Exploring Social Posterior Collapse in Variational Autoencoder for Interaction Modeling

Chen Tang
Department of Mechanical Engineering
University of California Berkeley
Berkeley, CA
chen_tang@berkeley.edu

Wei Zhan
Department of Mechanical Engineering
University of California Berkeley
Berkeley, CA
wzhan@berkeley.edu

Masayoshi Tomizuka
Department of Mechanical Engineering
University of California Berkeley
Berkeley, CA
tomizuka@berkeley.edu

Abstract

Multi-agent behavior modeling and trajectory forecasting are crucial for the safe navigation of autonomous agents in interactive scenarios. Variational Autoencoder (VAE) has been widely applied in multi-agent interaction modeling to generate diverse behavior and learn a low-dimensional representation for interacting systems. However, existing literature did not formally discuss if a VAE-based model can properly encode interaction into its latent space. In this work, we argue that one of the typical formulations of VAEs in multi-agent modeling suffers from an issue we refer to as social posterior collapse, i.e., the model is prone to ignoring historical social context when predicting the future trajectory of an agent. It could cause significant prediction errors and poor generalization performance. We analyze the reason behind this under-explored phenomenon and propose several measures to tackle it. Afterward, we implement the proposed framework and experiment on real-world datasets for multi-agent trajectory prediction. In particular, we propose a novel sparse graph attention message-passing (sparse-GAMP) layer, which helps us detect social posterior collapse in our experiments. In the experiments, we verify that social posterior collapse indeed occurs. Also, the proposed measures are effective in alleviating the issue. As a result, the model attains better generalization performance when historical social context is informative for prediction.

1 Introduction

Modeling the behavior of intelligent agents is an essential subject for autonomous systems. Safe operations of autonomous agents require accurate prediction of other agents' future motions. In particular, social interaction among agents should be modeled, so that downstream planning and decision-making modules can safely navigate the robots through interactive scenarios. Generative latent variable models are popular modeling options for their ability to generate diverse and naturalistic behavior [1, 2, 3, 4]. In this work, we focus on one category of generative models, Variational Autoencoder (VAE) [5], which has been widely used in multi-agent behavior modeling and trajectory prediction [1, 4, 6, 7, 8, 9]. It is desirable as it learns a low-dimensional representation of the original high-dimensional data.

35th Conference on Neural Information Processing Systems (NeurIPS 2021)

However, VAEs do not necessarily learn a good representation of the data [10]. For instance, prior works in sequence modeling have found out that the model tends to ignore the latent variables if the decoder is overly powerful [11, 12, 13] (e.g., an autoregressive decoder). It leads us to wonder whether a VAE-based model can always learn a good representation for a multi-agent interacting system. It is a rather general question as researchers may look for different properties of the latent space, for instance, interpretability [14, 15] and multi-modality [4, 6]. In this work, we focus on a fundamental aspect of this general question: Does the latent space always properly model interaction? Formally, given a latent variable model of an interacting system, where a latent variable governs each agent's behavior, we wonder if the VAE learns to encode social context into the latent variables.

It raises such concern since VAE handles two distinct tasks in training and testing. At the training stage, it learns to reconstruct a datum instead of generating one. For multi-agent interaction modeling, the sample for reconstruction is a set of trajectories of all the agents. Ideally, the model should learn to model the interaction among agents and jointly reconstruct the trajectories. However, there is no mechanism to prevent the model from separately reconstructing the trajectories. In fact, it could even be more efficient at the early stage of training when the model has not learned an informative embedding of social context. We then end up with a model that models each agent's behavior separately without social context, which might suffer from over-estimated variance and large prediction error. More importantly, since the joint behavior of the agents is a consequence of their interactions, ignoring the causes may lead to poor generalization ability [15, 16]. We find that a typical formulation of VAE for multi-agent interaction is indeed prone to ignoring historical social context (i.e., interactions over the observed time horizon). We refer to this phenomenon as social posterior collapse. The issue has never been discussed in the literature. Considering those potential defects, we think it is necessary to study such a crucial and fundamental issue.

In this work, we first abstract the VAE formulation from a wide range of existing literature [1, 4, 6, 8]. Then we analyze social posterior collapse under this formulation and propose several measures to alleviate the issue. Afterward, we study the issue under real-world settings with a realization of the abstract formulation we design. In particular, we propose a novel sparse graph attention message-passing (sparse-GAMP) layer and incorporate it into the model, which helps us detect and analyze social posterior collapse. Our experiments show that social posterior collapse occurs in real-world prediction tasks and that the proposed measures can effectively alleviate the issue. We also evaluate how social posterior collapse affects the model performance on these tasks. The results suggest that the model without social posterior collapse can attain better generalization performance if historical social context is informative for prediction.

2 Social Posterior Collapse in Variational Autoencoder Interaction Models

2.1 A Latent Variable Model for Interaction Modeling

Given an interacting system with n agents, an interaction-aware trajectory prediction model takes all the agents' observed historical trajectories, denoted by $\{\mathbf{x}_i\}_{i=1}^n$, as input and predicts the future trajectories of all the agents or a subset of enquired agents. We denote the collection of future trajectories by $\{\mathbf{y}_i\}_{i=1}^n$. In this work, we focus on the latent variable model illustrated in Fig. 1, which is abstracted from existing literature on VAEs for multi-agent behavior modeling [1, 4, 6, 8]. We model interaction by introducing a set of variables $\{\mathbf{T}_i\}_{i=1}^n$, which aggregates each agent's state and its observation of other agents. Formally, each $\mathbf{T}_i$ is modeled as a deterministic function of $\{\mathbf{x}_i\}_{i=1}^n$, i.e., $\mathbf{T}_i = f_i\left(\{\mathbf{x}_i\}_{i=1}^n\right)$. Afterward, the agents make decisions based on the aggregated information over the predicted horizon. Latent variables $\{\mathbf{z}_i\}_{i=1}^n$ are introduced to model the inherent uncertainty in each agent's behavior. It should be noticed that interaction over the predicted horizon is not modeled explicitly in this formulation. Although it can be achieved by exchanging information between agents recurrently (e.g., social pooling in [17]), it is a common practice to avoid explicit modeling of future interaction in consideration of computational cost and memory [3, 8, 18].

In this work, we train the model as a VAE, where an encoder $q\left(\mathbf{z}|\mathbf{x}, \mathbf{y}\right)$ is introduced to approximate the posterior distribution $p\left(\mathbf{z}|\mathbf{x}, \mathbf{y}\right)$ for efficient sampling at the training stage[1]. The performance of variational inference is optimized if the KL-divergence between the posteriors, i.e. $D_{KL}\left[q\left(\mathbf{z}|\mathbf{x}, \mathbf{y}\right) \| p\left(\mathbf{z}|\mathbf{x}, \mathbf{y}\right)\right]$, is small [10]. To derive a better approximation, we can incorporate

[1]The vectors $\mathbf{x}, \mathbf{y}$ and $\mathbf{z}$ collect the corresponding variables for all n agents.

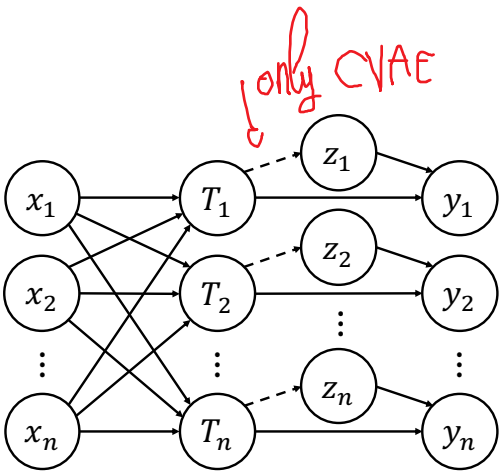

Figure 1: A latent variable model for interaction modeling. The dash edges from T_i to z_i apply only in the Conditional Variational Autoencoder (CVAE) setting.

inductive bias based on the characteristics of the true posterior into the encoder function. We introduce the following simple proposition that guides our model design. Its proof can be found in Appx. A.

Proposition 1. *For any $i = 1, 2, ..., n$ and $j = 1, 2, ..., n$, 1) If $j \neq i$, $\mathbf{z}_i$ and $\mathbf{y}_j$ are conditionally independent given $\mathbf{x}$ and $\mathbf{y}_i$; 2) If $j \neq i$, $\mathbf{z}_i$ and $\mathbf{z}_j$ are conditionally independent given $\mathbf{x}$; 3) $\mathbf{z}_i$ and $\mathbf{x}_j$ are conditionally independent given $\mathbf{T}_i$.*

Following the proposition, we decompose the posterior distribution as $\prod_{i=1}^{n} p(\mathbf{z}_i|\mathbf{T}_i, \mathbf{y}_i)$. The decomposition suggests two insights. First, the encoder does not need to aggregate future context information when inferring the posterior distribution of $\mathbf{z}_i$ for each agent. Second, the historical context variable $\mathbf{T}_i$ is all we need for the historical information of the agent i. The encoder and decoder can share the same function to encode historical information.

2.2 Social Posterior Collapse

We then design a VAE model reflecting the characteristics of the true posterior discovered in Sec. 2.1. To simplify the problem, we further make the assumption of homogeneous agents. The model has three basic building blocks: 1) A function modeling the historical context, i.e., $\mathbf{T}_i = f_\theta(\mathbf{x}_i, \mathbf{x})$; 2) A function decoding the distribution of the future trajectory $\mathbf{y}_i$ given $\mathbf{T}_i$ and $\mathbf{z}_i$, i.e., $p_\phi(\mathbf{y}_i|\mathbf{T}_i, \mathbf{z}_i)$; 3) A function approximating the posterior of $\mathbf{z}_i$ conditioned on $\mathbf{T}_i$ and $\mathbf{y}_i$, i.e., $q_\psi(\mathbf{z}_i|\mathbf{T}_i, \mathbf{y}_i)$. They build up the encoder and decoder of the VAE model as follows:

$$q_{\theta,\psi}(\mathbf{z}|\mathbf{x}, \mathbf{y}) = \prod_{i=1}^{n} q_\psi(\mathbf{z}_i|f_\theta(\mathbf{x}_i, \mathbf{x}_{-i}), \mathbf{y}_i), \qquad p_{\theta,\phi}(\mathbf{y}|\mathbf{x}, \mathbf{z}) = \prod_{i=1}^{n} p_\phi(\mathbf{y}_i|f_\theta(\mathbf{x}_i, \mathbf{x}_{-i}), \mathbf{z}_i). \tag{1}$$

We consider a continuous latent space and model q_ψ and p_ϕ as diagonal Gaussian distributions. The model is trained by maximizing the evidence lower bound (ELBO):

$$\max_{\theta,\psi,\phi} \mathbb{E}_{\mathbf{x},\mathbf{y}\sim D}\left[\mathbb{E}_{\mathbf{z}\sim q_{\theta,\psi}(\mathbf{z}|\mathbf{x},\mathbf{y})}\left[\log p_{\theta,\phi}(\mathbf{y}|\mathbf{x}, \mathbf{z})\right] - \beta D_{KL}\left[q_{\theta,\psi}(\mathbf{z}|\mathbf{x}, \mathbf{y}) \,\|p(\mathbf{z})\right]\right]. \tag{2}$$

However, we find a critical issue of this naive formulation during experiments, which is the social posterior collapse phenomenon mentioned before. With the help of the sparse graph attention mechanism introduced in Sec. 3, we find that the model is prone to ignoring historical social context when reconstructing the future trajectory of one agent. Equivalently, the generative model collapses to the one with all the variables of other agents marginalized out.

We think the reason behind it is similar to, but different from, the well-known phenomenon in VAE training—posterior collapse [10]. Due to the KL regularization term in ELBO, the variational posterior distribution is likely to collapse towards the prior. It is particularly likely to occur at the early stage of training when the latent space is still uninformative [19]. In our case, the posterior of $\mathbf{z}_i$ collapses into $q_{\theta,\psi}(\mathbf{z}_i|\mathbf{x}_i, \mathbf{y}_i)$. It does minimize the KL-divergence: if both $q_{\theta,\psi}(\mathbf{z}_i|\mathbf{x}_i, \mathbf{y}_i)$ and $q_{\theta,\psi}(\mathbf{z}_i|\mathbf{x}, \mathbf{y}_i)$ exactly approximate the true posteriors, then conditioning on more context information

increases the expected value of KL regularization:

$$\mathbb{E}_D\left[D_{KL}\left[p\left(\mathbf{z}_i|\mathbf{x},\mathbf{y}_i\right)\|p(\mathbf{z}_i)\right]\right] = I(\mathbf{z}_i;\mathbf{x},\mathbf{y}_i) \geqslant I(\mathbf{z}_i;\mathbf{x}_i,\mathbf{y}_i) = \mathbb{E}_D\left[D_{KL}\left[p\left(\mathbf{z}_i|\mathbf{x}_i,\mathbf{y}_i\right)\|p(\mathbf{z}_i)\right]\right].$$

However, we argue that an additional factor contributes to the social posterior collapse problem, which makes it a unique phenomenon for interaction modeling. At the training stage, the goal is to reconstruct the future trajectories $\mathbf{y}$. The trajectory itself contains all the information needed for reconstruction. The history of the agent i provides complementary information such as the current state and static environmental information. There is no explicit regulation in the current framework to prevent the model from extracting information solely from $\mathbf{x}_i$ and $\mathbf{y}_i$. In fact, it is a more efficient coding scheme when the model has not learned an informative representation of interaction. Techniques such as KL annealing [13, 19] rely on various scheduling schemes of β to prevent KL vanishing, which has been shown effective in mitigating the posterior collapse problem. However, reducing β could merely privilege the model to gather more information from $\mathbf{y}_i$, which is consistent with the reconstruction objective. Therefore, we need to explore alternative solutions to tackle the social posterior collapse problem deliberately.

We start with changing the model into a conditional generative one [20, 21]. Additional edges $\{\mathbf{T}_i \rightarrow \mathbf{z}_i\}_{i=1}^n$ are added into the original graph, which are annotated with dash lines in Fig. 1. We follow the practice in [20] and formulate the model as a Conditional Variational Autoencoder (CVAE). It is straightforward to verify that the conclusion of Proposition 2.1 still applies. We still model the encoder and decoder as in Eqn. (1). The CVAE framework introduces an additional module —a function approximating the conditional prior $p_\eta\left(\mathbf{z}_i|\mathbf{T}_i\right)$, which becomes $p_{\theta,\eta}\left(\mathbf{z}_i|\mathbf{x}\right)$ after incorporating $f_\theta(\mathbf{x})$. The objective then becomes:

$$\mathcal{L}(\theta,\psi,\phi,\eta) = \mathbb{E}_{\mathbf{x},\mathbf{y}\sim D}\left[\mathbb{E}_{\mathbf{z}\sim q_{\theta,\psi}(\mathbf{z}|\mathbf{x},\mathbf{y})}\left[\log p_{\theta,\phi}\left(\mathbf{y}|\mathbf{x},\mathbf{z}\right)\right] - \beta D_{KL}\left[q_{\theta,\psi}\left(\mathbf{z}|\mathbf{x},\mathbf{y}\right)\|p_{\theta,\eta}(\mathbf{z}|\mathbf{x})\right]\right].$$

Compared to Eqn. (2), we no longer penalize the encoder for aggregating context information but only restrict the information encoded from future trajectories. Therefore, the encoder does not need to ignore context information to fulfill the information bottleneck. However, the model still lacks a mechanism to encourage context information encoding deliberately. To achieve the goal, we propose to incorporate an auxiliary prediction task into the training scheme. Concretely, we introduce another module $\mathbf{y}_i = g_\zeta\left(\mathbf{T}_i,\mathbf{z}_i\right)$. Composing g_ζ with f_θ gives us another decoder $\mathbf{y} = h_{\theta,\zeta}(\mathbf{x},\mathbf{z})$. The difference is that the latent variables are always sampled from $p_\eta\left(\mathbf{z}_i|\mathbf{T}_i\right)$. The auxiliary task is training this trajectory decoder which shares the same context encoder and conditional prior with the CVAE. Because the auxiliary model does not have access to the ground-truth future trajectory, it needs to utilize context information for accurate prediction. Consequently, it encourages the model to encode context information into $\mathbf{T}$ and $\mathbf{z}$. We use mean squared error (MSE) loss as the objective function. The overall objective minimizes a weighted sum of the two objective functions.

The training scheme looks similar to the one in [20], where they trained a Gaussian stochastic neural network (GSNN) together with the CVAE model. However, ours is different from theirs in some major aspects. The GSNN model shares the same decoder with the CVAE model. The primary motivation of incorporating another learning task is to optimize the generation procedure during training directly. In contrast, our auxiliary prediction model has a separate trajectory decoder. It is because we do not want the auxiliary task to interfere with the decoding procedure of the CVAE model. We find that sharing the decoder leads to less diversity in trajectory generation, which is not desirable.

3 Sparse Graph Attention Message-Passing Layer

Before jumping to introducing the specific model we develop for real-world prediction tasks, we would like to present a novel sparse graph attention message-passing (sparse-GAMP) layer, which helps us detect and analyze the social posterior collapse phenomenon.

3.1 Sparse Graph Attention Mechanism

Our sparse-GAMP layer incorporates α-entmax [22] as the graph attention mechanism. α-entmax is a sparse transformation that unifies softmax and sparsemax [23]. Sparse activation functions have drawn growing attention recently because they can induce sparse and interpretable outputs. In the

context of VAEs, they have been used to sparsify discrete latent space for efficient marginalization [24] and tractable multimodal sampling [25]. In our case, we mainly use α-entmax to induce a sparse and interpretable attention map within the encoder for diagnosing social posterior collapse.

We are particularly interested in the 1.5-entmax variant, which is smooth and can be exactly computed. It is also easy to implement on GPUs using existing libraries (e.g., PyTorch [26]). Concretely, the 1.5-entmax function maps a d-dimensional input $\mathbf{s} \in \mathbb{R}^d$ into $\mathbf{p} \in \Delta^d = \left\{\mathbb{R}^d : \mathbf{p} \geqslant 0, \|\mathbf{p}\|_1 = 1\right\}$ as $\mathbf{p} = [\mathbf{s}/2 - \tau\mathbf{1}]_+^2$, where τ is a unique threshold value computed using $\mathbf{s}$. Upon the theoretical results in [22], we derive an insightful proposition which makes 1.5-entmax a merited option in our framework. The proof of the proposition can be found in Appx. B.

Proposition 2. *Let $s_{[d]} \leqslant \cdots \leqslant s_{[1]}$ denote the sorted coordinates of $\mathbf{s}$. Define the top-ρ mean, unnormalized variance, and induced threshold for $\rho \in \{1, ..., d\}$ as*

$$M_{\mathbf{s}}(\rho) = \frac{1}{\rho}\sum_{j=1}^{\rho} s_{[j]},\ S_{\mathbf{s}}(\rho) = \sum_{j=1}^{\rho}\left(s_{[j]} - M_{\mathbf{s}}(\rho)\right)^2,\ \tau_{\mathbf{s}}(\rho) = \begin{cases} M_{\mathbf{s}}(\rho) - \sqrt{\frac{1-S_{\mathbf{s}}(\rho)}{\rho}}, & S_{\mathbf{s}}(\rho) \leqslant 1, \\ +\infty, & \text{otherwise.} \end{cases}$$

Let $\mathbf{s}' \in \mathbb{R}^{d+1}$ satisfy $s'_i = s_i$ for $i \leqslant d$, and define $\mathbf{p} = 1.5\text{-entmax}(\mathbf{s})$ and $\mathbf{p}' = 1.5\text{-entmax}(\mathbf{s}')$.

Then we have: 1) If $\frac{s'_{d+1}}{2} \leqslant \frac{s_{[d]}}{2} - 1$, then $p'_i = p_i$ for $i = 1, ..., d$ and $p'_{d+1} = 0$; 2) If $p_i > 0$ for $i = 1, .., d$, then $p'_i = p_i$ for $i = 1, 2, ..., d$ and $p'_{d+1} = 0$ iff $s'_{d+1} \leqslant 2\tau_{\mathbf{s}/2}(d)$.

The first statement of the proposition provides a sufficient condition for augmenting an input vector without affecting its original attention values. It is useful when applying 1.5-entmax to graph attention. Unlike typical neural network models, graph neural networks (GNN) operate on graphs whose sizes vary over different samples. Meanwhile, nodes within the same graph may have different numbers of incoming edges. Therefore, the 1.5-entmax function has inputs of varying dimensions even within the same batch of training samples, making it inefficient to compute using available primitives. The proposition suggests a simple solution to this problem. Given $\{\mathbf{s}_j\}_{j=1}^m$ with $\mathbf{s}_j \in \mathbb{R}^{d_j}$, we can compute a dummy value as $\min_{j\in\{1,...,m\}, i\in\{1,...,d_j\}} s_{j,i} - 2$. We can augment the input vectors with this dummy value to transform them into a matrix in $\mathbb{R}^{m\times d^*}$, where d^* is the largest value of d_j. The dummy elements will not affect the attention computation, and existing primitives based on matrix computation can be directly used.

The second statement implies that the activated coordinates determine a unique threshold value for augmented coordinates. This property is beneficial when modeling interactions with a large number of agents (e.g., dense traffic scenes). It ensures that the effects of interacting agents will not be diluted by the irrelevant agents, which could potentially improve the robustness of the model [27]. In this work, we mainly utilize the interpretability of the sparse graph attention, but we will investigate its application in generalization in future work.

3.2 Sparse-GAMP

To obtain the sparse-GAMP layer, we combine the sparse graph attention with a message-passing GNN [28, 29]. Given a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with vertices $v \in \mathcal{V}$ and edges $e = (v, v') \in \mathcal{E}$. we define the sparse-GAMP layer as a composition of a node-to-edge message-passing step $v \to e$ and an edge-to-node message-passing step $e \to v$:

$$\begin{aligned} v \to e:\ & \mathbf{h}_{(i,j)} = f_e\left(\left[\mathbf{h}_i, \mathbf{h}_j, \mathbf{u}_{(i,j)}\right]\right), \\ e \to v:\ & \hat{\mathbf{h}}_j = \sum_{i\in\mathcal{N}_j} w_{(i,j)}\mathbf{h}_{(i,j)},\ \text{where } \mathbf{w}_j = \mathcal{G}\text{-entmax}\left(\left\{\mathbf{h}_{(i,j)}\right\}_{i\in\mathcal{N}_j}\right), \end{aligned} \tag{3}$$

In our prediction model, we will use this sparse-GAMP layer to model the function for historical social context encoding, i.e., $\mathbf{T}_i = f_\theta\left(\mathbf{x}_i, \mathbf{x}\right)$.

4 Social-CVAE

In this section, we design a realization of the abstract framework studied in Sec. 2 for real-world trajectory prediction tasks. We choose to design the model with GNNs, which enable a flexible graph

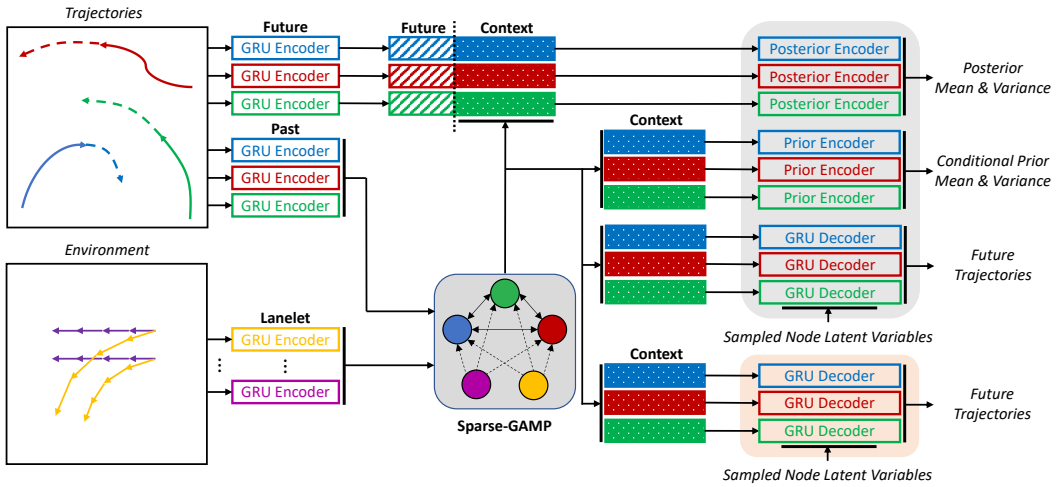

Figure 2: Social-CVAE architecture. We adopt a graph representation and use sparse-GAMP to encode context information. The modules within the orange box are the auxiliary decoders.

representation of data. Several GNN-based approaches achieved the state-of-the-art performance on trajectory prediction task [30, 31, 32]. The resulting model is depicted in Fig. 2, which we refer to as social-CVAE. During training, we first encode the historical and future trajectories of all the agents using a Gated Recurrent Unit (GRU) [33] network. For vehicle trajectory prediction tasks, we incorporate the map information in a manner similar to [34]. Different from [34] where road boundaries are modeled as nodes in the scene graph, we adopt the representation in [35] where the map is denoted as a graph consisting of lanelet nodes, i.e., drivable road segments. For each lanelet, it is represented by its left and right boundaries composed by two sequences of points. We use another GRU network to encode them. Although not utilized in this work, the lanelet representation allows us to combine routing information in a natural manner as in [32]. We will investigate it in future work.

To encode the historical context information, we construct a graph using the embeddings of historical trajectories and lanelets if available. Each pair of agent nodes has a bidirectional edge connecting each other. For each lanelet node, we add edges connecting it to all the agent nodes. If the map information is not available, we add self-edges for all the agents to enable direct self-encoding channels. One sparse-GAMP layer is applied to encode historical context for all the agents. If social posterior collapse occurs, the agent-to-agent edges, except for self-edges, are more likely to receive zero attention weights thanks to the sparse attention. Therefore, we can use the percentage of unattended agent-to-agent edges as a metric to detect social posterior collapse. It is a more objective metric than looking into the quantitative magnitude of attention weights. We can assert that an agent node does not contribute to the output if its attention is strictly zero. However, we need to be more careful when comparing the importance of two agents based on non-zero attention weights [36, 37]. For the same reason, we do not consider techniques such as multiple message-passing layers [34, 31, 32] or separating self-edges from aggregation [8]. Even though they can potentially boost the performance, it might become inconclusive to analyze the model based on the attention map. After computing the historical context, we use a MLP to model the conditional Gaussian prior $p_\eta(\mathbf{z}_i|\mathbf{T}_i)$. To model the variational posterior, we concatenate the future trajectory embedding with each agent's historical context and use another MLP to output the posterior mean and variance. For the decoder, we use another GRU to decode the future trajectory for each agent separately. The auxiliary decoder shares the same structure but always samples the latent variables from the conditional prior regardless of training or testing.

5 Experiments

In this section, we report the experimental results on two trajectory prediction tasks. The main purpose is not achieving state-of-the-art performance but to study the social posterior collapse problem. We compare social-CVAE with two variants: 1) A model without the auxiliary task; 2) A model without

the auxiliary task or the conditional prior. They correspond to the vanilla CVAE and VAE formulations discussed in Sec.2. We are curious about whether the proposed measures can alleviate social posterior collapse. We will briefly introduce the experiment settings and present the main results. Please refer to Appx. C-F for more details, visualization, and additional experiments.

5.1 Vehicle Trajectory Prediction

The first task is the vehicle trajectory prediction problem, where we are asked to predict the future trajectory of one target vehicle given the historical trajectories of itself and other surrounding vehicles. We train the models on two datasets, INTERACTION Dataset [38] and Argoverse Motion Forecasting dataset [39]. To evaluate the prediction performance, we use the standard minimum Average Displacement Error ($minADE$) and minimum Final Displacement Error ($minFDE$) over K sampled trajectories as metrics. And we follow [39] to define $minADE$ as ADE of the trajectory with minimum FDE. Additionally, we define a unique metric, Agent Ratio (AR), to study social posterior collapse:

$$\text{AR} = \frac{\sum_{i=1, i\neq j}^{n} \mathbf{1}(\omega_{(i,j)} \neq 0)}{n-1},$$

where the agent j refers to the predicted target vehicle and $w_{(i,j)}$ is the attention weight assigned to the edge from the agent i to the target vehicle. It equals to the percentage of surrounding vehicles which receive non-zero attention. A value close to zero implies that the model ignores the majority of the surrounding vehicles, which is a sign of social posterior collapse.

INTERACTION Dataset. The INTERACTION dataset provides three categories of driving scenarios: intersection (IS), roundabout (RA), and highway merging (HM). For each experiment, we ran five trials to account for the randomness in initialization. We then evaluated them on the validation sets and computed the mean and standard deviation of the evaluated metrics over all the trials. The best models were selected for testing on the regular (R) and generalization (G) tracks of the INTERPRET Challenge [2]. The results are summarized in Table 1. The CVAE variants have AR values similar to the VAE variants on IS and HM. It is consistent with our argument that merely changing the model to a conditional one is insufficient.

In contrast, our social-CVAE model consistently attains high AR values. However, the CVAE variant achieves similar prediction performance as ours on the validation sets of IS and RA, even if the CVAE variant suffers from the social posterior collapse problem. It is because the driving behavior within intersections and roundabouts depends highly on locations. When the map is less informative (e.g., validation set in HM) or a novel scenario is encountered (e.g., generalization track), our social-CVAE model, which does not have social posterior collapse outperforms the other variants. Notably, we compare it with another instance that attains a AR value close to zero even with the auxiliary task. Their test performances, especially on the generalization track, are pretty different. It shows that it is mainly the historical social context that improves the prediction performance. Even if the auxiliary task also contributes, the improvement is not significant, especially in novel scenes, unless it prevents social posterior collapse.

Argoverse Dataset. Similar to the INTERACTION dataset, we trained five models with random initialization for each case and evaluated them on the validation sets. The best social-CVAE instance was selected for submission to the Argoverse Motion Forecasting Challenge. Although achieving state-of-the-art performance is not our objective, we still report the testing result in Appx. F and compare it with the other models on the leaderboard in order to provide the audience a complete picture of the model. Here, we mainly focus on the validation results in Table 2. The conclusion is consistent. The difference is that the social-CVAE model outperforms the others even on the validation set. It is because the validation set of the Argoverse dataset is collected in different areas of the cities, which is more analogous to the generalization track of the INTERPRET Challenge.

5.2 Pedestrian Trajectory Prediction

The second task is the pedestrian trajectory prediction problem, where we are asked to forecast the future trajectories of all the pedestrians in each scenario. We trained the models on the well-

[2] The challenge adopts a different group of evaluation metrics, please check their website for the formal definitions: `http://challenge.interaction-dataset.com/prediction-challenge/intro`

Table 1: INTERACTION Dataset Validation and Test Results

Scene	Model	AR(%)	K = 1		K = 6	
			*min*FDE	*min*ADE	*min*FDE	*min*ADE
IS	VAE	1.85 ± 4.02	1.32 ± 0.03	0.42 ± 0.01	0.73 ± 0.01	0.26 ± 0.01
	CVAE	0.18 ± 0.36	$\mathbf{1.27 \pm 0.02}$	$\mathbf{0.41 \pm 0.01}$	$\mathbf{0.68 \pm 0.02}$	$\mathbf{0.24 \pm 0.01}$
	Ours	$\mathbf{15.8 \pm 10.4}$	$\mathbf{1.27 \pm 0.02}$	$\mathbf{0.41 \pm 0.01}$	0.70 ± 0.02	0.25 ± 0.01
RA	VAE	0.05 ± 0.04	1.34 ± 0.02	0.42 ± 0.01	0.76 ± 0.01	0.26 ± 0.01
	CVAE	13.4 ± 14.9	1.32 ± 0.01	0.42 ± 0.01	0.73 ± 0.02	$\mathbf{0.25 \pm 0.01}$
	Ours	$\mathbf{19.5 \pm 15.4}$	$\mathbf{1.29 \pm 0.03}$	0.42 ± 0.01	$\mathbf{0.72 \pm 0.01}$	0.26 ± 0.01
HM	VAE	8.68 ± 8.36	0.87 ± 0.08	0.29 ± 0.03	0.42 ± 0.04	0.16 ± 0.01
	CVAE	5.42 ± 10.2	0.83 ± 0.03	0.28 ± 0.03	0.40 ± 0.05	0.16 ± 0.02
	Ours	$\mathbf{15.5 \pm 8.13}$	$\mathbf{0.65 \pm 0.09}$	$\mathbf{0.22 \pm 0.02}$	$\mathbf{0.32 \pm 0.04}$	$\mathbf{0.13 \pm 0.01}$

Track	Model	AR(%)	K = 6			K = 50		
			ADE	FDE	MoN	ADE	FDE	MoN
R	VAE	0.35 ± 5.37	0.5685	1.7573	0.2238	0.5712	1.7709	0.1036
	CVAE	0.07 ± 1.43	0.5323	1.6425	0.2144	0.5410	1.6725	**0.0983**
	Ours*	0.88 ± 2.49	0.5157	1.5823	0.2195	0.5158	1.5823	0.1106
	Ours	$\mathbf{24.8 \pm 20.2}$	**0.4665**	**1.4174**	**0.2011**	**0.4662**	**1.4187**	0.1050
G	VAE	0.02 ± 1.27	1.3428	3.8542	0.8193	1.3436	3.8564	0.5181
	CVAE	0.05 ± 2.20	1.4517	4.2179	0.8743	1.3824	3.9972	0.5676
	Ours*	0.37 ± 2.74	1.1615	3.3927	0.6811	1.1617	3.3939	**0.4218**
	Ours	$\mathbf{15.2 \pm 14.7}$	**0.9205**	**2.7049**	**0.6075**	**0.9186**	**2.6969**	0.4891

Ours* refers to an instance of social-CVAE that still suffers from the social posterior collapse problem. The results are presented in the format of mean $\pm$ std. On the validation set, the mean and std are computed over multiple trials. On the test set, the mean and std of AR are computed over all the samples.

Table 2: Argoverse Dataset Validation Results

Model	AR(%)	K = 1		K = 6	
		*min*FDE	*min*ADE	*min*FDE	*min*ADE
VAE	1.27 ± 1.00	4.17 ± 0.17	1.86 ± 0.07	2.33 ± 0.06	1.30 ± 0.04
CVAE	0.40 ± 0.31	3.85 ± 0.01	1.73 ± 0.01	2.09 ± 0.02	1.19 ± 0.01
Ours	$\mathbf{28.5 \pm 15.7}$	$\mathbf{3.52 \pm 0.03}$	$\mathbf{1.59 \pm 0.02}$	$\mathbf{1.98 \pm 0.02}$	$\mathbf{1.15 \pm 0.01}$

The results are presented in the format of mean $\pm$ std computed over multiple trials.

established ETH/UCY dataset, which combines two datasets, ETH [40] and UCY [41]. Following prior works [2, 42, 4, 7, 9], we adopt the leave-one-out evaluation protocol and do not use any environmental information. All the prediction metrics are computed with $K = 20$ samples. Similar to the vehicle case, we ran five trials for each experiment. The results are summarized in Table 3, where we report the testing results for each group and their average over all the groups. The comparison between our model and other methods can be found in Appx. F. In Table 3, we see that changing to CVAE formulation leads to a significant boost in terms of AR, implying that the model gives more attention to surrounding agents. However, neither changing to CVAE nor the auxiliary task improves prediction performance. It is because historical social context is less informative for pedestrian trajectory prediction. The ETH/UCY dataset is collected in unconstrained environments with few objects. Also, compared to vehicles, pedestrians have fewer physical constraints in 2D motion, resulting in shorter temporal dependency in their movement. It is consistent with the results in [9], where the authors proposed a transformer-based model that achieves the current state-of-the-art performance on ETH/UCY. The visualized attention maps in [9] show that social context in nearby timesteps is more important to their prediction model. In contrast, the historical trajectories of other agents always receive low attention weights. Therefore, even if the auxiliary prediction task

encourages our model to encode historical social context, it does not lead to either a larger AR value or better prediction performance. It suggests that the simplified latent variable model studied in this work might not be sufficient to model pedestrian motion. Explicit future interaction should be considered, and an autoregressive decoder as the one in [9] should be used to account for it.

6 Discussion

Limitations. As the first study on social posterior collapse, we cannot explore every aspect of this subject. Many possible factors could contribute to this phenomenon. Our experimental results can only speak for the particular model we develop and the tasks we study. The occurrence of social posterior collapse and its effect on the overall performance may vary across different problems and different model architectures. For instance, our finding from the pedestrian trajectory prediction task is pretty different from its vehicle counterpart. The format of the graph representation, especially how the environmental information is incorporated, also matters. Also, the diagnosis based on the AR value could be inconclusive under different graph representations. In Appx. E.2, we show that the AR values of the three models become similar after removing the lanelet nodes for the highway merging subset of the INTERACTION dataset. However, it does not mean that social posterior collapse does not occur. Our social-CVAE model can maintain consistent prediction accuracy without map information, while the baseline VAE model has a significant drop in performance. It implies that the baseline VAE model does not properly utilize historical social context, but we cannot assert that social posterior collapse occurs due to the lack of direct evidence.

Nevertheless, we do demonstrate that social posterior collapse is not unique to our sparse-GAMP module (Appx. E.1). Even if we switch to other aggregation functions, we can still find evidence suggesting its occurrence. Our sparse graph attention function, $\mathcal{G}$-entmax, is a convenient and flexible toolkit that allows us to monitor and analyze social posterior collapse without compromising performance. In the future, we will work along this direction to explore alternative diagnosis toolkits that can be applied to detect social posterior collapse for a broader range of models.

Connections to Related Works. We want to wrap up our discussion with a glance at those prior works on VAE-based multi-agent behavior modeling. We are curious about if any elements in their models have implicitly tackled this issue. However, we would like to emphasize that it is still necessary to explicitly study social posterior collapse under their settings in the future, which could provide helpful guidance to avoid social posterior collapse in model design. Due to the space limit, we only discuss works using techniques potentially related to social posterior collapse, especially those provided evidence that interacting behaviors were appropriately modeled (e.g., attention maps or visualized prediction results in interactive scenarios).

Trajectron [6] and Trajectron++ [4] which are formulated as CVAEs establish the previous state-of-the-art results on ETH/UCY. Different from ours, they adopted a discrete latent space to account for the multi-modality in human behavior. They did not examine how their models utilize social context. However, we think that a discrete latent space could potentially alleviate the social posterior collapse issue. Compared to a continuous random variable, a discrete one can only encode a limited amount of information. It prevents the model from bypassing social context since a discrete latent variable is not sufficient to encode all the information of a long trajectory. For the same reason, NRI [29], which adopted a discrete latent space for interacting system modeling, is also potentially relevant. PECNet [7] is another CVAE-based trajectory prediction model. Instead of conditioning on the entire future trajectory, they proposed the Endpoint VAE where the posterior latent variables only condition on the endpoints. It could also be a potential solution as it limits the amount of information the latent space can encode from the future trajectory.

In [1], Suo et al. proposed a multi-agent behavior model for traffic simulation under the CVAE framework. They demonstrated that their method was able to simulate complex and realistic interactive behaviors. In particular, they augmented ELBO with a commonsense objective that regularizes the model from synthesizing undesired interactions (e.g., collisions). We think it plays a similar role as our auxiliary prediction task. The last work we would like to discuss is the AgentFormer model [9] mentioned in Sec. 5.2. Their results suggest that incorporating an autoregressive decoder and a future social context encoder could be more effective in interaction modeling, especially for systems without long-term dependency (e.g., pedestrians). However, as we have mentioned before, the autoregressive decoder introduces another issue if it is overly powerful.

Table 3: ETH/UCY Dataset Leave-one-out Testing Results

Model	ETH			HOTEL		
	AR(%)	$minFDE$	$minADE$	AR(%)	$minFDE$	$minADE$
VAE	74.0 ± 7.33	0.98 ± 0.07	0.63 ± 0.03	29.0 ± 28.1	0.28 ± 0.01	0.19 ± 0.01
CVAE	$\mathbf{90.9 \pm 6.38}$	$\mathbf{0.94 \pm 0.10}$	$\mathbf{0.61 \pm 0.05}$	69.8 ± 41.7	$\mathbf{0.27 \pm 0.01}$	$\mathbf{0.18 \pm 0.01}$
Ours	83.2 ± 12.6	1.08 ± 0.10	0.68 ± 0.04	$\mathbf{72.9 \pm 21.5}$	$\mathbf{0.27 \pm 0.02}$	$\mathbf{0.18 \pm 0.01}$
Model	ZARA1			ZARA2		
	AR(%)	$minFDE$	$minADE$	AR(%)	$minFDE$	$minADE$
VAE	40.3 ± 22.8	$\mathbf{0.38 \pm 0.01}$	$\mathbf{0.22 \pm 0.01}$	30.1 ± 17.1	$\mathbf{0.32 \pm 0.01}$	$\mathbf{0.18 \pm 0.01}$
CVAE	$\mathbf{89.4 \pm 7.65}$	0.39 ± 0.01	$\mathbf{0.22 \pm 0.01}$	$\mathbf{64.2 \pm 25.6}$	0.35 ± 0.01	0.19 ± 0.01
Ours	61.9 ± 5.71	$\mathbf{0.38 \pm 0.01}$	$\mathbf{0.22 \pm 0.01}$	58.6 ± 15.7	0.37 ± 0.02	0.20 ± 0.01
Model	UNIV			Average		
	AR(%)	$minFDE$	$minADE$	AR(%)	$minFDE$	$minADE$
VAE	0.06 ± 0.09	$\mathbf{0.55 \pm 0.01}$	$\mathbf{0.31 \pm 0.01}$	34.7 ± 29.4	$\mathbf{0.50 \pm 0.27}$	$\mathbf{0.30 \pm 0.17}$
CVAE	$\mathbf{41.2 \pm 25.1}$	0.63 ± 0.05	0.35 ± 0.02	$\mathbf{71.1 \pm 29.5}$	0.51 ± 0.25	0.31 ± 0.17
Ours	36.3 ± 14.4	0.63 ± 0.02	0.35 ± 0.01	62.6 ± 21.0	0.54 ± 0.29	0.33 ± 0.19

The results are presented in the format of $\text{mean} \pm \text{std}$ computed over multiple trials.

7 Conclusion

In this work, we point out an under-explored issue, which we refer to as social posterior collapse, in the context of VAEs in multi-agent modeling. We argue that one of the commonly adopted formulations of VAEs in multi-agent modeling is prone to ignoring historical social context when predicting the future trajectory of an agent. We analyze the reason behind and propose several measures to alleviate social posterior collapse. Afterward, we design a GNN-based realization of the general framework incorporating the proposed measures, which we refer to as social-CVAE. In particular, social-CVAE incorporates a novel sparse-GAMP layer which helps us detect and analyze social posterior collapse. In our experiments, we show that social posterior collapse occurs in real-world trajectory prediction problems and that the proposed measures effectively alleviate the issue. Also, the experimental results imply that social posterior collapse could cause poor generalization performance in novel scenarios if the future movement of the agents indeed depends on their historical social context. In the future, we will utilize the toolkit developed in this work to explore social posterior collapse in a broader range of model architectures and interacting systems.

Acknowledgements

We would like to thank Liting Sun, Xiaosong Jia, and Jiachen Li for insightful discussion. We thank Vade Shah for helping us with the experiments. We thank Chenfeng Xu and Hengbo Ma for their valuable feedback. We would also like to thank the anonymous reviewers for their helpful review comments. This work is supported by Denso International America, Inc.

References

[1] S. Suo, S. Regalado, S. Casas, and R. Urtasun, "TrafficSim: Learning to simulate realistic multi-agent behaviors," *arXiv preprint arXiv:2101.06557*, 2021.

[2] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social GAN: Socially acceptable trajectories with generative adversarial networks," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[3] V. Kosaraju, A. Sadeghian, R. Martín-Martín, I. Reid, H. Rezatofighi, and S. Savarese, "Social-BiGAT: Multimodal trajectory forecasting using bicycle-GAN and graph attention networks," in *Advances in Neural Information Processing Systems*, 2019.

[4] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone, "Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data," *arXiv preprint arXiv:2001.03093*, 2020.

[5] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[6] B. Ivanovic and M. Pavone, "The Trajectron: Probabilistic multi-agent trajectory modeling with dynamic spatio-temporal graphs," in *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2375–2384, 2019.

[7] K. Mangalam, H. Girase, S. Agarwal, K.-H. Lee, E. Adeli, J. Malik, and A. Gaidon, "It is not the journey but the destination: Endpoint conditioned trajectory prediction," in *European Conference on Computer Vision (ECCV)*, pp. 759–776, Springer, 2020.

[8] J. Li, H. Ma, Z. Zhang, and M. Tomizuka, "Social-WAGDAT: Interaction-aware trajectory prediction via wasserstein graph double-attention network," *arXiv preprint arXiv:2002.06241*, 2020.

[9] Y. Yuan, X. Weng, Y. Ou, and K. Kitani, "AgentFormer: Agent-aware transformers for socio-temporal multi-agent forecasting," *arXiv preprint arXiv:2103.14023*, 2021.

[10] X. Chen, D. P. Kingma, T. Salimans, Y. Duan, P. Dhariwal, J. Schulman, I. Sutskever, and P. Abbeel, "Variational lossy autoencoder," *arXiv preprint arXiv:1611.02731*, 2016.

[11] S. R. Bowman, L. Vilnis, O. Vinyals, A. M. Dai, R. Jozefowicz, and S. Bengio, "Generating sentences from a continuous space," *arXiv preprint arXiv:1511.06349*, 2015.

[12] M. Fraccaro, S. K. Sønderby, U. Paquet, and O. Winther, "Sequential neural models with stochastic layers," *arXiv preprint arXiv:1605.07571*, 2016.

[13] I. Serban, A. Sordoni, R. Lowe, L. Charlin, J. Pineau, A. Courville, and Y. Bengio, "A hierarchical latent variable encoder-decoder model for generating dialogues," in *AAAI Conference on Artificial Intelligence*, 2017.

[14] Y. Hu, W. Zhan, L. Sun, and M. Tomizuka, "Multi-modal probabilistic prediction of interactive behavior via an interpretable model," in *IEEE Intelligent Vehicles Symposium (IV)*, pp. 557–563, IEEE, 2019.

[15] C. Tang, N. Srishankar, S. Martin, and M. Tomizuka, "Grounded relational inference: domain knowledge driven explainable autonomous driving," *arXiv preprint arXiv:2102.11905*, 2021.

[16] P. de Haan, D. Jayaraman, and S. Levine, "Causal confusion in imitation learning," *arXiv preprint arXiv:1905.11979*, 2019.

[17] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social LSTM: Human trajectory prediction in crowded spaces," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[18] S. Khandelwal, W. Qi, J. Singh, A. Hartnett, and D. Ramanan, "What-if motion prediction for autonomous driving," *arXiv preprint arXiv:2008.10587*, 2020.

[19] H. Fu, C. Li, X. Liu, J. Gao, A. Celikyilmaz, and L. Carin, "Cyclical annealing schedule: A simple approach to mitigating KL vanishing," in *Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2019.

[20] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," *Advances in Neural Information Processing Systems*, vol. 28, pp. 3483–3491, 2015.

[21] B. Ivanovic, K. Leung, E. Schmerling, and M. Pavone, "Multimodal deep generative models for trajectory prediction: A conditional variational autoencoder approach," *IEEE Robotics and Automation Letters (RA-L)*, vol. 6, no. 2, pp. 295–302, 2020.

[22] B. Peters, V. Niculae, and A. F. Martins, "Sparse sequence-to-sequence models," *arXiv preprint arXiv:1905.05702*, 2019.

[23] A. Martins and R. Astudillo, "From softmax to sparsemax: A sparse model of attention and multi-label classification," in *International Conference on Machine Learning (ICML)*, pp. 1614–1623, PMLR, 2016.

[24] G. M. Correia, V. Niculae, W. Aziz, and A. F. Martins, "Efficient marginalization of discrete and structured latent variables via sparsity," *arXiv preprint arXiv:2007.01919*, 2020.

[25] M. Itkina, B. Ivanovic, R. Senanayake, M. J. Kochenderfer, and M. Pavone, "Evidential sparsification of multimodal latent spaces in conditional variational autoencoders," *arXiv preprint arXiv:2010.09164*, 2020.

[26] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *arXiv preprint arXiv:1912.01703*, 2019.

[27] A. Goyal, A. Lamb, J. Hoffmann, S. Sodhani, S. Levine, Y. Bengio, and B. Schölkopf, "Recurrent independent mechanisms," *arXiv preprint arXiv:1909.10893*, 2019.

[28] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *International Conference on Machine Learning (ICML)*, pp. 1263–1272, PMLR, 2017.

[29] T. Kipf, E. Fetaya, K.-C. Wang, M. Welling, and R. Zemel, "Neural relational inference for interacting systems," in *International Conference on Machine Learning (ICML)*, pp. 2688–2697, PMLR, 2018.

[30] H. Zhao, J. Gao, T. Lan, C. Sun, B. Sapp, B. Varadarajan, Y. Shen, Y. Shen, Y. Chai, C. Schmid, *et al.*, "TNT: Target-driven trajectory prediction," *arXiv preprint arXiv:2008.08294*, 2020.

[31] M. Liang, B. Yang, R. Hu, Y. Chen, R. Liao, S. Feng, and R. Urtasun, "Learning lane graph representations for motion forecasting," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12347 LNCS, pp. 541–556, 2020.

[32] W. Zeng, M. Liang, R. Liao, and R. Urtasun, "LaneRCNN: Distributed representations for graph-centric motion forecasting," *arXiv preprint arXiv:2101.06653*, 2021.

[33] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[34] J. Gao, C. Sun, H. Zhao, Y. Shen, D. Anguelov, C. Li, and C. Schmid, "VectorNet: Encoding hd maps and agent dynamics from vectorized representation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11522–11530, 2020.

[35] P. Bender, J. Ziegler, and C. Stiller, "Lanelets: Efficient map representation for autonomous driving," in *IEEE Intelligent Vehicles Symposium (IV)*, pp. 420–425, 2014.

[36] S. Jain and B. C. Wallace, "Attention is not explanation," in *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 3543–3556, 2019.

[37] S. Wiegreffe and Y. Pinter, "Attention is not not explanation," in *Conference on Empirical Methods in Natural Language Processing and International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 11–20, 2019.

[38] W. Zhan, L. Sun, D. Wang, H. Shi, A. Clausse, M. Naumann, J. Kummerle, H. Konigshof, C. Stiller, A. de La Fortelle, *et al.*, "Interaction dataset: An international, adversarial and cooperative motion dataset in interactive driving scenarios with semantic maps," *arXiv preprint arXiv:1910.03088*, 2019.

[39] M.-F. Chang, J. Lambert, P. Sangkloy, J. Singh, S. Bak, A. Hartnett, D. Wang, P. Carr, S. Lucey, D. Ramanan, *et al.*, "Argoverse: 3d tracking and forecasting with rich maps," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8748–8757, 2019.

[40] S. Pellegrini, A. Ess, K. Schindler, and L. van Gool, "You'll never walk alone: Modeling social behavior for multi-target tracking," in *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 261–268, 2009.

[41] A. Lerner, Y. Chrysanthou, and D. Lischinski, "Crowds by example," in *Computer Graphics Forum*, vol. 26, pp. 655–664, Wiley Online Library, 2007.

[42] A. Sadeghian, V. Kosaraju, A. Sadeghian, N. Hirose, H. Rezatofighi, and S. Savarese, "Sophie: An attentive GAN for predicting paths compliant to social and physical constraints," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1349–1358, 2019.

[43] J. Pearl, *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. Elsevier, 2014.

[44] M. Fey and J. E. Lenssen, "Fast graph representation learning with PyTorch Geometric," in *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019.

[45] R. Liaw, E. Liang, R. Nishihara, P. Moritz, J. E. Gonzalez, and I. Stoica, "Tune: A research platform for distributed model selection and training," *arXiv preprint arXiv:1807.05118*, 2018.

[46] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016.

[47] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[48] M. Ye, T. Cao, and Q. Chen, "TPCN: Temporal point cloud networks for motion forecasting," *arXiv preprint arXiv:2103.03067*, 2021.

[49] F. Giuliari, I. Hasan, M. Cristani, and F. Galasso, "Transformer networks for trajectory forecasting," in *International Conference on Pattern Recognition (ICPR)*, pp. 10335–10342, 2021.

[50] C. Yu, X. Ma, J. Ren, H. Zhao, and S. Yi, "Spatio-temporal graph transformer networks for pedestrian trajectory prediction," in *European Conference on Computer Vision (ECCV)*, pp. 507–523, Springer, 2020.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] In Sec. 6, we mention that our findings are limited to the particular model architecture and tasks studied in this work.
   (c) Did you discuss any potential negative societal impacts of your work? [No] We believe our work should not have any negative societal impacts.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes] See supplementary materials.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [No] Because the code is proprietary.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See supplementary materials.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] We do not estimate the amount of computation but provide the specification of the computational resources.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes] See supplementary materials.
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] Yes, we mention how we obtained the data in the supplementary materials. Most datasets and packages we used are open-source. For the remaining INTERACTION dataset which is not open-source, we mention that we have been granted the access for research usage.

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] We checked the format of the information provided by the datasets. There are no elements from the dataset that could potentially contain any personal information or offensive content.

5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Proof for Proposition 1

Proof. Because the graph in Fig. 1 is a directed acyclic graph (DAG), we can apply the d-separation criterion [43] to analyze the conditional independence. For each pair of i and j with $i \neq j$, the node $\mathbf{z}_i$ is connected to $\mathbf{y}_j$ through $\mathbf{T}_i$ and $\mathbf{T}_j$. And any path between $\mathbf{T}_i$ and $\mathbf{T}_j$ has a triple $\mathbf{T}_i \leftarrow \mathbf{x}_k \rightarrow \mathbf{T}_j$ for some $k = 1, 2, ..., n$, which is inactive after conditioning on $\mathbf{x}$. Therefore, we can apply the d-separation criterion to conclude that $(\mathbf{z}_i \perp\!\!\!\perp \mathbf{y}_j) \mid \mathbf{x}, \mathbf{y}_i$. The same inactive triples also imply that $(\mathbf{z}_i \perp\!\!\!\perp \mathbf{z}_j) \mid \mathbf{x}$. For the third statement, any path between $\mathbf{z}_i$ and $\mathbf{x}_j$ has a inactive path $\mathbf{x}_j \rightarrow \mathbf{T}_i \rightarrow \mathbf{y}_i$. Therefore, the d-separation criterion implies $(\mathbf{z}_i \perp\!\!\!\perp \mathbf{x}_j) \mid \mathbf{T}_i$. □

B Proof for Proposition 2

Proof. Proposition 3 in [22] suggests that the threshold value equals to $\tau_{\mathbf{s}/2}(\rho^*)$ with any ρ^* satisfying $\tau_{\mathbf{s}/2}(\rho^*) \in [\frac{s_{[\rho^*+1]}}{2}, \frac{s_{[\rho^*]}}{2}]$. If d satisfies the condition, then $\tau_{\mathbf{s}/2}(d) \in (-\infty, \frac{s_{[d]}}{2}]$ which is clearly finite. Therefore, $\tau_{\mathbf{s}/2}(d) = M_{\mathbf{s}/2}(d) - \sqrt{\frac{1-S_{\mathbf{s}/2}(d)}{d}} \geqslant \frac{s_{[d]}}{2} - 1$ by definition. Given $\frac{s'_{d+1}}{2} \leqslant \frac{s_{[d]}}{2} - 1$, we have $\tau_{\mathbf{s}'/2}(d) = \tau_{\mathbf{s}/2}(d) \in [\frac{s'_{d+1}}{2}, \frac{s'_{[d]}}{2}]$. Therefore, $\tau_{\mathbf{s}/2}(d)$ is still the threshold value. If d is not a valid ρ^*, then there exists $\rho^* \in \{1, ..., d-1\}$ defining the threshold value. Because $\frac{s'_{d+1}}{2} < \frac{s_{[d]}}{2}$, augmenting the input vector does not affect the threshold value. In both cases, the threshold value is unchanged which leads to the first statement of the proposition.

Now we prove the second statement of the proposition. Because $p_i > 0$ for $i = 1, .., d$, the threshold value is smaller than $\frac{s_{[d]}}{2}$, which makes d the only possible ρ^*. If $s'_{d+1} \leqslant 2\tau_{\mathbf{s}/2}(d)$, then $\tau_{\mathbf{s}/2}(d)$ defines the threshold value for $\mathbf{s}'$. Therefore, $p'_i = p_i$ for $i = 1, 2, ..., d$ and $p'_{d+1} = 0$. Instead, if we are given $p'_i = p_i$ for $i = 1, 2, ..., d$ and $p'_{d+1} = 0$, the threshold satisfies $\tau_{\mathbf{s}'/2}(\rho^*) \in \left[\frac{s'_{d+1}}{2}, \frac{s_{[d]}}{2}\right]$. Therefore, d is a valid threshold index for both $\mathbf{s}$ and $\mathbf{s}'$, and $s'_{d+1} \leqslant 2\tau_{\mathbf{s}'/2}(d) = 2\tau_{\mathbf{s}/2}(d)$. □

C Experiment Details

In this section, we report additional experiment details for the experiments in Sec. 5, including data processing scheme, implementation details, and hyper-parameters used.

C.1 Data Processing Scheme

INTERACTION Dataset. The access to the dataset is granted for non-commercial usage through its official website: `https://interaction-dataset.com` (Copyright©All Rights Reserved). First, we use the script from the official code repository to split the training and validation sets and segment the data. Each sample has a length of 4 seconds, with an observation window of 1 second and a prediction window of 3 seconds. The sampling frequency is 10Hz. We further augment the dataset by iteratively assigning each vehicle in a sample as the target vehicle. Afterward, we follow the common practice to translate and rotate the coordinate system, such that the target vehicle locates at the origin with a zero-degree heading angle at the last observed frame.

Regarding map information, the INTERACTION dataset provides maps in a format compatible with the lanelet representation. For each lanelet, we fit its boundaries to two B-splines through spline regression so that we can uniformly sample a fixed number of points on each boundary. Apart from the coordinates, we add additional discrete features (e.g., boundary type, the existence of a stop sign) to each boundary point.

Argoverse Dataset. We download the dataset (v1.1) including the training, validation and testing subsets from its official website: `https://www.argoverse.org/data.html` (Copyright©2020 Argo AI, LLC). Each sample in the Argoverse Dataset has a length of 5 seconds, with an observation window of 2 seconds and a prediction window of 3 seconds. The sampling frequency is 10Hz. We follow a similar scheme to process the dataset. However, since each sample contains vehicles located in many city blocks, we first filter out those irrelevant vehicles and road segments. We remove surrounding vehicles whose distance to the target vehicle is larger than a certain threshold value at the last observed frame. To identify irrelevant road segments, we use a heuristic-based graph search

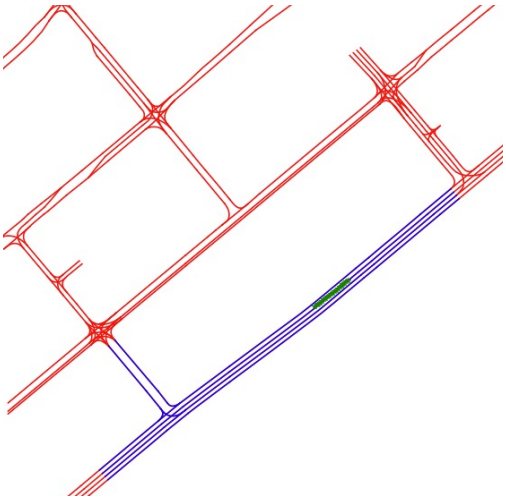

Figure 3: Road segments found by the graph search algorithm. We denote the observed trajectory of the target vehicle by the green dots. The road segments returned by the graph search algorithm are highlighted in the color of blue, whereas the other road segments are drawn in the color of red.

algorithm similar to the one used in [32] to obtain the road segments of interest (ROI). Instead of setting a threshold Euclidean distance, we decide whether a road segment is relevant by estimating the traveling distance from the target vehicle's location to it. The algorithm is summarized in Alg. 1. Given x, the coordinates of the target vehicle at the last observed frame, we set the traveling distance threshold $d_{\max}$ based on the displacement of the target vehicle during the observed time horizon. A larger distance threshold is necessary if the target vehicle is driving at high speed. We initialize the graph search by finding road segments close to x from the map. Afterward, we expand the searching graph by adding adjacent, preceding, and succeeding road segments and finding all segments within the threshold of traveling distances. We use simple heuristics to compute the traveling distance between two segments. If two segments are adjacent, the distance is set to zero. If one segment is a predecessor or successor of the other segment, we set the distance to be the average of their centerlines' length. Fig. 3 shows an example of the resulting road segments.

Another issue is that the heading angles of the vehicles are not provided. We need to estimate the heading angles of the target vehicles in order to rotate the coordinate system. However, the trajectory data are noisy with tracking errors. Simply interpolating the coordinates between consecutive time steps results in noisy estimation. Instead, we first estimate the heading angle at each observed frame by interpolating the coordinates and then obtain a smooth estimation of the heading angle at the last observed frame as follows:

$$\hat{\psi}_{T_h} = \sum_{t=0}^{T_h} \lambda^{T_h - t} \psi_t,$$

where T_h is the number of observed frames, ψ_t is the estimated heading at the t^{th} frame, $\lambda \in (0, 1)$ is the forgetting factor, and $\hat{\psi}_{T_h}$ is the smoothed heading estimation at the last observed frame.

ETH/UCY Dataset. For the ETH/UCY datasets, we adopt the leave-one-out evaluation protocol as in prior works [2, 42, 4, 7, 9] to obtain five groups of datasets: ETH & HOTEL (from ETH) and UNIV, ZARA1 & ZARA2 (from UCY). The data from the corresponding scenario are left out as testing data in each group, and the remaining data are used for training and validation. We used the segmented datasets provided by the code base of social-GAN (MIT License) [2]. Each sample has a length of 8 seconds, with an observation window of 3.2 seconds and a prediction window of 4.8 seconds. The sampling frequency is 2.5Hz. We do not use any visual or semantic information for fair comparisons to prior works. The origin of the coordinate system is translated to the mean position of all agents at the last observed frame. Random rotation [42, 4] is adopted for data augmentation.

Algorithm 1: ROI Graph Search

function ROIGRAPHSEARCH($x, d_{\max}, d_{\text{init}}, map$)
  $segments \leftarrow map.\text{segmentsContainXY}(x)$ ▷ A set of segments containing x
  while $segments$ is empty and $d_{\text{init}} < d_{\max}$ **do** ▷ Search local region if empty
    $segments \leftarrow map.\text{segmentsInBoundingBox}(x, d_{\text{init}})$
    $d_{\text{init}} \leftarrow 2d_{\text{init}}$
  end while
  $pool \leftarrow$ initialize a FIFO queue
  for $segment$ in $segments$ **do**
    $node \leftarrow \text{Node}(segment, 0, 0)$ ▷ Create node with segment, length and distance
    $pool \leftarrow \text{Insert}(node, pool)$
  end for
  $ROI \leftarrow$ an empty dictionary of nodes
  while $pool$ is not empty **do**
    $node \leftarrow \text{Pop}(pool)$
    if $node.segment$ not in ROI or $ROI[segment].distance \geqslant node.distance$ **then**
      $ROI[segment] \leftarrow node$
    end if
    $children \leftarrow map.\text{adjacentSegments}(node.segment)$ ▷ Get adjacent road segments
    for $child$ in $children$ **do**
      $length \leftarrow map.\text{segmentCenterline}(child)$ ▷ Get centerline length
      $node \leftarrow \text{Node}(child, length, node.distance)$
      $pool \leftarrow \text{Insert}(node, pool)$
    end for
    $predecessors \leftarrow map.\text{predecessor}(node.segment)$ ▷ Get preceding road segments
    $successors \leftarrow map.\text{successor}(node.segment)$ ▷ Get succeeding road segments
    $children \leftarrow predecessors + successors$ ▷ Combine predecessors and successors
    for $child$ in $children$ **do**
      $length \leftarrow map.\text{segmentCenterline}(child)$
      $distance \leftarrow \frac{1}{2}(length + node.length) + node.distance$
      if $distance \leqslant d_{\max}$ **then**
        $node \leftarrow \text{Node}(child, length, distance)$
        $pool \leftarrow \text{Insert}(node, pool)$
      end if
    end for
  end while
  return ROI
end function

C.2 Implementation Details

We implement our social-CVAE model using Pytorch 1.8 (Copyright©Facebook, Inc) [26] and Pytorch Geometric (PyG) 1.7 (MIT License) [44], a geometric deep learning extension library for Pytorch which implements various popular GNN models. The sparse-GAMP network is implemented by adapting the base message-passing class from PyG. The $\mathcal{G}$-entmax function is implemented using the entmax package (MIT License) from [22]. The function f_e in Eqn. (3) consists of two MLP networks: one network encodes $\mathbf{h}_i$ and $\mathbf{h}_j$ separately; one network generates $\mathbf{h}_{(i,j)}$ after concatenating the node embeddings with $\mathbf{u}_{(i,j)}$. We select the hyper-parameters through cross-validation with the aid of Tune (Copyright©2021 The Ray Team) [45]. All the MLPs used in the model are one-layer MLPs with layer normalization applied [46]. All the networks, including MLPs and GRUs, have 64 hidden units. For the experiments on the INTERACTION dataset, we choose a latent space of 16 dimensions. For the experiments on the Argoverse dataset and the ETH/UCY dataset, we choose a latent space of 32 dimensions. Regarding the loss function, we follow the common practice to fix the variance of $p_{\theta,\phi}(\mathbf{y}|\mathbf{x}, \mathbf{z})$. Consequently, the reconstruction loss is equivalent to an MSE loss, and the

overall objective becomes maximizing the following objective function:

$$\hat{\mathcal{L}}(\theta,\psi,\phi,\eta,\zeta) = \mathbb{E}_{\mathbf{x},\mathbf{y}\sim D}\left[\mathbb{E}_{\mathbf{z}\sim q_{\theta,\psi}(\mathbf{z}|\mathbf{x},\mathbf{y})}\left\|h_{\theta,\phi}(\mathbf{x},\mathbf{z})-\mathbf{y}\right\|^2 - \beta D_{KL}\left[q_{\theta,\psi}\left(\mathbf{z}|\mathbf{x},\mathbf{y}\right)\|p_{\theta,\eta}(\mathbf{z}|\mathbf{x})\right]\right] - \alpha\mathbb{E}_{\mathbf{x},\mathbf{y}\sim D,\mathbf{z}\sim p_{\theta,\eta}(\mathbf{z}|\mathbf{x})}\left\|h_{\theta,\zeta}(\mathbf{x},\mathbf{z})-\mathbf{y}\right\|^2.$$

For the vehicle prediction experiments, we set $\beta = 0.03$. For the pedestrian prediction experiments, we set $\beta = 0.01$. For the weight of the auxiliary loss function, we set $\alpha = 0.3$ on the INTERACTION dataset, $\alpha = 0.5$ on the Argoverse dataset, and $\alpha = 0.2$ on the ETH/UCY dataset. We use Adam [47] to optimize the objective function. To generate trajectories for evaluation, if $K = 1$, we sample a single trajectory by taking the mean from the prior or conditional prior as the latent variables. If $K > 1$, we randomly sample the latent variables to generate multiple trajectories. However, the Argoverse Motion Forecasting Challenge evaluates the metrics for $K = 1$ by picking one trajectory out of all the submitted samples. Therefore, for evaluation on the Argoverse dataset, we randomly sample $K - 1$ trajectories and leave the last one as the trajectory corresponding to the mean value of the latent variables. For the vehicle prediction experiments, all the models are trained for 100 epochs with a batch size of 40. For the pedestrian experiments, we choose a batch size of 20. All the models were trained with four RTX 2080 Ti and an Intel Core i9-9920X (12 Cores, 3.50 GHz). However, only a quarter of a single GPU is required to train one model.

D Visualization

In this section, we visualize the experiment results for the vehicle prediction task. Fig. 4 and Fig. 5 show several examples from the INTERACTION dataset and the Argoverse dataset respectively. For each instance, we compare the outputs of the three model variants we study in Sec. 5. The target vehicles' historical trajectories and ground-truth future trajectories are denoted by blue and red dots, respectively. We sample six predicted trajectories from each model and plot them in dash lines of different colors. We also approximate the density function of the prediction output using a kernel density estimator and visualize it with a color map. Since we are particularly interested in monitoring the social posterior collapse phenomenon, we visualize the attention map by highlighting the surrounding vehicles and lanelets that receive non-zero attention weights. Particularly, if a vehicle received non-zero attention, we use the same way as the target vehicle to annotate its historical and future trajectories. Otherwise, its trajectories are annotated with grey dots. If a lanelet node receives non-zero attention, we highlight its boundaries or centerline with orange lines. The VAE and CVAE variants ignore all the surrounding vehicles in all the visualized instances, including those close to the target vehicles. They only pay attention to the lanelet nodes, which results in insensible prediction results, for instance, colliding into preceding vehicles (e.g., the second row in Fig. 4 and Fig. 5). In contrast, the social-CVAE models assign attention weights to vehicles that might potentially interact with the target vehicles and maintain sparse attention maps in dense traffic scenes (e.g., the last two rows in Fig. 4).

E Ablation Study

E.1 Effect of Aggregation Functions

In this section, we present an ablation study on the aggregation functions used in the message-passing network. Because of the sparsity nature of α-entmax, the usage of sparse graph attention may induce social posterior collapse. Therefore, we would like to study whether social posterior collapse will still occur if we switch to other aggregation functions. We consider two variants for comparison. The first one is replacing $\mathcal{G}$-entmax with the conventional softmax function, resulting in the following message-passing operations:

$$v \to e: \ \mathbf{h}_{(i,j)} = f_e\left(\left[\mathbf{h}_i, \mathbf{h}_j, \mathbf{u}_{(i,j)}\right]\right),$$
$$e \to v: \quad \hat{\mathbf{h}}_j = \sum_{i\in\mathcal{N}_j} w_{(i,j)}\mathbf{h}_{(i,j)}, \ \text{ where } \mathbf{w}_j = \text{softmax}\left(\left\{\mathbf{h}_{(i,j)}\right\}_{i\in\mathcal{N}_j}\right),$$

The second variant is using the max aggregation instead of the weighted sum. The message-passing layer becomes the one in Eqn. (4) - (5). The max aggregation takes the element-wise maximum along the dimension of the hidden unit. Therefore, it allows the number of activated nodes to be

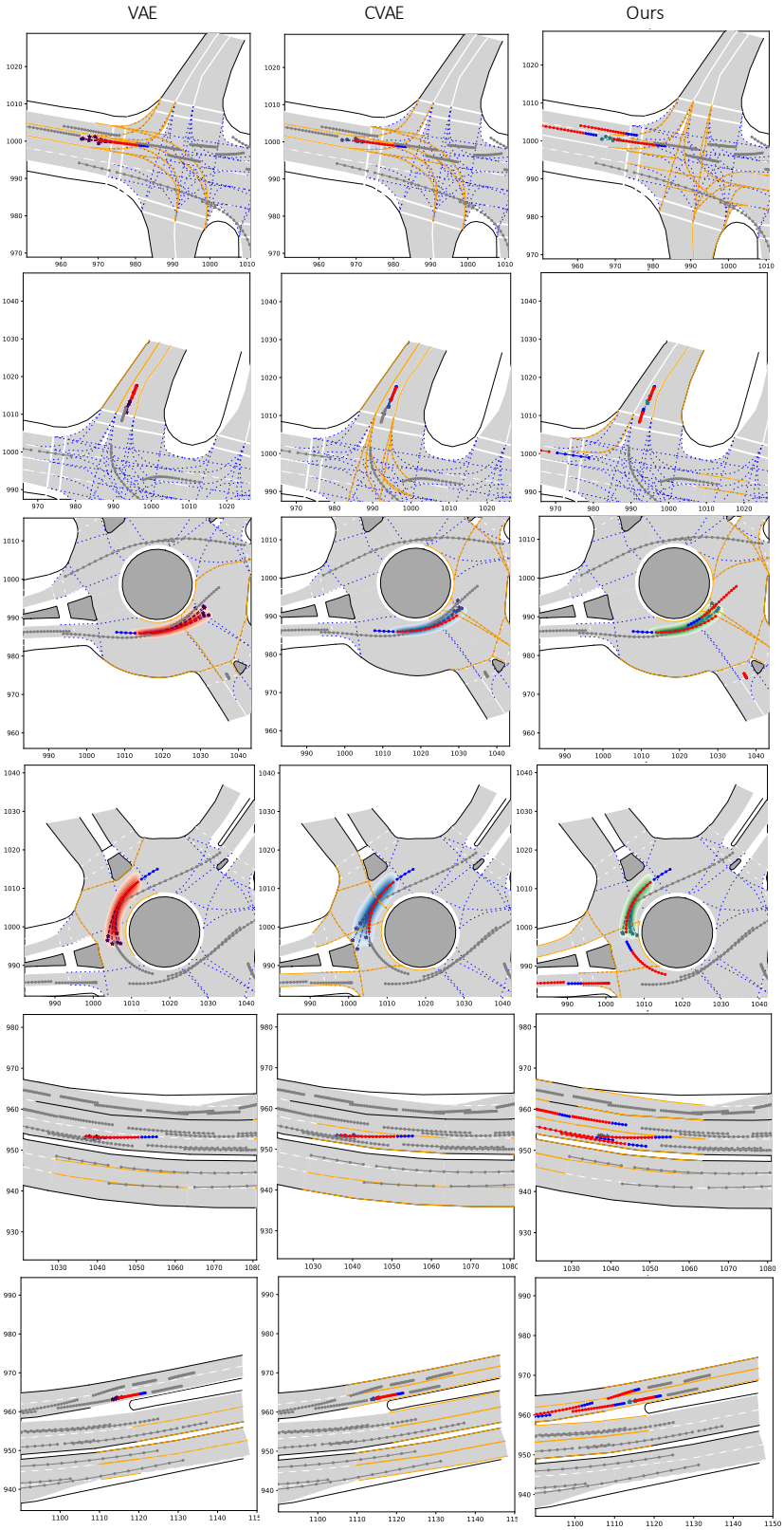

Figure 4: Visualizing prediction results on the Interaction dataset.

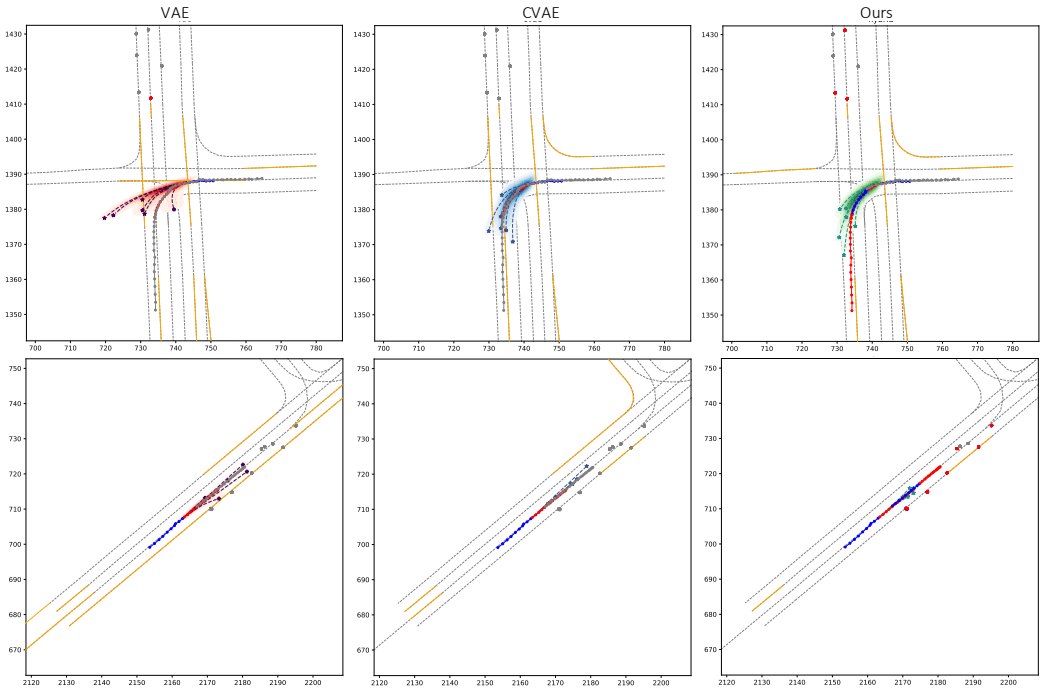

Figure 5: Visualizing prediction results on the Argoverse dataset.

at most the dimension of $\mathbf{h}_{(i,j)}$. The social posterior collapse issue should be able to be avoided if $\mathcal{G}$-entmax is the reason for its occurrence.

$$v \to e : \mathbf{h}_{(i,j)} = f_e\left(\left[\mathbf{h}_i, \mathbf{h}_j, \mathbf{u}_{(i,j)}\right]\right), \tag{4}$$

$$e \to v : \quad \hat{\mathbf{h}}_j = \text{max-aggregate}\left(\{\mathbf{h}_{(i,j)}\}_{i \in N_j}\right). \tag{5}$$

The issue that remains is the evaluation metric. Unlike sparse-GAMP, we do not have the privilege to detect social posterior collapse by monitoring the magnitude of AR. We need to find alternative and unified evaluation metrics to compare models with different aggregation functions. We adopt the two feature importance measures used in [36] as our evaluation metrics: 1) gradient-based measures of feature importance, and 2) differences in model output induced by leaving features out. However, instead of studying the importance of single features, we are interested in the contribution of a single agent to model output. Consequently, we define a customized gradient-based measure as follows:

$$\tau_{g,i} = \frac{1}{2(n-1)T_h} \sum_{j=1, j\neq i}^{n} \left\| \frac{\partial \hat{\mathbf{y}}_{i,T_p}}{\partial \mathbf{x}_j} \right\|_{1,1},$$

where i is the index of the target agent, T_p is the number of predicted frames, $\hat{\mathbf{y}}_{i,T_p}$ is the predicted state of the target agent at the last frame, $\partial \hat{\mathbf{y}}_{i,T_p} / \partial \mathbf{x}_j$ is the partial Jacobian matrix of $\hat{\mathbf{y}}_{i,T_p}$ regarding the observed trajectory of the agent j. And $\|\cdot\|_{1,1}$ defines the entry-wise 1-norm which sums the absolute values of the matrix's entries. $\tau_{g,i}$ essentially measures the average gradient of the model output regarding the observations of the surrounding agents. If the model ignores social context, a small $\tau_{g,i}$ is expected. Also, we define a customized leave-one-out ADE metric as follows:

$$loo\text{ADE}_i = \frac{1}{n-1} \sum_{j=1, j\neq i}^{n} \text{ADE}\left(\hat{\mathbf{y}}(\mathbf{x}_{-j}), \hat{\mathbf{y}}(\mathbf{x})\right),$$

where $\mathbf{x}_{-j}$ denotes the observed trajectories with the agent j masked out. $loo\text{ADE}_i$ equals to the average ADE between the normal prediction output and the prediction output with each surrounding agent masked out. Similar to $\tau_{g,i}$, we expect a small $loo\text{ADE}_i$ if social posterior collapse occurs.

We conduct the ablation study on the vehicle prediction task. On both datasets, we repeated the experiments but replaced the models with their variants with different aggregation functions. We are mainly interested in comparing the values of τ_g and $loo\text{ADE}$ across the models with the same aggregation functions. Because the other aggregation functions do not encourage a sparse model structure, any surrounding agent could contribute to τ_g and $loo\text{ADE}$. It makes it less informative to compare them against the model with sparse attention. The results on the validation sets are summarized in Table 4 and 5. The trends in τ_g and $loo\text{ADE}$ are similar regardless of the aggregation functions. In most circumstances, our social-CVAE model attains the highest scores with large margins, whereas the VAE variant has the lowest τ_g and $loo\text{ADE}$. Meanwhile, the comparisons in prediction performance are also consistent with the case of sparse-GAMP. Therefore, we can conclude that social posterior collapse is not unique to the models with sparse-GAMPs.

We also observe that changing the formulation from VAE to CVAE always leads to an increase in τ_g and $loo\text{ADE}$ when the alternative aggregation functions are used. In contrast, the values could stay unchanged in the case of sparse-GAMP. We are then curious if merely changing the formulation can alleviate social posterior collapse when sparse-GAMP is not used. We take the softmax variant as an example and investigate its attention maps. However, simply computing the ratio of non-zero attention weights is meaningless because the attention is no longer sparse. To solve this problem, we refine our definition of AR as follows:

$$\text{AR}_\delta = \frac{\sum_{i=1,i\neq j}^{n} \mathbf{1}(\omega_{(i,j)} \geqslant \delta)}{n-1}, \quad \delta \in [0,1].$$

Now AR_δ is a function of a threshold value δ. Instead of looking at a single value at $\delta = 0$, we are interested in seeing how AR_δ changes when increasing δ from 0 to 1. In Fig. 6, we plot $\text{AR}_\delta(\%)$ versus the threshold δ for all the models with softmax and entmax functions on the Argoverse dataset. By switching to the conditional model, the softmax variant assigns relatively larger attention weights to surrounding agents. However, the increase in AR_δ mainly occurs at small threshold values. Compared to the social-CVAE models, the ratio of agents receiving large attention weights is lower. It is consistent with our argument that merely changing the formulation is insufficient to alleviate social posterior collapse. Another interesting observation is that the AR_δ curves for the two variants of social-CVAE models coincide when $\delta \geqslant 0.1$. It implies that our sparse graph attention does not prevent the model from identifying interacting agents. It just filters out agents that are recognized as irrelevant to maintain a sparse and interpretable attention map. We can also observe from the evaluated prediction metrics that the sparse graph attention does not interfere with the prediction performance. The social-CVAE models with either attention mechanism achieve similar prediction accuracy. In short, our sparse graph attention function provides a convenient and flexible toolkit that allows us to monitor and analyze social posterior collapse without compromising performance. Although we can still analyze the models without it, these universal metrics, i.e., τ_g and $loo\text{ADE}$, are computationally expensive, especially for evaluation at run time.

E.2 Effect of Environmental Information

In this section, we investigate the effect of environmental information on social posterior collapse. In Sec. 5, we find the models behave quite differently on the pedestrian prediction task and its vehicle counterpart. In our experiments on the ETH/UCY dataset, although switching to CVAE leads to a larger AR value, it does not boost prediction performance. Moreover, the auxiliary task neither improves prediction performance nor further increases the AR value. In the main text, we argue that it is because, unlike vehicles, pedestrians do not have a long-term dependency on their interaction. As a result, the model cannot benefit from encoding historical social context. However, we also adopt different input representations for the two prediction problems. In the vehicle prediction task, the input graphs have additional lanelet nodes, contributing to the difference in model behavior.

To answer this question, we trained models for vehicle prediction with the same representation as pedestrians, which means removing lanelet nodes and adding self-edges instead. We choose to study this problem on the highway merging subset of the INTERACTION dataset because the interaction between vehicles depends less on the map than urban driving. We follow the same practice to run five trials for each model variant and report all the metrics' mean and standard deviation. The results are summarized in Table 6. We also copy the results from Table 1 for convenient comparison.

We think social posterior collapse still occurs in the baseline VAE model. As social context affects highway driving to a larger degree than road structure, our social-CVAE model that encodes social

Table 4: INTERACTION Dataset Aggregation Function Comparison

Scene	Aggreg.	Model	τ_g ($\times 10^{-4}$)	looADE ($\times 10^{-3}$)	K = 6	
					minADE	minFDE
IS	max	VAE	0.39 ± 0.10	0.52 ± 0.20	0.28 ± 0.01	0.79 ± 0.01
		CVAE	3.86 ± 1.30	6.38 ± 2.76	$\mathbf{0.24 \pm 0.01}$	$\mathbf{0.69 \pm 0.01}$
		Ours	$\mathbf{14.8 \pm 5.39}$	$\mathbf{20.3 \pm 3.25}$	0.25 ± 0.01	0.70 ± 0.01
	softmax	VAE	0.90 ± 1.37	1.41 ± 2.35	0.26 ± 0.01	0.73 ± 0.02
		CVAE	5.00 ± 1.04	10.6 ± 2.16	$\mathbf{0.24 \pm 0.01}$	$\mathbf{0.67 \pm 0.01}$
		Ours	$\mathbf{33.6 \pm 19.0}$	$\mathbf{17.7 \pm 1.37}$	0.25 ± 0.02	0.70 ± 0.01
	entmax	VAE	0.05 ± 0.09	0.20 ± 0.42	0.26 ± 0.01	0.73 ± 0.02
		CVAE	0.02 ± 0.03	0.02 ± 0.03	$\mathbf{0.24 \pm 0.01}$	$\mathbf{0.68 \pm 0.01}$
		Ours	$\mathbf{29.6 \pm 2.50}$	$\mathbf{8.73 \pm 5.50}$	0.25 ± 0.01	0.70 ± 0.02
RA	max	VAE	0.15 ± 0.05	0.36 ± 0.17	0.30 ± 0.01	0.86 ± 0.02
		CVAE	5.12 ± 2.18	8.42 ± 2.41	$\mathbf{0.26 \pm 0.01}$	$\mathbf{0.74 \pm 0.01}$
		Ours	$\mathbf{36.3 \pm 12.4}$	$\mathbf{28.8 \pm 1.40}$	0.28 ± 0.01	0.76 ± 0.02
	softmax	VAE	1.62 ± 1.57	4.51 ± 3.30	0.27 ± 0.01	0.79 ± 0.02
		CVAE	8.63 ± 2.85	18.4 ± 1.18	$\mathbf{0.26 \pm 0.01}$	$\mathbf{0.73 \pm 0.01}$
		Ours	$\mathbf{40.6 \pm 10.2}$	$\mathbf{22.8 \pm 0.74}$	0.27 ± 0.01	$\mathbf{0.73 \pm 0.01}$
	entmax	VAE	0.00 ± 0.00	0.01 ± 0.01	0.26 ± 0.01	0.75 ± 0.01
		CVAE	11.8 ± 10.9	5.91 ± 5.70	$\mathbf{0.25 \pm 0.01}$	$\mathbf{0.72 \pm 0.01}$
		Ours	$\mathbf{71.0 \pm 71.4}$	$\mathbf{12.9 \pm 8.64}$	0.26 ± 0.01	0.73 ± 0.02
HM	max	VAE	0.45 ± 0.36	0.40 ± 0.18	0.20 ± 0.02	0.52 ± 0.05
		CVAE	37.8 ± 13.3	7.00 ± 0.68	0.15 ± 0.01	0.36 ± 0.01
		Ours	$\mathbf{196 \pm 53.0}$	$\mathbf{8.28 \pm 0.57}$	$\mathbf{0.13 \pm 0.01}$	$\mathbf{0.31 \pm 0.01}$
	softmax	VAE	8.89 ± 0.69	3.92 ± 2.68	0.17 ± 0.01	0.42 ± 0.02
		CVAE	47.0 ± 8.01	$\mathbf{11.5 \pm 0.53}$	0.15 ± 0.01	0.34 ± 0.01
		Ours	$\mathbf{176 \pm 36.0}$	9.42 ± 0.40	$\mathbf{0.14 \pm 0.01}$	$\mathbf{0.32 \pm 0.01}$
	entmax	VAE	10.7 ± 22.4	1.80 ± 3.36	0.16 ± 0.01	0.41 ± 0.04
		CVAE	16.4 ± 36.6	1.93 ± 4.14	0.15 ± 0.02	0.38 ± 0.04
		Ours	$\mathbf{205 \pm 158}$	$\mathbf{6.23 \pm 3.27}$	$\mathbf{0.13 \pm 0.02}$	$\mathbf{0.32 \pm 0.04}$

Table 5: Argoverse Dataset Aggregation Function Comparison

Aggreg.	Model	τ_g ($\times 10^{-3}$)	looADE ($\times 10^{-2}$)	K = 6	
				minADE	minFDE
max	VAE	0.11 ± 0.12	0.24 ± 0.31	1.47 ± 0.09	2.64 ± 0.07
	CVAE	3.91 ± 2.06	3.33 ± 1.82	1.22 ± 0.02	2.15 ± 0.05
	Ours	$\mathbf{13.5 \pm 2.35}$	$\mathbf{13.9 \pm 0.77}$	$\mathbf{1.18 \pm 0.04}$	$\mathbf{2.07 \pm 0.09}$
softmax	VAE	0.12 ± 0.06	0.29 ± 0.12	1.36 ± 0.11	2.40 ± 0.07
	CVAE	5.85 ± 1.64	5.62 ± 1.00	1.16 ± 0.01	1.98 ± 0.01
	Ours	$\mathbf{13.1 \pm 2.17}$	$\mathbf{8.83 \pm 1.27}$	$\mathbf{1.13 \pm 0.01}$	$\mathbf{1.95 \pm 0.03}$
entmax	VAE	0.02 ± 0.02	0.07 ± 0.06	1.29 ± 0.04	2.32 ± 0.05
	CVAE	0.01 ± 0.01	0.02 ± 0.01	1.19 ± 0.01	2.08 ± 0.02
	Ours	$\mathbf{7.30 \pm 2.95}$	$\mathbf{5.77 \pm 2.34}$	$\mathbf{1.15 \pm 0.02}$	$\mathbf{1.97 \pm 0.02}$

context can maintain consistent prediction accuracy without map information. On the other hand, removing lanelet nodes leads to a significant drop in the baseline VAE model's prediction accuracy. It indicates that the baseline VAE model does not utilize social context well but relies heavily on map

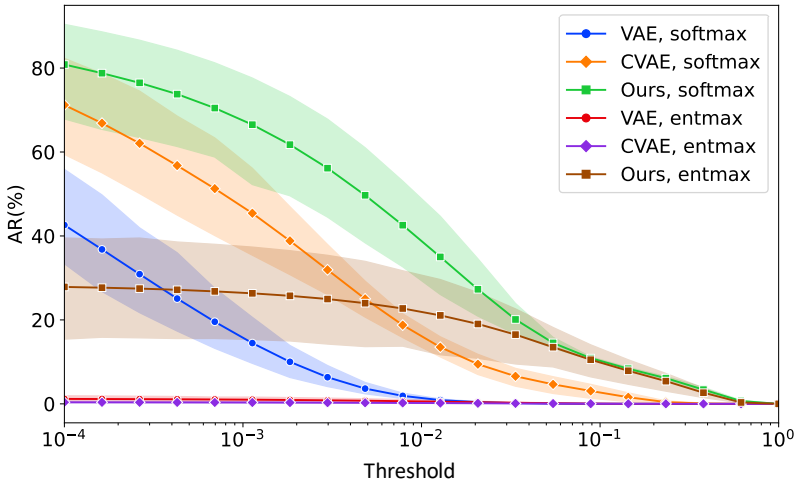

Figure 6: $\mathrm{AR}_\delta(\%)$ vs. Threshold on the Argoverse Dataset.

Table 6: INTERACTION Dataset without Map - HM

Map	Model	AR(%)	K = 1		K = 6	
			*min*FDE	*min*ADE	*min*FDE	*min*ADE
Yes	VAE	8.68 ± 8.36	0.87 ± 0.08	0.29 ± 0.03	0.42 ± 0.04	0.16 ± 0.01
	CVAE	5.42 ± 10.2	0.83 ± 0.03	0.28 ± 0.03	0.40 ± 0.05	0.16 ± 0.02
	Ours	$\mathbf{15.5 \pm 8.13}$	$\mathbf{0.65 \pm 0.09}$	$\mathbf{0.22 \pm 0.02}$	$\mathbf{0.32 \pm 0.04}$	$\mathbf{0.13 \pm 0.01}$
No	VAE	24.4 ± 14.4	0.98 ± 0.08	0.34 ± 0.02	0.51 ± 0.05	0.20 ± 0.02
	CVAE	$\mathbf{43.5 \pm 8.72}$	0.72 ± 0.02	0.25 ± 0.01	0.36 ± 0.01	0.15 ± 0.01
	Ours	33.7 ± 5.77	$\mathbf{0.62 \pm 0.01}$	$\mathbf{0.22 \pm 0.01}$	$\mathbf{0.32 \pm 0.01}$	$\mathbf{0.13 \pm 0.01}$

information, which is verified by its lower AR value. In particular, one of the five VAE models gets nearly zero AR value.

The analysis becomes intricate when it comes to comparing our model against the CVAE variant. Introducing the auxiliary task still improves prediction accuracy. Also, we note that the CVAE model itself has a lower prediction error after removing lanelet nodes. If the historical social context is the dominating factor determining prediction performance, we may draw two conclusions. First, the CVAE model suffers less from social posterior collapse under the graph representation without lanelet nodes. Second, the auxiliary prediction task can further encourage the model to encode social context. However, we do not have direct evidence showing that the auxiliary task encourages the model to encode social context —the auxiliary task does not lead to a larger AR value.

We think the reason behind this is that removing lanelet nodes makes the attention map less likely to become sparse. In most driving scenarios, the number of lanelet nodes dominates the number of agents. Removing lanelet nodes reduces the number of incoming edges for each agent node. Consequently, the agent-to-agent edges compete with a single self-edge to gain attention instead of numerous lanelet-to-agent edges. It implies that the format of representation affects how the model suffers from social posterior collapse. More importantly, it shows the limitation of using AR as the single metric to analyze social posterior collapse. It is particularly effective when the agent vertices are of a high degree, but the analysis may become inconclusive otherwise. In future studies, we will investigate other metrics and tools that can be applied to a broader range of problems.

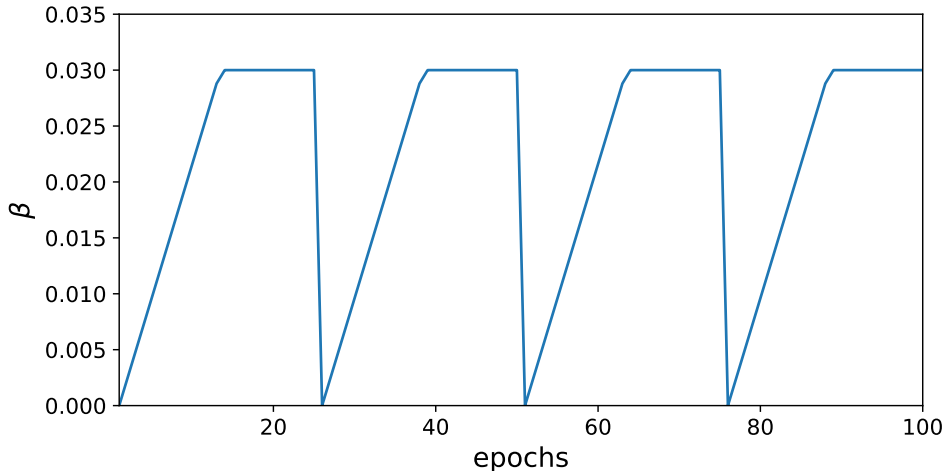

Figure 7: The cyclical annealing schedule adopted in our experiments. Each annealing cycle consists of twenty-five training epochs. The magnitude of β increases linearly from zero to the maximum value in the first half of the cycle, and then keeps unchanged in the remaining epochs. The maximum value of is set to be 0.03, which is the value used in previous experiments with constant β.

E.3 Effect of KL Annealing

In Sec. 2.2, we argue that social posterior collapse is different from the well-known posterior collapse issue, and typical techniques that address posterior collapse, e.g., KL annealing, may not be effective in mitigating social posterior collapse. In this section, we test if one of the annealing methods —cyclical annealing schedule [19] —can effectively alleviate social posterior collapse. Again, we use the HM scenario from the INTERACTION dataset as an example to study this problem. When training the baseline VAE and CVAE models, we incorporate the cyclical annealing schedule plotted in Fig. 7 to adjust the magnitude of β over training epochs. The results are summarized in Table 7. The cyclical annealing schedule neither improves prediction performance nor raises AR value consistently. The VAE model with a cyclical schedule does have a larger average AR value. However, two out of the five trials attains zero AR, which causes the large standard deviation. In summary, the cyclical annealing schedule does not alleviate the social posterior collapse issue in the experiments, which is consistent with our previous argument.

F Testing Results on Argoverse and ETH/UCY

In this section, we report the testing results on Argoverse and ETH/UCY datasets and compare the results with other models in the literature. It should be noted that achieving state-of-the-art performance is not our target. The testing results are provided to give the audience a complete picture

Table 7: INTERACTION Dataset KL Annealing Experiment - HM

Model	Cycling	AR(%)	K = 1		K = 6	
			$minFDE$	$minADE$	$minFDE$	$minADE$
VAE	No	8.68 ± 8.36	0.87 ± 0.08	0.29 ± 0.03	0.42 ± 0.04	0.16 ± 0.01
	Yes	17.7 ± 16.5	0.96 ± 0.10	0.33 ± 0.03	0.52 ± 0.04	0.20 ± 0.01
CVAE	No	5.42 ± 10.2	0.83 ± 0.03	0.28 ± 0.03	0.40 ± 0.05	0.16 ± 0.02
	Yes	7.65 ± 10.5	0.84 ± 0.07	0.28 ± 0.02	0.41 ± 0.03	0.16 ± 0.01
Ours	-	$\mathbf{15.5 \pm 8.13}$	$\mathbf{0.65 \pm 0.09}$	$\mathbf{0.22 \pm 0.02}$	$\mathbf{0.32 \pm 0.04}$	$\mathbf{0.13 \pm 0.01}$

Table 8: Argoverse Dataset Testing Results

Model	K = 1		K = 6	
	$minFDE$	$minADE$	$minFDE$	$minADE$
TNT [30]	4.9593	2.1740	1.4457	0.9097
LaneGCN [31]	3.7786	1.7060	1.3640	0.8679
LaneRCNN [32]	3.6916	1.6852	1.4526	0.9038
TPCN [48]	**3.6386**	**1.6376**	**1.3535**	**0.8546**
Social-CVAE (Ours)	4.2748	1.9276	2.4881	1.3568

Table 9: ETH/UCY Testing Results

Model	$minADE/minFDE$, K = 20					
	ETH	HOTEL	UNIV	ZARA1	ZARA2	Average
SGAN[2]	0.81/1.52	0.72/1.61	0.60/1.26	0.34/0.69	0.42/0.84	0.58/1.18
SoPhie[42]	0.70/1.43	0.76/1.67	0.54/1.24	0.30/0.63	0.38/0.78	0.54/1.15
Transformer-TF[49]	0.61/1.12	0.18/0.30	0.35/0.65	0.22/0.38	0.17/0.32	0.31/0.55
STAR[50]	0.36/0.65	0.17/0.36	0.31/0.62	0.26/0.55	0.22/0.46	0.26/0.53
PECNet[7]	0.54/0.87	0.18/0.24	0.35/0.60	0.22/0.39	0.17/0.30	0.29/0.48
Trajectron++[4]	0.39/0.83	0.12/0.21	**0.20**/**0.44**	**0.15**/0.33	**0.11**/0.25	0.19/0.41
AgentFormer[9]	**0.26**/**0.39**	**0.11**/**0.14**	0.26/0.46	**0.15**/**0.23**	0.14/**0.24**	**0.18**/**0.29**
VAE (Ours)	0.59/0.90	0.18/0.26	0.31/0.54	0.21/0.37	0.17/0.31	0.29/0.48
CVAE (Ours)	0.56/0.84	0.18/0.26	0.33/0.58	0.21/0.38	0.18/0.33	0.29/0.48
Social-CVAE (Ours)	0.64/0.99	0.18/0.27	0.35/0.62	0.21/0.37	0.19/0.34	0.32/0.52

of the model. For the Argoverse dataset, we collect the results of other models from the leaderboard. Even though there are other models on the leaderboard with better performance, we focus on those from published papers to get insights on the reasons behind the performance gap. For the ETH/UCY dataset, we collect the results from the corresponding papers.

On the Argoverse dataset, the performance of our social-CVAE model is similar to TNT [30], which adopted a similar map representation with us, in terms of $minFDE$ and $minADE$ when $K = 1$. It implies that we could improve our performance by adopting alternative map representations, such as those proposed in LaneGCN [31] and TPCN [48]. Another observation is that compared to the other approaches, our model has a large performance margin in the case of $K = 6$. It is because sampling from a high-dimensional continuous distribution is inefficient, making it less effective in modeling the multi-modality of driving behavior. Using discrete latent space as in [6] and [4] or conditioning the latent space on goal points [7] could be better options under the CVAE framework. On the ETH/UCY dataset, the VAE variant of our model has better or comparable performance to all the other models except for Trajectron++ and AgentFormer. As mentioned in the main text, we are particularly interested in the formulation of AgentFormer. In future studies, we will incorporate a similar autoregressive decoder into our model and investigate social posterior collapse under this setting.