Perelyn

IOCMA 2026 Conference — The 2nd International Online Conference on Mathematics and Applications — 10-12 June 2026 | Online — MDPI

Curvature-guided Graph-to-Hypergraph Structural Lifting Alleviates Over-Squashing in Graph Learning

M. Banf

Perelyn GmbH, Munich, Bavaria, Germany

michael.banf@perelyn.com

Summary

Many real-world systems exhibit complex, higher-order interactions that standard GNNs cannot capture thus motivating Topological Deep Learning and the concept of *lifting*. We propose a structural lifting based on Forman-Ricci curvature, which identifies network backbones, i.e. coarse, structure-preserving geometries, and maps them as hyperedges into a hypergraph $\mathcal{H}$, virtually shortening distances between relevant nodes and alleviating over-squashing. Using GCN and GAT, lifting improves performance in 75% of cases, while hypergraph models (EDGNN, AllSetTransformer) benefit in 80% of cases.

The challenge: over-squashing in graph learning

Recent research [1–3] shows that certain graph structures are inherently ill-suited for effective message passing. Specifically, MPNNs (e.g., GCN, GAT) [4] struggle in capturing long-range dependencies, where model outputs rely on interactions between distant nodes [5]. When messages from distant nodes must traverse few paths and compress into fixed-size representations, critical signals are lost — a phenomenon known as over-squashing [2, 3, 5] (see figure 1).

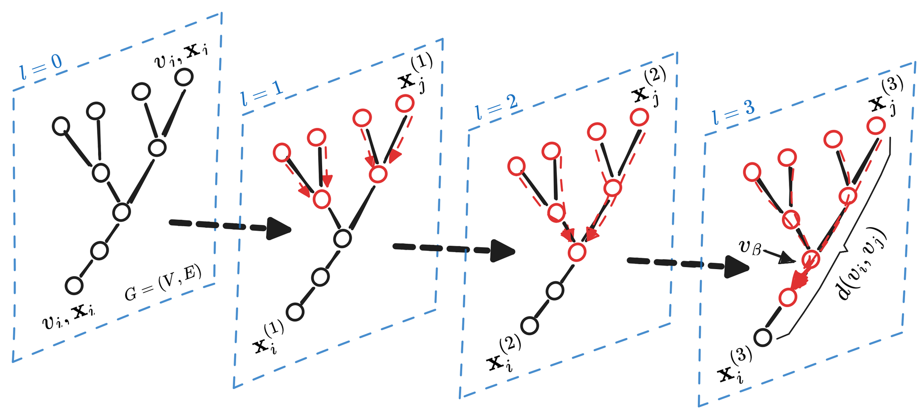

Figure 1: Layer-wise information propagation along edges of G. Due to G's structure, node v_β forms information bottleneck. In addition, as $d(v_i, v_j) = 4$ but number of layers $l = 3$, information from v_j never reaches v_i.

To this end, over-squashing can be quantified as the lack of sensitivity of an MPNN at a given node to the input features of another remote node [1, 6]. Consider $G = (V, E)$ with node features $\mathbf{x}_i \in \mathbb{R}^m$ for each $v_i \in V$; a general MPNN updates features layer-wise by:

$$\mathbf{x}_i^{(0)} := \mathbf{x}_{v_i}, \quad \mathbf{x}_i^{(l)} = \phi^{(l)}\left(\mathbf{x}_i^{(l-1)}, \bigoplus_{j\in\mathcal{N}(v_i)} \psi^{(l)}\left(\mathbf{x}_i^{(l-1)}, \mathbf{x}_j^{(l-1)}, \mathbf{e}_{i,j}\right)\right) \quad (1)$$

Here, $\mathbf{x}_i^{(l)}$ and $\mathbf{x}_i^{(l-1)}$ denote the l-th and l-1-th layers' node features representations of v_i, $\bigoplus$ is a differentiable, permutation invariant function, e.g., sum or mean, ψ and ϕ are differentiable message aggregation and node update functions, $\mathcal{N}(v_i)$ is the node neighborhood of v_i and $\mathbf{e}_{i,j}$ denote some (optional) edge features from node v_j to v_i. The receptive field $\mathcal{R}^l$ at the l-th layer is defined as $\mathcal{R}^l\{v_i, v_j \in V : d(v_i, v_j) \leq l\}$, where $d(v_i, v_j)$ represents the shortest path distance in G between v_i and v_j. The Jacobian $\partial \mathbf{x}_i^{(l)} / \partial \mathbf{x}_j$ quantifies the sensitivity of representation $\mathbf{x}_i^{(l)}$ at v_i to a specific input feature $\mathbf{x}_j$ at v_j within $\mathcal{R}_l$. Small absolute values of this Jacobian are indicative of poor information propagation, or over-squashing, i.e. the incapacity of $\mathbf{x}_i^{(l)}$ to be influenced by $\mathbf{x}_j$ at a distance of l. Notably, this Jacobian is upper-bounded by powers of G's adjacency matrix, so G's structure directly governs over-squashing [1, 6].

The metric: curvature on graphs

Analyzing the geometric properties of G has been proposed to better understand over-squashing [1]. One such property is curvature, which, in its continuous form, measures how space bends locally. In two dimensions, Gaussian curvature K defines whether a surface is flat (Euclidean), positively curved (e.g. a sphere), or negatively curved (e.g. a saddle) (figure 2).

Such curvature notions extend to graphs [1], linking local geometry to global structure. As an analogy of geodesic dispersion, i.e. whether two parallel geodesics starting from nearby points converge ($K > 0$), remain parallel ($K = 0$), or diverge ($K < 0$), on graphs, consider an edge with two edges starting at its respective endpoints. In a discrete spherical geometry (e.g. a clique graph), the edges would meet at another node to form a triangle. In a discrete Euclidean geometry (e.g. a grid), the edges would stay parallel and form a rectangle or grid, whereas in a discrete hyperbolic geometry (e.g. a tree), the mutual distance of the edge endpoints would increase.

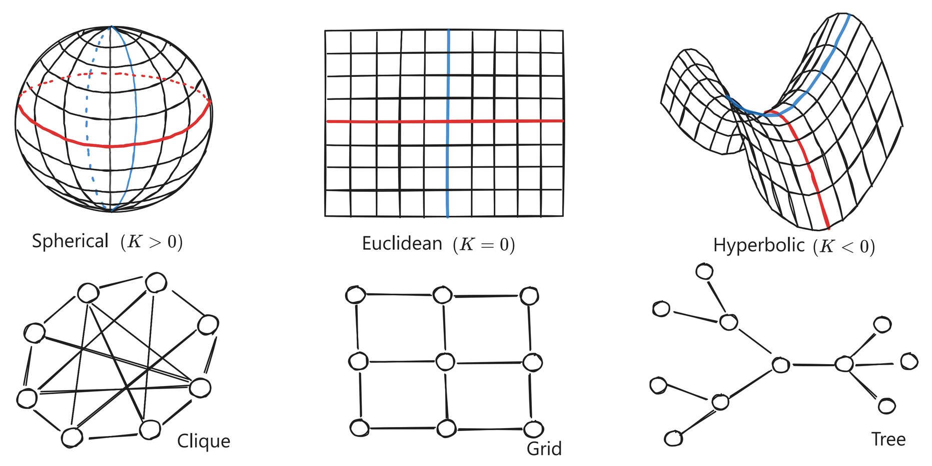

Figure 2: In differential geometry, curvature on manifolds, such as Gaussian curvature K, relates to information propagation. Replacing geodesics for edges, one can find discrete graph analogous for different types of curvature.

To measure curvature on graphs, a variety of metrics have been proposed, predominantly Ollivier-Ricci [7] as well as several variations of Forman-Ricci curvature [1, 2, 8, 9]. In particular, Ollivier-Ricci curvature [10] translates previously described notions of positive and negative curvature directly to the discrete domain [1]. Forman-Ricci correlates with Ollivier-Ricci [9] but scales more efficiently to large networks. However, the purely edge-based Forman-Ricci [8, 9, 11] biases towards negative values, hence lacks the intuition of Ollivier-Ricci curvature. Concretely, highly negative values tend to describe edges within clusters whereas values near zero indicate bridges between clusters.

The remedy: graph-to-hypergraph structural lifting

While curvature-guided local graph rewiring has been proposed to address bottlenecks [1, 2], one fundamental restriction of graphs is their limitation to pairwise relations. Topological Deep Learning instead encodes higher-order relationships beyond pairwise constraints [12].

We propose a remedy that uses Forman-Ricci curvature to identify network backbones, i.e. structure-preserving, coarse geometries connecting major communities. We then perform a structural lifting [13, 14] $\gamma : G \to \mathcal{H}$, treating backbones as hyperedges to virtually shorten distances between relevant nodes.

$$\mathrm{Ric}_{\mathrm{F}}(e) = \omega(e)\left(\frac{\omega(v_i)}{\omega(e)} + \frac{\omega(v_j)}{\omega(e)} - \sum_{\substack{e_{v_i} \sim e \\ e_{v_j} \sim e}} \left[\frac{\omega(v_i)}{\sqrt{\omega(e)\omega(e_{v_i})}} + \frac{\omega(v_j)}{\sqrt{\omega(e)\omega(e_{v_j})}}\right]\right) \quad (2)$$

where e_{v_i}, e_{v_j} are the edges connected to nodes v_i and v_j. For an unweighted network graph, i.e. all edge weights $w = 1$, equation 2 reduces to $Ric_F(e) = 4 - \deg(v_i) - \deg(v_j)$. Thus $Ric_F(e) << 0$ if both v_i and v_j have high degrees. In particular, edges with high negative curvature, i.e. $Ric_F(e) << 0$, should play a special role for the spreading out and hence for, e.g., information dispersal in a network, whereas edges with $Ric_F(e) \approx 0$ rather constitute long-range interactions, bottlenecks, interfaces or boundaries (see figure 3).

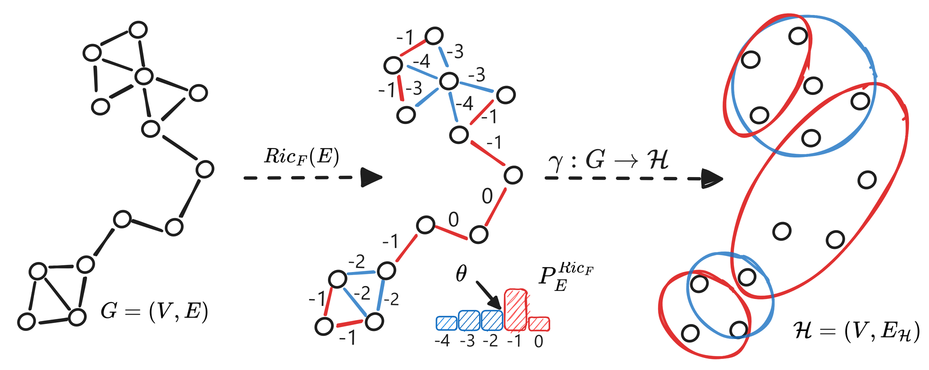

Figure 3: Forman-Ricci curvature based structural lifting $\gamma : G \to \mathcal{H}$ given $\theta = -1$ for an unweighted graph G. Red and blue hyper-edges correspond to $Ric_F(e) \geq \theta$ and $Ric_F(e) < \theta$, respectively.

Subsequently, we use $Ric_F(e)$ values per each e to define a structural lifting $\gamma : G \to \mathcal{H}$, mapping $G = (V, E)$ to $\mathcal{H} = (V, \mathcal{E}_\mathcal{H})$, where $\mathcal{H}$ generalizes G by allowing hyperedges to connect any number of nodes [13, 14].

Hyperparameter θ separates two hyperedge types: cluster edges ($Ric_F(e) < \theta$) and long-range/bottleneck edges ($Ric_F(e) \geq \theta$); θ can be set as a quantile of the empirical curvature distribution (figure 3).

We then define a hyperedge $e_{S,\theta} \in \mathcal{E}_\mathcal{H}$ in $\mathcal{H}$ where S denotes a subset of V in G. For each node $v_i \in S$, v_i is an endpoint to an edge e in G with $Ric_F(e) < \theta$ (or $Ric_F(e) \geq \theta$ respectively), and $\exists v_j \in S$, with $v_j \in \mathcal{N}_1(v_i)$, i.e. v_i, v_j are connected via e in G. Figure 3 illustrates $\gamma : G \to \mathcal{H}$ given $\theta = -1$.

Experiments

We evaluate on graph classification benchmarks and a molecular property prediction dataset (table 1). Results are averaged across four metrics (precision, recall, AUROC, accuracy). GCN and GAT consistently improve on lifted vs. non-lifted data in 75% (6/8) of cases. Hypergraph models (EDGNN, AllSetTransformer, UniGNN2) also benefit, improving in 80% (10/12) of cases.

Dataset / Method	GCN	GAT	AST	EDGNN	UniGNN2
NCI1	0.71 ± 0.03	0.73 ± 0.04	0.65 ± 0.07	0.70 ± 0.06	0.63 ± 0.08
NCI1$_{Ric_F}$	**0.72 ± 0.03**	0.73 ± 0.04	**0.68 ± 0.05**	**0.71 ± 0.04**	**0.72 ± 0.03**
NCI109	0.70 ± 0.03	0.71 ± 0.04	0.70 ± 0.06	0.68 ± 0.09	0.67 ± 0.09
NCI109$_{Ric_F}$	**0.71 ± 0.03**	**0.72 ± 0.04**	0.69 ± 0.07	**0.69 ± 0.07**	**0.70 ± 0.05**
OGBG Molhiv	0.78 ± 0.14	0.77 ± 0.17	0.78 ± 0.20	0.76 ± 0.17	0.78 ± 0.15
OGBG Molhiv$_{Ric_F}$	**0.79 ± 0.13**	0.77 ± 0.17	**0.79 ± 0.19**	**0.78 ± 0.16**	0.78 ± 0.16
PROTEINS	0.69 ± 0.04	0.69 ± 0.04	0.66 ± 0.08	0.67 ± 0.07	0.69 ± 0.06
PROTEINS$_{Ric_F}$	**0.70 ± 0.02**	**0.70 ± 0.04**	**0.70 ± 0.07**	**0.73 ± 0.04**	**0.70 ± 0.09**

Table 1: Performance comparison on original and lifted datasets across several graph and hypergraph neural network architectures.

Implementation

An earlier version of our curvature-based structural lifting has been made available as part of TopoBench [15] a modular Python library designed to standardize benchmarking and accelerate research in Topological Deep Learning.

References

[1] J. Topping, F. Di Giovanni, et al., Understanding over-squashing and bottlenecks on graphs via curvature, ICLR (2022)

[2] H. Attali, D. Buscaldi, N. Pernelle, Rewiring Techniques to Mitigate Oversquashing and Oversmoothing in GNNs: A Survey, arXiv (2024)

[3] S. Akansha, Over-squashing in Graph Neural Networks: A comprehensive survey, Neurocomputing (2025)

[4] B. Nguyen, A. Yodaiken, P. Veličković: Alignment of MPNNs and Graph Transformers, PMLR (2024)

[5] Dwivedi, V. et al., Long Range Graph Benchmark, NeurIPS (2022)

[6] M. Black, et al., Understanding Oversquashing in GNNs through the Lens of Effective Resistance, CoRR (2023)

[7] J. Sia, E. Jonckheere, P. Bogdan, Ollivier-Ricci Curvature-Based Method to Community Detection in Complex Networks, Sci Rep (2019)

[8] E. Saucan, et al., Discrete Curvatures and Network Analysis, MATCH (2018)

[9] A. Samal, et al., Comparative analysis of two discretizations of Ricci curvature for complex networks, Sci Rep (2018)

[10] Y. Ollivier, Ricci curvature of Markov chains on metric spaces, J. Funct. Anal. (2009)

[11] M. Weber, E. Saucan, J. Jost, Characterizing complex networks with Forman-Ricci curvature and associated geometric flows, Journal of Complex Networks (2017)

[12] G. Bernárdez et al. (2024) ICML Topological Deep Learning Challenge 2024: Beyond the Graph Domain, PMLR (2024)

[13] M. Papillon et al., Architectures of topological deep learning: a survey on topological neural networks, arXiv (2023)

[14] M. Hajij, et al., Topological deep learning: going beyond graph data, NeurIPS (2023)

[15] L. Telyatnikov, et al., TopoBench: A Framework for Benchmarking Topological Deep Learning, arXiv (2024)