Look Before You Lift: Visual and Quantitative Diagnostics for Topological Deep Learning

Mathilde Papillon* PAPILLON@UCSB.EDU
Guillermo Bernárdez* GUILLERMO_BERNARDEZ@UCSB.EDU
Álvaro Ballón Barreiro
University of California Santa Barbara

Marco Montagna
Sapienza Universitá di Roma

Rémi Devaux
Antoine Jardin
Arlequin AI

Nina Miolane
University of California Santa Barbara

Abstract

Topological deep learning (TDL) methods rely on lifting raw data into higher-order discrete domains such as simplicial complexes, cell complexes, and hypergraphs. In practice, this lifting step is often treated as a black box: practitioners select a lifting and then tune architectures, with limited visibility into whether the induced higher-order connectivity is meaningful for the downstream task. To address this missing diagnostic layer, we propose a visualization technique called TopoExplorer that leverages the strictly augmented Hasse graph form of topological datasets for exploratory data analysis. For the first time, practitioners can easily visualize the incidence- and adjacency-based neighborhoods that define the lifted dataset, as well as read off key graph metrics that describe its structural and feature landscape. Via an extensive set of experiments across many datasets and liftings, we show that several of these metrics correlate with downstream model performance, suggesting they can help inform TDL preprocessing design. Our perspective reframes the TDL workflow from *lift-train* to *lift-look-design-train*, enabling more principled, interpretable, and efficient model development. TopoExplorer is hosted at topoexplorer.pagekite.me/, and its source code is available at github.com/geometric-intelligence/topoexplorer.

Keywords: Topological Deep Learning, Exploratory Data Analysis, Lifting, Graph Metrics.

1. Introduction

Many real-world systems involve interactions among groups that go beyond pairwise relationships, like atoms in a molecular ring or groups of friends in a social network. Topological Deep Learning (TDL) (Bodnar, 2023; Hajij et al., 2023) generalizes graph neural networks (GNNs) to model and learn from these higher-order interactions. In this work, we specifically refer to TDL in terms of methods that learn on discrete higher-order domains; methods that use other tools to consider data topology, such as persistent homology, fall outside our scope.

*Equal contribution and co-corresponding authors.

To do so, many TDL pipelines begin with conventional data representations, such as point clouds or graphs, and apply a *lifting procedure*. Intuitively, lifting acts as a structural scaffolding step: it uses connectivity rules or feature-based criteria to explicitly construct higher-order elements, such as mapping cliques of edges to filled simplicial faces. A TDL model is then trained on the resulting topological domain. While this workflow has produced promising empirical results, it also introduces a fundamental source of uncertainty: *what structure did the lifting actually create, and does it reflect the relational patterns that matter for the task?*

In many practical settings, lifting choices are made from a menu of available constructions, often with minimal structural inspection. This issue grows as toolkits make more lifting families available across simplicial, cellular, and hypergraph domains. For instance, TopoBench (Telyatnikov et al., 2025) has demonstrated the breadth of this design space through a systematic benchmark of liftings and tasks across domains. Yet breadth also increases ambiguity: different liftings can induce drastically different connectivity patterns, even on the same dataset. We argue that TDL needs a first-class interface between lifting and learning. Strictly augmented Hasse graph decompositions (introduced in Papillon et al. (2025)) provide such an interface by making incidence- and adjacency-based neighborhoods explicit in graph form, enabling practitioners to inspect how connectivity propagates across cell ranks before training. This is particularly relevant for neighborhood-driven architectures such as TopoTune (Papillon et al., 2025) and HOPSE (Bernárdez et al., 2026). By characterizing lifted connectivity a priori, such an interface would enable practitioners in making better informed lifting and neighborhood design choices.

Contributions. We introduce **TopoExplorer**, a visual and quantitative diagnostic framework grounded in strictly augmented Hasse graph decompositions. The interactive app is hosted at topoexplorer.pagekite.me/ and made available open-source at github.com/geometric-intelligence/topoexplorer. Beyond mere visualization, TopoExplorer allows practitioners to extract key structural and feature-based graph metrics from the lifted complex; it reframes the standard TDL workflow from a blind *lift-train* approach to a more principled *lift-look-design-train* approach. To the best of our knowledge, this is the first exploratory data and visualization framework for better informing lifting design decisions in TDL. Through the analysis of extensive experiments using the default liftings in TopoBench, we show that some pre-training metrics computed in TopoExplorer correlate with downstream model performance, demonstrating the usefulness of such a structure-aware design pipeline.

2. Background

This section introduces key concepts underpinning TDL. For brevity and generality, the following discussion centers on combinatorial complexes, which subsume all of the discrete topological domains leveraged by TDL (Hajij et al., 2023).

Combinatorial Complex. A *combinatorial complex* is a triple $(\mathcal{V}, \mathcal{C}, \mathrm{rk})$ consisting of a set $\mathcal{V}$, a subset $\mathcal{C}$ of the powerset $\mathcal{P}(\mathcal{V})\backslash\{\emptyset\}$, and a rank function $\mathrm{rk} : \mathcal{C} \to \mathbb{Z}_{\geq 0}$ with the following properties:

1. for all $v \in \mathcal{V}, \{v\} \in \mathcal{C}$ and $\mathrm{rk}(\{v\}) = 0$;
2. the function rk is order-preserving, i.e., if $\sigma, \tau \in \mathcal{C}$ satisfy $\sigma \subseteq \tau$, then $\mathrm{rk}(\sigma) \leq \mathrm{rk}(\tau)$.

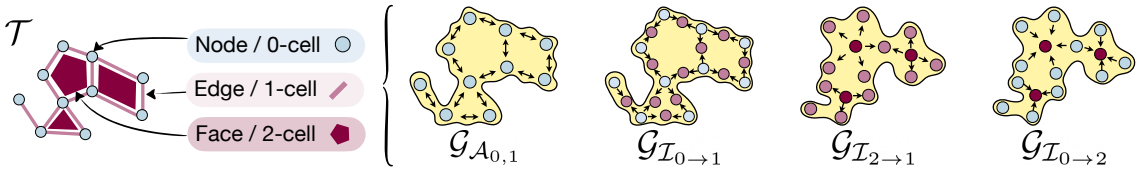

Figure 1: **Neighborhood expansions of a complex.** Given a complex $\mathcal{T}$ (left), three examples of strictly augmented Hasse graphs $\mathcal{G}_{\mathcal{N}}$ corresponding to 4 neighborhood functions: adjacency of nodes w.r.t. edges ($\mathcal{A}_{0,1}$), incidence from nodes to edges ($\mathcal{I}_{0\to1}$), incidence from faces to nodes ($\mathcal{I}_{2\to1}$), and incidence from nodes to faces ($\mathcal{I}_{0\to2}$). Adopted from Bernárdez et al. (2026)

The elements of $\mathcal{V}$ are the nodes, while the elements of $\mathcal{C}$ are called cells (i.e., group of nodes). The rank of a cell $\sigma \in \mathcal{C}$ is $k := \mathrm{rk}(\sigma)$, and we call it a k-cell. $\mathcal{C}$ simplifies notation for $(\mathcal{V}, \mathcal{C}, \mathrm{rk})$, and its dimension is defined as the maximal rank among its cells: $\dim(\mathcal{C}) := \max_{\sigma\in\mathcal{C}} \mathrm{rk}(\sigma)$.

Neighborhood Relations: Adjacencies and Incidences. Combinatorial complexes can be equipped with a notion of neighborhood among cells that induces topological structures [1]. Traditional neighborhood functions include *adjacencies* ($\mathcal{A}_{t,s}$), connecting cells of the same rank t —via relations w.r.t. cells of another rank s— and *incidences* ($\mathcal{I}_{s\to t}$), linking cells of different ranks from a source rank s to a target rank t (see Figure 1 for some visualizations, Appendix A for full definitions).

Strictly augmented Hasse graphs. Given a complex $\mathcal{T}$, each particular neighborhood function $\mathcal{N}$ induces a strictly augmented Hasse graph $\mathcal{G}_{\mathcal{N}} = (\mathcal{C}_{\mathcal{N}}, \mathcal{E}_{\mathcal{N}})$ (Papillon et al., 2025), defined as the directed graph whose nodes and edges are given, respectively, by

$$\mathcal{C}_{\mathcal{N}} = \{\sigma \in \mathcal{C} \mid \mathcal{N}(\sigma) \neq \emptyset\} \quad \text{and} \quad \mathcal{E}_{\mathcal{N}} = \{(\tau, \sigma) \mid \sigma, \tau \in \mathcal{C}_{\mathcal{N}},\ \tau \in \mathcal{N}(\sigma)\}. \tag{1}$$

Figure 1 visually shows some examples of the graph expansions induced by these neighborhood functions. Notably, *strictly* comes from the fact that the set of nodes of $\mathcal{G}_{\mathcal{N}}$ is restricted to $\mathcal{C}_{\mathcal{N}}$, i.e., to those cells that have at least one neighbor according to $\mathcal{N}$.

Lifting. A lifting transformation is a map $\mathcal{L} : \mathcal{D}_{\text{in}} \to \mathcal{D}_{\text{out}}$ between two discrete topological domains, defined by a fixed rule applied uniformly to the input domain. While traditionally conceptualized as promoting lower-order data (e.g., a simple graph) to a higher-order complex, $\mathcal{L}$ more broadly defines any structural translation between topological spaces. Formally, given an input domain $\mathcal{D}_{\text{in}} = (V_{\text{in}}, \mathcal{C}_{\text{in}}, \mathrm{rk}_{\text{in}})$, the lifting function identifies specific structural motifs or sub-patterns $\mathcal{P} \subseteq \mathcal{C}_{\text{in}}$ and maps them to cells in a target complex $\mathcal{D}_{\text{out}} = (V_{\text{out}}, \mathcal{C}_{\text{out}}, \mathrm{rk}_{\text{out}})$. A ubiquitous example in graph-based TDL is the clique lifting, where $\mathcal{L}$ maps $(k+1)$-cliques in $\mathcal{D}_{\text{in}}$ to k-cells in $\mathcal{D}_{\text{out}}$. However, this flexible formulation equally supports transformations between domains of the same dimension or mappings from higher to lower-order representations (e.g., skeletal projections).

1. Formally, denoting by $\mathcal{P}(\cdot)$ the power set, a neighborhood function $\mathcal{N} : \mathcal{C} \to \mathcal{P}(\mathcal{C})$ on a complex $\mathcal{T}$ maps each cell $\sigma \in \mathcal{C}$ to a collection of *neighbor cells* $\mathcal{N}(\sigma) \subset \mathcal{C}$.

3. Related Work

Higher-Order Representation Learning Topological Deep Learning (TDL), as surveyed in Papillon et al. (2023), has emerged as a generalization of Graph Neural Networks (GNNs), extending learning from pairwise edges to higher-order interactions. Early work focused on Simplicial Neural Networks (SNNs) (Ebli et al., 2020) and Cell Complex Neural Networks (CCNNs) (Bodnar et al., 2021), which leverage the rigid hierarchy of algebraic topology. All of these works were unified under the umbrella of Combinatorial Complexes (CCs) (Hajij et al., 2024). Modern TDL architectures are increasingly modular, defined by neighborhood operators. TopoTune (Papillon et al., 2025) introduces general topological neural networks that decompose complexes into strictly augmented Hasse graphs, letting any GNN backbone process incidence and adjacency neighborhoods as separate channels. HOPSE (Bernárdez et al., 2026) uses the same decompositions for higher-order positional encodings at linear cost. All of these works generally assume the input complex is fixed, leaving the choice of lifting as a separate, often unexamined preprocessing step.

The TDL Lifting Ecosystem A central TDL challenge is lifting point clouds and graphs into higher-order domains in a principled and meaningful way. Motivated by this, the 2024 Topological Deep Learning Challenge (Bernárdez et al., 2024) catalogued more than 30 different liftings between different pairs of domains, which were later integrated into TopoBench (Telyatnikov et al., 2025) to standardize pipelines. While the TopoBench ecosystem broadens the lifting menu, it offers no diagnostics for the structural quality of the resulting complexes. Rieck (2025) argues that such liftings may only artificially inflate model performance by increasing parameter size while relying on largely irrelevant domain structure. Only one systematic study of various liftings exists and is tied to hypergraph neural networks (Montagna et al., 2026). Our work helps address this broader gap by introducing a general diagnostic tool capable of analyzing structural properties across any higher-order domain. On the visualization side, general-purpose network exploration tools such as Gephi (Bastian et al., 2009) and Cytoscape (Shannon et al., 2003) support interactive inspection of pairwise graphs, but no existing tool exposes the cross-rank neighborhood structure of lifted higher-order domains.

Graph Metrics and Model Performance Structural descriptors are essential for understanding when and why graph learning succeeds. Reproducibility studies revealed that arbitrary data splits and hyperparameter choices mask true model capabilities (Shchur et al., 2018), and that structure-agnostic baselines often outperform complex architectures under fair protocols (Errica et al., 2020). Dwivedi et al. (2023) introduced datasets targeting structural motifs and over-smoothing, while synthetic frameworks like GraphWorld (Palowitch et al., 2022) systematically vary graph properties to study their effect on learning. On the diagnostic side, Pei et al. (2020) identified heterophily as a primary failure mode for message passing, though Platonov et al. (2023) showed that standard heterophily metrics poorly predict performance in practice. In discrete geometry, Weber et al. (2017) introduced Forman-Ricci curvature as an edge-based network characteristic; subsequent work proved that negatively curved edges cause over-squashing (Topping et al., 2022), motivating curvature-based rewiring (Fesser and Weber, 2024b) and structural encodings (Fesser and Weber, 2024a) as corrective strategies. These results demonstrate that geometric structural

metrics can provide principled, actionable signals for graph learning. TopoExplorer extends this perspective to higher-order domains by leveraging the (strictly augmented) Hasse graph representation. This allows computing and displaying classical graph metrics as well as edge-specific Forman–Ricci curvature for these higher-order domains.

Topological Metrics and Model Performance Related efforts have begun extending such diagnostics to higher-order domains. Telyatnikov et al. (2023) and Sarker et al. (2024) extended the concept of homophily to hypergraphs and simplicial complexes, explicitly tying higher-order structural properties to the empirical performance of higher-order neural networks. Such metrics remain specialized and domain-specific. By representing arbitrary higher-order domains as strictly augmented Hasse graphs, TopoExplorer instead enables practitioners to compute a broad range of graph diagnostics across the full topological hierarchy.

4. TopoExplorer: Interact and Diagnose Any Lifted Topological Dataset

We present an interactive visual and quantitative diagnostic framework built on top of the TopoBench ecosystem (Telyatnikov et al., 2025). Its central objective is to expose the structural implications of a lifting $\mathcal{L}$ and its hyperparameters before any computational resources are expended on model training. By decomposing the resulting combinatorial complex $\mathcal{C}$ into its strictly augmented Hasse graphs (Papillon et al., 2025), the interface provides a direct view of how information will flow through each topological neighborhood.

4.1. Interface Design

Stratified rank layout. To avoid TopoExplorer displaying confusingly dense cell intersections, every cell is represented as a distinct node, organized into hierarchical layers by rank (0-cells at the bottom, 1-cells in the middle, 2-cells at the top). Fig. 2 shows an example of this construction.

Neighborhood representation. *i.* Incidence neighborhoods (boundary and coboundary) are represented as directed edges *between* two layers: for example, the edges connecting the bottom layer (0-cells) to the second layer (1-cells) represent $\mathcal{I}_{0\to 1}$. If selected, TopoExplorer's incidence dashboard exclusively depicts these neighborhoods to reveal cross-rank connectivity. This shows how higher-order cells are supported by their lower-rank constituents and whether the lifting has generated orphan cells or clusters of cells that may act as sinks during message passing. *ii.* Adjacency neighborhoods (both lower and upper adjacencies) are represented as edges *within* one layer. If selected, TopoExplorer's adjacency dashboard, restricted to just these neighborhoods, reveals how the intra-rank, peer-to-peer connectivity evolves across and over ranks.

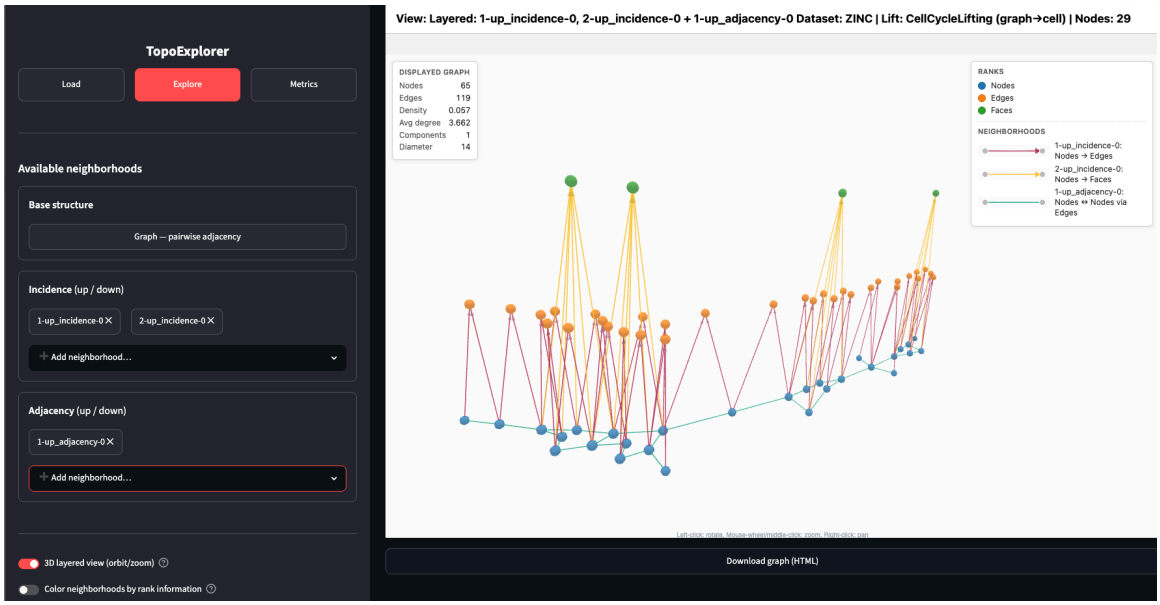

Figure 2: Neighborhood selection and representation in TopoExplorer. Example of a cellular complex from the ZINC (Irwin et al., 2012) dataset, with blue 0-cells (nodes), orange 1-cells (edges) and green 2-cells (faces) represented in terms of Hasse graphs, where the color-coded edges encode neighborhood-specific relations. For a full-page version of this figure, we refer to Appendix C.

4.2. Quantitative Signatures

Each Hasse graph $G_{\mathcal{N}}$ is paired with real-time connectivity metrics to support principled architecture design, all listed and defined in Appendix B. We provide some examples here:

- **Edge Sparsity and Density:** Near-100% sparsity in a given neighborhood graph gives practitioners a principled basis for pruning that message-passing channel in TopoTune, saving compute without sacrificing expressivity.
- **Degree Statistics:** TopoExplorer summarizes the degree distribution of each Hasse graph through its average, maximum, and minimum degree, degree assortativity, and per-node degree centralities (Table 1). Highly skewed degree statistics signal topological bottlenecks, flagging where normalization techniques or attention mechanisms may be necessary.
- **Component Analysis:** The number of connected components determines whether structural encodings will capture global context or remain localized.

We emphasize that these metrics carry no universal thresholds; they are intended to be interpreted comparatively, across candidate liftings, neighborhood choices, and hyperparameter configurations for a given dataset. TopoExplorer also offers an HTML export file recording all computed metrics, allowing users to record and compare values across configurations.

4.3. Practitioner Workflow

1. **Select a dataset.** Choose a topological domain and dataset. The app displays descriptive metadata (task, number of features, number of classes, etc.).
2. **Configure a lifting.** Select a target domain (hypergraph, simplicial, cell, combinatorial) and lifting method $\mathcal{L}$ from the TopoBench catalogue, together with its hyperparameters θ, yielding a configuration $\mathcal{L}(\theta)$.
3. **Load and lift.** Click *Load graph* to load and cache the dataset via the TopoBench API. The transform $\mathcal{L}(\theta)$ is applied to the selected sample G_i, producing a complex $\mathcal{T} = (\mathcal{V}, \mathcal{C}, \mathrm{rk})$.
4. **Select neighborhoods.** Choose one or more neighborhood types: graph adjacency, graph incidence, higher-order adjacency (through various ranks), higher-order incidence (between various ranks). Each is represented as a Hasse graph $\mathcal{G}_{\mathcal{N}}$ that can be inspected individually or combined to compare information flow across ranks.
5. **Adjust visualization settings.** For inductive datasets, navigate between data samples G_i; the lifting $\mathcal{L}(\theta)$ is reapplied automatically. Adjust display filters (minimum cell degree, maximum cells per rank) without modifying the underlying complex.
6. **Render and export.** Explore the interactive visualization, open it in a standalone browser window, or export it as a self-contained HTML file.

Beyond the hosted app. Practitioners are not restricted to the datasets and liftings served by the hosted instance. Because TopoExplorer is released open-source, the repository can be cloned and adapted to accommodate custom datasets, liftings, and metrics, and thereby embedded within an existing experimental pipeline. Extensive documentation accompanying the code details the available metrics and their interpretation, along with the recommended diagnostic workflows.

4.4. TopoExplorer in Action: Structural Diagnostics for Neighborhood Selection

A central motivation for TopoExplorer is the hypothesis that the structure induced by a neighborhood operator contains useful signals about its downstream utility. Because TopoExplorer exposes the strictly augmented Hasse graph $\mathcal{G}_{\mathcal{N}}$ associated with any neighborhood $\mathcal{N}$, practitioners can inspect these structural properties before training a model. We evaluate this hypothesis through a case study relating (strictly augmented) Hasse graph diagnostics to model performance.

Structural metrics. For a neighborhood $\mathcal{N}$, the induced Hasse graph (Eq. 1) is amenable to standard graph analysis. We consider two descriptors available directly within TopoExplorer: (i) the *spectral radius* $\lambda_{\max}(\mathcal{G}_{\mathcal{N}})$, which captures overall connectivity and expansion, and (ii) the *clustering coefficient* $C(\mathcal{G}_{\mathcal{N}})$, which measures local cohesiveness. Because incidence Hasse

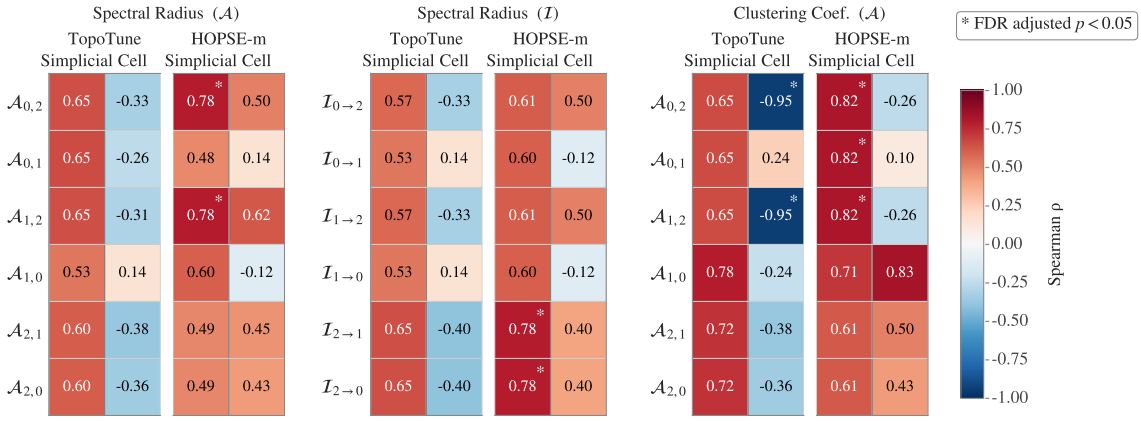

Figure 3: **Spearman ρ between Hasse graph metrics and normalized model performance.** We explore (left) spectral radius $\lambda_{\max}$ on adjacency Hasse graphs $\mathcal{G}_{\mathcal{A}_{t,s}}$, (middle) spectral radius on incidence Hasse graphs $\mathcal{G}_{\mathcal{I}_{s\to t}}$, and (right) clustering coefficient C on adjacency Hasse graphs. The simplicial and cellular domains draw from 12 and 8 datasets, respectively. A $*$ marks significant correlations.

graphs are bipartite, clustering coefficients are only defined for adjacency neighborhoods. Metrics are computed per sample and averaged at the dataset level (Figure 4).

Experimental setup. We study whether these pre-training descriptors associate with the performance of TopoTune (Papillon et al., 2025) and HOPSE (Bernárdez et al., 2026), two architectures that process adjacency ($\mathcal{A}_{t,s}$) and incidence ($\mathcal{I}_{s\to t}$) neighborhoods as separate channels. We use the best-configuration results from the large-scale experiments on 20 benchmark datasets (12 simplicial, 8 cellular) previously reported in each of these works. We compute the Spearman correlation between each dataset-level metric and performance normalized relative to the strongest graph baseline (GCN (Kipf and Welling, 2017), GIN (Xu et al., 2019), GAT (Veličković et al., 2018)). Correlations are computed separately for simplicial and cellular domains, with Benjamini-Hochberg correction ($\alpha = 0.05$).

Observations. Figure 3 reveals two key findings. First, the strongest associations are concentrated in neighborhoods involving higher-order cells, particularly 2-cell adjacencies and incidence neighborhoods. This provides correlational evidence that higher-order connectivity patterns are associated with downstream performance, though establishing that these signals reliably improve design choices requires further controlled experiments. Second, simple Hasse-graph descriptors correlate with model gains. On simplicial datasets, larger spectral radius and clustering coefficient generally associate with stronger improvements, especially for HOPSE-m. Positive trends are also observed for incidence neighborhoods. While relationships are weaker and less consistent on cellular datasets, the overall results suggest that neighborhood structure contains predictive signals about neighborhood utility before training.

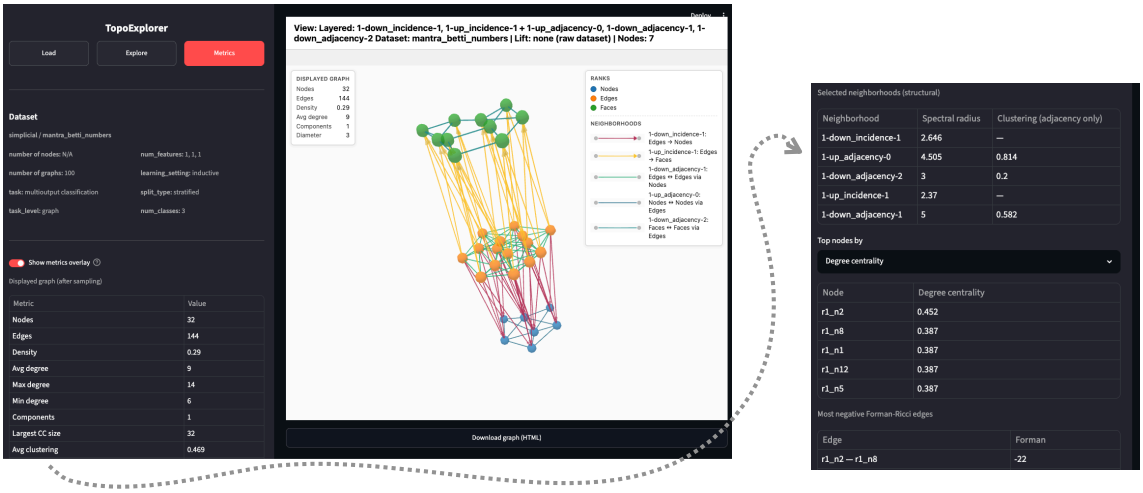

Figure 4: TopoExplorer displaying an interactive rendering of MANTRA (Ballester et al., 2025) simplicial dataset sample (see full-page version in Appendix C). For each of the 5 selected neighborhoods, the interface (see right panel screenshot) reports the spectral radius $\lambda_{\max}(\mathcal{G}_{\mathcal{N}})$ and clustering coefficient $C(\mathcal{G}_{\mathcal{N}})$ alongside other metrics (see full list in Appendix B).

Pre-training utility. These descriptors are directly available within TopoExplorer. As shown in Figure 4, selecting a neighborhood automatically renders its Hasse graph together with its structural metrics. Beyond visualization, TopoExplorer therefore provides interpretable diagnostics that can help guide neighborhood selection prior to model training.

5. Conclusion and Future Work

We present TopoExplorer, an interactive diagnostic framework that makes the structural properties of strictly augmented Hasse graphs $\mathcal{G}_{\mathcal{N}}$ visible and measurable before any model training. By exposing neighborhood-level metrics such as spectral radius and clustering coefficient directly in the application interface, TopoExplorer bridges the gap between the lifting step and the neighborhood operator selection that conditions the performance of general TDL architectures. Our case study demonstrates that these pre-training structural descriptors carry domain- and model-dependent associations with downstream performance, providing interpretable signals to inform the *lift-look-design-train* workflow we advocate. Future work includes broadening the space of structural descriptors surfaced by TopoExplorer, including geometric and feature-based metrics across all neighborhood types, and examining their predictive value for a wider range of liftings and architectures.

Acknowledgements

Mathilde Papillon acknowledges funding from Noyce Foundation and NSF Career 240158. Guillermo Bernárdez acknowledges funding from NSF 2602079 and Arlequin AI. Marco Montagna ackowledges support from Sapienza grant RG123188B3EF6A80 (CENTS). Rémi Devaux and Antoine Jardin gratefully acknowledge the NOESIS project, part of the "Pionniers de l'IA" program, for providing the resources that supported this work on TopoExplorer. Nina Miolane acknowledges funding from the NSF Career 2240158.

References

Rubén Ballester, Ernst Röell, Daniel Bin Schmid, Mathieu Alain, Sergio Escalera, Carles Casacuberta, and Bastian Rieck. MANTRA: The manifold triangulations assemblage. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=X6y5CC44HM.

Mathieu Bastian, Sebastien Heymann, and Mathieu Jacomy. Gephi: an open source software for exploring and manipulating networks. In *Proceedings of the international AAAI conference on web and social media*, volume 3, pages 361–362, 2009.

Guillermo Bernárdez, Lev Telyatnikov, Marco Montagna, Federica Baccini, Mathilde Papillon, Miquel Ferriol-Galmés, Mustafa Hajij, Theodore Papamarkou, Maria Sofia Bucarelli, Olga Zaghen, et al. Icml topological deep learning challenge 2024: Beyond the graph domain. *Proceedings of Machine Learning Research*, 251:420–428, 2024.

Guillermo Bernárdez, Marco Montagna, Louis Van Langendonck, Martin Carrasco, Amirreza Akbari, Louisa Cornelis, Mathilde Papillon, Pere Barlet-Ros, Nina Miolane, and Lev Telyatnikov. Hopse: Scalable higher-order positional and structural encoder for combinatorial representations, 2026. URL https://arxiv.org/abs/2505.15405.

Cristian Bodnar. *Topological deep learning: graphs, complexes, sheaves*. PhD thesis, University of Cambridge, 2023.

Cristian Bodnar, Fabrizio Frasca, Nina Otter, Yuguang Wang, Pietro Lio, Guido F Montufar, and Michael Bronstein. Weisfeiler and Lehman Go Cellular: CW Networks. *Advances in Neural Information Processing Systems*, 34:2625–2640, 2021.

Vijay Prakash Dwivedi, Chaitanya K Joshi, Anh Tuan Luu, Thomas Laurent, Yoshua Bengio, and Xavier Bresson. Benchmarking graph neural networks. *Journal of Machine Learning Research*, 24(43):1–48, 2023.

S. Ebli, M. Defferrard, and G. Spreemann. Simplicial neural networks. In *Advances in Neural Information Processing Systems Workshop on Topological Data Analysis and Beyond*, 2020.

Federico Errica, Marco Podda, Davide Bacciu, and Alessio Micheli. A fair comparison of graph neural networks for graph classification. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=HygDF6NFPB.

Lukas Fesser and Melanie Weber. Effective structural encodings via local curvature profiles. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun, editors, *International Conference on Learning Representations*, volume 2024, pages 22571–22589, 2024a. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/6188c02ccc16a7587716de2efd754033-Paper-Conference.pdf.

Lukas Fesser and Melanie Weber. Mitigating over-smoothing and over-squashing using augmentations of forman-ricci curvature. In *Learning on Graphs Conference*, pages 19–1. PMLR, 2024b.

Mustafa Hajij, Ghada Zamzmi, Theodore Papamarkou, Nina Miolane, Aldo Guzmán-Sáenz, Karthikeyan Natesan Ramamurthy, Tolga Birdal, Tamal K. Dey, Soham Mukherjee, Shreyas N. Samaga, Neal Livesay, Robin Walters, Paul Rosen, and Michael T. Schaub. Topological deep learning: Going beyond graph data, 2023.

Mustafa Hajij, Theodore Papamarkou, Ghada Zamzmi, Karthikeyan Natesan Ramamurthy, Tolga Birdal, and Michael T. Schaub. *Topological Deep Learning: Going Beyond Graph Data*. Online, 2024. URL http://tdlbook.org. Published online on August 6, 2024.

John J Irwin, Teague Sterling, Michael M Mysinger, Erin S Bolstad, and Ryan G Coleman. ZINC: a free tool to discover chemistry for biology. *Journal of Chemical Information and Modeling*, 52(7):1757–1768, 2012.

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

Marco Montagna, Simone Scardapane, and Lev Telyatnikov. Lift me up: The impact of liftings on hypergraph neural networks. In *ICLR 2026 GRAM Workshop*, 2026.

John Palowitch, Anton Tsitsulin, Brandon Mayer, and Bryan Perozzi. Graphworld: Fake graphs bring real insights for gnns. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, pages 3691–3701, 2022.

Mathilde Papillon, Sophia Sanborn, Mustafa Hajij, and Nina Miolane. Architectures of topological deep learning: A survey of message-passing topological neural networks. *arXiv preprint arXiv:2304.10031*, 2023.

Mathilde Papillon, Guillermo Bernardez, Claudio Battiloro, and Nina Miolane. Topotune: A framework for generalized combinatorial complex neural networks. In *International Conference on Machine Learning*, pages 47924–47952. PMLR, 2025.

Hongbin Pei, Bingzhe Wei, Kevin Chen-Chuan Chang, Yu Lei, and Bo Yang. Geom-gcn: Geometric graph convolutional networks. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=S1e2agrFvS.

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 *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=tJbbQfw-5wv.

Bastian Rieck. Have graph – will lift? the case for higher-order benchmarks. *Proceedings of Workshop on Geometry, Topology, and Machine Learning (GTML)*, 2025.

Arnab Sarker, Natalie Northrup, and Ali Jadbabaie. Higher-order homophily on simplicial complexes. *Proceedings of the National Academy of Sciences of the United States of America*, 121(12), Mar 2024. doi: https://doi.org/10.1073/pnas.2315931121.

Paul Shannon, Andrew Markiel, Owen Ozier, Nitin S Baliga, Jonathan T Wang, Daniel Ramage, Nada Amin, Benno Schwikowski, and Trey Ideker. Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome research*, 13(11):2498–2504, 2003.

Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. Pitfalls of graph neural network evaluation. *Relational Representation Learning Workshop, NeurIPS 2018*, 2018.

Lev Telyatnikov, Maria Sofia Bucarelli, Guillermo Bernardez, Olga Zaghen, Simone Scardapane, and Pietro Lio. Hypergraph neural networks through the lens of message passing: a common perspective to homophily and architecture design. *arXiv preprint arXiv:2310.07684*, 2023.

Lev Telyatnikov, Guillermo Bernardez, Marco Montagna, Mustafa Hajij, Martin Carrasco, Pavlo Vasylenko, Mathilde Papillon, Ghada Zamzmi, Michael T Schaub, Jonas Verhellen, Pavel Snopov, Bertran Miquel-Oliver, Manel Gil-Sorribes, Alexis Molina, VICTOR GUALLAR, Theodore Long, Julian Suk, Patryk Rygiel, Alexander V Nikitin, Giordan Escalona, Michael Banf, Dominik Filipiak, Liliya Imasheva, Max Schattauer, Alvaro L. Martinez, Halley Fritze, Marissa Masden, Valentina Sánchez, Manuel Lecha, Andrea Cavallo, Claudio Battiloro, Matthew Piekenbrock, Mauricio Tec, George Dasoulas, Nina Miolane, Simone Scardapane, and Theodore Papamarkou. Topobench: A framework for benchmarking topological deep learning. *Journal of Data-centric Machine Learning Research*, 2025. URL https://openreview.net/forum?id=07sTzyEVtY.

Jake Topping, Francesco Di Giovanni, Benjamin Paul Chamberlain, Xiaowen Dong, and Michael M. Bronstein. Understanding over-squashing and bottlenecks on graphs via curvature. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=7UmjRGzp-A.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=rJXMpikCZ.

Duncan J Watts and Steven H Strogatz. Collective dynamics of 'small-world'networks. *nature*, 393(6684):440–442, 1998.

Melanie Weber, Emil Saucan, and Jürgen Jost. Characterizing complex networks with forman-ricci curvature and associated geometric flows. *Journal of Complex Networks*, 5(4): 527–550, 2017.

Melanie Weber, Jürgen Jost, and Emil Saucan. Detecting the coarse geometry of networks. In *NeurIPS Workshop on Relational Representation Learning*, 2018.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=ryGs6iA5Km.

Appendix A. Definition of a Neighborhood

Here we provide the complete mathematical definition for neighborhoods of a combinatorial complex $\mathcal{T}$. The incidence neighborhood of a t-cell $\sigma \in \mathcal{T}$ with respect to a rank s is defined as

$$\mathcal{I}_{s\to t}(\sigma) = \begin{cases} \{\tau \in \mathcal{C} \,|\, \mathrm{rk}(\tau) = s, \sigma \subset \tau\} & \text{if } t < s; \\ \{\tau \in \mathcal{C} \,|\, \mathrm{rk}(\tau) = s, \tau \subset \sigma\} & \text{if } t > s. \end{cases} \tag{2}$$

A s-cell $\tau \in \mathcal{T}$ is said to be co-incident of σ if it contains σ and $s > t$; analogously, a s-cell $\tau \in \mathcal{T}$ is incident of σ if it is contained by σ and $s < t$. Incidences in turn induce adjacency neighborhoods of cells of the same rank as follows:

$$\mathcal{A}_{s,t}(\sigma) = \{\tau \in \mathcal{C} \,|\, \mathrm{rk}(\tau) = t, \mathcal{I}_{s\to t}(\sigma) \cap \mathcal{I}_{s\to t}(\tau) \neq \emptyset\} \tag{3}$$

In this case, two t-cells σ and τ are said to be adjacent if both are contained in a s-cell $\delta \in \mathcal{T}$.

Appendix B. TopoExplorer metrics

TopoExplorer reports structural quantities on the graph loaded and configured in its visualization panel. Unless noted otherwise, complex- and node-level statistics are computed on the *displayed* graph, i.e. the subgraph rendered after dataset loading, optional lifting, and user-controlled sampling. They are evaluated on an undirected view so that the same definitions apply to plain adjacency, incidence/bipartite, and layered neighborhood displays. Neighborhood-level statistics, by contrast, are computed on the full neighborhood matrix associated with each selected view (e.g., rank-k adjacency or cross-rank incidence), before sampling is applied. Table 1 summarizes the metrics exposed in the *Metrics* tab; entries marked as advanced are optional and may be approximated on large graphs (e.g., betweenness via k-node sampling above $\sim$800 nodes). Edge-local quantities such as Forman-Ricci curvature and edge betweenness appear in edge tooltips and in summary tables for extreme edges.

Appendix C. Full-Page Screenshots of TopoExplorer

For legibility, we reproduce the two interface screenshots from the main text at full-page scale.

Table 1: Metrics provided by TopoExplorer's *Metrics* tab. Complex- and node-level quantities are computed on the *displayed* graph (after sampling and lifting), treated as undirected. Neighborhood-level quantities are computed on the full neighborhood matrix for each selected view, before sampling.

Metric	**Definition**
Complex-level metrics (displayed graph)	
Nodes	Number of vertices in the displayed graph.
Edges	Number of edges in the displayed graph.
Density	Edge density $m/\binom{n}{2}$ for $n > 1$; 0 if $n \leq 1$.
Avg degree	Mean vertex degree over all nodes.
Max degree	Maximum vertex degree.
Min degree	Minimum vertex degree.
Components	Number of connected components (undirected view).
Largest CC size	Number of nodes in the largest connected component.
Avg clustering	Global average local clustering coefficient (Watts and Strogatz, 1998).
Transitivity	Global clustering coefficient (fraction of closed triples among connected triples).
Diameter	Maximum eccentricity in the largest connected component; omitted if the graph exceeds size limits.
Degree assortativity	Pearson correlation of degrees at edge endpoints; available when advanced metrics are enabled.
Neighborhood-level metrics (selected neighborhoods)	
Spectral radius	For each selected neighborhood matrix: largest $\lvert\lambda\rvert$ if square (symmetric adjacency), otherwise largest singular value. Computed on the full matrix, not the sampled display.
Clustering (adjacency only)	Average local clustering coefficient of the undirected graph induced by a square adjacency neighborhood.
Node-level metrics (displayed graph)	
Degree	Number of neighbors of the node (undirected view).
Degree centrality	Normalized degree $k/(n-1)$.
Clustering	Local clustering coefficient of the node.
Betweenness	Betweenness centrality; for graphs with >800 nodes, approximated by k-node sampling (advanced).
Closeness	Closeness centrality based on shortest-path distances (advanced).
Eccentricity	Maximum shortest-path distance from the node to any other node in its connected component (advanced).
Eigenvector centrality	Centrality proportional to the sum of neighbors' centralities (advanced).
PageRank	Stationary score under a random-walk model with damping (advanced).
Forman-Ricci	Per-edge curvature on the displayed graph: $4 - \deg(u) - \deg(v)$ under unit edge weights (Weber et al., 2018).
Edge betweenness	Betweenness centrality of an edge (advanced).

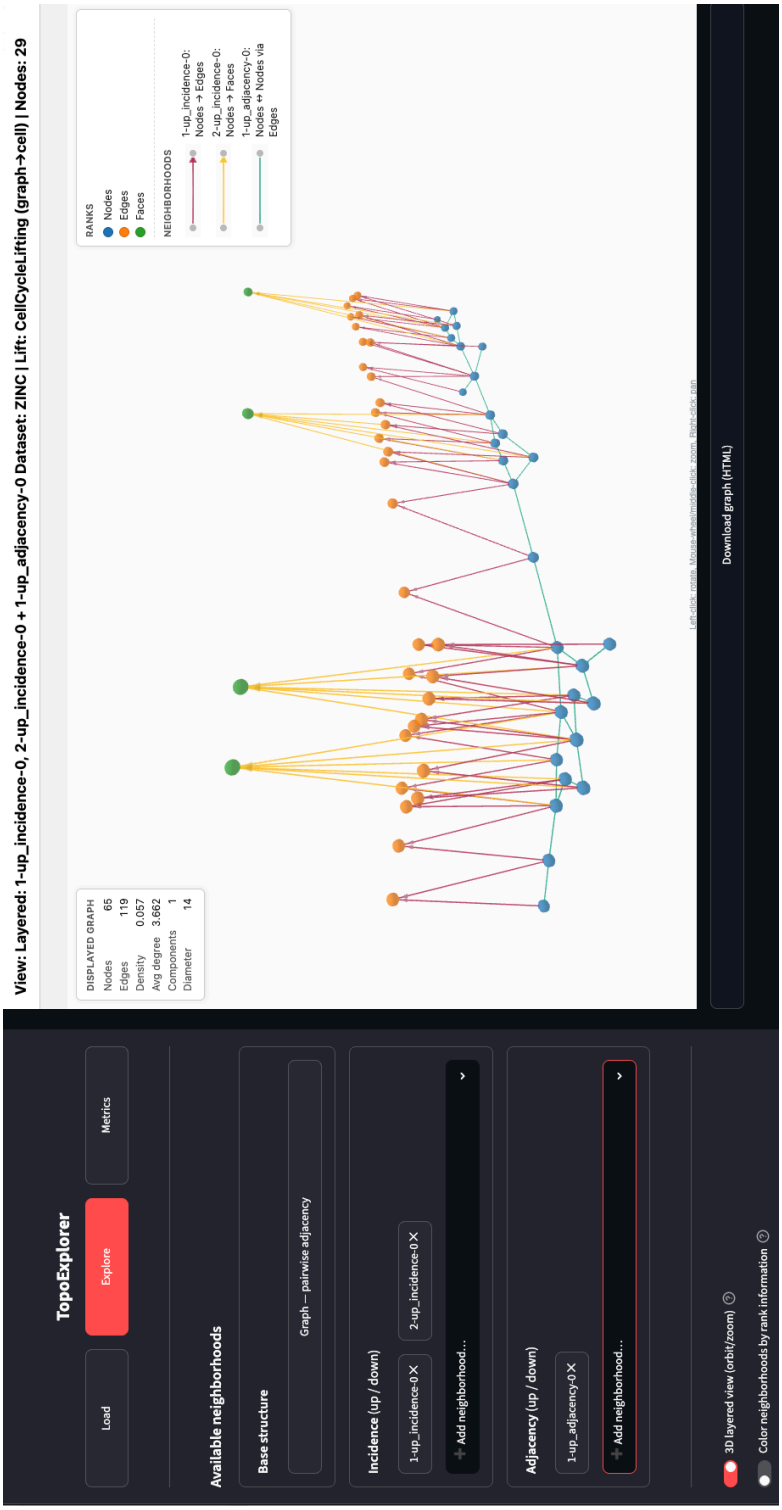

Figure 5: Full-page version of Figure 2: neighborhood selection and representation in TopoExplorer on the ZINC (Irwin et al., 2012) dataset

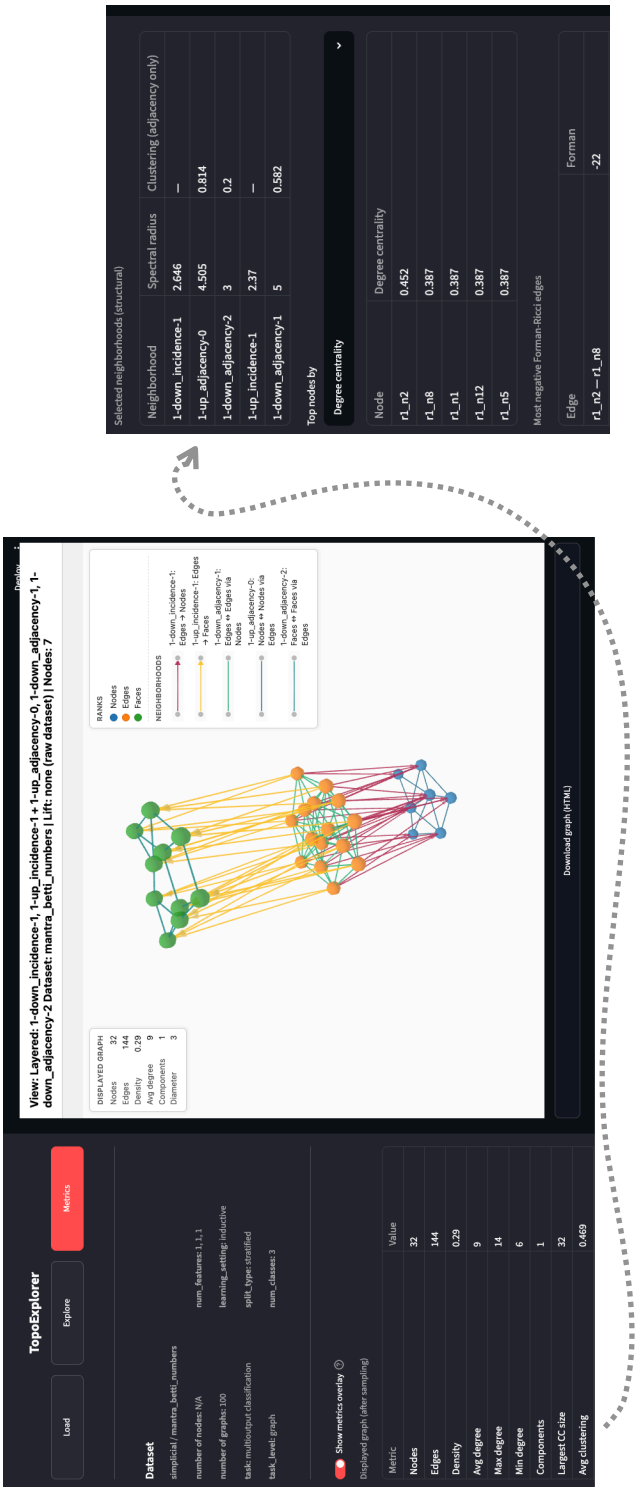

Figure 6: Full-page version of Figure 4: TopoExplorer on a MANTRA (Ballester et al., 2025) simplicial dataset sample, with the metrics panel enlarged.