TOPOLOGICALLY REGULARIZED DATA EMBEDDINGS

Anonymous authors
Paper under double-blind review

ABSTRACT

Unsupervised feature learning often finds low-dimensional embeddings that capture the structure of complex data. For tasks for which expert prior topological knowledge is available, incorporating this into the learned representation may lead to higher quality embeddings. For example, this may help one to embed the data into a given number of clusters, or to accommodate for noise that prevents one from deriving the distribution of the data over the model directly, which can then be learned more effectively. However, a general tool for integrating different prior topological knowledge into embeddings is lacking. Although differentiable topology layers have been recently developed that can (re)shape embeddings into prespecified topological models, they have two important limitations for representation learning, which we address in this paper. First, the currently suggested topological losses fail to represent simple models such as clusters and flares in a natural manner. Second, these losses neglect all original structural (such as neighborhood) information in the data that is useful for learning. We overcome these limitations by introducing a new set of topological losses, and proposing their usage as a way for *topologically regularizing* data embeddings to naturally represent a prespecified model. We include thorough experiments on synthetic and real data that highlight the usefulness and versatility of this approach, with applications ranging from modeling high-dimensional single cell data, to graph embedding.

1 INTRODUCTION

Motivation Modern data often arrives in complex forms that complicate their analysis. For example, high-dimensional data cannot be visualized directly, whereas relational data such as graphs lack the natural vectorized structure required by various machine learning models (Bhagat et al., 2011; Kazemi & Poole, 2018; Goyal & Ferrara, 2018). Representation learning aims to derive mathematically and computationally convenient representations to process and learn from such data. However, obtaining an effective representation is often challenging, for example, due to the accumulation of noise in high-dimensional biological expression data (Vandaele et al., 2021). In other examples such as community detection in social networks, graph embeddings struggle to clearly separate communities due to the few interconnections between them. In such cases, expert prior knowledge of the topological model may improve learning from, visualizing, and interpreting the data. Unfortunately, a general tool for incorporating prior topological knowledge in representation learning is lacking.

In this paper, we introduce such tool under the name of *topological regularization*. Here, we build on the recently developed differentiation frameworks for optimizing data to capture topological properties of interest (Gabrielsson et al., 2020; Solomon et al., 2021; Carriere et al., 2021). Unfortunately, such *topological optimization* has been poorly studied within the context of representation learning. For example, the used *topological losses* are indifferent to any structure other than topological, such as neighborhood information, which may be useful for learning. Therefore, topological optimization often destructs natural and informative properties of the data in favor of the topological loss.

Our proposed method of *topological regularization effectively resolves this by learning an embedding representation that incorporates the topological prior*. As we will see in this paper, these priors can be directly postulated through topological loss functions. For example, if the prior is that the data lies on a circular model, we design a loss function that is lower whenever a more prominent cycle is present in the embedding. By extending the previously suggested topological losses to fit a wider set of models, we show that topological regularization effectively embeds data according to a variety of topological priors, ranging from clusters, cycles, and flares, to any combination of these.

Related Work Certain methods that incorporate topological information into representation learning have already been developed. For example, Deep Embedded Clustering (Xie et al., 2016) simultaneously learns feature representations and cluster assignments using deep neural networks. Constrained embeddings of Euclidean data on spheres have also been studied by Bai et al. (2015). However, such methods often require an extensive development for one particular kind of input data and topological model. Contrary to this, incorporating topological optimization into representation learning provides a simple yet versatile approach towards combining data embedding with topological priors, that generalizes well to any input data as long as the output is a point cloud embedding.

Topological autoencoders (Moor et al., 2020) already combine topological optimization with a data embedding procedure. The main difference here is that the topological information used for optimization is obtained from the original high-dimensional data, and not passed as a prior. While this may sound as a major advantage—and certainly can be as shown by Moor et al. (2020)—obtaining such topological information heavily relies on distances between observations, which are often meaningless and unstable in high dimensions (Aggarwal et al., 2001). Furthermore, certain constructions such as the α-*filtration* obtained from the Delanauy triangulation—which we will use extensively and is further discussed in Appendix A—are expensive to obtain from high-dimensional data (Cignoni et al., 1998), and are best computed from the low-dimensional embedding.

Contributions We include a sufficient background on *persistent homology*—the main tool behind topological optimization—in Appendix A (note that all of its concepts important for this paper are summarized in Figure 1). We summarize the previous idea behind topological optimization of point clouds (Section 2.1). We also introduce a new set of losses to model a wider variety of models in a natural manner (Section 2.2), which can be used to topologically regularize embeddings, for which the result—not necessarily the input—is a point cloud (Section 2). We include experiments on synthetic and real data that show the usefulness and versatility of topological regularization, and provide additional insights into the performance of data embedding methods (Section 3). We discuss open problems in topological representation learning and conclude on our work in Section (4).

2 METHODS

The main purpose of this paper is to present a method to incorporate *prior topological knowledge* in a point cloud embedding $\boldsymbol{E}$ (dimensionality reduction, graph embedding, ...) of a data set $\mathbb{X}$. As will become clear below, these topological priors can be directly postulated through *topological loss functions* $\mathcal{L}_{\text{top}}$. Then, the goal is to find an embedding that that minimizes a total loss

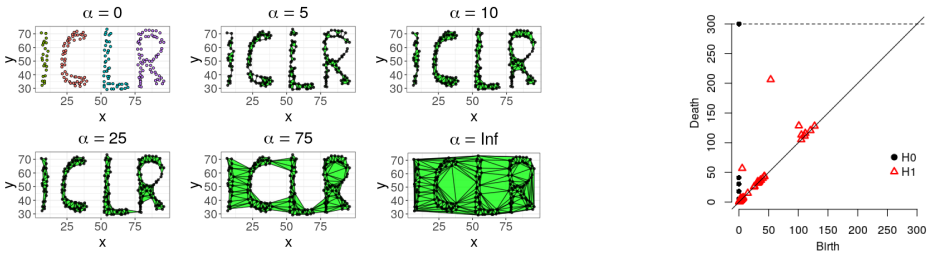
$$\mathcal{L}_{\text{tot}}(\boldsymbol{E}, \mathbb{X}) \coloneqq \mathcal{L}_{\text{emb}}(\boldsymbol{E}, \mathbb{X}) + \lambda_{\text{top}} \mathcal{L}_{\text{top}}(\boldsymbol{E}), \tag{1}$$

where $\mathcal{L}_{\text{emb}}$ is a loss that aims to preserve structural attributes of the original data, and $\lambda_{\text{top}} > 0$ controls the strength of *topological regularization*. Note that, $\mathbb{X}$ itself is not required to be a point cloud, or reside in the same space as $\boldsymbol{E}$, which is especially useful for representation learning.

In this section, we mainly focus on topological optimization of point clouds, that is, the loss $\mathcal{L}_{\text{top}}$. The basic idea behind this recently introduced method—as presented by Gabrielsson et al. (2020)—is illustrated in Section 2.1. We also show that direct topological optimization may neglect important structural information such as neighborhoods, which can effectively be resolved through (1). Hence, as we will also see in Section 3, while representation learning may benefit from topological losses for incorporating prior topological knowledge, topological optimization itself may also benefit from other structural losses as to represent the topological prior in a more truthful manner. Nevertheless, some topological models remain difficult to represent in a natural manner through topological optimization. Therefore, we introduce a new set of topological losses, and provide an overview of how different topological models can be postulated through them in Section 2.2. Experiments with and comparisons to topological regularization of embeddings through (1) will be presented in Section 3.

2.1 BACKGROUND ON TOPOLOGICAL OPTIMIZATION OF POINT CLOUDS

Topological optimization is performed through a *topological loss function* evaluated on the *persistence diagram(s)* of the data (Carlsson, 2009). These diagrams—obtained through a method termed *persistent homology* and further discussed in Appendix A—summarize all from the finest to coarsest topological holes (connected components, cycles, voids, ...) in the data, as illustrated in Figure 1.



(a) (b)

Figure 1: The two basic concepts from persistent homology important for our method. (a) Persistent homology quantifies topological changes in a *filtration*, i.e., a changing sequence of simplicial complexes ordered by inclusion, parameterized by a time parameter $t = \alpha \in \mathbb{R}$, is constructed from a point cloud. Various topological holes are either born or die during this filtration. Here the filtration starts of with one connected component per data point (0-dimensional holes), which can only merge together (resulting in the death of such components) when including additional edges. For larger values of α, we observe the birth of cycles (1-dimensional holes), which are consecutively filled in (and thus die) when α increases even further. Eventually, one single connected component persists indefinitely. (b) The results from persistent homology are commonly visualized through a *persistence diagram*. Here, a tuple (b, d) marks a topological hole—in this case a connected component (H0) or a cycle (H1)—that is born at time b and that dies at (possibly infinite) time d in a filtration.

While methods that learn from persistent homology are now both well-developed and diverse (Pun et al., 2018), optimizing the data representation for the persistent homology thereof has only been gaining recent attention (Gabrielsson et al., 2020; Solomon et al., 2021; Carriere et al., 2021). Persistent homology has a rather abstract mathematical foundation within the field of algebraic topology (Hatcher, 2002), and its computation is inherently combinatorial (Zomorodian & Carlsson, 2005). This complicates working with usual derivatives for optimization. To accommodate for this, topological optimization makes use of Clarke subderivatives (Clarke, 1990), whose applicability to persistence builds on arguments from o-minimal geometry (van den Dries, 1998; Carriere et al., 2021). Fortunately, thanks to the recent work of Gabrielsson et al. (2020) and Carriere et al. (2021), powerful tools for topological optimization have been developed for software libraries such as PyTorch and TensorFlow, allowing their application without deeper knowledge of these mathematical subjects.

Mathematically, topological optimization optimizes the data representation with respect to the topological information summarized by its persistence diagram(s) $\mathcal{D}$. We will use the same approach by Gabrielsson et al. (2020), where all (birth, death) tuples $(b_1, d_1), (b_2, d_2), \ldots, (b_{|\mathcal{D}|}, d_{|\mathcal{D}|})$ in $\mathcal{D}$ are first ordered according to decreasing persistence $d_k - b_k$. The points (b, d) with $d = \infty$ (these are usually plotted on top of the diagram, such as in Figure 1b,) form the *essential part* $\mathcal{D}^{\text{ess}}$ of $\mathcal{D}$. The points with finite coordinates form the *regular part* $\mathcal{D}^{\text{reg}}$ of $\mathcal{D}$. For $i_{\text{ess}} \leq j_{\text{ess}}$, $i_{\text{reg}} \leq j_{\text{reg}}$, and functions $g_{\text{ess}} : \mathbb{R} \to \mathbb{R}$, $g_{\text{reg}} : \mathbb{R}^2 \to \mathbb{R}$, we can now define a *topological loss function*

$$\mathcal{L}_{\text{top}}(\mathcal{D}) \coloneqq \sum_{\substack{i_{\text{ess}} \leq k \leq j_{\text{ess}} \\ (b_k, d_k) \in \mathcal{D}^{\text{ess}}}} g_{\text{ess}}(b_k) + \sum_{\substack{i_{\text{reg}} \leq k \leq j_{\text{reg}} \\ (b_k, d_k) \in \mathcal{D}^{\text{reg}}}} g_{\text{reg}}(b_k, d_k). \tag{2}$$

It turns out that for many useful definitions of g_{ess} and g_{reg}, $\mathcal{L}_{\text{top}}(\mathcal{D})$ has a well-defined Clarke subdifferential with respect to the parameters defining the filtration from which the persistence diagram $\mathcal{D}$ is obtained. In this paper, we will consistently use the α-*filtration* as shown in Figure 1a (see Appendix A for its formal definition), and these parameters are entire point clouds $\mathbb{X} \in (\mathbb{R}^d)^n$ of size n in the d-dimensional Euclidean space. $\mathcal{L}_{\text{top}}(\mathcal{D})$ can then be easily optimized with respect to these parameters through standard stochastic subgradient algorithms (Carriere et al., 2021).

Within this entire paper, we only use the regular part of the diagram (this coincides with letting $g_{\text{ess}} \equiv 0$), and let $g_{\text{reg}} : \mathbb{R}^2 \to \mathbb{R} : (b, d) \mapsto \mu(d - b)$ be proportional to the *persistence function*. By having ordered the points by persistence, $\mathcal{L}_{\text{top}}$ is now a *function of persistence* on $\mathbb{R}^{|\mathcal{D}|}$, i.e., it is invariant to permutations of the points in $\mathcal{D}$ (Carriere et al., 2021). The factor of proportionality $\mu \in \{1, -1\}$ indicates whether we want to *minimize* ($\mu = 1$) or *maximize* ($\mu = -1$) persistence, i.e, the prominence of the topological hole, or thus, how well clusters, cycles, ..., are (not) represented.

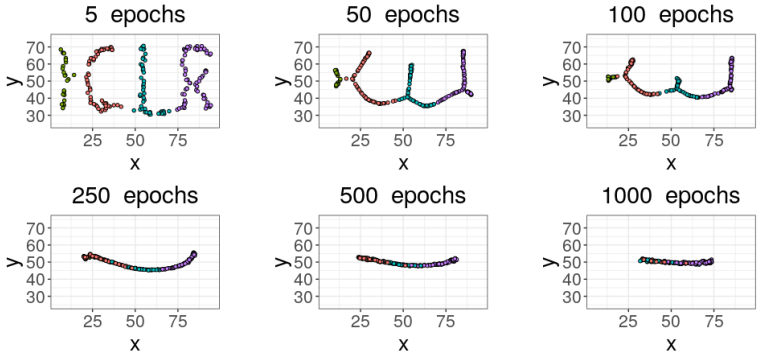


Figure 2: The data set in Figure 1a, optimized to have a low total 0-dimensional persistence. Points are colored according to their initial grouping along one of the four letters in the 'ICLR' acronym.

The topological loss function in (2) then reduces to

$$\mathcal{L}_{\text{top}}(\boldsymbol{E}) \coloneqq \mathcal{L}_{\text{top}}(\mathcal{D}) = \mu \sum_{\substack{i:=i_{\text{reg}} \leq k \leq j_{\text{reg}}=:j \\ (b_k,d_k)\in\mathcal{D}^{\text{reg}}}} (d_k - b_k)\,. \tag{3}$$

Here, the data matrix $\boldsymbol{E}$ (in this paper the embedding) defines the diagram $\mathcal{D}$ through persistent homology of the α-filtration of $\boldsymbol{E}$, and a persistence (topological hole) dimension to optimize for.

For example, consider (3) with $i = 2$, $j = \infty$, $\mu = 1$, restricted to 0-dimensional persistence (measuring the prominence of connected components) of the α-filtration. Figure 2 shows the data from Figure 1 optimized for this loss function for various epochs. The optimized point cloud quickly resembles a single connected component for smaller numbers of epochs. This is the single goal of the loss (3), which neglects all other structural structural properties of the data such as its underlying cycles (e.g., the circular hole in the 'R') or local neighborhoods. Larger numbers of epochs mainly affect the scale of the data. While this scale has an absolute effect on the total persistence, the point cloud visually represents a single connected topological component equally well. We also observe that while local neighborhoods are preserved well during the first epochs simply by nature of the topological optimization procedure, they are increasingly distorted for a larger number of epochs.

2.2 Newly Proposed Topological Loss Functions

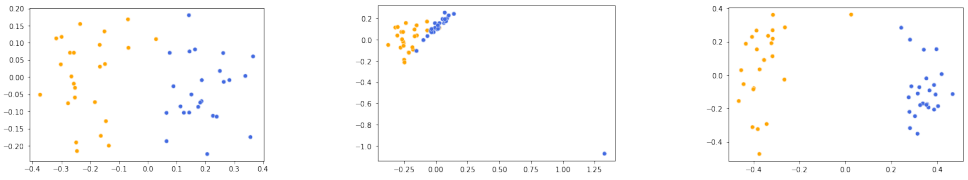
In this paper, the prior topological knowledge incorporated into the point cloud data matrix embedding $\boldsymbol{E}$ is directly postulated through a topological loss function. For example, letting $\mathcal{D}$ be the 0-dimensional persistence diagram of $\boldsymbol{E}$, and choosing $i = 2$, $j = \infty$, and $\mu = 1$ in (3), corresponds to the criterion that $\boldsymbol{E}$ should represents one closely connected component, as illustrated in Figure 2. Therefore, we often regard a *topological loss* as a *topological prior*, and vice versa.

Unfortunately, although persistent homology effectively measures the prominence of topological holes, topological optimization is often ineffective for representing such holes in a natural manner. An extreme example of this are clusters, despite the fact that they are captured through the simplest form of persistence, i.e., 0-dimensional. This is shown in Figure 3, where we sampled data $\boldsymbol{E}$ from two Gaussian distributions centered at different means in $\mathbb{R}^2$ (Figure 3a). Optimizing the point cloud for (at least) two clusters can be done by defining $\mathcal{L}_{\text{top}}(\boldsymbol{E})$ as in (3), letting $\mathcal{D}$ be the 0-dimensional persistence diagram of $\boldsymbol{E}$, $i = j = 2$, and $\mu = -1$. However, we observe that topological optimization simply displaces one single point away from all other points (Figure 3b). Note that purely topologically, this is indeed a correct representation of two clusters.

To encourage more natural holes, we propose to conduct the topological optimization for the loss

$$\widetilde{\mathcal{L}}_{\text{top}}(\boldsymbol{E}) \coloneqq \mathbb{E}\left(\mathcal{L}_{\text{top}}\left(\{\boldsymbol{x} \in \mathbf{S} : \mathbf{S} \text{ is a random sample of } \boldsymbol{E} \text{ with sampling fraction } f_{\mathcal{S}}\}\right)\right), \tag{4}$$

where $\mathcal{L}_{\text{top}}$ is defined as in (3). In practice, during each optimization iteration, $\widetilde{\mathcal{L}}_{\text{top}}$ is approximated by the mean of $\mathcal{L}_{\text{top}}$ evaluated over $n_{\mathcal{S}}$ random samples of $\boldsymbol{E}$. The idea behind this approach is that *a topological model that is naturally present in the data should be represented well by many subsets*



(a) Initial point cloud. (b) Ordinary top. optimization. (c) Optimization with (4).

Figure 3: A point cloud sampled from two ground truth clusters (labeled by color) topologically optimized without and with sampling. The optimization with sampling results in more natural clusters.

of the data. Figure 3 shows the result for a sampling fraction $f_{\mathcal{S}} = 0.1$ and $n_{\mathcal{S}} = 1$. The new data representation visualizes the clusters already well and far more naturally. An added benefit of the new loss (4) is that topological optimization can be conducted significantly faster for reasonably lower $n_{\mathcal{S}}$, as the α-filtration and persistent homology are evaluated on smaller samples.

In summary, various topological priors can now be formulated through topological losses as follows.

k-dimensional holes Optimizing for k-dimensional holes ($k = 0$ for clusters), can generally be done through (3) or (4), by letting $\mathcal{D}$ be the corresponding k-dimensional persistence diagram. The terms i and j in the summation are used to express how many holes one exactly, at least, or at most wants. Finally, μ can be chosen to either decrease ($\mu = 1$) or increase ($\mu = -1$) persistence.

Flares Persistent homology is invariant to certain topological changes. For example, both a linear 'I'-structured model and a bifurcating 'Y'-structured model consist of one connected component, and no higher-dimensional holes. These models are indistinguishable based on the (persistent) homology thereof, even though they are topologically different in terms of their singular points.

Capturing such additional topological phenomena is possible through a refinement of persistent homology under the name of *functional persistence*, also well discussed and illustrated by Carlsson (2014). The idea is that instead of evaluating persistent homology on a data matrix $\boldsymbol{E}$, we evaluate it on a subset $\{\boldsymbol{x} \in \boldsymbol{E} : f(\boldsymbol{x}) \leq \tau\}$ for a well chosen function $f : \boldsymbol{E} \to \mathbb{R}$ and hyperparameter τ.

Inspired by this approach, for a diagram $\mathcal{D}$ of a point cloud $\boldsymbol{E}$, we propose the topological loss

$$\widetilde{\mathcal{L}}_{\text{top}}(\boldsymbol{E}) \coloneqq \mathcal{L}_{\text{top}}\left(\{\boldsymbol{x} \in \boldsymbol{E} : f_{\boldsymbol{E}}(\boldsymbol{x}) \leq \tau\}\right), \text{ informally denoted } \left[\mathcal{L}_{\text{top}}(\mathcal{D})\right]_{f_{\boldsymbol{E}}^{-1}]-\infty,\tau]}, \tag{5}$$

where f is a real-valued function on $\boldsymbol{E}$, possibly dependent on $\boldsymbol{E}$—which changes during optimization—itself, τ a hyperparameter, and $\mathcal{L}_{\text{top}}$ is an ordinary topological loss as defined by (3). In particular, we will focus on the case where f equals a scaled centrality measure on $\boldsymbol{E}$:

$$f_{\boldsymbol{E}} \equiv \mathcal{E}_{\boldsymbol{E}} \coloneqq 1 - \frac{g_{\boldsymbol{E}}}{\max g_{\boldsymbol{E}}}, \text{ where } g_{\boldsymbol{E}} : \boldsymbol{E} \to \mathbb{R} : \boldsymbol{x} \mapsto \left\| \boldsymbol{x} - \frac{1}{|\boldsymbol{E}|} \sum_{\boldsymbol{y} \in \boldsymbol{E}} \boldsymbol{y} \right\|. \tag{6}$$

For $\tau \geq 1$, $\widetilde{\mathcal{L}}_{\text{top}}(\boldsymbol{E}) = \mathcal{L}_{\text{top}}(\boldsymbol{E})$. For sufficiently small $\tau > 0$, $\widetilde{\mathcal{L}}_{\text{top}}$ evaluates $\mathcal{L}_{\text{top}}$ on the points 'far away' from the center of $\boldsymbol{E}$. As we will see in the experiments below, this is especially useful in conjunction with 0-dimensional persistence to optimize for flares in the point cloud representation.

Combinations Naturally, through linear combination of loss functions, different topological priors can be combined, e.g., if we want the represented model to both be connected and include a cycle.

3 Experiments

In this section, we show how our proposed topological regularization of data embeddings (1) leads to a powerful and versatile approach for representation learning. In particular, we show that

- embeddings benefit from prior topological knowledge through topological regularization;
- conversely, topological optimization may also benefit from incorporating structural information as captured through embedding losses, leading to more qualitative representations;

- subsequent learning tasks may benefit from expert prior topological knowledge.

In Section 3.1, we show how topological regularization improves standard PCA dimensionality reduction and allows better understanding of its performance when noise is accumulated over many dimensions. In Section 3.2, we present applications to high-dimensional single cell trajectory data and graph embedding. Quantitative results are discussed in Section 3.3.

Topological optimization was performed in Pytorch, using code adapted from Gabrielsson et al. (2020). Appendix B discusses a supplementary graph embedding experiment where we embed the Harry Potter network according to a circular prior. Data sizes, hyperparameters, losses, and optimization times are summarized in Tables 1 & 2. All code for this project is available on `https://dropbox.com/sh/2n1z9fnh436869e/AAC5LMKIxi7CiCCwILAPgBXDa?dl=0`.

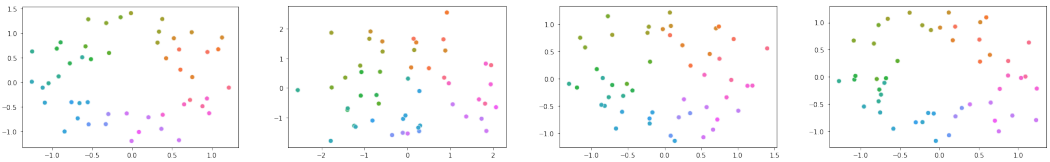
3.1 Synthetic Data

We sampled 50 points uniformly from the unit circle in $\mathbb{R}^2$. We then added 500-dimensional noise to the resulting data matrix $\boldsymbol{X}$, where the noise in each dimension is sampled uniformly from $[-0.45, 0.45]$. Since the additional noisy features are irrelevant to the topological (circular) model, an ideal projection embedding $\boldsymbol{X}$ is its restriction to its first two data coordinates (Figure 4a).

However, it is probabilistically unlikely that that the irrelevant features will have a zero contribution to a PCA embedding of the data (Figure 4b). Measuring the feature importance of each feature as the sum of its two absolute contributions (the loadings) to the projection, we observe that most of the 498 irrelevant features have a small nonzero effect on the PCA embedding (Figure 5). Intuitively, each added feature slightly shifts the projection plane away from the plane spanned by the first two coordinates. As a result, the circular hole is less prominent in the PCA embedding of the data.

We can regularize this embedding using a topological loss function $\mathcal{L}_{\text{top}}$ measuring the persistence of the most prominent 1-dimensional hole in the embedding ($i = j = 1$ in (3)). For a simple Pytorch compatible implementation, we used $\mathcal{L}_{\text{emb}}(\boldsymbol{W}, \boldsymbol{X}) \coloneqq \text{MSE}\left(\boldsymbol{X}\boldsymbol{W}\boldsymbol{W}^T, \boldsymbol{X}\right)$, as to minimize the reconstruction error between $\boldsymbol{X}$ and its linear projection obtained through $\boldsymbol{W}$. To this, we added the loss $10^4\mathcal{L}_{\perp}(\boldsymbol{W})$, where $\mathcal{L}_{\perp}(\boldsymbol{W}) \coloneqq \|\boldsymbol{W}^T\boldsymbol{W} - \boldsymbol{I}_2\|_2$ is used to encourage orthonormality of the matrix $\boldsymbol{W}$ to be optimized, initialized with the PCA-loadings. The resulting embedding is shown in Figure 4d, which better captures the circular hole (with $\mathcal{L}_{\perp}(\boldsymbol{W}) \sim 0.03$). Furthermore, we see that irrelevant features now more often contribute less to the embedding according to $\boldsymbol{W}$ (Figure 5).

For comparison, Figure 4c shows the optimization of $\boldsymbol{W}$ without accounting for the reconstruction loss ($\mathcal{L}_{\perp}(\boldsymbol{W})$ still included). From this and also from Figure 5, we observe that $\boldsymbol{W}$ struggles more to converge to the correct projection, resulting in a slightly less prominent hole ($\mathcal{L}_{\perp}(\boldsymbol{W}) \sim 0.2$).



(a) First two data coordinates. (b) Ordinary PCA embedding. (c) Top. optimized projection. (d) Top. regularized embedding.

Figure 4: Various representations of the 500-dimensional synthetic data $\boldsymbol{X}$. The coloring represents the positioning of points without noise.

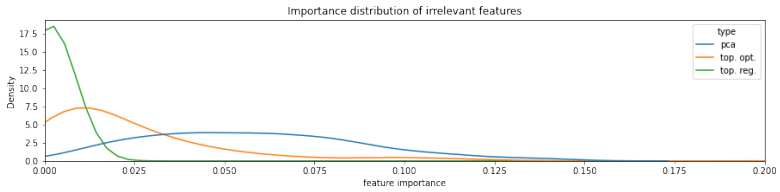
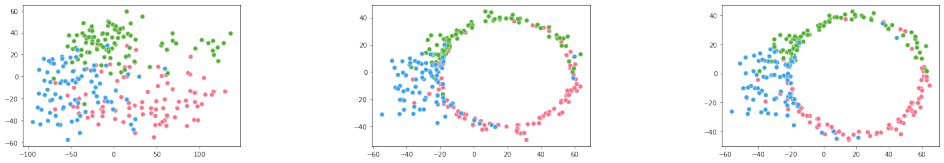
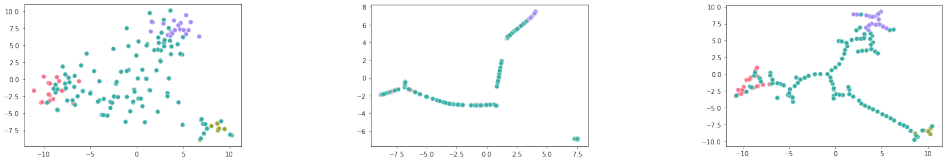


Figure 5: Feature importance densities of the 498 irrelevant features in the PCA embedding (blue), the top. optimized PCA embedding (orange), and the top. regularized PCA embedding (green).



(a) Ordinary PCA embedding. (b) Top. optimized projection. (c) Top. regularized embedding.

Figure 6: Various representations of the cyclic cell data. Colors represent the cell grouping.



(a) Ordinary UMAP embedding. (b) Top. optimized embedding. (c) Top. regularized embedding.

Figure 7: Various representations of the bifurcating cell data. Colors represent the cell grouping.

3.2 Real Data

Circular Cell Trajectory Data We considered a single cell trajectory data set of 264 cells in a 6812-dimensional gene expression space (Cannoodt et al., 2018; Saelens et al., 2019). The ground truth model—which can be considered a snapshot of the cells at a fixed time—is a circular model connecting three distinct cell groups through cell differentiation. It has been shown by Vandaele (2020) that real single cell data with such models are difficult to embed in a circular manner.

To explore this, we repeated the experiment with the same losses as in Section 3.1 on this data, where the (expected) topological loss is now modified through (4) with $f_{\mathcal{S}} = 0.25$, and $n_{\mathcal{S}} = 1$. From Figure 6a, we see that while the ordinary PCA embedding does somehow respect the positioning of the cell groups (marked by their color), it indeed struggles to embed the data in a manner that visualizes the present circular hole. However, as shown in Figure 6c, by topologically regularizing the embedding we are able to embed the data much better in a circular manner ($\mathcal{L}_{\perp}(\boldsymbol{W}) \sim 4e^{-3}$).

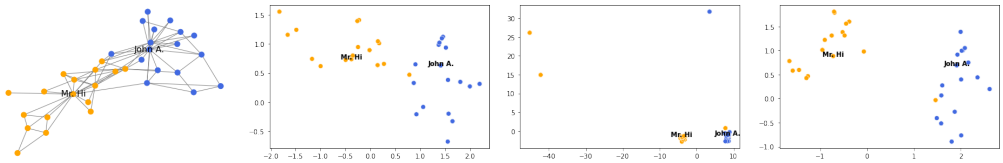
Figure 6b shows the optimization of $\boldsymbol{W}$ without the reconstruction loss. The embedding is similar to the one in Figure 6c, with the pink colored cell group slightly more dispersed ($\mathcal{L}_{\perp}(\boldsymbol{W}) \sim 4e^{-3}$).

Bifurcating Cell Trajectory Data We considered a second cell trajectory data set of 154 cells in a 1770-dimensional expression space (Cannoodt et al., 2018). The ground truth here is a bifurcating model connecting four different cell groups through cell differentiation. However, this time we used the UMAP loss for the embeddings. We used a topological loss $\mathcal{L}_{\text{top}} \equiv \mathcal{L}_{\text{conn}} - \mathcal{L}_{\text{flare}}$, where $\mathcal{L}_{\text{conn}}$ measures the total (sum of) finite 0-dimensional persistence in the embedding to encourage connectedness of the representation, and $\mathcal{L}_{\text{flare}}$ is as in (5), measuring the persistence of the third most prominent 0-dimensional hole in $\{\boldsymbol{y} \in \boldsymbol{E} : \mathcal{E}_{\boldsymbol{E}}(\boldsymbol{y}) \leq 0.75\}$, where $\mathcal{E}_{\boldsymbol{E}}$ is as in (6). Thus, $\mathcal{L}_{\text{flare}}$ is used to optimize for a 'flare' with (at least) three clusters away from the embedding mean. We observe that while the ordinary UMAP embedding is more 'blobby' (Figure 7a), the topologically regularized embedding is more constrained towards a connected bifurcating shape (Figure 7c).

For comparison, we conducted topological optimization for the loss $\mathcal{L}_{\text{flare}}$ of the initialized UMAP embedding without the UMAP embedding loss. The resulting embedding is now more fragmented (Figure 7b). We thus see that topological optimization may also benefit from the embedding loss.

Graph Embedding The topological loss in (1) can be evaluated on any embedding, and does not require a point cloud as original input. We can thus use topological regularization for embedding a graph G, to learn a representation of the nodes of G in $\mathbb{R}^d$ that well respects properties of G.

To explore this, we considered the Karate network (Zachary, 1977), a well known and studied network within graph mining that consists of two different communities. The communities are represented by two key figures (John A. and Mr. Hi), as shown in Figure 8a. To embed the graph, we used a DeepWalk variant adapted from Dagar et al. (2020). While the ordinary DeepWalk em-



(a) The Karate network. (b) Ordinary DeepWalk embedding. (c) Top. optimized embedding. (d) Top. regularized embedding.

Figure 8: The Karate network and various of its embeddings.

bedding (Figure 8b) well respects the ordering of points according to their communities, the two communities remained close to each other. We thus regularized this embedding using the topological loss as defined by (4), where $\mathcal{L}_{\text{top}}$ measures the persistence of the second most prominent 0-dimensional hole, and $f_{\mathcal{S}} = 0.25$, $n_{\mathcal{S}} = 10$. The resulting embedding (Figure 8d) now nearly perfectly separates the two ground truth communities present in the graph.

Topological optimization of the initialized DeepWalk embedding with the same topological loss but without the DeepWalk loss creates some natural community structure, but also results in a few outliers (Figure 8c). Thus, although our introduced loss (4) enables more natural topological modeling to some extent, we again observe that using this in conjunction with embedding losses, i.e., our proposed method of topological regularization, leads to the best qualitative results.

3.3 Quantitative Evaluation

Table 3 summarizes the embedding and topological losses we obtained for the ordinary embeddings, the topologically optimized embeddings (initialized with the ordinary embeddings, but not using the embedding loss), as well as for the topologically regularized embeddings. As one would expect, topological regularization balances the embedding losses between the embedding losses of the ordinary and topologically optimized embeddings. More interestingly, topological regularization may actually result in a more optimal, i.e., lower topological loss than topological optimization only, here in particular for the synthetic cycle data and Harry Potter graph. This suggest that combining topological information with other structural information may facilitate convergence to the correct embedding model, as we also qualitatively confirmed for these data sets (see also Appendix B). We also observe that there are more significant differences in the obtained topological losses than in the embedding losses with and without regularization. This suggests that the optimum region for the embedding loss may be somewhat flat with respect to the corresponding region for the topological loss. Thus, slight shifts in the local embedding optimum, e.g., as caused by noise, may result in much worse topological embedding models, which can be resolved through topological regularization.

Table 1: Summarization of the data, hyperparameters and optimization times.

data	**size**	**method**	**lr**	**epochs**	λ_{top}	***t* w/o top**	***t* with top**
Synthetic Cycle	50×500	PCA	1e-1	500	1e1	<1s	5s
Cell Cycle	264×6812	PCA	5e-4	1000	1e2	<1s	35s
Cell Bifurcating	154×1770	UMAP	1e-1	100	1e1	<1s	8s
Karate	34×78	DeepWalk	1e-2	50	5e1	29s	29s
Harry Potter	58×217	InnerProd	1e-1	100	1e-1	36s	34s

Table 2: Summary of the topological losses computed from persistence diagrams $\mathcal{D}$ with points (b_k, d_k) ordered by persistence $d_k - b_k$. Note that for 0-th dimensional homology diagrams $d_1 = \infty$.

data	**top. loss function**	**dimension of hole**	$f_{\mathcal{S}}$	$n_{\mathcal{S}}$
Synthetic Cycle	$-(d_1 - b_1)$	1	N/A	N/A
Cell Cycle	$-(d_1 - b_1)$	1	0.25	1
Cell Bifurcating	$\sum_{k=2}^{\infty}(d_k - b_k) - [d_3 - b_3]_{\mathcal{E}_{\boldsymbol{E}}^{-1}]-\infty,0.75]}$	0 - 0	N/A	N/A
Karate	$-(d_2 - b_2)$	0	0.25	10
Harry Potter	$-(d_1 - b_1)$	1	N/A	N/A

Table 3: Embedding/reconstruction and topological losses of the final embeddings. Lowest in bold.

data	embedding loss			topological loss		
	ord. emb.	top. opt.	top. reg.	ord. emb.	top. opt.	top. reg.
Synthetic Cycle	$\mathbf{6.3e^{-2}}$	$6.7e^{-2}$	$6.6e^{-2}$	-0.15	-0.35	$\mathbf{-0.75}$
Cell Cycle	$\mathbf{6.70}$	7.00	6.99	-13.4	$\mathbf{-50.9}$	-49.7
Cell Bifurcating	$\mathbf{8576}$	9933	8871	117	$\mathbf{23}$	63
Karate	$\mathbf{2006}$	N/A	2112	-1.3	$\mathbf{-28.5}$	-2.4
Harry Potter	$\mathbf{0.20}$	1.11	0.23	-0.82	-2.37	$\mathbf{-3.05}$

Table 4: Embedding performance evaluations for label prediction. Highest in bold.

data	metric	ord. emb.	top. opt.	top. reg.
Synthetic Cycle	r^2	0.56 ± 0.47	0.77 ± 0.24	$\mathbf{0.85 \pm 0.14}$
Cell Cycle	accuracy	0.79 ± 0.07	0.79 ± 0.07	$\mathbf{0.81 \pm 0.07}$
Cell Bifurcating	accuracy	0.79 ± 0.08	0.81 ± 0.07	$\mathbf{0.82 \pm 0.08}$
Karate	accuracy	$\mathbf{0.97 \pm 0.08}$	0.91 ± 0.14	$\mathbf{0.97 \pm 0.08}$

We also evaluated the quality of the embedding visualizations presented in this section, by assessing how informative they are for predicting the ground data truth labels. For the Synthetic Cycle data, these labels are the 2D coordinates of the noise-free data on the unit circle in $\mathbb{R}^2$, and we used a multi-ouput support vector regressor model. For the cell trajectory data and Karate network, we used the ground truth cell groupings and community assignments, respectively, and a support vector machine model. All points in the 2D embeddings were then split into 90% points for training and 10% for testing. Consecutively, we used 5-fold cross-validation on the training data to tune the regularization hyperparameter $C \in \{1\mathrm{e}-2, 1\mathrm{e}-1, 1, 1\mathrm{e}1, 1\mathrm{e}2\}$. All other settings were the default from SCIKIT-LEARN. The performance of the final tuned and trained model was then evaluated on the test data, through the r^2 coefficient of determination for the regression problem, and the accuracy for all classification problems. Finally, we repeated this entire experiment 100 times. The averaged test performance metrics and their standard deviations are summarized in Table 4. From this, we observe that topological regularization consistently leads to the more informative visualization embeddings.

4 Discussion and Conclusion

We proposed a new approach for representation learning under the name of *topological regularization*, which builds on the recently developed differentiation frameworks for topological optimization. This led to a versatile and effective way for embedding data according to expert prior topological knowledge, directly postulated through (some newly introduced) topological loss functions.

A clear limitation of topological regularization is that expert prior topological knowledge is not always available. How to select the best out of a list of topological priors is thus open to further research. Furthermore, designing topological loss functions currently requires some understanding of persistent homology, and it may be useful to study how to facilitate that design process for lay users. From a foundational perspective, our work provides new research opportunities into extending the developed theory for topological optimization (Carriere et al., 2021) to our newly introduced losses and their integration into data embeddings. Finally, topological optimization based on combinatorial structures other than the α-complex may be of both theoretical and practical interest. For example, point cloud optimization based on graph-approximations such as the minimum spanning tree (Vandaele et al., 2021), or varying the functional threshold τ in the loss (5) alongside the filtration time (Chazal et al., 2009), may lead to new topological loss functions with fewer hyperparameters.

Nevertheless, through our approach, we already provided new and important insights into the performance of embedding methods, such as their potential inability to converge to the correct topological model due to the flatness of the embedding loss near its (local) optimum, with respect to the topological loss. Furthermore, we quantitatively showed that including prior topological knowledge provides a promising way to improve consecutive—even non-topological—learning tasks. In conclusion, topological regularization enables both improving and better understanding representation learning methods, for which we provided and thoroughly illustrated the first directions in this paper.

REFERENCES

Charu C Aggarwal, Alexander Hinneburg, and Daniel A Keim. On the surprising behavior of distance metrics in high dimensional space. In *International conference on database theory*, pp. 420–434. Springer, 2001.

Shuanghua Bai, Huo-Duo Qi, and Naihua Xiu. Constrained best euclidean distance embedding on a sphere: a matrix optimization approach. *SIAM Journal on Optimization*, 25(1):439–467, 2015.

Smriti Bhagat, Graham Cormode, and S Muthukrishnan. Node classification in social networks. In *Social network data analytics*, pp. 115–148. Springer, 2011.

Robrecht Cannoodt, Wouter Saelens, Helena Todorov, and Yvan Saeys. Single-cell -omics datasets containing a trajectory, October 2018. URL `https://doi.org/10.5281/zenodo.1443566`.

Gunnar Carlsson. Topology and data. *Bulletin of the American Mathematical Society*, 46(2):255–308, jan 2009. ISSN 0273-0979.

Gunnar Carlsson. Topological pattern recognition for point cloud data. *Acta Numerica*, 23:289–368, 2014.

Mathieu Carriere, Frédéric Chazal, Marc Glisse, Yuichi Ike, Hariprasad Kannan, and Yuhei Umeda. Optimizing persistent homology based functions. In *International Conference on Machine Learning*, pp. 1294–1303. PMLR, 2021.

Frédéric Chazal, David Cohen-Steiner, Leonidas J Guibas, Facundo Mémoli, and Steve Y Oudot. Gromov-hausdorff stable signatures for shapes using persistence. In *Computer Graphics Forum*, volume 28, pp. 1393–1403. Wiley Online Library, 2009.

Paolo Cignoni, Claudio Montani, and Roberto Scopigno. Dewall: A fast divide and conquer delaunay triangulation algorithm in ed. *Computer-Aided Design*, 30(5):333–341, 1998.

Frank H Clarke. *Optimization and nonsmooth analysis*. SIAM, 1990.

A. Dagar, A. Pant, S. Gupta, and S. Chandel. graph_nets, 2020. URL `https://github.com/dsgiitr/graph_nets`.

Rickard Brüel Gabrielsson, Bradley J Nelson, Anjan Dwaraknath, and Primoz Skraba. A topology layer for machine learning. In *International Conference on Artificial Intelligence and Statistics*, pp. 1553–1563. PMLR, 2020.

Palash Goyal and Emilio Ferrara. Graph embedding techniques, applications, and performance: A survey. *Knowledge-Based Systems*, 151:78–94, 2018.

Allen Hatcher. *Algebraic topology*. Cambridge University Press, 2002. ISBN 0521795400.

Seyed Mehran Kazemi and David Poole. Simple embedding for link prediction in knowledge graphs. In *Advances in neural information processing systems*, pp. 4284–4295, 2018.

Michael Moor, Max Horn, Bastian Rieck, and Karsten Borgwardt. Topological autoencoders. In *International conference on machine learning*, pp. 7045–7054. PMLR, 2020.

Nina Otter, Mason A. Porter, Ulrike Tillmann, Peter Grindrod, and Heather A. Harrington. A roadmap for the computation of persistent homology. *EPJ Data Science*, 6(1):17, Aug 2017. ISSN 2193-1127.

Chi Seng Pun, Kelin Xia, and Si Xian Lee. Persistent-homology-based machine learning and its applications–a survey. *arXiv preprint arXiv:1811.00252*, 2018.

Steffen Rendle, Walid Krichene, Li Zhang, and John Anderson. Neural collaborative filtering vs. matrix factorization revisited. In *Fourteenth ACM Conference on Recommender Systems*, pp. 240–248, 2020.

Wouter Saelens, Robrecht Cannoodt, Helena Todorov, and Yvan Saeys. A comparison of single-cell trajectory inference methods. *Nature Biotechnology*, 37:1, 04 2019.

Yitzchak Solomon, Alexander Wagner, and Paul Bendich. A fast and robust method for global topological functional optimization. In *International Conference on Artificial Intelligence and Statistics*, pp. 109–117. PMLR, 2021.

The GUDHI Project. *GUDHI User and Reference Manual*. GUDHI Editorial Board, 3.4.1 edition, 2021. URL `https://gudhi.inria.fr/doc/3.4.1/`.

Lou van den Dries. *Tame topology and o-minimal structures*, volume 248. Cambridge university press, 1998.

Robin Vandaele. Topological data analysis of metric graphs for evaluating cell trajectory data representations. Master's thesis, Ghent University, 2020.

Robin Vandaele, Yvan Saeys, and Tijl De Bie. Mining topological structure in graphs through forest representations. *Journal of Machine Learning Research*, 21(215):1–68, 2020.

Robin Vandaele, Bastian Rieck, Yvan Saeys, and Tijl De Bie. Stable topological signatures for metric trees through graph approximations. *Pattern Recognition Letters*, 147:85–92, 2021.

Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *International conference on machine learning*, pp. 478–487. PMLR, 2016.

W.W. Zachary. An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33:452–473, 1977.

Afra Zomorodian and Gunnar Carlsson. Computing Persistent Homology. *Discrete & Computational Geometry*, 33(2):249–274, feb 2005. ISSN 0179-5376.

A INTRODUCTION TO PERSISTENT HOMOLOGY

Persistent homology quantifies the change in topological *holes* (connected components, loops, voids, ...) across a *filtration*, which is an ordered sequence of *simplicial complexes*

$$\mathcal{F} = K_0 \subseteq K_1 \subseteq \ldots \subseteq K_N = K$$

of an initial complex K. A simplicial complex K can be seen as a generalization of a graph, that apart from nodes (0-simplices) and edges (1-simplices), also includes higher-dimensional *simplices* such as triangles (2-simplices), tetrahedra (3-simplices), ..., with the added constraint that if K contains a simplex σ, every simplex $\sigma' \subseteq \sigma$ must also be contained in K. A simplex σ is commonly written as the set of its included *vertices*, $\sigma = \{v_0, \ldots, v_k\}$, and its dimension is by definition k.

An example filtration is shown in Figure 1a in the main paper. Here, the initial complex K is the *Delanauy triangulation* of a point cloud data set $\boldsymbol{E} \subseteq \mathbb{R}^d$ (here $d = 2$). This triangulation, i.e., simplicial complex, is a subdivision of the convex hull of $\boldsymbol{E}$ into simplices such that any two simplices σ, σ' intersect in a common face $\tau \subseteq \sigma \cap \sigma'$ of K, or not at all, and such that the set of vertices of the simplices are contained in $\boldsymbol{E}$, and such that no point in $\boldsymbol{E}$ is inside the circum(hyper)sphere of any d-simplex. Note that this complex is also shown in Figure 1a in the main paper (at time $\alpha = \infty$).

The filtration constructed from K in Figure 1a equals the *α-filtration*. Here, every simplex σ in K is assigned a filtration value α, which equals the square of the circumradius of σ if its circumsphere contains no other vertices than those in σ, in which case σ is said to be *Gabriel*, and as the minimum of the filtration values of the $(|\sigma| + 1)$-simplices containing σ that make it not Gabriel otherwise. At time $t = \alpha$, the complex in the α-filtration includes all simplices with filtration value at most α. Although not required to understand the basic ideas presented in the main paper, for a good overview of how the α-filtration is constructed, we refer the interested reader to The GUDHI Project (2021).

What is most important is that the α-filtration constructed from a point cloud $\boldsymbol{E}$ is well able to capture topological properties of the underlying model of $\boldsymbol{E}$. For example, in Figure 1a, we see that at some time in the filtration, the simplicial complex includes four connected components, one for each of the letter 'I', 'C', 'L', and 'R'. We also see that at some time, the complex captures the cycle in the letter 'R', and later, it captures the larger cycle composed by the letters 'C' and 'L'. These correspond to *topological holes* in the underlying model of $\boldsymbol{E}$. A 0-dimensional hole is a gap between components, a 1-dimensional hole is a cycle or a loop, a 2-dimensional hole is a void, and in general, an n-dimensional hole can be regarded as the inside of an n-sphere. Here, true topological holes, i.e., those of the underlying model, tend to *persist* longer in the α-filtration.

Persistent homology now tracks and quantifies these topological holes, of which the results are commonly visualized by means of a *persistence diagram*. A persistence diagram contains a tuple (b, d) for each topological hole of a fixed dimension that is born at time b and that dies at time d in a filtration. Persistence diagrams for different dimensions of holes in the same data are usually plotted on top of each other, as in Figure 1b in the main paper. Holes that persist longer correspond to more elevated points in the diagram, and capture more prominent topological properties of the underlying model. Tuples (b, d) for which $d = \infty$, which occur when a hole never dies in the filtration (e.g., at some point, the α-filtration will always remain connected), are usually plotted on top of the diagram.

Note that topological optimization—that is, optimizing the data representation with respect to its persistence diagram(s) and one of the main tools for our proposed method of topological regularization—is especially effective when conducted through the α-filtration constructed from a low-dimensional data embedding matrix $\boldsymbol{E} \subseteq \mathbb{R}^d$. In particular when $d = 2$, e.g., for data visualization applications—which was also the focus in the experiments section in the main paper—the α-filtration can be rapidly constructed from $\boldsymbol{E}$, whereas its computational cost increases exponentially for larger dimensions d. A potential solution to this is to use *Vietoris-Rips filtrations* instead. These are filtrations constructed from the *Vietoris-Rips complex* of the data $\boldsymbol{E}$, which includes a simplex for every possible subset of points in $\boldsymbol{E}$, of which the dimensions are constrained by the homology dimension (plus one) of interest in practice. While Vietoris-Rips complexes, and thus, the filtrations thereof, can be constructed more rapidly in higher dimensions, they tend to include far more simplices than the α-filtration, which inherently complicates the subsequent computation of persistent homology. Thus, optimizing the loss for topological regularization (equation (1) in the main paper) is most efficient through α-filtrations for low-dimensional data embedding matrices $\boldsymbol{E}$. For more details on the computational cost of persistent homology as well as the associated filtrations, we refer to Otter et al. (2017).

B SUPPLEMENTARY EXPERIMENTS

We considered an additional experiment on the Harry Potter graph obtained from `https://github.com/hzjken/character-network`. This graph is composed of characters from the Harry Potter novel (the nodes in the graph), and edges marking friendly relationships between them (Figure 9). Only the largest connected component is used. This graph has previously been analyzed by Vandaele et al. (2020), who identified a circular model therein that transitions between the 'good' and 'evil' characters from the novel.

To embed the Harry Potter graph, we used a simple graph embedding model where the sigmoid of the inner product between embedded nodes captures the (Bernoulli) probability of an edge occurrence (Rendle et al., 2020). Thus, this probability will be high for nodes close to each other in the embedding, and low for more distant nodes. These probabilities are then optimized to match the binary edge indicator vector. Figure 10a shows the result of this embedding, along with the circular model presented by Vandaele et al. (2020). For clarity, character labels are only annotated for a subset of the nodes (the same as by Vandaele et al. (2020)).

We furthermore regularized this embedding using a topological loss function $\mathcal{L}_{\text{top}}$ that measures the persistence of the most prominent 1-dimensional hole in the embedding (see also Table 2 in the main paper), the result of which is shown in Figure 10c. Interestingly, the topologically regularized embedding now better captures the circularity of the model identified by Vandaele et al. (2020), and focuses more on distributing the characters along it. Note that although this model is included in the visualizations, it is not used to derive the embeddings, nor is it derived from them.

For comparison, Figure 10b shows the result of optimizing the initialized ordinary graph embedding for the same topological loss, but without the graph embedding loss. We observe that this results in a sparse enlarged cycle. Most characters are positioned poorly along the circular model, and concentrate near a small region. Interestingly, even though we only optimized for the topological loss here, it is actually less optimal, i.e., higher, than when we applied topological regularization (see also Table 3 in the main paper). This is also a result from the sparsity of the circle, which constitutes to a larger birth time, and thus a lower persistence, of the corresponding hole.

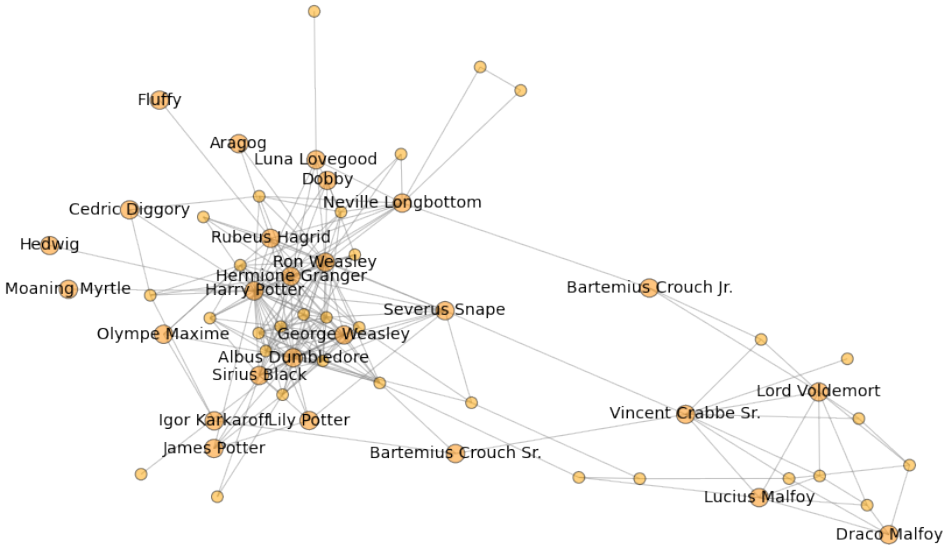
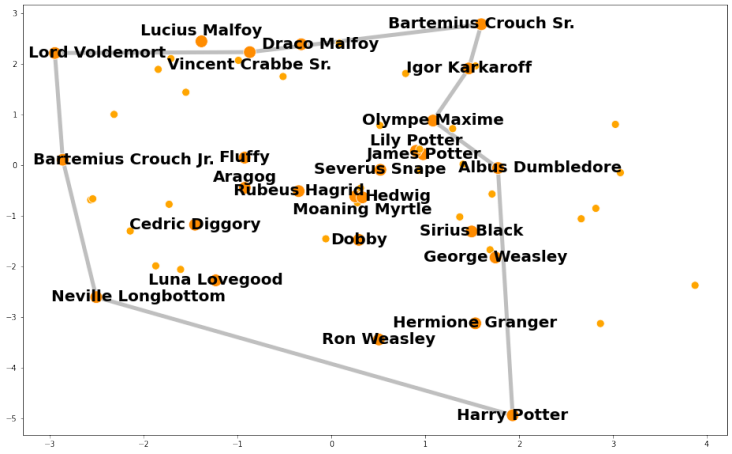
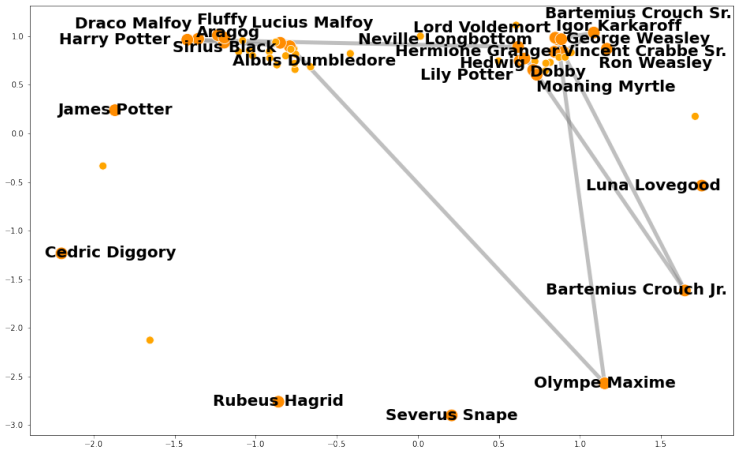


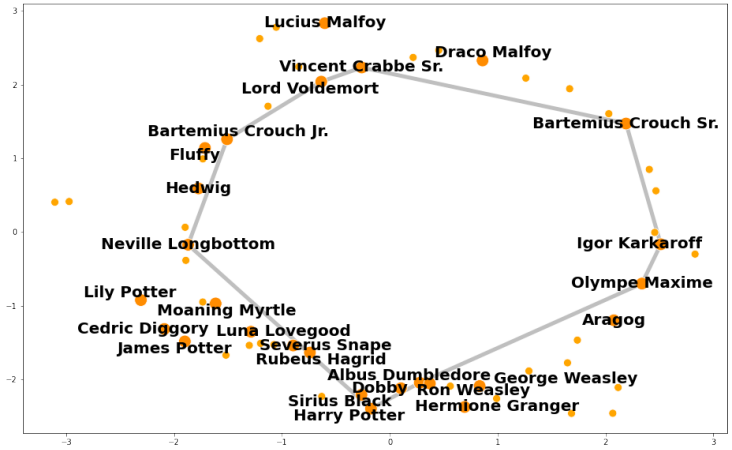
Figure 9: The major connected component in the Harry Potter graph. Edges mark friendly relationships between characters.



(a) Ordinary graph embedding.



(b) Topologically optimized embedding (initialized with the ordinary graph embedding).



(c) Topologically regularized embedding.

Figure 10: Various embeddings of the Harry Potter graph and the circular model therein.