

# 1 Introduction

...

## 2 Preliminary

...

### 2.1 Topological Data Analysis (TDA)

In recent years, TDA has found its applications in various areas such as machine learning, artificial intelligence, data science, neuroscience, and so on (Giunti et al., 2022). Especially in the area of graph representation learning, TDA has shown the power of enhancing popular GNNs on different tasks by augmenting potentially useful topological features represented by TDA methods (Hofer et al., 2017; Dehmamy et al., 2019; Carrière et al., 2020; Horn et al., 2022). One successful tool is *persistent homology*. In this work, we focus on graphs. The persistent homology helps a model understand how topological features (like connected components and cycles) appear and persist during a generating process of a graph. One can think of it like watching a video of a graph being constructed edge by edge according to some learnable ordering. Persistent homology tracks which structural patterns emerge and how long they last during this process. Various models have been proposed to apply TDA to graph representation learning (Carrière et al., 2020; Horn et al., 2022; Hofer et al., 2020; Xin et al., 2023; Yan et al., 2022; Zhao et al., 2020; Carrière & Blumberg, 2020; Zhang et al., 2022). We give a brief introduction to the basic concepts of topological data analysis (TDA) and persistent homology. For a more detailed introduction, we refer readers to (Edelsbrunner & Harer, 2010; Dey & Wang, 2022).

**Graph Filtration:** For an edge-weighted graph  $G = (V, E, f : E \rightarrow \mathbb{R})$ , we can create a sequence of nested subgraphs called a graph filtration. The weight function  $f$  can be thought of as assigning "importance scores" to edges. For example, a graph filtration  $\mathcal{F}(G) := \{G_{\leq t} \mid t \in 1 - f(E)\}$  with  $G_{\leq t}$  being the subgraph of  $G$  on edges with  $1 - f(e) \leq t$  shows us how the graph grows as we include edges in order of decreasing importance scores. By tradition, we set  $G_{-\infty} = \emptyset$  and  $G_{\infty} = G$  to be the first and last elements in the graph filtration.

**Persistent Homology:** On such a graph filtration, one can track all the connected components and cycles appearing and disappearing (merged with others) during the process. If we encode the lifecycle (birth, death) of each component or cycle as an interval on the real line, it turns out that there is a unique way to represent such information as a multi-sets of intervals which is equivalent to a well-studied algebraic structure in topological data analysis, known as persistent homology. Through the theory of TDA, this representation can be proved to be a topological invariant of the graph filtration. We refer readers to appendix B and (Edelsbrunner & Harer, 2010; Zomorodian & Carlsson, 2004; Ghrist, 2008) for further details about persistent homology. Here for conciseness, one can treat persistent homology as a differentiable permutation invariant representation which encodes lifecycles of topological features (connected components and cycles) as a multiset of intervals. We denote such a representation as  $\mathcal{T} \circ \mathcal{F}(G)$ . See Figure 1 as an example of a graph filtration (cubes on top line) and its persistent homology (the multiset of intervals shown as horizontal lines in the bottom). We will visit this example later after we introduce our model.

**Comparing Topological Features:** One can compare two edge-weighted graphs by computing distances between the persistent homologies of their corresponding graph filtrations. A commonly used one is the bottleneck distance  $d_B$  (Definition B.1 in Appendix B). Recall that persistent homology is represented as a multiset of intervals, which is equivalent to a multiset of pair of points (endpoints of intervals) in  $\mathbb{R}^2$ . The bottleneck distance can be essentially understood as a variant of Wasserstein distance between these two multisets in  $(\mathbb{R}^2, \|\cdot\|_{\infty})$  with  $\infty$ -norm metric. Our topological discrepancy will be essentially constructed based on this bottleneck distance.

## 3 Method of TOPING

...

We denote  $\mathcal{F}(G_{\leq t})$  to be the filtration of the subgraphs on edges with filtration values greater than  $1 - t$ . Symmetrically, let  $\mathcal{F}(G_{\geq t})$  to be the filtration of the subgraphs on edges with filtration values smaller than or equal to  $1 - t$ . We consider the following property:

**persistent homology gap:** For some threshold value  $t \in [0, 1]$ , there is a significant difference between the two filtrations  $\mathcal{F}(G_{\leq t})$  and  $\mathcal{F}(G_{\geq t})$  with respect to some topological invariant  $\mathcal{T}$ . Here we use persistent homology as our

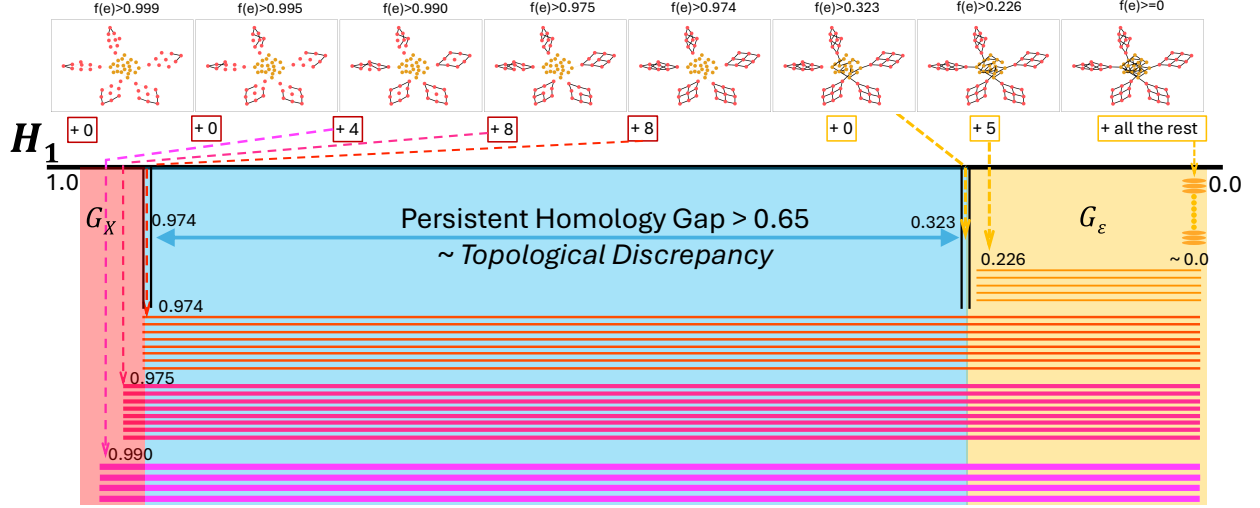


Figure 1: The top row is a learned graph filtration on an example graph through our method. Red and yellow points correspond to ground truth rationale subgraph  $G_X^*$  and noisy subgraph  $G_\epsilon^*$  respectively. Each snapshot is a subgraph of  $G$  on edges with filtration values greater than a decreasing threshold value indicated on top of each snapshot. Below each subgraph, the number in small box is the size of cycle basis for the current subgraph, which equals to the dimension of the cycle space  $H_1$ , also known as the 1st betti number. The bottom part shows the 1st persistent barcode. Each horizontal bar corresponds to a topological feature (basic cycle).

topological invariant  $\mathcal{T} = H_p(\cdot)$  for  $p = 0, 1$ , and bottleneck distance (Definition B.1 in Appendix B) to measure the topological difference. That is  $d_B(\mathcal{T} \circ \mathcal{F}(G_{\leq t}), \mathcal{T} \circ \mathcal{F}(G_{\geq t}))$ .

**Remark 3.1.** The underlying idea of this property is that, if we track the generating process of the rationale subgraph  $G_X$  and the noise subgraph  $G_\epsilon$ , we hope to see in general two very different evolutionary paths on the topological structures during the process. The methods based on our persistent rationale filtration framework should be able to capture such difference.

For briefness of notations, we denote  $\mathcal{T}_X = \mathcal{T} \circ \mathcal{F}(G_{\leq t})$  and  $\mathcal{T}_\epsilon = \mathcal{T} \circ \mathcal{F}(G_{\geq t})$  for the persistent homologies of a given filtration and a prefixed  $t \in [0, 1]$ . By default,  $t = 0.5$ . We also denote the induced probability distributions of persistent homologies as  $\mathbb{P}(\mathcal{T}_X)$  and  $\mathbb{P}(\mathcal{T}_\epsilon)$  respectively.

**Definition 3.2** (Topological Discrepancy). The topological discrepancy  $d_{topo}$  between  $\mathbb{P}(\mathcal{T}_X)$  and  $\mathbb{P}(\mathcal{T}_\epsilon)$  is defined as

$$d_{topo}(\mathbb{P}(\mathcal{T}_X), \mathbb{P}(\mathcal{T}_\epsilon)) \triangleq \inf_{\pi \in \Pi(\mathbb{P}(\mathcal{T}_X), \mathbb{P}(\mathcal{T}_\epsilon))} \mathbb{E}_{(P, Q) \sim \pi} [d_B(P, Q)] \quad (1)$$

where  $\Pi(\mathbb{P}(\mathcal{T}_X), \mathbb{P}(\mathcal{T}_\epsilon))$  is the set of all couplings between  $\mathbb{P}(\mathcal{T}_X)$  and  $\mathbb{P}(\mathcal{T}_\epsilon)$ , and  $d_B(P, Q)$  is the bottleneck distance between the persistent homologies  $P$  and  $Q$ .

Essentially,  $d_{topo}$  is the 1-Wasserstein distance between the distributions of persistent homologies  $\mathcal{T} \circ \mathcal{F}(G_{\leq t})$  and  $\mathcal{T} \circ \mathcal{F}(G_{\geq t})$  under the metric  $d_B$ . Now we are ready to design a high-level model together with a loss function approximating an  $f_\phi^*$  based on our topological discrepancy property.

$$\mathcal{L}(\phi) = \mathbb{E}_G [\mathcal{L}_{ce}(\hat{y}_G, y_G)] - \alpha \mathcal{L}_{topo}(\mathbb{P}(\mathcal{T}_X), \mathbb{P}(\mathcal{T}_\epsilon)) \quad (2)$$

The topological constraint term  $\mathcal{L}_{topo}$  is realized by the topological discrepancy  $d_{topo}$ . The prediction loss term  $\mathcal{L}_{ce}$  is the standard cross-entropy loss between the predicted label  $\hat{y}_G$  and the ground truth label  $y_G$ . The predicted label  $\hat{y}_G = h_\phi \sigma f_\phi(G)$  is obtained by applying prediction network  $h_\phi$  on the subgraph  $G_X$  extracted through  $\sigma$  from the filtration  $f_\phi(G)$ .  $h_\phi$  and  $f_\phi$  share the same backbone  $GNN_\phi$  model, which outputs a permutation equivalent representation (node or edge representation).  $f_\phi = MLP_f \circ GNN_\phi$  applies a simple multi-layer perceptron (MLP) model to get a 1-dimensional edge representation as the filtration function.  $h_\phi = MLP_h \circ Pool \circ GNN_\phi$  first applies a pooling layer  $Pool$  (e.g. standard sum-pooling layer) to transform the permutation equivalent representation  $GNN_\phi$  to a permutation invariant graph representation. Afterwards, such graph representation  $Pool \circ GNN_\phi$ , together with the persistent homologies  $\mathcal{T}_X$  and  $\mathcal{T}_\epsilon$  through concatenation, are fed to a final classifier  $MLP_h$  to predict  $\hat{y}_G$ . We omit other details of learnable parameters in MLP models for simplicity. See Figure 2 as a high-level illustration of the architecture of our model.

...

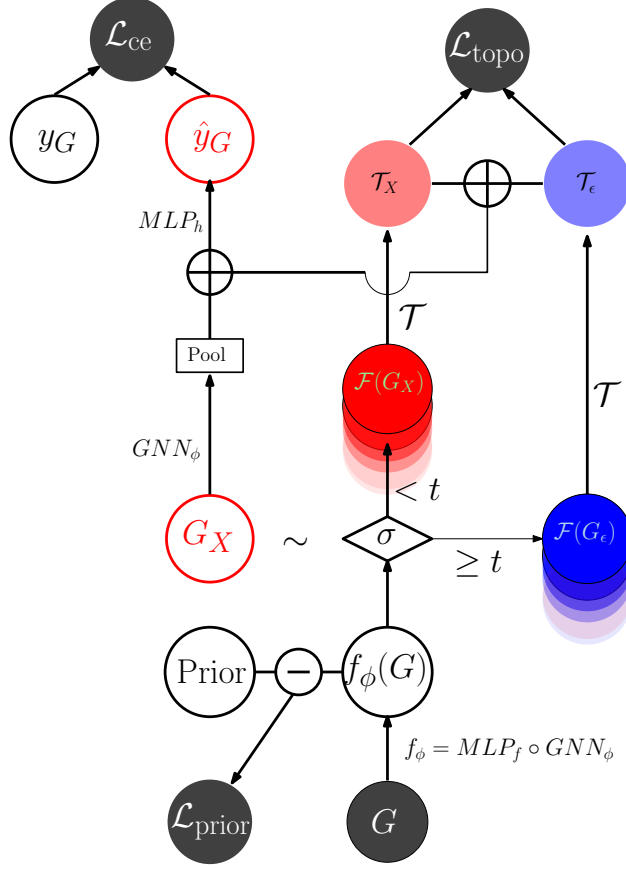


Figure 2: TOPING’s high-level model architecture. Starts from a backbone  $GNN_\phi$  followed by a simple  $MLP_f$  to obtain a scalar edge embedding  $f_\phi(G)$  as the filtration function. After pass an extractor  $\sigma$ , two graph filtrations  $\mathcal{F}(G_X)$  and  $\mathcal{F}(G_\epsilon)$  are induced for computing topological features  $\mathcal{T}_X$  and  $\mathcal{T}_\epsilon$ , and their topological loss  $\mathcal{L}_{\text{topo}} = d_{\text{topo}}$ . On the predictor part, the same backbone  $GNN_\phi$  followed by a standard pooling layer  $Pool$  maps a extracted  $G_X$  to a graph representation. Together with the topological features  $\mathcal{T}_X$  and  $\mathcal{T}_\epsilon$  by concatenation, the graph representation is fed to a final classifier  $MLP_h$  for the label prediction  $\hat{y}_G$ , compared with the ground truth label  $y_G$  under standard cross-entropy loss  $\mathcal{L}_{\text{ce}}$ . A prior regularization loss  $\mathcal{L}_{\text{prior}}$  is used to regularize the filtration function  $f_\phi$  to be close to a mixture of Gaussian prior distribution.

## 4 Experiments

**What could be learned by  $d_{\text{topo}}$ :** To get some intuition on what could be learned by our topological discrepancy, we illustrate, in Figure 1, a learned filtration on an example graph, and its 1st persistent homology (barcode). First check the top part of our learned graph filtration. Note that here we did not do any normalization on the filtration values. Observe that all edges in  $G_X$  have weights  $\geq 0.974$  and all edges in  $G_\epsilon$  have weights  $\leq 0.323$ , which means the rationale filtration we learned is quite consistent with the ground truth rationale. Taking a closer look at the filtration, one can see that the generating process of the rationale filtration is well-ordered and fast. A clear pattern of the generating process can be observed. However, the noisy graph is generated in a more chaotic way. Only five cycles are generated until the threshold value reaches 0.226. Most of the cycles are generated at the end close to 0. Now check the bottom part, the persistent barcode. Each horizontal bar corresponds to a topological feature, a basic cycle. The left end of each bar indicates the first time it appears in the filtration. The most important information one can get from the barcode is that, within the interval  $[0.974, 0.226]$ , the barcode does not change at all. That means the persistent topological structure of the graph is stable within this region. The length of the interval  $(0.974 - 0.226)$  is related to what we called *persistent homology gap*, which is a measure of the difference of topological structures between the  $G_X$  and  $G_\epsilon$ . Note that this gap is very close to the gap between minimal edge weight in  $G_X$  and maximal edge weight in  $G_\epsilon$ . In fact, they will be exactly the same if we also consider the 0-th persistent homology. Such a gap will be approximated by what

we proposed *topological discrepancy*, and our final target function is designed to maximize persistent homology gaps statistically over all data.

...

## **5 Conclusion and Future Work**

...

## References

- M.F. Atiyah and I.G. MacDonald. *Introduction To Commutative Algebra*. Addison-Wesley series in mathematics. Avalon Publishing, 1994. ISBN 9780813345444. URL <https://books.google.com/books?id=H0ASFid4x18C>.
- Mathieu Carrière and Andrew Blumberg. Multiparameter persistence image for topological machine learning. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 22432–22444. Curran Associates, Inc., 2020. URL <https://proceedings.neurips.cc/paper/2020/file/fdfff71fcab656abfbefaaabecab1a7f6d-Paper.pdf>.
- Mathieu Carrière, Frédéric Chazal, Yuichi Ike, Theo Lacombe, Martin Royer, and Yuhei Umeda. Perslay: A neural network layer for persistence diagrams and new graph topological signatures. In Silvia Chiappa and Roberto Calandra (eds.), *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pp. 2786–2796. PMLR, 26–28 Aug 2020. URL <https://proceedings.mlr.press/v108/carriere20a.html>.
- Frédéric Chazal, Vin de Silva, and Steve Oudot. Persistence stability for geometric complexes. *Geometriae Dedicata*, 173(1):193–214, Dec 2014. ISSN 1572-9168. doi: 10.1007/s10711-013-9937-z. URL <https://doi.org/10.1007/s10711-013-9937-z>.
- Nima Dehmamy, Albert-László Barabási, and Rose Yu. *Understanding the Representation Power of Graph Neural Networks in Learning Graph Topology*. Curran Associates Inc., Red Hook, NY, USA, 2019. URL <https://proceedings.neurips.cc/paper/2019/file/73bf6c41e241e28b89d0fb9e0c82f9ce-Paper.pdf>.
- Tamal K. Dey and Cheng Xin. Computing bottleneck distance for 2-D interval decomposable modules. In Bettina Speckmann and Csaba D. Tóth (eds.), *34th International Symposium on Computational Geometry (SoCG 2018)*, volume 99, pp. 32:1–32:15, Dagstuhl, Germany, 2018. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi: 10.4230/LIPIcs.SocG.2018.32. URL <https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.SocG.2018.32>.
- Tamal Krishna Dey and Yusu Wang. *Computational Topology for Data Analysis*. Cambridge University Press, 2022.
- Herbert Edelsbrunner and John Harer. *Computational Topology - an Introduction*. American Mathematical Society, 2010. ISBN 978-0-8218-4925-5.
- Robert Ghrist. Barcodes: The persistent topology of data. *Bulletin of the American Mathematical Society*, 45:61–75, 2008. doi: 10.1090/S0273-0979-07-01191-3. URL <http://www.ams.org/bull/2008-45-01/S0273-0979-07-01191-3/>.
- Barbara Giunti, Jānis Lazovskis, and Bastian Rieck. DONUT: Database of original & non-theoretical uses of topology, 2022. URL <https://donut.topology.rocks>.
- Allen Hatcher. *Algebraic topology*. Cambridge University Press, Cambridge, 2002. ISBN 0-521-79160-X; 0-521-79540-0.
- Christoph D. Hofer, Roland Kwitt, Marc Niethammer, and Andreas Uhl. Deep learning with topological signatures. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 1634–1644, 2017. URL [https://proceedings.neurips.cc/paper\\_files/paper/2017/file/883e881bb4d22a7add958f2d6b052c9f-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2017/file/883e881bb4d22a7add958f2d6b052c9f-Paper.pdf).
- Christoph D. Hofer, Roland Kwitt, and Marc Niethammer. Learning representations of persistence barcodes. *Journal of Machine Learning Research*, 20(126):1–45, 2019. URL <http://jmlr.org/papers/v20/18-358.html>.
- Christoph D. Hofer, Florian Graf, Bastian Rieck, Marc Niethammer, and Roland Kwitt. Graph filtration learning. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 4314–4323. PMLR, 2020. URL <http://proceedings.mlr.press/v119/hofer20b.html>.
- Max Horn, Edward De Brouwer, Michael Moor, Yves Moreau, Bastian Rieck, and Karsten M. Borgwardt. Topological graph neural networks. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL <https://openreview.net/forum?id=oxxUMeFwEHd>.
- J. D. Horton. A polynomial-time algorithm to find the shortest cycle basis of a graph. *SIAM Journal on Computing*, 16(2):358–366, 1987. doi: 10.1137/0216026. URL <https://doi.org/10.1137/0216026>.
- Dieter Jungnickel. *Graphs, Networks and Algorithms*. Springer Publishing Company, Incorporated, 3rd edition, 2007. ISBN 3540727795.
- Telikepalli Kavitha, Christian Liebchen, Kurt Mehlhorn, Dimitrios Michail, Romeo Rizzi, Torsten Ueckerdt, and Katharina A. Zweig. Cycle bases in graphs characterization, algorithms, complexity, and applications. *Computer*

- Science Review*, 3(4):199–243, 2009. ISSN 1574-0137. doi: <https://doi.org/10.1016/j.cosrev.2009.08.001>. URL <https://www.sciencedirect.com/science/article/pii/S1574013709000483>.
- Soham Mukherjee, Shreyas N. Samaga, Cheng Xin, Steve Oudot, and Tamal K. Dey. D-GRIL: end-to-end topological learning with 2-parameter persistence, 2024. URL <https://arxiv.org/abs/2406.07100>.
- Cheng Xin, Soham Mukherjee, Shreyas N. Samaga, and Tamal K. Dey. GRIL: A 2-parameter persistence based vectorization for machine learning. In *Proceedings of 2nd Annual Workshop on Topology, Algebra, and Geometry in Machine Learning (TAG-ML)*, volume 221 of *Proceedings of Machine Learning Research*, pp. 313–333. PMLR, 7 2023. URL <https://proceedings.mlr.press/v221/xin23a.html>.
- Zuoyu Yan, Tengfei Ma, Liangcai Gao, Zhi Tang, Yusu Wang, and Chao Chen. Neural approximation of graph topological features. In *Proceedings of the 36th International Conference on Neural Information Processing Systems, NIPS '22*, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.
- Simon Zhang, Soham Mukherjee, and Tamal K. Dey. GEFL: Extended filtration learning for graph classification. In Bastian Rieck and Razvan Pascanu (eds.), *Proceedings of the First Learning on Graphs Conference*, volume 198 of *Proceedings of Machine Learning Research*, pp. 16:1–16:26. PMLR, 09–12 Dec 2022. URL <https://proceedings.mlr.press/v198/zhang22b.html>.
- Qi Zhao, Ze Ye, Chao Chen, and Yusu Wang. Persistence enhanced graph neural network. In Silvia Chiappa and Roberto Calandra (eds.), *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pp. 2896–2906. PMLR, 2020.
- Afra Zomorodian and Gunnar Carlsson. Computing persistent homology. In *Proceedings of the twentieth annual symposium on Computational geometry*, pp. 347–356, 2004. URL <https://link.springer.com/content/pdf/10.1007/s00454-004-1146-y.pdf>.

## A List of Notations

...

## B More details of TDA

**Homology Vector Spaces:** The general definition of homology groups and homology vector spaces under fields can be checked in (Hatcher, 2002). Here we only focus on the special case of 0-th and 1-st homology vector spaces (homology in brief) over field  $\mathbb{F}_2 = \{0, 1\}$  on graphs. For a given graph  $G$ , the 0-th homology  $H_0(G)$  is defined as the vector space with basis to be all connected components of  $G$ . The 1-st homology  $H_1(G)$  is the vector space on the set of all cycles in  $G$ , with addition operation defined as the symmetric difference of cycles. The zero vector in  $H_1(G)$  is the empty cycle. It can be shown that  $H_1(G)$  is a vector space. This vector space is also called cycle space and its basis is known as cycle basis and well studied in graph theory (Horton, 1987; Kavitha et al., 2009; Jungnickel, 2007). For example, for a graph  $G = (V, E = \emptyset)$ , the 0-th persistent homology  $H_0(G) = \{0, 1\}^{|V|}$  is a  $|V|$ -dimensional vector space with basis being the set of all points.  $H_1(G) = 0$  is trivial.

**Persistent Homology:** Persistent Homology of graphs is a kind of "generalization" of homology on a weighted graph  $G = (V, E, f : E \rightarrow \mathbb{R})$ . Starting from a graph filtration  $\mathcal{F}(G)$  given by the reversed order of edge weights

$$\mathcal{F}(G) : \emptyset \subseteq \dots \rightarrow G_{\leq t_1} \subseteq G_{\leq t_2} \subseteq G_{\leq t_3} \subseteq \dots \subseteq H_p(G), \quad (3)$$

if we apply the  $p$ -homology functor to the graph filtration, each subgraph  $G_{\leq t}$  is mapped to a homology vector space  $H_p(G_{\leq t})$ . More than that, each inclusion  $G_{\leq t} \subseteq G_{\leq t'}$  naturally induces a linear map  $H_p(G_{\leq t}) \rightarrow H_p(G_{\leq t'})$  (more details about the constructions of these induced linear maps can be checked in (Hatcher, 2002)). In this way, we get a chain of homology vector spaces connected by linear maps

$$H_p(\mathcal{F}(G)) : 0 \rightarrow \dots \rightarrow H_p(G_{\leq t_1}) \rightarrow H_p(G_{\leq t_2}) \rightarrow H_p(G_{\leq t_3}) \rightarrow \dots \rightarrow H_p(G). \quad (4)$$

Such structure  $H_p(\mathcal{F}(G))$  is called the *persistent homology* of the graph filtration  $\mathcal{F}(G)$ , or say, the graph filtration induced by edge filtration  $f$ . Such algebraic structure in fact is a graded module over the polynomial ring  $\mathbb{F}_2[t]$  (Zomorodian & Carlsson, 2004), which is well studied in commutative algebraic (Atiyah & MacDonald, 1994). By the structure theorem of finitely generated modules over a principal ideal domain, we know that  $H_p(\mathcal{F}(G))$  can be essentially decomposed into a direct sum of cyclic modules (Theorem 2.1 in (Zomorodian & Carlsson, 2004)). Each indecomposable cyclic module is uniquely determined by a pair of numbers  $(t_1, t_2)$ , which essentially corresponds to the the lifecycle of a persistent topological feature. The multiset of all such pairs is called the *persistence diagram* of the graph filtration. Or equivalently, it can be viewed as a multiset of intervals, which is known as the *persistent barcode* of the graph filtration. It is also shown in (Zomorodian & Carlsson, 2004) that the persistent barcode (or equivalent persistence diagram) is a complete discrete invariant of the persistent homology, which is guaranteed by the structure theorem. In other words, the persistent barcode is the most expressive possible representation of the persistent homology. It fully encodes all topological features represented by the persistent homology.

### Bottleneck Distance:

**Definition B.1** (Bottleneck Distance). *Let  $P_1$  and  $P_2$  be two persistent barcodes. A partial matching  $\pi$  between  $P_1$  and  $P_2$  is a subset of  $P_1 \times P_2$  such that each point in  $P_1$  and  $P_2$  appears in at most one pair in  $\pi$ . For any  $p = (p_1, p_2) \in \mathbb{R}^2$ , denote  $\bar{p} = (p_2 - p_1, p_2 - p_1)$ . Geometrically,  $\bar{p}$  is the closest point of  $p$  to the diagonal line  $\Delta = \{(x, x) \mid x \in \mathbb{R}\}$ . The bottleneck distance between  $P_1$  and  $P_2$  is defined as:*

$$d_B(P_1, P_2) = \inf_{\pi} \max \left\{ \max_{(p,q) \in \pi} \|p - q\|_{\infty}, \max_{p \in P_1 \setminus \pi_1} \|p - \bar{p}\|_{\infty}, \max_{q \in P_2 \setminus \pi_2} \|q - \bar{q}\|_{\infty} \right\} \quad (5)$$

where:

- $\pi$  ranges over all partial matchings between  $P_1$  and  $P_2$
- $\|\cdot\|_{\infty}$  denotes the  $\infty$ -norm
- $\pi_1$  and  $\pi_2$  denote the projections of  $\pi$  onto  $P_1$  and  $P_2$  respectively

Intuitively, bottleneck distance measures the minimum cost of transforming one barcode to another by moving each point to another point in the other barcode. The cost is measured by the maximum distance between matched points in a partial matching, or the maximum distance between the rest unmatched points to the diagonal line  $\Delta$ . The bottleneck distance is a metric on the space of persistent barcodes, which is well studied in topological data analysis (Edelsbrunner

& Harer, 2010; Zomorodian & Carlsson, 2004; Chazal et al., 2014; Dey & Xin, 2018). It is shown in (Edelsbrunner & Harer, 2010) that the bottleneck distance is a stable metric on the space of persistent barcodes, which means that small perturbations in the input data will not change the bottleneck distance too much. This property is crucial for the stability of the persistent homology, which is a key property for the robustness and differentiability of all vector representations based on persistent homology (Carrière et al., 2020; Hofer et al., 2019, 2020; Xin et al., 2023; Mukherjee et al., 2024).

## **C More Details of $d_{\text{topo}}$**

...

## **D More Details about the Experiments**

...