the prediction accuracy of neighboring nodes while maintaining a high attack success rate, effectively constraining collateral damage on the graph.

2 Related Work

Graph backdoor attacks pose a significant threat in GNNs [Liu *et al.*, 2024; Feng *et al.*, 2024; Jin *et al.*, 2026]. Attackers inject malicious triggers through subgraphs or feature perturbations in selected training nodes, creating shortcuts to predetermined malicious class [Zhang *et al.*, 2025; Jin *et al.*, 2025b]. Early methods such as SBA [Zhang *et al.*, 2021] adopt fixed subgraphs as triggers. To enhance attack effectiveness, GTA [Xi *et al.*, 2021] introduces a trainable trigger generator for sample-specific perturbations, improving success rates. UGBA [Dai *et al.*, 2023] further refines target selection, reducing the attack budget and improving stealth by generating similar triggers and targets. Furthermore, One-to-*N* [Xu and Picek, 2023] can simultaneously manipulate predictions toward different target classes using different triggers. In the domain of self-supervised learning, GCBA [Zhang *et al.*, 2023] utilizes trigger dispersion to poison the unlabeled pre-training process. Similarly focusing on stealth, DPGBA [Zhang *et al.*, 2024] utilizes adversarial learning to create in-distribution triggers. On the other hand, some backdoor methods focus on perturbing node features while preserving graph topology to enhance stealth. For instance, CGBA [Xia *et al.*, 2025] uses fixed feature triggers, while SPEAR [Ding *et al.*, 2025] extends to adaptive ones. However, feature-based attacks inherently affect more original clean nodes, resulting in greater collateral damage. To mitigate this, our work employs structure-based methods.

3 Preliminary Analysis

In this section, we formally define the problem of graph backdoor attacks within an inductive semi-supervised node classification setting and identify the collateral damage phenomenon arising from the message-passing mechanism.

3.1 Notations

We use $\mathcal{G} = (\mathcal{V}, \mathcal{A}, \mathcal{E}, \mathcal{X}, \mathcal{Y})$ to denote an undirected graph, where $\mathcal{V} = \{v_1, \ldots, v_N\}$ represents the set of N nodes. Graph structure is represented by an adjacency matrix $\mathcal{A} \in \mathbb{R}^{N\times N}$, while $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ denotes the set of edges and the attribute matrix is given by $\mathcal{X} \in \mathbb{R}^{N\times d}$. In this paper, we focus on the backdoor attack for the semi-supervised node classification task within the inductive setting. Specifically, the training graph $\mathcal{G}_{\mathrm{T}}$ includes a small subset of labeled nodes $\mathcal{V}_L \subseteq \mathcal{V}$ with labels $Y_L = \{y_1, \ldots, y_{N_L}\}$. The remaining nodes of $\mathcal{G}_{\mathrm{T}}$ are unlabeled, denoted as $\mathcal{V}_U$. We denote $\mathcal{V}_{\mathrm{B}} = \mathcal{V}_L \cup \mathcal{V}_U$ as the node set for the training graph, while the test nodes are denoted as $\mathcal{V}_T$, i.e., $\mathcal{V}_T \cap \mathcal{V}_{\mathrm{B}} = \emptyset$. Clean nodes are denoted as $\mathcal{V}_C = \mathcal{V}_T \setminus \mathcal{V}_B$. Moreover, a subset of target nodes $\mathcal{V}_p \subset \mathcal{V}_U$ is selected and injecting backdoor trigger g_i into each $v_i \in \mathcal{V}_p$, resulting in embedded nodes $v_i \oplus g_i$.

3.2 Threat Model

Attacker's Goal

The objective of the attack is to inject backdoor triggers to a set of target nodes and mislead the poisoned model θ^* to misclassify target nodes as the target label y_t, while retaining accurate classification predictions on clean nodes. We can formalize attack objective as follows:

$$\begin{cases} f_{\theta^*}(v_i \oplus g_i) = y_t, & \forall v_i \in \mathcal{V}_p \\ f_{\theta^*}(v_j) = y_j, & \forall v_j \in \mathcal{V}_C \end{cases}. \tag{1}$$

Attacker's Knowledge and Capability

Like most graph attacks [Zügner *et al.*, 2020; Sun *et al.*, 2020], attackers can access the training graph while remaining unaware of the victim GNNs model, including its architecture, training hyper-parameters, or defense mechanisms. Regarding capability [Zhang *et al.*, 2024], the attacker is able to select a subset of nodes within a predefined budget, attach adaptive triggers to poison the graph. During inference, the attacker retains the ability to attach trigger to target nodes.

3.3 Collateral Damage Analysis

Due to the recursive message-passing mechanism of GNNs, node representations are learned by aggregating information from local neighborhoods. In GCN [Kipf and Welling, 2017], the embedding of a neighbor is directly influenced by the features of connected nodes via the layer-wise propagation rule:

$$\mathbf{H}^{(l+1)} = \sigma\left(\hat{\mathbf{A}}\mathbf{H}^{(l)}\mathbf{W}^{(l)}\right), \tag{2}$$

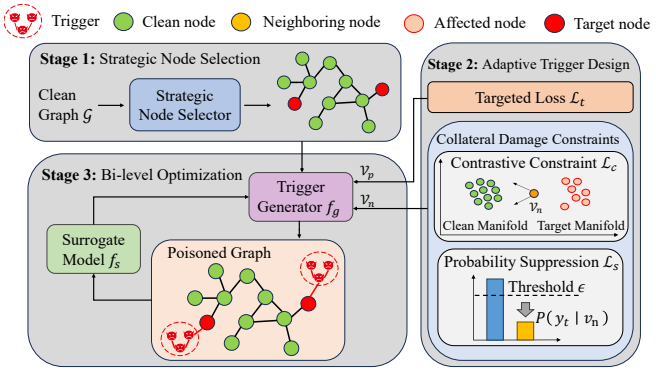


Figure 2: Framework of CDCA.

where $\mathbf{H}^{(l)}$ denotes the node embeddings at layer l (with $\mathbf{H}^{(0)} = \mathbf{X}$), $\mathbf{W}^{(l)}$ is the trainable weight matrix, and $\sigma(\cdot)$ is an activation function. The normalized adjacency matrix is given by $\hat{\mathbf{A}} = \tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}$, where $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}_N$ includes self-loops and $\tilde{\mathbf{D}}$ is the degree matrix.

Consider a set of clean neighboring nodes $\mathcal{V}_n$ adjacent to the target nodes $\mathcal{V}_p$. When a target node is injected with a strong trigger optimized for a target class y_t, the aggregation process propagates this malicious signal to neighboring clean nodes, shifting their predictive distributions towards y_t. We define this unintended corruption of clean neighbors as **Collateral Damage**. Accordingly, we emphasize preserving the neighboring prediction accuracy in Eq. 1 by requiring:

$$f_{\theta^*}(v_n) = y_n, \forall v_n \in \mathcal{V}_n. \tag{3}$$

Collateral damage manifests as a localized cluster of high-confidence target-class predictions. Such a conspicuous distribution creates a distinct statistical footprint, which not only degrades model utility on local clean nodes but also creates anomalies that render the backdoor vulnerable to detection.

4 Methodology

In this section, we present the proposed neighborhood-aware backdoor attack framework, which aims to achieve effective backdoor attacks while explicitly suppressing collateral damage on neighboring nodes. Two challenges must be addressed: (i) how to learn adaptive triggers that reliably induce target nodes to be classified as y_t; and (ii) how to prevent the malicious signal from contaminating the neighboring clean nodes $\mathcal{V}_n$, thereby preserving local clean accuracy and improving stealthiness. To tackle these challenges, CDCA consists of a target node selector that prefers low-centrality local outliers, and an adaptive trigger generator f_g trained via bi-level optimization, and a surrogate model f_s. The trigger generator is optimized not only for the targeted objective on $\mathcal{V}_p$, but also with explicit constraints on neighborhood $\mathcal{V}_n$. The framework is illustrated in Fig. 2.

4.1 Strategic Node Selection

To maximize the utilization of the attack budget, the selection of target nodes is critical. By analyzing the layer-wise propagation rule Eq. 2, we examine how degree affects message passing. For a target node injected with a trigger $v_p \oplus g_i$, its embedding $\mathbf{h}_{v_p}$ is computed by aggregating messages as:

$$\mathbf{h}_{v_p} = \sigma\Big(\sum_{v_k \in \mathcal{N}(v_p)} \frac{1}{\sqrt{d_{v_p} d_{v_k}}} \mathbf{W}\mathbf{x}_{v_k}\Big), \tag{4}$$

where d_{v_p} denotes the degree of v_p and $\mathcal{N}(v_p)$ represents the set of all neighboring nodes of v_p and itself. Considering the contribution from the trigger g_p specifically, the influence of g_p is just a part of the whole term. This formulation reveals critical insights. First, the trigger signal facilitates widespread diffusion across the graph through numerous connections; second, message passing from clean neighboring nodes tends to dilute the malicious signal, reducing attack efficacy. Consequently, we prioritize lower-degree nodes in our node selection strategy. This approach minimizes information interference from neighbors, allowing the trigger pattern to better dominate the classification of the target node while reducing structural perturbation. From a representational geometry perspective, local outlier nodes positioned far from the class-wise distribution center are more vulnerable to backdoor signals, while neighbors with locally consistent representations are more robust to the collateral damage. Collectively, these insights guide our selection of low-degree outliers, which can be effectively poisoned with minor perturbations while naturally constraining unintended damage.

To implement our strategy, we pre-train a benign GCN encoder to obtain node embeddings and compute a selection score $\mathcal{S}(v_i)$ for each candidate node $v_i \in \mathcal{V}_U$:

$$\mathcal{S}(v_i) = \mathrm{N}(\|\mathbf{h}_{v_i} - \mathbf{c}(v_i)\|_2) - \mathrm{N}(d_{v_i}), \tag{5}$$

where $\mathbf{c}(v_i)$ represents the centroid of v_i's local neighborhood, and $\mathrm{N}(\cdot)$ is a normalization function. By ranking the top-K nodes with the highest $\mathcal{S}(v_i)$ as $\mathcal{V}_p$, we select target nodes, ensuring that the injected trigger signal remains potent and is not diluted by the graph structure.

4.2 Neighborhood-Aware Trigger Generation

The target node selector identifies nodes with representative feature and structural patterns as attack entry points. Conditioned on the selected nodes, the adaptive trigger generator learns node-specific trigger subgraphs under multiple stealthiness and neighborhood-awareness constraints. To ensure that the learned triggers generalize to unseen test nodes, we adopt a bi-level optimization strategy, where the surrogate model guides trigger learning while the trigger generator balances attack effectiveness and neighborhood constraint. For each target node $v_i \in \mathcal{V}_p$, we generate a trigger subgraph g_i and attach it to v_i to form the poisoned graph.

Design of Trigger Generator

To generate adaptive triggers that align with the local semantics of attacked nodes, we design a trigger generator that conditions trigger construction on the feature representation of each target node. For a target node $v_i \in \mathcal{V}_p$ with feature vector $\mathbf{x}_{v_i}$, the generator produces both features and structure of the trigger subgraph. Formally, we project the node features into a latent space using a multi-layer perceptron:

$$\mathbf{h}_i = \mathrm{MLP}(\mathbf{x}_{v_i}), \tag{6}$$

which captures node-specific semantic information. Based on this representation, the generator simultaneously outputs the trigger node features and trigger adjacency structure:

$$\mathbf{X}^{g_i} = \mathbf{W}_f \mathbf{h}_i, \quad \mathbf{A}^{g_i} = \mathbf{W}_a \mathbf{h}_i, \tag{7}$$

where $\mathbf{W}_f$ and $\mathbf{W}_a$ are learnable parameters for feature and structure generation.

Since graphs are inherently discrete, directly optimizing a continuous adjacency matrix is infeasible. Following prior work on binary neural networks [Xi *et al.*, 2021], we apply a straight-through estimator to binarize the generated adjacency matrix and the resulting trigger is defined as $g_i = (\mathbf{X}^{g_i}, \tilde{\mathbf{A}}^{g_i})$, which is attached to the target node v_i to construct the poisoned graph. During training, target nodes are expected to be misclassified into the target label y_t. At inference time, the same generator f_g is applied to test set, and the generated triggers activate the backdoor behavior, causing the compromised GNNs to predict the target class.

Attack Objective

Given a surrogate GNNs model f_s parameterized by θ_s, we train f_s on the poisoned graph to enforce targeted misclassification on targeted nodes while preserving supervision on clean labeled nodes. The attack objective is optimized as:

$$\mathcal{L}_t(\theta_s, \theta_g) = \sum_{v_i \in \mathcal{V}_p} \ell(f_s(v_i \oplus g_i), y_t) + \sum_{v_j \in \mathcal{V}_L} \ell(f_s(v_j), y_j)\,, \tag{8}$$

where $\ell(\cdot,\cdot)$ denotes the cross-entropy loss and $v_i \oplus g_i$ indicates the target node with the attached trigger. Compared to existing backdoor attacks that emphasize attack generalization by encouraging a broad set of training nodes to be classified into the target class, our objective focuses on attack precision. Specifically, $\mathcal{L}_t$ is simultaneously optimized with respect to the surrogate model and the trigger generator, and serves as a loss for both θ_s and θ_g. We explicitly retain the standard supervision on the labeled clean node set $\mathcal{V}_L$ to preserve the overall usability and predictive performance of the surrogate model, while restricting the targeted poisoned objective exclusively to the $\mathcal{V}_p$. This design reflects a more realistic threat model, where the attacker aims to preserve attack effectiveness while ensuring that the backdoor effect is precisely localized to the target nodes.

Neighbor Contrastive Constraint

While the attack objective is clear, backdoor signals can inevitably propagate to neighboring clean nodes through message passing. To mitigate such collateral damage without altering any node features, we introduce a neighbor contrastive regularization that constrains the representations of clean neighbors around target nodes.

Let $\mathcal{V}_n = \mathcal{N}(\mathcal{V}_p) \setminus \mathcal{V}_p$ be the set of clean neighbors of target nodes, and let $\mathbf{z}_v$ be the penultimate-layer embedding of node v extracted from f_s. Due to the attacker's limited capability, we assume that the attacker cannot modify the original nodes or edges in the graph and has no access to the ground-truth labels of the unlabeled nodes.

Therefore, we construct supervision without relying on ground-truth labels for neighbors by utilizing pseudo-labels y'_v obtained from the trained benign GCN during the node selection phase, which strictly adheres to a realistic black-box attack setting. Based on these pseudo-labels, we define (i) a *target manifold* $\mathcal{C}' = \{v \in \mathcal{V}_n : y_v = y_t\}$ and (ii) a *clean manifold* $\mathcal{C}$ consisting of nodes whose pseudo-labels are not y_t. Our evaluation across varying scenarios demonstrates that even with the inherently pronounced noise in OGB-arXiv or the small graph scale of Cora, the supervision signal is sufficient to separate the two distributions in the embedding space, proving empirically robust to guide the constraint process. We then formulate the neighbor contrastive constraint loss as:

$$\mathcal{L}_c = -\sum_{v_u \in \mathcal{V}_n} \log \frac{\sum\limits_{v_k \in \mathcal{C}} \mathcal{S}(\mathbf{z}_{v_u}, \mathbf{z}_{v_k})}{\sum\limits_{v_k \in \mathcal{C}} \mathcal{S}(\mathbf{z}_{v_u}, \mathbf{z}_{v_k}) + \sum\limits_{v_j \in \mathcal{C}'} \mathcal{S}\big(\mathbf{z}_{v_u}, \mathbf{z}_{v_j}\big)}, \tag{9}$$

where $\mathcal{S}(\cdot)$ is the cosine similarity. This contrastive objective encourages neighbors to remain close to the clean manifold while being distanced from the target-class manifold. Nodes in $\mathcal{C}'$ are considered negative samples, representing nodes suffering collateral damage, and these instances are actively addressed during optimization.

Probability Suppression

While $\mathcal{L}_c$ constrains the geometry of the embedding space, it does not explicitly regulate the classifier's output confidence. To further prevent neighbors from being misclassified as the target class, we introduce a probability-level suppression term. Let $P(y_t \mid v_u)$ denote the predicted probability of class y_t for neighbor node v_u. We impose a probability suppression, which can be formulated as:

$$\mathcal{L}_s = \frac{1}{|\mathcal{V}_n|} \sum_{v_u \in \mathcal{V}_n} \max(0,\ P(y_t \mid v_u) - \epsilon)\,. \tag{10}$$

Eq. 10 employs a margin-based penalty to avoid imposing a hard constraint that forces all neighboring nodes away from the target class because it is possible that some neighbors naturally belong to the target class. To prevent such neighbors from being over-penalized, we introduce a margin ϵ to relax the suppression constraint. This allows neighbors with a reasonable target-class confidence to remain unaffected, while only penalizing those whose target probability is abnormally amplified by backdoor propagation. Empirically, we set $\epsilon = 0.1$ on small datasets, and $\epsilon = 0.02$ on the larger OGB-arXiv dataset. Notably, these values are close to $1/|\mathcal{Y}|$, the expected probability of a random class, which is consistent with our analysis and empirical observations.

4.3 Bi-level Optimization

We jointly optimize the surrogate model and trigger generator using a bi-level optimization strategy. Model parameters after t iterations are denoted as θ^t and α is the learning rate for θ. In the lower-level optimization, the trigger generator θ_g is fixed and the surrogate model θ_s is updated as follows:

$$\theta_s^{(t+1)} = \theta_s^{(t)} - \alpha_s \nabla_{\theta_s} \mathcal{L}_t(\theta_s, \theta_g), \tag{11}$$

which ensures that the surrogate model accurately captures the backdoor behavior induced by the current trigger.

In the upper-level optimization, the surrogate model parameters are treated as fixed, and the trigger generator is updated by a weighted combination of attack effectiveness and collateral damage constraint objectives:

$$\theta_g^{(t+1)} = \theta_g^{(t)} - \alpha_g \nabla_{\theta_g} \Big(\mathcal{L}_t + \beta \mathcal{L}_c + \gamma \mathcal{L}_s \Big). \quad (12)$$

This optimization enables the trigger generator to adaptively ensure the attack success on $\mathcal{V}_p$ and suppression of unintended effects on neighboring clean nodes.

4.4 Enhancing Performance against Defenses

Alongside the sophisticated design in terms of collateral damage, it is also significant to boost the stealth of CDCA by bypassing existing defense methods. Inspired by previous work [Dai *et al.*, 2023; Zhang *et al.*, 2024], we integrate an Out-of-Distribution detection module into CDCA to compete with defenses. This detector f_d uses a binary classification loss in a Min-Max game to differentiate clean images from generated triggers, aiming to make triggers indistinguishable from in-distribution data. It can be formulated as:

$$\mathcal{L}_D = \sum_{v_i} \log(f_d(v_i)) + \sum_{v_g} \log(1 - f_d(v_g)), \quad (13)$$

where v_i is representative nodes selected from original graph and v_g is from trigger nodes. We name the augmented variants as CDCA-D.

5 Experiments

In this section, we present the experimental setup, results, and analyses based on the following research questions: **Q1:** Can CDCA exceed the performance of leading backdoor attacks while addressing collateral damage? **Q2:** Can CDCA effectively circumvent defense mechanisms? **Q3:** How does the number of target nodes affect the performance of CDCA? **Q4:** How does each component affect overall model performance? **Q5:** How sensitive is the model to parameter variations?

5.1 Experimental Settings

Datasets

To evaluate the effectiveness of CDCA, we conduct experiments on four real-world datasets, i.e., Cora, CiteSeer, PubMed [Sen *et al.*, 2008], and OGB-arXiv [Hu *et al.*, 2020], which are commonly applied in inductive semi-supervised node classification. The summaries are provided in Table 1.

Dataset	Nodes	Edges	Features	Classes
Cora	2,708	5,429	1,433	7
CiteSeer	3,327	4,552	3,703	6
PubMed	19,717	44,338	500	3
OGB-arXiv	169,343	1,166,243	128	40

Table 1: Dataset statistics.

Baseline and Defense Methods

We compare CDCA against several state-of-the-art backdoor baselines, providing a comprehensive evaluation of their effectiveness and stealthiness. The attack methods include three structure-based methods: GTA [Xi *et al.*, 2021], UGBA [Dai *et al.*, 2023], and DPGBA [Zhang *et al.*, 2024], as well as one feature-based method, SPEAR [Ding *et al.*, 2025].

To comprehensively assess the robustness of our approach, we further evaluate the effectiveness of CADA against various backdoor defense strategies, including Prune [Dai *et al.*, 2023], Isolate, and Outlier Detection (OD) [Zhang *et al.*, 2024]. Prune removes low-similarity edges that violate the homophily property in graphs to disrupt trigger connections during both training and inference. Isolate enhances Prune by isolating anomalous nodes linked to these edges. Moreover, OD employs the DOMINANT [Ding *et al.*, 2019] method to filter out nodes with high reconstruction errors from a graph autoencoder to remove anomalies during training.

Implementation Details

Following previous works [Dai *et al.*, 2023; Zhang *et al.*, 2024], we randomly assign the dataset into training, validation, and test sets in the ratio of 8:1:1. For evaluation, we randomly select half of the nodes in the test set as target nodes to calculate the Attack Success Rate (ASR) by embedding triggers. For the remaining clean nodes, we compute two utility metrics: Clean Accuracy (CA), which is evaluated on the original clean graph, and Prediction Accuracy (PAC), which measures the classification accuracy after trigger injection. We use GCN as the primary GNN model and vary the model layers among $\{2, 3, 4\}$ to study the relationship between model depth and collateral damage propagation. We deploy a 2-layer MLP as the adaptive triggers generator. In all experiments, class 0 serves as the target class and the trigger size is limited to three nodes. The attack budget on size of target nodes $\mathcal{V}_p$ is set to $\{40, 40, 90, \text{and } 565\}$ for Cora, CiteSeer, PubMed, and OGB-arXiv, respectively.

5.2 Backdoor Attack Performance (Q1)

We compare CDCA against baseline methods. Notably, as collateral damage spreads in the graph around the target nodes, exhibiting a ripple-like spreading pattern that gradually weakens with the number of hops. Therefore, we pay extra attention to the ASR and PAC of neighboring clean nodes of triggers, which are the most representative metrics for measuring the performance of CDCA. Specifically, we use ASR-K and PAC-K to denote the ASR and PAC of the K-hop neighborhood for the trigger, where the 1-hop neighbors are the target nodes. Detailed results for the 2-layer GCN are summarized in Table 2, while results on OGB-arXiv for the 3-layer and 4-layer models are presented in Fig. 3. Based on these results, we have following observations:

- CDCA consistently outperforms all baselines in minimizing collateral damage by achieving the lowest ASR-2 and highest PAC-2 across all datasets. This confirms that our neighbor contrastive constraints effectively confine the malicious influence to the target nodes.
- Despite the strict constraints on collateral damage, CDCA does not sacrifice attack effectiveness. CDCA

Dataset	Attack	Target Nodes		2-hop Nbr.	
		ASR-1 ↑	PAC-1 ↓	ASR-2 ↓	PAC-2↑
Cora	GTA	98.52	1.48	15.29	74.65
	UGBA	94.67	5.01	37.78	60.24
	DPGBA	98.22	1.56	36.33	61.58
	SPEAR	98.70	1.30	87.87	11.79
	CDCA	**99.56**	**0.44**	**4.95**	**81.60**
PubMed	GTA	94.57	3.23	12.13	80.89
	UGBA	94.21	5.02	13.06	80.56
	DPGBA	97.04	2.96	22.21	74.51
	SPEAR	95.89	4.11	80.07	6.93
	CDCA	**98.76**	**1.24**	**1.77**	**84.66**
CiteSeer	GTA	98.66	1.34	48.45	47.98
	UGBA	**100.0**	**0.00**	40.78	49.84
	DPGBA	98.32	1.68	41.07	39.97
	SPEAR	98.89	1.11	77.53	20.85
	CDCA	**100.0**	**0.00**	**3.59**	**75.37**
OGB-arXiv	GTA	93.35	6.29	11.00	63.37
	UGBA	99.62	0.38	2.55	70.15
	DPGBA	97.08	2.92	5.92	67.28
	SPEAR	96.89	3.11	56.13	33.05
	CDCA	**99.96**	**0.04**	**0.49**	**70.62**

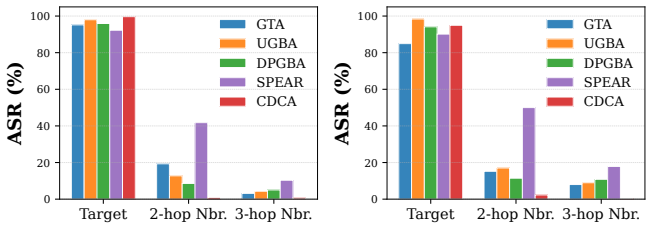
Table 2: Backdoor attack results (ASR% | PAC%) for 2-layer GCN within 2 hops trigger neighbors (Nbr.). The top two performances are highlighted in **bold** and underline.

achieves comparable or even superior ASR on target nodes (ASR-1) compared to state-of-the-art baselines. This validates CDCA's utility in practical scenarios where attackers seek to maximize success rates while minimizing detectable side effects.

- We observe that OGB-arXiv exhibits lower collateral damage compared to other datasets. This phenomenon is likely attributable to its higher graph density and average node degree. The dense connectivity tends to dilute the toxicity from the trigger during the message-passing process, thereby naturally weakening the propagation of malicious signals to the neighborhood.

5.3 Performance Against Defenses (Q2)

We compete CDCA and CDCA-D with defense mechanisms. The pruning threshold for prune and isolate is set to 10% for samples with lowest cosine similarity; and 3% for nodes with highest reconstruction loss for OD. The resulting performances are summarized in Table 3. The results indicate that although the standard CDCA shows a decline in attack performance against defense methods, the enhanced CDCA-D method demonstrates even better stealth across different datasets, surpassing UGBA and DPGBA. Besides, when no defenses are applied, CDCA-D do not exhibit a significant drop in ASR. Overall, these findings suggest that integrating the specific stealth modules into the CDCA framework significantly improves its performance against defense mechanisms while maintaining its effectiveness. This adaptability



(a) 3-layer (b) 4-layer

Figure 3: Backdoor attack results on OGB-arXiv for different layers GCN across 3-hops neighbors.

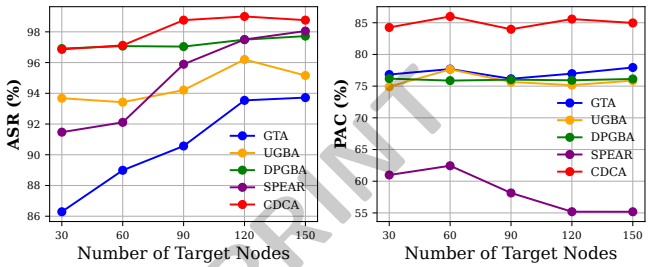


Figure 4: Impacts of attack budget on PubMed.

highlights our method's validity across various defensive environments, making it a robust choice for scenarios.

5.4 Impact of the Attack Budget (Q3)

We vary the number of target nodes as {30, 60, 90, 120, 150} to analyze the impact of the attack budget of CDCA. We report the results on PubMed in Fig. 4. Notably, ASR and PAC specifically refer to the metrics evaluated for target nodes ASR and the clean test set PAC in the following analysis. As expected, ASR shows an upward trend as the number of target nodes increases. Our method keeps outperforming the baselines at varying sizes of target nodes, which shows the effectiveness of CDCA. Moreover, the structural backdoor attacks exhibit slight fluctuations in PCA as the attack budget varies, while CDCA consistently maintains high prediction accuracy and reaches the highest level when the number of target nodes reaches 90, demonstrating intrinsic robustness. In contrast, the feature-based backdoor SPEAR shows a downward trend. Overall, our method achieves satisfactory performance even under a low attack budget.

5.5 Ablation Studies (Q4)

We conduct ablation studies to isolate the effectiveness of specific components within our framework. Specifically, the variant excluding the node selection module is termed CDCA\S, where target nodes are randomly selected. Furthermore, to verify the contributions of the neighborhood contrastive constraint and the probability suppression module, we construct variants by removing these components individually, denoted as CDCA\C and CDCA\P. The results using a 3-layer GCN as the test model are presented in Fig. 5. We observe that CDCA\S suffers from a decline in both ASR and PAC. This shows that the vulnerabilities exploited by the node selection strategy in low-degree nodes generalize effectively

Dataset	Defense	Clean Graph	GTA		UGBA		DPGBA		CDCA		CDCA-D	
Cora	None	82.96	95.98	82.58	94.67	82.22	98.22	82.16	**99.56**	82.70	98.47	**82.96**
	Prune	79.63	17.63	83.06	95.56	81.85	90.88	79.63	36.89	80.74	**97.36**	**84.07**
	Isolate	79.64	35.01	78.64	94.22	78.89	85.33	79.93	36.44	79.63	**95.11**	**80.74**
	OD	83.80	0.04	83.47	0.00	83.62	96.31	81.85	0.00	82.59	**98.43**	**83.96**
PubMed	None	83.90	98.52	82.85	94.21	84.83	97.04	85.01	**98.76**	**85.18**	**98.76**	84.37
	Prune	83.60	28.10	84.05	98.01	83.66	90.18	84.63	20.72	84.12	**99.42**	**85.34**
	Isolate	83.82	12.74	84.11	96.33	83.82	86.76	84.32	19.18	83.82	**98.01**	**84.42**
	OD	83.80	0.03	**85.27**	19.30	84.88	89.82	81.13	47.49	84.68	**90.15**	84.98
CiteSeer	None	73.81	98.40	73.80	**100.0**	**74.10**	98.32	73.19	**100.0**	74.09	98.66	74.10
	Prune	72.51	13.21	71.39	**96.70**	72.89	91.41	72.89	23.44	74.10	92.62	**75.90**
	Isolate	72.69	12.31	73.19	90.54	**75.30**	80.29	73.00	3.17	74.40	**91.95**	75.12
	OD	73.10	0.00	74.10	0.00	73.31	**98.90**	73.12	0.67	71.99	96.85	**75.00**
OGB-arXiv	None	63.12	95.34	65.51	98.62	65.67	98.08	65.65	**99.96**	**65.76**	98.76	65.74
	Prune	62.45	0.01	63.97	93.99	64.46	90.12	62.76	9.59	62.26	**98.97**	**66.45**
	Isolate	63.42	0.00	65.79	99.62	65.67	89.22	66.06	4.80	**66.59**	**99.78**	66.18
	OD	63.31	0.01	65.23	10.13	65.32	92.21	65.06	1.07	64.58	**93.85**	**66.45**

Table 3: Defense performance comparison of target ASR (%) and clean test CA (%). Only clean accuracy is reported for clean graph. The top two performances are highlighted in **bold** and underline.

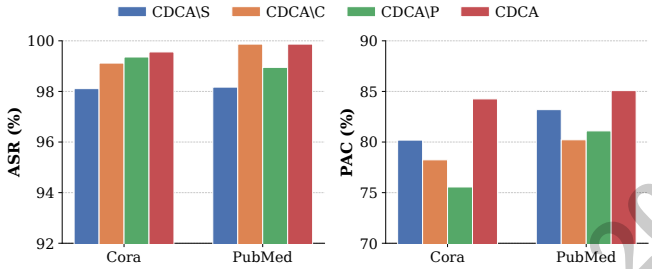
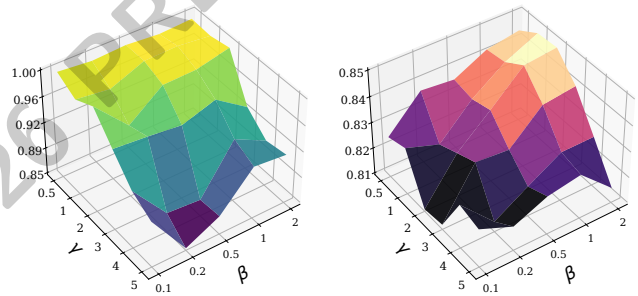


Figure 5: Comparison of CDCA with its variants.



(a) Impact on ASR (b) Impact on PAC

Figure 6: Hyper-parameter Sensitivity Analysis.

during the testing phase, exhibiting high attack efficacy even against high-degree hub nodes. Simultaneously, the drops in CDCA\C and CDCA\P verifies their critical function in confining malicious diffusion to preserve prediction accuracy.

5.6 Hyper-parameter Sensitivity Analysis (Q5)

To investigate the impact of hyperparameters on CDCA, we conduct a sensitivity analysis on β and γ, where β controls the strength of the neighborhood contrastive constraint and γ regulates the intensity of probability suppression. Specifically, we evaluate the performance by varying β within the set $\{0.1, 0.2, 0.5, 1, 2\}$ and γ over the range $\{0.5, 1, 2, 3, 4, 5\}$. Using a 4-layer GCN as the test model on the PubMed dataset, and the other settings align with section 5.1. Fig. 6 reports the impact of β and γ on ASR and PAC, where we observe that a larger β enforces a clearer geometric separation between clean neighbors, thereby directly preserving the semantic integrity of neighbors and boosting PAC. While necessary for stealth, an excessively high γ imposes a strict penalty on target-class features within the neighborhood, which effectively counteracts the trigger's influence and leads to a compromise in ASR.

6 Conclusion

In this paper, we empirically verify that existing graph backdoor attack methods universally suffer from collateral damages. This critical phenomenon stems from the inherent message-passing mechanism of GNNs, which causes malicious trigger signals to diffuse beyond their targets, thereby compromising surrounding clean nodes. To address these issues, we investigate a novel research direction: conducting effective, collateral damage constraint graph backdoor attacks. Specifically, we propose CDCA, which incorporates a strategic node selector to target low-centrality outliers and employs neighbor contrastive constraints and probability suppression to strictly confine malicious influence. Extensive experiments on real-world datasets demonstrate that CDCA effectively minimizes collateral damage across multi-hop neighbors when backdooring various target GNN models, while simultaneously maintaining a high ASR and PAC. Building on our insights into collateral damage, we expect that defense mechanisms can be improved in the future.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No.92370111, 62272340, 62422210, 62276187).

References

[Dai *et al.*, 2023] Enyan Dai, Minhua Lin, Xiang Zhang, and Suhang Wang. Unnoticeable backdoor attacks on graph neural networks. In *Proceedings of the ACM Web Conference*, page 2263–2273, 2023.

[Ding *et al.*, 2019] Kaize Ding, Jundong Li, Rohit Bhanushali, and Huan Liu. Deep anomaly detection on attributed networks. In *Proceedings of the 2019 SIAM International Conference on Data Mining*, pages 594–602, 2019.

[Ding *et al.*, 2025] Yuanhao Ding, Yang Liu, Yugang Ji, Weigao Wen, Qing He, and Xiang Ao. Spear: A structure-preserving manipulation method for graph backdoor attacks. In *Proceedings of the ACM Web Conference*, pages 1237–1247, 2025.

[Fan *et al.*, 2019] Wenqi Fan, Yao Ma, Qing Li, Yuan He, Eric Zhao, Jiliang Tang, and Dawei Yin. Graph neural networks for social recommendation. In *Proceedings of the ACM Web Conference*, pages 417–426, 2019.

[Feng *et al.*, 2024] Bingdao Feng, Di Jin, Xiaobao Wang, Fangyu Cheng, and Siqi Guo. Backdoor attacks on unsupervised graph representation learning. *Neural Networks*, 180:106668, 2024.

[Gilmer *et al.*, 2017] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural message passing for quantum chemistry. In *Proceedings of the 34th International Conference on Machine Learning*, page 1263–1272, 2017.

[Hamilton *et al.*, 2017] William L. Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. *Advances in Neural Information Processing Systems 30*, pages 1025–1035, 2017.

[Hu *et al.*, 2020] 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*, pages 22118–22133, 2020.

[Jin *et al.*, 2025a] Di Jin, Yujun Zhang, Bingdao Feng, Xiaobao Wang, Dongxiao He, and Zhen Wang. Backdoor attack on propagation-based rumor detectors. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 17680–17688, 2025.

[Jin *et al.*, 2025b] Di Jin, Yuxiang Zhang, Bingdao Feng, Xiaobao Wang, Dongxiao He, and Zhen Wang. Losplit: Loss-guided dynamic split for training-time defense against graph backdoor attacks. In *The 39th Annual Conference on Neural Information Processing Systems*, 2025.

[Jin *et al.*, 2026] Di Jin, Bingdao Feng, Xiaobao Wang, Yuxiang Zhang, Zechuan Zhang, Liang Yang, Dongxiao He, and Zhen Wang. Unveiling backdoor propagation in graphs: Neuron-centric defense mechanisms. In *Proceedings of the ACM Web Conference*, pages 1038–1048, 2026.

[Kipf and Welling, 2017] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations*, 2017.

[Li *et al.*, 2023] Rui Li, Xin Yuan, Mohsen Radfar, Peter Marendy, Wei Ni, Terrence J. O'Brien, and Pablo M. Casillas-Espinosa. Graph signal processing, graph neural network and graph learning on biological data: A systematic review. *IEEE Reviews in Biomedical Engineering*, 16:109–135, 2023.

[Li *et al.*, 2025] Runze Li, Di Jin, Xiaobao Wang, Dongxiao He, Bingdao Feng, and Zhen Wang. Single-node trigger backdoor attacks in graph-based recommendation systems. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pages 3072–3080, 2025.

[Liu *et al.*, 2021] Yang Liu, Xiang Ao, Zidi Qin, Jianfeng Chi, Jinghua Feng, Hao Yang, and Qing He. Pick and choose: A gnn-based imbalanced learning approach for fraud detection. In *Proceedings of the ACM Web Conference*, page 3168–3177, 2021.

[Liu *et al.*, 2022] Yang Liu, Xiang Ao, Fuli Feng, and Qing He. Ud-gnn: Uncertainty-aware debiased training on semi-homophilous graphs. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1131–1140, 2022.

[Liu *et al.*, 2024] Ziyang Liu, Chaokun Wang, Hao Feng, and Ziyang Chen. Efficient unsupervised graph embedding with attributed graph reduction and dual-level loss. *IEEE Transactions on Knowledge and Data Engineering*, 36(12):8120–8134, 2024.

[Lyu *et al.*, 2024] Xiaoting Lyu, Yufei Han, Wei Wang, Hangwei Qian, Ivor Tsang, and Xiangliang Zhang. Cross-context backdoor attacks against graph prompt learning. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, page 2094–2105, 2024.

[Sen *et al.*, 2008] 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.

[Sun *et al.*, 2020] Yiwei Sun, Suhang Wang, Xianfeng Tang, Tsung-Yu Hsieh, and Vasant Honavar. Adversarial attacks on graph neural networks via node injections: A hierarchical reinforcement learning approach. In *Proceedings of the ACM Web Conference*, pages 673–683, 2020.

[Veličković *et al.*, 2018] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *6th International Conference on Learning Representations*, 2018.

[Wu *et al.*, 2020] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S Yu. A com-

prehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, 32(1):4–24, 2020.

[Xi *et al.*, 2021] Zhaohan Xi, Ren Pang, Shouling Ji, and Ting Wang. Graph backdoor. In *30th USENIX Security Symposium*, pages 1523–1540, 2021.

[Xia *et al.*, 2025] Hui Xia, Xiangwei Zhao, Rui Zhang, Shuo Xu, and Luming Wang. Clean-label graph backdoor attack in the node classification task. In *Proceedings of the AAAI Conference on Artificial Intelligence*, page 21626–21634, 2025.

[Xu and Picek, 2023] Jing Xu and Stjepan Picek. Poster: Multi-target & multi-trigger backdoor attacks on graph neural networks. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, page 3570–3572, 2023.

[Xu *et al.*, 2019] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *7th International Conference on Learning Representations*, 2019.

[Yan *et al.*, 2025] Fengyu Yan, Xiaobao Wang, Dongxiao He, Longbiao Wang, Jianwu Dang, and Di Jin. Hetergp: bridging heterogeneity in graph neural networks with multi-view prompting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 21895–21903, 2025.

[Yang *et al.*, 2023] Yachao Yang, Yanfeng Sun, Fujiao Ju, Shaofan Wang, Junbin Gao, and Baocai Yin. Multi-graph fusion graph convolutional networks with pseudo-label supervision. *Neural Networks*, 158:305–317, 2023.

[Zhang and Zitnik, 2020] Xiang Zhang and Marinka Zitnik. Gnnguard: Defending graph neural networks against adversarial attacks. *Advances in Neural Information Processing Systems 33*, pages 9263–9275, 2020.

[Zhang *et al.*, 2021] Zaixi Zhang, Jinyuan Jia, Binghui Wang, and Neil Zhenqiang Gong. Backdoor attacks to graph neural networks. In *The 26th Symposium on Access Control Models and Technologies*, pages 15–26, 2021.

[Zhang *et al.*, 2023] Hangfan Zhang, Jinghui Chen, Lu Lin, Jinyuan Jia, and Dinghao Wu. Graph contrastive backdoor attacks. In *International Conference on Machine Learning*, pages 40888–40910, 2023.

[Zhang *et al.*, 2024] Zhiwei Zhang, Minhua Lin, Enyan Dai, and Suhang Wang. Rethinking graph backdoor attacks: A distribution-preserving perspective. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, page 4386–4397, 2024.

[Zhang *et al.*, 2025] Zhiwei Zhang, Minhua Lin, Junjie Xu, Zongyu Wu, Enyan Dai, and Suhang Wang. Robustness inspired graph backdoor defense. In *The 13th International Conference on Learning Representations, Singapore, April 24-28*, 2025.

[Zügner *et al.*, 2020] Daniel Zügner, Oliver Borchert, Amir Akbarnejad, and Stephan Günnemann. Adversarial attacks on graph neural networks: Perturbations and their patterns. *ACM Transactions on Knowledge Discovery from Data*, 14(5):1–31, 2020.