

# Disentangled Spatiotemporal Graph Generative Models

**Yuanqi Du<sup>1</sup>, Xiaojie Guo<sup>2 \*</sup>, Hengning Cao<sup>1</sup>, Yanfang Ye<sup>3</sup>, Liang Zhao<sup>4 †</sup>**

<sup>1</sup>George Mason University, Fairfax, US

<sup>2</sup>JD.COM Silicon Valley Research Center, Mountain View, CA, US

<sup>3</sup>University of Notre Dame, Notre Dame, US

<sup>4</sup>Emory University, Atlanta, US

\*Equally contributing as first author

†The corresponding author

## Abstract

Spatiotemporal graph represents a crucial data structure where the nodes and edges are embedded in a geometric space and can evolve dynamically over time. Nowadays, spatiotemporal graph data is becoming increasingly popular and important, ranging from microscale (e.g. protein folding), to middle-scale (e.g. dynamic functional connectivity), to macro-scale (e.g. human mobility network). Although disentangling and understanding the correlations among spatial, temporal, and graph aspects have been a long-standing key topic in network science, they typically rely on network processing hypothesized by human knowledge. This usually fit well towards the graph properties which can be predefined, but cannot do well for the most cases, especially for many key domains where the human has yet very limited knowledge such as protein folding and biological neuronal networks. In this paper, we aim at pushing forward the modeling and understanding of spatiotemporal graphs via new disentangled deep generative models. Specifically, a new Bayesian model is proposed that factorizes spatiotemporal graphs into spatial, temporal, and graph factors as well as the factors that explain the interplay among them. A variational objective function and new mutual information thresholding algorithms driven by information bottleneck theory have been proposed to maximize the disentanglement among the factors with theoretical guarantees. Qualitative and quantitative experiments on both synthetic and real-world datasets demonstrate the superiority of the proposed model over the state-of-the-arts by up to 69.2% for graph generation and 41.5% for interpretability.

## Introduction

There are two major directions on graph learning research in machine learning: 1) graph representation learning (Kipf and Welling 2016a; Veličković et al. 2017; Hamilton, Ying, and Leskovec 2017), which aims at encoding graph structural information into (low-dimensional) vector space, and 2) graph generation (You et al. 2018; Simonovsky and Komodakis 2018), which reversely aims at constructing a graph-structured data from low-dimensional space containing the graph generation rules or distribution. In this paper, we focus on the second direction, specifically for spatiotemporal graphs. Spatiotemporal graph represents a vital data structure where the nodes and edges are embedded and

evolve in a geometric space, as shown in Fig. 1. Nowadays, spatiotemporal graph data is becoming increasingly popular and important, ranging from epidemic, transportation to biological network modeling (Dye and Gay 2003; Ingraham et al. 2021; Stopher and Meyburg 1975; Teng 1985; Guo et al. 2020a; Rahman et al. 2021; Du et al. 2020; Xu et al. 2021a,b; Zhao 2021). For example, the epidemic spreading network and the protein folding process can both be represented as spatiotemporal graphs, respectively. Spatiotemporal graphs cannot be modeled using either the spatial, graph, or temporal information individually, but require the simultaneous characterization of both the data and their interactions, which results in various patterns (Barthélemy 2011). Spatial and graph aspects of information are usually correlated. For example, geographically nearby people tend to befriend in a social network. A pair of atoms that are very close in space tend to have a bond. Moreover, the above interplay between spatial and graph aspects is a dynamic process, thus, the consideration in time aspect is inevitable for a comprehensive modeling. Recently, although spatiotemporal graph deep learning has stimulated a surge of research for graph representation learning (Cui et al. 2019; Wu et al. 2019; Yu, Yin, and Zhu 2017; Khodayar et al. 2019; Roy et al. 2021; Wu et al. 2019; Yu, Yin, and Zhu 2017; Fu et al. 2021), however, deep generative models for spatiotemporal graph have not been well explored.

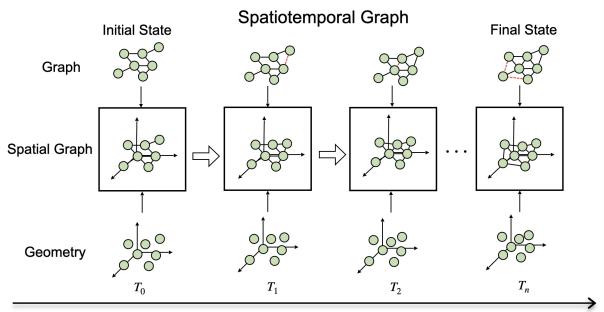


Figure 1: Spatiotemporal graphs represent crucial data structures where the nodes and edges are embedded in a geometric space and their attribute values can evolve dynamically over time. Each column represents the formulation of a snapshot of a spatiotemporal graph.

Modeling and understanding the generative process of spatiotemporal graphs are a long-lasting research topic in domains such as graph theory and network science. Traditional methods usually extend and integrate network models in spatial networks (e.g., protein and molecule structures) and temporal graphs (e.g., traffic networks and epidemic spreading networks) into spatiotemporal graphs which captures some predefined properties of a graph, e.g., degree distribution, structure of community, clustering patterns. However, these models heavily rely on the predefined network process and rich knowledge of the graph properties, while the network properties and generation principles always remain unknown in the real-world applications, such as models that explain the mechanisms of mental diseases in brain networks during an activity of human beings and protein structure folding. Another line of research works is computational simulation models of spatiotemporal graphs customized for specific applications such as epidemics, brain simulator, and transportation simulation (Dye and Gay 2003; Stopher and Meyburg 1975; Teng 1985). However, they are domain-specific with enormously detailed prior knowledge involved. This motivates us to propose the spatiotemporal graph models which can automatically learn the underlying spatial, temporal, and graph processes as well as their interplay.

Recent advanced deep generative models, such as variational auto-encoders (GraphVAE) (Kipf and Welling 2016b), have made important progress towards modeling and understanding each of the three components in spatiotemporal graphs, including graph data (e.g., citation networks) (Kipf and Welling 2016a), spatial networks (e.g., proteins) (Ingraham et al. 2021) and temporal networks (e.g., traffic networks) (Zhang et al. 2020; Zhou et al. 2020), respectively. However, none of the work is designed for spatiotemporal graphs which cannot be effectively modeled by simply integrating the existing techniques, due to several significant challenges: (1) **It is difficult to build a generic spatiotemporal graph generative framework to model spatial, graph, and temporal jointly.** This framework needs to not only capture the intrinsic generic properties shared across all types of spatiotemporal graphs, but also can automatically learn network models that tailor well for specific applications. (2) **It is difficult to effectively distinguish and capture the correlation among the spatial, temporal, and graph information.** As shown in Figure 1, in spatiotemporal graphs, some network information are entangled, such as the only spatial-related information (Row 2) and only graph-related information (Row 3), while some spatial and graph properties are correlated (Row 1). (3) **It is difficult to theoretically ensure the disentanglement among the independent patterns.** In light of model interpretability, disentangling the independent patterns can help better control the properties of the generated spatialtemporal network. Current methods are limited in handling the disentanglement of spatial, temporal, and graph independent properties or factors.

To solve the aforementioned challenges, we propose, to the best of our knowledge, the first generic deep generative model framework that models and disentangles spatiotem-

poral graph data. Specifically, we first propose a novel deep Bayesian network that factorizes spatiotemporal graphs into the time-variant, time-invariant, spatial-graph joint, and independent factors based on inductive bias (Liu et al. 2021). A new objective driven by information-bottleneck theory has been proposed that can maximize the disentanglement of different factors as well as latent variables inside each factor, with theoretical guarantees. To optimize this objective function, a novel information-iterative-thresholding algorithm has been proposed to jointly optimize the objective and optimize its hyperparameters on information bottlenecks with theoretical analysis on optimal conditions. Extensive quantitative and qualitative experiments on two synthetic and two real-world datasets show the superiority of our proposed model over the state-of-the-art graph generative models by up to 69.2% for spatiotemporal graph generation and 41.5% for interpretability.

## Related Work

**Spatiotemporal Graph Deep Representation Learning.** This domain benefits a lot from the deep representation learning techniques for images, sequence, and network data, such as convLSTM. Currently, the most well-developed field is spatiotemporal forecasting in traffic networks (Cui et al. 2019; Wu et al. 2019; Yu, Yin, and Zhu 2017; Khodayar et al. 2019; Roy et al. 2021; Wu et al. 2019; Yu, Yin, and Zhu 2017). Other works (Wang et al. 2019; Zhang et al. 2020) either study spatial network data or temporal graph data. For example, (Wang et al. 2019) considers the spatial locations of nodes, but not capable of generating temporal features. (Zhang et al. 2020) models the temporal features explicitly but do not include spatial features.

**Spatiotemporal Graph Generation** is to generate diverse spatiotemporal graphs, motivated by network design and interpretation (Barthélemy 2011). Traditional spatiotemporal network generation largely relies on human-defined heuristics/prior knowledge about the network being modeled, such as epidemic modeling, transportation modeling, protein modeling, etc. (Stopher and Meyburg 1975; Dye and Gay 2003; Teng 1985), which however almost pose no generalizability to other domain (Dahiyat and Mayo 1997). Another line of models are based on prescribed structural assumptions, such as probabilistic models (Barabási, Albert, and Jeong 1999), configuration models (Bender and Canfield 1978), and stochastic block models (Xu and Hero 2014). These prescribed models aim to model some pre-defined graph properties, e.g. community structures, clustering patterns, etc, which, however, are not sufficient for real-world graph datasets where the prescribed rules are unknown (Holme 2015; Rozenshtein and Gionis 2019). Deep generative models have rarely been used to tackle the spatiotemporal graph generation problem (Zhang 2019).

**Deep Generative Models on Graphs.** Deep generative models achieve great success in computer vision, natural language processing, etc. Recently, increasing attention has been attracted to model graph-structured data by deep generative model. Work in (Simonovsky and Komodakis 2018; You et al. 2018; Wang et al. 2018; Shi et al. 2020; Guo, Du,

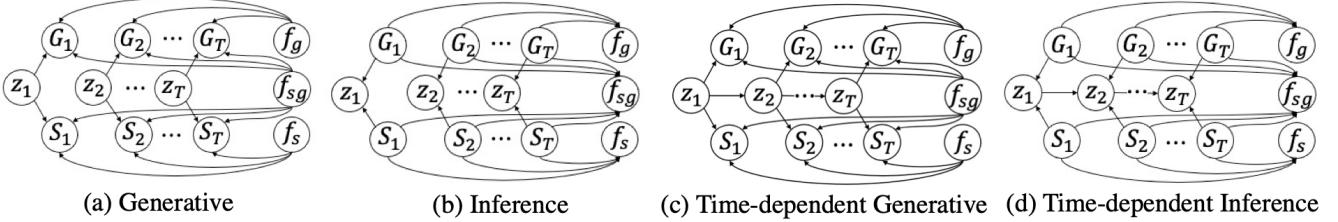


Figure 2: Graphical illustration of the proposed models. (a) The Bayesian network of the proposed probabilistic distribution of spatiotemporal graphs. (b) The approximate inference process of the posterior of latent variables, with conditional independence assumption across time snapshots. (c) The Bayesian network of the alternative probabilistic distribution of spatiotemporal graphs, with dependence assumption across time snapshots. (d) The alternative approximate inference model of the posterior of the proposed model, with dependence assumption across time snapshots.

and Zhao 2020; Du et al. 2021a) transfers popular deep generative models, such as GANs, VAEs, RNNs, Flow-based models, etc., into graph-structured data to model proteins, molecules, etc. and show promising results. For example, graphRNN utilizes an auto-regressive generative model to generate a sequence of nodes and edges by an LSTM model (You et al. 2018). Graphite (Grover, Zweig, and Ermon 2019) and VGAE (Kipf and Welling 2016b) focus on the node-level embedding and form edges between each pair of nodes to generate a graph. GraphVAE (Kipf and Welling 2016b) represents each graph by its edge feature (i.e. adjacency matrix) and node features, and utilizes a VAE model to learn the distribution of the graphs conditioned on the latent representation at the graph-level.

**Disentangled Representation Learning** Disentanglement learning aims to learn a disentangled representation that keeps latent variables separate and interpretable for the variations in the data. It has been shown that such disentangled representation improves the robustness against the adversarial attack and increases the generalizability of the model (Alemi et al. 2016). This inspires much work in VAE to study how to disentangle the latent representations which expose real-world semantic factors and the solutions include adding, removing, or altering the weight of the objective term in the generative tasks (Chen et al. 2018; Kim and Mnih 2018; Kumar, Sattigeri, and Balakrishnan 2017; Lopez et al. 2018; Zhao, Song, and Ermon 2017). Recently, the idea to learn a disentangled representation has also been applied to graph generative models. (Guo et al. 2020b; Ma et al. 2019) disentangles the latent variables that expose semantic factors of the nodes, edges, and the graphs. However, learning a disentangled representation for spatiotemporal graph modeling remains largely unexplored.

## Methodology

### Problem Formulation

A spatiotemporal graph is defined as  $(S_{1:T}, G_{1:T})$ , where  $T$  represents number of time frames of the spatiotemporal graphs, and  $S_{1:T} = \{S_1, S_2, \dots, S_T\}$ ,  $G_{1:T} = \{G_1, \dots, G_T\}$ .  $S_t = (\mathcal{V}_t, C_t)$  represents the geometric information of  $t$ -th snapshot of a spatiotemporal graph, where  $\mathcal{V}_t$  denotes a set of  $N$  nodes and  $C_t \in R^{N \times 3}$  denotes 3D geometric infor-

mation.  $G_t = (\mathcal{V}_t, \mathcal{E}_t, X_t, E_t)$  represents the graph information of  $t$ -th snapshot, where  $\mathcal{E}_t \subseteq \mathcal{V}_t \times \mathcal{V}_t$  is the set of edges.  $E_t \in R^{N \times N \times K}$  refers to the edge weights or adjacent matrix, and  $K$  refers to the edge feature dimension.  $X_t \in R^{N \times M}$  denotes the node feature and  $M$  is the length of the node feature vector. It is worth noting that the geometric information  $S_t$  can not be simply treated as part of node features since this type of representation cannot capture and maintain some properties, such as translation- and rotation-invariances (Fuchs et al. 2020).

This paper aims at proposing a generic data-driven framework for modeling spatiotemporal graphs, under fundamental, necessary factors. First, for any spatiotemporal graphs, there could be patterns that are time-variant and time-invariant. While time-invariant, spatial and graph information could either be correlated or independent, hence it is important to distinguish and capture these different semantic factors via different latent variables. More concretely, the goal is to learn a posterior  $p(S_{1:T}, G_{1:T}|Z, F)$  of the spatiotemporal graphs given four groups of generative latent variables  $Z = z_{1:T} \in R^{L_1}$  for time-variant features and  $F = (f_s \in R^{L_2}, f_g \in R^{L_3}, f_{sg} \in R^{L_4})$  for time-invariant features, where we need to capture and disentangle time-variant factors  $z_{1:T}$ , time-invariant geometric factors  $f_s$ , graph factors  $f_g$  and spatial-graph joint factors  $f_{sg}$ .  $L_1, L_2, L_3$ , and  $L_4$  are the number of variables in each group of factors, respectively. The encoding and generative process of our proposed SpatioTemporal Graph Disentangled Variational Auto-Encoder (STGD-VAE) model is illustrated in Fig. 2(a) and Fig. 2(b). Another implementation of the proposed model following the common time-dependency, namely, STGD-VAE-Dep is illustrated in Fig. 2(c) and Fig. 2(d), and detailed in Supplementary Material.

### The Objective on Spatiotemporal Graph Generative Modeling

To learn the conditional probability  $p(S_{1:T}, G_{1:T}|z_{1:T}, F)$ , it is equal to maximizing the marginal likelihood of the observed spatiotemporal graph sequence  $(S_{1:T}, G_{1:T})$  in expectation over the distribution of the latent representation as  $E_{p_\theta(z_{1:T}, F)} p_\theta(S_{1:T}, G_{1:T}|z_{1:T}, F)$ . The prior distribution of the latent spaces is described as  $p(z_{1:T}, F)$  with the ob-

servation of a spatiotemporal graph sequence  $(S_{1:T}, G_{1:T})$ , which, however, is intractable. Therefore, a variational objective is proposed to tackle this problem, where the posterior distribution is approximated by another distribution  $q_\phi(z_{1:T}, F|S_{1:T}, G_{1:T})$ . The objective can be written as minimizing the Kullback-Leibler Divergence (KLD) between the true prior distribution and the approximate posterior distribution. In order to encourage this disentanglement property of  $q_\phi(z_{1:T}, F|S_{1:T}, G_{1:T})$ , we introduce a constraint by trying to match the inferred posterior configurations of the latent factors to the prior  $p(z_{1:T}, f_s, f_g, f_{sg})$ . This can be achieved if we set each prior to be an isotropic unit Gaussian, i.e.,  $\mathcal{N}(\mathbf{0}, \mathbf{1})$ , leading to a constrained optimization problem as:

$$\begin{aligned} \max_{\theta, \phi} \quad & E_{S_{1:T}, G_{1:T} \sim \mathcal{D}} [E_{q_\phi(z_{1:T}, F|S_{1:T}, G_{1:T})} \\ & \log p_\theta(S_{1:T}, G_{1:T}|z_{1:T}, F)] \\ \text{s.t. } & D_{KL}(q_\phi(z_{1:T}, F|S_{1:T}, G_{1:T})||p(z_{1:T}, F)) < I \end{aligned} \quad (1)$$

where  $\mathcal{D}$  refers to the observed dataset of the spatiotemporal graphs and  $I$  specifies the information that flows via the latent representation.

The above objective can be further decomposed for simple estimation and implementation of each component based on different pre-defined dependence and independence assumptions in the problem formulation, as stated in Lemma .

Given the assumption that: (1)  $S_{1:T} \perp G_{1:T}|(z_{1:t}, F)$ ; (2)  $S_{1:T} \perp f_g$  and  $G_{1:T} \perp f_s$ ; (3)  $G_i \perp G_j|(z_i, z_j, f_g, f_{sg})$  and  $S_i \perp S_j|(z_t, z_k, f_s, f_{sg})$ ; (4)  $z_{1:T} \perp (f_s, f_g, f_{sg})$ , and  $z_1 \perp z_2 \dots \perp z_T$ , the objective of spatiotemporal graph generation can be expressed as

$$\begin{aligned} \max_{\theta, \phi} \quad & E_{S_{1:T}, G_{1:T} \sim \mathcal{D}} E_{q_\phi(z_{1:t}, F|G_{1:T}, S_{1:T})} \\ & \sum_{t=1}^T [\log p_\theta(G_t|z_t, f_g, f_{sg}) + \log p_\theta(S_t|z_t, f_s, f_{sg})] \\ \text{s.t.} \quad & \sum_{t=1}^T D_{KL}(q_\phi(z_t|G_t, S_t)||p(z_t)) < I_t \\ & D_{KL}(q_\phi(f_g|G_{1:T})||p(f_g)) < I_g \\ & D_{KL}(q_\phi(f_s|S_{1:T})||p(f_s)) < I_s \\ & D_{KL}(q_\phi(f_{sg}|S_{1:T}, G_{1:T})||p(f_{sg})) < I_{sg} \\ & I_{sg} \leq C_{sg}, \quad I_t \leq C_t. \end{aligned} \quad (2)$$

In Lemma ,  $I$  is decomposed into four mutual-exclusive information capacity,  $I_s$ ,  $I_g$ ,  $I_{sg}$ , and  $I_t$  in Eq. 2. The detailed proof of Lemma can be found in Supplementary Material.

### Maximizing the Disentanglement among Spatial, Temporal and Graph Factors

One of our goals is to maximize the disentanglement of spatial, temporal, and graph factors. So for example if a factor is merely related to spatial information, we do not want it to be explained by the spatial-graph joint factor  $f_{sg}$ . Analogously, if a factor is invariant to time, we do not want it to be explained by the time-variant factor  $z_t$ . However, this cannot be guaranteed by Equation 2, whose constraints can only enforce variable-level disentanglement within each type of factor instead of a maximized disentanglement across spatial, temporal, and graph factors.

To address the above issue, we first re-interpret the constraints by information bottleneck theory (Burgess, Higgins, and Pal 2018). The posterior distribution  $q_\phi(z_t|S_t, G_t)$ ,  $q_\phi(f_g|G_{1:T})$ ,  $q_\phi(f_s|S_{1:T})$ , and  $q_\phi(f_{sg}|G_{1:T}, S_{1:T})$  are interpreted as information bottleneck for the reconstruction task  $E_{q_\phi(Z|G_{1:T}, S_{1:T})} \log p_\theta(S_{1:T}|z_{1:T}, f_s, f_{sg})$  and  $E_{q_\phi(Z|G_{1:T}, S_{1:T})} \log p_\theta(G_{1:T}|z_{1:T}, f_g, f_{sg})$ . We propose that, by constraining the information flowing through each time-variant variable  $z_t$  to be less than the information entropy of time-variant information  $C_t$ , namely  $I_t \leq C_t$ ,  $z_t$  will capture and only capture the time-variant information. We also propose that, by constraining the information flowing through the spatial-graph joint variable  $f_{sg}$  to be less than the information entropy of the time-invariant correlated spatial-graph factor  $C_{sg}$ , namely  $I_{sg} \leq C_{sg}$ ,  $f_{sg}$  will only capture the time-invariant spatial-graph correlated factor. The new objective function is as follows:

$$\begin{aligned} \max_{\theta, \phi} \quad & E_{S_{1:T}, G_{1:T} \sim \mathcal{D}} E_{q_\phi(z_{1:T}, F|S_{1:T}, G_{1:T})} \\ & \sum_{t=1}^T [\log p_\theta(G_t|z_t, f_g, f_{sg}) + \log p_\theta(S_t|z_t, f_s, f_{sg})] \\ \text{s.t.} \quad & \sum_{t=1}^T D_{KL}(q_\phi(z_t|S_t, G_t)||p(z_t)) < I_t \\ & D_{KL}(q_\phi(f_g|G_{1:T})||p(f_g)) < I_g \\ & D_{KL}(q_\phi(f_s|S_{1:T})||p(f_s)) < I_s \\ & D_{KL}(q_\phi(f_{sg}|S_{1:T}, G_{1:T})||p(f_{sg})) < I_{sg} \\ & I_{sg} \leq C_{sg}, \quad I_t \leq C_t. \end{aligned} \quad (3)$$

This objective has the properties stated in Theorem 1.

**Theorem 1** *The objective in Equation 3 guarantees that  $z_t$  captures and only captures the time-variant information while  $f_{sg}$  captures and only captures the spatial-graph joint information.*

The above theorem is proved based on the condition that (1) the sum of  $I_s$ ,  $I_g$ , and  $I_{sg}$  are large enough to contain the time-variant information, and  $I_s$ ,  $I_g$  are large enough to contain the time-invariant spatial-exclusive and graph-exclusive information, (2)  $I_t \leq C_t$ , and  $I_{sg} \leq C_{sg}$ . Due to the space limit, the detailed proof is provided in Supplementary Material.

### Spatiotemporal Graph Mutual Information Thresholding Algorithm

Eq. 3 is a challenging constrained nonconvex problem that also requires learning its hyperparameters of information bottleneck threshold  $I_{sg}$  and  $I_t$ . This section proposes a novel algorithm along with its optimal condition analysis with respect to the information bottleneck threshold.

Given  $I_s$  and  $I_g$  are constants, the second and third constrain can be rewritten based on the Lagrangian algorithm under KKT condition (Mangasarian 1994) as:

$$\begin{aligned} \mathcal{R}_1 = & \beta_2(D_{KL}(q_\phi(f_s|S_{1:T})||p(f_s))) \\ & + \beta_3(D_{KL}(q_\phi(f_g|G_{1:T})||p(f_g))) \end{aligned} \quad (4)$$

where the Lagrangian multipliers  $\beta_2$  and  $\beta_3$  are the regularization coefficients that control the capacity of the latent space information  $f_s$  and  $f_g$ , respectively.

Next, since  $I_t$  and  $I_{sg}$  in the first constraint is a trainable parameter which ensures  $I_t \leq C_t$  and  $I_{sg} \leq C_{sg}$ , it can be written as a Lagrangian under the KKT condition as

$$\mathcal{R}_2 = \beta_1 \left( \prod_{t=1}^T D_{KL}(q_\phi(z_t|G_t, S_t) || p(z_t)) - I_t \right) \quad (5)$$

$$\mathcal{R}_3 = \beta_4 (D_{KL}(q_\phi(f_{sg}|S_{1:T}, G_{1:T}) || p(f_{sg})) - I_{sg}) \quad (6)$$

Finally, we can optimize the overall objective as:

$$\begin{aligned} \max_{\theta, \phi} \quad & E_{S_{1:T}, G_{1:T} \sim \mathcal{D}} E_{q_\phi(z_{1:T}, f_{1:T} | S_{1:T}, G_{1:T})} \\ & \sum_{t=1}^T [\log p_\theta(G_t | z_t, f_g, f_{sg}) + \log p_\theta(S_t | z_t, f_{st}, f_{sg})] \\ & - \mathcal{R}_1 - \mathcal{R}_2 - \mathcal{R}_3 \\ \text{s.t.} \quad & I_{sg} \leq C_{sg}, \quad I_t \leq C_t \end{aligned} \quad (7)$$

It is very hard to directly optimize the above objective since  $C_t$  and  $C_{sg}$  are unknown. To circumvent the challenge, we propose a novel thresholding strategy consisting of two stages: the first stage is to optimize  $I_{sg}$ , the second stage is to optimize  $I_t$ , as detailed in Algorithm 1. In short, we increase  $I_{sg}$  by  $\alpha$  in every  $K$  until a stopping criteria is satisfied while keeping  $I_t$  at a very low value (Lines 1-6 in Algorithm 1). Then, we stop increasing  $I_{sg}$  and increase  $I_t$  by  $\gamma$  every  $J$  epoch until a stopping criterion is satisfied (Lines 8-14 in Algorithm 1).

---

#### Algorithm 1: Information-iterative-thresholding algorithm

**Require:** The initialized parameter set  $\mathcal{W}$ ; the initialized  $I_t = \epsilon$  and  $I_{sg} = \epsilon$  ( $I_t \notin \mathcal{W}$ ,  $I_{sg} \notin \mathcal{W}$  and  $\epsilon$  is a very small number, e.g.  $1 \times 10^{-5}$ ); the increase step  $\gamma$ ,  $\alpha$  for optimizing  $I_t$  and  $I_{sg}$ ; the number of epochs  $J, K$  of optimization for each updated  $I_t$  and  $I_{sg}$ .

**Ensure:** The optimized parameter set  $\mathcal{W}$ .

```

1: while  $\mathcal{R}_3 < 0$  do {stopping criterion for  $I_{sg}$ }
2:    $I_{sg} := I_{sg} + \alpha$ 
3:   for epoch = 1 :  $K$  do {increase  $I_{sg}$  every  $K$  epoch}
4:     Compute the gradient of  $\mathcal{W}$  via backpropagation.
5:     Update  $\mathcal{W}$  based on gradient with  $I_{sg}$  and  $I_t$  fixed.
6:   end for
7: end while
8: while  $\mathcal{R}_2 < 0$  do {stopping criterion for  $I_t$ }
9:    $I_t := I_t + \gamma$ 
10:  for epoch = 1 :  $J$  do {increase  $I_t$  every  $J$  epoch}
11:    Compute the gradient of  $\mathcal{W}$  via backpropagation.
12:    Update  $\mathcal{W}$  based on gradient with  $I_t$  and  $I_{sg}$  fixed.
13:  end for
14: end while
```

---

The proposed optimization strategy guarantees that  $z_t$  captures and only captures the time-variant information while  $f_{sg}$  captures and only captures the spatial-graph joint information based on the following theorem.

**Theorem 2** *The latent variable  $z_t$  captures and only captures the time-variant information if  $\mathcal{R}_2 < 0$  is satisfied. The latent variable  $f_{sg}$  captures and only captures the time-invariant spatial-graph correlated information if  $\mathcal{R}_3 < 0$  is satisfied.*

Notably, at initial stage,  $\mathcal{R}_3 = 0$  and  $\mathcal{R}_2 = 0$ , we then gradually increase  $I_t$  and  $I_{sg}$ , and at each step while well-trained,  $\prod_{t=1}^T D_{KL}(q_\phi(z_t|G_t, S_t) || p(z_t))$  and  $D_{KL}(q_\phi(f_{sg}|G_{1:T}, S_{1:T}) || p(f_{sg}))$  will keep increasing to catch  $I_t$  and  $I_{sg}$ . When  $\mathcal{R}_3 < 0$  and  $\mathcal{R}_2 < 0$ , it indicates that the information that captured by  $I_t$  and  $I_{sg}$  do not increase anymore, namely  $I_t = C_t$  and  $I_{sg} = C_{sg}$ . Thus, the whole optimization process can be stopped. During the whole process, the two constraint  $I_t \leq C_t$  and  $I_{sg} \leq C_{sg}$  are always satisfied, namely,  $z_t$  always captures and only captures the time-variant information and  $f_{sg}$  always captures and only captures the time-invariant spatial-graph correlated information. The detailed proof can be found in Supplementary Material. In practice, we set  $\beta_1, \beta_2, \beta_3$ , and  $\beta_4$  as 1 and the model is not sensitive to these parameters.

Our model consists of four encoders which model  $q_\phi(f_s|S_{1:T}, G_{1:T})$ ,  $q_\phi(f_g|S_{1:T}, G_{1:T})$ ,  $q_\phi(f_{sg}|S_{1:T}, G_{1:T})$ , and  $q_\phi(z_t|S_{1:T}, G_{1:T})$  respectively. There are also two decoders to model  $p_\theta(G_t|z_t, f_g, f_{sg})$  and  $p_\theta(S_t|z_t, f_g, f_{sg})$ , respectively. Specifically, we utilize a typical graph convolution neural network to encode the graph factors and a typical convolution neural network for the spatial factors. For the spatial-graph correlated factors, we utilize a Spatial-Network Message Passing Neural Network (S-MPNN) (Guo, Du, and Zhao 2021), which considers both the spatial and graph information while passing messages. In terms of the temporal factors, we consider that could involve both spatial and graph variance, thus, we take another S-MPNN for the temporal factors. For decoders, we utilize a typical convolution neural network for the spatial factors, and a similar graph decoder proposed in NED-VAE (Guo, Wu, and Zhao 2018) for the graph factors. Due to the space limits, the detailed description of the encoders and decoders, as well as the time complexity analysis are provided in the Supplementary Material.

## Experiments

### Datasets

We validate the effectiveness of our proposed models on two synthetic datasets and two real-world datasets, (1) *Dynamic Waxman Random Graphs*, (2) *Dynamic Random Geometry Graphs*, (3) *Protein Folding Dataset*, and (4) *Traffic Dataset MERT-LA* (Du et al. 2021b). The first two are well-known spatial network datasets (Bradonjić, Hagberg, and Percus 2007; Waxman 1988), which randomly place nodes in a geometry and the edges are connected by predefined distance measures, with variances through the time dimension. The protein folding dataset consists of the folding steps of a protein with 8 amino acids (Guo et al. 2020b). The traffic dataset contains sequences of graphs which contains traffic speeds connected by 207 sensors (Jagadish et al. 2014). The details of generation process of the datasets can be found in Supplementary Material.

### Comparison Methods

To validate the proposed models in spatiotemporal graph generation, despite that no previous deep models specially designed for the spatiotemporal graph generation task, we

Table 1: The evaluation results for the generated spatiotemporal graphs for different datasets (*KLD\_cls* refers to KLD of graph clustering coefficient, *KLD\_ds* refers to KLD of graph density, *KLD\_bet* refers to KLD of betweenness centrality, and *KLD\_tcorr* refers to KLD of temporal correlation. (Best results are highlighted in bold, The KLD evaluation on the traffic dataset is not reported because the metrics are on graph topology, while the graph topology for the traffic dataset is unchanged.)

Dataset	Method	Node	Spatial	Edge	KLD_cls	KLD.ds	KLD.bet	KLD.tcorr	AvgMI
DWR Graph	DSBM	N/A	N/A	54.95%	0.90	1.10	0.63	0.73	N/A
	GraphVAE	0.57	0.57	57.14%	1.63	1.82	0.91	0.85	N/A
	GraphRNN	N/A	N/A	55.24%	1.97	2.50	1.00	1.35	N/A
	beta-VAE	0.0012	0.0011	69.05%	0.43	1.61	1.82	0.36	2.25
	beta-TCVAE	0.0013	0.0012	69.04%	0.47	1.37	1.56	0.08	2.33
	NEND-IPVAE	0.016	0.0008	65.80%	1.39	1.82	2.78	0.11	2.52
	STGD-VAE	<b>0.0003</b>	<b>0.0001</b>	<b>69.99%</b>	<b>0.14</b>	0.74	<b>0.40</b>	<b>0.03</b>	<b>2.03</b>
DRG Graph	STGD-VAE-Dep	0.0191	0.0005	66.28%	0.45	<b>0.55</b>	0.54	0.38	2.04
	DSBM	N/A	N/A	81.88%	1.77	2.87	3.38	0.64	N/A
	GraphVAE	0.56	0.74	85.75%	4.46	2.65	1.60	3.08	N/A
	GraphRNN	N/A	N/A	85.32%	0.57	1.24	2.40	0.85	N/A
	beta-VAE	0.0013	0.0017	91.75%	0.34	1.24	1.47	2.15	2.29
	beta-TCVAE	0.0018	0.0019	91.62%	0.52	1.58	1.46	2.38	2.24
	NED-IPVAE	0.0175	0.0018	89.84%	0.37	1.05	1.72	0.23	2.42
Protein	STGD-VAE	<b>0.0004</b>	<b>0.0015</b>	<b>91.88%</b>	<b>0.14</b>	0.72	0.28	<b>0.11</b>	<b>2.07</b>
	STGD-VAE-Dep	0.0008	0.0017	91.28%	<b>0.14</b>	<b>0.71</b>	<b>0.26</b>	1.67	2.08
	DSBM	N/A	N/A	70.78%	1.00	0.93	1.15	1.53	N/A
	GraphVAE	N/A	553.82	62.54%	1.26	1.44	1.48	1.90	N/A
	GraphRNN	N/A	N/A	71.17%	1.05	1.15	1.43	0.83	N/A
	beta-VAE	N/A	52.74	85.58%	0.16	<b>0.14</b>	0.46	0.61	1.04
	beta-TCVAE	N/A	35.05	95.80%	0.27	0.58	<b>0.34</b>	0.71	1.09
Traffic	NED-IPVAE	N/A	36.12	92.48%	1.08	0.79	0.44	2.64	1.15
	STGD-VAE	N/A	28.77	<b>99.79%</b>	0.33	0.21	0.53	<b>0.23</b>	<b>0.70</b>
	STGD-VAE-Dep	N/A	<b>28.42</b>	96.79%	<b>0.13</b>	0.54	1.55	0.24	0.76
	DSBM	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	GraphVAE	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	GraphRNN	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	beta-VAE	7.15	N/A	N/A	N/A	N/A	N/A	N/A	1.37
	beta-TCVAE	8.50	N/A	N/A	N/A	N/A	N/A	N/A	1.18
	NED-IPVAE	31.95	N/A	N/A	N/A	N/A	N/A	N/A	1.18
	STGD-VAE	6.78	N/A	N/A	N/A	N/A	N/A	N/A	<b>0.69</b>
	STGD-VAE-Dep	<b>5.13</b>	N/A	N/A	N/A	N/A	N/A	N/A	1.06

compare with some state-of-the-art graph generation model for generic graphs, including GraphRNN (You et al. 2018), a graph generation method that utilizes recurrent neural network structure to model graphs as sequences of edges and nodes, GraphVAE (Kipf and Welling 2016b), a variational auto-encoder based graph generative model focusing on handling small graphs, and a traditional algorithm DSBM (Xu and Hero 2014) which utilizes stochastic block model for static graph generation and combines it with Markov chains to achieve temporal graph generation. For GraphRNN and GraphVAE, we train individual models for each time step respectively for generating temporal graph sequences.

To validate the significance of our proposed disentanglement objective and optimization strategy for spatiotemporal graphs modeling, several SOTA disentanglement techniques are compared including beta-VAE (Higgins, Matthey, and Pal 2016), beta-TC-VAE (Chen et al. 2018), and NED-IPVAE (Guo et al. 2020b), where the architecture utilized is the same as our model, except with different disentanglement objectives.

## Evaluation of the Performance in Spatiotemporal Graph Generation

To validate the power of our proposed models to learn the representations of spatiotemporal graphs, we evaluate the

reconstruction performance among different elements of a spatiotemporal graph, e.g. node, edge, and spatial locations by calculating the difference between the real ones and reconstructed ones. The mean square errors (MSE) is calculated for node attribute prediction and spatial prediction. The accuracy is used for evaluating edge connection prediction. To evaluate the generation performance, we select several common evaluation metrics to measure graph-structured data in multiple aspects, including graph statistics and temporal statistics. Specifically, we calculate a commonly used measurement for probability distributions, Kullback-Leibler Divergence (KLD), between the real training data distribution and the generated data distribution in terms of: (1) graph density, (2) average clustering coefficient, (3) betweenness centrality, and (4) temporal correlation in the spatiotemporal graphs.

The quantitative evaluation on the two synthetic datasets and two real-world datasets are reported in Table 1. First of all, for dynamic Waxman random graphs, STGD-VAE achieves the best overall evaluation which not only reconstructs the node, spatial, and edge attributes very well, but also learns the underlying graph, temporal statistical distribution of the training set. To be specific, STGD-VAE outperforms the two disentangled VAE models, beta-VAE and beta-TCVAE by 1.3% in reconstructing the graph edge connection. beta-VAE and beta-TCVAE fail to learn the exact

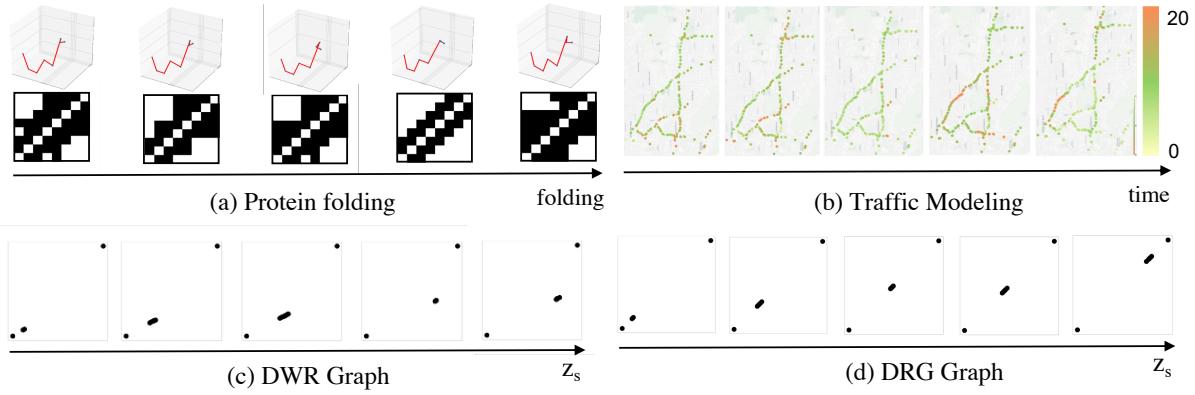


Figure 3: Qualitative evaluation on two synthetic datasets and two real-world datasets.

underlying property statistical distribution of the graphs. We consider this as a benefit of the disentanglement of different types of features, which allows for clearer and better representations and results in a roughly 62.5% improvement. For the dynamic random geometry graph dataset, the results are very similar to DWR Graph which mainly due to the similar way of generating the synthetic datasets, but with difference geometries. Obviously, our proposed STGD-VAE outperforms the best baseline model, beta-VAE, by at most 69.2% in the reconstruction evaluation and roughly 58.8% in learning the property distribution.

In the two real-world datasets, the proposed STGD-VAE and STGD-VAE-Dep outperform other models in reconstructing the spatial locations of the nodes and predicting the contacts between residues by a large margin, which results in up to 19% increase in predicting the spatial locations, and up to 4% increase in contact prediction. However, in some of the graph statistics, our proposed models fail to beat all the baseline models because some of the graph statistics are not designed for the real-world protein structures. However, our proposed models still achieve comparable results in those evaluations. In the last traffic dataset, the graph topology and geometry stay the same, the only changing part is the node feature, which represent the traffic speeds. Our models achieve the best two in the speed prediction task and leave the third as highest as a 25.9% gap.

### Evaluation of the Performance in Disentangled Representation learning

**Quantitative Evaluation.** For quantitative evaluation, the typical **avgMI score** (Locatello et al. 2019) is utilized here to evaluate the power of the encoder to learn disentangled representations towards the predefined semantic factors by measuring the mutual distance between the latent variables and the semantic properties. Thus it is calculated based on the learned latent variables and the input graphs in the testing sets. As shown in Table 1, for all the four datasets, beta-VAE, beta-TCVAE, and NED-IPVAE achieve similar results in terms of avgMI score. GraphRNN, GraphVAE, and DSBM are not capable of disentanglement learning, and thus, no score is reported on this evaluation metric. Between

the two proposed models, STGD-VAE and STGD-VAE-Dep achieve comparable results in most of the cases, STGD-VAE is slightly better and have an up to 41.5% improvement over the best baselines.

**Qualitative Evaluation** As in the conventional qualitative evaluation in disentanglement representation learning (Chen et al. 2018; Higgins, Matthey, and Pal 2016), we change the value of one latent variable continuously while fixing the remaining variables to see the variation of the semantic factor it controls. In Fig. 3(a) and 3(b), we visualize the folding process of the protein structures and the traffic modeling process. We can observe that the residues on the right side are slightly folding up and moving towards left. For the traffic dataset, it is worth noting that the traffic speed is constantly changing in different time steps which reflects the real-time traffic situations. In Fig. 3(c) and 3(d), we also visualize the changes of the generated graphs when the latent factor  $z_s$  of our STGD-VAE model change from -5 to 5 in the dynamic Waxman random graph and the dynamic random geometry graph dataset, respectively. Clearly, the spatial location is changed accordingly, from the left-bottom corner to nearly the right-top corner, which shows that the latent variables learn and expose the semantic factors well.

### Conclusion

In this paper, we introduce STGD-VAE and STGD-VAE-Dep, to the best of our knowledge, the first general deep generative model framework for spatiotemporal graphs. Specifically, we propose a new Bayesian model that factorizes spatiotemporal graphs into spatial, temporal, and graph factors as well as the factors that model the interactions among them. Moreover, a variational objective function and a new mutual information thresholding algorithm based on information bottleneck are proposed to maximize the disentanglement among the factors with theoretical guarantees. The comparison with six state-of-the-art deep generative models validates the superiority of our proposed models from multiple tasks, including, graph generation and disentangled representation learning.

## Acknowledgement

This work was supported by the National Science Foundation (NSF) Grant No. 1755850, No. 1841520, No. 2007716, No. 2007976, No. 1942594, No. 1907805, a Jeffress Memorial Trust Award, Amazon Research Award, NVIDIA GPU Grant, and NSF Computing Innovative Fellowship (Subaward No. 2021CIF-Emory-05).

## References

- Alemi, A. A.; Fischer, I.; Dillon, J. V.; and Murphy, K. 2016. Deep variational information bottleneck. *arXiv preprint arXiv:1612.00410*.
- Barabási, A.-L.; Albert, R.; and Jeong, H. 1999. Mean-field theory for scale-free random networks. *Physica A: Statistical Mechanics and its Applications*, 272(1-2): 173–187.
- Barthélemy, M. 2011. Spatial networks. *Physics Reports*, 499(1-3): 1–101.
- Bender, E. A.; and Canfield, E. R. 1978. The asymptotic number of labeled graphs with given degree sequences. *Journal of Combinatorial Theory, Series A*, 24(3): 296–307.
- Bradonjić, M.; Hagberg, A.; and Percus, A. G. 2007. Giant component and connectivity in geographical threshold graphs. In *International Workshop on Algorithms and Models for the Web-Graph*, 209–216. Springer.
- Burgess, C. P.; Higgins, I.; and Pal, A. e. a. 2018. Understanding disentangling in  $\beta$ -VAE. *arXiv preprint arXiv:1804.03599*.
- Chen, R. T.; Li, X.; Grosse, R. B.; and Duvenaud, D. K. 2018. Isolating sources of disentanglement in variational autoencoders. In *NeurIPS*, 2610–2620.
- Cui, Z.; Henrickson, K.; Ke, R.; and Wang, Y. 2019. Traffic graph convolutional recurrent neural network: A deep learning framework for network-scale traffic learning and forecasting. *IEEE Transactions on Intelligent Transportation Systems*, 21(11): 4883–4894.
- Dahiyat, B. I.; and Mayo, S. L. 1997. De novo protein design: fully automated sequence selection. *Science*, 278(5335): 82–87.
- Du, W.; Zhang, H.; Du, Y.; Meng, Q.; Chen, W.; Shao, B.; and Liu, T.-Y. 2021a. Equivariant vector field network for many-body system modeling. *arXiv preprint arXiv:2110.14811*.
- Du, Y.; Guo, X.; Shehu, A.; and Zhao, L. 2020. Interpretable Molecule Generation via Disentanglement Learning. In *Proceedings of the 11th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, 1–8.
- Du, Y.; Wang, S.; Guo, X.; Cao, H.; Hu, S.; Jiang, J.; Varala, A.; Angirekula, A.; and Zhao, L. 2021b. GraphGT: Machine Learning Datasets for Graph Generation and Transformation. In *The 35th Conference on Neural Information Processing Systems (NeurIPS 2021)*.
- Dye, C.; and Gay, N. 2003. Modeling the SARS epidemic. *Science*, 300(5627): 1884–1885.
- Fu, T.; Xiao, C.; Qian, C.; Glass, L. M.; and Sun, J. 2021. Probabilistic and Dynamic Molecule-Disease Interaction Modeling for Drug Discovery. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 404–414.
- Fuchs, F. B.; Worrall, D. E.; Fischer, V.; and Welling, M. 2020. SE (3)-transformers: 3D roto-translation equivariant attention networks. *arXiv preprint arXiv:2006.10503*.
- Grover, A.; Zweig, A.; and Ermon, S. 2019. Graphite: Iterative generative modeling of graphs. In *International conference on machine learning*, 2434–2444. PMLR.
- Guo, X.; Du, Y.; Tadepalli, S.; Zhao, L.; and Shehu, A. 2020a. Generating tertiary protein structures via an interpretative variational autoencoder. *arXiv preprint arXiv:2004.07119*.
- Guo, X.; Du, Y.; and Zhao, L. 2020. Property Controllable Variational Autoencoder via Invertible Mutual Dependence. In *International Conference on Learning Representations*.
- Guo, X.; Du, Y.; and Zhao, L. 2021. Disentangled Deep Generative model for Spatial Networks. *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.
- Guo, X.; Wu, L.; and Zhao, L. 2018. Deep graph translation. *arXiv preprint arXiv:1805.09980*.
- Guo, X.; Zhao, L.; Qin, Z.; Wu, L.; Shehu, A.; and Ye, Y. 2020b. Interpretable Deep Graph Generation with Node-Edge Co-Disentanglement. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1697–1707.
- Hamilton, W. L.; Ying, R.; and Leskovec, J. 2017. Inductive representation learning on large graphs. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 1025–1035.
- Higgins, I.; Matthey, L.; and Pal, A. e. a. 2016. beta-vae: Learning basic visual concepts with a constrained variational framework.
- Holme, P. 2015. Modern temporal network theory: a colloquium. *The European Physical Journal B*, 88(9): 1–30.
- Ingraham, J.; Garg, V. K.; Barzilay, R.; and Jaakkola, T. S. 2021. Generative models for graph-based protein design.
- Jagadish, H. V.; Gehrke, J.; Labrinidis, A.; Papakonstantinou, Y.; Patel, J. M.; Ramakrishnan, R.; and Shahabi, C. 2014. Big data and its technical challenges. *Communications of the ACM*, 57(7): 86–94.
- Khodayar, M.; Mohammadi, S.; Khodayar, M. E.; Wang, J.; and Liu, G. 2019. Convolutional graph autoencoder: A generative deep neural network for probabilistic spatio-temporal solar irradiance forecasting. *IEEE Transactions on Sustainable Energy*, 11(2): 571–583.
- Kim, H.; and Mnih, A. 2018. Disentangling by factorising. *ICML*.
- Kipf, T. N.; and Welling, M. 2016a. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.
- Kipf, T. N.; and Welling, M. 2016b. Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308*.

- Kumar, A.; Sattigeri, P.; and Balakrishnan, A. 2017. Variational inference of disentangled latent concepts from unlabeled observations. *arXiv preprint arXiv:1711.00848*.
- Liu, Z.; Chen, Y.; Du, Y.; and Tegmark, M. 2021. Physics-Augmented Learning: A New Paradigm Beyond Physics-Informed Learning. *arXiv preprint arXiv:2109.13901*.
- Locatello, F.; Bauer, S.; Lucic, M.; Raetsch, G.; Gelly, S.; Schölkopf, B.; and Bachem, O. 2019. Challenging common assumptions in the unsupervised learning of disentangled representations. In *international conference on machine learning*, 4114–4124. PMLR.
- Lopez, R.; Regier, J.; Jordan, M. I.; and Yosef, N. 2018. Information constraints on auto-encoding variational bayes. *arXiv preprint arXiv:1805.08672*.
- Ma, J.; Cui, P.; Kuang, K.; Wang, X.; and Zhu, W. 2019. Disentangled graph convolutional networks. In *International Conference on Machine Learning*, 4212–4221. PMLR.
- Mangasarian, O. L. 1994. *Nonlinear programming*. SIAM.
- Rahman, T.; Du, Y.; Zhao, L.; and Shehu, A. 2021. Generative Adversarial Learning of Protein Tertiary Structures. *Molecules*, 26(5): 1209.
- Roy, A.; Roy, K. K.; Ali, A. A.; Amin, M. A.; and Rahman, A. 2021. Unified Spatio-Temporal Modeling for Traffic Forecasting using Graph Neural Network. *arXiv preprint arXiv:2104.12518*.
- Rozenshtein, P.; and Gionis, A. 2019. Mining temporal networks. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 3225–3226.
- Shi, C.; Xu, M.; Zhu, Z.; Zhang, W.; Zhang, M.; and Tang, J. 2020. Graphaf: a flow-based autoregressive model for molecular graph generation. *arXiv preprint arXiv:2001.09382*.
- Simonovsky, M.; and Komodakis, N. 2018. Graphvae: Towards generation of small graphs using variational autoencoders. In *International Conference on Artificial Neural Networks*, 412–422. Springer.
- Stopher, P. R.; and Meyburg, A. H. 1975. Urban transportation modeling and planning.
- Teng, P. 1985. A comparison of simulation approaches to epidemic modeling. *Annual Review of Phytopathology*, 23(1): 351–379.
- Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; and Bengio, Y. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.
- Wang, H.; Wang, J.; Wang, J.; Zhao, M.; Zhang, W.; Zhang, F.; Xie, X.; and Guo, M. 2018. Graphgan: Graph representation learning with generative adversarial nets. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.
- Wang, X.; Li, Z.; Jiang, M.; Wang, S.; Zhang, S.; and Wei, Z. 2019. Molecule property prediction based on spatial graph embedding. *Journal of chemical information and modeling*, 59(9): 3817–3828.
- Waxman, B. M. 1988. Routing of multipoint connections. *IEEE journal on selected areas in communications*, 6(9): 1617–1622.
- Wu, Z.; Pan, S.; Long, G.; Jiang, J.; and Zhang, C. 2019. Graph wavenet for deep spatial-temporal graph modeling. *arXiv preprint arXiv:1906.00121*.
- Xu, K. S.; and Hero, A. O. 2014. Dynamic stochastic block-models for time-evolving social networks. *IEEE Journal of Selected Topics in Signal Processing*, 8(4): 552–562.
- Xu, R.; Guo, Y.; Han, X.; Xia, X.; Xiang, H.; and Ma, J. 2021a. OpenCDA: an open cooperative driving automation framework integrated with co-simulation. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, 1155–1162. IEEE.
- Xu, R.; Xiang, H.; Xia, X.; Han, X.; Liu, J.; and Ma, J. 2021b. OPV2V: An Open Benchmark Dataset and Fusion Pipeline for Perception with Vehicle-to-Vehicle Communication. *arXiv preprint arXiv:2109.07644*.
- You, J.; Ying, R.; Ren, X.; Hamilton, W.; and Leskovec, J. 2018. Graphrnn: Generating realistic graphs with deep autoregressive models. In *International Conference on Machine Learning*, 5708–5717. PMLR.
- Yu, B.; Yin, H.; and Zhu, Z. 2017. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint arXiv:1709.04875*.
- Zhang, L. 2019. STGGAN: Spatial-temporal graph generation. In *Proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 608–609.
- Zhang, L.; Zhao, L.; Qin, S.; and Pfoser, D. 2020. Tg-gan: Continuous-time temporal graph generation with deep generative models. *arXiv e-prints*, arXiv–2005.
- Zhao, L. 2021. Event Prediction in the Big Data Era: A Systematic Survey. *ACM Computing Surveys (CSUR)*, 54(5): 1–37.
- Zhao, S.; Song, J.; and Ermon, S. 2017. Infovae: Information maximizing variational autoencoders. *arXiv preprint arXiv:1706.02262*.
- Zhou, D.; Zheng, L.; Han, J.; and He, J. 2020. A data-driven graph generative model for temporal interaction networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 401–411.