Learning Disentangled Representation for Spatiotemporal Graph Generation

Yuanqi Du, Xiaojie Guo, Hengning Cao
Department of Computer Science
George Mason University
Fairfax, VA 22030
{ydu6, xguo7, hcao7}@gmu.edu

Yanfang Ye
Department of Computer and Data Sciences
Case Western Reserve University
Cleveland, OH 44106
yanfang.ye@case.edu

Liang Zhao
Department of Computer Science
Emory University
Atlanta, GA 30322
liang.zhao@emory.edu

Abstract

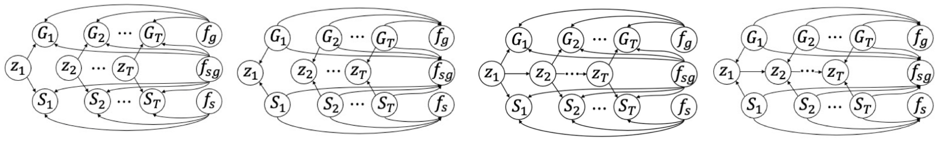
Modeling and understanding spatiotemporal graphs have been a long-standing research topic in network science and typically replies on network processing hypothesized by human knowledge. In this paper, we aim at pushing forward the modeling and understanding of spatiotemporal graphs via new disentangled 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-art by up to 69.2% for graph generation and 41.5% for interpretability.

1 Introduction

Spatiotemporal graph represents a vital data structure where the nodes and edges are embedded and evolve in a geometric space. Nowadays, spatiotemporal graph data is becoming increasingly popular and important, ranging from epidemic, transportation to biological network modeling [8, 16, 25, 26]. 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 [1]. Spatial and graph aspects of information are usually correlated. For example, geographically nearby people tend to befriend in a social network. 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 [5, 28, 31], however, deep generative models for spatiotemporal graphs have not been well explored.

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)

NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications



(a) Generative (b) Inference (c) Time-dependent Generative (d) Time-dependent Inference

Figure 1: Graphical illustration of the proposed models.

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 [8, 25, 26, 11, 6, 9, 23]. However, they are domain-specific with enormously detailed prior knowledge involved. This motivates us to propose the spatiotemporal graph models which can automatically learning the underlying spatial, temporal, and graph processes as well as their interplay.

In this paper, we propose, to the best of our knowledge, the first general deep generative model framework that models and disentangles spatiotemporal 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. 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.

2 Methodology

2.1 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 $S_{1:T} = \{S_1, S_2, ...S_T\}, G_{1:T} = \{G_1, ...G_T\}$. $S_t = (\mathcal{V}_t, C_t)$ represents the geometric information of t-th snapshot of a spatiotemporal graph, where $\mathcal{V}_t$ denotes a set of N nodes and $C_t \in \mathbb{R}^{N \times 3}$ denotes 3D geometric information. $G_t = (\mathcal{V}_t, \mathcal{E}_t, X_t, E_t)$ 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. $E_t \in \mathbb{R}^{N \times N \times K}$ refers to the edge weights or adjacent matrix, and K refers to the edge feature dimension. $X_t \in \mathbb{R}^{N \times M}$ denotes the node feature and M is the length of the node feature vector.

This paper aims at proposing a general 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 $Z = z_{1:T} \in \mathbb{R}^{L_1}$ for time-variant features and $F = (f_s \in \mathbb{R}^{L_2}, f_g \in \mathbb{R}^{L_3}, f_{sg} \in \mathbb{R}^{L_4})$ 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 **S**patio**T**emporal **G**raph **D**isentangled **V**ariational **A**uto-**E**ncoder (STGD-VAE) model is illustrated in Fig. 1(a) and Fig. 1(b). Another implementation of the proposed model following the common time-dependency assumption, namely, STGD-VAE-Dep is illustrated in Fig. 1(c) and Fig. 1(d), and detailed in the appendix.

2.2 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 $\mathbb{E}_{p_\theta(z_{1:T},F)} p_\theta(S_{1:T}, G_{1:T}|z_{1:T}, F)$. The prior distribution of the latent spaces is described as $p(z_{1:T}, F)$ with the observation 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:

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

The detailed objective and proposed mutual information thresholding algorithm can be found in the appendix.

3 Experiments

3.1 Experiment Set-up

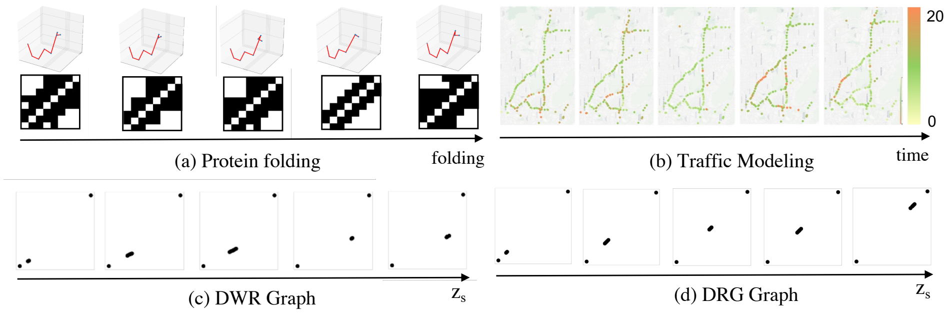
We validate the effectiveness of our proposed models on two synthetic datasets and two real-world datasets [7], (1) *Dynamic Waxman Random Graphs*, (2) *Dynamic Random Geometry Graphs*, (3) *Protein Folding Dataset*, and (4) *Traffic Dataset MERT-LA*. Despite no previous deep generative models specifically designed for spatiotemporal graph generation, we compare with some general graph generation models, including GraphRNN [30], GraphVAE [20], and a traditional algorithm DSBM [29]. In terms of disentanglement evaluation, we also apply and compare with beta-VAE [15], beta-TC-VAE [4], and NED-IPVAE [13] to our proposed method.

3.2 Results

Quantitative Evaluation. We quantitatively evaluate the performance of our proposed model in two synthetic datasets and two real-world datasets by three types of evaluations, as shown in Table 1. We first evaluate the reconstruction via calculating the difference (e.g. mean sqaured errors) between the real graphs and the reconstructed graphs. Then, we evaluate the learnt graph property (i.e. (1) graph density, (2) average clustering coefficient, (3) betweenness centrality, and (4) temporal correlation) distribution comparing to the training one via Kullback-Leibler Divergence (KLD). Finally, we calculate the avgMI score [21] which evaluates the disentanglement of the learnt latent space. We make several observations from the table. Firstly, STGD-VAE achieves the best overall results in the two synthetic datasets, both in reconstruction, distribution and disentanglement evaluation. Secondly, both STGD-VAE and STGD-VAE-Dep perform well in the two real-world datasets. Lastly, the proposed disentanglement of spatiotemporal graph factors greatly improve the disentanglement quality of the latent space.

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.

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	**0.0003**	**0.0001**	**69.99%**	**0.14**	0.74	**0.40**	**0.03**	**2.03**
	STGD-VAE-Dep	0.0191	0.0005	66.28%	0.45	**0.55**	0.54	0.38	2.04
DRG Graph	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
	STGD-VAE	**0.0004**	**0.0015**	**91.88%**	**0.14**	0.72	0.28	**0.11**	**2.07**
	STGD-VAE-Dep	0.0008	0.0017	91.28%	**0.14**	**0.71**	**0.26**	1.67	2.08
Protein	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	**0.14**	0.46	0.61	1.04
	beta-TCVAE	N/A	35.05	95.80%	0.27	0.58	**0.34**	0.71	1.09
	NED-IPVAE	N/A	36.12	92.48%	1.08	0.79	0.44	2.64	1.15
	STGD-VAE	N/A	28.77	**99.79%**	0.33	0.21	0.53	**0.23**	**0.70**
	STGD-VAE-Dep	N/A	**28.42**	96.79%	**0.13**	0.54	1.55	0.24	0.76
Traffic	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	**0.69**
	STGD-VAE-Dep	**5.13**	N/A	N/A	N/A	N/A	N/A	N/A	1.06



(a) Protein folding

(b) Traffic Modeling

(c) DWR Graph

(d) DRG Graph

Figure 2: Qualitative evaluation on two synthetic datasets and two real-world datasets.

Qualitative Evaluation. As in the conventional qualitative evaluation in disentanglement representation learning [4, 15], 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. 2(a) and 2(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. 2(c) and 2(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.

4 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 several deep generative models validates the superiority of our proposed models from multiple tasks, including graph generation and disentangled representation learning. In the future, we plan to extend this framework to more types of sptiotemporal graphs.

References

[1] M. Barthélemy. Spatial networks. *Physics Reports*, 499(1-3):1–101, 2011.

[2] M. Bradonjić, A. Hagberg, and A. G. Percus. Giant component and connectivity in geographical threshold graphs. In *International Workshop on Algorithms and Models for the Web-Graph*, pages 209–216. Springer, 2007.

[3] C. P. Burgess, I. Higgins, and A. e. a. Pal. Understanding disentangling in β-vae. *arXiv preprint arXiv:1804.03599*, 2018.

[4] R. T. Chen, X. Li, R. B. Grosse, and D. K. Duvenaud. Isolating sources of disentanglement in variational autoencoders. In *NeurIPS*, pages 2610–2620, 2018.

[5] Z. Cui, K. Henrickson, R. Ke, and Y. Wang. 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, 2019.

[6] Y. Du, X. Guo, A. Shehu, and L. Zhao. Interpretable molecule generation via disentanglement learning. In *Proceedings of the 11th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, pages 1–8, 2020.

[7] Y. Du, S. Wang, X. Guo, H. Cao, S. Hu, J. Jiang, A. Varala, A. Angirekula, and L. Zhao. Graphgt: Machine learning datasets for deep graph generation and transformation. 2021.

[8] C. Dye and N. Gay. Modeling the sars epidemic. *Science*, 300(5627):1884–1885, 2003.

[9] X. Guo, Y. Du, and L. Zhao. Property controllable variational autoencoder via invertible mutual dependence. In *International Conference on Learning Representations*, 2020.

[10] X. Guo*, Y. Du*, and L. Zhao. Disentangled deep generative model for spatial networks. *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2021.

[11] X. Guo, S. Tadepalli, L. Zhao, and A. Shehu. Generating tertiary protein structures via an interpretative variational autoencoder. *arXiv preprint arXiv:2004.07119*, 2020.

[12] X. Guo, L. Wu, and L. Zhao. Deep graph translation. *arXiv preprint arXiv:1805.09980*, 2018.

[13] X. Guo, L. Zhao, Z. Qin, L. Wu, A. Shehu, and Y. Ye. Interpretable deep graph generation with node-edge co-disentanglement. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1697–1707, 2020.

[14] A. Hagberg, P. Swart, and D. S Chult. Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2008.

[15] I. Higgins, L. Matthey, and A. e. a. Pal. beta-vae: Learning basic visual concepts with a constrained variational framework. 2016.

[16] J. Ingraham, V. K. Garg, R. Barzilay, and T. S. Jaakkola. Generative models for graph-based protein design. 2021.

[17] H. V. Jagadish, J. Gehrke, A. Labrinidis, Y. Papakonstantinou, J. M. Patel, R. Ramakrishnan, and C. Shahabi. Big data and its technical challenges. *Communications of the ACM*, 57(7):86–94, 2014.

[18] H. Kim and A. Mnih. Disentangling by factorising. *ICML*, 2018.

[19] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[20] T. N. Kipf and M. Welling. Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308*, 2016.

[21] F. Locatello, S. Bauer, M. Lucic, G. Raetsch, S. Gelly, B. Schölkopf, and O. Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *international conference on machine learning*, pages 4114–4124. PMLR, 2019.

[22] O. L. Mangasarian. *Nonlinear programming*. SIAM, 1994.

[23] T. Rahman, Y. Du, L. Zhao, and A. Shehu. Generative adversarial learning of protein tertiary structures. *Molecules*, 26(5):1209, 2021.

[24] M. Simonovsky and N. Komodakis. Graphvae: Towards generation of small graphs using variational autoencoders. In *International Conference on Artificial Neural Networks*, pages 412–422. Springer, 2018.

[25] P. R. Stopher and A. H. Meyburg. Urban transportation modeling and planning. 1975.

[26] P. Teng. A comparison of simulation approaches to epidemic modeling. *Annual Review of Phytopathology*, 23(1):351–379, 1985.

[27] B. M. Waxman. Routing of multipoint connections. *IEEE journal on selected areas in communications*, 6(9):1617–1622, 1988.

[28] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang. Graph wavenet for deep spatial-temporal graph modeling. *arXiv preprint arXiv:1906.00121*, 2019.

[29] K. S. Xu and A. O. Hero. Dynamic stochastic blockmodels for time-evolving social networks. *IEEE Journal of Selected Topics in Signal Processing*, 8(4):552–562, 2014.

[30] J. You, R. Ying, X. Ren, W. Hamilton, and J. Leskovec. Graphrnn: Generating realistic graphs with deep auto-regressive models. In *International Conference on Machine Learning*, pages 5708–5717. PMLR, 2018.

[31] B. Yu, H. Yin, and Z. Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint arXiv:1709.04875*, 2017.

A Spatiotemporal Graph Generative Modeling Objective

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 $\mathbb{E}_{p_\theta(z_{1:T},F)} p_\theta(S_{1:T}, G_{1:T}|z_{1:T}, F)$. The prior distribution of the latent spaces is described as $p(z_{1:T}, F)$ with the observation 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:

$$\begin{aligned}\max_{\theta,\phi} \quad & \mathbb{E}_{S_{1:T},G_{1:T}\sim\mathcal{D}}[\mathbb{E}_{q_\phi(z_{1:T},F|S_{1:T},G_{1:T})} \\ & \qquad\qquad \log p_\theta(S_{1:T}, G_{1:T}|z_{1:T}, F)] \\ \text{s.t.} & D_{KL}(q_\phi(z_{1:T}, F|S_{1:T}, G_{1:T})||p(z_{1:T}, F)) < I \end{aligned} \tag{2}$$

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 A.1.

Lemma A.1. *Given the assumption that: (1) $S_{1:T} \perp G_{1T}|(z_{1:t}, F)$; (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 $z_1 \perp z_2 \cdots \perp z_T$, the objective of spatiotemoral graph generation can be expressed as*

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

Proof. In Lemma A.1, I is decomposed into four mutual-exclusive information capacity, I_s, I_g, I_{sg}, and I_t in Eq. 3. □

A.1 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 3, 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 [3]. 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 $E_{q_\phi(Z|G_{1:T},S_{1:T})} logp_\theta(S_{1:T}|z_{1:T}, f_s, f_{sg})$ 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:

$$
\begin{aligned}
\max_{\theta,\phi} \quad & \mathbb{E}_{S_{1:T},G_{1:T}\sim\mathcal{D}}\mathbb{E}_{q_\phi(z_{1:T},F|S_{1:T},G_{1:T})} \\
& \sum\nolimits_{t=1}^{T}[\log p_\theta(G_t|z_t,f_g,f_{sg}) + \log p_\theta(S_t|z_t,f_s,f_{sg})] \\
\text{s.t.} \quad & \sum\nolimits_{t=1}^{T} D_{KL}(q_\phi(z_t|S_t,G_t)||p(z_t)) < I_t \\
& D_{KL}(q_\phi(f_g|G_{1:T})||p(f_g)) < I_g \\
& D_{KL}(q_\phi(f_s|S_{1:T})||p(f_s)) < I_s \\
& D_{KL}(q_\phi(f_{sg}|S_{1:T},G_{1:T})||p(f_{sg})) < I_{sg} \\
& I_{sg} \leq C_{sg}, \quad I_t \leq C_t.
\end{aligned} \tag{4}
$$

This objective has the properties stated in Theorem A.2.

Theorem A.2. *The objective in Equation 4 guarantees that z_t captures and only captures the time-variant information while f_{sg} captures and only captures the spatial-graph joint information.*

Proof. 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 $I_{sg} \leq C_{sg}$. □

A.2 Spatiotemporal Graph Mutual Information Thresholding Algorithm

Eq. 4 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 [22] as:

$$
\begin{aligned}
\mathcal{R}_1 = & \beta_2(D_{KL}(q_\phi(f_s|S_{1:T})||p(f_s))) \\
& + \beta_3(D_{KL}(q_\phi(f_g|G_{1:T})||p(f_g)))
\end{aligned} \tag{5}
$$

where the Lagrangian multipliers β_2 and β_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

$$\mathcal{R}_2 = \beta_1(\prod\nolimits_{t=1}^{T} D_{KL}(q_\phi(z_t|G_t,S_t)||p(z_t)) - I_t) \tag{6}$$

$$\mathcal{R}_3 = \beta_4(D_{KL}(q_\phi(f_{sg}|S_{1:T},G_{1:T})||p(f_{sg})) - I_{sg}) \tag{7}$$

Finally, we can optimize the overall objective as:

$$
\begin{aligned}
\max_{\theta,\phi} \quad & \mathbb{E}_{S_{1:T},G_{1:T}\sim\mathcal{D}}\mathbb{E}_{q_\phi(z_{1:T},F|S_{1:T},G_{1:T})} \\
& \sum\nolimits_{t=1}^{T}[\log p_\theta(G_t|z_t,f_g,f_{sg}) + \log p_\theta(S_t|z_t,f_{s_t},f_{sg})] \\
& - \mathcal{R}_1 - \mathcal{R}_2 - \mathcal{R}_3 \\
& \text{s.t.} \quad I_{sg} \leq C_{sg}, \quad I_t \leq C_t
\end{aligned} \tag{8}
$$

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 three stages: the first stage is to optimize I_{sg}, the second stage is to optimize I_t, and the final stage to optimize the whole model parameters, as detailed in Algorithm 1. In short, we increase I_{sg} by α 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 γ every J epoch until a stopping criterion is satisfied (Lines 8-14 in Algorithm 1).

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.

Algorithm 1 Information-iterative-thresholding algorithm

```
Input: 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 $I_{sg}$.
Output: The optimized parameter set $\mathcal{W}$.
while $\mathcal{R}_3 < 0$ do {stopping criterion for $I_{sg}$}
    $I_{sg} := I_{sg} + \alpha$
    for epoch $= 1 : K$ do {increase $I_{sg}$ every $K$ epoch}
        Compute the gradient of $\mathcal{W}$ via backpropagation.
        Update $\mathcal{W}$ based on gradient with $I_{sg}$ and $I_t$ fixed.
    end for
end while
while $\mathcal{R}_2 < 0$ do {stopping criterion for $I_t$}
    $I_t := I_t + \gamma$
    for epoch $= 1 : J$ do {increase $I_t$ every $J$ epoch}
        Compute the gradient of $\mathcal{W}$ via backpropagation.
        Update $\mathcal{W}$ based on gradient with $I_t$, $I_s$, $I_g$ and $I_{sg}$ fixed.
    end for
end while
```

Theorem A.3. *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.*

Proof. 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. In practice, we set β_1, β_2, β_3, and β_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) [10], 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 [12] for the graph factors.

B Proof of Theorem 1

Proof. To assist the proof, we introduce four groups of semantic factors. We assume the time-variant information is simulated via one type of semantic factor as $l_{1:T}$, time-invariant spatial-related information is simulated via two types of semantic factors as s^+, s^-, and time-invariant graph-related information is simulated via two parts of semantic factors as g^+, g^- , which follows the convention in the disentangled representation learning domain [4, 15, 18]. The simulation becomes $S = \textbf{Sim}(s^+, s^-, l_{1:T})$ and $G = \textbf{Sim}(g^+, g^-, l_{1:T})$. Here $s^+ \perp s^-$, $g^+ \perp g^-$, $s^+ \perp g^+$, $s^- \not\perp g^-$, $l_{1:T} \perp g^+$, $l_{1:T} \perp g^-$, $l_{1:T} \perp s^+$, and $l_{1:T} \perp s^-$. That is, s^+ and g^+ refers to the time-invariant semantic factors of spatial and graph information, respectively. s^- and g^- refers to the time-invariant correlated semantic factors of spatial and graph information. $l_{1:T}$ refers to the time-variant factors of all time-variant information.

First, the objective can be rewritten based on the information bottleneck theory as:

$$\begin{aligned}
\max_{\theta,\phi} \quad & I(z_{1:T}, f_s, f_{sg}; S_{1:T}) + I(z_{1:T}, f_g, f_{sg}; G_{1:T}) && (9)\\
\text{s.t.} \quad & I(z_{1:T}; S_{1:T}, G_{1:T}) \leq C_t, && (10)\\
& I(f_s; S_{1:T}) \leq I_s, && (11)\\
& I(f_g; G_{1:T}) \leq I_g, && (12)\\
& I(f_{sg}; S_{1:T}) + I(f_{sg}; G_{1:T}) \leq C_{sg} && (13)
\end{aligned}$$

Due to the independence among different latent variables, the objective can be further rewritten to extend the time-invariant parts:

$$\max_{\theta,\phi} \quad I(z_{1:T}; S_{1:T}) + I(z_{1:T}; G_{1:T}) + I(f_s, f_{sg}; S_{1:T}) + I(f_g, f_{sg}; G_{1:T}) \tag{14}$$

It is further extended as:

$$\begin{aligned}
\max_{\theta,\phi} \quad & I(z_{1:T}; s^+) + I(z_{1:T}; s^-) + 2 * I(z_{1:T}; l_{1:T}) + I(z_{1:T}; g^+) + I(z_{1:T}; g^-) + I(f_s, f_{sg}; s^+, s^-)\\
& + I(f_s, f_{sg}; l_{1:T}) + I(f_g, f_{sg}; g^+, g^-) + I(f_g, f_{sg}; l_{1:T}) \qquad (15)
\end{aligned}$$

Since $I(f_s, f_{sg}; l_{1:T}) = 0$, and $I(f_g, f_{sg}; l_{1:T}) = 0$, the time-variant information can not be expressed by the time-invariant latent variables, which are copied for all time frames. Thus, we rewrite the objective and cancel the factor as:

$$\begin{aligned}
\max_{\theta,\phi}(z_{1:T}; s^+) + I(z_{1:T}; s^-) + 2 * I(z_{1:T}; l_{1:T}) + I(z_{1:T}; g^+) + I(z_{1:T}; g^-) + I(f_s, f_{sg}; s^+, s^-)\\
+ I(f_g, f_{sg}; g^+, g^-)\\
\leq I(z_{1:T}; s^+) + I(z_{1:T}; s^-) + I(z_{1:T}; l_{1:T}) + I(z_{1:T}; g^+) + I(z_{1:T}; g^-) + I(f_s, f_{sg}; s^+, s^-)\\
+ I(f_g, f_{sg}; g^+, g^-)
\end{aligned} \tag{16}$$

We rewrite constraints in Eq. 9 to 17, 18 and 19 as:

$$\begin{aligned}
I(z_{1:T}; s^+) + I(z_{1:T}; s^-) + I(f_s, f_{sg}; s^+, s^-) \leq H(s^+, s^-) && (17)\\
I(z_{1:T}; g^+) + I(z_{1:T}; g^-) + I(f_g, f_{sg}; g^+, g^-) \leq H(g^+, g^-) && (18)\\
I(z_{1:T}, s^+) + I(z_{1:T}, s^-) + I(z_{1:T}, g^+) + I(z_{1:T}, g^-) + I(z_{1:T}, l_{1:T}) \leq C_t && (19)
\end{aligned}$$

We add Eq. 17, 18 and 19 together and have:

$$\begin{aligned}
\max_{\theta,\phi} \quad & 2 * I(z_{1:T}, s^+) + 2 * I(z_{1:T}, s^-) + 2 * I(z_{1:T}, g^+) + 2 * I(z_{1:T}, g^-) + I(z_{1:T}, l_{1:T})\\
& + I(f_g, f_{sg}; g^+, g^-) + I(f_s, f_{sg}; s^+, s^-) \leq H(g^+, g^-) + H(s^+, s^-) + C_t \qquad (20)
\end{aligned}$$

Referring to Eq. 16, we have:

$$\begin{aligned}
\max_{\theta,\phi} \quad & I(z_{1:T}, s^+) + I(z_{1:T}, s^-) + I(z_{1:T}, g^+) + I(z_{1:T}, g^-) + I(z_{1:T}, l_{1:T}) + I(f_g, f_{sg}; g^+, g^-)\\
& + I(f_s, f_{sg}; s^+, s^-) \leq H(g^+, g^-) + H(s^+, s^-) + C_t \qquad (21)
\end{aligned}$$

Since $H(g^+, g^-) + H(s^+, s^-) + C_t > 0$, $I(z_{1:T}, g^+) \geq 0$, $I(z_{1:T}, g^+) \geq 0$, $I(z_{1:T}, s^+) \geq 0$, $I(z_{1:T}, s^-) \geq 0$, and we ignore the time-invariant part, $I(f_g, f_{sg}; g^+, g^-) + I(f_s, f_{sg}; s^+, s^-)$ for now, only when $I(z_{1:T}, g^+) + I(z_{1:T}, g^+) + I(z_{1:T}, s^+) + I(z_{1:T}, s^-) = 0$, the left-hand side of the Inequality Eq. 21 reaches the maximum for time-variant variables. Thus, the maximum is achieved only when the time-variant variables $z_{1:T}$ have no correlation with any of the time-invariant variables. Next, we deal with the time-invariant part, the objective function becomes:

$$\max_{\theta,\phi} \quad I(f_g, f_{sg}; g^+, g^-) + I(f_s, f_{sg}; s^+, s^-) + I(z_{1:T}, l_{1:T}) \tag{22}$$

Now, we extend the time-invariant spatial, graph and joint variables as:

$$\begin{aligned} \max_{\theta,\phi} \quad I(f_g, f_{sg}; g^+, g^-) &= I(f_g, g^+) + I(f_g, g^-) + I(f_{sg}, g^+) + I(f_{sg}, g^-) \\ I(f_s, f_{sg}; s^+, s^-) &= I(f_s, s^+) + I(f_s, s^-) + I(f_{sg}, s^+) + I(f_{sg}, s^-) \end{aligned} \tag{23}$$

Since $f_s \perp z_{1:T}$ and $s^- \not\perp\!\!\!\perp g^-$, we have $I(f_s, s^-) = 0$ and $I(f_g, g^-) = 0$, the object can be rewritten as:

$$\max_{\theta,\phi} \quad I(f_g, g^+) + I(f_{sg}, g^+) + I(f_{sg}, g^-) + I(f_s, s^+) + I(f_{sg}, s^+) + I(f_{sg}, s^-) + I(z_{1:T}, l_{1:T}) \tag{24}$$

Since $f_s \perp f_{sg}$ and $f_g \perp f_{sg}$, there is no mutual information between information in s^+ captured by f_s and information in s^+ captured by f_{sg}, the Eq. 9, 10 become:

$$\begin{aligned} I(f_s; s^+) + I(f_{sg}, s^+) \leq I(s^+; s^+) = H(s^+) \\ I(f_g; g^+) + I(f_{sg}, g^+) \leq I(g^+; g^+) = H(g^+) \end{aligned} \tag{25}$$

The constraint in Eq. 13 is equivalent to Eq. 26:

$$I(f_{sg}; s^+) + I(f_{sg}, s^-) + I(f_{sg}; g^+) + I(f_{sg}, g^-) \leq C_{sg} \tag{26}$$

By adding Eq. 25 and Eq. 26, we have:

$$\begin{aligned} I(f_s; s^+) + 2 * I(f_{sg}, s^+) + I(f_{sg}; s^-) + I(f_g, g^+) &+ 2 * I(f_{sg}, g^+) + I(f_{sg}, g^-) \\ &\leq H(s^+) + H(g^+) + C_{sg} \end{aligned} \tag{27}$$

To rewrite it by canceling the factors, we get:

$$\begin{aligned} I(f_s; s^+) + I(f_{sg}; s^+) + I(f_g, g^+) + I(f_g, g^-) &+ I(f_{sg}, s^+) + I(f_{sg}, g^+) \\ &\leq H(s^+) + H(g^+) + C_{sg} \end{aligned} \tag{28}$$

Since $I(z_{sg}; s^+) \geq 0$ and $I(z_{sg_g}; g^+) \geq 0$, and $H(s^+)$, $H(g^+)$ are constants, only when $I(z_{sg}; s^+) = 0$ and $I(z_{sg}; g^+) = 0$, the Inequality Eq. 28 reaches the maximum. Therefore, only when z_{sg} only captures information from spatial-graph joint semantic factors s^+ and g^+, the optimum of the objective is reached.

Overall, only when the time-variant variables $z_{1:T}$ have no correlation with any of the time-invariant variables and z_{sg} only captures information from time-invariant spatial-graph joint semantic factors s^+ and g^+, the optimum of the objective is reached.

□

C Proof of Theorem 2

Proof. First, at the 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}.

Next, when $\mathcal{R}_3 < 0$ and $\mathcal{R}_2 < 0$, namely the information that captured by I_t and I_{sg} do not increase anymore, we can conclude that $I_t = C_t$ and $I_{sg} = C_{sg}$ (proved as follows). 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, the z_t always captures and only captures time-variant information and f_{sg} always captures and only captures time-invariant spatial-graph correlated information.

Here we prove that when the information that captured by I_t and I_{sg} do not increase anymore, it indicates that $I_t = C_t$ and $I_{sg} = C_{sg}$ and we have achieved the optimal objective. To assist the proof, we introduce four groups of semantic factors. We assume the time-variant information is simulated via one type of semantic factor as $l_{1:T}$, time-invariant spatial-related information is simulated via two types of semantic factors as s^+, s^-, and time-invariant graph-related information is simulated via two parts of semantic factors as g^+, g^- , which follows the convention in the disentangled representation learning domain [4, 15, 18]. The simulation becomes $S = \textbf{Sim}(s^+, s^-, l_{1:T})$ and $G = \textbf{Sim}(g^+, g^-, l_{1:T})$. Here $s^+ \perp s^-$, $g^+ \perp g^-$, $s^+ \perp g^+$, $s^- \not\perp\!\!\!\perp g^-$, $l_{1:T} \perp g^+$, $l_{1:T} \perp g^-$, $l_{1:T} \perp s^+$, and $l_{1:T} \perp s^-$. That is, s^+ and g^+ refers to the time-invariant semantic factors of spatial and graph information, respectively. s^- and g^- refers to the correlated semantic factors of spatial and graph information.

(1) Given $z_{1:T}$ captures all the time-variant semantic factors, namely $\prod_{t=1}^{T} D_{KL}(q_\phi(z_t|G_t,S_t)||p(z_t)) = c_t$, the information captured by z_t will not increase anymore, we have $I(z_{1:T}, l_{1:T}) = I(l_{1:T}, l_{1:T})$; Given f_s captures all the time-invariant spatial-independent semantic factors s^+, namely $D_{KL}(q_\phi(f_{sg}|G_{1:T},S_{1:T})||p(f_{sg}))$, the information captured by f_{sg} will not increase anymore, and we have $I(f_s, s^+) = I(s^+, s^+)$; Given f_g captures all the time-invariant graph-independent semantic factors g^+, we have $I(f_g, g^+) = I(g^+, g^+)$; Given f_{sg} captures all the time-invariant spatial-graph correlated semantic factors s^-, g^-, we have $I(f_{sg}, s^+) = 0$, $I(f_{sg}, g^+) = 0$, and $I(f_{sg}, s^-) = I(s^-, s^-)$, $I(f_{sg}, g^-) = I(g^-, g^-)$. Thus, the value of already achieved loss is equal to $I(s^+, s^+) + I(s^-, s^-) + I(g^+, g^+) + I(g^-, g^-) + I(l_{1:T}, l_{1:T})$.
(2) Next, we rewrite the original objective function as follows:

$$\begin{aligned} \max_{\theta,\phi} \quad & I(z_{1:T}; s^+) + I(z_{1:T}; s^-) + 2 * I(z_{1:T}; l_{1:T}) + I(z_{1:T}; g^+) + I(z_{1:T}; g^-) + I(f_s, s^+) \\ & + I(f_{sg}, s^+) + I(f_{sg}, s^-) + I(f_g, g^+) + I(f_{sg}, g^+) + I(f_{sg}, g^-) + I(f_s, l_{1:T}) \\ & + I(f_{sg}, l_{1:T}) + I(f_g, l_{1:T}) \end{aligned} \tag{29}$$

Since $f_s \perp f_{sg}$, $f_g \perp f_{sg}$ and $z_{1:T} \perp f_s$, $z_{1:T} \perp f_g$, $z_{1:T} \perp f_{sg}$, we have:

$$I(f_s, s^+) + I(f_{sg}, s^+) + I(z_{1:T}, s^+) \leq I(s^+, s^+) \tag{30}$$

$$I(f_g, g^+) + I(f_{sg}, g^+) + I(z_{1:T}, g^+) \leq I(g^+, g^+) \tag{31}$$

$$I(f_{sg}, s^-) + I(z_{1:T}, s^-) \leq I(s^-, s^-) \tag{32}$$

$$I(f_{sg}, g^-) + I(z_{1:T}, g^-) \leq I(g^-, g^-) \tag{33}$$

$$I(z_{1:T}, l_{1:T}) \leq I(l_{1:T}, l_{1:T}) \tag{34}$$

Thus, the optimal loss is $I(s^+, s^+) + I(s^-, s^-) + I(g^+, g^+) + I(g^-, g^-) + I(l_{1:T}, l_{1:T})$. In this situation, the optimal loss is already achieved. If we continue increase I_t and I_{sg}, the information captured by z_t and f_{sg} will not increase anymore, thus we got $\mathcal{R}_2 < 0$ and $\mathcal{R}_3 < 0$. which is a signal for stopping the optimization process.

□

D Time-dependent Objective Function

As demonstrated in section 2, the objective leads to the maximization problem below:

$$\max_{\theta,\phi} \quad \mathbb{E}_{S_{1:T},G_{1:T}\sim D}[\mathbb{E}_{q_\phi(z_{1:T},F|S_{1:T},G_{1:T})} log p_\theta(S_{1:T},G_{1:T}|z_{1:T},f_s,f_g,f_{sg})] \tag{35}$$
$$\text{s.t.} \quad D_{KL}(q_\phi(z_{1:T},F|S_{1:T},G_{1:T})||p(z_{1:T},f_s,f_g,f_{sg}) < \epsilon$$

where D refers to the observed dataset of the spatial network dynamics.This is equal to maximize the evidence lower-bound (ELBO), as follows:

$$\max_{\theta,\phi} \quad \mathbb{E}_{S_{1:T},G_{1:T}\sim D}[\mathbb{E}_{q_\phi(z_{1:T},F|S_{1:T},G_{1:T})} log p_\theta(S_{1:T},G_{1:T}|z_{1:T},f_s,f_g,f_{sg})] \tag{36}$$
$$-\lambda D_{KL}(q_\phi(z_{1:T},f_s,f_g,f_{sg}|S_{1:T},G_{1:T})||p(z_{1:T},f_s,f_g,f_{sg})$$

First, we decompose the main objective term based on the assumption that $S_{1:T} \perp G_{1:T}|(z_{1:T},f_s,f_g,f_{sg})$:

$$\mathbb{E}_{q_\phi(z_{1:T},F|S_{1:T},G_{1:T})}[\log p_\theta(G_{1:T}|z_{1:T},f_s,f_g,f_{sg}) + \log p_\theta(S_{1:T}|z_{1:T},f_s,f_g,f_{sg})] \tag{37}$$

According to the assumption $S_{1:T} \perp f_g$ and $G_{1:T} \perp f_s$, we have:

$$\mathbb{E}_{q_\phi(z_{1:T},F|S_{1:T},G_{1:T})}[\log p_\theta(G_{1:T}|z_{1:T},f_g,f_{sg}) + \log p_\theta(S_{1:T}|z_{1:T},f_s,f_{sg})] \tag{38}$$

Then, we extend the objective through the time dimension as:

$$\mathbb{E}_{q_\phi(z_{1:T},F|S_{1:T},G_{1:T})}[\log p_\theta(G_1,G_2,\cdots,G_T|z_1,z_2,\cdots,z_T,f_g,f_{sg})$$
$$+ \log p_\theta(S_1,S_2,\cdots,S_T|z_1,z_2,\cdots,z_T,f_s,f_{sg})] \tag{39}$$

Since the sequence is dependent through time, $z_1 \not\perp\!\!\!\perp z_2 \cdots \not\perp\!\!\!\perp z_T$, we simplify the objective as:

$$\mathbb{E}_{q_\phi(z_{1:T},F)|S_{1:T},G_{1:T})}[\log p_\theta(G_1|z_1,f_g,f_{sg})p_\theta(z_2|z_1)\cdots p_\theta(z_T|z_{T-1})p_\theta(G_T|z_T,f_g,f_{sg})$$
$$+ \log p_\theta(S_1|z_1,f_s,f_{sg})p_\theta(z_2|z_1)\cdots p_\theta(z_T|z_{T-1})p_\theta(S_T|z_T,f_s,f_{sg})] \tag{40}$$

We re-organize the objective by putting the product together as:

$$\mathbb{E}_{q_\phi(z_{1:T},F)|S_{1:T},G_{1:T})}[\log \prod_{t=1}^{T} p_\theta(G_t|z_t,f_g,f_{sg}) \prod_{t=2}^{T} p_\theta(z_t|z_{t-1})$$
$$+ \log \prod_{t=1}^{T} p_\theta(S_t|z_t,f_s,f_{sg}) \prod_{t=2}^{T} p_\theta(z_t|z_{t-1})] \tag{41}$$

Finally, the objective is written as the following by taking the product out of log function as summation:

$$\mathbb{E}_{q_\phi(z_{1:T},F)|S_{1:T},G_{1:T})}[\sum_{t=1}^{T} \log p_\theta(G_t|z_t,f_g,f_{sg}) + \sum_{t=2}^{T} \log p_\theta(z_t|z_{t-1}) + \sum_{t=1}^{T} \log p_\theta(S_t|z_t,f_s,f_{sg})$$
$$+ \log \sum_{t=2}^{T} p_\theta(z_t|z_{t-1})] \tag{42}$$

Next, we extend the encoder part $q_\phi(z_{1:T},F|S_{1:T},G_{1:T})$ F to f_s, f_g, and f_{sg}, and decompose it based on the assumption that $p(z_{1:T})$ and $p(f_s)$, $p(f_g)$, and $p(f_{sg})$ are independent given $S_{1:T}$ and $G_{1:T}$ as:

$$q_\phi(z_{1:T},f_s,f_g,f_{sg}|S_{1:T},G_{1:T}) = q_\phi(z_{1:T}|S_{1:T},G_{1:T})q_\phi(f_s|S_{1:T})p_\phi(f_g|G_{1:T})q_\phi(f_{sg}|S_{1:T},G_{1:T}) \tag{43}$$
$$= q_\phi(z_1|G_1,S_1)q_\phi(z_2|z_1,G_2,S_2)q_\phi(z_3|z_2,G_3,S_3)\cdots q_\phi(z_T|z_{T-1},S_T,G_T)q_\phi(f_s|S_{1:T})q_\phi(f_g|G_{1:T})q_\phi(f_{sg}|S_{1:T},G_{1:T})$$
$$= \prod_{t=2}^{T} q_\phi(z_t|z_{t-1},S_t,G_t)q_\phi(z_1|S_1,G_1) \prod_{t=1}^{T} q_\phi(f_s|S_t)p_\phi(f_g|G_t)q_\phi(f_{sg}|S_t,G_t)$$

Then the objective is written as:

$$\begin{aligned}
\max_{\theta,\phi} \quad & \mathbb{E}_{S_{1:T},G_{1:T}\sim D}\mathbb{E}_{q_\phi(z_{1:T},F|S_{1:T},G_{1:T})}[\sum_{t=1}^{T}\log p_\theta(G_t|z_t,f_g,f_{sg}) + \sum_{t=2}^{T}\log p_\theta(z_t|z_{t-1}) \\
& + \sum_{t=1}^{T}\log p_\theta(S_t|z_t,f_s,f_{sg}) + \log\sum_{t=2}^{T}p_\theta(z_t|z_{t-1})] \\
\text{s.t.} \quad & D_{KL}(\prod_{t=2}^{T}q_\phi(z_t|z_{t-1},S_t,G_t)q_\phi(z_1|S_1,G_1)||p(z_{1:T})) < I_t. \\
& D_{KL}(q_\phi(f_g|G_{1:T})||p(f_g)) < I_g. \\
& D_{KL}(q_\phi(f_s|S_{1:T})||p(f_s)) < I_s \\
& D_{KL}(q_\phi(f_{sg}|S_{1:T},G_{1:T})||p(f_{sg})) < I_{sg}
\end{aligned} \tag{44}$$

Next, we focus on the first constrain on time. Since z_t are dependent on each other, we assume a conditional prior distribution of z_t instead of Normal distribution. That is, $p(z_t|z_{t-1}) \sim \mathcal{N}(\mu(z_{t-1});\sigma(z_{t-1}))$. Here $\mu(\cdot)$ and $\sigma(\cdot)$ can be implemented by any functions like neural networks. And $p(z_1) \sim \mathcal{N}(0;1)$. Thus, we further decompose the first constrain as:

where $D_{KL}(\prod_{t=2}^{T}q_\phi(z_t|z_{t-1},S_T,G_T)q_\phi(z_1|S_1,G_1)||p(z_{1:T}))$ is derived by:

$$\begin{aligned}
& D_{KL}(\prod_{t=2}^{T}q_\phi(z_t|z_{t-1},S_T,G_T)q_\phi(z_1|S_1,G_1)||p(z_{1:T})) \qquad (45) \\
& = D_{KL}(q_\phi(z_1|S_1,G_1),q_\phi(z_2|z_1,S_2,G_2),\cdots,q_\phi(z_t|z_{t-1},S_t,G_t)||p(z_1),p(z_1|z_2),\cdots,p(z_{t-1}|z_t)) \\
& = \sum_z q_\phi(z_1|S_1,G_1)q_\phi(z_2|z_1,S_2,G_2)\cdots q_\phi(z_t|z_{t-1},S_t,G_t)\log\frac{q_\phi(z_1|S_1,G_1)q_\phi(z_2|z_1,S_2,G_2)\cdots q_\phi(z_t|z_{t-1},S_t,G_t)}{p(z_1)p(z_1|z_2)\cdots p(z_{t-1}|z_t)} \\
& = E[\log\frac{q_\phi(z_1|S_1,G_1)q_\phi(z_2|z_1,S_2,G_2)\cdots q_\phi(z_t|z_{t-1},S_t,G_t)}{p(z_1)p(z_1|z_2)\cdots p(z_{t-1}|z_t)}] \\
& = E[\log q_\phi(z_1|S_1,G_1)q_\phi(z_2|z_1,S_2,G_2)\cdots q_\phi(z_t|z_{t-1},S_t,G_t) - \log p(z_1)p(z_1|z_2)\cdots p(z_{t-1}|z_t)] \\
& = E[\log q_\phi(z_1|S_1,G_1)q_\phi(z_2|z_1,S_2,G_2)\cdots q_\phi(z_T|z_{T-1},S_T,G_T) - \log p(z_1)p(z_2|z_1)\cdots p(z_T|z_{T-1}))] \\
& = E[\log\prod_{t=2}^{T}q_\phi(z_t|z_{t-1},S_t,G_t)q_\phi(z_1|,S_1,G_1) - \log\prod_{t=2}^{T}p(z_t|z_{t-1})p(z_1)] \\
& = E[\sum_{t=2}^{T}\log q_\phi(z_t|z_{t-1},S_t,G_t) + \log q_\phi(z_1|S_1,G_1) - \sum_{t=2}^{T}\log p(z_t|z_{t-1}) + \log p(z_1)] \\
& = \sum_{t=2}^{T}E[\log\frac{q_\phi(z_t|z_{t-1},S_t,G_t)}{p(z_t|z_{t-1})} + \log\frac{q_\phi(z_1|,S_1,G_1)}{p(z_1)}] \\
& = \sum_{t=2}^{T}D_{KL}(q_\phi(z_t|z_{t-1},S_t,G_t)||p(z_t|z_{t-1})) + D_{KL}(q_\phi(z_1|S_t,G_t)||p(z_1))
\end{aligned}$$

E Architectures & Hyperparameters

Our model consists of four components which takes care of four types of latent variables f_s, f_g, f_{sg}, and z_t. To accommodate the disentangled information in each latent space and variance through time, we have four encoders, which models $q_\phi(f_s|S_{1:T},G_{1:T})$, $q_\phi(f_g|S_{1:T},G_{1:T})$, $q_\phi(f_{sg}|S_{1:T},G_{1:T})$, and $q_\phi(z_t|S_{1:T},G_{1:T})$, respectively. Each encoder learns a unique mean and standard deviation and each latent variable is randomly sampled from the Gaussian distribution, respectively. To encode the time-invariant spatial information via modelling $q_\phi(f_s|S_{1:T},G_{1:T})$, we implement a 1D convolution neural network. To encode the time-invariant graph information via modelling $q_\phi(f_g|S_{1:T},G_{1:T})$, we implement a typical graph convolution neural network [19]. To encode the time-invariant

Table 2: Encoders architectures (Each layers is expressed in the format of <*filter_size*><*layer type*><*Num_channel*><*Activation function*><*stride size*>. *FC* refers to the fully connected layers). *c-deconv* and *c-conv* refers to the cross edge deconvolution and convolution respectively. The activation functions after each layer are all ReLU except the last layers.

Spatial Encoder	Joint Encoder	Graph Encoder	Time Encoder
Input: $L \in \mathbb{R}^{25\times 2}$	Input: E, L	Input: $E \in \mathbb{R}^{25\times 25}$,$F \in \mathbb{R}^{25}$	Input: E, L
5 conv1D.10. stride 1	S-MPNN.20	GCN.10	S-MPNN.20
5 conv1D.10. stride 1	S-MPNN.50	GCN.20	S-MPNN.50
5 conv1D.20. stride 1	FC.200.	FC.100.	FC.200.
FC.100.	FC.200	FC.100	FC.200
FC.100			

Table 3: Decoders architectures (Each layers is expressed in the format as <*filter_size*><*layer type*><*Num_channel*><*Activation function*><*stride size*>. *FC* refers to the fully connected layers). *c-deconv* and *c-conv* refers to the cross edge deconvolution and convolution respectively. The activation functions after each layer are all ReLU except the last layers.

Graph Decoder(for edge)	Graph Decoder(for node)	Spatial Decoder
Input:$f_g \in \mathbb{R}^{100}$,$f_{sg} \in \mathbb{R}^{200}$,$z_t \in \mathbb{R}^{200}$	Input:$f_g \in \mathbb{R}^{100}$,$f_{sg} \in \mathbb{R}^{200}$,$z_t \in \mathbb{R}^{200}$	Input:$f_s \in \mathbb{R}^{100}$,$f_{sg} \in \mathbb{R}^{200}$,$z_t \in \mathbb{R}^{200}$
FC.500	FC.500	FC.500
5 conv1D.50. stride 1	5 conv1D.50. stride 1	5 conv1D.50. stride 1
5 × 5 deconv.20. stride 1	5 conv1D.20. stride 1	5 conv1D.20. stride 1
FC.1	FC.1	5 conv1D.10. stride 1
		FC.2

spatial-graph correlated information via modelling $q_\phi(f_{sg}|S_{1:T}, G_{1:T})$, we implement a spatial-graph convolution neural network [10]. To encode time-variant information, we implement another spatial-graph convolution neural network to capture the variance among different timesteps. With regard to decoders, we implement two decoders which one decodes the spatial information and the other one decodes the graph topological information. The input to the decoders are the concatenation of the latent presentation f_s, f_g, f_{sg} and z_t. For example, to decode the graph topological information which includes nodes and edges, the input of the graph decoder is the concatenation of f_g, f_{sg}, and z_t. Similarly, to decode the spatial information, the input of the spatial decoder is the concatenation of f_s, f_{sg}, and z_t. To construct the edge feature or adjacency matrix, the input vector is mapped into a node-level feature vector through a fully connected layer first. Then, a matrix is constructed by replicating the vector. The edge's hidden representation matrix is constructed from the latent representation by a node-to-edge deconvolution layer [12] which decodes each node-level representation by making sense of the contributions from each node to its related edge's hidden representation. Finally, the edge feature or adjacency matrix is constructed by an edge-edge deconvolution layer, which each hideden edge feature contributes to the generation of its adjacent edges. The spatial decoder is typical a set of 1D convolution layers. Similarly, the node features of the graphs are also generated by a set of typical 1D convolution layers. The detailed hyperparameters for encoders and decoders of our models are shown in Table 2 and Table 3, respectively.

F Model Complexity Analysis

The proposed STND-VAE requires $O(N^2)$ time complexity for spatial-graph convolution, $O(N)$ time complexity for spatial convolution and $O(N)$ time complexity for typical graph convolution with respect to number of N nodes in the graphs. In terms of encoders for time-variant features, our model amounts to $O(N^2)$ time complexity. In total, our model takes $O(N^2)$ time complexity, which is scalable compared to most of the existing graph generation models. For example, graphVAE [24] amounts $O(N^4)$ time consumption in the worst case and graphRNN amounts $O(N^2)$ time complexity.

G Dataset

Dynamic Waxman Random Graphs. The dynamic Waxman random graphs are generated by uniformly placing n nodes at random in a rectangular domain [27] through a time sequence t. First, the graph edge connection is modeled by pairwise distance d between any two nodes with an edge probability of $\beta e^{-d/\alpha L}$, where L is the maximum distance between any pair of nodes, and α, β are predefined parameters[1]. Second, the spatial locations of the nodes are uniformly generated within the rectangular domain, where the coordinates of the four vertexes of the rectangular domain for generating the spatial locations of the nodes are (p, p), $(p, p + n \times s)$, $(p + n \times s, p)$, $(p + n \times s, p + n \times s)$, respectively. Here the absolute position of the graphs is ranged from 1 to 11, and the location density of the nodes is ranged from 4 to 11. The node attribute, i.e. node color, is sampled from a random Gaussian distribution with mean b ranging from 1 to 11. Finally, the temporal attributes are modeled by multiplying a time factor associated with a node attribute, i.e. node size. In the end, we have four types of latent factors corresponding to semantic factors in the dynamic Waxman random graphs dataset, which the graph-exclusive factor b controlling node color semantic factor, the spatial-exclusive factor p controlling the node spatial location semantic factor, the spatial-graph correlated factor s controlling graph and spatial density, and the time-variant factor t controlling node size varying through the sequence. In total, we have 2500 sequences of length 8 for training and 500 sequences of length 8 for testing.

Dynamic Random Geometry Graphs. The dynamic random geometry graphs are generated by uniformly placing n nodes at random in a rectangular domain [2] through a time sequence t. First, the graph topology is modeled by pairwise distance d larger than a threshold θ between any two nodes with an edge[2]. Similarly, the spatial locations and temporal attributes are generated in the same way as in the dynamic Waxman random graphs. Finally, we have four types of latent factors corresponding to semantic factors in the dynamic random geometry graphs dataset, which the graph-exclusive factor b controlling node color semantic factor, the spatial-exclusive factor p controlling the node spatial location semantic factor, the spatial-graph correlated factor s controlling graph and spatial density, and the time-variant factor t controlling node size varying through the sequence. In total, we have 2500 sequences of length 8 for training and 1000 sequences of length 8 for testing.

Protein Folding Dataset. Protein structures are naturally spatial graphs, which each node represents an amino acid with a spatial location and edge represents contacts between two amino acids (d < 8 Å). The protein folding dataset includes a series of protein structures representing the folding process of a protein sequence $AGAAAAGA$ of length 8. In the protein folding dataset [13], the graph density (reflected by the density of the inter-residue contacts) and the folding degrees (reflected by spatial locations of amino acids) of protein are spatial-graph correlated factors. The folding phrase ranging from 1 to 1000 represents temporal attributes during the folding process. In total, we have 4750 of length 8 for training and 4750 sequences of length 8 for testing.

Traffic Dataset MERT-LA. Traffic data is one of the most common data type to study spatiotemporal graphs. MERT-LA [17] is collected by Los Angeles Metropolitan Transportation Authority(LA-Metro), and processed by University of Southern California's Integrated Media Systems Center. This dataset contains traffic information collected from loop detectors in the highway of Los Angeles County by 207 sensors for four continuous months (from Mar 1-st 2012 to Jun 30-th 2012). We set one time step as one hour and total number of time steps in one sequence as 4. For the total of 714 samples, 500 samples are used for training and 214 samples are used for testing.

[1] The default parameters in Networkx [14] package are β as 0.4 and α as 0.1.

[2] θ parameter is set as 12 in our experiment