Improving Unsupervised Hierarchical Representation with Reinforcement Learning

Ruyi An[1*] **Yewen Li**[1*†] **Xu He**[3] **Pengjie Gu**[1] **Mengchen Zhao**[4]
Dong Li[3] **Jianye Hao**[3] **Chaojie Wang**[1,2] **Bo An**[1,2] **Mingyuan Zhou**[5]
[1]Nanyang Technological University [2]Skywork AI, Singapore [3]Huawei Noah's Ark Lab
[4]South China University of Technology [5]The University of Texas at Austin

https://github.com/ruyianry/rep_hierarchy_rl

Abstract

Learning representations to capture the very fundamental understanding of the world is a key challenge in machine learning. The hierarchical structure of explanatory factors hidden in data is such a general representation and could be potentially achieved with a hierarchical VAE. However, training a hierarchical VAE always suffers from the "posterior collapse", where the data information is hard to propagate to the higher-level latent variables, hence resulting in a bad hierarchical representation. To address this issue, we first analyze the shortcomings of existing methods for mitigating the posterior collapse from an information theory perspective, then highlight the necessity of regularization for explicitly propagating data information to higher-level latent variables while maintaining the dependency between different levels. This naturally leads to formulating the inference of the hierarchical latent representation as a sequential decision process, which could benefit from applying reinforcement learning (RL). Aligning RL's objective with the regularization, we first introduce a skip-generative path to acquire a reward for evaluating the information content of an inferred latent representation, and then the developed Q-value function based on it could have a consistent optimization direction of the regularization. Finally, policy gradient, one of the typical RL methods, is employed to train a hierarchical VAE without introducing a gradient estimator. Experimental results firmly support our analysis and demonstrate that our proposed method effectively mitigates the posterior collapse issue, learns an informative hierarchy, acquires explainable latent representations, and significantly outperforms other hierarchical VAE-based methods in downstream tasks.

1. Introduction

Deriving meaningful representations of data with minimal supervision is a central challenge in machine learning [5], while existing research has predominantly concentrated on the discriminative approach [10] that relies on meticulously crafted preprocessing pipelines like pretext tasks [22, 30, 62, 65, 95] and data augmentations [3, 59]. Methods such as contrastive learning [8, 9, 11, 12, 14–16, 27, 29, 32, 34, 36, 49, 87, 97, 98] exemplify the success of this discriminative approach. However, representations derived from these methods are only limited to tasks invariant to the preprocessing pipelines [24], *e.g.* representations learned with the random cropping data augmentation cannot be applied to pixel-level localization tasks [83, 84], limiting their broader applicability. To transcend these confines, Bengio et al. [5] advocate for the pursuit of a universal and fundamental understanding of the world-a "*general-purpose prior*"-enabling the direct learning of representations without prior knowledge or assumptions about downstream tasks. Such representations could be learned by generative ap-

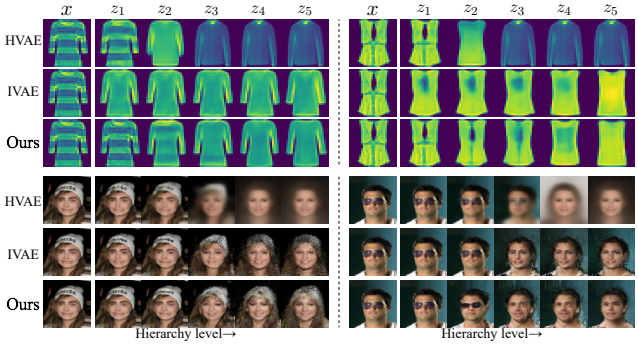

Figure 1. Visualization of different levels' latent representations of hierarchical VAEs on FashinMNIST and CelebA [53] to demonstrate the learned hierarchical structure. HVAE fails to learn a 5-level hierarchy, where higher-level latent representations display ***posterior collapse***, collapsing into the same modes of "clothing" and "female faces" regardless of the input. For IVAE, reconstructions of certain consecutive layers appear highly similar, suggesting a disrupted hierarchy. In contrast, our method preserves detailed information of inputs at low levels and captures increasingly abstract semantics at higher levels, mitigating *posterior collapse* and establishing an informative hierarchy.

[*]Equal contribution, authors listed alphabetically by last name.
[†]Corresponding to: Yewen Li <*yewen001@e.ntu.edu.sg*>.

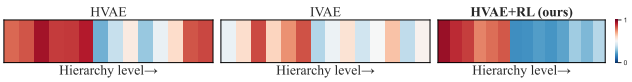

Figure 2. Absolute Pearson's correlation between data representations of the inference network of a hierarchical VAE trained on FashionMNIST and another trained on MNIST [48], given the same data point from FashionMNIST. Our method enables the model to learn low-level information shared on different datasets and high-level semantic information in a hierarchical manner.

proaches [47, 79], which utilize a deep generative model [21, 31, 40, 43, 68, 81] to capture the posterior distribution of the underlying explanatory factors in the observed data, *i.e.*, *the general-purpose prior*, such as the hierarchical structure [76, 85, 86], disentanglement [38], and temporal and spatial coherence [71]. In this work, we focus on one specific aspect of this *general-purpose prior*, the hierarchical structure of data, which could be potentially achieved by learning a hierarchy of multi-level abstract latent representations with a hierarchical VAE [5, 79].

However, actualizing a meaningful hierarchical structure with a hierarchical VAE faces a considerable issue named "*posterior collapse*" [51, 57, 76], where the data information fails to be propagated to higher-level latent variables. Specifically, their posterior distribution $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ would be equal to an uninformative prior distribution $p(\boldsymbol{z})$ [13, 51, 82], undermining hierarchical VAEs' capability to capture meaningful representations. Existing literature proposing mitigation approaches for *posterior collapse* mainly focuses on two aspects: the limited capacity of the model architecture [19, 20, 57, 67, 76, 80] and the improper training objective [20, 41, 51, 70, 88]. However, our theoretical analysis from the perspective of information theory [18] presented in Section 3.1 suggests that current methodologies focusing on model architectures fail to explicitly enforce regularization on the inference of higher-level latent variables, where *posterior collapse* may still occur; and approaches targeting training objectives might disrupt the hierarchical dependencies between latent variables, compromising the model's structural integrity. Experimental results for better understanding can be seen in Figs. 1 and 2. These limitations contribute to the significant lag in the quality and performance of learned representation using hierarchical VAEs and have resulted in their limited adoption in representation learning tasks recently [47, 79].

To tackle the issue of *posterior collapse* that hinders acquiring an informative hierarchical latent representation, where different levels' latent representations are inferred sequentially from inputs and high-level latent representations are reliant on the abstraction of low-level ones, it is imperative to consider not only the information content of an inferred single-level latent variable but also its impact on subsequent, higher-level latent variables. This naturally inspires us to formulate the inference of the hierarchical latent representation as a sequential decision process. Inspired by the huge success of reinforcement learning (RL) in learning a high-quality sequence, we tried to apply RL to the learning of a hierarchical VAE for a better hierarchical latent representation. Please note that although some methods have also successfully introduced RL into tasks related to generative models [6, 50, 94], none of these methods can be directly applied to train a hierarchical VAE with a focus on learning informative hierarchical latent presentations. A detailed discussion can be found in Appendix A.3.

Overall, the contribution of this work could be summarized as follows:

- We highlight limitations of existing approaches of addressing *posterior collapse* for an informative hierarchical latent representation from an information theory perspective and appeal for more effective regularization of the inference process to explicitly propagate data information to higher-level latent variables and maintain the dependency between different levels .To our knowledge, this work is the first to conceptualize the hierarchical VAE's inference as a sequential decision process and employ an RL approach, specifically *policy gradient*, to regularize this process. Our method is broadly applicable to both primary types of hierarchical VAEs: top-down and bottom-up structures;
- Technically, we set up an RL formulation tailored to hierarchical VAEs training, notably with a *skip-generative path* that skips its lower levels' latent variables by marginalization to acquire a reward assessing the information content of the inferred latent variables without introducing additional networks or modules; then, the Q-value function developed based on it could have a consistent optimization direction of the regularization;
- Experiments demonstrate the mitigation of the *posterior collpase*, learned informative hierarchy, explanation ability of latent representations, and superior performance in VAE-based representation learning tasks.

2. Preliminaries

2.1. Hierarchical Variational Autoencoder

An L-layer hierarchical VAE is an extension of the vanilla VAE [43, 68], where the generative process is extended to multi-step generation, formulated as $p_\theta(\boldsymbol{x}, \boldsymbol{z}_{1:L}) = p_\theta(\boldsymbol{x}|\boldsymbol{z}_1)p_\theta(\boldsymbol{z}_L)\prod_{l=1}^{L-1} p_\theta(\boldsymbol{z}_l|\boldsymbol{z}_{l+1})$. The $p_\theta(\boldsymbol{z}_L)$ is typically modeled as a multivariate Gaussian distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$ and θ represents the parameters of the decoder. The inference process of hierarchical VAEs could be implemented in two main approaches, bottom-up (BU) and top-down (TD):

$$\text{BU: } q_\phi(\mathbf{z}|\boldsymbol{x}) = q_\phi(\boldsymbol{z}_1|\boldsymbol{x})\prod_{l=2}^{L} q_\phi(\boldsymbol{z}_l|\boldsymbol{z}_{l-1}, \boldsymbol{x}), \quad (1)$$

$$\text{TD: } q_\phi(\mathbf{z}|\boldsymbol{x}) = q_\phi(\boldsymbol{z}_L|\boldsymbol{x})\prod_{l=L}^{2} q_\phi(\boldsymbol{z}_{l-1}|\boldsymbol{z}_l, \boldsymbol{x}), \quad (2)$$

where ϕ represents the parameters of the encoder. We provide a demonstration of these processes in Fig. 3. As

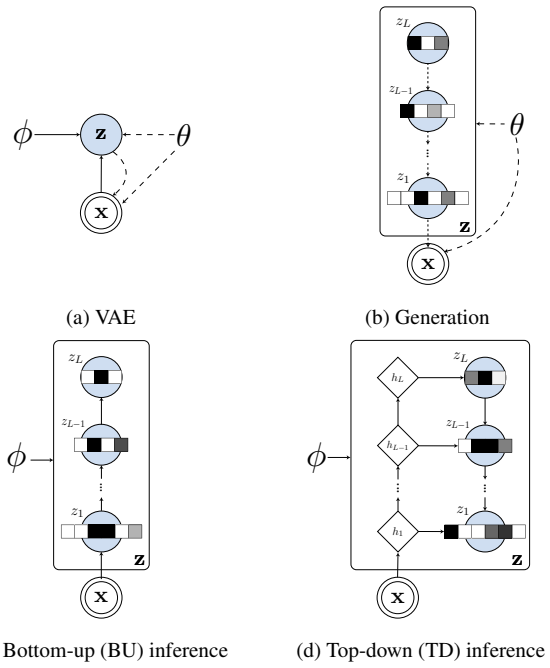

Figure 3. Probabilistic graphical illustration of VAE and hierarchical VAE during inference and generation. Solid lines (→) denote inference while dashed lines (⇢) denote generation.

the inference process is to produce a sequence of latent variables step by step, we denote a hierarchy $\boldsymbol{z}^{1:T} = \{\boldsymbol{z}^1, ..., \boldsymbol{z}^t, ..., \boldsymbol{z}^T\}$, where $T = L$. In BU inference scheme, $\boldsymbol{z}^{1:T} = \{\boldsymbol{z}_1, ..., \boldsymbol{z}_t, ..., \boldsymbol{z}_L\}$; in TD inference scheme, $\boldsymbol{z}^{1:T} = \{\boldsymbol{z}_L, ..., \boldsymbol{z}_t, ..., \boldsymbol{z}_1\}$. For brevity, we use $\mathbf{z}$ to represent $\boldsymbol{z}^{1:T}$ in the following parts. Additionally, we introduce j for representing the lower level variable $\boldsymbol{z}_j$ *w.r.t.* $\boldsymbol{z}^t$ defined by

$$j = \begin{cases} t-1, & \text{in BU hierarchical VAE,} \\ T-t, & \text{in TD hierarchical VAE.} \end{cases} \tag{3}$$

The training objective of a hierarchical VAE is to maximize the variational evidence lower bound (ELBO), denoted as $\mathcal{L}_{\boldsymbol{x}}$, of the training data's marginal log-likelihood $\log p(\boldsymbol{x})$ by jointly updating the parameters θ and ϕ, expressed as:

$$\begin{aligned} \mathcal{L}_{\boldsymbol{x}} &= \log p(\boldsymbol{x}) - D_{\mathrm{KL}}[q_\phi(\mathbf{z}|\boldsymbol{x})||p_\theta(\mathbf{z}|\boldsymbol{x})] \\ &= \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}|\mathbf{z})] - D_{\mathrm{KL}}[q_\phi(\mathbf{z}|\boldsymbol{x})||p(\mathbf{z})], \end{aligned} \tag{4}$$

where $D_{\mathrm{KL}}(\cdot||\cdot)$ refers to the Kullback–Leibler (KL) divergence. Defining $q_\phi(\boldsymbol{z}_1|\boldsymbol{z}_0, \boldsymbol{x}) := q_\phi(\boldsymbol{z}_1|\boldsymbol{x})$ and $p_\theta(\boldsymbol{z}_L|\boldsymbol{z}_{L+1}) := p_\theta(\boldsymbol{z}_L)$, for BU and TD hierarchical VAEs, the respective detailed expressions of their ELBOs are:

$$\begin{aligned} \text{BU: } \mathcal{L}_{\boldsymbol{x}} &= \mathbb{E}_{q_\phi(\mathbf{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}|\boldsymbol{z}_1)] \\ &\quad - \textstyle\sum_{l=1}^{L} D_{\mathrm{KL}}[(q_\phi(\boldsymbol{z}_l|\boldsymbol{z}_{l-1}, \boldsymbol{x})||p_\theta(\boldsymbol{z}_l|\boldsymbol{z}_{l+1}))], \end{aligned} \tag{5}$$

$$\begin{aligned} \text{TD: } \mathcal{L}_{\boldsymbol{x}} &= \mathbb{E}_{q_\phi(\mathbf{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}|\boldsymbol{z}_1)] \\ &\quad - \textstyle\sum_{l=1}^{L} D_{\mathrm{KL}}[(q_\phi(\boldsymbol{z}_l|\boldsymbol{z}_{l+1}, \boldsymbol{x})||p_\theta(\boldsymbol{z}_l|\boldsymbol{z}_{l+1}))]. \end{aligned} \tag{6}$$

After convergence, we could extract level l's latent representation by the maximum a *posteriori* (MAP) estimation of the learned latent variables $\boldsymbol{z}_l$, defined as $\boldsymbol{z}_l^* = \arg\max_{\boldsymbol{z}_l} q_\phi(\boldsymbol{z}_l|\boldsymbol{x})$ [5, 43].

However, training hierarchical VAEs often suffers from the issue known as *posterior collapse*, which can be formally defined as follows.

Definition 1 (*posterior collapse* [7, 13, 82, 88])**.** Given a hierarchical VAE $p(\boldsymbol{x}, \mathbf{z}; \theta, \phi)$, paramters' value $\phi = \hat{\phi}$, $\theta = \hat{\theta}$, and for any data $\boldsymbol{x}$ in a dataset $\mathcal{D}_{\boldsymbol{x}} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, .., \boldsymbol{x}_N\}$, the posterior of inference step t's latent variable $\boldsymbol{z}^t$ collapses if

$$q_\phi(\boldsymbol{z}^t|\boldsymbol{z}^{t-1}, \boldsymbol{x}; \hat{\phi}) = p_\theta(\boldsymbol{z}^t|\boldsymbol{z}_{j+2}; \hat{\theta}), \forall \boldsymbol{x} \in \mathcal{D}_{\boldsymbol{x}}. \tag{7}$$

The *posterior collapse* issue mainly occurs in higher-level latent variables of hierarchical VAEs. Stochastic latent variables $\mathbf{z}$ can be manually divided into lower-level variables, $\boldsymbol{z}_{<k} = \{\boldsymbol{z}_1, ..., \boldsymbol{z}_{k-1}\}$ and higher-level ones, $\boldsymbol{z}_{\geq k} = \{\boldsymbol{z}_k, ..., \boldsymbol{z}_L\}$ where $k \in \{2, ..., L\}$. Though $z_{<k}$ may not suffer from *posterior collapse* that could encode sufficient information to reconstruct the inputs well, the corresponding high-level stochastic latent variables $\mathbf{z}_{\geq k}$ could be inclined to *collapse* into priors, *i.e.*, $q_\phi(\boldsymbol{z}_{\geq k}|\boldsymbol{x}) \approx p_\theta(\boldsymbol{z}_{\geq k})$. An illustration can be seen in Fig. 1. Consequently, these higher-level posteriors will hold limited relevance to its input $\boldsymbol{x}$, resulting in non-informative representations.

2.2. Reinforcement Learning

We consider a standard RL setup in continuous action space and discrete timesteps, formalized as a Markov decision process (MDP) defined by a tuple $\langle \mathcal{S}, \mathcal{A}, P, r \rangle$ [33, 52, 72, 94]. Here, the state space $\mathcal{S}$ and action space $\mathcal{A}$ are continuous, the state transition probability function $P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ represents the probability density of the next state $s_{t+1} \in \mathcal{S}$ given the current state $s_t \in \mathcal{S}$ and action $a_t \in \mathcal{A}$, and $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ denotes the reward function on each transition.

At each time step $t \in \{1, 2, ..., T\}$, an agent observes a state $s_t \in \mathcal{S}$, takes an action $a_t \in \mathcal{A}$, and receives a scalar reward $r_t : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$. The state s_t could be described in various ways, such as the entire history of the observations $s_t = (o_1, ..., o_t)$ [94] or only the current observation $s_t = o_t$ [33]. An agent's behavior could be defined by a policy $\pi(a_t|s_t)$, which maps states to a probability distribution over actions. The return at time step t is defined as the sum of the discounted future reward $R_t = \sum_{i=t}^{T} \gamma^{(i-t)} r(s_i, a_i)$ with a discount factor $\gamma \in (0, 1)$. The expectation of the return could be flexibly formulated by the Q-value function $Q^\pi(s_t, a_t)$ after taking an action a_t in state s_t and thereafter following policy π:

$$\begin{aligned} Q^\pi(s_t, a_t) &= \mathbb{E}_{s_{i>t}\sim P, a_{i>t}\sim\pi}[R_t|s_t, a_t] \\ &= r(s_t, a_t) + \gamma \mathbb{E}_{s_{t+1}\sim P, a_t\sim\pi}[Q^\pi(s_{t+1}, a_{t+1})], \end{aligned} \tag{8}$$

which not only considers the reward of the action to be taken on the current state but also its effect on the future decision process. Let us assume that drawing from $a_t \sim \pi(a_t|s_t)$ can

be realized with reparameterization as $a_t = f(s_t, \epsilon_t), \epsilon_t \sim p(\epsilon)$. We could directly apply the *policy gradient* method [52, 78, 94] on the model parameters Θ with gradient ascent to maximize the $Q^\pi(s_t, a_t)$ at a time step t:

$$\Theta \leftarrow \Theta + \nabla_\Theta Q^\pi(s_t, a_t). \tag{9}$$

We sum all time step t's policy gradient together to represent the optimization direction $\nabla_\Theta \mathcal{J}$ of the RL, expressed as

$$\nabla_\Theta \mathcal{J} \simeq \sum_{t=1}^{T} \mathbb{E}_{s_t \sim P, \epsilon_t \sim p(\epsilon)} \nabla_\Theta Q^\pi(s_t, a_t),\ a_t = f(s_t, \epsilon_t). \tag{10}$$

For the discussion on **Related Works**, we move it to Appendix A.

3. Improving Unsupervised Hierarchical Representation with RL

We first provide an analysis of the existing two major approaches mitigating *posterior collapse* for learning a hierarchical representation with a hierarchical VAE. However, we found that they either lack regularization for propagating data information to higher-level latent variables or break the dependency between different levels' latent variables. Therefore, to address these drawbacks, a better training objective from the perspective of information theory could naturally lead to an RL scheme with a proper design of the Q-value function as the regularization of latent representations. Finally, we develop an RL training scheme with the policy gradient method to train a hierarchical VAE.

3.1. Analysis of Representation Learning with Hierarchical VAEs

The aim of representation learning with a hierarchical VAE is to learn a hierarchy of latent variables $\mathbf{z}$ that could represent the input data $\boldsymbol{x}$. We begin with rethinking how different approaches are affecting the representation learning with a hierarchical VAE. We first rewrite the ELBO from an information theory standpoint [18]:

$$\begin{aligned}
&\mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x})}[\text{ELBO}(\boldsymbol{x})] \\
=&\mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x})}[\mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\boldsymbol{x})} \log p_\theta(\boldsymbol{x}|\boldsymbol{z}_1)] - \mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x})}[D_{\text{KL}}(q_\phi(\mathbf{z}|\boldsymbol{x})||p(\mathbf{z}))] \\
=&\mathcal{I}_{p,q}(\boldsymbol{x}, \boldsymbol{z}_1) - \mathcal{I}_q(\boldsymbol{x}, \boldsymbol{z}_1) - D_{\text{KL}}(q(\mathbf{z})||p(\mathbf{z})) - \mathcal{H}_p(\boldsymbol{x}),
\end{aligned} \tag{11}$$

where the mutual information between $\boldsymbol{x}$ and $\boldsymbol{z}$ under different distribution is defined by

$$\begin{aligned}
\mathcal{I}_{p,q}(\boldsymbol{x}, \boldsymbol{z}_1) &= \mathbb{E}_{p(\boldsymbol{x})q_\phi(\boldsymbol{z}_1|\boldsymbol{x})}[\log \frac{p(\boldsymbol{x}, \boldsymbol{z}_1)}{p(\boldsymbol{x})q(\boldsymbol{z}_1)}] \\
&= \mathbb{E}_{p(\boldsymbol{x})q_\phi(\boldsymbol{z}_1|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}|\boldsymbol{z}_1)] - \mathbb{E}_{p(\boldsymbol{x})}[\log p(\boldsymbol{x})],
\end{aligned} \tag{12}$$

$$\begin{aligned}
\mathcal{I}_q(\boldsymbol{x}, \boldsymbol{z}_1) &= \mathbb{E}_{q(\boldsymbol{x})q_\phi(\boldsymbol{z}_1|\boldsymbol{x})}[\log \frac{q(\boldsymbol{x}, \boldsymbol{z}_1)}{q(\boldsymbol{x})q(\boldsymbol{z}_1)}] \\
&= \mathbb{E}_{q(\boldsymbol{x})q_\phi(\boldsymbol{z}_1|\boldsymbol{x})}[\log q_\phi(\boldsymbol{x}|\boldsymbol{z}_1)] - \mathbb{E}_{q(\boldsymbol{x})}[\log q(\boldsymbol{x})],
\end{aligned} \tag{13}$$

where $q(x)$ is the training data's distribution and it is assumed to be equal to the true data distribution $p(\boldsymbol{x})$; $p_\theta(\boldsymbol{x}|\boldsymbol{z}_1)$ and $q_\phi(\boldsymbol{x}|\boldsymbol{z}_1)$ could be expressed by

$$p_\theta(\boldsymbol{x}|\boldsymbol{z}_1) = \frac{p_\theta(\boldsymbol{z}_1|\boldsymbol{x})p(\boldsymbol{x})}{q(\boldsymbol{z}_1)} \tag{14}$$

$$q_\phi(\boldsymbol{x}|\boldsymbol{z}_1) = \frac{q_\phi(\boldsymbol{z}_1|\boldsymbol{x})q(\boldsymbol{x})}{q(\boldsymbol{z}_1)} \simeq \frac{q_\phi(\boldsymbol{z}_1|\boldsymbol{x})p(\boldsymbol{x})}{q(\boldsymbol{z}_1)}. \tag{15}$$

Thus, the target of the ELBO in Eq. 11 effectively becomes minimizing the gap between the bottom-most latent variable's posterior $q_\phi(\boldsymbol{z}_1|\boldsymbol{x})$ and its corresponding true posterior $p_\theta(\boldsymbol{z}_1|\boldsymbol{x})$ and meanwhile the posterior of higher-level latent variables $\{\boldsymbol{z}_l\}_{l=1:L}$ are trained to be close to their respective priors by the $D_{\text{KL}}[q(\mathbf{z})||p(\mathbf{z})]$ term.

Since the latent variables above $\boldsymbol{z}_1$ do not receive direct regularization to maximize $\mathcal{I}(\boldsymbol{x}, \boldsymbol{z}_l), l > 1$, it could potentially lead to the *posterior collapse* phenomenon. Therefore, approaches that only modify the model structure [25, 57, 76, 80] still do not impose additional regularization to higher-level latent variables in the training objective, where the *posterior collapse* could still occur.

Existing regularization strategies in the literature aim to refine the posterior distribution in order to cultivate good representations [41, 51, 54]. These methods typically focus on the relationship between observed data $\boldsymbol{x}$ and a single-level latent variable $\boldsymbol{z}^t$, which could be interpreted as an additional regularization on the ELBO expressed as

$$\mathcal{I}_q(\boldsymbol{x}, \mathbf{z}) \simeq \sum_{t=1}^{T} \mathcal{I}_q(\boldsymbol{x}, \boldsymbol{z}^t). \tag{16}$$

However, by optimizing all latent variables $\{\boldsymbol{z}^t\}_{t=1:T}$ with a shared objective $\mathcal{I}(\boldsymbol{x}, \boldsymbol{z}^t)$ simultaneously, these approaches may hurt the hierarchical dependency amongst them. This is due to the implicit incentive for all latent variables to assimilate, potentially undermining the hierarchical structure. Learning to generate a sequence $\{\boldsymbol{x}, \boldsymbol{z}^1, ..., \boldsymbol{z}^T\}$ with only one loss after finishing generating the whole sequence could also bring the imbalance issue, where the concentration of optimization to the mediate items is not clear [94]. These limitations hinder the exploitation of the full capabilities inherent in hierarchical VAEs.

We leave a detailed discussion for the above two approaches in Appendix B.

In summary, an effective training scheme should maximize the information content hidden in latent variables of all layers, while maintaining the dependencies among latent variables. Specifically, the regularization on $\boldsymbol{z}^t$ should account for not only $\mathcal{I}(\boldsymbol{x}, \boldsymbol{z}^t)$, but also the influence on the subsequent latent variables $\mathcal{I}(\boldsymbol{x}, \boldsymbol{z}^i|\boldsymbol{z}^{i-1}), i > t$, *i.e.*,

$$\mathcal{I}_q(\boldsymbol{x}, \boldsymbol{z}^{t:T}) = \sum_{i=t}^{T} \mathcal{I}_q(\boldsymbol{x}, \boldsymbol{z}^i|\boldsymbol{z}^{i-1}). \tag{17}$$

By incorporating a discount factor $\gamma \in (0, 1)$, which allocates greater weight to the mutual information with proximal latent variables, the regularization for $\boldsymbol{z}^t$ becomes:

$$\mathcal{L}_\mathcal{I}(\boldsymbol{x}, \boldsymbol{z}^t) = \sum_{i=t}^{T} \gamma^{i-t} \mathcal{I}(\boldsymbol{x}, \boldsymbol{z}^i|\boldsymbol{z}^{i-1}). \tag{18}$$

Therefore, the optimization direction of learning the se-

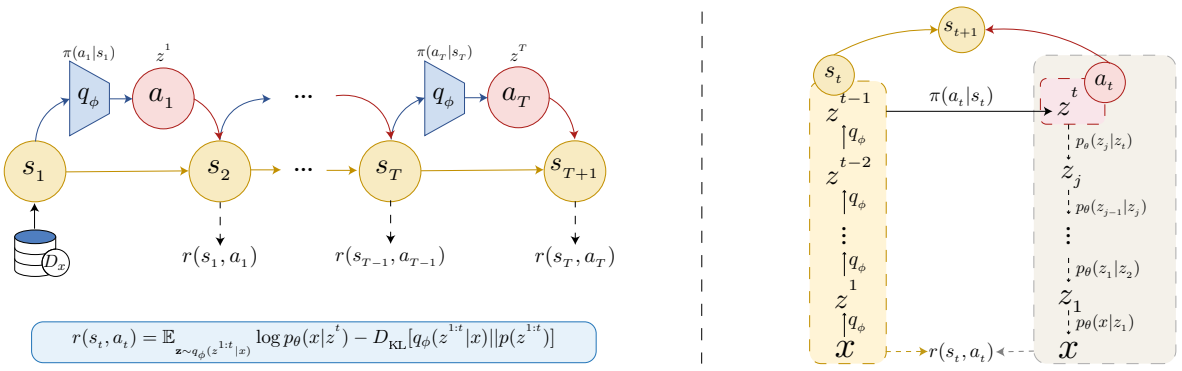

Figure 4. Illustration of modeling the inference process of a hierarchical VAE as a sequential decision process (**left**) and the details of the elements of RL s_t, a_t, s_{t+1} (**right**). A reward $r(s_t, a_t)$ is derived with the *skip-generative path* depicted in the gray box.

quence $\{\boldsymbol{x}, \boldsymbol{z}^1, ..., \boldsymbol{z}^T\}$ can then be expressed by

$$\nabla_{\phi,\theta}\mathcal{J} \simeq \sum_{t=1}^{T} \mathbb{E}_{p(\boldsymbol{x}),q_\phi,p_\theta} \nabla_{\phi,\theta}\mathcal{L}_{\mathcal{I}}(\boldsymbol{x}, \boldsymbol{z}^t). \quad (19)$$

The resemblance of the optimization direction to Eq. 10 naturally leads to formulating the learning of hierarchical VAE as a sequential decision process, prompting the application of RL methods. Particularly, we are driven to develop a proper Q-value function $Q(s_t, a_t)$ to play the role of the regularization $\mathcal{L}_{\mathcal{I}}(\boldsymbol{x}, \boldsymbol{z}^t)$ to achieve a consistent expression of the two optimization objectives.

3.2. Training Hierarchical VAEs with RL

Inspired by the tremendous success of RL methods in stably propagating information along long Markov Decision chains that maximize the cumulative rewards tallied at every step, *i.e.*, the Q-value $Q^\pi(s_t, a_t)$ in Eq. 8, we formulate the progressive inference process of an L-layer hierarchical VAEs as an L-step sequential decision process. Specifically, given an input $\boldsymbol{x}$ from a training set $\mathcal{D}_{\boldsymbol{x}}$, we need a hierarchical VAE parameterized with $\{\theta, \phi\}$ to produce a sequence $Y_{1:T} = \{\boldsymbol{x}, \boldsymbol{z}^1, \boldsymbol{z}^2, ..., \boldsymbol{z}^T\}, T = L$.

Now, we can set up the basic elements for training with RL. At each time step t, the state s_t is characterized as a tuple including the observed input and already produced latent variables $\boldsymbol{z}^{1:t-1} = \{\boldsymbol{z}^1, ..., \boldsymbol{z}^{t-1}\}$ up to the step:

$$s_t \triangleq \{\boldsymbol{x}, \boldsymbol{z}^{1:t-1}\}, \quad (20)$$

where $s_1 = \{\boldsymbol{x}\}$. After receiving a state s_t, the encoder $q_\phi(\mathbf{z}|\boldsymbol{x})$ of the hierarchical model acts as the policy network $\pi(a_t|s_t)$ to produce the latent variable as an action:

$$\pi_\phi(a_t|s_t) \triangleq q_\phi(\boldsymbol{z}^t|\boldsymbol{x}, \boldsymbol{z}^{1:t-1}) = q_\phi(\boldsymbol{z}^t|\boldsymbol{z}^{t-1}, \boldsymbol{x}), \quad (21)$$

$$a_t \triangleq z^t \sim q_\phi(\boldsymbol{z}^t|\boldsymbol{z}^{t-1}, \boldsymbol{x}). \quad (22)$$

Once executing the a_t, we will get to the next state according the state transition probability function P, specified as

$$P(s_{t+1}|s_t, a_t) \triangleq (\delta_{\boldsymbol{x}}, \delta_{\boldsymbol{z}^{1:t}}), \quad (23)$$

which is deterministic and reflects the unchanged observed data $\boldsymbol{x}$ and the inclusion of new inferred latent variables at the successor state. δ_v is a Dirac delta distribution that only manifests a non-zero impulse at v.

Reward function design. When getting such a transition $\{s_t, a_t, s_{t+1}\}$, we need to design a reward function $r(s_t, a_t)$. The principle of the design is to evaluate the information content $\mathcal{I}(\boldsymbol{x}, \boldsymbol{z}^t|\boldsymbol{z}^{t-1})$ hidden in the latent variable $\boldsymbol{z}^t$ that represents the inputs, then we could formulate the Q-value function to regularize $\boldsymbol{z}^t$ by $\sum_{i=t}^{T} \mathcal{I}(\boldsymbol{x}, \boldsymbol{z}^i|\boldsymbol{z}^{i-1})$. We design a reward function $r(s_t, a_t)$ to evaluate the information between $\boldsymbol{x}$ and $\boldsymbol{z}^t$ as below:

$$r(s_t, a_t) = \mathbb{E}_{\mathbf{z}\sim q_\phi(\boldsymbol{z}^{1:t}|\boldsymbol{x})} \log p_\theta(\boldsymbol{x}|\boldsymbol{z}^t) - D_{\mathrm{KL}}[q_\phi(\boldsymbol{z}^{1:t}|\boldsymbol{x})||p(\boldsymbol{z}^{1:t})], \quad (24)$$

where $p_\theta(\boldsymbol{x}|\boldsymbol{z}^t)$ is a *skip-generative path*, expressed as

$$p_\theta(\boldsymbol{x}|\boldsymbol{z}^t) = \int_{\boldsymbol{z}_{1:j}} p_\theta(\boldsymbol{x}|\boldsymbol{z}_1) \prod_{i=j}^{1} p_\theta(\boldsymbol{z}_i|\boldsymbol{z}_{i+1}), \boldsymbol{z}^t \sim q(\boldsymbol{z}^t|\boldsymbol{z}^{<t}, \boldsymbol{x}),$$

in which $\boldsymbol{z}_{1:j}$ are marginalized out by the integral. Kindly note that the *skip-generative path* $p_\theta(\boldsymbol{x}|\boldsymbol{z}^t)$ is different from the generative path in vanilla hierarchical VAEs, where $p_\theta(\boldsymbol{x}|\boldsymbol{z}^{1:t}) = p_\theta(\boldsymbol{x}|\boldsymbol{z}_1)$ after observing $\boldsymbol{z}^{1:t}$. The $r(s_t, a_t)$ based on the *skip-generative path* could empirically force the training on maximizing the mutual information between $\boldsymbol{x}$ and $\boldsymbol{z}^t$ after observing $\boldsymbol{z}^{t-1}$, expressed by

$$\begin{aligned} &\mathbb{E}_{p(\boldsymbol{x})}[r(s_t, a_t)] \quad (25)\\ =&\mathbb{E}_{p(\boldsymbol{x})q(\boldsymbol{z}^{1:t}|\boldsymbol{x})} \log p_\theta(\boldsymbol{x}|\boldsymbol{z}^t) - \mathbb{E}_{p(\boldsymbol{x})} D_{\mathrm{KL}}[q_\phi(\boldsymbol{z}^{1:t}|\boldsymbol{x})||p(\boldsymbol{z}^{1:t})]\\ =&\mathcal{I}_{p,q}(\boldsymbol{x}, \boldsymbol{z}^t) - \mathcal{I}_q(\boldsymbol{x}, \boldsymbol{z}^1) - D_{\mathrm{KL}}[q(\boldsymbol{z}^{1:t})||p(\boldsymbol{z}^{1:t})] - \mathcal{H}_p(\boldsymbol{x}) \end{aligned}$$

Therefore, as shown in Eq. 25, the reward function $r(s_t, a_t)$ is to evaluate how $\mathcal{I}(\boldsymbol{x}, \boldsymbol{z}^t)$ is closer or larger than $\mathcal{I}(\boldsymbol{x}, \boldsymbol{z}^1)$. Thus, the reward function could be an approximate evaluation of $\mathcal{I}_q(\boldsymbol{x}, \boldsymbol{z}^t|\boldsymbol{z}^{t-1})$ after observing the inferred $\boldsymbol{z}^{t-1}$, *i.e.*, $r(s_t, a_t) \simeq \mathcal{I}_q(\boldsymbol{x}, \boldsymbol{z}^t|\boldsymbol{z}^{t-1})$.

Q-value function. To explicitly add regularization on a current inferred latent variable $\boldsymbol{z}^t$'s influence on its following latent variables $\boldsymbol{z}^{>t}$, we propose to maximize its corresponding Q-value, which considers not only its own reward $r(s_t, a_t)$, but also the future accumulative rewards, *i.e.*,

$$\begin{aligned} Q^\pi(s_t, a_t) =& r(s_t, a_t) + \gamma \mathbb{E}_\pi[Q^\pi(s_{t+1}, a_{t+1})]\\ =& r(s_t, a_t) + \mathbb{E}_\pi[\sum_{i=t+1}^{T} \gamma^{i-t} r(s_i, a_i)]. \end{aligned} \quad (26)$$

With the efficient reparameterization trick [43], we could do a single-sample based Monte Carlos (MC) estimation for the expectation of the Q-value function, *i.e.*, $Q^\pi(s_t, a_t) \simeq \sum_{i=t}^{T} \gamma^{i-t} r(s_i, a_i)$. Please note that the reward function is differentiable to the model's parameters, hence we could directly use gradient descent method [42, 69] to maximize the $Q_\pi(s_t, a_t)$, where θ and ϕ are trained simultaneously.

Therefore, at every time step t, maximizing its corresponding Q-value function is approximately maximizing the mutual information between $\boldsymbol{x}$ and $\boldsymbol{z}^t$ and the influence of the inferred $\boldsymbol{z}^t$ in the following latent variables $\boldsymbol{z}^{t+1:T}$ that depends on it, expressed as:

$$Q^\pi(s_t, a_t) \simeq \sum_{i=t}^{T} \gamma^{i-t} \mathcal{I}_q(\boldsymbol{x}, \boldsymbol{z}^i | \boldsymbol{z}^{i-1}) = \mathcal{L}_\mathcal{I}(\boldsymbol{x}, \boldsymbol{z}^t), \quad (27)$$

which is consistent with the expression of Eq. 18 and the analysis in Section 3.1.

Finally, to learn a high-quality sequence $\{\boldsymbol{x}, \boldsymbol{z}^1, ..., \boldsymbol{z}^T\}$ aligning with the optimization direction in Eq. 19, we define the optimization direction on the Q-value function:

$$\nabla_{\theta,\phi} \mathcal{J} \simeq \sum_{t=1}^{T} \mathbb{E}_{s_i \sim P, a_i \sim \pi} \nabla_{\theta,\phi} Q^\pi(s_t, a_t). \quad (28)$$

As a_t could be obtained by reparameterization and $Q^\pi(s_t, a_t)$ is differentiable to model parameters, we could implement the policy gradient by directly maximizing $Q^\pi(s_t, a_t)$ at every time step t by Eq. 9, thereby eliminating the need for a gradient estimator like REINFORCE [89] and avoiding the high variance associated with these estimators [77]. We summarize the overall training scheme in Algorithm 1.

Algorithm 1 Optimizing a hierarchical VAE with RL.

Input: A training dataset $\mathcal{D}_{\boldsymbol{x}}$, training hyperparameters;
Output: An L-layer hierarchical VAE parameterized with an encoder $q_\phi(\mathbf{z}|\boldsymbol{x})$ and a decoder $p_\theta(\boldsymbol{x}|\mathbf{z})$;
Initialization: Randomly initialize the parameters θ, ϕ;
for epoch = 1 to max_epochs **do**
  Sample a batch of N training samples $\boldsymbol{x}^n$ from $\mathcal{D}_{\boldsymbol{x}}$;
  for time step t = 1 to L **do**
    for $i = t$ to L **do**
      Follow Eq. 22, but with reparameterization, to sample an action $a_i^n \sim \pi(a_i^n | s_i^n)$;
      Get reward $r(s_i^n, a_i^n)$ by Eq. 24;
    end for
    Get $Q^\pi(s_t^n, a_t^n) \simeq \sum_{i=t}^{T} \gamma^{i-t} r(s_i^n, a_i^n)$;
    Update θ, ϕ using the policy gradient for the current time step t by Eq. 9;
  end for
end for

4. Experiments

Our experiments engage in comparisons with several hierarchical VAE variants and various unsupervised methods to assess if our method of training hierarchical VAEs with RL yields superior performance in **1)** mitigating *posterior collapse* at higher latent representations, **2)** learning a clear hierarchical representation, **3)** attaining an explainable latent representation, and more importantly, **4)** offering better representation than other hierarchical VAE-based unsupervised learning methods on downstream tasks. Additionally, we present several ablation studies to further evaluate different settings' influence on our method's performance.

4.1. Experimental Setup

Datasets: Our method is benchmarked on 5 datasets: FashionMNIST [91], CIFAR10 and CIFAR100 [44], TinyImageNet [46], and STL-10 [17]. Each method is trained using standard train splits and assessed on corresponding test splits.

Evaluation metrics: For representation evaluation, we follow the commonly adopted *downstream linear classification* [5, 39, 63], using a linear support vector machine (SVM) [37] classifier on representations from frozen models. Classification accuracy is reported to measure linear separability, acting as a proxy for mutual information of representations with the labels. Additionally, we apply the MINE statistics network [4] to estimate the mutual information $\widetilde{\mathcal{I}}(\boldsymbol{x}, \boldsymbol{z}_L)$, between input $\boldsymbol{x}$ and the top latent variable $\boldsymbol{z}_L$.

Baselines: We compare our method with two groups of baselines: **i)** those modifying the inference process without changing the training objective (ELBO); **ii)** those modifying the training objective to add more regularization on the inference process. For **i**, we include a BU hierarchical VAE (**HVAE**) powered with residual connections [35], a TD hierarchical VAE (**LVAE**) with a ladder inference [76], and a more sophisticated bidirectional structure hierarchical VAE (**BIVA**) [57]. For **ii**, to our best knowledge, the most relevant work to our method is informative hierarchical VAE (**IVAE**) [51], which introduces additional layer-wise regularization on latent representations of a hierarchical VAE. Additionally, we include a vanilla VAE (**VAE**) [43] and an adversarial autoencoder (**AAE**) [58] of single-level to assess the performance improvement brought by hierarchical structures.

Implementation details: We apply our RL optimization approach to hierarchical VAEs with BU (**HVAE+RL**) and TD (**LVAE+RL**) inference schemes. For FashionMNIST, hierarchical models use latent dimensions of $\{128, 64, 32, 16, 8\}$, and single-layer models use 8. For other image datasets, hierarchical models' latent dimensions are $\{128\text{conv}, 64\text{conv}, 256, 128, 64\}$, and single-layer models have $\{64\}$. The discount factor, γ, is set to 0.9. Run on one NVIDIA A100 GPU with PyTorch [64], models are optimized using Adam optimizer [42] at a learning rate of 3e-4. Training epochs are set to 1000 for baselines and $1000/L$ for our method, with outcomes averaged over 5 random seeds.

Detailed experimental setup are in Appendix C.

Table 1. Classification acc. and $\widetilde{\mathcal{I}}(\boldsymbol{x}, \boldsymbol{z}_L)$ comparisons. Our methods' results are marked in purple. The best two results are in **bold**.

Method	FashionMNIST		CIFAR10		CIFAR100		TinyImageNet		STL-10 (32 × 32-downsampled)	
	Acc.	$\widetilde{\mathcal{I}}(\boldsymbol{x}, \boldsymbol{z}_L)$	Acc.	$\widetilde{\mathcal{I}}(\boldsymbol{x}, \boldsymbol{z}_L)$	Acc.	$\widetilde{\mathcal{I}}(\boldsymbol{x}, \boldsymbol{z}_L)$	Acc.	$\widetilde{\mathcal{I}}(\boldsymbol{x}, \boldsymbol{z}_L)$	Acc.	$\widetilde{\mathcal{I}}(\boldsymbol{x}, \boldsymbol{z}_L)$
AAE [58]	72.11±0.27	4.08±0.67	38.36±0.20	5.91±1.03	17.56±0.45	6.19±0.95	8.73±0.58	3.37±0.67	36.64±0.17	6.65±1.10
VAE [43]	75.08±0.09	4.36±0.35	39.98±0.05	6.32±0.65	19.05±0.20	**13.76**±1.85	8.92±0.36	3.76±1.21	37.75±0.07	6.87±0.75
BIVA [57]	28.67±0.64	3.48±0.48	10.40±0.13	5e-9±0.00	1.07±0.05	8e-9±0.00	0.67±0.06	9e-8±0.00	10.02±0.20	1.41±0.38
IVAE [51]	76.25±0.30	2.44±0.52	39.26±0.25	6.41±0.69	19.49±0.33	8.91±0.96	8.07±0.69	3.92±0.98	37.98±0.41	2.86±0.95
HVAE [35]	10.73±0.08	0.31±0.08	10.40±0.05	3e-8±0.00	1.21±0.10	7e-7±0.00	0.63±0.06	5e-10±0.00	10.24±0.07	0.97±0.17
HVAE+RL(ours)	**78.23**±0.13	**5.01**±0.70	**46.46**±0.15	**11.39**±1.20	**25.21**±0.21	**17.01**±2.41	**12.76**±0.39	**11.88**±2.87	**43.76**±0.15	**10.93**±1.73
LVAE [76]	19.70±0.21	1.73±0.32	15.32±0.10	5e-5±0.00	10.38±0.89	1.04±0.39	4.77±0.47	0.54±0.12	11.89±0.10	0.45±0.08
LVAE+RL(ours)	**78.48**±0.15	**4.37**±0.58	**46.35**±0.17	**6.91**±0.86	**25.41**±0.20	9.00±1.19	**13.41**±0.44	**4.48**±1.71	**41.86**±0.26	**10.77**±1.94

Table 2. Density estimation in *bits per dimension* (bpd) [21] and KL divergence at the topmost level of hierarchical VAEs.

Model	FashionMNIST		CIFAR10	
	bpd	KL	bpd	KL
VAE	0.306±0.041	17.9±0.3	3.74±0.01	263.0±4.0
BIVA	0.302±0.071	3e-4±0.0	2.85±0.02	2e-3±0.0
IVAE	0.403±0.096	28.9±0.7	3.48±0.05	233.0±7.0
HVAE	0.315±0.067	6e-5±0.0	4.55±0.02	1e-3±0.0
HVAE+RL	0.305±0.089	41.2±0.4	4.36±0.03	265.0±8.0
LVAE	0.313±0.050	9e-4±0.0	2.91±0.03	2e-3±0.0
LVAE+RL	0.335±0.072	32.1±0.5	3.21±0.07	181.0±6.0

4.2. Quantitative Results

Considering the highest-level latent representation $\boldsymbol{z}_L$ should ideally encapsulate the most abstract representation, such as categorical distinctions, we use top-level representations for linear classification. Additionally, given that $\boldsymbol{z}_L$ could be the most prone to *posterior collapse* among all levels, we estimate its information content $\mathcal{I}(\boldsymbol{x}, \boldsymbol{z}_L)$ via MINE. Table 1 reports the classification accuracy on various datasets and estimated mutual information. Applying RL to the hierarchical VAEs (HVAE+RL & LVAE+RL) significantly improves accuracy compared to their counterparts trained on ELBO (HVAE & LVAE), and also outperforms other methods, including IVAE, across all datasets. This demonstrates our approach's superior representation quality. Moreover, $\widetilde{\mathcal{I}}(\boldsymbol{x}, \boldsymbol{z}_L)$ generally correlates classification performance, with our methods enabling higher mutual information in the same backbone models than the baselines in most cases, showing that our method extracts rich information from data. Notably, HVAE+RL yields significantly high $\mathcal{I}(\boldsymbol{x}, \boldsymbol{z}_L)$ on complex datasets like CIFAR100 and TinyImageNet.

In Table 2, we present negative log-likelihood scores in bpd, where a lower bpd indicates better marginal likelihood estimation, along with KL divergences at the highest layer. Intriguingly, we find no direct correlation between the performance in estimating marginal log-likelihood $\log p(\boldsymbol{x})$ using ELBO and the representation learning efficacy, as gauged by accuracy and $\widetilde{\mathcal{I}}(\boldsymbol{x}, \boldsymbol{z}_L)$ in Table 1. For instance, despite BIVA and LVAE's superior marginal likelihood estimation on CIFAR10, their performance in representation learning tasks lags behind IVAE and our methods. This observation lends empirical support to our analysis in Section 3.1 that solely optimizing ELBO does not necessarily enhance hierarchical representation learning. Additionally, we note that a low KL divergence often signals poor downstream performance and reduced $\widetilde{\mathcal{I}}(\boldsymbol{x}, \boldsymbol{z}_L)$, indicative of *posterior collapse* as defined in Definition 1.

4.3. Qualitative Results

To elucidate the efficacy of our method in learning informative hierarchical representations, we first examine the hierarchies learned by different approaches in Fig. 1 by projecting latent representations of different levels to the input space via the *skip-generative path*. Fig. 1 shows training with RL effectively mitigates *posterior collapse*, yielding a more informative latent hierarchy. This enhancement in hierarchical representation is further evident in Fig. 2, reinforcing the conclusions drawn in Section 3.1.

An interesting property of hierarchical VAEs is their explainable latent space, which we probe using t-SNE and latent traversal at the topmost level for generation, as shown in Fig. 5. The results demonstrate notable improvements achieved by applying RL on mapping inputs to an explainable semantic space, to which the improvement of performance in linear classification (Table 1) can be attributed. Furthermore, our exploration into latent traversal sheds light on the explainability of the latent space, with individual latent dimensions corresponding to specific abstract features like widths and positions of hollow parts, thus underscoring the interpretability of our learned representations.

4.4. Ablation Study

Total number of levels. We investigate the impact of varying L in hierarchical VAEs on downstream task performance using representations from the topmost latent variable in Fig. 6. We observe that all baselines suffer from classification performance degeneracy with increasing latent levels: *posterior collapses* largely impair downstream task performance. In comparison, our methods are much more stable and achieve superior performance across all level number settings, indicating its robustness. More importantly, these results suggest our approach enables the possibility of learning a higher hierarchy of latent representation.

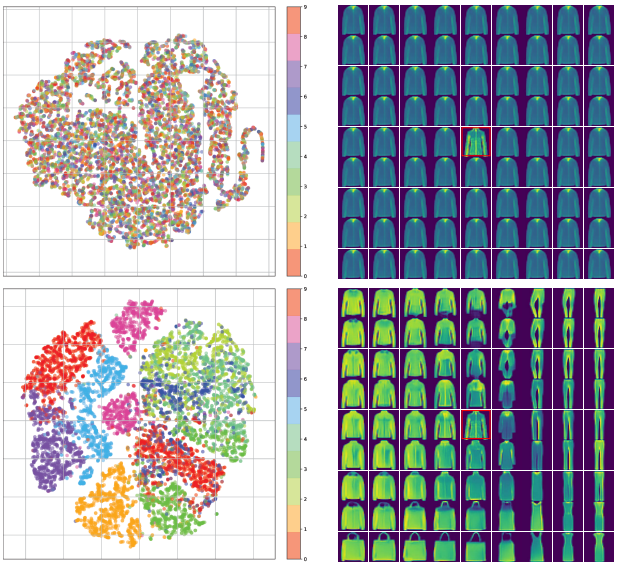

Figure 5. Panels showing the t-SNE and latent traversal generation on the highest latent variable in FashionMNIST, for vanilla HVAE (**top**) and HVAE+RL(**bottom**). The anchor point (original input) for latent traversal is bounded in red in grid center. Traversals are spaced by inverse Gaussian cdf to align with the prior distribution.

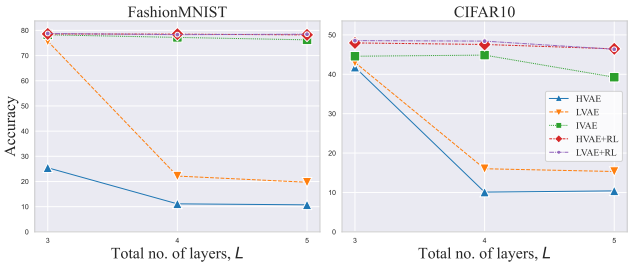

Figure 6. Classification acc. of hierarchical VAEs of different levels.

Table 3. Classification acc. of different representation source.

Objective	Repre. source	FashionMNIST		CIFAR10	
		HVAE	LVAE	HVAE	LVAE
ELBO	z_L only	10.73±0.08	19.70±0.21	10.40±0.05	15.32±0.10
	full $\mathbf{z}$	79.96±0.23	80.25±0.27	50.09±0.67	49.87±0.39
RL	z_L only	78.23±0.13	78.48±0.15	46.46±0.15	46.35±0.17
	full $\mathbf{z}$	84.56±0.25	84.77±0.30	52.87±0.54	52.89±0.48

Latent representation source. The substantial cumulative dimension of all hierarchical latent variables $\mathbf{z}$, while potentially having greater representation capacity, may not be practical for all tasks. To explore this, we compare in Table 3 the downstream task performance of representations sourcing from the full $\mathbf{z}$ versus the most abstract topmost latent representation $\boldsymbol{z}_L$ to assess if $\boldsymbol{z}_L$ can maintain performance comparable to full $\mathbf{z}$. We observe that with RL, models effectively retain comparable classification performance using just $\boldsymbol{z}_L$, showing efficient learning of key abstract information at the topmost layer.

Discount factor γ. Exploring the impact of focusing solely on the current step's reward, we set the discount factor γ to 0 in our method. Besides, we ablate two typical settings of γ, 0.9 and 0.98, in the Q-value function to assess robustness. As in Table 4, models using $Q(s_t, a_t)$ for step-wise optimization outperform those relying only on $r(s_t, a_t)$, highlighting the benefits in downstream tasks of considering future accumulative rewards, *i.e.*, regularizing dependencies between different levels' latent representations. Moreover, the stable performance across γ confirms our approach's robustness.

Table 4. Classification acc. of RL using different discount factors γ, where γ = 0 means we only maximize $r(s_t, a_t)$ at time step t.

Objective	γ	FashionMNIST		CIFAR10	
		HVAE	LVAE	HVAE	LVAE
$r(s_t, a_t)$	0	75.63±0.11	75.52±0.10	41.91±0.15	42.16±0.13
$Q(s_t, a_t)$	0.8	78.03±0.16	77.98±0.23	46.45±0.30	46.42±0.27
$Q(s_t, a_t)$	0.85	78.11±0.24	78.34±0.19	45.89±0.24	45.97±0.31
$Q(s_t, a_t)$	0.9	78.23±0.13	78.48±0.15	46.46±0.15	46.35±0.17
$Q(s_t, a_t)$	0.95	77.95±0.19	78.53±0.16	46.21±0.21	46.25±0.25
$Q(s_t, a_t)$	0.98	78.01±0.10	78.28±0.17	46.31±0.20	46.37±0.19

5. Limitation and Conclusion

Limitation and Future work. Though the (hierarchical) VAEs have advantages in modeling flexibility and sampling speed compared to other deep generative models [92] and are the most suitable models to demonstrate our idea with a theoretical interpretation, hierarchical VAE-based methods' application on downstream representation tasks could be somewhat outdated recently. Nevertheless, it's noteworthy that our idea of training hierarchical models with RL is not exclusively limited to hierarchical VAEs. Instead, it holds the potential when extended to general hierarchical models like diffusion models [40, 55] , which feature more levels of latent representation and employ a fixed, non-trainable encoder for the efficient loss computation as independent terms over the hierarchy [61]. Our approach may be extended to learn a diffusion model with a trainable encoder, which may potentially enhance their representation capacity.

Conclusion. In this paper, we investigate the unsupervised representation learning with hierarchical VAEs from the perspective of information theory. We identify an urgent need to apply reinforcement learning to learn a high-quality hierarchy. The learned hierarchical representations have shown their superiority in improving the downstream task performance and mitigating the notorious *posterior collapse*. Additionally, an interesting finding is that learning a better ELBO (original training objective of hierarchical VAEs) is not necessary for learning a better representation, which has been supported by both our theoretical analysis and empirical results. Beyond hierarchical VAEs, our method may inspire future works to apply reinforcement learning to learn a hierarchical representation in time ahead.

Acknowledgment. This research is supported by the National Research Foundation, Singapore, under its Competitive Research Programme (Grant No. NRF-CRP23-2019-0006).

References

[1] Korbinian Abstreiter, Sarthak Mittal, Stefan Bauer, Bernhard Schölkopf, and Arash Mehrjou. Diffusion-based representation learning. *arXiv preprint arXiv:2105.14257*, 2021.

[2] Alexander A. Alemi, Ian Fischer, Joshua V. Dillon, and Kevin Murphy. Deep variational information bottleneck. In *International Conference on Learning Representations*, 2017.

[3] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. In *NeurIPS*, 2019.

[4] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeshwar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and Devon Hjelm. Mutual information neural estimation. In *ICML*, 2018.

[5] Yoshua Bengio, Aaron C. Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(8):1798–1828, 2013.

[6] Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. *arXiv preprint arXiv:2305.13301*, 2023.

[7] Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew M. Dai, Rafal Józefowicz, and Samy Bengio. Generating sentences from a continuous space. In *CoNLL*, 2016.

[8] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *NeurIPS*, 2020.

[9] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021.

[10] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

[11] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E. Hinton. Big self-supervised models are strong semi-supervised learners. In *NeurIPS*, 2020.

[12] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *CVPR*, 2021.

[13] Xi Chen, Diederik P. Kingma, Tim Salimans, Yan Duan, Prafulla Dhariwal, John Schulman, Ilya Sutskever, and Pieter Abbeel. Variational lossy autoencoder. In *ICLR*, 2017.

[14] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *ICCV*, 2021.

[15] Zihan Chen, Hongyuan Zhu, Hao Cheng, Siya Mi, Yu Zhang, and Xin Geng. LPCL: localized prominence contrastive learning for self-supervised dense visual pre-training. *Pattern Recognit.*, 135:109185, 2023.

[16] Yuanzheng Ci, Chen Lin, Lei Bai, and Wanli Ouyang. Fast-MoCo: Boost momentum-based contrastive learning with combinatorial patches. In *ECCV*, 2022.

[17] Adam Coates, Andrew Y. Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *AISTATS*, 2011.

[18] Thomas M Cover. *Elements of Information Theory*. John Wiley & Sons, 1999.

[19] Bin Dai, Ziyu Wang, and David P. Wipf. The usual suspects? Reassessing blame for VAE posterior collapse. In *ICML*, 2020.

[20] Adji B. Dieng, Yoon Kim, Alexander M. Rush, and David M. Blei. Avoiding latent variable collapse with generative skip models. In *AISTATS*, 2019.

[21] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real NVP. In *ICLR*, 2017.

[22] Carl Doersch, Abhinav Gupta, and Alexei A. Efros. Unsupervised visual representation learning by context prediction. In *ICCV*, 2015.

[23] Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. In *ICLR*, 2017.

[24] Yann Dubois, Stefano Ermon, Tatsunori B. Hashimoto, and Percy Liang. Improving self-supervised learning by characterizing idealized representations. In *NeurIPS*, 2022.

[25] Fabian Duffhauss, Ngo Anh Vien, Hanna Ziesche, and Gerhard Neumann. FusionVAE: A deep hierarchical variational autoencoder for RGB image fusion. In *ECCV*, 2022.

[26] Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Alex Lamb, Martin Arjovsky, Olivier Mastropietro, and Aaron Courville. Adversarially learned inference. In *ICLR*, 2017.

[27] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. In *ICCV*, 2021.

[28] Ying Fan, Olivia Watkins, Yuqing Du, Hao Liu, Moonkyung Ryu, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, Kangwook Lee, and Kimin Lee. Reinforcement learning for fine-tuning text-to-image diffusion models. In *NeurIPS*, 2023.

[29] Yuting Gao, Jia-Xin Zhuang, Shaohui Lin, Hao Cheng, Xing Sun, Ke Li, and Chunhua Shen. DisCo: Remedy self-supervised learning on lightweight models with distilled contrastive learning. In *ECCV*, 2022.

[30] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *ICLR*, 2018.

[31] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[32] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent - A new approach to self-supervised learning. In *NeurIPS*, 2020.

[33] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *ICML*, 2018.

[34] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, 2006.

[35] Jakob D Drachmann Havtorn, Jes Frellsen, Soren Hauberg, and Lars Maaløe. Hierarchical VAEs know what they don't know. In *ICML*, 2021.

[36] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.

[37] M.A. Hearst, S.T. Dumais, E. Osuna, J. Platt, and B. Scholkopf. Support vector machines. *IEEE Intelligent Systems and Their Applications*, 13(4):18–28, 1998.

[38] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-VAE: Learning basic visual concepts with a constrained variational framework. In *ICLR*, 2016.

[39] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. In *ICLR*, 2019.

[40] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

[41] Yoon Kim, Sam Wiseman, Andrew C. Miller, David A. Sontag, and Alexander M. Rush. Semi-amortized variational autoencoders. In *ICML*, 2018.

[42] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[43] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014.

[44] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Canadian Institute for Advanced Research, 2009.

[45] Mingi Kwon, Jaeseok Jeong, and Youngjung Uh. Diffusion models already have a semantic latent space. In *ICLR*, 2023.

[46] Ya Le and Xuan S. Yang. Tiny ImageNet visual recognition challenge. Stanford University, 2015.

[47] Phuc H. Le-Khac, Graham Healy, and Alan F. Smeaton. Contrastive representation learning: A framework and review. *IEEE Access*, 8:193907–193934, 2020.

[48] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proc. IEEE*, 86(11):2278–2324, 1998.

[49] Kibok Lee, Yian Zhu, Kihyuk Sohn, Chun-Liang Li, Jinwoo Shin, and Honglak Lee. i-mix: A domain-agnostic strategy for contrastive representation learning. In *ICLR*, 2021.

[50] Yewen Li, Chaojie Wang, Zhibin Duan, Dongsheng Wang, Bo Chen, Bo An, and Mingyuan Zhou. Alleviating "posterior collapse" in deep topic models via policy gradient. In *NeurIPS*, 2022.

[51] Yewen Li, Chaojie Wang, Xiaobo Xia, Tongliang Liu, xin miao, and Bo An. Out-of-distribution detection with an adaptive likelihood ratio on informative hierarchical VAE. In *NeurIPS*, 2022.

[52] Timothy P. Lillicrap, Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. In *ICLR*, 2016.

[53] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, 2015.

[54] Ali Lotfi-Rezaabad and Sriram Vishwanath. Learning representations by maximizing mutual information in variational autoencoders. In *ISIT*, 2020.

[55] Calvin Luo. Understanding diffusion models: A unified perspective. *arXiv preprint arXiv:2208.11970*, 2022.

[56] Xuezhe Ma, Xiang Kong, Shanghang Zhang, and Eduard H Hovy. Decoupling global and local representations via invertible generative flows. In *ICLR*, 2021.

[57] Lars Maaløe, Marco Fraccaro, Valentin Liévin, and Ole Winther. BIVA: A very deep hierarchy of latent variables for generative modeling. In *NeurIPS*, 2019.

[58] Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, Ian Goodfellow, and Brendan Frey. Adversarial autoencoders. *arXiv preprint arXiv:1511.05644*, 2015.

[59] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *CVPR*, 2020.

[60] Sarthak Mittal, Korbinian Abstreiter, Stefan Bauer, Bernhard Schölkopf, and Arash Mehrjou. Diffusion based representation learning. In *ICML*, 2023.

[61] Beatrix Miranda Ginn Nielsen, Anders Christensen, Andrea Dittadi, and Ole Winther. DiffEnc: Variational diffusion with a learned encoder. In *ICLR*, 2024.

[62] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*, 2016.

[63] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[64] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019.

[65] Deepak Pathak, Philipp Krähenbühl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016.

[66] Konpat Preechakul, Nattanat Chatthee, Suttisak Wizadwongsa, and Supasorn Suwajanakorn. Diffusion autoencoders: Toward a meaningful and decodable representation. In *CVPR*, 2022.

[67] Ali Razavi, Aäron van den Oord, Ben Poole, and Oriol Vinyals. Preventing posterior collapse with delta-VAEs. In *ICLR*, 2019.

[68] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *ICML*, 2014.

[69] Herbert Robbins and Sutton Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, pages 400–407, 1951.

[70] Mihaela Rosca, Balaji Lakshminarayanan, and Shakir Mohamed. Distribution matching in variational inference. *arXiv preprint arXiv:1802.06847*, 2018.

[71] Tim Salimans, Andrej Karpathy, Xi Chen, and Diederik P. Kingma. PixelCNN++: Improving the PixelCNN with discretized logistic mixture likelihood and other modifications. In *ICLR*, 2017.

[72] John Schulman, Sergey Levine, Pieter Abbeel, Michael I. Jordan, and Philipp Moritz. Trust region policy optimization. In *ICML*, 2015.

[73] Huajie Shao, Shuochao Yao, Dachun Sun, Aston Zhang, Shengzhong Liu, Dongxin Liu, Jun Wang, and Tarek F. Abdelzaher. ControlVAE: Controllable variational autoencoder. In *ICML*, 2020.

[74] Huajie Shao, Yifei Yang, Haohong Lin, Longzhong Lin, Yizhuo Chen, Qinmin Yang, and Han Zhao. Rethinking controllable variational autoencoders. In *CVPR*, 2022.

[75] Abhishek Sinha, Jiaming Song, Chenlin Meng, and Stefano Ermon. D2C: Diffusion-decoding models for few-shot conditional generation. In *NeurIPS*, 2021.

[76] Casper Kaae Sønderby, Tapani Raiko, Lars Maaløe, Søren Kaae Sønderby, and Ole Winther. Ladder variational autoencoders. In *NeurIPS*, 2016.

[77] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[78] Richard S. Sutton, David A. McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *NeurIPS*, 1999.

[79] Michael Tschannen, Olivier Bachem, and Mario Lucic. Recent advances in autoencoder-based representation learning. In *NeurIPS Workshop on Bayesian Deep Learning*, 2018.

[80] Arash Vahdat and Jan Kautz. NVAE: A deep hierarchical variational autoencoder. In *NeurIPS*, 2020.

[81] Aäron van den Oord, Nal Kalchbrenner, Lasse Espeholt, Koray Kavukcuoglu, Oriol Vinyals, and Alex Graves. Conditional image generation with PixelCNN decoders. In *NeurIPS*, 2016.

[82] Aäron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *NeurIPS*, 2017.

[83] Benquan Wang, Yewen Li, Eng Aik Chan, Giorgio Adamo, Bo An, Zexiang Shen, and Nikolay I Zheludev. Optical localization of nanoparticles in sub-rayleigh clusters. In *CLEO/Europe-EQEC*, 2023.

[84] Benquan Wang, Ruyi An, Eng Aik Chan, Giorgio Adamo, Jin-Kyu So, Yewen Li, Zexiang Shen, Bo An, and Nikolay I Zheludev. Retrieving positions of closely packed sub-wavelength nanoparticles from their diffraction patterns. *Applied Physics Letters*, 2024.

[85] Chaojie Wang, Hao Zhang, Bo Chen, Dongsheng Wang, Zhengjue Wang, and Mingyuan Zhou. Deep relational topic modeling via graph poisson gamma belief network. In *NeurIPS*, 2020.

[86] Chaojie Wang, Bo Chen, Zhibin Duan, Wenchao Chen, Hao Zhang, and Mingyuan Zhou. Generative text convolutional neural network for hierarchical document representation learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(4):4586–4604, 2023.

[87] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *ICML*, 2020.

[88] Yixin Wang, David M. Blei, and John P. Cunningham. Posterior collapse and latent variable non-identifiability. In *NeurIPS*, 2021.

[89] Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.*, 8:229–256, 1992.

[90] Weilai Xiang, Hongyu Yang, Di Huang, and Yunhong Wang. Denoising diffusion autoencoders are unified self-supervised learners. In *ICCV*, 2023.

[91] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

[92] Zhisheng Xiao, Karsten Kreis, and Arash Vahdat. Tackling the generative learning trilemma with denoising diffusion GANs. In *ICLR*, 2022.

[93] Xingyi Yang and Xinchao Wang. Diffusion model as representation learner. In *ICCV*, 2023.

[94] Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. SeqGAN: Sequence generative adversarial nets with policy gradient. In *AAAI*, 2017.

[95] Richard Zhang, Phillip Isola, and Alexei A. Efros. Colorful image colorization. In *ECCV*, 2016.

[96] Zijian Zhang, Zhou Zhao, and Zhijie Lin. Unsupervised representation learning from pre-trained diffusion probabilistic models. In *NeurIPS*, 2022.

[97] Mingkai Zheng, Shan You, Fei Wang, Chen Qian, Changshui Zhang, Xiaogang Wang, and Chang Xu. Ressl: Relational self-supervised learning with weak augmentation. In *NeurIPS*, 2021.

[98] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. Image BERT pre-training with online tokenizer. In *ICLR*, 2022.