Adaptive Multi-stage Density Ratio Estimation for Learning Latent Space Energy-based Model

Zhisheng Xiao [1] **Tian Han** [2]

Abstract

This paper studies the fundamental problem of learning energy-based model (EBM) in the latent space of the generator model. Learning such prior model typically requires running costly Markov Chain Monte Carlo (MCMC). Instead, we propose to use noise contrastive estimation (NCE) to discriminatively learn the EBM through density ratio estimation between the latent prior density and latent posterior density. However, the NCE typically fails to accurately estimate such density ratio given large gap between two densities. To effectively tackle this issue and further learn more expressive prior model, we develop the adaptive multi-stage density ratio estimation which breaks the estimation into multiple stages and learn different stages of density ratio sequentially and adaptively. The latent prior model can be gradually learned using ratio estimated in previous stage so that the final latent space EBM prior can be naturally formed by product of ratios in different stages. The proposed method enables informative and much sharper prior than existing baselines, and can be trained efficiently. Our experiments demonstrate strong performances in terms of image generation and reconstruction as well as anomaly detection.

1. Introduction

Deep generative model provides a powerful framework for representing complex data distributions and have seen many successful applications in image and video synthesis (Karras et al., 2019; Saito et al., 2020), representation learning (Oord et al., 2017) as well as unsupervised or semi-supervised learning (Izmailov et al., 2020; Pang et al., 2020b). Such model, referred to as *generator model*, usually consists of low-dimensional latent variables that follow non-informative prior distribution, and a top-down network that maps such latent vector to the observed example. The informative prior model in the latent space (Pang et al., 2020a; Aneja et al., 2021) can be learned to further improve the expressive power of the whole model. Specifically, we consider learning energy-based model (EBM) in the latent space as our informative prior for the generator model.

Learning latent space EBM can be challenging and requires iterative Markov Chain Monte Carlo (MCMC) sampling step which is computationally expensive and sensitive to hyperparameters. In this paper, we instead propose to use noise contrastive estimation (NCE) (Gutmann & Hyvärinen, 2012) for learning EBM prior via density ratio estimation. The EBM is learned discriminatively by classifying the latent vector sampled from the prior density and the latent sampled from posterior density. Instead of variational learned inference (Kingma & Welling, 2014; Rezende et al., 2014) which needs a separate inference network designed, we obtain the posterior latent sample through short-run Langevin dynamics (Nijkamp et al., 2020) to ensure more accurate inference. However, the success of NCE depends on the closeness of prior density and posterior density (Hoffman & Johnson, 2016). Given large gap between two densities, NCE typically fails to accurately estimate such density ratio which leads to inaccurate EBM modeling.

To effectively tackle the inaccurate estimation issue and further learn more expressive prior model, we develop the adaptive multi-stage density ratio estimation for latent space EBM training. The proposed model breaks the density estimation into multiple stages and learn different stages of density ratio sequentially. Thanks to the low-dimensionality of the latent space and the short-run style posterior inference, in each stage, the gap between prior and posterior density could be kept in check which makes NCE easier. The density ratio estimated in previous stage can be further integrated into the current prior model as a correction term to build more expressive prior density for later stage. With such framework, the final latent space EBM prior can then be naturally formed by product of ratios in different stages on top of the initial base prior.

Contributions: 1) we propose an EBM prior on genera-

*Equal contribution [1]Computational and Applied Mathematics, University of Chicago, Chicago, IL [2]Stevens Institute of Technology, Hoboken, NJ. Correspondence to: Zhisheng Xiao <zxiao@uchicago.edu>.

Work in preogress.

tor model which is modelled through estimation of density ratios in multiple stages. 2) We develop the adaptive multi-stage noise contrastive estimation to learn different stages of ratios sequentially and adaptively. The ratio estimated in previous stage can be integrated to form the more informative prior in the later stage. 3) we demonstrate strong empirical results to illustrate the proposed method.

2. Background

2.1. Maximum likelihood learning of deep latent variable models

Let $\mathbf{x} \in \mathbb{R}^D$ be an observed example such as an image, and $\mathbf{z} \in \mathbb{R}^d$ be the latent variables where $d < D$. A latent variable generative model (a.k.a, *generator model*) factorize the joint distribution of $(\mathbf{x}, \mathbf{z})$ as

$$p_\theta(\mathbf{x}, \mathbf{z}) = p(\mathbf{z}) p_\theta(\mathbf{x}|\mathbf{z}), \tag{1}$$

where $p(\mathbf{z})$ is the prior distribution over latent variables z, $p_\theta(\mathbf{x}|\mathbf{z})$ is the top-down generation model with parameters θ. Usually the prior distribution is chosen to be a simple one such as $\mathcal{N}(0, \mathbf{I}_d)$, but it can also be more expressive with learnable parameters (Pang et al., 2020a). The generation model is the same as that in VAE (Kingma & Welling, 2014), i.e., $\mathbf{x} = g_\theta(\mathbf{z}) + \epsilon$ with g_θ to be the decoder network and $\epsilon \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_D)$, so that $p_\theta(\mathbf{x}|\mathbf{z}) = \mathcal{N}(g_\theta(\mathbf{z}), \sigma^2 \mathbf{I}_D)$. As in VAE, σ^2 takes a pre-specified value.

Given a set of N training samples $\{\mathbf{x}_i, i = 1, \ldots, N\}$ from the unknown data distribution $p_{\text{data}}(\mathbf{x})$, the model p_θ can be trained by maximizing the log likelihood over training samples $\mathcal{L}(\theta) = \frac{1}{N} \sum_{i=1}^N \log p_\theta(\mathbf{x}_i)$. Maximizing the log likelihood $\mathcal{L}(\theta)$ can be accomplished by gradient ascent where the gradient can be obtained from

$$\begin{aligned}\nabla_\theta \log p_\theta(\mathbf{x}) &= \frac{1}{p_\theta(\mathbf{x})} \nabla_\theta p_\theta(\mathbf{x}) \\ &= \int [\nabla_\theta \log p_\theta(\mathbf{x}, \mathbf{z})] \frac{p_\theta(\mathbf{x}, \mathbf{z})}{p_\theta(\mathbf{x})} d\mathbf{z} \\ &= \mathbb{E}_{p_\theta(\mathbf{z}|\mathbf{x})} [\nabla_\theta \log p_\theta(\mathbf{x}, \mathbf{z})]. \end{aligned} \tag{2}$$

$\nabla_\theta \log p_\theta(\mathbf{x}, \mathbf{z})$ can be easily computed according to the form of $\log p_\theta(\mathbf{x}, \mathbf{z})$, however, approximating the expectation requires drawing samples from $p_\theta(\mathbf{z}|\mathbf{x})$, which can be difficult. Sampling from the intractable posterior $p_\theta(\mathbf{z}|\mathbf{x})$ requires MCMC, and one convenient MCMC algorithm is Langevin Dynamics (LD) (Neal et al., 2011). Given a step size $s > 0$, and an initial value $\mathbf{z}_0$, the Lanegvin dynamics iterates

$$\mathbf{z}^{k+1} = \mathbf{z}^k + \frac{s}{2} \nabla_{\mathbf{z}} \log p_\theta(\mathbf{z}|\mathbf{x}) + \sqrt{s} \omega_k, \tag{3}$$

where $\omega_k \sim \mathcal{N}(0, \mathbf{I})$. For sufficiently small step size s, the marginal distribution of $\mathbf{z}_k$ will converge to $p_\theta(\mathbf{z}|\mathbf{x})$ as $k \to \infty$. However, it is not feasible to run Langevin dynamics until convergence, and in practice the iteration in Eq. 3 is run for finite iterations, which yields a Markov chain with an invariant distribution approximately close to the original target distribution. When $\mathbf{z}_0$ is initialized from the noise distribution, the algorithm is called noise-initialized short-run LD (Nijkamp et al., 2019; 2020).

2.2. Learning EBMs with discriminative density ratio estimation

Suppose there are two distributions with density functions $p(\mathbf{x})$ and $q(\mathbf{x})$ from which we can sample, we can estimate the density ratio[1] $r(\mathbf{x}) = \frac{p(\mathbf{x})}{q(\mathbf{x})}$ by training a classifier to distinguish samples from p and q (Sugiyama et al., 2012). Specifically, we can train the binary classifier $D : \mathbb{R}^n \to (0, 1)$ by minimizing the binary cross-entropy loss

$$\min_D -\mathbb{E}_{\mathbf{x} \sim q(\mathbf{x})} [\log D(\mathbf{x})] - \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} [\log(1 - D(\mathbf{x}))].$$

The objective is minimized when $D(\mathbf{x}) = \frac{q(\mathbf{x})}{q(\mathbf{x}) + p(\mathbf{x})}$ (Goodfellow et al., 2014), and denoting the classifier at optimality by $D^*(\mathbf{x})$, we have $r(\mathbf{x}) = \frac{q(\mathbf{x})}{p(\mathbf{x})} \approx \frac{D^*(\mathbf{x})}{1 - D^*(\mathbf{x})}$.

Such a technique can be useful for training Energy-based models (EBMs). Given samples from the true data distribution $p_{\text{data}}(\mathbf{x})$ and a base distribution $q(\mathbf{x})$ that we can sample from, we consider EBMs of the form $p_\phi(\mathbf{x}) = \frac{1}{Z} r_\phi(\mathbf{x}) q(\mathbf{x})$, where Z is the normalizing constant and r_ϕ is an unconstrained positive function. With this parametrization, obviously the optimal r_ϕ equals the density-ratio $\frac{p_{\text{data}}(\mathbf{x})}{q(\mathbf{x})}$. In fact, if $r_\phi(\mathbf{x})$ is trained with density ratio estimation, the normalizing constant Z is simply 1. Therefore, the problem of learning an EBM becomes the problem of estimating a density-ratio, which can be solved by discriminative density ratio estimation. Typically the base distribution $q(\mathbf{x})$ is chosen to be Gaussian, resulted in so-called noise contrastive estimation (NCE) (Gutmann & Hyvärinen, 2012).

Although NCE provides a promising way to train EBMs without running MCMC, the accuracy of the density ratio estimation depends on the closeness between the two distributions. The ratio estimator is often severely inaccurate when the gap between p and q is large (Rhodes et al., 2020).

3. Adaptive Multi-stage Desnity Ratio Estimation

In this section, we introduce our propose adaptive multi-stage density ratio estimation on latent space in details.

[1]Assuming $q(\mathbf{x}) > 0$ when $p(\mathbf{x}) > 0$.

3.1. From NCE to TRE

In NCE, the ratio $r(\mathbf{x}) = \frac{p(\mathbf{x})}{q(\mathbf{x})}$ can be estimated by minimizing

$$\mathcal{L}(\phi) = -\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})} \log\left(\frac{r_\phi(\mathbf{x})}{1+r_\phi(\mathbf{x})}\right) - \mathbb{E}_{\mathbf{x}\sim q(\mathbf{x})} \log\left(\frac{1}{1+r_\phi(\mathbf{x})}\right), \quad (4)$$

where $r_\phi(\mathbf{x})$ is a non-negative ratio estimating model implemented as the exponential of an unconstrained neural network with scalar output. The minimizer ϕ^* satisfies $r_{\phi^*}(\mathbf{x}) = \frac{p(\mathbf{x})}{q(\mathbf{x})}$, and training such a ratio model r_ϕ is equivalent to training the binary classifier D described in Sec. 2.2 (Gutmann & Hyvärinen, 2012).

As discussed in Rhodes et al. (2020), when the gap between p and q is large, the ratio estimator is often severely inaccurate. Intuitively, when there is a big gap between p and q, the task of discriminating samples from them becomes too easy, and thus the estimator is not enforced to capture the information accurately. Motivated by this, Rhodes et al. (2020) propose Telescoping density-Ratio Estimation (TRE), which breaks the density ratio estimation task into a collection of harder sub-tasks. Denoting $p \equiv p_0$ and $q \equiv p_m$, TRE express the density ratio as a telescoping product

$$\frac{p_0(\mathbf{x})}{p_m(\mathbf{x})} = \frac{p_0(\mathbf{x})}{p_1(\mathbf{x})}\frac{p_1(\mathbf{x})}{p_2(\mathbf{x})}\cdots\frac{p_{m-2}(\mathbf{x})}{p_{m-1}(\mathbf{x})}\frac{p_{m-1}(\mathbf{x})}{p_m(\mathbf{x})},$$

where each p_k is chosen such that a classifier cannot easily distinguish it from its two neighbouring densities. An estimate of the original density ratio can be expressed by the product of intermediate ratios:

$$r_\phi(\mathbf{x}) = \prod_{k=0}^{m-1} r_{\phi_k}(\mathbf{x}) \approx \prod_{k=0}^{m-1} \frac{p_k(\mathbf{x})}{p_{k+1}(\mathbf{x})} = \frac{p_0(\mathbf{x})}{p_m(\mathbf{x})}. \quad (5)$$

To train TRE, we need samples from the intermediate distributions $p_k(\mathbf{x})$, which can be obtained by simply taking linear combinations of samples from original distributions p_0 and p_m. The training is done by simultaneously optimizing Eq. 4 for each intermediate density ratio estimation task, which is a multi-task learning problem (Ruder, 2017).

Although TRE makes significant improvement over simple NCE on density ratio estimation, it is still difficult to apply the technique to energy-based modeling. On one hand, EBMs in high-dimensional data space such as image space can be highly complex and multi-modal, making them extremely far away from simple noise distribution. On the other hand, the intermediate distributions are pre-designed through linear transition, making them less effective to connect complicated target densities. In Rhodes et al. (2020), TRE only obtains limited success on training EBMs through density estimation on MNIST dataset.

3.2. Multi-stage density ratio estimation in latent space

Instead of modeling directly on high-dimensional data space, it is easier to introduce low-dimensional latent variables and learn an EBM in latent space, while also learning a mapping from the latent space to the data space (Bengio et al., 2013; Kumar et al., 2019). We follow this approach and attempt to model a latent space EBM using contrastive estimation.

The latent EBM can be learned discriminatively by estimating the ratio between prior density and the posterior density. Due to low-dimensionality of the latent space, such densities can be much easier to deal with than those in high dimensional data space. However, it presents new challenges. Firstly, while the target density in data space is given and fixed (i.e., empirical data distribution), posterior density in latent space is driven by the prior density and the inference on the posterior can be hard. Secondly, while the prior is typically assumed to be un-informative and fixed (e.g., unit Gaussian), the expressiveness of the model is limited.

Inspired by Rhodes et al. (2020), we propose to learn the latent space EBM of the below form through multiple stages

$$p_\phi(\mathbf{z}) = \prod_{k=0}^{m-1} r_{\phi_k}(\mathbf{z})\, p_0(\mathbf{z}), \quad (6)$$

where $p_0(\mathbf{z})$ is the unit Gaussian base distribution, and r_{ϕ_k} is the intermediate density ratio learned in each stage. Such proposed model shares the similar root as the Product-of-Expert (PoE) (Hinton, 2002) where r_{ϕ_k} in each stage can be treated as individual expert model, and it has the potential to produce much sharper distribution than the one with single expert model built such as (Pang et al., 2020a).

Naively, one can follow Rhodes et al. (2020) and train $p_\phi(\mathbf{z})$ by applying TRE on the latent space of a pre-trained generator model, where samples from intermediate distributions are produced by linear combination between prior $p_0(\mathbf{z})$ and inferred aggregated posterior density $q(\mathbf{z})$. However, we empirically observe that the relative improvement over a single NCE is small. Possible reasons include the difficulty of tuning hyper-parameters for multi-task learning and easier matching between two densities in latent space. Moreover, the prior is fixed to be the Gaussian base, renders relatively weaker model. We instead propose multi-stage *adaptive* latent density ratio estimation, which learns different stages of density ratios (each stage corresponds to learning one r_{ϕ_k}) sequentially and adaptively during the training of latent structure. See Sec. 5.5 for an ablation study.

3.3. Learning latent EBMs with adaptive multi-stage density ratio estimation

Our proposed generator model specifies the distribution on joint space $(\mathbf{x}, \mathbf{z})$:

$$p_{\theta,\phi}(\mathbf{x}, \mathbf{z}) = p_\phi(\mathbf{z}) p_\theta(\mathbf{x}|\mathbf{z}), \quad (7)$$

where $p_\phi(z)$ is the prior model specified in Eq. 6, and $\phi = \{\phi_0, \ldots, \phi_{m-1}\}$ that collects parameters for all intermediate learned ratios.

It is tempting to apply maximum likelihood estimation (MLE) to train such model. However, there are several challenges: (1) learning of latent EBM $p_\phi(\mathbf{z})$ needs costly and hard mixing MCMC sampling. (2) the prior $p_\phi(\mathbf{z})$ needs to have a fixed form during training and cannot be adaptively adjusted. To alleviate the aforementioned limitations, we therefore break the density ratio estimation of $p_\phi(\mathbf{z})$ into m stages, learn and build the prior sequentially and adaptively. Specifically, in the k^{th} stage, we consider the generator model of the form

$$p_{\theta,\phi_k}(\mathbf{x}, \mathbf{z}) = p_{\phi_k}(\mathbf{z}) p_\theta(\mathbf{x}|\mathbf{z}), \tag{8}$$

where $p_{\phi_k}(\mathbf{z}) = \prod_{i=0}^{k-1} r_{\phi_i}(\mathbf{z})\, p_0(\mathbf{z})$. The whole training procedure iterates between the maximum likelihood estimation of generation model θ and the sequential contrastive estimation of prior model ϕ.

MLE for generation model θ: The generation model can be trained by maximizing the marginal log-likelihood $p_\theta(\mathbf{x})$. In k^{th} stage, the complete data log-likelihood of the model $p_{\theta,\phi_k}(\mathbf{x}, \mathbf{z})$ can be expressed as

$$\begin{aligned}\log p_{\theta,\phi_k}(\mathbf{x}, \mathbf{z}) &= \log \left[p_{\phi_k}(\mathbf{z}) p_\theta(\mathbf{x}|\mathbf{z})\right] \\ &= \log p_{\phi_k}(\mathbf{z}) - \frac{1}{2}\left[\left\|\mathbf{x} - g_\theta(\mathbf{z})\right\|^2 / \sigma^2\right] + C\end{aligned}$$

where g_θ is the decoder and C is a constant independent of θ. The generation model parameter θ is then updated using the gradient based on Eq. 2 with a batch of training n samples $\mathbf{x}_i$:

$$\theta_{t+1} = \theta_t + \eta_t \sum_{i=1}^{n} \mathbb{E}_{p_{\theta_t}(\mathbf{z}_i|\mathbf{x}_i)} \left[\frac{\partial}{\partial \theta} \log p_{\theta,\phi_k}(\mathbf{x}_i, \mathbf{z}_i) \Big|_{\theta=\theta_t} \right],$$

where η_t is the learning rate. The expectation over the posterior can be approximated by running short-run Lanegvin dynamics in Eq. 3. Note that the running LD to sample from $p_\theta(\mathbf{z}|\mathbf{x})$ is equivalent to sample from $p_\theta(\mathbf{x}, \mathbf{z})$ with fixed $\mathbf{x}$.

Adaptive multi-stage NCE for prior ϕ: The prior model $p_\phi(\mathbf{z})$ can be sequentially and adaptively learned to bridge the gap between prior and posterior densities in the previous stages. Specifically, in k^{th} stage, the correction term r_{ϕ_k} can be trained to estimate the density ratio between $p_{\phi_k}(\mathbf{z})$ and its aggregated posterior $q_k(\mathbf{z})$ through contrastive estimation using Eq. 4. The appealing advantage of this estimator is that it simply trains a binary classifier rather than using expensive MCMC sampling. The optimality of such logistic loss leads to the estimated $r_{\phi_k}(\mathbf{z}) \approx \frac{q_k(\mathbf{z})}{p_{\phi_k}(\mathbf{z})}$.

The prior model in the $(k+1)^{th}$ stage can then be sequentially adapted to match the previous aggregated posterior

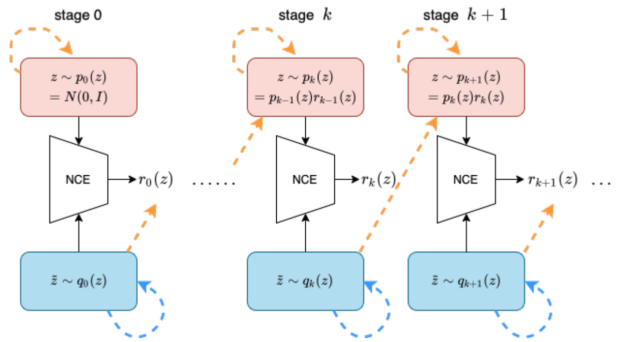

Figure 1. Training adaptive multi-stage density ratio estimation. We estimate the density ratio $r_k(\mathbf{z})$ in each stage using contrastive estimation which trains a classifier to distinguish samples from the prior $p_{\phi_k}(\mathbf{z})$ and samples from the aggregate posterior $q_k(\mathbf{z})$. Posterior samples are obtained by short-run LD (blue dashed curve), prior samples can be obtained either by short-run LD (orange dashed curve) or using persistent chain (orange dashed line). The ratio estimated in stage k can be integrated to form a new prior in stage $k+1$. The whole prior is adapted across multiple stages and learned sequentially.

$q_k(\mathbf{z})$, i.e., $p_{\phi_{k+1}}(\mathbf{z}) = q_k(\mathbf{z}) \approx r_{\phi_k}(\mathbf{z}) p_{\phi_k}(\mathbf{z})$. Given the new prior, we similarly infer the posterior using short-run LD in Eq. 3. Then density ratio estimator $r_{\phi_{k+1}}$ can then be learned through contrastive estimation to match the updated prior and its aggregated posterior which is further used to adapt the prior in next stage. Particularly, we have

$$\frac{q_{m-1}(\mathbf{z})}{p_0(\mathbf{z})} = \frac{q_{m-1}(\mathbf{z})}{p_{\phi_{m-1}}(\mathbf{z})} \frac{q_{m-2}(\mathbf{z})}{p_{\phi_{m-2}}(\mathbf{z})} \cdots \frac{q_0(\mathbf{z})}{p_0(\mathbf{z})},$$

where $q_k(\mathbf{z})$ is the aggregated posterior for prior $p_{\phi_k}(\mathbf{z})$. The above telescoping product holds since the new prior is designed to match the aggregated posterior in previous stage, i.e., $p_{\phi_{k+1}}(\mathbf{z}) \approx q_k(\mathbf{z})$. Each stage estimates the ratio $r_{\phi_k}(\mathbf{z}) \approx \frac{q_k(\mathbf{z})}{p_{\phi_k}(\mathbf{z})}$ via contrasive estimation. Then the aggregated posterior $q_{m-1}(\mathbf{z})$ can be obtained via

$$q_{m-1}(\mathbf{z}) = r_{\phi_{m-1}}(\mathbf{z})\, p_{\phi_{m-1}}(\mathbf{z}) = \prod_{k=0}^{m-1} r_{\phi_k}(\mathbf{z})\, p_0(\mathbf{z}),$$

Our final prior model can then be obtained by matching such aggregated posterior $q_{m-1}(\mathbf{z})$ which has the same form as Eq. 6. The proposed training is illustrated in Figure 1.

Our method shares some high-level ideas with boosting (Freund et al., 1999), as our method can be thought of training weak classifiers sequentially on increasingly harder tasks.

Sampling from prior $p_{\phi_k}(\mathbf{z})$: The density ratio estimation of r_{ϕ_k} in each stage requires the samples from the prior $p_{\phi_k}(\mathbf{z})$ and posterior. The posterior samples are inferred through short-run Langevin dynamics which can be efficient and accurate. For drawing prior samples from $p_{\phi_k}(\mathbf{z})$, we can either use short-run prior Langevin dynamics or persistent update.

One one hand, we could directly utilize the short-run Langevin on the $p_{\phi_k}(\mathbf{z})$ to obtain prior samples. Though costly MCMC is needed, it tends to be much more effective than the existing approach (Pang et al., 2020a), which considers single complex EBM prior that has highly multi-modal energy landscape, making Langevin exploration ineffective and hard to mix. Our prior model evolved from simple unit Gaussian and can be much simpler and less multi-modal for quick mixing of the Langevin sampling.

The samples from prior can also be obtained in a persistent chain manner to avoid the prior Langevin altogether. When introducing the $(k+1)^{th}$ stage of density ratio estimation, we assume that the current estimator $r_{\phi_k}(\mathbf{z})$ performs well in modeling the ratio between the current aggregated posterior distribution $q_k(\mathbf{z})$ and current prior $p_{\phi_k}(\mathbf{z})$. Therefore, we simply use samples from $q_k(\mathbf{z})$ to approximately serve as samples from the new prior $p_{\phi_{k+1}}(\mathbf{z})$ for the learning of $r_{\phi_{k+1}}(\mathbf{z})$. In practice, it is achieved by maintaining a memory matrix that stores a posterior samples $\tilde{\mathbf{z}}_i$ associated to each data point $\mathbf{x}_i$. Note that we only need to keep one memory matrix throughout the training, as only the posterior samples from the previous stage are needed.

Test time sampling: After obtaining the density ratio estimators in each stage and form the final EBM prior $p_\phi(\mathbf{z})$, we can sample latent variables $\mathbf{z} \sim p_\phi(\mathbf{z})$ and produce sample $\mathbf{x}$ by decoding $\mathbf{z}$. Sampling from $p_\phi(\mathbf{z})$ can be done by either running Langvin dynamics with $\nabla_{\mathbf{z}} \log p_\phi(\mathbf{z}) = \nabla_{\mathbf{z}} \left(\sum_{i=0}^{m-1} \log r_{\phi_i}(\mathbf{z}) - \frac{1}{2}\|\mathbf{z}\|^2 \right)$, or Sampling-Importance-Resampling (SIR) techniques.

4. Related Work

Latent variable deep generative models: Our proposed method aims to improve the performance of latent variable deep generative models. Such models consist of a decoder for generation, and require an inference mechanism to infer latent variables. VAEs (Kingma & Welling, 2014; Vahdat & Kautz, 2020) learn the decoder network by simultaneously training a tractable inference network (encoder) to approximate the intractable posterior distribution of the latent variables. Alternatively, Han et al. (2017); Xie et al. (2019); Nijkamp et al. (2020) infer the latent variables by Langevin sampling from the posterior distribution without using a encoder. Our method follows the latter approach that uses Langevin sampling to infer latent variables.

Discriminative contrastive estimation for learning generative models: As introduced in Section 2.2, discriminative contrastive estimation can be applied to learning EBMs. Gao et al. (2020) use a normalizing flow (Papamakarios et al., 2021) as the base distribution for contrastive estimation. Aneja et al. (2021) refine the prior distribution of a pre-trained VAE by noise contrastive estimation. However, such a method may fail if the empirical latent distribution (called aggregated posterior) is far away from the Gaussian noise. Rhodes et al. (2020) propose telescoping density-ratio estimation, which breaks the estimation into several sub-problems. The method is connected to a range of methods leverage sequences of intermediate distributions such as Gelman & Meng (1998); Marinari & Parisi (1992); Kirkpatrick et al. (1983).

Generator model with flexible prior: Our method trains an energy-based prior on the latent space by proposed adaptive multi-stage NCE, so our work is related to the broader line of previous papers on introducing flexible prior distribution. Tomczak & Welling (2018) parameterized the prior based on the posterior inference model, and (Bauer & Mnih, 2019) proposed to construct priors using rejection sampling. Some previous work adopt a two-stage approach, which first trains a latent variable model with simple prior, and then trains a separate prior model to match the aggregated posterior distribution. For example, 2s-VAE (Dai & Wipf, 2019) trains another VAE in the latent space; Ghosh et al. (2019) fit a Gaussian mixture model on latent codes. Additional work in this line include (Oord et al., 2017; Esser et al., 2021; Xiao et al., 2019; 2021; Patrini et al., 2020).

Pang et al. (2020a) have the closest connection to our work. Similar to us, they introduce an EBM on the latent space. Both the latent space EBM and the generator network are learned jointly by maximum likelihood, and in particular the training involves short-run MCMC sampling from both the prior and posterior distributions. In contrast, we sequentially learn a more expressive EBM with our novel adaptive multi-stage NCE, which avoids running MCMC for EBM prior. We also show improved results on image generation and outlier detection tasks.

5. Experiments

In this section, we present a set of experiments which highlight the effectiveness of our proposed method. We want to show that our method can (i) learn an generator model with expressive prior distribution from which visually realistic images can be synthesized, (ii) generalize well by faithfully reconstructing test images during training, and (iii) successfully perform anomaly detection. To show the performance of our method, we mainly include SVHN (Netzer et al., 2011), CelebA (Liu et al., 2015) and CIFAR-10 (Krizhevsky et al., 2010) in our study. Besides, we also include studies on the training dynamics and the Langevin sampling, as well as ablation studies to better understand our method. Details about the experiments, including network architecture, the choices of the model hyper-parameters and the optimization method for each dataset can be found in Appendix A.

5.1. Image Synthesis and Reconstruction

We evaluate the quality of the generated and reconstructed images. Ideally, if the model is well-trained, the EBM prior on latent space will fit the marginal distribution of latent variables, which in turn leads to realistic samples and faithful reconstructions. We benchmark our model against a variety of previous methods including VAE (Kingma & Welling, 2014), Alternating Back-propogation (ABP) (Han et al., 2017) and Short-run Inference (SRI) (Nijkamp et al., 2020) which assume a simple standard Gaussian prior distribution for the latent vector, as well as recent two-stage methods such as 2-stage VAE (Dai & Wipf, 2019), RAE (Ghosh et al., 2019) and NCP-VAE (Aneja et al., 2021), whose prior distributions are learned with posterior samples in a second stage after the generator is trained. We also compare our method with LEBM (Pang et al., 2020a), which learns a EBM prior adaptively during training the generator, while the EBM prior is trained by maximum likelihood instead of density ratio estimation. To make fair comparisons, we follow the protocol as in (Pang et al., 2020a).

Synthesis: We report the quantitative results of FID (Heusel et al., 2017) in Table 1, where we observe that across all datasets, our proposed method achieves superior generation performance compared to baseline models based with simple or learned prior distribution.

We show qualitative results of generated samples in Figure 2, where we observe that our model can generate diverse, sharp and high-quality samples. Additional qualitative samples are presented in Appendix C. To test our method's scalability, we trained a larger generator on CelebA-HQ (128×128) and show samples in Figure 3, and we see that the model can produce realistic samples.

Reconstruction: Note that the posterior Langevin dynamics should not only help to learn the latent space EBM prior model but also produce samples that approximately come from true posterior distribution $p_\theta(\mathbf{z}|\mathbf{x})$ of the generator model. To verify this, we evaluate the accuracy of the posterior inference by looking at reconstruction error on test images. We quantitatively compare reconstructions of test images with baseline models using mean square error (MSE) in Table 1. We observe that our method consistently obtain lower reconstruction error than competing methods do. We also provide qualitative results of reconstruction in Appendix B.

5.2. Anomaly Detection

Anomaly detection is another task that can be used to evaluate the generator model. With a generator and an EBM prior model trained on the in-distribution data, the posterior $p_\theta(\mathbf{z}|\mathbf{x})$ would have separated probability densities for in-distribution and out-of-distribution (anomalous) samples. In particular, we decide whether a test sample $\mathbf{x}$ is anomalous or not by first sampling $\mathbf{z}$ from the posterior $p_\theta(\mathbf{z}|\mathbf{x})$ by short-run Langevin dynamics, and then computing the joint density $p_{\theta,\phi}(\mathbf{x}, \mathbf{z}) = p_\theta(\mathbf{x}|\mathbf{z})p_\phi(\mathbf{z})$. A higher value of log joint density indicates the test sample is more likely to be a normal sample.

Following the experimental settings in (Kumar et al., 2019; Zenati et al., 2018), we set each class in the MNIST dataset as an anomalous class and leave the other 9 classes as normal. Note that it is a challenging task and all previous methods do not perform well. To evaluate the performance, we use the log-posterior density to compute the area under the precision-recall curve (AUPRC) (Fawcett, 2006). We compare our method with related models in Table 2, where we observe that our method obtains significant improvements.

5.3. Analyzing Training Loss

In Figure 4, we plot the evolution of the density ratio estimation loss (Eq. 4) for each stage of estimation during training. Our experiment has 4 estimation stages, resulted in 4 density ratio estimators. We observe that the loss for the first stage, which estimates the density ratio between unit Gaussian prior $p_0(\mathbf{z})$ and aggregated posterior is significantly lower than later stages, which estimate the ratio between the updated prior and updated posterior. This observation is consistent with our intuition: directly discriminating between Gaussian prior and posterior is very easy, while introducing additional stages of estimation make the task more difficult, and hence the estimated density ratio is more reliable.

5.4. Analyzing Langevin Dynamics

In Figure 5, we visualize the transition of Langevin dynamics initialized from $p_0(\mathbf{z})$ towards $p_\phi(\mathbf{z})$ on a model trained on CelebA. The LD iterates for 200 steps, which is longer than the LD for training (30 steps). We expect that with a well-trained $p_\phi(\mathbf{z})$, the trajectory of a Markov chain should transit towards samples of higher quality. Indeed, we observe that the quality of synthesis improves significantly with as the LD progresses. In addition, we observe human faces with different identities along the LD, suggesting that the Markov chain can mix between different modes of the prior distribution. This indicates that the density function of learned EBM prior has a smooth geometry that allows MCMC to mix well.

5.5. Ablation Study

To better understand our proposed method, we conduct ablation study on number of density ratio estimators and training methods. We use CelebA for the ablation experiments.

Number of stages. The most important hyper-parameter of our method is the number of density ratio estimators, or

Table 1. MSE(↓) and FID(↓) obtained from models trained on different datasets. For our reported results, the FID is computed based on 50k generated images and 50k real images and the MSE is computed based on 10k test images.

	SVHN		CelebA		CIFAR-10	
	MSE	FID	MSE	FID	MSE	FID
VAE (Kingma & Welling, 2014)	0.019	46.78	0.021	65.75	0.057	106.37
ABP (Han et al., 2017)	-	49.71	-	51.50	-	-
SRI (Nijkamp et al., 2020)	0.018	44.86	0.020	61.03	-	-
SRI (L=5) (Nijkamp et al., 2020)	0.011	35.32	0.015	47.95	-	-
2s-VAE (Dai & Wipf, 2019)	0.019	42.81	0.021	44.40	0.056	72.90
RAE (Ghosh et al., 2019)	0.014	40.02	0.018	40.95	0.027	74.16
NCP-VAE (Aneja et al., 2021)	0.020	33.23	0.021	42.07	0.054	78.06
LEBM (Pang et al., 2020a)	0.008	29.44	0.013	37.87	0.020	70.15
Adaptive CE (ours)	**0.004**	**26.19**	**0.009**	**35.38**	**0.008**	**65.01**

(a) SVHN (b) CelebA (c) CIFAR-10

Figure 2. Samples generated from our models trained on SVHN, CelebA and CIFAR-10 datasets.

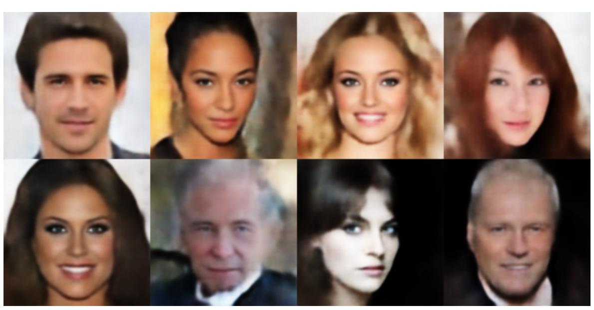

Figure 3. Samples from our model trained on CelebA-HQ.

equivalently, the number of training stages. We present the FID score of models trained with different number of stages in the first part of Table 3. The line of 0 stage means no latent EBM at all, i.e., simply training a generator model by short-run inference and sampling from it by decoding $\mathbf{z} \sim p_0(\mathbf{z})$.

We make the following observations. Firstly, we see that directly sampling latent variables from p_0 leads to poor FID score, while any latent EBM trained by density ratio estimation can significantly improve the performance, suggesting the necessity of learning the latent EBM. Secondly, we see that multi-stage density ratio estimation can further significantly improve the performance of single-stage estimation. The results indicate that multi-stage density ratio estimation facilitate the training of latent EBM by gradually making

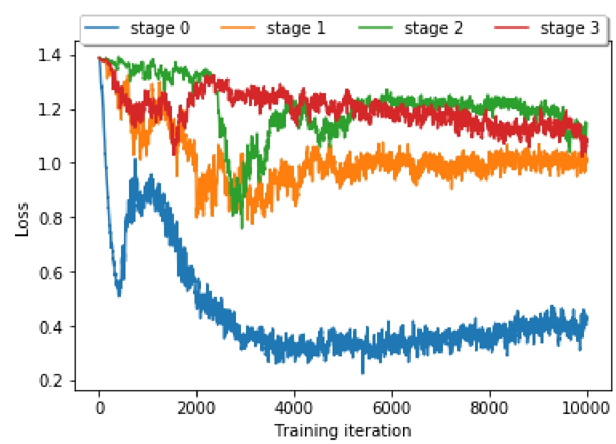

Figure 4. Density ratio estimation loss for each estimation stage.

Table 2. AUPRC(↑) scores for unsupervised anomaly detection on MNIST. Numbers are taken from (Pang et al., 2020a) and results for our model are averaged over last 10 trials to account for variance.

Heldout Digit	1	4	5	7	9
VAE (Kingma & Welling, 2014)	0.063	0.337	0.325	0.148	0.104
ABP (Han et al., 2017)	0.095 ± 0.03	0.138 ± 0.04	0.147 ± 0.03	0.138 ± 0.02	0.102 ± 0.03
MEG (Kumar et al., 2019)	0.281 ± 0.04	0.401 ± 0.06	0.402 ± 0.06	0.290 ± 0.04	0.342 ± 0.03
BiGAN-σ (Zenati et al., 2018)	0.287 ± 0.02	0.443 ± 0.03	0.514 ± 0.03	0.347 ± 0.02	0.307 ± 0.03
LEBM (Pang et al., 2020a)	0.336 ± 0.01	0.630 ± 0.02	0.619 ± 0.01	0.463 ± 0.01	0.413 ± 0.01
Adaptive CE (ours)	**0.531 ± 0.02**	**0.729 ± 0.02**	**0.742 ± 0.01**	**0.620 ± 0.02**	**0.499 ± 0.01**

Figure 5. Transition of Langevin dynamics initialized from $p_0(\mathbf{z})$ towards $p_\phi(\mathbf{z})$ for 200 steps.

Table 3. Results for ablation study on CelebA dataset.

	Number of stages	FID
Adaptive	0	62.78
	1	44.17
	2	39.85
	4	**35.38**
	8	35.84
Non-adaptive	0	62.78
	1	43.84
	2	42.61
	4	42.48
	8	43.06

the estimation task harder. We observe that the FID score does not improve for more than 4 stages, and therefore we choose 4 as the number of stages for our main experiments.

Training method: adaptive vs. non-adaptive. It is important to distinguish our method from TRE in Rhodes et al. (2020). TRE assumes the target distribution to be fixed, therefore, if we adopt TRE, the posterior distribution $p_\theta(\mathbf{z}|\mathbf{x})$ will be a fixed one throughout the training. In contrast, our training method is adaptive in the sense that the target posterior is updated by incorporating the current EBM prior into the joint distribution when a new stage is introduced. To quantitatively compare these two approaches, we also train non-adaptive version of the model and report the numbers in the second part of Table 3. We observe that models trained with non-adaptive multi-stage density ratio estimation obtain significantly worse results. Therefore, we believe that it is crucial to learn density ratios sequentially with adaptive posterior.

5.6. Parameter Efficiency

One potential disadvantage of our method is its parameter inefficiency from multiple estimator networks. Moreover, since the training is sequential, we cannot share parameters between estimators as done in Rhodes et al. (2020). Fortunately, our EBM is on the latent space so the network is light-weighted. For example, with 4 density ratio estimators, the number of parameters in the prior EBM is only around 1% of the number of parameters in the generator. In addition, we confirm the larger number of parameters in the latent EBM is not the cause of improvements, as we train a single stage model with $4\times$ size and observe no improvement.

6. Conclusions

In this paper, we propose adaptive multi-stage density ratio estimation, which is an effective method for learning a EBM prior for a generator model. Our method learns the latent EBMs by introducing multiple density ratio estimators that learn the density ratio between prior and posterior sequentially and adaptively. We demonstrate the effectiveness of our method by conducting comprehensive experiments, and empirical results show the advantage of our method on generation, reconstruction and anomaly detection tasks. As future directions, our method can potentially be applied to modeling the latent space of generator models in other domains, such as text (Pang & Wu, 2021) and graph. We also tend to develop more advanced and efficient inference schemes for posterior density estimation.

References

Aneja, J., Schwing, A., Kautz, J., and Vahdat, A. A contrastive learning approach for training variational autoencoder priors. In *Neural Information Processing Systems (NeurIPS)*, 2021.

Bauer, M. and Mnih, A. Resampled priors for variational autoencoders. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 66–75. PMLR, 2019.

Bengio, Y., Mesnil, G., Dauphin, Y., and Rifai, S. Better mixing via deep representations. In *International conference on machine learning*, pp. 552–560. PMLR, 2013.

Dai, B. and Wipf, D. Diagnosing and enhancing vae models. *arXiv preprint arXiv:1903.05789*, 2019.

Esser, P., Rombach, R., and Ommer, B. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12873–12883, 2021.

Fawcett, T. An introduction to roc analysis. *Pattern recognition letters*, 27(8):861–874, 2006.

Freund, Y., Schapire, R., and Abe, N. A short introduction to boosting. *Journal-Japanese Society For Artificial Intelligence*, 14(771-780):1612, 1999.

Gao, R., Nijkamp, E., Kingma, D. P., Xu, Z., Dai, A. M., and Wu, Y. N. Flow contrastive estimation of energy-based models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7518–7528, 2020.

Gelman, A. and Meng, X.-L. Simulating normalizing constants: From importance sampling to bridge sampling to path sampling. *Statistical science*, pp. 163–185, 1998.

Ghosh, P., Sajjadi, M. S., Vergari, A., Black, M., and Schölkopf, B. From variational to deterministic autoencoders. *arXiv preprint arXiv:1903.12436*, 2019.

Glorot, X. and Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 249–256. JMLR Workshop and Conference Proceedings, 2010.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

Gutmann, M. U. and Hyvärinen, A. Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. *Journal of Machine Learning Research*, 13(11):307–361, 2012.

Han, T., Lu, Y., Zhu, S.-C., and Wu, Y. N. Alternating back-propagation for generator network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Hinton, G. E. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.

Hoffman, M. D. and Johnson, M. J. Elbo surgery: yet another way to carve up the variational evidence lower bound. In *Workshop in Advances in Approximate Bayesian Inference, NIPS*, volume 1, 2016.

Izmailov, P., Kirichenko, P., Finzi, M., and Wilson, A. G. Semi-supervised learning with normalizing flows. In *International Conference on Machine Learning*, pp. 4615–4630. PMLR, 2020.

Karras, T., Laine, S., and Aila, T. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4401–4410, 2019.

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

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. In *The International Conference on Learning Representations (ICLR)*, 2014.

Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P. Optimization by simulated annealing. *science*, 220(4598):671–680, 1983.

Krizhevsky, A., Nair, V., and Hinton, G. Cifar-10 (canadian institute for advanced research). *URL http://www. cs. toronto. edu/kriz/cifar. html*, 5:4, 2010.

Kumar, R., Ozair, S., Goyal, A., Courville, A., and Bengio, Y. Maximum entropy generators for energy-based models. *arXiv preprint arXiv:1901.08508*, 2019.

Liu, Z., Luo, P., Wang, X., and Tang, X. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pp. 3730–3738, 2015.

Marinari, E. and Parisi, G. Simulated tempering: a new monte carlo scheme. *EPL (Europhysics Letters)*, 19(6): 451, 1992.

Neal, R. M. et al. Mcmc using hamiltonian dynamics. *Handbook of markov chain monte carlo*, 2(11):2, 2011.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*, 2011.

Nijkamp, E., Hill, M., Zhu, S.-C., and Wu, Y. N. Learning non-convergent non-persistent short-run mcmc toward energy-based model. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, pp. 5232–5242, 2019.

Nijkamp, E., Pang, B., Han, T., Zhou, L., Zhu, S.-C., and Wu, Y. N. Learning multi-layer latent variable model via variational optimization of short run mcmc for approximate inference. In *European Conference on Computer Vision*, pp. 361–378, 2020.

Oord, A. v. d., Vinyals, O., and Kavukcuoglu, K. Neural discrete representation learning. *arXiv preprint arXiv:1711.00937*, 2017.

Pang, B. and Wu, Y. N. Latent space energy-based model of symbol-vector coupling for text generation and classification. In *International Conference on Machine Learning*, pp. 8359–8370. PMLR, 2021.

Pang, B., Han, T., Nijkamp, E., Zhu, S.-C., and Wu, Y. N. Learning latent space energy-based prior model. *NeurIPS*, 2020a.

Pang, B., Nijkamp, E., Cui, J., Han, T., and Wu, Y. N. Semi-supervised learning by latent space energy-based model of symbol-vector coupling. *arXiv preprint arXiv:2010.09359*, 2020b.

Papamakarios, G., Nalisnick, E., Rezende, D. J., Mohamed, S., and Lakshminarayanan, B. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22(57):1–64, 2021.

Patrini, G., van den Berg, R., Forre, P., Carioni, M., Bhargav, S., Welling, M., Genewein, T., and Nielsen, F. Sinkhorn autoencoders. In *Uncertainty in Artificial Intelligence*, pp. 733–743. PMLR, 2020.

Radford, A., Metz, L., and Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

Rezende, D. J., Mohamed, S., and Wierstra, D. Stochastic backpropagation and approximate inference in deep generative models. In *International conference on machine learning*, pp. 1278–1286. PMLR, 2014.

Rhodes, B., Xu, K., and Gutmann, M. U. Telescoping density-ratio estimation. *arXiv preprint arXiv:2006.12204*, 2020.

Ruder, S. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017.

Saito, M., Saito, S., Koyama, M., and Kobayashi, S. Train sparsely, generate densely: Memory-efficient unsupervised training of high-resolution temporal gan. *International Journal of Computer Vision*, 128:2586–2606, 2020.

Sugiyama, M., Suzuki, T., and Kanamori, T. *Density ratio estimation in machine learning*. Cambridge University Press, 2012.

Tomczak, J. and Welling, M. Vae with a vampprior. In *International Conference on Artificial Intelligence and Statistics*, pp. 1214–1223. PMLR, 2018.

Vahdat, A. and Kautz, J. NVAE: A deep hierarchical variational autoencoder. In *Neural Information Processing Systems (NeurIPS)*, 2020.

Xiao, Z., Yan, Q., and Amit, Y. Generative latent flow. *arXiv preprint arXiv:1905.10485*, 2019.

Xiao, Z., Kreis, K., Kautz, J., and Vahdat, A. Vaebm: A symbiosis between variational autoencoders and energy-based models. In *International Conference on Learning Representations*, 2021.

Xie, J., Gao, R., Zheng, Z., Zhu, S.-C., and Wu, Y. N. Learning dynamic generator model by alternating back-propagation through time. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 5498–5507, 2019.

Zenati, H., Foo, C. S., Lecouat, B., Manek, G., and Chandrasekhar, V. R. Efficient gan-based anomaly detection. *arXiv preprint arXiv:1802.06222*, 2018.

A. Experimental Details

In this section, we introduce the detailed settings of our experiments.

A.1. Datasets

We mainly study our method with SVHN (Netzer et al., 2011) $(32\times32\times3)$, CIFAR-10 (Krizhevsky et al., 2010) $(32\times32\times3)$, and CelebA (Liu et al., 2015) $(64\times64\times3)$. Following Pang et al. (2020a), we use the full training set of SVHN $(73,257)$ and CIFAR-10 $(50,000)$, and take $40,000$ examples of CelebA as training data following (Nijkamp et al., 2019). The training images are resized and scaled to $[-1,1]$.

A.2. Network architectures.

For experiments on SVHN, CelebA and CIFAR-10, each density ratio estimator network has a simple fully-connected structure described in Table 4.

Table 4. Network structures for density ratio estimator. LReLU indicates the Leaky ReLU activation function. The slope in Leaky ReLU is set to be 0.1.

Layers	In-Out Size
Input: z	100
Linear, LReLU	200
Linear, LReLU	200
Linear	1

We let the generator network having a simple deconvolution structure, similar to DCGAN (Radford et al., 2015). The generator network for each dataset is depicted in Table 5.

Table 5. Network structures for the generator networks of SVHN, CelebA, CIFAR-10 (from top to bottom). convT(n) indicates a transposed convolutional operation with n output channels. LReLU indicates the Leaky-ReLU activation function. The slope in Leaky ReLU is set to be 0.2.

Layers	In-Out Size	Stride
Input: x	1x1x100	-
4x4 convT(ngf x 8), LReLU	4x4x(ngf x 8)	1
4x4 convT(ngf x 4), LReLU	8x8x(ngf x 4)	2
4x4 convT(ngf x 2), LReLU	16x16x(ngf x 2)	2
4x4 convT(3), Tanh	32x32x3	2

Layers	In-Out Size	Stride
Input: x	1x1x100	-
4x4 convT(ngf x 8), LReLU	4x4x(ngf x 8)	1
4x4 convT(ngf x 4), LReLU	8x8x(ngf x 4)	2
4x4 convT(ngf x 2), LReLU	16x16x(ngf x 2)	2
4x4 convT(ngf x 1), LReLU	32x32x(ngf x 1)	2
4x4 convT(3), Tanh	64x64x3	2

Layers	In-Out Size	Stride
Input: x	1x1x128	-
8x8 convT(ngf x 8), LReLU	8x8x(ngf x 8)	1
4x4 convT(ngf x 4), LReLU	16x16x(ngf x 4)	2
4x4 convT(ngf x 2), LReLU	32x32x(ngf x 2)	2
3x3 convT(3), Tanh	32x32x3	1

A.3. Training Hyper-parameters

We introduce some hyper-parameter setting for training our model. For the main experiments, we have 4 density ratio estimation stages. We adopt the persistent approach for generating samples from prior distribution. For the posterior sampling Langevin dynamics, we use step size 0.1, and run the LD for 30 steps for SVHN and CelebA, and 40 steps on CIFAR-10.

The parameters for the density ratio estimators and image generators are initialized with Xavier initialization (Glorot & Bengio, 2010). We train both the generator and density ratio estimators using Adam (Kingma & Ba, 2014) optimizer. The learning rate for the generator is $1e-4$ and the learning rate for the density ratio estimator is $5e-5$. We train the model for 100 epochs for SVHN and CelebA, where a new estimation stage is introduced every 25 epochs. For CIFAR-10, we train the model for 200 epochs for SVHN and CelebA, where a new estimation stage is introduced every 50 epochs.

During test stage, we run LD on the learned EBM prior with step size 0.1 for 100 steps.

B. Reconstruction Samples

In Figure 6, we provide some qualitative examples of reconstructing test images. We see that our model can reconstruct unseen images faithfully.

C. Additional Qualitative Results

We provide additional qualitative samples from our models trained on SVHN, CelebA and CIFAR-10 in Figure 7.

Figure 6. Qualitative results of reconstruction on test images. Left: real images from test set. Right: reconstructed images by sampling from the posterior.

Figure 7. Additional randomly generated samples from our models.