PERSISTENCE-BASED CONTRASTIVE LEARNING WITH GRAPH NEURAL RECURRENT NETWORKS FOR TIME-SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

In the recent years, combinations of graph convolution and recurrent architectures have emerged as a new powerful alternative for multivariate spatio-temporal forecasting, with applications ranging from biosurveillance to traffic monitoring. However, such methods often tend to suffer from vulnerability to noise and limited generalization abilities, especially when semantics and structural properties of time series evolve over time. To address these limitations, we propose a simple yet flexible and highly effective framework, i.e., Persistence-based Contrastive Learning with Graph Neural Recurrent Networks (PCL-GCRN). The key idea behind PCL-GCRN is the notion of topological invariance that we introduce to contrastive graph learning for multivariate spatio-temporal processes. PCL-GCRN allows us to simultaneously focus on multiple most important data shape characteristics at different granularities that play the key role in the learning performance. As a result, PCL-GCRN leads to richer data augmentation, improved performance, and enhanced robustness. Our extensive experiments on a broad range of real-world datasets, from spatio-temporal forecasting of traffic to monkeypox surveillance, suggest that PCL-GCRN yields competitive results both in terms of prediction accuracy and robustness, outperforming 19 competing approaches.

1 INTRODUCTION

Graph neural networks (GNNs) have recently emerged as a new promising approach for multivariate time series forecasting, allowing for simultaneous modelling of complex spatio-temporal interdependencies. However, such GNNs tend to be vulnerable to noisy observations and are often limited in their generalization ability, especially when semantics and structural properties of time series evolve over time. These limitations can be addressed using the concept of contrastive learning.

Contrastive learning aims to obtain informative representations from unlabelled data which are consistent under various augmented views. This task is approached by exploiting feature invariance under certain transformations and maximizing the agreement among the derived representations. The intuition underlying this idea is to construct representations such that, without relying on the labelled data, data samples with similar features are close, while samples with different features are distinguished even prior to performing any task such as classification. This in return not only improves performance of downstream tasks but also enhances generalization abilities and robustness.

Data augmentation is the key prerequisite for contrastive learning and is arguably its most critical element, especially in conjunction with the analysis of graphs which exhibit rich structural information in a diverse set of contexts, from spread of infectious agent on social networks to predictive analytics on blockchain transaction graphs. As shown by You et al. (2020), the impact of data augmentation techniques may vary substantially across types of graph structured data and their underlying properties. For instance, attribute masking is found to be more beneficial for denser graphs, while edge perturbation leads to better performance on social networks and tends to deteriorate learning results on biochemical graphs. In turn, perturbation of subgraphs which may be viewed as structural and functional “motifs”, systematically results in the competitive gains, and such phenomenon sustains across a broad range of datasets. Generally, while the preferred combination of augmentation techniques tends to be data-specific, fusing multiple augmentation strategies which mine different structural and contextual graph properties, appears to be the most promising approach for self-supervised graph learning (Sun et al., 2020; Jiao et al., 2020). In turn, developing methods for automatic selection of data augmentation types and their optimal combination is one of the emerging directions in self-supervised learning (Liu et al., 2022). Finally, while contrastive learning continues to gain popularity for analysis of multivariate time series (Yue et al., 2022; Yang & Hong, 2022; Woo et al., 2021), very little remains known on data augmentation of graph-structured representations of spatio-temporal processes (Opolka et al., 2019; Liu et al., 2021).

Here we make a first step toward fusing these emerging research directions by introducing the concepts of topological invariance and persistent homology (PH) to contrastive learning of spatio-temporal graphs. PH is a subfield of topological data analysis (TDA) which tracks evolution of data shape patterns at various scales, where by shape we broadly mean data properties that are invariant under continuous transformations. Such common shape patterns include, for example, independent components, holes, and voids, which are described via associated simplicial complexes and summaries of thereof. (Note that, for instance, nodes can be viewed as 0-dimensional simplices, edges are 1-dimensional simplices, and so on.) Inspired by the recent proliferation of TDA into machine learning, we propose to leverage these ideas on topological invariance and develop a novel persistence-based data augmentation. The key approach here is to perturb the extracted shape features, targeting the shape patterns that persist over multiple scales and, as such, are likelier to contain the most valuable latent information on the underlying object. The benefits of the new methodology are multifold. First, PH extracts and focuses on the most characteristic shape patterns in an objective manner, thereby allowing us to systematically account for both local and global structural properties at different granularities. As a result, we *automatically* encompass the conventional augmentation types, focusing on perturbation of nodes, edges, and subgraphs. Second, we simultaneously consider shape patterns of various dimensions, hence, resulting in a richer data augmentation. Finally, since the extracted topological features are expected to reflect data properties invariant under continuous transformations, the resulting shape characteristics are intrinsically more robust than many conventional features. As a consequence, the persistence-based data augmentation enhances robustness of the downstream tasks. We apply the proposed methodology to contrastive learning of spatio-temporal graphs, in conjunction with multivariate time series forecasting, and validate its utility on a diverse set of processes, from traffic networks to spread of monkeypox.

Significance of our contributions can be summarized as follows:

- This is the first approach to introduce the concepts of topological invariance not only to contrastive learning of graphs and spatio-temporal processes but to self-supervised learning, in general. By leveraging the machinery of persistent homology, we propose a novel persistence-based contrastive learning (PCL) for spatio-temporal graphs which allows us to systematically extract the most inherent latent data shape characteristics at different granularities that play the key role in the learning performance.
- The new persistence-based data augmentation simultaneously account for both local and global structural properties and automatically integrate shape characteristics of various dimensions, thereby resulting in richer data augmentation.
- Inspired by the notion of landmarks in computer vision and the recent results on PH on landmarks, we bring the landmark ideas to contrastive graph learning, which enables us to further enhance robustness and reduce computational costs.
- To validate PCL-GCRN, we perform extensive experiments on 6 benchmark datasets, in conjunction with spatio-temporal graph contrastive learning and the downstream task of multivariate time series forecasting. Our results indicate that PCL-GCRN outperforms 19 competitors on 6 datasets both in terms of forecasting performance and robustness.

2 Related Works

A natural deep learning solution of time-series datasets modeling consists of Recurrent Neural Networks (RNNs) and its successors such as Long-Short-Term-Memory (LSTM) networks and Gated Recurrent Unit (GRU) networks. GNNs is an effective framework for hidden representation learning of graph structures through message passing on the graph. To capture spatial and temporal dependencies, many studies attempt to use GNNs to incorporate spatial dependencies into the RNN framework, especially in traffic prediction problem on transportation networks (Yu et al., 2018b; Guo et al., 2019; Pan et al., 2019). DCRNN (Li et al., 2018) integrates diffusion graph convolutional layer into a GRU network for long-term traffic forecasting. Additionally, Guo et al. (2019) employs attention mechanism to learn spatial and temporal dependencies, so that the dynamic spatio-temporal correlations can be captured. Furthermore, Wu et al. (2019b) develops a self-adaptive adjacency matrix to perform graph convolutions without pre-defined graph structure. Song et al. (2020) introduces a spatio-temporal synchronous graph convolution operation to capture the localized spatial-temporal correlations by localized spatial-temporal graphs.

Although the aforementioned spatio-temporal graph convolution methods have been proven to be effective in forecasting tasks in many real-world applications, most existing methods ignore the complexity of dynamic graph structural information and require external guidance, for example, labels (which are important but hard to obtain) to learn a promising graph-level representation. Inspired by the promising performance of self-supervised learning (Hjelm et al., 2018; He et al., 2020), some efforts have been dedicated to develop contrastive learning with graph augmentations (Velickovic et al., 2019; You et al., 2020; Qiu et al., 2020; Liu et al., 2022), to tackle spatio-temporal graph representations (Oord et al., 2018; Eldele et al., 2021). For example, Yue et al. (2022) designs hierarchical contrasting method in both instance-wise and temporal dimensions to capture contextual information in time-series data. Similarly, Woo et al. (2021) applies contrastive learning to learn disentangled seasonal-trend representations through a causal lens. Also, Eldele et al. (2021) proposes a contrastive learning with two different correlated views to obtain robust representation. In this paper, we employ persistent homology to capture hidden shape information on the underlying graph structure and design an efficient persistence-based data augmentation. Additionally, we propose a novel persistence-based contrastive learning to improve spatio-temporal graph representations of multivariate time series.

3 METHODOLOGY

Spatio-temporal Data as Graph Structures Spatio-temporal networks have recently proven to be a promising abstraction to describe complex dependence properties in multivariate time series. The spatial-temporal networks can be represented as a sequence of discrete snapshots, $\{\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_\mathcal{T}\}$, where $\mathcal{G}_t = \{\mathcal{V}_t, \mathcal{E}_t, A_t\}$ is the graph structure at time step t, $t = 1, 2, \ldots, \mathcal{T}$. In $\mathcal{G}_t$, $\mathcal{V}_t$ is a node set containing N nodes, i.e., $\mathcal{V}_t = \{v_i\}_i^N$ and $\mathcal{E}_t \subseteq \mathcal{V}_t \times \mathcal{V}_t$ is an edge set. For a weighted graph, each edge is assigned a corresponding weight by function $\omega : \mathcal{E}_t \mapsto \mathbb{R}_+$. Then the weighted adjacency matrix A_t is defined as $a^t_{uv} = \omega(u, v)$. Let F be the number of different node features associated each node $v \in \mathcal{V}_t$ (where $F \geq 1$); then, a $N \times F$ feature matrix X_t serves as the input to the backbone model for time-series forecasting. The problem of time-series forecasting can be described as: given τ historical observations $\mathcal{X}^\tau = \{X_{t-\tau}, X_{t-\tau+1}, \ldots, X_{t-1}\}$, we aim to find a multivariate forecasting model $\mathfrak{M}(\cdot)$ to predict future observations in the next ζ timestamps, i.e., $\{X_t, X_{t+1}, \ldots, X_{t+\zeta-1}\} = \mathfrak{M}(X_{t-\tau}, X_{t-\tau+1}, \ldots, X_{t-1})$.

3.1 PRELIMINARIES ON PERSISTENT HOMOLOGY

Persistent homology is a branch in topological data analysis which tracks evolution of the various data shape patterns along various user-selected geometric dimensions (Edelsbrunner et al., 2000; Zomorodian & Carlsson, 2005). By utilizing a multi-scale approach to shape description, PH addresses the intrinsic limitations of classical homology and allows for retrieval of shape patterns that tend to persist over multiple scales and, hence, are likelier to play the important role for a given downstream task. The main idea is to select some suitable scale parameters α and then to assess changes in shape (or more formally homology) that occur to $\mathcal{G}$ which evolves with respect to α. That is, we no longer study $\mathcal{G}$ as a single object but consider a *filtration* $\mathcal{G}_{\alpha_1} \subseteq \ldots \subseteq \mathcal{G}_{\alpha_n} = \mathcal{G}$, induced by monotonic changes of α. To make the process of pattern counting more systematic and efficient, we build an abstract simplicial complex $\mathscr{K}(\mathcal{G}_{\alpha_j})$ on each $\mathcal{G}_{\alpha_j}$, resulting in a filtration of complexes $\mathscr{K}(\mathcal{G}_{\alpha_1}) \subseteq \ldots \subseteq \mathscr{K}(\mathcal{G}_{\alpha_n})$. Due to its computational benefits, one of the most widely used choices is a Vietoris-Rips ($\mathcal{VR}$) complex. Turing to the choice of the scale parameter, if we select a scale parameter as a shortest (weighted) path between any two nodes, then the abstract simplicial complex $\mathscr{K}(\mathcal{G}_{\alpha_*})$ is generated by subgraphs $\mathcal{G}^{'}$ of bounded diameter α_*. In turn, if $\mathcal{G}$ is an edge-weighted graph $(\mathcal{V}, \mathcal{E}, w)$, with the edge-weight function $w : \mathcal{E} \mapsto \mathbb{R}$, then for each α_j we can consider only induced subgraphs of $\mathcal{G}$ with maximal degree of α_j, leading to a degree sublevel set filtration. (For the detailed discussion on graph filtrations see Hofer et al. (2020).)

Armed with this construction, we can now monitor data shape patterns such as independent components, holes, and cavities which (dis)appear as scale α changes (i.e., for each topological feature ρ we record the indices b_ρ and d_ρ of $\mathscr{K}(\mathcal{G}_{b_\rho})$ and $\mathscr{K}(\mathcal{G}_{d_\rho})$, where ρ is first and last observed, respectively). We say that a pair (b_ρ, d_ρ) represents the birth and death times of ρ, and $(d_\rho - b_\rho)$ is its corresponding lifespan (or persistence). Intuitively, shape features with longer lifespans are considered more valuable, while features with shorter lifespans are often associated with topological noise. The extracted topological information over the filtration $\{\mathscr{K}_{\alpha_j}\}$ can be summarized in a form of a multiset in $\mathbb{R}^2$ called *persistence diagram (PD)* $\mathcal{D} = \{(b_\rho, d_\rho) \in \mathbb{R}^2 : d_\rho > b_\rho\} \cup \Delta$ (here $\Delta = \{(t,t) | t \in \mathbb{R}\}$ is the diagonal set containing points counted with infinite multiplicity; including Δ allows us to compare different PDs based on the cost of the optimal matching between their points).

Finally, inspired by the recent results on PH on witness complexes and landmarks on graphs (Arafat et al., 2019), we construct $\mathscr{K}(\mathcal{G}_{\alpha_*})$ on a set of *landmark* nodes. Here we select landmarks based on the node degree centrality. This approach allows us both to reduce computational costs and to focus only on the most intrinsic shape properties, thereby reducing the impact of topological noise.

3.2 TWO-STREAM SPATIAL GRAPH CONVOLUTION

To systematically incorporate both global topological information and node features from the spatial dimension, in this section, we introduce the components of our proposed two-stream spatial graph convolution as follows.

$\mathfrak{K}$-[multi-hop] Chebyshev convolution The multi-hop Chebyshev convolution aims to learn higher-order structural information encoded in the graph topology by taking multi-hop neighborhood of every node as guidance. Here, we adopt a random walk-based Chebyshev convolution to implement this module.

Given a graph Laplacian L, the multi-hop Chebyshev polynomial with the neighborhood radius $\gamma = K$, i.e., $T_r(\tilde{L}^K)$ is recursively defined as

$$T_r(\tilde{L}^K) = 2\tilde{L}^K T_{r-1}(\tilde{L}^K) - T_{r-2}(\tilde{L}^K), \tag{1}$$

where $T_0 = 1$, $T_1 = \tilde{L}^K$, $\tilde{L}^K = (2L/\lambda_{max} - I_N)^K$ represents the K-hop renormalized graph Laplacian (which is normalized to $[-1, 1]$, λ_{max} is the maximum eigenvalue of the graph Laplacian, and I_N is an identity matrix), and $K \geq 1$ denotes the power of the renormalized graph Laplacian. With the power order $K \geq 1$, for each node u, we can extend the message passing process from a larger neighborhood when applying the graph convolution operation. However, it is impossible to manually tune the optimal neighborhood size at each timestamp. Our $\mathfrak{K}$-[multi-hop] Chebyshev convolution technique is then motivated by the following question: *can we design an operator to comprehensively collect spatial and spectral information across a wider range?* We answer this affirmative by concatenating $\mathfrak{K}$ weighted multi-hop Chebyshev polynomial graph filter tensor to make multi-level global topological information available to graph convolution operation, and empowering graph convolution to capture neighbors' information from time-dependent graph representations.

Definition 3.1 (Weighted Multi-hop Chebyshev Polynomial Graph Filter Tensor (WMCheby-GFT))**.** Given the spatial network $\mathcal{G}_t$ at timestamp t, let $\tilde{L}^K$ denote K-hop renormalized graph Laplacian ($K \in \{1, 2, \ldots, \mathfrak{K}\}$). The weighted multi-hop Chebyshev polynomial graph filter tensor is defined as

$$\vec{\tilde{L}} = [I_N, \alpha_1 \cdot T_r(\tilde{L}), \alpha_2 \cdot T_r(\tilde{L}^2), \ldots, \alpha_K \cdot T_r(\tilde{L}^K), \ldots, \alpha_{\mathfrak{K}} \cdot T_r(\tilde{L}^{\mathfrak{K}})] \in \mathbb{R}^{(\mathfrak{K}+1) \times N \times N}, \tag{2}$$

where the attentional coefficient $\alpha_K^u = \text{Softmax}(\theta_K^u) = \exp(\theta_K^u) / \sum_{K=1}^{\mathfrak{K}} \exp(\theta_K^u)$ indicates the importance of node u in $T_r(\tilde{L}^K)$, $\theta_K^u = \text{MLP}(\sigma(\text{MLP}(T_r(\tilde{L}^K))))$ (here MLP denotes the multilayer perceptron), and $\sigma(\cdot)$ is a non-linear function, i.e., the $\tanh$ function.

WMCheby-GFT allows us to adaptively capture the global topological information and partial similarities between neighborhoods with various radii $\gamma \in \{1, 2, \ldots, \mathfrak{K}\}$. Finally, the $\mathfrak{K}$-[multi-hop] Chebyshev convolution can be formulated as

$$Z_C^{(\ell)} = (\vec{\tilde{L}} Z_C^{(\ell-1)})^\top \Theta_C, \tag{3}$$

where $(\cdot)^\top$ denotes transpose, $\Theta_C \in \mathbb{R}^{(\mathfrak{K}\times d^C_{in})\times d^C_{out}}$ is the trainable weight matrix to perform feature transformation in each layer (where d^C_{in} and d^C_{out} are the input and output dimensions of $(\ell-1)$-th layer respectively), and $Z^{(\ell-1)}_C \in \mathbb{R}^{N\times d^C_{in}}$ and $Z^{(\ell)}_C \in \mathbb{R}^{N\times d^C_{out}}$ are the input and output of the $(\ell-1)$-th layer, respectively. (Note that, $Z^{(0)}_C = X_t \in \mathbb{R}^{N\times F}$ is the node features of graph at timestamp t; for simplicity, we omit the subscript t for notations in Eq. 3.)

$\mathfrak{Q}$-[multi-hop] adaptive spatial graph convolution However, the $\mathfrak{K}$-[multi-hop] Chebyshev convolution based on the pre-defined graph structure above may suffer from two limitations. First, the topological relationship among nodes may change dynamically over time for time-evolving graphs. Second, using the binary correlation matrix based on the pre-defined graph structure may limit generalization ability. To overcome these limitations, inspired by the recent success of adaptive dependency matrix (Wu et al., 2019a; Bai et al., 2020), we propose to use the node embedding dictionary to design a $\mathfrak{Q}$-[self-adaptive] spatial graph convolution. Let $E_\phi = (e_{1,\phi}, e_{2,\phi}, \dots, e_{N,\phi}) \in \mathbb{R}^{N\times d_E}$, which is a trainable node embedding dictionary for all N nodes (d_E is the dimension of node embedding). Then, computed by the inner product between E_ϕ and $E^\top_\phi$, i.e., $\ddot{L} = \text{Softmax}(\sigma(< E_\phi, E^\top_\phi >))$, we can obtain the self-adaptive adjacency matrix (where the non-linear function $\sigma(\cdot)$ is empirically set to ReLU function). Similar to WMCheby-GFT, in order to exploit the global topological information of the spatio-temporal graphs and to further improve the learning performance, we propose to employ a weighted multi-hop self-adaptive adjacency matrix tensor for aggregation of information from different neighborhood radii.

Definition 3.2 (Weighted Multi-hop Self-adaptive Adjacency Matrix Tensor (WMS-AMT))**.** Given the spatial network $\mathcal{G}_t$ at timestamp t, let $\ddot{L}^K$ denote K-hop self-adaptive adjacency matrix, where $K \in \{1, 2, \dots, \mathfrak{Q}\}$. The weighted multi-hop self-adaptive adjacency matrix tensor is then defined as

$$\vec{\ddot{L}} = [I_N, \beta_1 \cdot \ddot{L}, \beta_2 \cdot \ddot{L}^2, \dots, \beta_K \cdot \ddot{L}^K, \dots, \beta_{\mathfrak{Q}} \cdot \ddot{L}^{\mathfrak{Q}}] \in \mathbb{R}^{(\mathfrak{Q}+1)\times N\times N}, \tag{4}$$

where the attentional coefficient $\beta^u_K = \text{Softmax}(\theta^u_K) = \exp(\theta^u_K)/\sum^{\mathfrak{Q}}_{K=1}\exp(\theta^u_K)$ indicates the importance of node u in $\ddot{L}^K$, $\theta^u_K = \text{MLP}(\sigma(\text{MLP}(\ddot{L}^K)))$.

Armed with WMS-AMT as designed above, we define a graph convolution working on the adaptive graph structure representation as $Z^{(\ell)}_A = (\vec{\ddot{L}} Z^{(\ell-1)}_A)^\top E_\phi \Theta_A$. Here $\Theta_A \in \mathbb{R}^{N\times(\mathfrak{Q}\times d^A_{in})\times d^A_{out}}$ denotes learnable parameters (where d^A_{in} and d^A_{out} are the input and output dimensions of $(\ell-1)$-th layer respectively), and $Z^{(\ell-1)}_A \in \mathbb{R}^{N\times d^A_{in}}$ and $Z^{(\ell)}_A \in \mathbb{R}^{N\times d^A_{out}}$ are the input and output of the $(\ell-1)$-th $\mathfrak{Q}$-[multi-hop] adaptive spatial graph convolutional layer, respectively. (Here $Z^{(0)}_A = X_t \in \mathbb{R}^{N\times F}$). As such, by setting the attention mechanism for self-adaptive adjacency matrix tensor, we allow to use a large K for long-range modeling with controllable oversmoothing. Lastly, we get the final embedding matrix used for spatial information modeling as

$$Z^{(\ell)} = f_C \cdot Z^{(\ell)}_C + f_A \cdot Z^{(\ell)}_A, \tag{5}$$

where f_C and f_A are importance weights for outputs of $\mathfrak{K}$-[multi-hop] Chebyshev convolution and $\mathfrak{Q}$-[multi-hop] adaptive spatial graph convolution, i.e., $Z^{(\ell)}_C$ and $Z^{(\ell)}_A$ respectively. It is worth to note that, in our experiments, the importance weight $f_\square$ (i.e., $\square \in \{C, A\}$) can be considered as either attention mechanism (which can perform adaptive aggregation based on a multi-head aggregation module) or weighting factor (which can be set either as a hyperparameter or a fixed scalar). To capture both spatial and temporal correlations in time-series, we feed the final embedding $Z^{(\ell)}$ into Gated Recurrent Units (GRU) for future time points forecasting (see more details in Appendix A). For each timestamp t, the loss function of GRU with two-stream spatial graph convolution is formulated as $\mathcal{L}_0 = ||\hat{X}_t - X_t||^2_2$, where X_t is the ground truth value at timestamp t to forecast, and $\hat{X}_t$ is the predicted value for the timestamp t.

3.3 PERSISTENCE-BASED CONTRASTIVE LEARNING

Before diving into our proposed persistence-based contrastive learning, we first introduce graph data augmentations (You et al., 2020), and then we present how to develop persistence-based data augmentation and corresponding persistence-based contrastive learning (i.e., contrastive learning with topological signatures of PH) with application to spatio-temporal forecasting. We further perform ablation studies (see Table 7) on both **topological** and **graph** contrastive learning to judge the usefulness and effectiveness of each contrastive learning strategy in spatio-temporal forecasting tasks. Given a graph $\mathcal{G}_t$ $(t = 1, 2, \ldots, \mathcal{T})$ with its adjacency matrix A_t and node feature matrix X_t, in practice, there are three different ways to perturb the $\mathcal{G}_t$, i.e., (i) DropNode, (ii) edge perturbation, and (iii) node feature shuffling. Specifically, (i) in DropNode: DropNode technique randomly drops out a certain rate of nodes (also removes all edges connected to the dropping nodes; we denote the resulting perturbed adjacency matrix from DropNode as $\dddot{A}_t^{\mathcal{V}}$), (ii) in edge perturbation: it adds (+) or drops out (-) a certain rate of edges of the input graph by random (we denote the resulting perturbed adjacency matrices from edge perturbations as $\dddot{A}_t^{\mathcal{E}-}$ and $\dddot{A}_t^{\mathcal{E}+}$ respectively), and (iii) in node feature shuffling, the perturbed node feature $\dddot{X}_t$ is obtained by the row-wise shuffling of X_t. With such design of graph augmentation strategy, we have positive sample (X, A) and negative sample $(\dddot{X}, \dddot{A})$ (*spoiler alert: we do not use negative sample!*) (for simplicity, we omit the timestamp t and the layer ℓ for convenience), and we will now summarize the procedure of topological contrastive learning which can be decomposed into seven steps

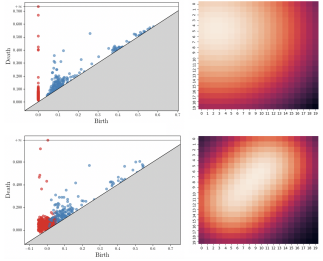

Figure 1: The visualization of persistence-based data augmentation (see Appendix B for more details.)

S1 Obtain node embedding Z by feeding positive sample (X, A) into backbone model (e.g., our proposed two-stream spatial graph convolution; see Section 3.2).

S2 Reconstruct an adjacency matrix $\hat{A}$ via node embedding Z (derived from the two-stream spatial graph convolution in Eq. 5), i.e., $\hat{A} = \sigma(ZZ^{\top})$. Note that, since $\hat{A}$ is reconstructed based on Z (i.e., fusing representations of adaptive learned graph structure (Z_A) and pre-defined graph structure (Z_C)), and hence is close to ground truth.

S3 Employ persistent homology on the reconstructed graph structure (based on $\hat{A}$) and extract corresponding topological signature, e.g., persistence diagram $\mathcal{D}$, persistence image $\mathcal{PI}$ (see Section 3.1). In practice, we choose a set of landmark points (nodes) from the graph (e.g., $\hat{A}[\textit{indices}, \textit{indices}]$ where *indices* denote the indices of landmark points in the graph) and then build the abstract simplicial complex on this set.

S4 Perturb topological signature of PH (e.g., persistence diagram $\mathcal{D} = \{(b_\rho, d_\rho) \in \mathbb{R}^2 | b_\rho < d_\rho\}$) by adding Gaussian noise, i.e., $\dddot{\mathcal{D}} = \{(b'_\rho, d'_\rho) \in \mathbb{R}^2\}$ where $b'_\rho = b_\rho + \eta_b$ and $d'_\rho = d_\rho + \eta_d$ and $\eta_b, \eta_d \sim \mathcal{U}(a_U, b_U)$ (where a_U and b_U are hyperparameters in the uniform distribution). As illustrated in Figure 3, substantial changes are seen for both persistence diagram and persistence image after performing persistence-based data augmentation.

S5 Positive and negative topological signatures $\mathcal{D}$ and $\dddot{\mathcal{D}}$ are fed into a MLP respectively, and we obtain latent topological representations $H = \text{MLP}(\mathcal{D}) \in \mathbb{R}^{N \times d^T_{out}}$ and $\dddot{H} = \text{MLP}(\dddot{\mathcal{D}}) \in \mathbb{R}^{N \times d^T_{out}}$.

S6 The topological representations from positive and negative topological signatures are summarized through READOUT function $\mathcal{R}(\cdot)$, i.e., from obtained local patches to the global content $s = \mathcal{R}(H) = \frac{1}{N}(\sum_{u \in \mathcal{V}} h_u) \in \mathbb{R}^{d^T_{out}}$ and $\dddot{s} = \mathcal{R}(\dddot{H}) = \frac{1}{N}(\sum_{v \in \mathcal{V}} h_v) \in \mathbb{R}^{d^T_{out}}$.

S7 Use a standard binary cross-entropy (BCE) loss as contrastive loss (i) $\mathcal{L}_1$: between positive pairs and negative ones *and* (ii) $\mathcal{L}_2$: between negative pairs and positive ones, where

$$\begin{aligned} \mathcal{L}_1 &= -\frac{1}{2N}\left(\sum_{u \in \mathcal{V}} \mathbb{E}_{\mathcal{D}}[\log(\Xi(h_u, s))] + \sum_{v \in \mathcal{V}} \mathbb{E}_{\dddot{\mathcal{D}}}[\log(1 - \Xi(\dddot{h}_v, s))]\right), \\ \mathcal{L}_2 &= -\frac{1}{2N}\left(\sum_{v \in \mathcal{V}} \mathbb{E}_{\dddot{\mathcal{D}}}[\log(\Xi(\dddot{h}_v, \dddot{s}))] + \sum_{u \in \mathcal{V}} \mathbb{E}_{\mathcal{D}}[\log(1 - \Xi(h_u, \dddot{s}))]\right), \end{aligned} \tag{6}$$

where $\Xi(\cdot, \cdot)$ is a contrastive discriminator, which is defined as $\Xi(h_u, s) = \text{Sigmoid}(h_u^{\top} \Theta_{\Xi} s)$ (where Θ_{Ξ} denotes the learnable weight matrix).

The key benefits of topological contrastive learning are twofold: (i) **hidden information**: compared to traditional graph contrastive learning, conducting contrastive learning on topological representation instead of graph itself allows for learning and incorporating hidden shape characteristics - *higher-order structural information* - that cannot be accurately captured by GNNs (since GNNs only collate information over neighborhoods of each node) and (ii) **Less computational costs**: to obtain augmented representation $\ddot{H}_{Graph}$ of graph contrastive learning, we need to feed negative sample $(\ddot{X}, \ddot{A})$ to the backbone model (typically, GNN-based encoder is considered), i.e., $\ddot{H}_{Graph}$ = GNN($\ddot{X}, \ddot{A}$); however, graph-level representation can be conveniently obtained by applying MLP on augmented topological signature, i.e., $\ddot{H}_{Topo}$ = MLP($\ddot{\mathcal{D}}$); for instance, on monkeypox dataset, under PCL-GCRN model, the total number of parameters of topological contrastive learning is 4966201 and the total number of parameters of graph contrastive learning is 6231161. Eventually, we present the final objective function $\mathcal{L}$ for the spatio-temporal forecasting task, which can be written as follows

$$\mathcal{L} = \pi_0 \times \mathcal{L}_0 + \pi_1 \times \mathcal{L}_1 + \pi_2 \times \mathcal{L}_2, \tag{7}$$

where π_0, π_1, π_2 are hyperparameters which balance the contributions of multi-step prediction task and different contrastive tasks. The overall architecture of the PCL-GCRN model is shown in Figure 2.

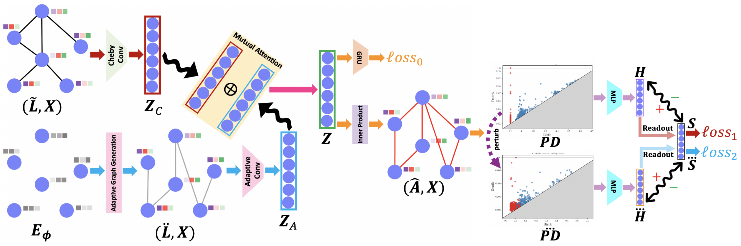

Figure 2: The overall architecture of PCL-GCRN.

4 Experiments

Datasets We conduct experiments on 6 benchmark spatio-temporal graphs, i.e., (i) two real-world traffic flow datasets: PeMSD3 and PeMSD4, (ii) the spread of coronavirus disease COVID-19 at county-level in states of California (CA) and Pennsylvania (PA), (iii) the spread of monkeypox in the United States, and (iv) the COVID-19 dataset records the daily global confirmed cases of states/provinces in United States, Australia, Canada, and China. More details and the detailed statistics of datasets are described in the Appendix C.1 (Table 6).

Baselines We compare our PCL-GCRN model with 19 state-of-the-art (SOAs) baselines: HA, VAR (Hamilton, 2020), FC-LSTM (Sutskever et al., 2014b), GRU-ED (Cho et al., 2014), TCN (Bai et al., 2018), DCRNN (Li et al., 2018), STGCN (Yu et al., 2018b), GraphWaveNet (Wu et al., 2019b), ASTGCN (Guo et al., 2019), MSTGCN (Guo et al., 2019), STSGCN (Song et al., 2020), AGCRN (Bai et al., 2020), StemGNN Cao et al. (2020), LSGCN (Huang et al., 2020), STFGNN (Li & Zhu, 2021), TCVAE He et al. (2022), Z-GCNETs (Chen et al., 2021), STGODE (Fang et al., 2021), and TS2Vec Yue et al. (2022). For more details, please refer to Appendix C.2.

Experiment settings In our experiments, (i) for PeMSD3 and PeMSD4, we use traffic flow data from the past 1 hour to predict the flow for the next hour with batch size as 64 (i.e., we consider the window size $\tau = 12$ and horizon $\zeta = 12$); (ii) for COVID-19 biosurveillance and monkeypox datasets, we set the window size τ as 5, set the horizon ζ as 15, and set the batch size as 8. Note that, for fair comparison, we split the data into training set, validation set, and test set in the same way as the baselines, i.e., $6 : 2 : 2$ on PeMSD3, PeMSD4, and COVID-19 datasets, and we only consider traffic flow as node feature (i.e., $F = 1$; without involving traffic speed and occupancy rate). For CA, PA, and monkeypox datasets, we split into training and test sets with the split ratio $8 : 2$. We evaluate the performances by the mean absolute error (MAE), root mean squared error (RMSE), mean absolute percentage error (MAPE (%)). For more details, please refer to Appendix C.3.

4.1 Experimental Results

Table 1 summarizes all results of SOAs and our proposed PCL-GCRN on PeMSD3 and PeMSD8 datasets. We reuse the metrics of the baselines already reported in the corresponding papers. The experimental results show that PCL-GCRN achieves better performance across both datasets. Specifically, for both PeMSD3 and PeMSD4 datasets, (i) the improvement gain of PCL-GCRN over the runner-ups range from 0.92% to 1.12% in MAE, (ii) the improvement gain of PCL-GCRN over the runner-ups range from 0.32% to 2.26% in RMSE, and (iii) the improvement gain of PCL-GCRN over the runner-ups range from 0.55% to 2.63% in MAPE. Table 2 demonstrates COVID-19 hospitalization prediction results on CA and PA, and confirmed monkeypox cases prediction result in the United States. We find that PCL-GCRN achieves state-of-the-art performance on all three datasets. Specifically, PCL-GCRN yields 5.55%, 8.87%, and 16.56% relative gains in RMSE over the runner-ups (i.e., Z-GCNETs and STGODE). The results of COVID-19 confirmed cases prediction are shown in Table 3. As expected, we see that our PCL-GCRN model outperforms all baselines on MAPE. Specifically, PCL-GCRN, i.e., our model equipped with topological contrastive learning can improve upon TS2VEC (i.e., model based on graph contrastive learning) by a margin of 238.75%.

Table 1: Forecasting performance on PeMSD3 and PeMSD4 datasets.

Model	**PeMSD3**			**PeMSD4**		
	MAE	RMSE	MAPE (%)	MAE	RMSE	MAPE (%)
HA	31.58	52.39	33.78	38.03	59.24	27.88
VAR Hamilton (2020)	23.65	38.26	24.51	24.54	38.61	17.24
FC-LSTM Sutskever et al. (2014b)	21.33	35.11	23.33	26.77	40.65	18.23
GRU-ED Cho et al. (2014)	19.12	32.85	19.31	23.68	39.27	16.44
TCN Bai et al. (2018)	18.87	32.24	18.63	26.31	39.59	17.20
DCRNN Li et al. (2018)	17.99	30.31	18.34	21.20	37.23	14.15
STGCN Yu et al. (2018a)	17.55	30.42	17.34	21.16	35.69	13.83
GraphWaveNet Wu et al. (2019a)	19.12	32.77	18.89	28.15	39.88	18.52
ASTGCN Guo et al. (2019)	17.34	29.56	17.21	22.81	34.33	16.60
MSTGCN Guo et al. (2019)	19.54	31.93	23.86	23.96	37.21	14.33
STSGCN Song et al. (2020)	17.48	29.21	16.78	21.23	33.69	13.90
AGCRN Bai et al. (2020)	16.10	28.18	15.23	19.83	32.30	12.97
LSGCN Huang et al. (2020)	17.94	29.85	16.98	21.53	33.86	13.18
Z-GCNETs Chen et al. (2021)	16.64	28.15	16.39	19.50	31.61	12.78
STGODE Fang et al. (2021)	16.50	27.84	16.69	20.84	32.82	13.77
STFGNN Li & Zhu (2021)	16.77	28.34	16.30	19.83	31.88	13.02
PCL-GCRN (ours)	**15.93**	**27.21**	**14.83**	**19.32**	**31.51**	**12.71**

Table 2: Forecasting performance on COVID-19 hospitalizations in CA, PA, and monkeypox in USA.

Model	**CA**			**PA**			**Monkeypox**		
	MAE	RMSE	MAPE (%)	MAE	RMSE	MAPE (%)	MAE	RMSE	MAPE (%)
FC-LSTM (Sutskever et al., 2014a)	167.86	502.29	90.71	47.60	108.74	69.37	320.89	835.51	92.23
DCRNN (Li et al., 2018)	107.20	492.10	69.83	47.49	107.21	67.15	390.12	899.10	100.37
STGCN (Yu et al., 2018b)	102.88	470.52	69.73	52.69	106.78	69.36	390.59	880.59	81.87
TCN (Bai et al., 2018)	110.82	492.82	70.00	49.80	105.07	69.86	323.43	846.02	76.25
AGCRN (Bai et al., 2020)	87.24	448.27	66.30	44.69	103.79	63.45	283.39	787.46	32.44
Z-GCNETs (Chen et al., 2021)	81.22	356.35	62.81	43.52	106.22	65.89	247.67	743.33	35.33
STGODE Fang et al. (2021)	82.93	368.07	45.19	44.75	100.88	67.23	218.12	749.55	32.20
PCL-GCRN (ours)	**72.78**	**336.59**	**44.87**	**35.50**	**91.93**	**63.25**	**183.62**	**620.25**	**22.41**

Table 3: Forecasting performance on COVID-19 confirmed cases in the whole world (MAPE (%)).

Dataset	**StemGNN**	**AGCRN**	**TCVAE**	**TS2VEC**	**PCL-GCRN (ours)**
COVID-19	335.37	180.96	198.12	341.08	**102.33**

4.2 Ablation Studies

Different components in PCL-GCRN We conduct an ablation study to examine the contributions of different components in PCL-GCRN and results are presented in Table 4. We compare our PCL-GCRN model with three ablated variants, i.e., (i) PCL-GCRN without $\mathfrak{K}$-[multi-hop] Chebyshev convolution (i.e., W/o $\mathfrak{K}$-[multi-hop] Chebyshev convolution), (ii) PCL-GCRN without $\mathfrak{Q}$-[multi-hop] adaptive spatial graph convolution (i.e., W/o $\mathfrak{Q}$-[multi-hop] adaptive spatial graph convolution), and (iii) PCL-GCRN without topological contrastive learning (i.e., W/o Topological contrastive learning). From Table 4, we can observe that our PCL-GCRN consistently achieves large-margin outperformance over all variants on both CA and monkeypox datasets, suggesting that all three designed components contribute to the success of PCL-GCRN. Moreover, we find that $\mathfrak{Q}$-[multi-hop] adaptive spatial graph convolution always improve the performance, i.e., the reason is that $\mathfrak{Q}$-[multi-hop] adaptive spatial graph convolution can capture the hidden spatial information through time in spatio-temporal datasets.

Table 4: Ablation study of the PCL-GCRN architecture.

Dataset	Architecture	MAE	RMSE	MAPE (%)
CA	**PCL-GCRN**	**72.78**	**336.59**	**44.87**
	W/o $\mathfrak{K}$-[multi-hop] Chebyshev convolution	72.83	358.10	45.52
	W/o $\mathfrak{Q}$-[multi-hop] adaptive spatial graph convolution	82.19	395.08	53.29
	W/o Topological contrastive learning	72.87	354.74	47.90
Monkeypox	**PCL-GCRN**	**183.62**	**620.25**	**22.41**
	W/o $\mathfrak{K}$-[multi-hop] Chebyshev convolution	186.54	627.38	25.75
	W/o $\mathfrak{Q}$-[multi-hop] adaptive spatial graph convolution	380.06	793.02	69.50
	W/o Topological contrastive learning	188.01	620.98	35.72

Persistence-based data augmentation vs. graph augmentation To demonstrate the effectiveness of our proposed persistence-based data augmentation strategy, we conduct experiments of PCL-GCRN on PeMSD4, CA, and monkeypox datasets that compare it to its ablated variant, i.e., GCL-GCRN (i.e., topological contrastive learning is replaced by graph contrastive learning). Furthermore, for GCL-GCRN, we consider 4 types of graph augmentation, i.e., node dropping, attribute perturbation (i.e., node feature shuffling), edge perturbation (+) by adding random edges, and edge perturbation (-) by removing existing edges. In our experiments, we set the node/edge dropping rate and edge noise rate to 10%. Results are summarized in Table 7 (see Appendix D). Clearly, our PCL-GCRN model convincingly outperforms GCL-GCRN with different graph augmentation strategies across all three datasets. Specifically, persistence-based data augmentation can improve upon attribute perturbation (in terms of RMSE) by a margin of 0.92, 3.10, and 13.25 on PeMSD4, CA, and monkeypox respectively. From above ablation study, it is evident that the performance gain is due to our proposed persistence-based data augmentation.

4.3 Robustness Analysis

To assess robustness of PCL-GCRN, we consider adding two types of random noise to the spatio-temporal dataset, i.e., Gaussian noise $\text{Norm}(0, \delta^2)$ (where $\delta = 1$) and Poisson noise $\text{Poisson}(\lambda)$ (where $\lambda = 1$). For each random noise, we add 50% noises to the training set. As Table 5 suggests, compared with the most recent baseline (i.e., AGCRN) and the variant model (i.e., GCL-GCRN which is based on graph contrastive learning), our PCL-GCRN achieves superior performances across two types of random noise. Specifically, PCL-GCRN can improve upon the runner-up by a margin of 3.98 and 2.28 on monkeypox datasets with Gaussian noise and Poisson noise respectively. As a result, we can conclude that PCL-GCRN is noticeably more robust to different types of noise than its competitors and hence may be viewed as a preferred choice for real-world applications under the scenarios of limited, incomplete, or corrupted data records.

Table 5: Robustness study on monkeypox dataset (MAE).

Dataset	Noise	GCL-GCRN	AGCRN	PCL-GCRN (ours)
Monkeypox	Norm(0,1)	205.34	249.03	**201.36**
	Poisson(1)	208.69	230.65	**206.31**

5 Conclusion

By capitalising on the concepts of persistent homology and the associated notion of topological invariance, we have introduced a novel persistence-based data augmentation approach for contrastive learning of spatio-temporal graphs. The new augmentation approach simultaneously accounts for multiple structural characteristics of the observed data at both local and global levels, resulting in competitive performance gains and substantially higher robustness. In the future we will explore the utility of topological metrics for assessing semantic similarity of graph-structured data.

REFERENCES

Naheed Anjum Arafat, Debabrota Basu, and Stéphane Bressan. Topological data analysis with ϵ-net induced lazy witness complex. In *International Conference on Database and Expert Systems Applications*, pp. 376–392, 2019.

Lei Bai, Lina Yao, Can Li, Xianzhi Wang, and Can Wang. Adaptive graph convolutional recurrent network for traffic forecasting. *Advances in Neural Information Processing Systems*, 33, 2020.

Shaojie Bai, J Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018.

Defu Cao, Yujing Wang, Juanyong Duan, Ce Zhang, Xia Zhu, Congrui Huang, Yunhai Tong, Bixiong Xu, Jing Bai, Jie Tong, and Qi Zhang. Spectral temporal graph neural network for multivariate time-series forecasting. In *Advances in Neural Information Processing Systems*, volume 33, pp. 17766–17778, 2020.

Yuzhou Chen, Ignacio Segovia-Dominguez, and Yulia R Gel. Z-GCNETs: Time zigzags at graph convolutional networks for time series forecasting. *International Conference on Machine Learning*, 2021.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

H. Edelsbrunner, D. Letscher, and A. Zomorodian. Topological persistence and simplification. In *IEEE FOCS*, pp. 454–463, 2000.

Emadeldeen Eldele, Mohamed Ragab, Zhenghua Chen, Min Wu, Chee Keong Kwoh, Xiaoli Li, and Cuntai Guan. Time-series representation learning via temporal and contextual contrasting. *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, 2021.

Zheng Fang, Qingqing Long, Guojie Song, and Kunqing Xie. Spatial-temporal graph ode networks for traffic flow forecasting. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pp. 364–373, 2021.

Shengnan Guo, Youfang Lin, Ning Feng, Chao Song, and Huaiyu Wan. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 922–929, 2019.

James Douglas Hamilton. *Time series analysis*. Princeton university press, 2020.

Hui He, Qi Zhang, Kun Yi, Kaize Shi, Simeng Bai, Zhendong Niu, and Longbin Cao. Temporal conditional vae for distributional drift adaptation in multivariate time series. *arXiv preprint arXiv:2209.00654*, 2022.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9729–9738, 2020.

R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. In *International Conference on Learning Representations*, 2018.

Christoph Hofer, Florian Graf, Bastian Rieck, Marc Niethammer, and Roland Kwitt. Graph filtration learning. In *International Conference on Machine Learning*, pp. 4314–4323, 2020.

Rongzhou Huang, Chuyin Huang, Yubao Liu, Genan Dai, and Weiyang Kong. Lsgcn: Long short-term traffic prediction with graph convolutional networks. In Christian Bessiere (ed.), *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pp. 2355–2361, 2020.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

Yizhu Jiao, Yun Xiong, Jiawei Zhang, Yao Zhang, Tianqi Zhang, and Yangyong Zhu. Sub-graph contrast for scalable self-supervised graph representation learning. In *IEEE ICDM*, pp. 222–231, 2020.

Mengzhang Li and Zhanxing Zhu. Spatial-temporal fusion graph neural networks for traffic flow forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 4189–4196, 2021.

Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *International Conference on Learning Representations*, 2018.

Xu Liu, Yuxuan Liang, Yu Zheng, Bryan Hooi, and Roger Zimmermann. Spatio-temporal graph contrastive learning. *arXiv:2108.11873*, 2021.

Yixin Liu, Ming Jin, Shirui Pan, Chuan Zhou, Yu Zheng, Feng Xia, and Philip Yu. Graph self-supervised learning: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 2022.

Chris J Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Felix L Opolka, Aaron Solomon, Cătălina Cangea, Petar Veličković, Pietro Liò, and R Devon Hjelm. Spatio-temporal deep graph infomax. In *RLGM ICLR Workshop*, 2019.

Zheyi Pan, Yuxuan Liang, Weifeng Wang, Yong Yu, Yu Zheng, and Junbo Zhang. Urban traffic prediction from spatio-temporal data using deep meta learning. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 1720–1730, 2019.

Jiezhong Qiu, Qibin Chen, Yuxiao Dong, Jing Zhang, Hongxia Yang, Ming Ding, Kuansan Wang, and Jie Tang. Gcc: Graph contrastive coding for graph neural network pre-training. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1150–1160, 2020.

Ignacio Segovia Dominguez, Huikyo Lee, Yuzhou Chen, Michael Garay, Krzysztof M Gorski, and Yulia R Gel. Does air quality really impact covid-19 clinical severity: coupling nasa satellite datasets with geometric deep learning. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pp. 3540–3548, 2021.

Chao Song, Youfang Lin, Shengnan Guo, and Huaiyu Wan. Spatial-temporal synchronous graph convolutional networks: A new framework for spatial-temporal network data forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 914–921, 2020.

Fan-Yun Sun, Jordan Hoffmann, Vikas Verma, and Jian Tang. Infograph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization. In *ICLR*, 2020.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*, volume 27, 2014a.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27:3104–3112, 2014b.

Petar Velickovic, William Fedus, William L Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. Deep graph infomax. *Proceedings of the International Conference on Learning Representations*, 2 (3):4, 2019.

Gerald Woo, Chenghao Liu, Doyen Sahoo, Akshat Kumar, and Steven Hoi. Cost: Contrastive learning of disentangled seasonal-trend representations for time series forecasting. In *International Conference on Learning Representations*, 2021.

Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Chengqi Zhang. Graph wavenet for deep spatial-temporal graph modeling. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pp. 1907–1913, 2019a.

Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Chengqi Zhang. Graph wavenet for deep spatial-temporal graph modeling. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, 2019b.

Ling Yang and Shenda Hong. Unsupervised time-series representation learning with iterative bilinear temporal-spectral fusion. In *International Conference on Machine Learning*, pp. 25038–25054. PMLR, 2022.

Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. Graph contrastive learning with augmentations. *Advances in Neural Information Processing Systems*, 33: 5812–5823, 2020.

Bing Yu, Haoteng Yin, and Zhanxing Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pp. 3634–3640, 2018a.

Bing Yu, Haoteng Yin, and Zhanxing Zhu. Spatio-temporal graph convolutional networks: a deep learning framework for traffic forecasting. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pp. 3634–3640, 2018b.

Zhihan Yue, Yujing Wang, Juanyong Duan, Tianmeng Yang, Congrui Huang, Yunhai Tong, and Bixiong Xu. Ts2vec: Towards universal representation of time series. In *AAAI*, volume 36, pp. 8980–8987, 2022.

A. Zomorodian and G. Carlsson. Computing persistent homology. *Discrete & Computational Geometry*, 33(2):249–274, 2005.

A GATE RECURRENT UNIT WITH TWO-STREAM SPATIAL GRAPH CONVOLUTION

Following Li et al. (2018), equipped with the final embedding $Z_t^{(\ell)}$ derived from our two-stream spatial graph convolution, we employ Gated Recurrent Units (GRU) to learn the spatio-temporal correlations among time series and predict the attributes at each node at a future timestamp,

$$\begin{aligned}
\Re_t &= \psi\left(\Theta_{\Re}\left[\Omega_{t-1}, Z_t^{(\ell)}\right] + b_{\Re}\right),\\
\Im_t &= \psi\left(\Theta_{\Im}\left[\Omega_{t-1}, Z_t^{(\ell)}\right] + b_{\Im}\right),\\
\Omega_t &= \tanh\left(\Theta_{\Omega}\left[\Im_t \odot \Omega_{t-1}, Z_t^{(\ell)}\right] + b_{\Omega}\right),\\
\tilde{\Omega}_t &= \Re_t \odot \Omega_{t-1} + (1 - \Re_t) \odot \Omega_t,
\end{aligned} \tag{8}$$

where $\psi(\cdot)$ is a non-activation function (e.g., ReLU), $\odot$ is the elementwise product, $\Re_t$ is the update gate and $\Im_t$ is the reset gate. $b_{\Re}$, $b_{\Im}$, b_{Ω}, $\Theta_{\Re}$, $\Theta_{\Im}$, and Θ_{Ω} are learnable parameters. $\left[\Omega_{t-1}, Z_t^{(\ell)}\right]$ and Ω_t are the input and output of GRU model, respectively. Then, we can obtain $\tilde{\Omega}_t$ which contains both the spatio-temporal and time-aware information.

B ADDITIONAL DETAILS OF PERSISTENCE-BASED DATA AUGMENTATION

In Figure 3, Upper part (before augmentation) shows the persistence diagram $\mathcal{D}$ (*left*) and its corresponding persistence image $\mathcal{PI}$ (*right*) of the graph in PeMSD3 dataset. Lower part (after augmentation) shows the perturbed persistence diagram $\ddot{\mathcal{D}}$ (*left*; i.e., adding Gaussian noises on above clean $\mathcal{D}$) and its corresponding persistence image $\dddot{\mathcal{PI}}$ (*right*).

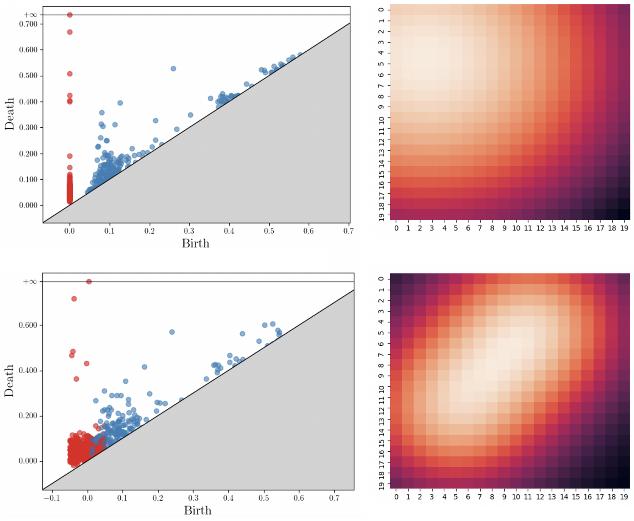

Figure 3: The visualization of persistence-based data augmentation.

C Reproducibility

C.1 Datasets

We conduct experiments on 6 benchmark spatio-temporal graphs, i.e., (i) two real-world traffic flow datasets: PeMSD3 and PeMSD4 (Guo et al., 2019; Song et al., 2020), (ii) the spread of coronavirus disease COVID-19 at county-level (i.e., number of COVID-19 hospitalizations) in states of California (CA) and Pennsylvania (PA) (Segovia Dominguez et al., 2021), (iii) the spread of monkeypox in the United States (recorded by the Centers for Disease Control and Prevention), and (iv) the COVID-19 dataset records the daily global confirmed cases of states/provinces in United States, Australia, Canada, and China (provided by the Center for Systems Science and Engineering at Johns Hopkins University).

Table 6: Summary of datasets used in multi-step spatio-temporal forecasting tasks. †We construct monkeypox and COVID-19 graph structures through applying Euclidean distance function on corresponding training sets.

Dataset	# Nodes	# Edges	Time range
PeMSD3	358	547	09/01/2018 - 11/30/2018
PeMSD4	307	340	01/01/2018 - 02/28/2018
CA	55	535	02/01/2020 - 12/31/2020
PA	60	278	02/01/2020 - 12/31/2020
Monkeypox	52	197†	07/20/2022 - 09/14/2022
COVID-19	289	2250†	01/22/2020 - 11/25/2021

C.2 Baselines

We compare our PCL-GCRN model with 19 state-of-the-art (SOAs) baselines: (i) statistical time series models: HA and VAR (Hamilton, 2020), (ii) RNN-based models: FC-LSTM (Sutskever et al., 2014b) and GRU-ED (Cho et al., 2014), (iii) generic temporal convolutional network: TCN (Bai et al., 2018), (iv) GCN-based models: DCRNN (Li et al., 2018), STGCN (Yu et al., 2018b), GraphWaveNet (Wu et al., 2019b), ASTGCN (Guo et al., 2019), MSTGCN (Guo et al., 2019), STSGCN (Song et al., 2020), AGCRN (Bai et al., 2020), StemGNN Cao et al. (2020), LSGCN (Huang et al., 2020), STFGNN (Li & Zhu, 2021), and TCVAE He et al. (2022), (v) topological-based GCN model: Z-GCNETs (Chen et al., 2021), (vi) ordinary differential equation (ODE)-based neural networks: STGODE (Fang et al., 2021), and (vii) time-series contrastive learning framework: TS2Vec Yue et al. (2022).

C.3 Experimental Settings

We implement our PCL-GCRN model with Pytorch framework on NVIDIA GeForce RTX 3090 GPU. We optimize all the models by Adam optimizer for maximum of 150 epochs. The learning rate is searched in $\{0.001, 0.003, 0.005, 0.01, 0.05, 0.1\}$ with weight decay rate of 0.3. The embedding dimension of node embedding dictionary E_ϕ is searched in $\{1, 2, 5, 10\}$ and the power orders $\mathfrak{K}$ and $\mathfrak{Q}$ in WMCheby-GFT and WMS-AMT are searched in $\{1, 2, 3, 4, 5\}$. Note that, we use Gumbel Softmax trick (Jang et al., 2016; Maddison et al., 2016) to sparsify the adaptive graph structure, and we search for hidden layer dimensions d^C_{out} and d^A_{out} of $\mathfrak{K}$-[multi-hop] Chebyshev convolution and $\mathfrak{Q}$-[multi-hop] adaptive spatial graph convolution in range of $\{16, 32, 64, 128, 256, 512\}$. Hyperparameters π_0, π_1, π_2 (in Eq. 7) are searched in $\{0.1, 0.5, 1.0\}$, $\{0.1, 0.5, 1.0\}$, and $\{0.1, 0.5, 1.0\}$. Furthermore, we perform grid-search for the number of landmark points selection within the range of $\{\lceil 0.1N \rceil, \lceil 0.2N \rceil, \lceil 0.3N \rceil, \lceil 0.4N \rceil, \lceil 0.5N \rceil\}$ and landmark points are selected based on node degree centrality scores. The source code is available at `www.dropbox.com/scl/fo/g5riw7kurppu39fxw02eh/h?dl=0&rlkey=b99xnxvaqllg38k21fhuacalf`.

D Additional Ablation Study

Table 7: Ablation study of augmentations in contrastive learning.

Dataset	Model	Augmentation	MAE	RMSE	MAPE (%)
PeMSD4	PCL-GCRN	Persistence-based data augmentation	**19.32**	**31.51**	**12.71**
	GCL-GCRN	Node dropping	19.33	31.59	12.75
		Attribute perturbation	19.50	32.43	12.87
		Edge perturbation (+)	19.47	31.95	12.83
		Edge perturbation (-)	19.37	31.78	12.79
CA	PCL-GCRN	Persistence-based data augmentation	**72.78**	**336.59**	**44.87**
	GCL-GCRN	Node dropping	72.95	342.31	48.52
		Attribute perturbation	73.07	339.69	45.76
		Edge perturbation (+)	73.29	352.25	46.99
		Edge perturbation (-)	76.44	344.74	46.92
Monkeypox	PCL-GCRN	Persistence-based data augmentation	**183.62**	**620.25**	**22.41**
	GCL-GCRN	Node dropping	193.76	629.83	45.56
		Attribute perturbation	189.17	633.50	33.45
		Edge perturbation (+)	191.18	640.02	34.92
		Edge perturbation (-)	188.34	633.19	28.78