Towards Inductive Robustness: Distilling and Fostering Wave-Induced Resonance in Transductive GCNs against Graph Adversarial Attacks

Ao Liu[1], Wenshan Li[2*], Tao Li[1], Beibei Li[1], Hanyuan Huang[1], Pan Zhou[3]

[1]Sichuan University
[2]Chengdu University of Information Technology
[3]Huazhong University of Science and Technology
{liuao, huanghanyuan}@stu.scu.edu.cn, helenali@cuit.edu.cn, {litao, libeibei}scu.edu.cn, panzhou@hust.edu.cn

Abstract

Graph neural networks (GNNs) have recently been shown to be vulnerable to adversarial attacks, where slight perturbations in the graph structure can lead to erroneous predictions. However, current robust models for defending against such attacks inherit the transductive limitations of graph convolutional networks (GCNs). As a result, they are constrained by fixed structures and do not naturally generalize to unseen nodes. Here, we discover that transductive GCNs inherently possess a distillable robustness, achieved through a wave-induced resonance process. Based on this, we foster this resonance to facilitate inductive and robust learning. Specifically, we first prove that the signal formed by GCN-driven message passing (MP) is equivalent to the edge-based Laplacian wave, where, within a wave system, resonance can naturally emerge between the signal and its transmitting medium. This resonance provides inherent resistance to malicious perturbations inflicted on the signal system. We then prove that merely three MP iterations within GCNs can induce signal resonance between nodes and edges, manifesting as a coupling between nodes and their distillable surrounding local subgraph. Consequently, we present Graph Resonance-fostering Network (GRN) to foster this resonance via learning node representations from their distilled resonating subgraphs. By capturing the edge-transmitted signals within this subgraph and integrating them with the node signal, GRN embeds these combined signals into the central node's representation. This node-wise embedding approach allows for generalization to unseen nodes. We validate our theoretical findings with experiments, and demonstrate that GRN generalizes robustness to unseen nodes, whilst maintaining state-of-the-art classification accuracy on perturbed graphs. *Appendices can be found on* `arXiv` *version*: https://arxiv.org/abs/2312.08651

Introduction

In recent years, graph neural networks (GNNs), through the capabilities afforded by inductive learning, have emerged as the most potent instruments for node classification tasks. Nevertheless, earlier transductive models, such as graph convolutional networks (GCNs) (Kipf and Welling 2017), have inadvertently introduced vulnerabilities to adversarial attacks within the GNN framework. It has been observed that perturbed graphs derived from GCNs serving as surrogate models have the potential to compromise the outputs of inductive GNNs when transferred. In real-world applications, where trust and accuracy are non-negotiable (Chen et al. 2021; Nadal et al. 2021; Zhao et al. 2022; Berberidis and Giannakis 2019; Xiao, Chen, and Shi 2019), such vulnerabilities can significantly jeopardize public trust (Kreps and Kriner 2020), distort human decision-making (Walt, Jack, and Christof 2019), and threaten human well-being (Samuel et al. 2019). Addressing the vulnerabilities introduced by transductive GCNs into the GNNs' community is of paramount importance.

Distinct from discrete feature data like images or text, graph data comprises a connected set of features through its topological structure. This interconnectedness naturally encourages the adoption of a global input-output mechanism to establish a learning channel from features to labels, a paradigm referred as transductive learning (Kipf and Welling 2017; Defferrard, Bresson, and Vandergheynst 2016; Bruna et al. 2013), with GCNs epitomizing this approach. This very transductive nature of GCNs offers adversaries an ideal environment for launching attacks (Liu et al. 2022). Leveraging this global input-output pattern, given sufficient computation, adversaries can invariably devise perturbations that are both concealment and effective (Sun et al. 2022). Given that adversaries exploit vulnerabilities inherent to transductive models to compromise the GNNs' communities, the formulation of a more robust transductive model has ascended as the prevailing defensive approach.

To defense adversarial attacks, early research predominantly sought to fortify GCN's tolerance to perturbations by adversarial training through random edge drops (Dai et al. 2018). Recently, a shift towards self-supervised training methods has been observed. These techniques sidestep the trap set by adversaries, which bait the model into misclassifying specific inputs. Instead of singularly focusing on enhancing the model's robustness to a given label space, they aim to expand the GCN's overall robustness to potential perturbed graphs. Key representatives of these research endeavors include: (1) In RGCN (Zhu et al. 2019), the Gaussian distributions are employed to replace the node hidden representations across each GCN layer, aiming to mitigate the adversarial modifications' impact. (2) By introducing a singular value decomposition (SVD) filter before the GCN

*Corresponding author

processing, GCN-SVD (Entezari et al. 2020) is designed to discard adversarial edges from the training dataset. (3) STABLE (Li et al. 2022) introduces enhancements in GCN's forward propagation by incorporating functions that sporadically reinstate edges which were approximately removed. (4) EGNN (Liu et al. 2021) leverages graph smoothing techniques to confine the permutation setting space, effectively excluding the majority of non-smooth permutations.

However, current research, aiming to improve GCN-based models into a robust transductive variant against attacks, inadvertently carries over transductive-introduced weaknesses (Hamilton, Ying, and Leskovec 2017). Specifically, these models can't handle unseen nodes and are limited to fixed structures, lacking generalization. This restricts their applicability. If adversaries slightly adjust tactics, defenders must retrain their models for safety. The cause is that GCNs' vulnerabilities are inherent. To enhance their robustness, these vulnerabilities require targeted solutions. Deviating from the context of GCNs could hinder a thorough analysis of attack mechanisms. This, in turn, would obstruct the transition from transductive robust models to inductive ones. Until we harness GCN's inherent robustness for inductive models, we will be stuck in a cycle of constantly refining transductive ones to address vulnerabilities.

In addressing this conundrum, our exploration unveiled an intriguing intrinsic source of robustness within the GCN itself. Without resorting to additional designs, merely deepening the standard 2-layer GCN to a 3-layer structure endows it with an innate (albeit partial) robustness. Importantly, the mechanism underpinning this robustness can be distilled. By purposefully fostering this intrinsic mechanism, it has paved the way for us to architect a robust inductive model. Employing this approach serves a dual purpose: On one hand, it facilitates a precision-oriented confrontation against the perturbations devised specifically by adversaries for transductive structures, ensuring the efficacy of our defensive strategies. On the other hand, it enables us to integrate this robustness mechanism into inductive frameworks, thereby achieving a seamless melding of inductiveness and robustness.

Specifically, we demonstrate that the vibrations of node signals within the GCN-driven message passing (MP) are equivalent to the edge-based waves, formulated by wave equations (Friedman and Tillich 2004; Shatah and Struwe 1993). Given this equivalence, it follows that GCNs inherently possess the potential for resonance (Kovalyov 1989), allowing them to harness the natural advantages of waves in defending against perturbations (Blas and Jenkins 2022). Then, we introduce a mathematical definition for the intensity of such resonance in GCNs. This definition, which outlines the scope and weights of a node's connections to its neighbors, concurrently adheres to four principles: universality, adaptation via MP, node-independence, and topological correlation. Subsequently, we demonstrated that for 3+ layer GCNs, an invariant mapping exists, translating GCNs' outputs into resonance intensity, manifesting as nodes capturing their surrounding local weighted structure.

Informed by these insights, we introduce the Graph Resonance-fostering Network (GRN) for inductive learning. The core of GRN is that it distills the structure resonating with nodes as local resonance subgraphs. Then, within this subgraph, GRN fosters the resonance by embedding both the node's signals and the signals transmitted through edges as central node's representation. This embedding approach is generalizable across graph structures. If the surrounding topology of a node (with unseen ones) can be clearly determined to distill the local resonance subgraph, robust and inductive graph learning is achieved. Our contributions are:

- We propose the first inductive and robust GNN.
- We prove that a 3-layer GCN inherently possesses an distillable robustness.
- We prove the equivalence between GCN-driven signal vibrations and edge-based waves.

Preliminaries

Notations We consider connected graphs $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ consisting $N = |\mathcal{V}|$ nodes. Let $\mathbf{A} \in \{0,1\}^{N\times N}$ be the adjacency matrix. Let generic symbol $\mathbf{L}$ be the Laplacian in its broadest sense. The feature and one-hot label matrix are $\mathbf{Z} \in \mathbb{R}^{N\times d_0}$ and $\mathbf{Y} \in \mathbb{R}^{N\times d_L}$ respectively. The edge connected nodes v_i and v_j is written as (v_i, v_j) or (v_j, v_i). The neighborhood $\mathcal{N}_i$ of a node v_i consists of all nodes v_j for which $(v_i, v_j) \in \mathcal{E}$. Let $\deg_i$ be the degree of node v_i. The feature vector and one-hot label of node v_i are $\mathbf{z}_i$ and $\mathbf{y}_i$.

GCN Under the topology $\mathbf{L}$, with $\mathbf{Z}$ as the input, the output at the k-th layer of a GCN is denoted as $\mathcal{M}(\mathbf{Z}, k; \mathbf{L})$. The k-th parameter matrix of $\mathcal{M}$ is $\mathbf{W}^{(k)} \in \mathbb{R}^{d_k\times d_{k+1}}$. $\mathbf{Z}^{(k)} = [\mathbf{z}_1^{(k)}, \ldots, \mathbf{z}_N^{(k)}]$ denotes the features in k-th MP.

Wave Equation The edge-based wave equation introduces a relationship between a graph-based signal $g = \textsc{Wave}(\mathbf{Z}, k; \mathbf{L})$ and its topological structure. Let c be a constant, it is defined as $\frac{\partial^2 g}{\partial k^2} = -\mathbf{L}^k g \cdot c$ (Friedman and Tillich 2004). Herein g can be instantiated as any discernible signal.

3-layer GCN Possesses Adversarial Robustness via Wave-induced Resonance

Equivalence of GCN-driven MP and Wave Equation Here we demonstrate that the signal vibrations driven by GCNs, are equivalent to waves on graph topologies and can be characterized by nonlinear wave equations.

Theorem 1. *Let $\mathcal{M}(\mathbf{Z}, k; \mathbf{L})$ and $\textsc{Wave}(\mathbf{Z}, k; \mathbf{L})$ denote the signals of the k-th MP and k-th wave respectively, under the topological structure represented by $\mathbf{L}$. It is established that for the given k and $\mathcal{M}(\cdot)$, there exists $\textsc{Wave}(\cdot)$ satisfies $\mathcal{M}(\mathbf{Z}, k; \mathbf{L}) = \textsc{Wave}(\mathbf{Z}, k; \mathbf{L}), \forall \mathbf{L} \in \widehat{\mathbf{L}}$, where $\widehat{\mathbf{L}}$ are the Laplacian matrices of all attribute graphs.*

This study draws an analogy between the node signals in GCN-driven MP and waves, considering edges as the transmission medium. Research indicates that in systems producing waves, resonance can arise between waves and their medium (Ahmad et al. 2023; Bykov, Bezus, and Doskolovich 2019). Building on this understanding, we can reaffirm our empirical observations about the GCN training pattern: in non-adversarial contexts, GCNs converge to

the most natural, or congruent with ground truth, signal MP paradigm during training. Under this premise, the messages transmitted by nodes and edges in the graph manifest a natural coupling state, maintaining a benign mapping relationship $\mathcal{M} : \mathcal{G} \rightarrow \mathbf{Y}$. The key to optimizing $\mathcal{M}$ is the resonance between node signals $\mathbf{Z}^{(\ell)}$ and edge signals $\mathbf{E}^{(\ell)}$. In adversarial situations, adversaries manipulate node signals by rewiring edges, which inadvertently induces unnatural, i.e., noncongruent with ground truth, MP patterns. Under this scheme, the benign resonance is disrupted, resulting in a malignant mapping relationship $\mathcal{M} : \mathcal{G}' \rightarrow \mathbf{Y}'$, where $\mathcal{G}'$ and $\mathbf{Y}'$ is the perturbed graph and label, respectively.

Mathematical Definition of Resonance in GCN Maintaining benign resonance becomes an intuitive defensive mechanism as it intrinsically resists unnatural perturbations. To actualize control over this resonance, thereby purposefully fostering resonance within the GCN, we subsequently delineate a detailed definition of this resonance. Thus, this definition should comply with the following conditions: 1) Every node within a graph should possess a computable resonance intensity, 2) the resonance intensity of all nodes should evolve in accordance with MP, 3) each node should maintain an independent resonance intensity, and 4) the stronger a node's connection to its surrounding topology, the greater its perceived resonance intensity.

To devise a methodology compliant with the desired conditions, we consider node v_i and utilize its latent representation (Kula 2015) $\bar{z}_i^{(k)} = \sum_j \mathbf{z}_{i,j}^{(k)}$ to quantify the intensity of the node signals. Furthermore, we use T_i, the count of edges among nodes in $\mathcal{N}_i$, to measure the connectivity strength specific to the edges at the given nodes. Then, we use the total number $p_i = \sum_j \mathbf{A}_{i,j}^2$ of walks of length 2 originating from v_i to any node in $\mathcal{G}$, to quantity the magnitude of connectivity density that v_i exhibits in the structure.

Accordingly, we propose the following definition to quantify the resonance intensity at node v_i:

Definition 1. *The resonance intensity of v_i on k-th MP is*

$$R(v_i; k) \overset{def.}{=} \bar{z}_i^{(k)} T_i + 2p_i + 8\deg_i. \tag{1}$$

The unique of defining resonance intensity can be articulated as follows: it not only allows for an interpretable quantification of the resonance on different nodes, but it is also directly observable within MP. This implies that under such a definition, the resonance intensity of any node at any given MP epoch on a graph can be independently calculated, obviating the need for the GCN computational paradigm.

Resonance arises in 3rd MP Definition 1 facilitates the quantification of resonance for any signal function on any graph, irrespective of whether or not it is driven by GCN. Nonetheless, an intriguing finding has been proven: the wave system constructed by GCN inherently and involuntarily arises resonance, which is outlined in the theorem:

Theorem 2. *Let $\bar{z}_i^{(k)}$ be the latent signal formed by GCN-driven MP, we have:*

$$R(v_i; k) \propto 64\bar{z}_i^{(k+3)} - 32. \tag{2}$$

Theorem 2 unveils an intriguing phenomenon: for $k \geq 3$, there subsists an invariant mapping, which transposes the GCN-driven signal into a resonance intensity that bears no correlation with the GCN paradigm. Given that Definition 1 has established the resonance intensity as a measure of the coupling strength between nodes and structure within the graph, we can thus characterize it as the degree of coupling. Consequently, it can be asserted that prior to 3rd MP iteration, the GCN appears to have yet to delve into the coupling paradigm between nodes and structure within the graph. However, subsequent to the 3rd MP, due to the persistent presence of the invariant mapping, it can be construed that the GCN has fortuitously assimilated the coupling paradigm within the graph during the 3rd MP, and perpetuates this paradigm into subsequent MPs.

Vast Perturbation Search Space of 3-layer GCN In light of the current absence of an effective method for quantifying the combined adversarial robustness of a specific graph and a GCN learning from said graph, we propose an intuitive approach. For a graph $\mathcal{G}$, comprised of $|\mathcal{V}|$ edges and represented by the adjacency matrix $\mathbf{A}$, and a GCN $\mathcal{M}$ with K layers, where the perturbation budget is denoted as r, the number of matrix multiplication-based forward propagations required by the attack model can be construed as the highest attack cost. In this context, the number of subgraphs is independent of node features, hence we employ $\mathbf{A}$ as the independent variable for the attack cost function, denoted as $\mathrm{Cost}(\mathbf{A}, r, K)$. We then present the following theorem:

Theorem 3. *For any specified graph with a node set $\mathcal{V}$ and an adjacency matrix $\mathbf{A}$, in conjunction with a K-layer GCN, and a maximum perturbation r, the following holds:*

$$\mathrm{Cost}(\mathbf{A}, r, K) \leq \begin{cases} C(|\mathcal{V}|, r), & \text{if } K < 3 \\ (K-1)C\left(\frac{|\mathcal{V}|^{K-1}}{2}, r\right), & \text{otherwise}, \end{cases} \tag{3}$$

where $C(\cdot, \cdot)$ denotes the number of combinations.

It's revealed that adversaries face the same computational cost for matrix multiplication-based forward propagations when K = 1 or 2. However, for $K \geq 3$, the cost dramatically increases, largely due to $C(\frac{|\mathcal{V}|^{K-1}}{2}, r)$. As an example, with the Cora dataset ($|\mathcal{V}| = 5429$) and a 1% perturbation rate ($r = 54$), the cost for K = 3 becomes exponentially larger. Thus, attacking a 3-layer GCN presents a vast search space for adversaries. This insight extends Theorem 2's real-world applications and our previous findings: a 3-layer GCN can naturally create resonance robustness. With our defined resonance, we can further boost this robustness proposefully.

Graph Resonance-fostering Network

Principle Overview We employ GRN to enhance the resonance of the GCN. The underlying concept of the GRN is articulated as follows. Definition 1 exhibits that for a node v_i, there exists a local graph structure that resonates, known as the local resonance subgraph (LRS) for node v_i, denoted as $G_i = (\mathcal{V}_{G_i}, \mathcal{E}_{G_i})$, used to represent the maximal subgraph structure that node v_i can capture. During end-to-end training, both the node signals $\mathbf{Z}^{(\ell)}$, and the signals transmitted through edges $\mathbf{E}^{(\ell)}$ concurrently vibrate within the LRS.

Consequently, for v_i, if a learnable parameter $\mathbf{W}^{(\ell)}$ capable of jointly embedding MP's result $\mathbf{A}_{G_i}\mathbf{Z}^{(\ell)}_{G_i}$, and $\mathbf{E}^{(\ell)}_{G_i}$, into v_i's output representation, this aggregation intentionally accomplishes a learnable resonance, generating a local-level embedding. This identical aggregate pattern is applied across all nodes to facilitate a mapping, thereby achieving a global-level forward propagation within the GRN.

In summary, a single forward propagation of the GRN is:

$$\mathbf{z}_i^{(\ell+1)} = \sigma\big(\text{MEAN}\big(\text{CONCAT}\big(\mathbf{A}_{G_i}\mathbf{Z}^{(\ell)}_{G_i}, \mathbf{E}^{(\ell)}_{G_i}\big)\mathbf{W}^{(\ell)}\big)\big). \quad (4)$$

Next, we provide explicit definitions for G_i and $\mathbf{E}^{\ell}_{G_i}$.

G_i: Local Resonance Subgraph As per Definition 1, LRS comprises three components: 1) edges formed amongst all first-order neighbors, as counted by T_i, with these edges' weights equal 1; 2) edges formed between it and all 2nd-(inevitably includes 1st-) order neighbors, as counted by p_i, with these edges' weights equal 2; 3) edges between it and all first-order neighbors, as counted by $\deg_i$, with these edges' weights equal 8. Consequently, the LRS can be viewed as a weighted graph, in which the weights of edges serve as attention for the joint combination of $\mathbf{Z}^{(\ell)}$ and $\mathbf{E}^{(\ell)}$. An illustrative example of the LRS is presented in Figure 1.

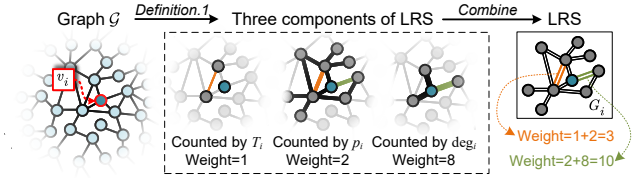


Figure 1: An illustration of local resonance subgraph.

$\mathbf{E}^{(\ell)}_{G_i}$: Edge-transmitted Signals In the MP driven by adjacency matrices, only signals at the nodes are observable, while the signals transmitted across each edge remain imperceptible. To ascertain the quantified signals on every specific edge within G_i, we first obtain the global edge-transmitted signals $\mathbf{E}^{\ell}_{\mathcal{G}}$. Then, $\mathbf{E}^{\ell}_{G_i}$ is subsequently derived through a sampling procedure on $\mathbf{E}^{\ell}_{\mathcal{G}}$ using the edge indices within G_i.

Specifically, within $\mathbf{E}^{\ell}_{\mathcal{G}}$, the edge-transmitted signals on (v_j, v_k) are denoted as $\mathbf{e}^{\ell}_{j,k}$. For $\ell > 0$, $\mathbf{e}^{\ell}_{j,k}$ is defined via a sequential procedure: 1) The edge (v_j, v_k) in $\mathcal{G}$ is deleted, producing a new graph $\mathcal{G}^{j,k}$ with its adjacency matrix $\mathbf{A}_{\mathcal{G}^{j,k}}$. 2) A new forward propagation is executed in the same layer on $\mathcal{G}^{j,k}$, obtaining a feature matrix $\mathbf{Z}^{(\ell)}_{\mathcal{G}^{j,k}}$. This matrix does not contain any messages transmitted through the edge (v_j, v_k). Consequently,

$$\mathbf{Z}^{(\ell)}_{\mathcal{G}^{j,k}} = \mathbf{A}_{\mathcal{G}^{j,k}}\mathbf{Z}^{(\ell-1)}\mathbf{W}^{(\ell-1)}. \quad (5)$$

The feature of node v_j in $\mathbf{Z}^{(\ell)}_{\mathcal{G}^{j,k}}$ denoted as $\mathbf{z}^{(\ell)}_{\mathcal{G}^{j,k},j}$, is obtained. 3) In $\mathcal{G}^{j,k}$, there is no edge between the nodes (v_j, v_k). Hence, the feature transmitted from node v_k to v_j (i.e., $\mathbf{e}^{(\ell)}_{j,k}$) is calculated by subtracting the feature obtained through the re-propagation on $\mathcal{G}^{j,k}$ (i.e., $\mathbf{z}^{(\ell)}_{G_i^{j,k},j}$) from the original feature (i.e., $\mathbf{z}^{(\ell)}$). Similarly, the signal transmitted through the pair (v_j, v_k) could be interpreted as the average of the mutually transmitted signals. At $\ell = 0$, since MP has not been initiated, $\mathbf{e}^{(\ell)}_{j,k}$ would ideally be 0. For end-to-end training, it is defined as a random infinitesimal value. In conclusion, $\mathbf{E}^{(\ell)}_{G_i}$ is determined as

$$\mathbf{E}^{(\ell)}_{G_i} = \text{CONCAT}\left(\left\{\mathbf{e}^{(\ell)}_{j,k} : v_j, v_k \in G_i\right\}\right),$$
$$\text{s.t. } \mathbf{e}^{(\ell)}_{j,k} = \begin{cases} \mathbf{z}^{(\ell)} - \frac{\mathbf{z}^{(\ell)}_{\mathcal{G}^{j,k},j} + \mathbf{z}^{(\ell)}_{\mathcal{G}^{j,k},k}}{2}, \textit{if } \ell > 0 \\ \epsilon, \quad \text{where } \epsilon \sim U(0, 1\times 10^{-7}), \textit{if } \ell = 0 \end{cases}. \quad (6)$$

Figure 2 illustrates the computation of $\mathbf{e}^{(\ell)}_{j,k}$.

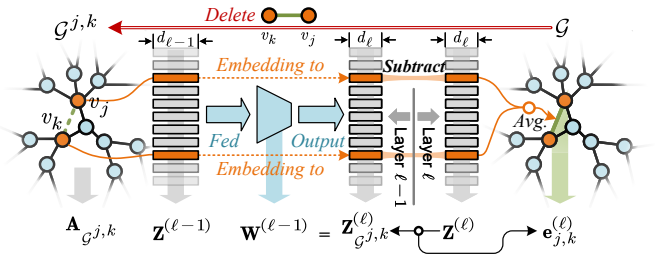


Figure 2: Schematic diagram illustrating the computation of the edge-transmitted signal between nodes v_j and v_k.

Simplifying the Computational Overhead of $\mathbf{E}^{(\ell)}_{G_i}$ Equation (5) explicates the method of re-propagation on $\mathcal{G}_{j,k}$. Given that there are $|\mathcal{V}|$ ways to choose (v_j, v_k), it necessitates the computation of $|\mathcal{V}|$ matrix multiplications (where $\mathbf{Z}^{(\ell-1)}_{\mathbf{W}} = \mathbf{Z}^{(\ell-1)}\mathbf{W}^{(\ell-1)}$ remains the same for all (v_j, v_k) selections and can be considered a constant matrix), thereby constituting the primary computational cost of $\mathbf{E}^{(\ell)}_{G_i}$. Here, we provide a computational method equivalent to Equation (5), reducing the $|\mathcal{V}|$ times to once.

Proposition 1. *By indexing and rearranging $\mathbf{Z}^{(\ell-1)}_{\mathbf{W}}$ by rows j and k to obtain a matrix $Q(\mathbf{Z}^{(\ell-1)}_{\mathbf{W}}; j, k) \in \mathbb{R}^{N\times d_{\ell-1}}$,*

$$\mathbf{Z}^{(\ell)}_{\mathcal{G}^{j,k}} = \mathbf{A}\mathbf{Z}^{(\ell-1)}_{\mathbf{W}} - Q\left(\mathbf{Z}^{(\ell-1)}_{\mathbf{W}}; j, k\right). \quad (7)$$

Evidently, a single matrix multiplication, i.e., $\mathbf{A}_G\mathbf{Z}^{(\ell-1)}_{\mathbf{W}}$, is sufficient to iterate over all (v_j, v_k) and yield the results.

Learning the Parameters Each layer of GRN only contains trainable parameters $\mathbf{W}^{(\ell)}$, and each has a distinct output representation $\mathbf{Z}^{(\ell)}$. Thus, in accordance with the requirements of the downstream task, GRN can accommodate either supervised or unsupervised loss functions, thereby tuning their weight matrices. Specifically, we denote the discrepancy function as $D(\cdot,\cdot)$. In semi-supervised scenarios, the loss function is $J_s(\mathbf{z}^{(K)}_i) = D(\mathbf{z}^{(K)}_i, \mathbf{y}_i)$; in unsupervised scenarios (Müller 2023), $J_u(\mathbf{z}^{(K)}_i) = D(\mathbf{z}^{(K)}_i, \{\mathbf{y}_j : v_j \in \mathcal{N}_i\})$. Depending on the downstream applications, $D(\cdot,\cdot)$ can take various forms, such as cross-entropy, etc. The general workflow of GRN is illustrated in Figure 3.

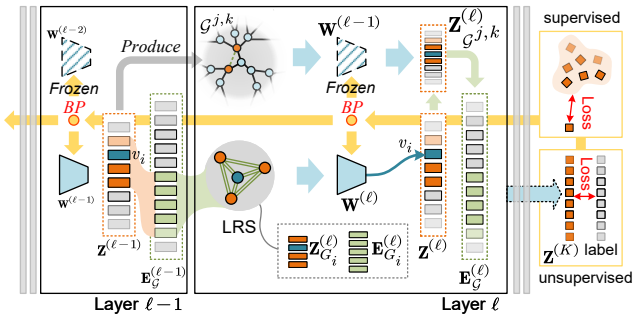


Figure 3: The general workflow of GRN.

Experiments

Datasets Our findings are evaluated on five real-world datasets widely used for studying graph adversarial attacks (Sun et al. 2020; Liu et al. 2022; Zhu et al. 2019; Entezari et al. 2020; Li et al. 2022), including Cora, Citeseer, Polblogs, and Pubmed.

Baselines *Comparison defending models.* We compare GRN with other defending models including: 1) RGCN which leverages the Gaussian distributions for node representations to amortize the effects of adversarial attacks, 2) GNN-SVD which is applied to a low-rank approximation of the adjacency matrix obtained by truncated SVD, 3) Pro-GNN (Jin et al. 2020) which can learn a robust GNN by the intrinsic properties of nodes, 4) Jaccard (Wu et al. 2019) which defends attacks based on the measured Jaccard similarity score, 5) EGNN (Liu et al. 2021) which filters out perturbations by l_1- and l_2-based graph smoothing. *Attack methods.* The experiments are designed under the following attack strategies: 1) Metattack (Zügner and Günnemann 2018), a meta-learning based attack, 2) CLGA (Sixiao et al. 2022), an unsupervised attack, 3) RL-S2V (Dai et al. 2018), a reinforcement learning based attack.

Pinpointing the Layer of Resonance In Theorem 2, we establish an equivalence relation between the k-th and the $k+3$-th layer's output latent representations, as derived from Equations (1) and (2). This elucidates that when $k=0$, the 3-th layer involuntarily captures local structures, thereby inducing resonance. To facilitate experimental variable control, we first demonstrate the equivalence relation under varying "gap layer numbers" (denoted as k_{gap}). If the equivalence between Equations (1) and (2) only holds when $k_{gap} \geq 3$, it substantiates the validity of Theorem 2. Specifically, we first train a 5-layer GCN, then obtain the resonance intensity denoted as $R_{def}(k) = \bar{z}_i^{(k)} T_i + 2p_i + 8\deg_i$, and the actual observed signal denoted as $R_{real}(k+k_{gap}) = 64\bar{z}_i^{(k+k_{gap})} - 32$, for each epoch. Given these observational variables, we delineate their transformations over the learning process using lists $\{R_{def}(k)\}$ and $\{R_{real}(k+k_{gap})\}$ respectively. Each list chronicles its corresponding variable's fluctuations across all epochs. Subsequently, we standardize (using the standardize function $\mathrm{STD}(\cdot)$) the sequences under different k_{gap} and calculate the absolute difference to obtain a difference sequence:

$$\mathbf{d}_{k,k_{gap}} = |\mathrm{STD}(\{R_{def}(k)\}) - \mathrm{STD}(\{R_{real}(k+k_{gap})\})|. \quad (8)$$

The parameter $\mathbf{d}_{k,k_{gap}}$, serving as an indicator variable, accurately encapsulates the discrepancy between $R_{def}(k)$ and $R_{real}(k+k_{gap})$. The experimental results are illustrated in Figure 4. Owing to the large number of nodes, we display the mean value (central line) and standard deviation (shadow areas) of all nodes. As epochs progress, $\mathbf{d}_{0,3}$ gradually converges to zero. After the initial several epochs, it significantly diverges from $\mathbf{d}_{0,1}$ and $\mathbf{d}_{0,2}$. This validates the intriguing phenomenon mentioned in Theorem 2: a correlation has been established between the signal at the 3-th layer and the graph's initial signal and structure. Subsequent experimental results echo the aforementioned findings, thereby affirming the correctness of Theorem 2 when $k > 0$.

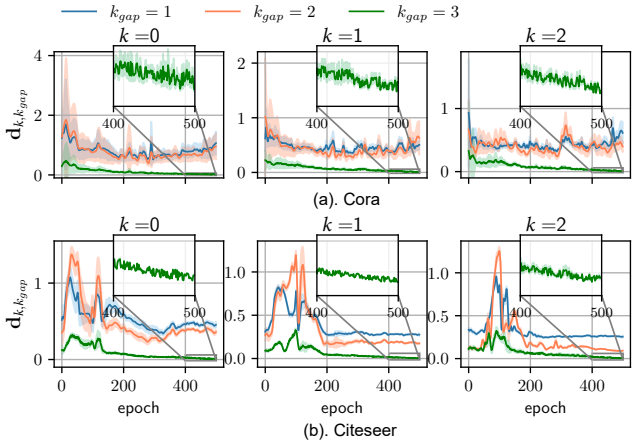
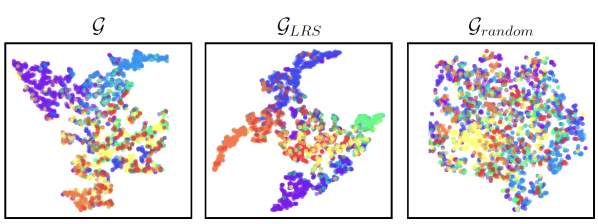
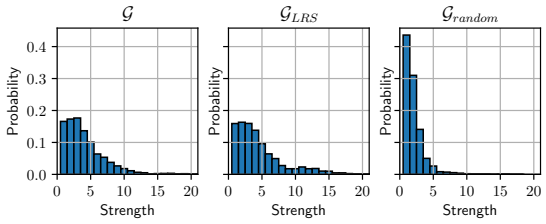


Figure 4: Values of $\mathbf{d}_{k,k_{gap}}$ under different k and k_{gap} settings. The $\mathbf{d}_{k,k_{gap}}$ encapsulates the resonance situation between the k-th layer and the $k+k_{gap}$-th layer. A smaller value indicates a stronger resonance.

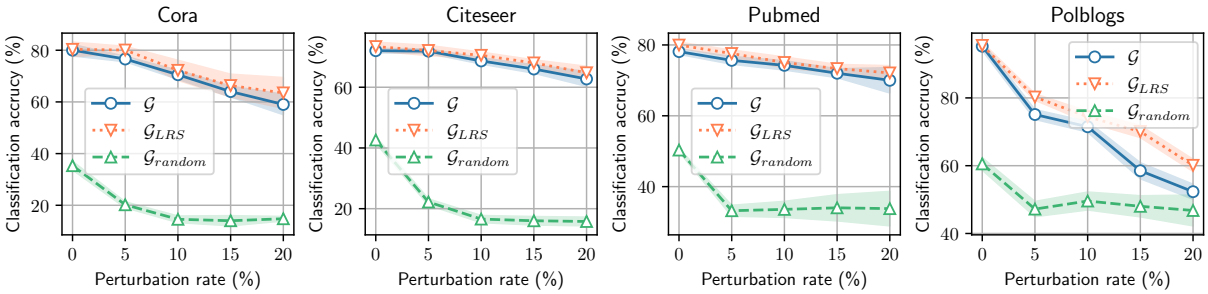
Attack Success Rate Cliff-like declines on 3-layer GCN Intuitively, the complexity of an attack tends to increase with the number of GCNs' layer. Observing the pattern of attack success rate (ASR) declines as the number of GCN layers increases helps validate our claim that the 3-layer GCN, derived from resonance, can significantly enhance robustness. Specifically, we start by initializing 10 GCNs with the number of layers ranging from 1 to 10. Next, we conduct experiments on 4 datasets using 3 typical attacks, setting the perturbation rate uniformly at 20%. We then train a surrogate model for each GCN separately, placing perturbations in the dataset, and clearing these perturbations after each attack. We repeat each attack five times and report the average ASR accuracy (depicted by the lines) and variation range (represented by the shaded areas). The results, as shown in Figure 6, clearly reveal a steep drop in ASR at the 3-layer GCN. However, further layer addition seems unable to significantly reduce the ASR, as the additional layers maintain the same resonance pattern as the 3-layer GCN to achieve adversarial robustness. These findings articulate the concept



(a) Visualizations of embedding of $\mathcal{G}$, $\mathcal{G}_{LRS}$ and $\mathcal{G}_{random}$.



(b) Strength distribution of $\mathcal{G}$, $\mathcal{G}_{LRS}$ and $\mathcal{G}_{random}$.



(c) Classification accuracy of $\mathcal{G}$, $\mathcal{G}_{LRS}$ and $\mathcal{G}_{random}$ using $\mathcal{M}_{\mathcal{G}}$ under different perturbation rates.

Figure 5: Experimental results of LRS-constructed graph $\mathcal{G}_{LRS}$ in relation to $\mathcal{G}$

of distilling the resonance from GCNs and fostering this resonance to design a inductive approach.

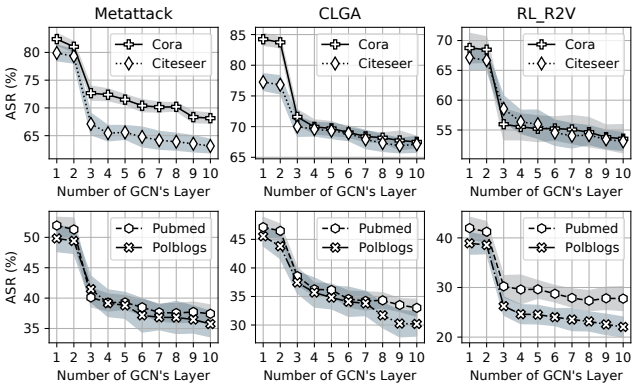


Figure 6: GCN's layer count and ASR relation.

How Robust of LRS-constructed Graphs Derived from a transductive model, the LRS captures distinct resonance regions and transforms a localized, unweighted graph (also perceivable as a graph with unitary weights) into a weighted one. The implementation of the LRS within the GRN enables the demarcation of a learnable resonance scope for inductive learning. Consequently, it becomes essential to validate the efficacy of the LRS through its embedding precision within the transductive model.

We initiate by presenting results obtained under non-adversarial conditions. We sum the LRS of all nodes in $\mathcal{G}$ and apply min-max scaling to all weights, thus creating a global weighted graph $\mathcal{G}_{LRS}$. Then, we train a standard 2-layer GCN $\mathcal{M}_{\mathcal{G}}$ on Cora dataset (denoted as $\mathcal{G}$) and visualize its well-trained representations. Then, we feed $\mathcal{G}_{LRS}$ into $\mathcal{M}_{\mathcal{G}}$ to generate its visualization. Lastly, for comparison, we create a random-weighted graph $\mathcal{G}_{random}$ whose edge distribution is the same as $\mathcal{G}$, and input it into $\mathcal{M}_{\mathcal{G}}$ to get the corresponding visualization. As Figure 5(a) shows, under identical weights, the representations of different categories in $\mathcal{G}_{LRS}$ are tighter than those in $\mathcal{G}$. This suggests that introducing LRS brings beneficial global weights, which enhance the model's performance in non-adversarial scenarios.

Then, we explored the similarity between $\mathcal{G}$ and $\mathcal{G}_{LRS}$. Using strength distribution, akin to degree distribution for unweighted graphs (Zügner, Akbarnejad, and Günnemann 2018), we found the two weighted graphs are notably similar. Figure 5(b) confirms this, showing a stark difference from $\mathcal{G}_{random}$. Therefore, $\mathcal{G}_{LRS}$ maintains the traits of $\mathcal{G}$.

We next assessed the classification accuracy of $\mathcal{G}_{LRS}$ under adversarial attacks using varying Metattack perturbation rates $p_r = \frac{r}{|\mathcal{E}|}$. Figure 5(c) shows that as p_r increases, $\mathcal{G}_{LRS}$'s accuracy consistently edges out $\mathcal{G}$. This suggests that the LRS introduces a resonance in the transductive model, marginally boosting its adversarial robustness.

Generalizable Robustness of GRN We assess the adversarial robustness of the 3-layer GRN under generalization demands by comparing its accuracy against other baselines on perturbed graphs. We partition a subset of the dataset as training set with proportions (also named "seen" rate) s_r as 20%, 40%, and 60%. The data within these proportions are deemed "seen" by the GRN, while the remaining data is categorized as "unseen". Utilizing Metattack as our attack approach, we adopt the standard 2-layer GCN as the surrogate model. By adjusting p_r, we derive the corresponding perturbed graphs. Then, we evaluate the classification performance of the baselines on these graphs, placing an emphasis on the accuracy upon model convergence. For each setting,

Dataset	p_r (%)	Defense Baselines					GRN		
		RGCN	GNN-SVD	Pro-GNN	Jaccard	EGNN	s_r=20%	s_r=40%	s_r=60%
Cora	0	83.49±0.57	81.14±0.79	85.01±0.40	81.74±0.36	85.00±0.40	83.74±1.68	86.79±2.27	**87.75±0.93**
	5	77.20±0.47	78.29±0.63	80.10±0.22	80.56±1.30	82.24±0.49	81.48±0.83	86.04±3.15	**86.24±1.54**
	10	72.65±0.40	70.81±1.77	74.45±0.28	75.07±1.28	76.38±0.35	79.51±1.62	81.38±2.58	**82.03±1.48**
	20	59.31±0.27	56.67±1.22	64.68±0.75	73.54±0.94	69.82±0.71	73.79±1.91	74.14±1.93	**74.81±1.52**
Citeseer	0	71.81±0.71	70.42±0.39	74.94±0.40	73.82±0.56	74.92±0.66	75.69±0.69	78.19±1.30	**83.26±0.73**
	5	71.22±0.61	68.86±0.47	72.45±0.88	71.41±0.65	73.60±0.45	75.40±0.95	77.54±1.04	**81.66±0.70**
	10	67.53±0.60	68.70±0.89	70.16±1.05	70.09±0.48	73.66±0.37	73.81±1.10	77.04±1.51	**80.32±0.49**
	20	63.20±1.70	57.95±1.48	55.84±1.28	67.22±1.32	65.91±1.20	71.06±1.20	72.72±1.28	**77.21±0.64**
Pubmed	0	84.57±0.39	83.25±0.35	84.96±0.08	84.87±0.10	85.94±0.10	80.26±0.43	85.51±0.66	**87.27±0.67**
	5	81.25±0.50	82.90±0.26	83.00±0.10	82.32±0.11	83.89±0.09	77.86±0.35	81.92±0.59	**83.36±0.62**
	10	78.96±0.43	80.35±0.21	80.82±0.20	80.77±0.11	**82.13±0.15**	76.62±0.55	79.29±0.60	80.99±0.58
	20	71.33±0.40	73.57±0.15	74.16±0.16	73.41±0.12	76.01±0.19	74.98±0.64	76.87±0.60	**77.15±0.56**
Polblogs	0	94.87±0.19	95.08±0.22	95.45±0.12	95.03±0.57	**95.70±0.34**	95.42±0.56	94.88±0.43	94.97±0.31
	5	73.28±0.18	88.86±0.58	90.98±0.69	90.97±0.61	89.97±1.25	90.18±0.43	**91.22±0.38**	89.37±0.46
	10	70.91±0.37	80.38±0.85	85.60±1.08	85.93±1.39	83.66±1.81	**86.30±0.70**	85.43±0.68	85.07±0.54
	20	57.97±0.41	55.33±2.07	73.52±0.53	70.47±1.27	75.87±0.88	**82.03±0.79**	81.96±0.72	81.56±0.18

Table 1: Classification accuracy (%) on the perturbed graph. p_r is the perturbation rate and s_r is the "seen" rate.

we executed 10 iterations, tabulating both the average outcome and its variability. Table 1 reports the results.

From the data, both clean and perturbed graphs show GRN with a s_r = 60% generally surpasses the baseline in accuracy. There are three exceptions: 1) For the Pubmed dataset at p_r = 10%, this is due to EGNN using graph smoothing to enhance adversarial robustness. In this case, perturbations may be more pronounced in a certain area, and EGNN could leverage this by smoothing concentrated perturbation patterns. However, these incidents are rare, and as p_r increases, GRN's accuracy returns to its peak. 2) With the Polblogs dataset at p_r = 0%, GRN is slightly behind EGNN by 0.28%. Yet, as p_r rises, the decline in GRN's accuracy is the least noticeable among all baselines, ensuring its top position. 3) An intriguing pattern emerging from the Polblogs dataset is the non-proportional relationship between the GRN's s_r and its accuracy. The peculiarity of the Polblogs dataset is that its nodes lack intrinsic features. Typically, scholars have used node degrees as proxies for these absent attributes. This substitution results in the inherent attributes of Polblogs leaning towards uniformity. Expanding the training set's scale exacerbate the oversmoothing phenomenon, culminating in diminished accuracy.

Ablation Studies GRNs combine edge-transmitted signal $\mathbf{E}_{G_i}^{(\ell)}$ and node signal $\mathbf{Z}_{G_i}^{(\ell)}$ for node v_i's representation. We initiate an ablation study to delve into this process. First, we embed only $\mathbf{Z}_{G_i}^{(\ell)}$, naming the model GRN$_{\mathbf{Z}}$. This appears similar to a 2-depth GraphSAGE with mean aggregation, indicating potential vulnerability to adversarial attacks. We then examine the combination order of $\mathbf{E}_{G_i}^{(\ell)}$ and $\mathbf{Z}_{G_i}^{(\ell)}$. The default GRN order is GRN$_{\mathbf{E},\mathbf{Z}}$. We test GRN$_{\mathbf{Z},\mathbf{E}}$ (reversed order) and GRN$_{shuf}$ (shuffled rows). Results (Table 2) show GRN$_{\mathbf{Z}}$ underperforms, especially in adversarial settings, emphasizing the importance of co-embedding both signals. Precisions of GRN$_{\mathbf{Z},\mathbf{E}}$, GRN$_{\mathbf{E},\mathbf{Z}}$, and GRN$_{shuf}$ are comparable due to the edge-transmitted signal, which, combined with the node signal through shared parameters, results in consistent performance regardless of order. This suggests that GRN has the capability to recognize a latent graph structure, wherein edge-transmitted signals function as latent node signals, contributing to adversarial robustness and insensitivity to signal order.

Dataset	p_r (%)	Standard	Ablated models		
		GRN$_{\mathbf{E},\mathbf{Z}}$	GRN$_{\mathbf{Z},\mathbf{E}}$	GRN$_{shuf}$	GRN$_{\mathbf{Z}}$
Cora	0	87.75	87.14±0.81	87.28±0.92	87.74±0.72
	5	86.24	86.18±0.75	86.34±0.98	85.56±0.87
	10	82.03	82.81±0.84	82.54±1.05	79.07±0.90
	20	74.81	74.53±0.80	74.42±1.36	63.54±1.30
Citeseer	0	83.26	83.07±0.64	83.52±0.92	82.23±0.97
	5	81.66	81.49±0.71	81.45±0.95	79.47±1.10
	10	80.32	80.10±0.65	80.46±1.17	74.39±1.44
	20	77.21	76.95±1.26	76.84±1.53	69.05±1.86

Table 2: Classification accuracy (%) of ablated models.

Conclusions

We addressed critical concerns surrounding the transductive nature of existing robust graph learning models. We began by establishing the equivalence between GCN-driven MP and edge-based waves. Subsequently, we demonstrated that a 3-layer GCN capitalizes on the unique resonance intrinsic to waves to achieve robustness. Delving deeper, we formalized this resonance as a coupling between a node and its surrounding local structure. We then introduced an inductive graph learning model, GRN. Experimental results not only corroborated our theoretical insights but also highlighted the exemplary robustness of the proposed GRN model.

Acknowledgments

This work is supported in part by the National Key Research and Development Program of China (No. 2020YFB1805400), the National Natural Science Foundation of China (No. U19A2068, No. 62372313), and the Sichuan Science and Technology Program (No. 2023YFG0113).

References

Ahmad, S.; Saifullah, S.; Khan, A.; and Wazwaz, A. M. 2023. Resonance, fusion and fission dynamics of bifurcation solitons and hybrid rogue wave structures of Sawada–Kotera equation. *Commun. Nonlinear. Sci. Numer. Simul.*, 119: 107117.

Berberidis, D.; and Giannakis, G. B. 2019. Node embedding with adaptive similarities for scalable learning over graphs. *IEEE Trans. Knowl. Data Eng.*, 33(2): 637–650.

Blas, D.; and Jenkins, A. C. 2022. Detecting stochastic gravitational waves with binary resonance. *Phys. Rev. D*, 105(6): 064021.

Bruna, J.; Zaremba, W.; Szlam, A.; and LeCun, Y. 2013. Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv:1312.6203*.

Bykov, D. A.; Bezus, E. A.; and Doskolovich, L. L. 2019. Coupled-wave formalism for bound states in the continuum in guided-mode resonant gratings. *Phys. Rev. A*, 99(6): 063805.

Chen, W.; Feng, F.; Wang, Q.; He, X.; Song, C.; Ling, G.; and Zhang, Y. 2021. Catgcn: Graph convolutional networks with categorical node features. *IEEE Trans. Knowl. Data Eng.*

Dai, H.; Li, H.; Tian, T.; Huang, X.; Wang, L.; Zhu, J.; and Song, L. 2018. Adversarial attack on graph structured data. In *Proc. 35th Int. Conf. Mach. Learn.*, 1115–1124.

Defferrard, M.; Bresson, X.; and Vandergheynst, P. 2016. Convolutional neural networks on graphs with fast localized spectral filtering. volume 29.

Entezari, N.; Al-Sayouri, S. A.; Darvishzadeh, A.; and Papalexakis, E. E. 2020. All you need is low (rank) defending against adversarial attacks on graphs. In *Proc. 13th Int. Conf. Web Search Data Min.*, 169–177.

Friedman, J.; and Tillich, J.-P. 2004. Wave equations for graphs and the edge-based Laplacian. *Pac. J. Math.*, 216(2): 229–266.

Hamilton, W.; Ying, Z.; and Leskovec, J. 2017. Inductive representation learning on large graphs. In *Proc. 31st Adv. Neural Inf. Proces. Syst.*

Jin, W.; Ma, Y.; Liu, X.; Tang, X.; Wang, S.; and Tang, J. 2020. Graph structure learning for robust graph neural networks. In *26th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, 66–74.

Kipf, T. N.; and Welling, M. 2017. Semi-supervised classification with graph convolutional networks. In *Proc. 5th Int. Conf. Learn. Represent.*

Kovalyov, M. 1989. Resonance-type behaviour in a system of nonlinear wave equations. *J. Differ. Equ.*, 77(1): 73–83.

Kreps, S.; and Kriner, D. 2020. Model uncertainty, political contestation, and public trust in science: Evidence from the COVID-19 pandemic. *Sci. Adv.*, 6(43): eabd4563.

Kula, M. 2015. Metadata Embeddings for User and Item Cold-start Recommendations. In *CORR.*, volume 1507.08439.

Li, K.; Liu, Y.; Ao, X.; Chi, J.; Feng, J.; Yang, H.; and He, Q. 2022. Reliable Representations Make A Stronger Defender: Unsupervised Structure Refinement for Robust GNN. In *Proc. 28th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*

Liu, A.; Li, B.; Li, T.; Zhou, P.; and Wang, R. 2022. AN-GCN: An Anonymous Graph Convolutional Network Against Edge-Perturbing Attacks. *IEEE Trans. Neural Netw. Learn. Syst.*

Liu, X.; Jin, W.; Ma, Y.; Li, Y.; Liu, H.; Wang, Y.; Yan, M.; and Tang, J. 2021. Elastic graph neural networks. In *Proc. 38th Int. Conf. Mach. Learn.*, 6837–6849.

Müller, E. 2023. Graph clustering with graph neural networks. *J. Mach. Learn. Res.*, 24: 1–21.

Nadal, S.; Abelló, A.; Romero, O.; Vansummeren, S.; and Vassiliadis, P. 2021. Graph-driven federated data management. *IEEE Trans. Knowl. Data Eng.*, 35(1): 509–520.

Samuel, F., G; John, B., D; Joichi, I.; Jonathan, Z., L; Andrew, B., L; and Isaac, K., S. 2019. Adversarial attacks on medical machine learning. *Science*, 363(6433): 1287–1289.

Shatah, J.; and Struwe, M. 1993. Regularity results for nonlinear wave equations. *Ann. Math.*, 138(3): 503–518.

Sixiao, Z.; Hongxu, C.; Xiangguo, S.; Yicong, L.; and Xu, G. 2022. Unsupervised Graph Poisoning Attack via Contrastive Loss Back-propagation. In *Proc. 31st Int. Conf. World Wide Web*.

Sun, L.; Dou, Y.; Yang, C.; Zhang, K.; Wang, J.; Philip, S. Y.; He, L.; and Li, B. 2022. Adversarial attack and defense on graph data: A survey. *IEEE Trans. Knowl. Data Eng.*

Sun, Y.; Wang, S.; Tang, X.; Hsieh, T.-Y.; and Honavar, V. 2020. Non-target-specific node injection attacks on graph neural networks: A hierarchical reinforcement learning approach. In *Proc. 29th Int. Conf. World Wide Web*, volume 3.

Walt, W.; Jack, C.; and Christof, T. 2019. Adversarial explanations for understanding image classification decisions and improved neural network robustness. *Nat. Mach. Intell.*, 1: 508—516.

Wu, H.; Wang, C.; Tyshetskiy, Y.; Docherty, A.; Lu, K.; and Zhu, L. 2019. Adversarial examples for graph data: deep insights into attack and defense. In *Proc. 28th Int. Joint Conf. Artif. Intel.*

Xiao, H.; Chen, Y.; and Shi, X. 2019. Knowledge graph embedding based on multi-view clustering framework. *IEEE Trans. Knowl. Data Eng.*, 33(2): 585–596.

Zhao, Y.; Zhou, H.; Zhang, A.; Xie, R.; Li, Q.; and Zhuang, F. 2022. Connecting embeddings based on multiplex relational graph attention networks for knowledge graph entity typing. *IEEE Trans. Knowl. Data Eng.*

Zhu, D.; Zhang, Z.; Cui, P.; and Zhu, W. 2019. Robust graph convolutional networks against adversarial attacks. In *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, 1399–1407.

Zügner, D.; Akbarnejad, A.; and Günnemann, S. 2018. Adversarial attacks on neural networks for graph data. In *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, 2847–2856.

Zügner, D.; and Günnemann, S. 2018. Adversarial Attacks on Graph Neural Networks via Meta Learning. In *Proc. 6th Int. Conf. Learn. Represent.*