# Generative Causal Representation Learning for Out-of-Distribution Motion Forecasting

**Shayan Shirahmad Gale Bagi** [1]    **Zahra Gharaee** [2]    **Oliver Schulte** [3]    **Mark Crowley** [1]

## Abstract

Conventional supervised learning methods typically assume i.i.d samples and are found to be sensitive to out-of-distribution (OOD) data. We propose Generative Causal Representation Learning (GCRL) which leverages causality to facilitate knowledge transfer under distribution shifts. While we evaluate the effectiveness of our proposed method in human trajectory prediction models, GCRL can be applied to other domains as well. First, we propose a novel causal model that explains the generative factors in motion forecasting datasets using features that are common across all environments and with features that are specific to each environment. Selection variables are used to determine which parts of the model can be directly transferred to a new environment without fine-tuning. Second, we propose an end-to-end variational learning paradigm to learn the causal mechanisms that generate observations from features. GCRL is supported by strong theoretical results that imply identifiability of the causal model under certain assumptions. Experimental results on synthetic and real-world motion forecasting datasets show the robustness and effectiveness of our proposed method for knowledge transfer under zero-shot and low-shot settings by substantially outperforming the prior motion forecasting models on out-of-distribution prediction.

## 1. Introduction

Human Trajectory Prediction (HTP) is a valuable task in many applications such as infrastructure design, traffic operations, crowd abnormality detection systems, evacuation situation analysis, and autonomous vehicles (Kothari et al., 2022). It is also more challenging compared to other types of agent trajectory prediction, such as vehicles, due to the complex behaviour of humans in different environments. Most of the proposed models in the literature of motion forecasting (Gupta et al., 2018; Kosaraju et al., 2019; Mangalam et al., 2020; Lee et al., 2017; Huang et al., 2019; Mohamed et al., 2020; Salzmann et al., 2020) rely on statistical inference, which has two important shortcomings. First, the learnt representations are not explainable and, therefore, are ineffective to transfer into a new environment. Secondly, statistical inference is susceptible to learn spurious correlations, which can significantly degrade the performance of the models in presence of a domain shift or measurement noise (Liu et al., 2022). In this work we aim to tackle these shortcomings from a causality perspective.

Causal representation learning has attracted much attention in various applications (Liu et al., 2021; Zhang et al., 2020a; Schölkopf et al., 2021; Liu et al., 2022). Causal inference eliminates the confounding bias in predictions, (Pearl et al., 2016) which is a common phenomenon in motion forecasting since inputs and targets come from the same interdependent time series. Furthermore, models based on causal inference learn meaningful features that do not rely on spurious correlations. The coexistence of target trajectories and observation noise, is an example of spurious correlation in motion forecasting (Liu et al., 2022). Causality can also identify the generative factors in the dataset (Suter et al., 2019). Physical laws, social norms, and motion styles are examples of generative factors in HTP. Under covariate shifts, a model that learns the causal structure of the data would require many fewer samples to adapt to the new environment because most of the modules can be reused without further training. This expectation is consistent with the Sparse Mechanism Shift hypothesis in causal representation learning (Schölkopf et al., 2021).

To this end, we propose Generative Causal Representation Learning (GCRL) that leverages causality to increase the identifiability and robustness of current motion forecasting models. While our target application is motion forecasting, our proposed solution can be applied to domain adapta-

---

[1]Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Canada [2]Department of Systems Design Engineering, University of Waterloo, Waterloo, Canada [3]School of Computing Science, Simon Fraser University, Burnaby, Canada. Correspondence to: Shayan Shirahmad Gale Bagi <sshirahm@uwaterloo.ca>.

tion in other applications. Oftentime, the training dataset is collected from different locations such as the ETH-UCY dataset (Lerner et al., 2007; Pellegrini et al., 2010). Considering this, we introduce a causal model that includes the invariant features, which are common across all types of environments, and variant features, which are environment-specific. Our causal model includes selection variables (Pearl & Bareinboim, 2014) to directly transfer some parts of the model to a new domain without fine-tuning. Second, we propose a new learning paradigm that can learn invariant and variant features simultaneously and eliminate the confounding bias using a backdoor adjustment (Pearl et al., 2016). Previous work (Liu et al., 2022) uses Invariant Risk Minimization (IRM), (Arjovsky et al., 2019) which requires solving a very complex minimization problem. Third, our proposed method uses a generative process to infer latent representations, which can capture multi-modal trajectory distributions, without adding any noise to the representations as proposed in (Huang et al., 2019).

**Contributions** Our key contributions can be summarized as follows:

**(1)** A learning paradigm which enables end-to-end training and eliminating confounding bias.

**(2)** Augmented causally factorized model to enable direct transportability and reusability of some parts of the model.

**(3)** A generative process to produce latent representations to tackle multi-modality of trajectories.

Our code is available at https://github.com/sshirahmad/GCRL.

## 2. Related Work
### 2.1. Causal Representation Learning
As pointed out in (Teshima et al., 2020), a transfer assumption is required in domain adaptation. Most of the methods in computer vision applications have parametric distribution families, hence, it is assumed that domain shifts affect the parameters of the distributions. Similar to our approach, INFER (Zhang et al., 2020b) uses an augmented version of Directed Acyclic Graphs (DAGs) to represent the casual structure of data. Their method assumes that the targets are not available in the test domain and perform unsupervised domain adaptation (DA) using a conditional GAN. CAE (Yang et al., 2021) proposes an end-to-end framework to jointly learn a model for causal structure learning and a deep autoencoder model. Markov Blanket features (Yu et al., 2020) are then extracted from the learnt causal model to be used in the test domain. CSG (Liu et al., 2021) leverages the Independent Causal Mechanisms (ICM) principle (Schölkopf et al., 2021) to learn the causal model for image classification task. In CSG, covariate shift is assumed to manifest itself in the priors of the latent variables in the form of spurious correlations.

CONTA (Zhang et al., 2020a) tackles the confounding bias in causal models by using the backdoor criterion (Pearl et al., 2016) for image segmentation task. To obtain invariant conditional distributions applicable in the test domain (Magliacane et al., 2018) uses the selection diagrams. CGNN (Goudet et al., 2017), on the other hand, applies maximum mean discrepancy between the generated distribution by a set of generative models and true distribution of the data to approximate the causal model of the data. DAG-GNN (Yu et al., 2019) uses a variational autoencoder to learn the adjacency matrix of the DAG that is supposed to represent the causal model of the data. In (Ke et al., 2019), the causal structure is learnt from both observational data and unknown interventions. In (Suter et al., 2019), causal disentanglement is studied and a set of metrics are introduced to measure the robustness of the learnt features.

Most of these methods attempt to learn the causal model of the data, however, in motion forecasting we can hypothesize a causal model based on the domain knowledge. Furthermore, identifying true causal model of the data is extremely challenging and impossible in an unsupervised setting with observational data (Locatello et al., 2019). Hence, the previous methods have not investigated causal discovery in real-world time-series data, which has exclusive challenges.

### 2.2. Motion Forecasting
Most of the deep-learning based motion forecasting models consist of an Encoder, an Interaction module, and a Decoder, (Kothari et al., 2022) which we will refer to as EID architecture. STGAT (Huang et al., 2019) uses a graph attention network to model the human-human interactions and LSTM cells to encode past trajectories and to predict future trajectories. On the other hand, STGCNN (Mohamed et al., 2020) consist of CNN layers only which significantly improves inference speed while performing on par with its RNN counterparts in terms of prediction error. Social-GAN (Gupta et al., 2018) focuses on the multi-modality of human trajectories and uses GANs to generate multiple future trajectories per sample. Social-BiGAT (Kosaraju et al., 2019) models both social and physical interactions using images of the environment and trajectories of pedestrians. DESIRE (Lee et al., 2017) proposes a conditional variational autoencoder (CVAE) to tackle the multi-modality of human trajectories. Trajectron++ (Salzmann et al., 2020) focuses on the dynamic constraints of agents when predicting trajectories and models not only the interactions between humans but also interactions of humans with other agents in the scene. PECNet (Mangalam et al., 2020) uses a CVAE to obtain a distribution for the endpoints of the pedestrians, which is employed to predict the future trajectories conditioned on the endpoints. According to the recent works (Liu et al., 2022; Chen et al., 2021), there are varying types of noise and intrinsic biases in the data resulting from human behaviour in different environments. The performance of the current

motion forecasting models is negatively affected when transferred to a new environment or in the presence of a noise (Liu et al., 2022). Furthermore, there are some confounding factors in motion forecasting applications, which can lead to biased predictions (Chen et al., 2021). Our work focuses on the robustness aspect of the motion forecasting problem. We propose a method applicable to motion forecasting models with an EID architecture, which will be discussed in the following sections.

## 3. Proposed Method

### 3.1. Formalism of Motion Forecasting

Consider a motion forecasting problem in a multi-agent environment with $M$ agents. Let's denote the states of the agents as $S_t = \{s_t^1, s_t^2, ..., s_t^M\}$ where $s_t^i = (x_t^i, y_t^i)$ are the 2D coordinates of agent $i$ at time $t$. The objective is to predict the states of agents $T_{pred}$ time steps into the future from observations of the previous $T_{obs}$ time steps. The model takes as input $X^i = \{s_1, s_2, ..., s_{T_{obs}}\}$ and predicts $Y^i = \{s_{T_{obs}+1}, s_{T_{obs}+2}, ..., s_{T_{obs}+T_{pred}}\}$ for every agent $i$.



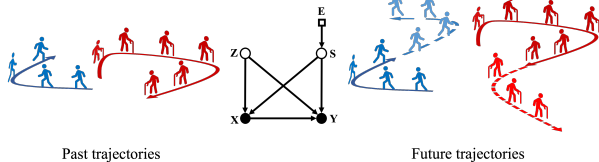Past trajectories            Future trajectories

Figure 1: The proposed causal model (center). Filled circles are observed variables and empty shapes are the unobserved variables. $X$ and $Y$ represent past trajectories and future trajectories to be predicted, respectively. $Z$ represents invariant features common across domains, such as physical laws, while $S$ represents variant features specific to each environment, such as motion styles. Finally, $E$ is the selection variable. Conditioning on $E$ allows us to switch between environments.

Deep learning based motion forecasting models are often composed of three modules; an encoder, an interaction module and a decoder (Kothari et al., 2022). The encoder takes previously observed states $X^i$ and produces representations of the observed states. The interaction module receives the representations from the encoder to capture the social interactions (human-human), the physical interactions (human-space), or both. Eventually, the decoder takes as input, the interaction vectors and predicts the future states of the agent $Y^i$. It is worth mentioning that any type of neural network, which can handle sequence-to-sequence modeling such as Recurrent Neural Networks (RNNs) or temporal Convolutional Neural Networks (CNNs) could be employed as the Encoder and the decoder of the model.

Training data is usually collected from a set of $K$ environ-

ments or domains, $E = \{e_1, e_2, ..., e_k\}$. An example set of environments could be 'Pedestrians crossing an intersection', 'People standing in front of a department store', or 'Pedestrians walking along the sidewalk'. In real-world datasets, it is common for the training and test environments to differ significantly (Chen et al., 2021), such as in the widely-used ETH-UCY benchmark. Therefore, the i.i.d assumption does not always hold in practice and Empirical Risk Minimization (ERM) techniques cannot be used to train the neural networks. In the following section, we briefly present some concepts of causality applied in our proposed method.

### 3.2. Background in Causality

Structural Causal Models (SCMs) are a way of describing causal features and their interactions, which are represented by Directed Acyclic Graphs (DAG) (Pearl et al., 2016). We say that $X$ is a direct cause of $Y$ when there is a directed edge from $X$ to $Y$ in the DAG. The cause and effect relation $X \rightarrow Y$ tells us that changing the value of $X$ can result in a change in the value of $Y$, but that the reverse is not true. A causal model receives as inputs a set of qualitative causal assumptions (A), a set of queries concerning the causal relations among variables (Q), and experimental or non-experimental data (D), presumably consistent with (A). A causal model makes predictions about the behavior of a system. The outputs of a causal model are then a set of logical implications of (A), data-dependent claims (C) represented by the magnitude or likelihoods of the queries (Q), and a list of testable statistical implications (T) (Pearl & Bareinboim, 2014).

A major drawback of deep learning models using statistical inference is that they do not consider the confounding bias. Normally, we would predict $p(Y|X)$, however, in the presence of confounding variables, $p(Y|X)$ cannot capture the true causal effect of $X$ on $Y$ and the predictions will be erroneous. Therefore, we are interested in the causal query $p(Y|do(X))$ which eliminates the confounding bias by surgically intervening on $X$. This causal query can be calculated using the Backdoor Criterion (Pearl et al., 2016). Our set of causal assumptions are causal sufficiency and faithfulness (Yu et al., 2020). Since we do not assume i.i.d samples, we use an augmented version of causal models called Selection Diagrams (Pearl & Bareinboim, 2014) to discriminate between samples of different domains. Selection variables $S$ are introduced to model the variations in each domain. Every member of $S$ corresponds to a mechanism by which the two populations differ. Therefore, we can condition on the values of $S$ for switching between domains (Pearl & Bareinboim, 2014).

### 3.3. Causal Formalism

Our proposed SCM for motion forecasting is shown in Figure 1. There are two causal variables in motion forecasting,

that can affect the trajectories of the pedestrians: invariant features and variant features (Liu et al., 2022). *Invariant features* do not vary across domains but can influence the trajectories of the pedestrians. These features can be associated with physical laws, traffic laws, social norms, and etc. In contrast, *variant features* vary across domains and can be associated with the motion styles of the pedestrians in an environment (Liu et al., 2022). In our proposed causal model we consider four endogenous variables: $S$ to represent variant features, $Z$ for invariant features, $X$ for past trajectories, and $Y$ for future trajectories.

We also introduce an additional exogenous variable $E$ as the selection variable (Pearl & Bareinboim, 2014) to account for the changing factors in each environment. The selection variable acts as an identifier of an environment such that the conditional probability of $p(X, Y | E = e_1)$ represents the distribution of the samples in an environment having the selection variable of $e_1$. In other words, we assume that all members of the dataset are sampled from a parent distribution over $X$, $Y$, and $E$. Furthermore, we assume that the proposed model is *causally sufficient*. That is, it explains all the dependencies without adding further causal variables. To build a causal graph, we reason about causal edges connecting the causal variables:

**(1)** There should be an edge from $S$ to $X$ and $Y$ because motion styles can influence the speed of the pedestrians.

**(2)** There should be an edge from $Z$ to $X$ and $Y$ because social norms can influence how closely pedestrians move next to each other.

**(3)** There should be an edge from $X$ to $Y$ because the location in the past determines where the pedestrian is going to be in the future.

**(4)** $S$ varies in each domain, hence, there should be an edge from selection variable $E$ to $S$ to account for all the changing factors in each domain.

### 3.4. Learning Latent Variables with Variational Inference

According to Figure 1, $S$ and $Z$ confound the causal effect of $X$ on $Y$ and the backdoor paths are $Y \leftarrow S \rightarrow X$ and $Y \leftarrow Z \rightarrow X$. Therefore, we need to use the backdoor criterion to eliminate the confounding effect. To this end, the causal effect of $X$ on $Y$ (i.e., $p(Y|do(X))$) is calculated. Before calculating this causal query, we need to recall the *S-admissibility* criterion:

**Definition 3.1.** "A set $T$ of variables satisfying $(Y \perp\!\!\!\perp S | T, X)$ in $D_{\bar{X}}$ will be called *S-admissible* (with respect to the causal effect of $X$ on $Y$)." (Pearl & Bareinboim, 2014)

Where $D_{\bar{X}}$ denotes an augmented SCM in which we intervene on X (i.e., the arrows entering X are deleted). According to this definition, in Figure 1, the set $\{S\}$ will be an E-admissible set, which means once we know about $S$, the

predictions ($Y$) will not be dependent on the environment ($E$) we're in.

We can now proceed to calculate $p(Y|do(X))$:

$$
\begin{aligned}
& p(Y|do(X)), E = e_i) \\
&= \int p(Y|X, S, Z, E)p(S|E)p(Z|E)dsdz \\
&= \int p(Y|X, S, Z)p(S|E)p(Z)dsdz \\
&= E_{p(S|E), p(Z)}[p(Y|X, S, Z)],
\end{aligned}
\tag{1}
$$

where $p(Y|do(X)), E = e_i)$ is the causal effect of $X$ on $Y$ in environment $i$. The first line follows from the backdoor criterion and the second line follows from the fact that $S$ is E-admissible (Pearl & Bareinboim, 2014) and $X$ is a collider. Equation 1 implies that in order to calculate the causal effect of $X$ on $Y$ in every environment, we need to stratify over $S$ and $Z$. Although we don't know the true posterior of the latent variables, we will approximate them using mean-field variational inference (Goodfellow et al., 2016).

The standard log-likelihood to train the model is given by:

$$
\max_p E_{p^*(x, y)}[\log p(x, y)]
\tag{2}
$$

where $p^*(x, y)$ is the distribution of samples in the dataset. Calculating $p(x, y)$ is intractable since $p(x, y) = \sum_{e \in E} \int (p(x, y, s, z, e)dsdz$ where $E$ is the set of environments. Mean-field variational inference is used to approximate the true posterior distribution of latent variables.

For this purpose, the *Evidence Lower Bound (ELBO)* function is used to train the model:

$$
\max_{p,q} E_{p^*(x, y)} \left[ E_{q(s, z, e|x, y)} \left[ \log \frac{p(x, y, s, z, e)}{q(s, z, e|x, y)} \right] \right]
\tag{3}
$$

Theoretically, the ELBO function will drive $q(s, z, e|x, y)$ towards its target $p(s, z, e|x, y)$ and the objective function in Equation 3 will become Equation 2. The approximate model is still intractable because we don't have access to the future trajectories in the test domain. Therefore, we replace it with $q(s, z, e, y|x)$ and the loss function becomes:

$$
\max_{p,q} E_{p^*(x)} E_{p^*(y|x)}[\log q(y|x)] +
$$

$$
E_{p^*(x)} \left[ E_{q(s, z, e, y|x)} \left[ \frac{p^*(y|x)}{q(y|x)} \log \frac{p(x, y, s, z, e)}{q(s, z, y, e|x)} \right] \right]
\tag{4}
$$

Assuming that $q(y|x)$ is Gaussian, the first term in the loss function of Equation 4 would be the negative of Mean Squared Error (MSE). Eventually, this term will drive $q(y|x)$ towards $p^*(y|x)$ and the second term will become a lower bound of $\log p(x)$ as stated in Theorem 3.2, which we prove below.

**Theorem 3.2.** *Let $p(x, y, s, z, e)$ be the joint distribution of latent and observed variables and $q(s, z, y, e|x)$ be the approximate posterior of latent variables and future trajectories given the past trajectories in GCRL, a lower bound on the log-likelihood of past trajectories is:*

$$E_{q(s,z,e,y|x)} \left[ \log \frac{p(x, y, s, z, e)}{q(s, z, y, e|x)} \right] \leq \log p(x)$$

*Proof.* According to the Causal Markov condition (Schölkopf et al., 2021), we can factorize $q(s, z, e, y|x) = q(y|x, s, z)q(z|x)q(s|x, e)q(e|x)$ and $p(x, s, z, y, e) = p(y|x, s, z)p(x|s, z)p(s|e)p(z)p(e)$. The first approximate model, $q(y|x, s, z)$, can be replaced with $p(y|x, s, z)$ since it is known. Secondly, since $q(s|x, e)$ is an approximation of its true posterior, we assume it to be $q(s|x)$. Therefore:

$$\begin{aligned}
&E_{q(s,z,y,e|x)} \left[ \log \frac{p(x, y, s, z, e)}{q(s, z, y, e|x)} \right] = \\
&E_{q(s,z,e|x)} \left[ \log \frac{p(x|s, z)p(s|e)p(z)p(e)}{q(z|x)q(s|x)q(e|x)} \right] = \\
&E_{q(s|x),q(z|x)} \left[ \log p(x|s, z) \right] - \\
&KL(q(z|x)||p(z)) - KL(q(s|x)||p(s)) - \\
&KL(q(e|x)||p(e|s)) \leq \log p(x)
\end{aligned} \tag{5}$$

The detailed proof is given in Appendix A. $\square$

As shown in Equation 5, $q(e|x)$ will be driven towards $p(e|s)$, hence, it is sensible to model $q(e|x)$ with $p(e|s)$. By leveraging the Causal Markov condition and replacing $q(e|x)$ with $p(e|s)$, we can obtain a lighter inference model (q) and the loss function in Equation 4 becomes:

$$\begin{aligned}
\max_{p,q} E_{p^*(x,y)} \Bigg[ &\log q(y|x) + \frac{1}{q(y|x)} \\
&E_{q(s|x),q(z|x)} \left[ p(y|x, s, z) \log \frac{p(x|s, z)p(s)p(z)}{q(s|x)q(z|x)} \right] \Bigg],
\end{aligned} \tag{6}$$

where $p(s) = \sum_{e \in E} p(s|e)p(e)$ which means that $S$ has a Gaussian mixture prior and $q(y|x) = E_{q(s|x),q(z|x)}[p(y|x, s, z)] = p(y|do(x))$ which can be calculated by ancestral sampling. The expectations in Equation 6 can be estimated using the Monte-Carlo method after applying the re-parametrization trick (Kingma & Welling, 2014). Consequently, GCRL learns: **(1)** To minimize the distance between groundtruth future trajectories and predicted future trajectories via maximizing $\log q(y|x)$, **(2)** To eliminate the confounding effect by estimating the causal effect of $X$ on $Y$ via $p(y|do(x))$, **(3)** To reconstruct past trajectories via maximizing $\log p(x|s, z)$, **(4)** Invariant representations via maximizing $\log \frac{p(z)}{q(z|x)}$, **(5)** Variant representations via maximizing $\log \frac{p(s)}{q(s|x)}$. Furthermore, since GCRL learns to predict the future trajectories with a generative approach, it can tackle the multi-modality of trajectories.

Gaussian Mixture priors have been previously used in popular VAE models (Jiang et al., 2017; Johnson et al., 2016). Variational models are susceptible to poor local minima and posterior collapse, it is important to show the identifiability of latent variables (Kivva et al., 2022). Variational models with GMM priors are proven to be identifiable (Kivva et al., 2022), hence, using a Gaussian Mixture prior for $S$ aligns with the theory as well. Furthermore, Gaussian Mixture Models (GMMs) are universal approximators, hence, $q(s|x)$ will be capable of producing arbitrary variant features. To obtain a better likelihood for latent variables, we use coupling layers (Dinh et al., 2017) to learn rich priors $p(s|e_i)$ and $p(z)$ from simple priors such as standard Normal distributions. These priors are referred to as *flow priors* and are also used in VAEs (Xu et al., 2019).

A general overview of our model is shown in Figure 2. The encoded past trajectories are used to model the approximate posteriors of the latents. We sample from these posteriors to reconstruct the past trajectories and predict the future trajectories. As shown in Figure 2, any motion forecasting model with an EID architecture can be used with our method provided that the decoder is a piece-wise affine injective function as this is required to obtain the weakest form of identifiability (i.e., identifiability up to an affine transformation (Kivva et al., 2022)).

### 3.5. Domain Adaptation Method

After learning the causal mechanisms and causal variables using Equation 6, we know that $q(z|x)$ will generate representations with a single Gaussian distribution and $q(s|x)$ will generate representations with a Gaussian Mixture Model (GMM) distribution. Therefore, as illustrated in Figure 3, all representations generated by $q(z|x)$ will be in the same range, whereas the representations of $q(s|x)$ will form clusters, each modeled by a component of the GMM. Since $Z$ is invariant, we can directly transfer it to the new domain without any fine-tuning. However, $S$ can be interpreted as a weighted sum of the representations learnt from different environments of the training domains, which may be used in the test domains as well. Depending on how related the test domains are to the training domains, we may need to fine-tune the components of the GMM and obtain a new prior for $S$. Thus, to fine-tune the model at inference time, we reuse the loss function in Equation 6 without the regularizing $Z$ posterior by omitting $q(z|x)$. Eventually, $q(s|x)$ will be driven towards the new prior and compensate for the domain shift in the test domain. The models to predict future trajectories $p(y|x, s, z)$ and to reconstruct past trajectories $p(x|s, z)$ also needs to be fine-tuned as the samples of $q(s|x)$ will be updated.
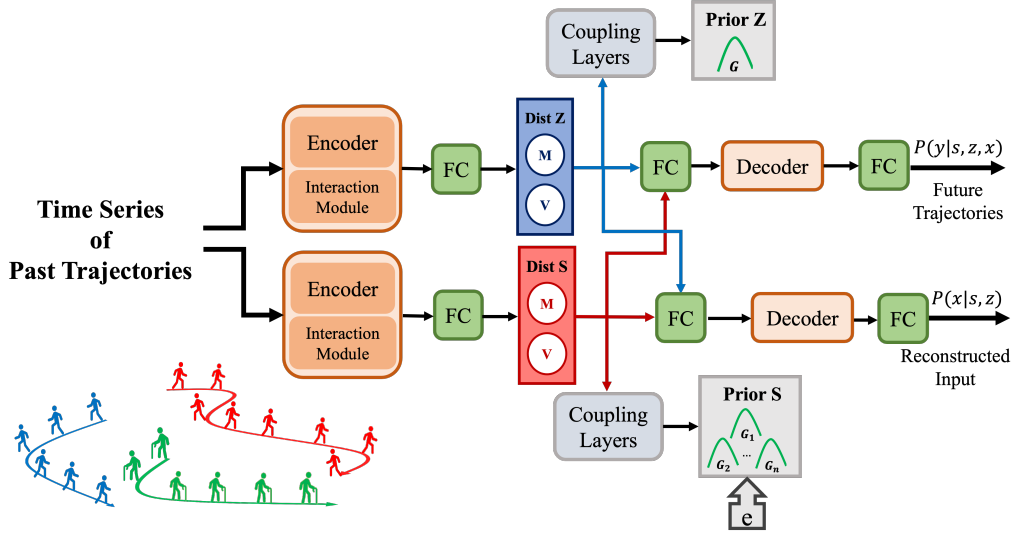
Figure 2: A general overview of the proposed method. The approximate posteriors of the latents are estimated using the encoded past trajectories and the priors of the latents are calculated using the coupling layers. The encoder, interaction module, and decoder of the model can be replaced by motion forecasting models that use an EID architecture.

Consequently, we only fine-tune the models of $q(s|x)$, $p(s)$, $p(y|x,s,z)$, and $p(x|s,z)$ while the models for, $p(z)$, and $q(z|x)$ can be arbitrarily complex as it is not required to fine-tune them in the test domain, but all the other models should be as simple as possible.
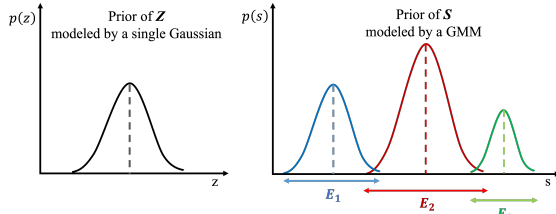


Figure 3: The priors of variant and invariant features. $E_i$ represents environment $i$.

## 4. Experiments

In our experiments we evaluate our models using two metrics often used in evaluation of the motion forecasting models:

$$ADE = \frac{\sum_{i=1}^{n} \sum_{t=T_{obs}+1}^{T_{pred}} ((\hat{x}_i^t - x_i^t)^2 + (\hat{y}_i^t - y_i^t)^2)^{\frac{1}{2}}}{n(T_{pred} - (T_{obs} + 1))} \tag{7}$$

$$FDE = \frac{\sum_{i=1}^{n} ((\hat{x}_i^{T_{pred}} - x_i^{T_{pred}})^2 + (\hat{y}_i^{T_{pred}} - y_i^{T_{pred}})^2)^{\frac{1}{2}}}{n}, \tag{8}$$

where $\hat{x}_i^t$ and $\hat{y}_i^t$ are the *predicted* horizontal and vertical coordinates of the pedestrian at time step $t$, while $x_i^t$ and

$y_i^t$ are the *actual* horizontal and vertical coordinates of the pedestrian at time step $t$.

### 4.1. Datasets

**ETH-UCY dataset**  This dataset contains the trajectory of 1536 detected pedestrians captured in five different environments {*hotel, eth, univ, zara1, zara2*}. All trajectories in the dataset are sampled every 0.4 seconds. Following the experimental settings of (Liu et al., 2022; Chen et al., 2021; Huang et al., 2019), we also use a leave-one-out approach for training and evaluating our model so to predict the future 4.8 seconds (12 frames), we utilize the previously observed 3.2 seconds (8 frames).

**Synthetic dataset**  This dataset published in (Liu et al., 2022) contains the trajectories of pedestrians in circle-crossing scenarios (Chen et al., 2019) where the minimum separation distance of pedestrians differ in each domain. There are 8 domains in the dataset with separation distances {0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8} meters. Each domain contains 10,000 trajectories for training, 3,000 trajectories for validation, and 5,000 trajectories for testing.

### 4.2. Robustness

To evaluate the robustness of our model in the presence of spurious correlations, we compare our method with (Liu et al., 2022; Chen et al., 2021). For a fair comparison, we use the STGAT (Huang et al., 2019) as our baseline model. Although, ETH-UCY contains five environments, it is not trivial to pinpoint the shifts in each environment. Therefore, we add a third dimension to the coordinates of

Table 1: Robustness of different methods in the ETH-UCY dataset with controlled spurious correlation.

| METHOD | ADE/FDE $\alpha = 8$ | ADE/FDE $\alpha = 16$ | ADE/FDE $\alpha = 32$ | ADE/FDE $\alpha = 64$ |
|---|---|---|---|---|
| BASELINE (HUANG ET AL., 2019) | **0.80/1.37** | 2.15/3.80 | 2.64/4.44 | 2.68/4.48 |
| COUNTERFACTUAL (CHEN ET AL., 2021) | 0.80/1.59 | 1.62/2.68 | 2.32/3.90 | 2.68/4.52 |
| INVARIANT $\lambda = 1.0$ (LIU ET AL., 2022) | 0.94/1.65 | 1.04/**1.76** | 1.52/2.55 | 1.96/3.35 |
| INVARIANT $\lambda = 3.0$ (LIU ET AL., 2022) | 0.91/1.67 | 0.99/1.87 | 1.18/2.20 | 1.27/2.33 |
| INVARIANT $\lambda = 5.0$ (LIU ET AL., 2022) | 0.98/1.79 | 1.00/1.83 | 1.06/1.90 | 1.56/2.58 |
| GCRL (OURS) | 0.97/1.8 | **0.97**/1.8 | **0.97**/1.8 | **0.97/1.8** |

the pedestrians, which measures observation noise and is modeled as in (Liu et al., 2022):

$$\gamma_t := (\dot{x}_{t+\delta t} - \dot{x}_t)^2 + (\dot{y}_{t+\delta t} - \dot{y}_t)^2$$
$$\sigma_t := \alpha(\gamma_t + 1), \tag{9}$$

where $\dot{x}_t = x_{t+1} - x_t$ and $\dot{y}_t = y_{t+1} - y_t$ reflect the velocity of the pedestrians within the temporal window length of $\delta t = 8$ and $\alpha$ is the noise intensity. For the training domains $\alpha \in \{1, 2, 4, 8\}$ while for the test domain $\alpha \in \{8, 16, 32, 64\}$. The test domain is the *eth* environment for this experiment. Since the value of $\alpha$ for the third input dimension in training domains were $\{1, 2, 4, 8\}$, the samples of the test domain with $\alpha \in \{16, 32, 64\}$ can be considered as out-of-distribution samples. To evaluate other methods in presence of observation noise, we have used the publicly available code from (Liu et al., 2022). The results in Table 1 demonstrate the robustness of our method against observation noise while performing comparably with other motion forecasting models for low $\alpha$. Since our proposed method also learns to reconstruct inputs, it eliminates the effect of noise by reconstructing uncorrupted inputs, hence, it is not sensitive to noise.

In real-world scenarios, multiple factors affect the data distribution simultaneously. Therefore, it is also useful to disentangle these changing factors and evaluate the robustness of GCRL in different environments with different noise intensities. As shown in Table 2, GCRL is robust in environments where there are multiple changing factors.

### 4.3. Domain Generalization

In this experiment, we evaluate the generalizability of our proposed method using the synthetic dataset (Liu et al., 2022). We will refer to (Liu et al., 2022) as IM in the remainder of the paper. For a fair comparison with IM, we use a PECNet (Mangalam et al., 2020) variant as our base model, train the model with five seeds and report the mean and standard deviation of the results. We will use the PECNet variant as our base model in the subsequent experiments. The Minimum Separation Distances (MSD) in the training and test domains are $\{0.1, 0.3, 0.5\}$ and
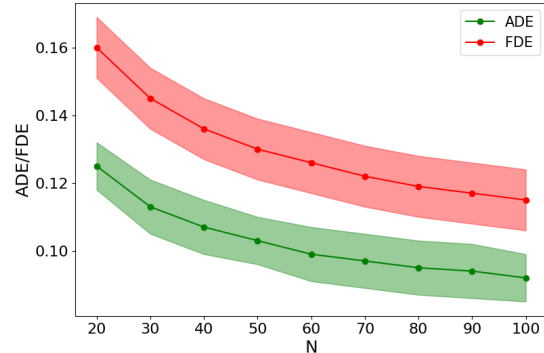


Figure 4: Comparison of ADE/FDE for different values of $N$.

$\{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8\}$ meters, respectively.

Since GCRL is a generative approach, we can generate multiple future trajectories per sample and select the best of them to tackle the multi-modality of trajectories as practiced in (Gupta et al., 2018; Huang et al., 2019; Chen et al., 2021). Therefore, we use a hyper-parameter N in testing to determine the number of generated trajectories per sample. Figure 4 illustrates the significant impact that a generative approach can have in the performance. According to Figure 5, the qualitative results also suggest that a generative approach can be useful in motion forecasting as it is able to generate diverse trajectories. We will use $N = 100$ in the following experiments.

As illustrated in Figure 6, our method is more robust to domain shifts compared to IM, and it is achieving slightly better ADE, which is 8.8% on average. It is evident that for OOD-Inter cases where the test domain shift is within the range of training domain shifts e.g., test domain shift=0.4, GCRL is reusable since ADE is insensitive to the domain shifts. On the other hand, for the test domain shifts out of the range of training domain shifts, the OOD-Extra cases, the model needs to be fine-tuned.

Table 2: GCRL performance in cases with multiple changing factors.

| METHOD | ADE/FDE $\alpha = 8$ | ADE/FDE $\alpha = 16$ | ADE/FDE $\alpha = 32$ | ADE/FDE $\alpha = 64$ |
|---|---|---|---|---|
| ETH | 0.97/1.8 | 0.97/1.8 | 0.97/1.8 | 0.97/1.8 |
| HOTEL | 0.64/1.3 | 0.64/1.3 | 0.64/1.3 | 0.64/1.3 |
| UNIV | 0.53/1.15 | 0.53/1.15 | 0.53/1.15 | 0.53/1.15 |
| ZARA1 | 0.4/0.86 | 0.4/0.86 | 0.4/0.86 | 0.4/0.86 |
| ZARA2 | 0.31/0.66 | 0.31/0.66 | 0.31/0.66 | 0.31/0.66 |



(a) Scene 1　　　　　　(b) Scene 2
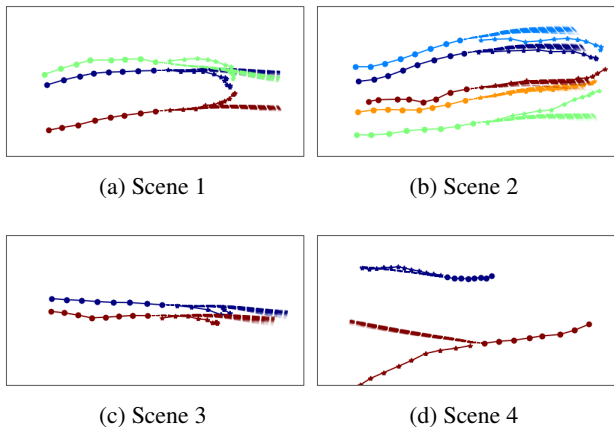
(c) Scene 3　　　　　　(d) Scene 4

Figure 5: Trajectories of pedestrians in different scenes. The circled trajectories are observed, the dashed trajectories are predicted, and the starred trajectories are groundtruth. In all of the scenes, 100 trajectories are predicted for each pedestrian.



Figure 6: Domain generalization with different style domain shifts. The mean and standard deviation of ADEs are calculated and plotted for 5 seeds.

### 4.4. Domain Adaptation

In this experiment, we evaluate the efficiency of our proposed method in knowledge transfer using the synthetic dataset for an OOD-Extra case. We train IM and GCRL with the previous setting and fine-tune different components of the model using a small number of batches from the test domain. The batch size is 64, hence, the number of samples used in fine-tuning will be $\{1, 2, 3, 4, 5, 6\} \times 64$. For IM, we fine-tune it using the best setting reported in the paper. For GCRL, we fine-tune our models for $p(y|x, s, z)$, $p(x|s, z)$, $p(s)$ and $q(s|x)$.

As shown in Figure 7, GCRL adapts to the new environment faster than IM and it is more robust to OOD-Extra shifts. GCRL improves the ADE by 34.3% from IM on average.

### 4.5. Identifiability

To evaluate the identifiability of $S$ and $Z$, we train GCRL with five different seeds on the synthetic dataset. With a GMM prior and a piecewise-affine injective decoder, identifiability up to an affine transformation can be achieved.
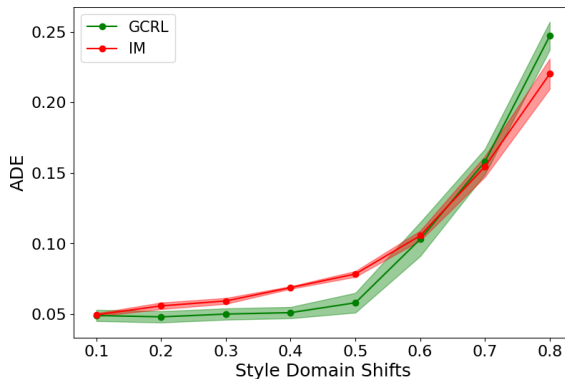
Therefore, we calculate Mean Correlation Coefficient (MCC) between pairs of learnt latents before and after applying an affine transformation $f$, which we refer to as Strong MCC and Weak MCC, respectively. $f$ can be learnt via least-squares minimization.

The results in Table 3 support the identifiability of $S$ as expected. $Z$, however, is not identifiable from observed trajectories and more data from the environments is required. This finding also aligns with the capability of past trajectories in explaining the features that $Z$ is supposed to represent such as physical laws. For example, in order to learn that pedestrians avoid obstacles in their paths, one needs to provide images of the environments. Despite the poor identifiability of $Z$, the empirical results in the next section indicate the invariance of $Z$ across domains.

Table 3: MCC of $S$ and $Z$ after and before applying an affine transformation.

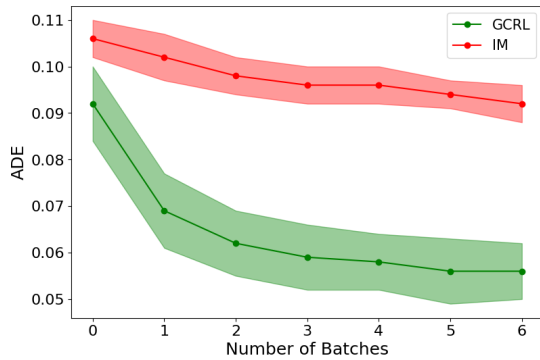| WEAK MCC OF S | WEAK MCC OF Z | STRONG MCC OF S | STRONG MCC OF Z |
|---|---|---|---|
| 0.956 | 0.049 | -0.16 | -0.025 |

Figure 7: Domain Adaptation with different number of batches

## 4.6. Ablation studies

In this experiment, we examine the individual contribution of the components of GCRL. First, we remove the coupling layers from the priors and replace them with learnable parameters. Second, we reconstruct past trajectories and predict future trajectories using only samples from $q(z|x)$. Third, we use only the samples of $q(s|x)$. We train the models on the synthetic dataset with $\{0.1, 0.3, 0.5\}$ shifts and test it on the domain with 0.6 shift.

Table 4: The effect of different components on performance.

| MODELS | ADE | FDE |
|---|---|---|
| VANILA GCRL | 0.0871 | 0.1047 |
| NO COUPLING LAYERS | 0.0772 | 0.0916 |
| Z ONLY | 0.1054 | 0.1347 |
| S ONLY | 0.2188 | 0.2418 |

As shown in Table 4, the model with only $Z$ performs on par with the default model, however, the performance deteriorates when using only $S$. It can be concluded that $Z$ is invariant across domains as expected, however, it lacks information, which leads to worse ADE. The model without coupling layers performs better than the default model for synthetic dataset, which indicates that fewer parameters would suffice for certain environments.

## 4.7. Discussion

Intuitively, fine-tuning a GMM is asking whether we can use a combination of learned representations in the test domain, or whether we need a new kind of representation (cluster). Therefore, instead of fine-tuning all parameters of $p(s)$ we can fine-tune the weights of the GMM only in cases such as OOD-Inter. According to Figure 6, the performance of GCRL degrades for samples with $MSD \geq 0.5$ and

performs worse than baseline for $MSD \geq 0.7$. The reason is we have trained the model on samples with $MSD = \{0.1, 0.3, 0.5\}$, hence, the intrapolated samples i.e., samples with $MSD \leq 0.5$, have similar variant representations S. In other words, the trained model works well for the interpolated samples, but for the extrapolated samples, the GMM prior fails to generalize and domain adaptation is required when transferring to a new domain.

In addition to the results presented in the paper, we conducted extensive ablation experiments including fine-tuning only the GMM weights and not fine-tuning other components of $p(s)$ which resulted in 34.1% improvement whereas the reported result in the paper is 34.3%. Consequently, we found the best performance was achieved when we fine-tuned all $p(s)$, $q(s|x)$, $p(x|s, z)$, and $p(y|x, s, z)$.

As for the identifiabiltity of Z, we applied extensive regularizations including contrastive modeling and constraints on the components of GMM, however, in our experiments we did not observe any improvement from these methods.

As a proof of concept we carried out experiments of our approach using two datasets, real and synthetic. The ETH-UCY dataset is the only dataset used by the baseline model (Huang et al., 2019). Since our contribution is focused on robustness and domain adaptation, we wanted to show that by utilizing a motion forecasting model that can better capture interactions between agents in complex datasets, GCRL could achieve the same robustness and efficiency in transfer learning. We also believe that given the right baseline model, our method can perform well in out-of-distribution prediction in other applications but, this is yet to be evaluated. GCRL requires a model with EID architecture and constraints on the decoder of the model to ensure identifiability of S which are some parts of its limitations. Our results suggest that considering a temporal graph instead of a summary graph and capturing human-human and human-space interactions can be a promising future work of GCRL.

## 5. Conclusions

We propose a method that leverages causality to learn meaningful features that can increase the robustness and transferability of deep learning models. In presence of spurious correlation, we demonstrated the robustness of our method while other human trajectory prediction models performed poorly compared to our method. Furthermore, our augmented causal model was able to enhance the transferability in a zero-shot and low-shot settings. It can be concluded from our results that incorporating causality in deep learning is a promising research direction towards robustness and explainability of deep learning models.

# References

Arjovsky, M., Bottou, L., Gulrajani, I., and Lopez-Paz, D. Invariant risk minimization, 2019. URL https://arxiv.org/abs/1907.02893.

Chen, C., Liu, Y., Kreiss, S., and Alahi, A. Crowd-robot interaction: Crowd-aware robot navigation with attention-based deep reinforcement learning. In *International Conference on Robotics and Automation, ICRA 2019, Montreal, QC, Canada, May 20-24, 2019*, pp. 6015–6022. IEEE, 2019. doi: 10.1109/ICRA.2019.8794134.

Chen, G., Li, J., Lu, J., and Zhou, J. Human trajectory prediction via counterfactual analysis. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9804–9813, 2021. doi: 10.1109/ICCV48922.2021.00968.

Dinh, L., Sohl-Dickstein, J., and Bengio, S. Density estimation using real NVP. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

Goodfellow, I., Bengio, Y., and Courville, A. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

Goudet, O., Kalainathan, D., Caillou, P., Guyon, I., Lopez-Paz, D., and Sebag, M. Causal generative neural networks, 2017. URL https://arxiv.org/abs/1711.08936.

Gupta, A., Johnson, J., Fei-Fei, L., Savarese, S., and Alahi, A. Social gan: Socially acceptable trajectories with generative adversarial networks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2255–2264, 2018. doi: 10.1109/CVPR.2018.00240.

Huang, Y., Bi, H., Li, Z., Mao, T., and Wang, Z. Stgat: Modeling spatial-temporal interactions for human trajectory prediction. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6271–6280, 2019. doi: 10.1109/ICCV.2019.00637.

Jiang, Z., Zheng, Y., Tan, H., Tang, B., and Zhou, H. Variational deep embedding: An unsupervised and generative approach to clustering. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pp. 1965–1972. ijcai.org, 2017. doi: 10.24963/ijcai.2017/273.

Johnson, M. J., Duvenaud, D., Wiltschko, A. B., Adams, R. P., and Datta, S. R. Composing graphical models with neural networks for structured representations and fast inference. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pp. 2946–2954, 2016.

Ke, N. R., Bilaniuk, O., Goyal, A., Bauer, S., Larochelle, H., Schölkopf, B., Mozer, M. C., Pal, C., and Bengio, Y. Learning neural causal models from unknown interventions, 2019. URL https://arxiv.org/abs/1910.01075.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

Kivva, B., Rajendran, G., Ravikumar, P., and Aragam, B. Identifiability of deep generative models without auxiliary information, 2022. URL https://arxiv.org/abs/2206.10044.

Kosaraju, V., Sadeghian, A., Martín-Martín, R., Reid, I., Rezatofighi, H., and Savarese, S. Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/d09bf41544a3365a46c9077ebb5e35c3-Paper.pdf.

Kothari, P., Kreiss, S., and Alahi, A. Human trajectory forecasting in crowds: A deep learning perspective. *IEEE Transactions on Intelligent Transportation Systems*, 23 (7):7386–7400, 2022. doi: 10.1109/TITS.2021.3069362.

Lee, N., Choi, W., Vernaza, P., Choy, C. B., Torr, P. H. S., and Chandraker, M. Desire: Distant future prediction in dynamic scenes with interacting agents. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2165–2174, 2017. doi: 10.1109/CVPR.2017.233.

Lerner, A., Chrysanthou, Y., and Lischinski, D. Crowds by example. *Computer Graphics Forum*, 26(3):655–664, 2007. doi: https://doi.org/10.1111/j.1467-8659.2007.01089.x. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-8659.2007.01089.x.

Liu, C., Sun, X., Wang, J., Tang, H., Li, T., Qin, T., Chen, W., and Liu, T.-Y. Learning causal semantic representation for out-of-distribution prediction. In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing*

*Systems*, volume 34, pp. 6155–6170. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/file/310614fca8fb8e5491295336298c340f-Paper.pdf.

Liu, Y., Cadei, R., Schweizer, J., Bahmani, S., and Alahi, A. Towards robust and adaptive motion forecasting: A causal representation perspective. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pp. 17060–17071. IEEE, 2022. doi: 10.1109/CVPR52688.2022.01657.

Locatello, F., Bauer, S., Lucic, M., Raetsch, G., Gelly, S., Schölkopf, B., and Bachem, O. Challenging common assumptions in the unsupervised learning of disentangled representations. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 4114–4124. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/locatello19a.html.

Magliacane, S., van Ommen, T., Claassen, T., Bongers, S., Versteeg, P., and Mooij, J. M. Domain adaptation by using causal inference to predict invariant conditional distributions. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper/2018/file/39e98420b5e98bfbdc8a619bef7b8f61-Paper.pdf.

Mangalam, K., Girase, H., Agarwal, S., Lee, K.-H., Adeli, E., Malik, J., and Gaidon, A. It is not the journey but the destination: Endpoint conditioned trajectory prediction. In *Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II*, pp. 759–776, Berlin, Heidelberg, 2020. Springer-Verlag. ISBN 978-3-030-58535-8. doi: 10.1007/978-3-030-58536-5_45. URL https://doi.org/10.1007/978-3-030-58536-5_45.

Mohamed, A., Qian, K., Elhoseiny, M., and Claudel, C. Social-stgcnn: A social spatio-temporal graph convolutional neural network for human trajectory prediction. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14412–14420, 2020. doi: 10.1109/CVPR42600.2020.01443.

Pearl, J. and Bareinboim, E. External Validity: From Do-Calculus to Transportability Across Populations. *Statistical Science*, 29(4):579 – 595, 2014. doi: 10.1214/14-STS486. URL https://doi.org/10.1214/14-STS486.

Pearl, J., Glymour, M., and P. Jewell, N. *Causal inference in statistics: A primer*. John Wiley and Sons, 2016.

Pellegrini, S., Ess, A., and Van Gool, L. Improving data association by joint modeling of pedestrian trajectories and groupings. In Daniilidis, K., Maragos, P., and Paragios, N. (eds.), *Computer Vision – ECCV 2010*, pp. 452–465, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg. ISBN 978-3-642-15549-9.

Salzmann, T., Ivanovic, B., Chakravarty, P., and Pavone, M. Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data. In *Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII*, pp. 683–700, Berlin, Heidelberg, 2020. Springer-Verlag. ISBN 978-3-030-58522-8. doi: 10.1007/978-3-030-58523-5_40. URL https://doi.org/10.1007/978-3-030-58523-5_40.

Schölkopf, B., Locatello, F., Bauer, S., Ke, N. R., Kalchbrenner, N., Goyal, A., and Bengio, Y. Toward causal representation learning. *Proceedings of the IEEE*, 109(5): 612–634, 2021. doi: 10.1109/JPROC.2021.3058954.

Suter, R., Miladinovic, D., Schölkopf, B., and Bauer, S. Robustly disentangled causal mechanisms: Validating deep representations for interventional robustness. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 6056–6065. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/suter19a.html.

Teshima, T., Sato, I., and Sugiyama, M. Few-shot domain adaptation by causal mechanism transfer. In III, H. D. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9458–9469. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/teshima20a.html.

Xu, H., Chen, W., Lai, J., Li, Z., Zhao, Y., and Pei, D. On the necessity and effectiveness of learning the prior of variational auto-encoder, 2019. URL https://arxiv.org/abs/1905.13452.

Yang, S., Yu, K., Cao, F., Liu, L., Wang, H., and Li, J. Learning causal representations for robust domain adaptation. *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–1, 2021. doi: 10.1109/TKDE.2021.3119185.

Yu, K., Guo, X., Liu, L., Li, J., Wang, H., Ling, Z., and Wu, X. Causality-based feature selection: Methods and evaluations. *ACM Comput. Surv.*, 53(5), sep 2020. ISSN 0360-0300. doi: 10.1145/3409382. URL https://doi.org/10.1145/3409382.

Yu, Y., Chen, J., Gao, T., and Yu, M. DAG-GNN: DAG structure learning with graph neural networks. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 7154–7163. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/yu19a.html.

Zhang, D., Zhang, H., Tang, J., Hua, X.-S., and Sun, Q. Causal intervention for weakly-supervised semantic segmentation. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 655–666. Curran Associates, Inc., 2020a. URL https://proceedings.neurips.cc/paper/2020/file/07211688a0869d995947a8fb11b215d6-Paper.pdf.

Zhang, K., Gong, M., Stojanov, P., Huang, B., Liu, Q., and Glymour, C. Domain adaptation as a problem of inference on graphical models. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA, 2020b. Curran Associates Inc. ISBN 9781713829546.

# A. Methodology details

**Lower bound on the log likelihood of past trajectories**   The objective function in GCRL is given in Equation 3. Since the future trajectories are unknown at test time, the approximate posterior $q(s, z, e|x, y)$ is intractable. Therefore, we rewrite the Equation 3 such that the approximate posterior is conditioned on only x:

$$\max_{p,q} E_{p^*(x,y)} \left[ E_{q(s,z,e|x,y)} \left[ \log \frac{p(x,y,s,z,e)}{q(s,z,e|x,y)} \right] \right] =$$

$$\max_{p,q} E_{p^*(x,y)} \left[ E_{q(s,z,e|x,y)} \left[ \log \frac{p(x,y,s,z,e)q(y|x)}{q(s,z,e,y|x)} \right] \right] =$$

$$\max_{p,q} E_{p^*(x,y)} \left[ \int \left[ \log \frac{p(x,y,s,z,e)q(y|x)}{q(s,z,e,y|x)} \right] q(s,z,e|x,y)dsdzde \right] =$$

$$\max_{p,q} E_{p^*(x,y)} \left[ \int \left[ \log \frac{p(x,y,s,z,e)}{q(s,z,e,y|x)} \right] q(s,z,e|x,y)dsdzde + \int \left[ \log q(y|x) \right] q(s,z,e|x,y)dsdzde \right] =$$

$$\max_{p,q} E_{p^*(x,y)} \left[ \int \left[ \log \frac{p(x,y,s,z,e)}{q(s,z,e,y|x)} \right] q(s,z,e|x,y)dsdzde + \log q(y|x) \right] =$$

$$\max_{p,q} E_{p^*(x,y)} \left[ \int \left[ \log \frac{p(x,y,s,z,e)}{q(s,z,e,y|x)} \right] \frac{q(s,z,e,y|x)}{q(y|x)}dsdzde + \log q(y|x) \right] =$$

$$\max_{p,q} E_{p^*(x,y)} \left[ \int \left[ \frac{1}{q(y|x)} \log \frac{p(x,y,s,z,e)}{q(s,z,e,y|x)} \right] q(s,z,e,y|x)dsdzde + \log q(y|x) \right] =$$

$$\max_{p,q} \int \left[ \frac{1}{q(y|x)} \log \frac{p(x,y,s,z,e)}{q(s,z,e,y|x)} \right] p^*(x,y)q(s,z,e,y|x)dsdzdedxdy + \int \left[ \log q(y|x) \right] p^*(x,y)dxdy =$$

$$\max_{p,q} \int \left[ \frac{1}{q(y|x)} \log \frac{p(x,y,s,z,e)}{q(s,z,e,y|x)} \right] p^*(y|x)p^*(x)q(s,z,e,y|x)dsdzdedxdy + \int \left[ \log q(y|x) \right] p^*(y|x)p^*(x)dxdy =$$

$$\max_{p,q} \int \left[ \frac{p^*(y|x)}{q(y|x)} \log \frac{p(x,y,s,z,e)}{q(s,z,e,y|x)} \right] q(s,z,e,y|x)p^*(x)dsdzdedxdy + \int \left[ \log q(y|x) \right] p^*(y|x)p^*(x)dxdy =$$

$$\max_{p,q} \int E_{q(s,z,e,y|x)} \left[ \frac{p^*(y|x)}{q(y|x)} \log \frac{p(x,y,s,z,e)}{q(s,z,e,y|x)} \right] p^*(x)dx + \int E_{p^*(y|x)} \left[ \log q(y|x) \right] p^*(x)dx =$$

$$\max_{p,q} E_{p^*(x)}E_{p^*(y|x)}[\log q(y|x)] + E_{p^*(x)}[E_{q(s,z,e,y|x)}[\frac{p^*(y|x)}{q(y|x)} \log \frac{p(x,y,s,z,e)}{q(s,z,y,e|x)}]]$$

$$(10)$$

which gives us the Equation 4. According to Equation 13, $q(y|x)$ will be derived towards $p^*(y|x)$ as desired and the objective function will become:

$$\max_{p,q} E_{p^*(x)}[E_{q(s,z,e,y|x)}[\frac{p^*(y|x)}{p^*(y|x)}\log\frac{p(x,y,s,z,e)}{q(s,z,y,e|x)}]] =$$

$$\max_{p,q} E_{p^*(x)}[E_{q(s,z,e,y|x)}[\log\frac{p(x,y,s,z,e)}{q(s,z,y,e|x)}]] =$$

$$\max_{p,q} E_{p^*(x)}[\int[\log\frac{p(x,y,s,z,e)}{q(s,z,y,e|x)}]q(s,z,e,y|x)dsdzdedy] =$$

$$\max_{p,q} E_{p^*(x)}[\int[\log\frac{p(y|s,z,x)p(x|s,z)p(z)p(s|e)p(e)}{q(y|s,z,x)q(z|x)q(s|x)q(e|x)}]q(y|s,z,x)q(z|x)q(s|x)q(e|x)dsdzdedy] =$$

$$\max_{p,q} E_{p^*(x)}[\int[\log\frac{p(x|s,z)p(z)p(s|e)p(e)}{q(z|x)q(s|x)q(e|x)}]p(y|s,z,x)q(z|x)q(s|x)q(e|x)dsdzdedy] =$$  (11)

$$\max_{p,q} E_{p^*(x)}[\int[\log\frac{p(x|s,z)p(z)p(s|e)p(e)}{q(z|x)q(s|x)q(e|x)}]q(z|x)q(s|x)q(e|x)dsdzde] =$$

$$\max_{p,q} E_{p^*(x)}[\int[\log\frac{p(x|s,z)p(z)p(s)p(e|s)}{q(z|x)q(s|x)q(e|x)}]q(z|x)q(s|x)q(e|x)dsdzde] =$$

$$\max_{p,q} E_{p^*(x)}[E_{q(s,z|x)}[\log p(x|s,z)] + E_{q(z|x)}[\log\frac{p(z)}{q(z|x)}] + E_{q(s|x)}[\log\frac{p(s)}{q(s|x)}] + E_{q(e|x)}[\log\frac{p(e|s)}{q(e|x)}]] \leq$$

$$\max_{p,q} E_{p^*(x)}[E_{q(s,z|x)}[\log p(x|s,z)]] \leq$$

$$\max_{p,q} E_{p^*(x)}[\log E_{q(s,z|x)}[p(x|s,z)]] = E_{p^*(x)}[\log p(x)]$$

which is evidently the lower bound of $\log p(x)$. $KL$ denotes the KL-divergence. Since the natural logarithm is a concave function, the second inequality follows from Jensen's inequality. The third equality in Equation 11 follows from the Causal Markov condition that $q(s,z,e,y|x) = q(y|x,s,z)q(z|x)q(s|x,e)q(e|x)$ and $p(x,s,z,y,e) = p(y|x,s,z)p(x|s,z)p(s|e)p(z)p(e)$. Furthermore, we are using mean-field variational inference in which latent variables are assumed to be independent i.e., $q(s|e,x) = q(s|x)$. The last equality holds because eventually $q(s|x) = p(s)$ and $q(z|x) = p(z)$.

**GCRL objective function**    According to Equation 11, eventually, $q(e|x)$ will be derived towards $p(e|s)$. Therefore, it is sensible to assume that $q(e|x) = p(e|s)$. This is useful because it helps us to obtain a GMM prior and avoid modelling of the approximate posterior of E. This can be shown via Bayes rule and replacing :

$$q(e|x) = p(e|s) = \frac{p(s|e)p(e)}{p(s)} = \frac{p(s|e)p(e)}{\sum_e p(s|e)p(e)} \tag{12}$$

Since we assume $p(e)$ and $p(s|e)$ are known, we can avoid modelling $q(e|x)$. By replacing $q(e|x)$ in Equation 4 and using the Causal Markov condition, we can obtain a GMM prior for S:

$$\max_{p,q} E_{p^*(x)}E_{p^*(y|x)}[\log q(y|x)] + E_{p^*(x)}[E_{q(s,z,e,y|x)}[\frac{p^*(y|x)}{q(y|x)}\log\frac{p(x,y,s,z,e)}{q(s,z,y,e|x)}]] =$$

$$\max_{p,q} E_{p^*(x,y)}[\log q(y|x) + E_{q(s,z,e|x)}[\frac{p(y|s,z,x)}{q(y|x)}\log\frac{p(y|s,z,x)p(x|s,z)p(z)p(s|e)p(e)}{q(y|s,z,x)q(z|x)q(s|x)q(e|x)}]] =$$

$$\max_{p,q} E_{p^*(x,y)}[\log q(y|x) + \frac{1}{q(y|x)}E_{q(s,z,e|x)}[p(y|s,z,x)\log\frac{p(x|s,z)p(z)p(s|e)p(e)}{q(z|x)q(s|x)q(e|x)}]] =$$  (13)

$$\max_{p,q} E_{p^*(x,y)}[\log q(y|x) + \frac{1}{q(y|x)}E_{q(s,z,e|x)}[p(y|s,z,x)\log\frac{p(x|s,z)p(z)p(s|e)p(e)p(s)}{q(z|x)q(s|x)p(s|e)p(e)}]] =$$

$$\max_{p,q} E_{p^*(x,y)}[\log q(y|x) + \frac{1}{q(y|x)}E_{q(s,z,e|x)}[p(y|s,z,x)\log\frac{p(x|s,z)p(z)p(s)}{q(z|x)q(s|x)}]]$$

where $p(s) = \sum_e p(s|e)p(e)$ that is a GMM prior for S. $q(y|x)$ can be written as:

$$q(y|x) = \int q(y|x,s,z)q(s,z|x)dsdz = \int p(y|x,s,z)q(s|x)q(z|x)dsdz = p(y|do(x)) = E_{q(s|x),q(z|x)}[p(y|x,s,z)] \tag{14}$$

where $p(y|do(x))$ is the causal effect of X on Y and can be calculated by ancestral sampling. We have obtained Equation 6 which is the objective function of GCRL.

## B. Experiment

In this section, we present the experimental details and more experimental results achieved when evaluating our framework. Following our proposed method in Section 3 of the paper, our final loss function is formulated by Equation 6.

### B.1. Experiment details

The list of all hyperparameters used by our model and their corresponding settings applied when conducting our experiments are represented in Tables 5 and 6.

Following Tables 5 and 6 and Equation 6, $S_{Dim}$ and $Z_{Dim}$ show the dimensions of the latent representation space for $S$ and $Z$ variables, respectively. $N^s_{q(y|x)}$ and $N^s_{E_{q(s|x),q(z|x)}}$ show the number of sampling from $S$ and $Z$ distributions for calculating the expectations of $q(y|x)$ and $E_{q(s|x),q(z|x)}$, respectively.

$R_{Input}$ shows the reconstructed input that is the *relative distance* of the pedestrians from their starting locations, $d_{rel}$, which are used in our experiments conducted by the ETH-UCY dataset for a fair comparison with the state-of-the-art methods. Our hyperparameter settings ruuning experiment with ETH-UCY dataset are represented in Table 5. Our model is trained for 300 epochs in experiments conducted with the ETH-UCY dataset.

Table 5: Detailed hyperparameter settings of the experiments with ETH-UCY dataset.

| Parameter | Setting | Parameter | Setting |
|---|---|---|---|
| $S_{Dim}$ | 8 | Learning Rate | Scheduler (one-cycle) |
| $Z_{Dim}$ | 8 | Optimizer | Adam |
| $N^s_{E_{q(y|x)}}$ | 10 | $n_{cluster}$ (GMM) | 5 |
| $N^s_{E_{q(s|x),q(z|x)}}$ | 10 | $R_{Input}$ | $d_{rel}$ |

**Note** if $N^s_{q(y|x)} = 1$, then in the second term of the loss function $p(y|x,s,z)$ cancels out $q(y|x)$, therefore, all components of the second term of the loss function will not include the predicted future trajectories $y$. Considering this, for the experiments where $N^s_{q(y|x)} = 1$, we use a hyperparameter $N$, which is defined as the number of generated trajectories per sample, to improve the performance. In this case, the term $log\, q(y|x)$ in Equation 6 becomes the variety loss (Gupta et al., 2018). We do this because our approach to causal representation learning is *generative*, and therefore, we can benefit from its generative aspect to improve the performance of our model. $N$ is set to 20 during training and 100 during evaluation in experiments with synthetic dataset. Hyperparameter settings for experiments with synthetic dataset are represented in Table 6. The *absolute locations* of the pedestrians $l_{abs}$ are used in our experiments with the synthetic dataset for a fair comparison with IM.

We trained our method GCRL in the experiments with synthetic dataset for 250 epochs. We fine-tuned the trained models for the domain adaptation task for 100 epochs. IM has a training paradigm conducted in 4 phases with the total number of 470 epochs followed by 300 epochs for finetuning. On the other hand, our proposed method offers an end-to-end learning process which converges with much less number of epochs both for training and fine-tuning tasks.

Table 6: Detailed hyperparameter settings of the experiments with Synthetic dataset.

| Parameter | Setting | Parameter | Setting |
|-----------|---------|-----------|---------|
| $S_{Dim}$ | 2 | Learning Rate | $5 \times 10^{-3}$ |
| $Z_{Dim}$ | 2 | Optimizer | Adam |
| $N^s_{E_{q(y|x)}}$ | 1 | $n_{cluster}$ (GMM) | 5 |
| $N^s_{E_{q(s|x),q(z|x)}}$ | 10 | $R_{Input}$ | $l_{abs}$ |

## B.2. Additional Experiments

**Motion forecasting with ETH-UCY**   We present motion forecasting results of all environments of the ETH-UCY dataset; 'ETH', 'HOTEL', 'UNIV', 'ZARA1' and 'ZARA2'. These experiments are conducted without the controlled spurious correlations. Table 7 presents a comparison between performances of GCRL and the base model STGAT (Huang et al., 2019) with similar settings.

Table 7: Motion forecasting results on different domains of ETH-UCY.

| Model | ADE/FDE | | | | | |
|-------|---------|---------|---------|---------|---------|---------|
|       | ETH | HOTEL | UNIV | ZARA1 | ZARA2 | **AVG** |
| STGAT | 0.8/1.53 | 0.52/1.08 | 0.51/1.12 | 0.39/0.87 | 0.3/0.64 | 0.5/1.05 |
| GCRL | 0.97/1.9 | 0.55/1.14 | 0.51/1.13 | 0.38/0.84 | 0.3/0.67 | 0.54/1.1 |

Figures 8a and 8b show the ADE while training and validating our model on the 'hotel', 'univ', 'zara1', and 'zara2' environments of the ETH-UCY dataset for 300 epochs, respectively. The trained model is tested on the 'eth' environment at inference. Figures 9a and 9b, on the other hand, show the FDE for training and validation, respectively. The convergence of prediction and reconstruction losses during training are shown in Figures 10a and 10b, respectively. Large values of reconstruction and prediction losses are due to the constant terms added to $p(x|, s, z)$ and $p(y|x, s, z)$ such as the log of determinant of covariance matrices. The regularization loss of $S$ and $Z$ distributions are also represented by Figures 11a and 11b, respectively. As shown in the Figures, the regularization losses converge to zero showing that the model drives the posterior distributions of $S$ and $Z$ towards their priors.

**Ablation study on ETH-UCY**   In Table 8, the results for Ablation studies are presented. We used different settings to evaluate our method in these experiments including: removing coupling layers, changing the dimensions of $S$ and $Z$ latent representations, and applying different number of clusters for GMM used to model prior of $S$.

Table 8: The effect of different components on performance when experimenting with the ETH-UCY dataset.

| MODELS | ADE | FDE |
|--------|-----|-----|
| VANILA GCRL | 0.9728 | 1.8696 |
| NO COUPLING LAYERS | 0.9881 | 1.8869 |
| LATENT DIMENSION:2 | 0.9736 | 1.8459 |
| NUMBER OF GMM CLUSTERS:2 | 0.9732 | 1.9534 |
| NUMBER OF GMM CLUSTERS:7 | 0.9775 | 1.9169 |

(a) Training

(b) Validation

Figure 8: ADE



(a) Training

(b) Validation

Figure 9: FDE



(a) Prediction Loss
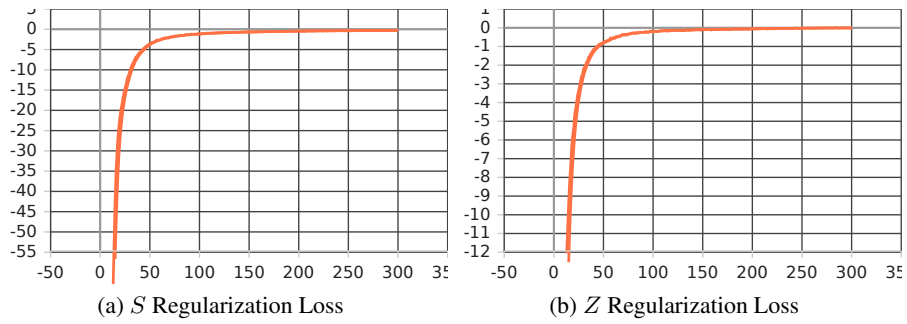
(b) Reconstruction Loss

Figure 10: Predictions
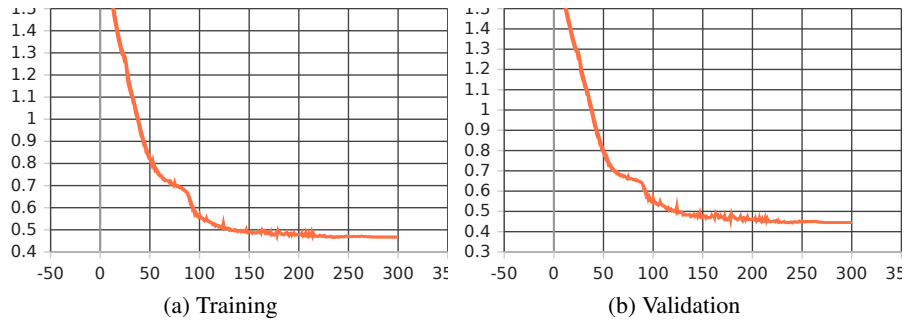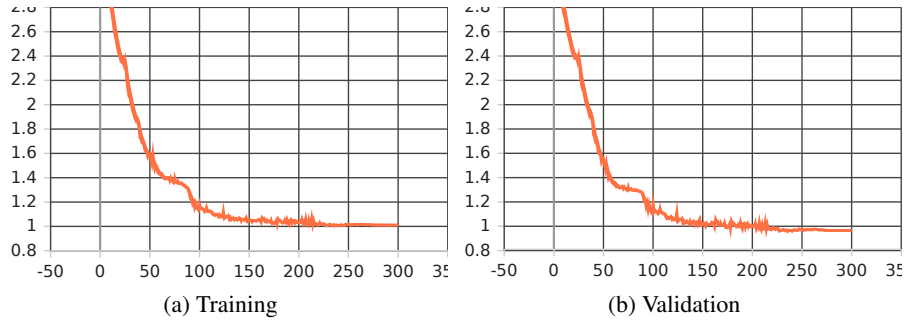


(a) $S$ Regularization Loss
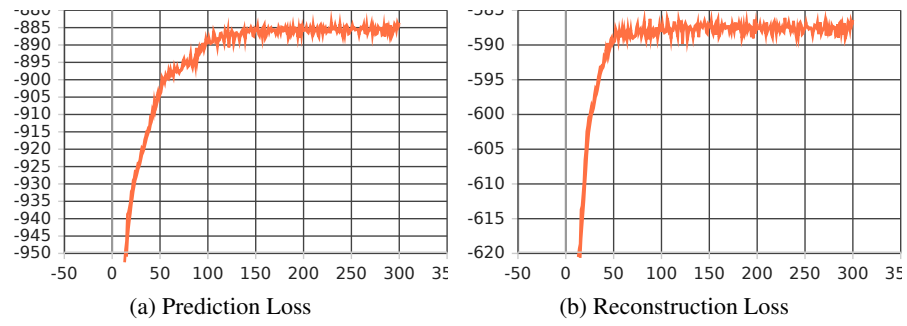
(b) $Z$ Regularization Loss

Figure 11: Regularizations