

Graph Rewiring in GNNs to Mitigate Over-Squashing and Over-Smoothing: A Survey

Hugo Attali¹, Nathalie Pernelle¹, Davide Buscaldi¹ and Fragkiskos D. Malliaros²

¹Université Sorbonne Paris Nord, CNRS, LIPN

²Université Paris-Saclay, CentraleSupélec, Inria

{attali, pernelle, buscaldi}@lipn.univ-paris13.fr, fragkiskos.malliaros@centralesupelec.fr

Abstract

Graph Neural Networks are powerful models for learning from graph-structured data, yet their effectiveness is often limited by two critical challenges: over-squashing, where information from distant nodes is excessively compressed, and over-smoothing, where repeated propagation makes node representations indistinguishable. Both phenomena stem from the interaction between message passing and the input topology, ultimately degrading information flow and limiting the performance of GNNs. In this survey, we examine graph rewiring techniques, a class of methods designed to modify the graph topology to enhance information propagation in GNNs. We provide a comprehensive review of state-of-the-art rewiring approaches, delving into their theoretical underpinnings, practical implementations, and performance trade-offs.

1 Introduction

Graph Neural Networks (GNNs) [Gori *et al.*, 2005; Scarselli *et al.*, 2008; Gilmer *et al.*, 2017] have emerged as powerful models for analyzing structured data represented as graphs across various fields [Zhou *et al.*, 2020]. However, they face significant challenges, especially in heterophilic graphs where nodes belonging to different classes are highly interconnected, and more generally in problems defined on graphs with pronounced long-range dependencies, where task-relevant information may lie several hops away from each node. In such scenarios, the ability to propagate long-range information becomes crucial, yet it is often hindered by phenomena such as over-squashing [Alon and Yahav, 2021; Topping *et al.*, 2022] and over-smoothing [Rusch *et al.*, 2023]. Beyond architectural changes, a growing line of work intervenes directly on the propagation operator used by message passing. A first family of approaches relies on edge reweighting, where existing connections are assigned adaptive strengths to better reflect latent affinities between nodes and to attenuate noisy edges [Zheng *et al.*, 2022]. While this can improve local aggregation, such reweighting remains constrained by the original graph support and therefore offers limited leverage when task-relevant evidence lies far beyond immediate neighborhoods, as is the case in heterophilic or

long-range dependencies settings. To explicitly increase the accessible dependency range, complementary work has studied multi-hop or higher-order aggregation strategies [Gong *et al.*, 2026]. These methods extend each node’s receptive field by replacing one-step neighborhoods with higher-order connectivity (e.g., powers of the adjacency matrix), thereby enabling interactions between distant nodes. From a graph perspective, they induce a topology by transforming the adjacency matrix and can thus be interpreted as an implicit form of rewiring. However, the benefits of multi-hop strategies come with a trade-off: aggregating over large hop neighborhoods can intensify information compression and accentuate over-squashing [Alon and Yahav, 2021; Topping *et al.*, 2022], while the graph densification may accelerate over-smoothing and hinder scalability on large graphs [Nguyen *et al.*, 2023; Giraldo *et al.*, 2023]. These limitations motivate graph rewiring that modifies the graph topology to promote long-range information flow while attenuating structural patterns that induce excessive mixing or compression, and thereby exacerbate over-smoothing and over-squashing. Complementing the over-squashing survey of [Akansha, 2025], this work provides a method-centric survey of rewiring techniques, covering both preprocessing and dynamic rewiring methods to mitigate over-squashing and over-smoothing.

Objectives and outline of the survey. Our goal is to provide a structured overview of rewiring approaches, with an emphasis on their underlying principles, design choices, and trade-offs they introduce. We highlight why such techniques are particularly relevant in challenging regimes, including heterophilic graphs and tasks exhibiting long-range dependencies, where informative signals may lie beyond the immediate neighborhood. Finally, we outline open problems and future directions, positioning graph rewiring as a principled way to intervene on graph structure in order to better understand and improve graph learning. We argue that progress will benefit from clearer problem formulations, more explicit assumptions, and evaluation protocols that make claims robust, comparable, and interpretable across settings.

2 Preliminary Concepts

2.1 Notation

We begin by introducing the notation used throughout this paper. A graph is represented as a tuple $G = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$

denotes the set of nodes and $\mathcal{E}$ the set of edges. The number of nodes is denoted by $N = |\mathcal{V}|$, and an edge connecting node i to node j is represented by $e_{ij} \in \mathcal{E}$. In this work, we focus on undirected graphs, meaning that if $e_{ij} \in \mathcal{E}$, then $e_{ji} \in \mathcal{E}$. We define the $N \times N$ adjacency matrix $\mathbf{A}$ such that $\mathbf{A}_{i,j} = 1$ if $(i, j) \in \mathcal{E}$, and 0 otherwise. Additionally, $\mathbf{D}$ denotes the diagonal matrix where $\mathbf{D}_{i,i} = d_i$, the degree of node i . The maximum and minimum degrees are denoted by $d_{\max}$ and $d_{\min}$, respectively. We note by $h_i^{(\ell)}$ the embedding of the node i at the ℓ layer.

2.2 Graph Neural Networks

Graph Neural Networks (GNNs) are a class of models designed to leverage the structure of graph data by iteratively propagating and updating node features through their neighborhoods. This process is typically framed within the Message Passing Neural Networks (MPNNs) paradigm, where node representations are refined by exchanging information with neighboring nodes. Each iteration of the GNN involves two operations, aggregation and update to compute embeddings h_i^ℓ at the layer ℓ based on message $m_i^{(\ell)}$ containing information on neighbors $\mathcal{N}(i)$, as follows:

$$\begin{aligned} m_i^{(\ell)} &= \text{AGGREGATE}^{(\ell)} \left(h_i^{(\ell-1)}, \left\{ h_j^{(\ell-1)} \mid j \in \mathcal{N}(i) \right\} \right), \\ h_i^{(\ell)} &= \text{UPDATE}^{(\ell)} \left(h_i^{(\ell-1)}, m_i^{(\ell)} \right). \end{aligned} \quad (1)$$

2.3 Heterophily and Long-range Dependencies

We call a graph heterophilic when adjacent nodes are not likely to share the same label [Zhu *et al.*, 2020]. In this regime, the 1-hop neighborhood could be a misleading predictor of a node’s label: local aggregation mixes signals coming from different classes, so useful evidence may emerge only after aggregating information over longer paths, or from more distant regions that are predictive of the target label.

Long-range dependencies refer to tasks where the target at a node (or graph) depends on information located at a large graph distance. In this context, since MPNNs propagate information locally (one hop per layer), capturing signals located at large graph distances typically requires increasing depth. Greater depth, however, tends to amplify two well-known limitations of message passing: (i) over-squashing, i.e., the aggregation of many distinct long-range signals into fixed-size node representations, and (ii) over-smoothing, i.e., the progressive loss of discriminative power caused by repeated neighborhood mixing. Consequently, heterophilic settings and long-range dependency tasks are particularly sensitive to these effects.

2.4 Over-squashing

Recent studies have highlighted that message passing neural networks struggle with tasks that require long-range dependencies [Di Giovanni *et al.*, 2023]. A useful way to formalize this limitation is through long-range influence, quantified by a Jacobian-based sensitivity: for two nodes i and j , let

$$J_{i \leftarrow j}^{(\ell)} = \left\| \frac{\partial h_i^{(\ell)}}{\partial h_j^{(0)}} \right\|$$

denote the Jacobian sensitivity of $h_i^{(\ell)}$ to perturbations of $h_j^{(0)}$. Over-squashing denotes the regime where this effect decays rapidly with graph distance, so that signals from an exponentially growing k -hop neighborhood are compressed into fixed-size representations [Di Giovanni *et al.*, 2023]. Over-squashing is amplified by the graph structure. If two far-apart regions are linked by only a few paths, most long-range messages must pass through the same small set of paths, which increases compression and makes distant information harder to preserve. These structural bottlenecks thus limit long-range communication in message passing [Topping *et al.*, 2022].

2.5 Over-smoothing

Over-smoothing is a complementary depth-related limitation of message passing [Rusch *et al.*, 2023]. Since each layer repeatedly mixes information across neighborhoods, deep propagation behaves like a diffusion process that progressively reduces variation across node representations. One standard way to formalize over-smoothing is via the asymptotic collapse of pairwise differences:

$$\max_{i,j} \|h_i^{(\ell)} - h_j^{(\ell)}\| \rightarrow 0 \quad (\ell \rightarrow \infty),$$

meaning that node embeddings become increasingly indistinguishable and thus less informative for discrimination [Cai and Wang, 2020]. In practice, this homogenization can be intensified when the graph becomes denser, as stronger mixing accelerates the loss of feature diversity.

2.6 Rewiring

Since these limitations are largely exacerbated by the graph topology, a natural approach is to modify the graph on which message passing is performed to mitigate them.

Definition. Let $G = (V, E)$ be a simple undirected graph. A *rewiring* is any mapping

$$\mathcal{R} : \mathcal{G} \rightarrow \mathcal{G}, \quad G \mapsto G^+ = (V^+, E^+),$$

where, possibly, $V \subseteq V^+$ (augmentation with virtual nodes) and where E^+ is typically obtained from E by adding and/or removing edges. We call a rewiring static if $\mathcal{R}$ is applied once prior to training. It is dynamic if the rewired graph depends on learnable parameters and/or evolves over training, i.e., $G^+ = \mathcal{R}_\theta(G)$ where θ denotes the learnable parameters of the GNN used during training.

The objective is to obtain a modified graph G^+ whose topology is more favorable to information propagation under an MPNN than that of the original graph G . Unlike traditional graph learning methods that typically adjust the UPDATE and/or AGGREGATE operations while maintaining fixed neighborhoods, rewiring methods approach the problem differently. They treat graph modification as allowing alterations to the neighborhood structure while keeping the message passing operations intact. Figure 1 presents an example of heuristic graph rewiring.

3 Topological and Connectivity Measures

Various topological measures have been utilized and proposed as a basis for constructing G^+ .

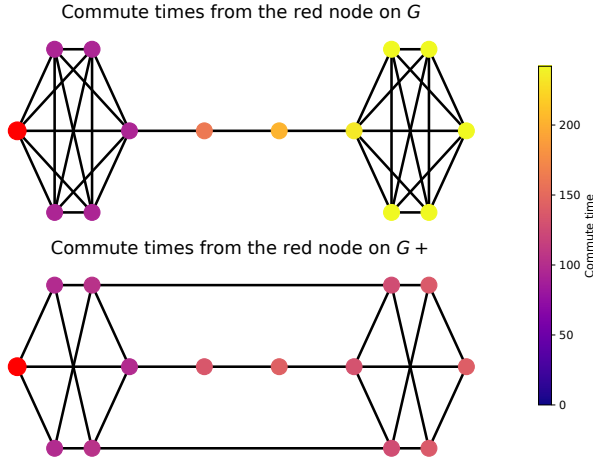


Figure 1: Example of a graph G and its rewired instance G^+ , with node colours encoding commute times (see Section 3.3) from the highlighted red node. In G^+ , intra-clique density is reduced while connectivity across the bottleneck is strengthened, which shortens and balances commute times. This yields a topology more favourable to information propagation, thereby mitigating over-smoothing and over-squashing.

3.1 Discrete Curvature on Graphs

A prominent concept in graph rewiring is discrete curvature. Discrete curvature draws the analogy between dispersion geodesics on a manifold and an edge on a graph. Intuitively, edge curvature reflects the local structure around the neighborhoods of two connected nodes. Edge curvature was first linked to bottleneck structures by [Topping *et al.*, 2022], demonstrating that highly negatively curved edges characterise bottleneck structures. Various discrete curvature measures have been studied for their potential to improve graph rewiring techniques.

Ollivier Curvature. Ollivier curvature is a discrete analogue of Ricci curvature based on optimal transport, measuring how similar the probability distributions supported on the local neighborhoods of two nodes are [Ollivier, 2007]. Let μ_i be the probability measure associated with node i , defined via a lazy random walk with parameter α :

$$\mu_i(j) = \begin{cases} \alpha & \text{if } j = i, \\ \frac{1 - \alpha}{d_i} & \text{if } j \in \mathcal{N}(i), \\ 0 & \text{otherwise,} \end{cases} \quad (2)$$

where d_i is the degree of node i and $\mathcal{N}(i)$ is its set of neighbors. Let $W_1(\mu_i, \mu_j)$ denote the 1-Wasserstein distance between μ_i and μ_j , that is, the minimal transport cost between these two probability measures when the ground cost is given by the graph shortest-path distance. The Ollivier curvature c_{ij}^O of an edge e_{ij} is then defined as

$$C_{i,j}^O = 1 - \frac{W_1(\mu_i, \mu_j)}{\text{dist}(i, j)}, \quad (3)$$

where $\text{dist}(i, j)$ denotes the shortest-path distance between nodes i and j . Unlike other curvature measures, Ollivier cur-

vature is bounded, with $c_{ij}^O \in [-2, 1]$ [Nguyen *et al.*, 2023], which can make it more interpretable.

Augmented Forman Curvature. Proposed by [Samal *et al.*, 2018], this measure extends the original definition of Forman Curvature [Forman, 2003] to incorporate the presence of triangles in a graph. For an undirected graph, the curvature of edge e_{ij} is given by:

$$C_{i,j}^{\text{AF}} = 4 - d_i - d_j + 3m, \quad (4)$$

where m is the number of triangles containing e_{ij} , and d_i and d_j are the degrees of nodes i and j , respectively.

Balanced Forman Curvature. [Topping *et al.*, 2022] introduced a more expressive combinatorial curvature measure, known as balanced Forman curvature, which considers not only triangles but also cycles of different lengths:

$$C_{i,j}^{\text{BF}} = \frac{2}{d_i} + \frac{2}{d_j} - 2 + 2 \frac{m}{\max\{d_i, d_j\}} + \frac{m}{\min\{d_i, d_j\}} + \frac{(\Gamma_{\max})^{-1}}{\max\{d_i, d_j\}} (\gamma_i + \gamma_j), \quad (5)$$

where $\Gamma_{\max}(i, j)$ is the maximum number of 4-cycles based at edge e_{ij} , and γ_i counts the number of 4-cycles at e_{ij} without diagonals.

Jost-Liu Curvature. The Jost-Liu curvature further refines edge-based curvature measures by incorporating the local clustering around an edge, through the number of triangles (3-cycles) containing e_{ij} [Jost and Liu, 2014]:

$$C_{i,j}^{\text{JL}} = - \left(1 - \frac{1}{d_i} - \frac{1}{d_j} - \frac{m}{d_i \wedge d_j} \right)_+ - \left(1 - \frac{1}{d_i} - \frac{1}{d_j} - \frac{m}{d_i \vee d_j} \right)_+ + \frac{m}{d_i \vee d_j}, \quad (6)$$

where d_i and d_j are the degrees of the nodes. A more detailed discussion on the role of different edge curvature measures is provided in [Tori *et al.*, 2025].

3.2 Effective Resistance

Effective resistance offers a more global perspective of how well two nodes communicate by considering all possible paths between them, rather than focusing solely on their local neighborhoods. It is defined as the inverse of the sum of the inverses of the lengths of all disjoint paths connecting two nodes, placing more weight on shorter paths [Chandra *et al.*, 1989]. The higher the effective resistance of an edge, the more it acts as a bottleneck in the graph. For vertices i and j connected by several disjoint paths, the effective resistance is given by:

$$\mathbf{R}_{i,j} = \left(\sum_{ij\text{-paths } p} \text{length}(p)^{-1} \right)^{-1}. \quad (7)$$

A lower $\mathbf{R}_{i,j}$ indicates easier communication and information flow between nodes i and j , making it a valuable metric for identifying key structural bottlenecks in the graph. Compared to curvature-based measures, effective resistance provides a global view of graph connectivity.

3.3 Random Walk-based Measures

Walk-based quantities provide natural proxies for long-range connectivity, such a hitting times or return probabilities, and can be related to over-squashing by quantifying the ease with which information can propagate between distant regions of the graph. Commute time is particularly informative: it measures the expected number of steps for a random walk to travel from i to j and back, so lower commute times indicate easier bidirectional communication between distant nodes (see Fig. 1). In the context of MPNNs, the number of random walks of a given length between two nodes controls how easily information can flow between them, especially at long distances. Theorem 4.1 in [Di Giovanni *et al.*, 2023] shows that this directly impacts over-squashing: distant nodes with few connecting walks exchange information only weakly. For nodes i and j at distance r in a graph G , let $\gamma_r(i, j)$ denote the number of random walks of length r between i and j . The theorem states that the Jacobian sensitivity between nodes decays with distance at a rate governed both by model parameters (Lipschitz constant c_σ , weights w , hidden dimension p) and graph properties (minimum degree $d_{\min}$, and walk counts $\gamma_r(i, j)$):

$$\frac{\partial h_i}{\partial h_j} \leq C_k \gamma_{r+k}(i, j) \left(\frac{2c_\sigma w p}{d_{\min}} \right)^r. \quad (8)$$

For large r , if $\gamma_{r+k}(i, j)$ is small, the Jacobian decays exponentially, which increases the risk of over-squashing. Equivalently, a large commute time between i and j on the graph, i.e., the expected number of steps needed to go from i to j and back, indicates poor connectivity and, consequently, stronger attenuation of messages between these nodes.

Building on this perspective, [Barbero *et al.*, 2024] proposes a local connectivity measure

$$\Omega_k(i, j) = (\tilde{\mathbf{A}}^k)_{i,j}, \quad \tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}, \quad (9)$$

which counts the number of walks from i to j of length k . For fixed k , a large $\Omega_k(i, j)$ (or, relatedly, a small commute time) indicates strong connectivity and many routes for information exchange, whereas low $\Omega_k(i, j)$ correlates with few available paths, larger commute times, and a higher susceptibility to over-squashing.

3.4 Spectral Gap

Another widely used measure is the spectral gap, which is closely related to the Cheeger constant [Chung and Graham, 1997] defined as:

$$\text{Ch}(G) = \min_{\substack{\mathcal{S} \subset \mathcal{V} \\ 1 \leq |\mathcal{S}| \leq |\mathcal{V}|/2}} \frac{|\partial \mathcal{S}|}{|\mathcal{S}|}. \quad (10)$$

where $\mathcal{S} \subset \mathcal{V}$ and $\partial \mathcal{S} = \{(i, j) \in \mathcal{E} : i \in \mathcal{S}, j \in \mathcal{V} \setminus \mathcal{S}\}$ denotes the edge boundary of $\mathcal{S}$. Small values of $\text{Ch}(G)$ indicate the presence of a sparse cut, i.e., two large vertex groups connected by few edges, revealing a bottleneck; conversely, large values suggest that every balanced partition has many crossing edges. Computing $\text{Ch}(G)$ exactly is intractable in general, but it can be bounded using the second smallest eigenvalue λ_2 of the Laplacian matrix $\mathbf{L}$ via

the discrete Cheeger inequality [Alon and Milman, 1984; Cheeger, 1970]:

$$\frac{\lambda_2}{2} \leq \text{Ch}(G) \leq \sqrt{2\lambda_2}. \quad (11)$$

Hence, a small λ_2 (small spectral gap) provides an efficient indicator of a near-disconnected structure: only a few edges may separate the graph into large components, which is precisely the type of bottleneck that can hinder information propagation in message passing GNNs.

4 Rewiring Methods in GNNs

Graph rewiring modifies the graph connectivity used for message passing to improve information propagation, typically with the goal of mitigating over-squashing and over-smoothing. Building on the bottleneck perspective of [Alon and Yahav, 2021], later methods propose algorithmic changes to the graph, most commonly by adding, removing, or reweighting connections either locally to relieve specific structural bottlenecks or globally to reshape the overall topology. We categorize rewiring methods into three categories:

- **Type of structural modification:** the structural operator defining how G is altered, whether through selective edge editing, global adjacency reconstruction, or augmentation via auxiliary nodes.
- **Rewiring signal and information used:** the information source defining the rewiring rule (structure, features, or learned signals) and the scale at which it operates (local or global).
- **Computational efficiency and hyperparameter sensitivity:** the computational overhead of the method and its robustness to rewiring hyperparameters.

Table ?? provides a summary of the primary rewiring methods that have been published to date. For each method, the table indicates whether it mitigates over-squashing (OSQ) and/or over-smoothing (OSM) and specifies the underlying measures upon which the rewiring technique is based. Additionally, we indicate whether the graph rewiring method uses node features during rewiring (UF) and whether rewiring is applied during training (DT).

4.1 Structural-fix Rewiring Strategies

Let $G = (\mathcal{V}, \mathcal{E})$ be a graph and let $\mathbf{S}_G : \mathcal{V} \times \mathcal{V} \rightarrow \mathbb{R}^p$ denote any topological or connectivity descriptor (e.g., curvature, effective resistance, walk-based or spectral measures as in Section 3) computed from G alone. A structural-fix rewiring method is a mapping

$$\mathcal{R} : \mathcal{G} \rightarrow \mathcal{G}, \quad G \mapsto G^+ = (\mathcal{V}^+, \mathcal{E}^+),$$

such that there exists a rule $\Phi : \mathbb{R}^p \rightarrow \{0, 1\}$ with

$$(i, j) \in \mathcal{E}^+ \iff \Phi(\mathbf{S}_G(i, j)) = 1,$$

i.e., edges are added or removed solely as a function of topological descriptors, without using node features, labels, or model parameters.

Curvature-based structural-fix methods. Let C_{ij} denote the curvature of an edge e_{ij} as defined in Section 3.1. Curvature-based structural-fix rewiring is defined as

$$s_G : \mathcal{V} \times \mathcal{V} \rightarrow \mathbb{R}, \quad s_G(i, j) = C_{i,j} \quad \text{for } (i, j) \in \mathcal{E},$$

together with a decision rule $\Phi : \mathbb{R} \rightarrow \{0, 1\}$ acting on curvature values [Topping *et al.*, 2022; Fesser and Weber, 2023; Giraldo *et al.*, 2023; Nguyen *et al.*, 2023; Attali *et al.*, 2025a]. A typical instance fixes thresholds $\tau_{\text{neg}} < 0 < \tau_{\text{pos}}$, to remove highly positively curved edges (τ_{pos}), and then adds the minimum number of edges to alleviate highly negatively curved edges (τ_{neg}). We introduce two sets: a deletion set $\mathcal{E}_{\text{del}} \subseteq \mathcal{E}$ and an addition set $\mathcal{E}_{\text{add}} \subseteq \bar{\mathcal{E}} := (\mathcal{V} \times \mathcal{V}) \setminus \mathcal{E}$. Finally, we construct $G^+ = (\mathcal{V}, \mathcal{E}^+)$ by jointly solving

$$\begin{aligned} (\mathcal{E}_{\text{del}}^*, \mathcal{E}_{\text{add}}^*) \in \arg \min_{\mathcal{E}_{\text{del}} \subseteq \mathcal{E}, \mathcal{E}_{\text{add}} \subseteq \bar{\mathcal{E}}} & |\mathcal{E}_{\text{del}}| + |\mathcal{E}_{\text{add}}| \\ \text{s.t. } & \mathcal{E}^+ = (\mathcal{E} \setminus \mathcal{E}_{\text{del}}) \cup \mathcal{E}_{\text{add}}, \\ & \tau_{\text{neg}} \leq C_{ij}^{(\mathcal{V}, \mathcal{E}^+)} \leq \tau_{\text{pos}}, \\ & \forall (i, j) \in \mathcal{E}^+. \end{aligned}$$

yielding $\mathcal{E}^+ := (\mathcal{E} \setminus \mathcal{E}_{\text{del}}^*) \cup \mathcal{E}_{\text{add}}^*$. This yields a rewired graph G^+ , where deletions target highly positive-curvature edges (mitigating over-smoothing) and additions create shortcuts across bottlenecks (mitigating over-squashing). Note that exists also methods may add edges only (set $\tau_{\text{pos}} = +\infty$) or delete edges only (set $\tau_{\text{neg}} = -\infty$) [Attali *et al.*, 2025a].

Resistance-based methods. Curvature offers a local view of connectivity. Resistance-based methods instead rely on the effective resistance $R_{i,j}^G$, which aggregates all paths between i and j and yields a global connectivity descriptor. In the structural-fix framework, one sets

$$s_G = R_{i,j}^G, \quad (i, j) \in \mathcal{V} \times \mathcal{V},$$

and seeks a rewired graph $G^+ = (\mathcal{V}, \mathcal{E}^+)$ that minimizes a global resistance functional, typically

$$G^+ \in \arg \min_{\tilde{G}} \sum_{i < j} R_{i,j}^{\tilde{G}} \quad \text{subject to an edge budget on } \tilde{\mathcal{E}},$$

as in [Black *et al.*, 2023]. In practice, the method (GTR) iteratively adds edges between node pairs with large $R_{i,j}^G$, strengthening long-range connectivity by bringing them closer in effective-resistance distance.

Walk methods. Because both curvature and effective resistance can be computationally demanding, walk-based descriptors have been proposed as cheaper proxies. For a fixed walk length k , one may define :

$$\Omega_G(i, j) = \sum_{\ell=1}^k (\mathbf{A}^\ell)_{ij},$$

i.e., the number of walks of length at most k between i and j , where $\mathbf{A}$ is the adjacency matrix of G . A structural-fix rule then promotes edges between pairs with few walks (low $s_G(i, j)$) and deprioritizes pairs with many alternative routes, consistent with the observation that nodes connected by many walks are less sensitive to over-squashing [Di Giovanni *et al.*, 2023]. In practice, this principle motivates schemes based on adjacency powers, such as LASER's k -power construction [Barbero *et al.*, 2024].

Spectral gap methods. Spectral methods act at a global scale. Let $\mathbf{L}$ denote a normalized Laplacian of G , and let $\gamma(G) = \lambda_2(L)$ be its spectral gap. Since a larger spectral gap is closely related to lower effective resistance and better expansion, spectral-gap based rewiring aims to construct a graph G^+ with increased $\gamma(G^+)$. Formally, one considers

$$G^+ \in \arg \max_{\tilde{G}} \gamma(\tilde{G}) \quad \text{subject to an edge-budget constraint on } \tilde{\mathcal{E}}.$$

and approximates this objective by iteratively adding edges that most improve the spectral gap (FOSR) [Karhadkar *et al.*, 2023]. Expander-based constructions [Deac *et al.*, 2022; Wilson *et al.*, 2024] are a canonical example of this principle: expanders achieve large spectral gap, small diameter, and high Cheeger constant with relatively few edges, thereby reducing bottlenecks and strongly mitigating over-squashing. More recent approaches such as spectral graph pruning [Jamadandi *et al.*, 2024] and spectrum preserving sparsification [Liang *et al.*, 2025] refine this idea by assigning a spectral importance score to each edge and selecting a sparse subgraph that, respectively, improves the spectral gap or approximately preserves the Laplacian spectrum, while still providing favourable conditions for long-range propagation.

Virtual node methods. In addition to modifying the edge set, some structural-fix methods augment the node set with auxiliary nodes. A canonical example is the master node: one considers the augmented graph $G^+ = (\mathcal{V} \cup \{M\}, \mathcal{E}')$, where

$$\mathcal{E}' = \mathcal{E} \cup \{(M, i) : i \in \mathcal{V}\},$$

so that every original node is connected to M . This construction reduces the diameter of the original node set to 2, decreases effective resistance between distant nodes, and thereby mitigates over-squashing [Southern *et al.*, 2025]. More recent fractal-node constructions introduce a collection of auxiliary nodes $\mathcal{F}$ together with attachment sets $\{S_f \subseteq \mathcal{V} : f \in \mathcal{F}\}$, yielding

$$G^+ = (\mathcal{V} \cup \mathcal{F}, \mathcal{E} \cup \{(f, i) : f \in \mathcal{F}, i \in S_f\}),$$

in order to create multi-scale shortcut structures that facilitate long-range communication without enforcing a single global hub topology [Choi *et al.*, 2025].

4.2 Feature-aware Rewiring Strategies

Beyond structural-fix rewiring, feature-aware methods use node attributes as the rewiring signal, which is particularly relevant when the observed topology is noisy, incomplete, or weakly aligned with the task.

Delaunay Rewiring (DR) [Attali *et al.*, 2024] discards $\mathcal{E}$ and reconstructs G^+ via Delaunay triangulation in a two-dimensional feature space, yielding a sparse planar graph. This triangulation avoids edges with highly negative curvature and thus alleviates over-squashing, and enforces a maximum clique size of three, preventing uncontrolled densification that would accelerate over-smoothing. TRIGON [Attali *et al.*, 2025b] further makes this principle task-adaptive by learning which candidate triangles (collected across the original graph, a k -NN feature similarity graph, and a Delaunay graph) should be retained through a differentiable selection module, yielding non-local triangulations that strengthen long-range connectivity.

Feature-similarity rewiring (ComFy) [Rubio-Madrigal *et al.*, 2025] rewires the graph by ranking candidate edge additions and deletions using a feature-affinity measure (typically cosine similarity) and applying the highest-ranked modifications. Community-aware variants constrain these modifications using an initial Louvain partition, so rewiring can strengthen within-community links or introduce controlled between-community connections without disrupting the original community layout. In GOKU [Liang *et al.*, 2025], features intervene during sparsification: edge retention is biased by both feature similarity and a spectral importance signal (effective resistance-based), making structurally important, feature-consistent edges more likely to be kept, while feature-inconsistent ones are more likely to be removed.

JDR [Linkerhäger *et al.*, 2025] targets regimes where both the topology and node features are noisy or misaligned: instead of relying on local similarity scores, it enforces global consistency between the graph and the feature geometry to obtain a topology more suitable for message passing. Concretely, it alternates between rewiring the graph and denoising the node features to maximize a spectral alignment between the top- L eigenvectors of the adjacency matrix and the top- L singular vectors of the original node features, thereby facilitating coherent long-range propagation.

4.3 Practical Trade-offs: Structural vs. Feature-aware Rewiring, Efficiency, and Hyperparameter Sensitivity

Performance attribution. A central practical issue is attribution: whether the observed gains truly reflect the effect of changing connectivity, or whether they partly arise from feature or model-dependent effects. While feature-aware rewiring methods often deliver stronger empirical performance precisely because they exploit task-relevant signals encoded in the node attributes, their improvements can be harder to interpret causally. Structural rewiring enables a more straightforward attribution, since the edited graph is determined from G alone; this makes it easier to relate improvements to explicit connectivity patterns and to use rewiring as a controlled probe of the role of structure in message passing. In contrast, feature-aware rewiring can be beneficial when the edges are not the most informative support for the task, but the interpretation of improvements becomes less specific: the rewiring signal is coupled to the feature space and may therefore act as an additional modeling step, so that improved accuracy cannot be attributed to topology reconstruction alone.

Hyperparameters. The interpretability of structural-fix rewiring is further complicated by two recurring limitations: sensitivity to rewiring hyperparameters and computational overhead. Most structural-fix rewiring methods introduce additional design choices, most commonly an edit budget (the number of edges added and/or removed) or threshold rules [Topping *et al.*, 2022; Karhadkar *et al.*, 2023; Nguyen *et al.*, 2023; Black *et al.*, 2023; Giraldo *et al.*, 2023; Barbero *et al.*, 2024]. Empirically, the benefits of curvature-based rewiring on real-world benchmarks are highly sensitive to both training and rewiring hyperparameters and vary across graphs; reported SOTA gains often arise from favorable hy-

perparameter configurations rather than consistent improvements over the original topology [Tori *et al.*, 2025]. Only a subset of approaches aims to reduce this tuning burden, for instance, by calibrating curvature-based decisions through mixture modeling [Fesser and Weber, 2023] or by adopting parameter-free geometric constructions in feature space [Attali *et al.*, 2024; Attali *et al.*, 2025b].

Complexity. Finally, scalability is often constrained by the cost of the underlying descriptors: depending on the estimator, curvature-based criteria can scale quadratically to cubically in the number of edges [Topping *et al.*, 2022; Nguyen *et al.*, 2023], effective resistance objectives can scale cubically in the number of nodes [Black *et al.*, 2023; Liang *et al.*, 2025], and spectral criteria require Laplacian decompositions [Karhadkar *et al.*, 2023]. In contrast, several approaches rely on descriptors and constructions that are more efficient to compute and do not scale quadratically with the number of nodes [Klicpera *et al.*, 2019; Alon and Yahav, 2021; Barbero *et al.*, 2024; Attali *et al.*, 2024; Linkerhäger *et al.*, 2025]. As a result, these methods typically incur substantially lower overhead in practice and are easier to deploy on large graphs.

4.4 Evaluation

Rewiring should not be evaluated solely through downstream accuracy, since its primary goal is to modify the graph geometry to improve information propagation. In practice, one typically compares a fixed backbone GNN trained and evaluated on the original graph G and on the rewired graph G^+ under a matched protocol. Beyond task-level scores, evaluation can include structure-level diagnostics that assess whether G^+ attenuates connectivity patterns known to exacerbate over-squashing and over-smoothing (e.g., shifts in edge-curvature distributions, decreases in effective resistance, or reductions in commute times), as well as model-level diagnostics that quantify whether long-range sensitivity has improved, for instance via a Jacobian-based sensitivity measure between distant nodes. Task-level validation is typically reported on node and graph prediction benchmarks, with particular emphasis on heterophilic datasets [Lim *et al.*, 2021; Platonov *et al.*, 2023] and on long-range interaction benchmarks [Dwivedi *et al.*, 2022], which directly stress the ability of GNNs to transmit signals across distant nodes.

5 Discussion

Recommendations. We suggest articulating rewiring contributions through three complementary criteria. First, the method should be motivated by the effective radius of the task, i.e., whether the predictive signal is expected to depend on information beyond a small neighborhood; when this is not the case, topology changes primarily targeting long-range communication may not be the most appropriate intervention. Second, the evaluation should clarify the role of the input graph [Bechler-Speicher *et al.*, 2025] by including a graph-agnostic baseline, such as DeepSets [Zaheer *et al.*, 2017], thereby distinguishing the benefits that truly arise from the relational structure from those attainable without the given connectivity. Additionally, a single rewiring method is unlikely

Models	Venues	OSQ	OSM	UF	DT	Measure
DIGL	[NeurIPS 2019] [Klicpera <i>et al.</i> , 2019]	✗	✗	✗	✗	PageRank
DropEdge	[ICLR 2020] [Rong <i>et al.</i> , 2019]	✗	✓	✗	✓	Random
PPRGo	[KDD 2020] [Bojchevski <i>et al.</i> , 2020]	✗	✗	✗	✗	PageRank
FA	[ICLR 2021] [Alon and Yahav, 2021]	✓	✗	✗	✗	Connectivity
SDRF	[ICLR 2022] [Topping <i>et al.</i> , 2022]	✓	✗	✗	✗	Balanced Forman curvature
RLEF	[IEEE AACCC 2022] [Banerjee <i>et al.</i> , 2022]	✓	✗	✗	✗	Cheeger constant
DiffWire	[LoG 2022] [Arnaiz-Rodríguez <i>et al.</i> , 2022]	✓	✗	✓	✓	Commute Time and Spectral Gap
EGP	[LoG 2022] [Deac <i>et al.</i> , 2022]	✓	✗	✗	✗	Expander graph
FOSR	[ICLR 2023] [Karhadkar <i>et al.</i> , 2023]	✓	✓	✗	✗	Spectral gap
BORF	[ICML 2023] [Nguyen <i>et al.</i> , 2023]	✓	✓	✗	✗	Ollivier curvature
GTR	[ICML 2023] [Black <i>et al.</i> , 2023]	✓	✗	✗	✗	Effective resistance
SJLR	[CIKM 2023] [Giraldo <i>et al.</i> , 2023]	✓	✓	✓	✓	Jost–Liu curvature
AFR-3	[LoG 2023] [Fesser and Weber, 2023]	✓	✓	✗	✗	Augmented Forman curvature
PR-MPNNs	[ICLR 2024] [Qian <i>et al.</i> , 2023]	✓	✗	✓	✓	Probabilistic learned rewiring
LASER	[ICLR 2024] [Barbero <i>et al.</i> , 2024]	✓	✗	✗	✗	Random walk
DR	[ICML 2024] [Attali <i>et al.</i> , 2024]	✓	✓	✓	✗	Delaunay triangulation
Pr Add/Delete	[NeurIPS 2024] [Jamadandi <i>et al.</i> , 2024]	✓	✓	✗	✗	Spectral gap
IPR-MPNN	[NeurIPS 2024] [Qian <i>et al.</i> , 2024]	✓	✗	✓	✓	Virtual nodes
CGP	[LoG 2024] [Wilson <i>et al.</i> , 2024]	✓	✗	✗	✗	Cayley expander
ComFy	[ICLR 2025] [Rubio-Madrigal <i>et al.</i> , 2025]	✓	✓	✓	✗	Feature similarity
VN	[ICLR 2025] [Southern <i>et al.</i> , 2025]	✓	✗	✗	✗	Virtual node
JDR	[ICLR 2025] [Linkerhäger <i>et al.</i> , 2025]	✓	✗	✓	✗	Spectral alignment
TRIGON	[CIKM 2025] [Attali <i>et al.</i> , 2025b]	✓	✓	✓	✓	Triangulation learning
GOKU	[ICML 2025] [Liang <i>et al.</i> , 2025]	✓	✗	✓	✗	Effective resistance
FN	[AAAI 2026] [Choi <i>et al.</i> , 2025]	✓	✗	✗	✗	Fractal node

Table 1: Comparison of various rewiring methods. OSQ: over-squashing; OSM: over-smoothing; UF: uses node features for rewiring; DT: rewiring performed during training (dynamic).

to be appropriate across diverse applications, graph families, and tasks. By altering aggregation neighborhoods, rewiring implicitly determines which relations are treated as informative for the downstream objective. This choice depends on the application-specific meaning of edges and on the task’s characteristic dependency scale in the graph. Accordingly, a rule that benefits one regime may harm another by violating domain constraints or by aggregating incompatible feature evidence. Third, reported gains should be supported by a joint analysis of effectiveness, computational cost, and attribution: downstream improvements should be reported together with the edit rewiring budget, runtime and memory overhead, and matched-budget controls (e.g., random or degree-preserving rewiring baselines, and ablations for feature-driven constructions) so that the source of the improvement is identifiable and comparisons across methods are meaningful.

Future directions. Building on the recommendations above, we outline broader research directions in which rewiring serves not only as an optimization tool, but also as a controlled structural intervention for understanding how topology shapes information propagation and learning in GNNs. A key direction is to better understand the interplay between graph structure and node features. Since message passing couples features through the graph, connectivity changes directly modulate how attribute information is propagated and combined, which can either reinforce or conflict

with the task signal. An important open problem is to clarify the respective roles of structure and features in long-range learning: when feature side mechanisms already convey the non-local signal [Hosseini *et al.*, 2025], when rewiring is indispensable, and how to combine both under matched budgets and constraints. More broadly, rewiring enables controlled neighborhood perturbations to study this interplay; yet principled criteria for predicting when structure and features are aligned or when their mismatch induces harmful mixing remain largely open [Gomes *et al.*, 2022].

Additionally, graph structure can also shape generalization through the stability of message passing, by controlling how strongly neighborhood aggregation propagates and amplifies perturbations across layers. Recent analyses make this link explicit by expressing bound constants in terms of structural quantities such as degree heterogeneity and spectral properties of the graph filter [Ju *et al.*, 2023]. This suggests a promising direction: leverage such bound-relevant structural proxies to anticipate when topology is likely to degrade generalization, and to guide rewiring toward controlling propagation amplification rather than optimizing accuracy alone.

References

[Akansha, 2025] Singh Akansha. Over-squashing in graph neural networks: A comprehensive survey. *Neurocomputing*, page 130389, 2025.

- [Alon and Milman, 1984] Noga Alon and Vitali D Milman. Eigenvalues, expanders and superconcentrators. In *FOCS*, pages 320–322, 1984.
- [Alon and Yahav, 2021] Uri Alon and Eran Yahav. On the bottleneck of graph neural networks and its practical implications. In *ICLR*, 2021.
- [Arnaiz-Rodríguez *et al.*, 2022] Adrián Arnaiz-Rodríguez, Ahmed Begga, Francisco Escolano, and Nuria Oliver. Diffwire: Inductive graph rewiring via the lov’sasz bound. *arXiv preprint arXiv:2206.07369*, 2022.
- [Attali *et al.*, 2024] Hugo Attali, Davide Buscaldi, and Nathalie Pernelle. Delaunay graph: Addressing over-squashing and over-smoothing using delaunay triangulation. In *ICML*, 2024.
- [Attali *et al.*, 2025a] Hugo Attali, Davide Buscaldi, and Nathalie Pernelle. Curvature constrained mpnns: Improving message passing with local structural properties. *Data & Knowledge Engineering*, 156:102382, 2025.
- [Attali *et al.*, 2025b] Hugo Attali, Thomas Papastergiou, Nathalie Pernelle, and Fragkiskos D Malliaros. Dynamic triangulation-based graph rewiring for graph neural networks. In *CIKM*, pages 87–97, 2025.
- [Banerjee *et al.*, 2022] Pradeep Kr Banerjee, Kedar Karhadkar, Yu Guang Wang, Uri Alon, and Guido Montúfar. Oversquashing in gnns through the lens of information contraction and graph expansion. In *AACCC*. IEEE, 2022.
- [Barbero *et al.*, 2024] Federico Barbero, Ameya Velingker, Amin Saberi, Michael M. Bronstein, and Francesco Di Giovanni. Locality-aware graph rewiring in GNNs. In *ICLR*, 2024.
- [Bechler-Speicher *et al.*, 2025] Maya Bechler-Speicher, Ben Finkelshtein, Fabrizio Frasca, Luis Müller, Jan Tönshoff, Antoine Siraudin, Viktor Zaverkin, Michael M. Bronstein, Mathias Niepert, Bryan Perozzi, Mikhail Galkin, and Christopher Morris. Position: Graph learning will lose relevance due to poor benchmarks. In *ICML*, 2025.
- [Black *et al.*, 2023] Mitchell Black, Zhengchao Wan, Amir Nayyeri, and Yusu Wang. Understanding oversquashing in GNNs through the lens of effective resistance. In *ICML*, pages 2528–2547. PMLR, 2023.
- [Bojchevski *et al.*, 2020] Aleksandar Bojchevski, Johannes Gasteiger, Bryan Perozzi, Amol Kapoor, Martin Blais, Benedek Rózemerczki, Michal Lukasik, and Stephan Günnemann. Scaling graph neural networks with approximate pagerank. In *KDD*, 2020.
- [Cai and Wang, 2020] Chen Cai and Yusu Wang. A note on over-smoothing for graph neural networks. *Graph Representation Learning*, 2020.
- [Chandra *et al.*, 1989] Ashok K Chandra, Prabhakar Raghavan, Walter L Ruzzo, and Roman Smolensky. The electrical resistance of a graph captures its commute and cover times. In *STOC*, 1989.
- [Cheeger, 1970] Jeff Cheeger. A lower bound for the smallest eigenvalue of the laplacian. In *Problems in Analysis: A Symposium in Honor of Salomon Bochner (PMS-31)*. Princeton University Press, 1970.
- [Choi *et al.*, 2025] Jeongwhan Choi, Seungjun Park, Sumin Park, Sung-Bae Cho, and Noseong Park. Are graph transformers necessary? efficient long-range message passing with fractal nodes in mpnns. *arXiv*, 2025.
- [Chung and Graham, 1997] Fan RK Chung and Fan Chung Graham. *Spectral graph theory*. American Mathematical Soc., 1997.
- [Deac *et al.*, 2022] Andreea Deac, Marc Lackenby, and Petar Veličković. Expander graph propagation. In *Learning on Graphs Conference*, pages 38–1. PMLR, 2022.
- [Di Giovanni *et al.*, 2023] Francesco Di Giovanni, Lorenzo Giusti, Federico Barbero, Giulia Luise, Pietro Lio, and Michael M Bronstein. On over-squashing in message passing neural networks: The impact of width, depth, and topology. In *ICML*, pages 7865–7885. PMLR, 2023.
- [Dwivedi *et al.*, 2022] Vijay Prakash Dwivedi, Ladislav Rampásek, Michael Galkin, Ali Parviz, Guy Wolf, Anh Tuan Luu, and Dominique Beaini. Long range graph benchmark. *NeurIPS*, 2022.
- [Fesser and Weber, 2023] Lukas Fesser and Melanie Weber. Mitigating over-smoothing and over-squashing using augmentations of forman-ricci curvature. In *LoG*, 2023.
- [Forman, 2003] Robin Forman. Bochner’s method for cell complexes and combinatorial ricci curvature. 2003.
- [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 *ICML*, 2017.
- [Giraldo *et al.*, 2023] Jhony H. Giraldo, Konstantinos Skianis, Thierry Bouwmans, and Fragkiskos D. Malliaros. On the trade-off between over-smoothing and over-squashing in deep graph neural networks. In *CIKM*, pages 566–576, 2023.
- [Gomes *et al.*, 2022] Diana Gomes, Frederik Ruelens, Kyriakos Efthymiadis, Ann Nowe, and Peter Vrancx. When are graph neural networks better than structure-agnostic methods? In *I Can’t Believe It’s Not Better Workshop: Understanding Deep Learning Through Empirical Falsification*, 2022.
- [Gong *et al.*, 2026] Cheng-Hua Gong, Yao Cheng, Jian-Xiang Yu, Can Xu, Cai-Hua Shan, Si-Qiang Luo, and Xi-ang Li. A survey on learning from graphs with heterophily: recent advances and future directions. *Frontiers of Computer Science*, 20(2):2002314, 2026.
- [Gori *et al.*, 2005] Marco Gori, Gabriele Monfardini, and Franco Scarselli. A new model for learning in graph domains. In *IJCNN*, volume 2, pages 729–734, 2005.
- [Hosseini *et al.*, 2025] Ryien Hosseini, Filippo Simini, Venkatram Vishwanath, Rebecca Willett, and Henry Hoffmann. Sketch-augmented features improve learning long-range dependencies in graph neural networks. In *NeurIPS*, 2025.

- [Jamadandi *et al.*, 2024] Adarsh Jamadandi, Celia Rubio-Madriral, and Rebekka Burkholz. Spectral graph pruning against over-squashing and over-smoothing. *NeurIPS*, 2024.
- [Jost and Liu, 2014] Jürgen Jost and Shiping Liu. Ollivier’s ricci curvature, local clustering and curvature-dimension inequalities on graphs. *Discrete & Computational Geometry*, 51(2):300–322, 2014.
- [Ju *et al.*, 2023] Haotian Ju, Dongyue Li, Aneesh Sharma, and Hongyang R Zhang. Generalization in graph neural networks: Improved pac-bayesian bounds on graph diffusion. In *AISTATS*. PMLR, 2023.
- [Karhadkar *et al.*, 2023] Kedar Karhadkar, Pradeep Kr Banerjee, and Guido Montúfar. Foser: First-order spectral rewiring for addressing oversquashing in gnns. In *ICLR*, 2023.
- [Klicpera *et al.*, 2019] Johannes Klicpera, Stefan Weißenberger, and Stephan Günnemann. Diffusion improves graph learning. In *NeurIPS*, 2019.
- [Liang *et al.*, 2025] Langzhang Liang, Fanchen Bu, Zixing Song, Zenglin Xu, Shirui Pan, and Kijung Shin. Mitigating over-squashing in graph neural networks by spectrum-preserving sparsification. In *ICML*, 2025.
- [Lim *et al.*, 2021] Derek Lim, Felix Hohne, Xiuyu Li, Sijia Linda Huang, Vaishnavi Gupta, Omkar Bhalerao, and Ser Nam Lim. Large scale learning on non-homophilous graphs: New benchmarks and strong simple methods. In *NeurIPS*, 2021.
- [Linkerhägner *et al.*, 2025] Jonas Linkerhägner, Cheng Shi, and Ivan Dokmanić. Joint graph rewiring and feature denoising via spectral resonance. In *ICLR*, 2025.
- [Nguyen *et al.*, 2023] Khang Nguyen, Nong Minh Hieu, Vinh Duc Nguyen, Nhat Ho, Stanley Osher, and Tan Minh Nguyen. Revisiting over-smoothing and over-squashing using ollivier-ricci curvature. In *ICML*, 2023.
- [Ollivier, 2007] Yann Ollivier. Ricci curvature of metric spaces. *Comptes Rendus Mathématique*, 345(11):643–646, 2007.
- [Platonov *et al.*, 2023] Oleg Platonov, Denis Kuznedelev, Michael Diskin, Artem Babenko, and Liudmila Prokhorenkova. A critical look at the evaluation of gnns under heterophily: are we really making progress? In *ICLR*, 2023.
- [Qian *et al.*, 2023] Chendi Qian, Andrei Manolache, Kareem Ahmed, Zhe Zeng, Guy Van den Broeck, Mathias Niepert, and Christopher Morris. Probabilistically rewired message-passing neural networks. *arXiv preprint arXiv:2310.02156*, 2023.
- [Qian *et al.*, 2024] Chendi Qian, Andrei Manolache, Christopher Morris, and Mathias Niepert. Probabilistic graph rewiring via virtual nodes. In *NeurIPS*, 2024.
- [Rong *et al.*, 2019] Yu Rong, Wenbing Huang, Tingyang Xu, and Junzhou Huang. Dropedge: Towards deep graph convolutional networks on node classification. In *ICLR*, 2019.
- [Rubio-Madriral *et al.*, 2025] Celia Rubio-Madriral, Adarsh Jamadandi, and Rebekka Burkholz. GNNs getting comfy: Community and feature similarity guided rewiring. In *ICLR*, 2025.
- [Rusch *et al.*, 2023] T Konstantin Rusch, Michael M Bronstein, and Siddhartha Mishra. A survey on over-smoothing in graph neural networks. *arXiv preprint arXiv:2303.10993*, 2023.
- [Samal *et al.*, 2018] Areejit Samal, RP Sreejith, Jiao Gu, Shiping Liu, Emil Saucan, and Jürgen Jost. Comparative analysis of two discretizations of ricci curvature for complex networks. *Scientific reports*, 8(1):8650, 2018.
- [Scarselli *et al.*, 2008] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. In *transactions on neural networks*. IEEE, 2008.
- [Southern *et al.*, 2025] Joshua Southern, Francesco Di Giovanni, Michael M. Bronstein, and Johannes F. Lutzeyer. Understanding virtual nodes: Oversquashing and node heterogeneity. In *ICLR*, 2025.
- [Topping *et al.*, 2022] Jake Topping, Francesco Di Giovanni, Benjamin Paul Chamberlain, Xiaowen Dong, and Michael M Bronstein. Understanding over-squashing and bottlenecks on graphs via curvature. *ICLR*, 2022.
- [Tori *et al.*, 2025] Floriano Tori, Vincent Holst, and Vincent Ginis. The effectiveness of curvature-based rewiring and the role of hyperparameters in GNNs revisited. In *ICLR*, 2025.
- [Wilson *et al.*, 2024] JJ Wilson, Maya Bechler-Speicher, and Petar Veličković. Cayley graph propagation. *arXiv preprint arXiv:2410.03424*, 2024.
- [Zaheer *et al.*, 2017] Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R Salakhutdinov, and Alexander J Smola. Deep sets. *Advances in neural information processing systems*, 30, 2017.
- [Zheng *et al.*, 2022] Xin Zheng, Yi Wang, Yixin Liu, Ming Li, Miao Zhang, Di Jin, Philip S Yu, and Shirui Pan. Graph neural networks for graphs with heterophily: A survey. *arXiv preprint arXiv:2202.07082*, 2022.
- [Zhou *et al.*, 2020] Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. Graph neural networks: A review of methods and applications. *AI open*, 2020.
- [Zhu *et al.*, 2020] Jiong Zhu, Yujun Yan, Lingxiao Zhao, Mark Heimann, Leman Akoglu, and Danai Koutra. Beyond homophily in graph neural networks: Current limitations and effective designs. In *NeurIPS*, 2020.