Learning Hierarchical Features with Joint Latent Space Energy-Based Prior

Jiali Cui[1] Ying Nian Wu[2] Tian Han[1]

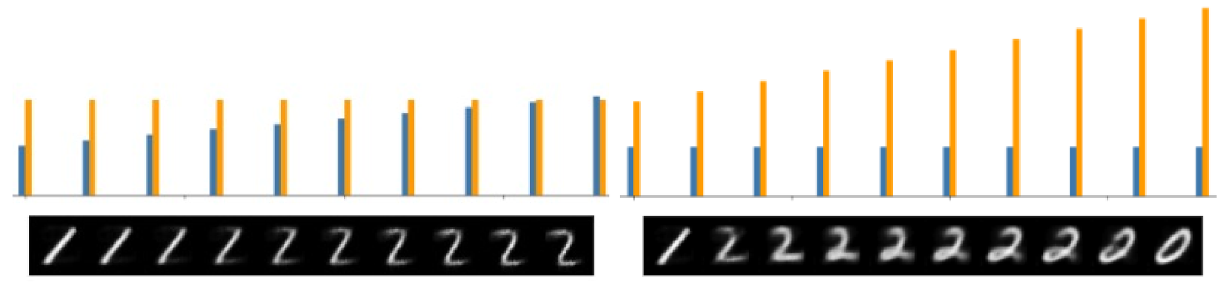
[1]Stevens Institute of Technology [2]University of California, Los Angeles

Problem I

Latent Space Energy-Based Model (LEBM). The LEBM can be defined with probability density as

$$p_\alpha(\mathbf{z}) = \frac{1}{\mathbb{Z}(\alpha)} \exp\left[f_\alpha(\mathbf{z})\right] p_0(\mathbf{z})$$

Limitation. Such a prior model is expressive in modeling the *intra-layer* relations among latent units. However, it mainly focuses on a *single-layer* latent space, which can make it challenging to capture data representations at different levels.



Visualization for LEBM by changing each unit of 2-dimensional $\mathbf{z}$. **Left**: changing the first unit. **Right**: changing the second unit. **Top:** the value of each unit, where the orange color indicates the first unit and the blue color indicates the second unit.

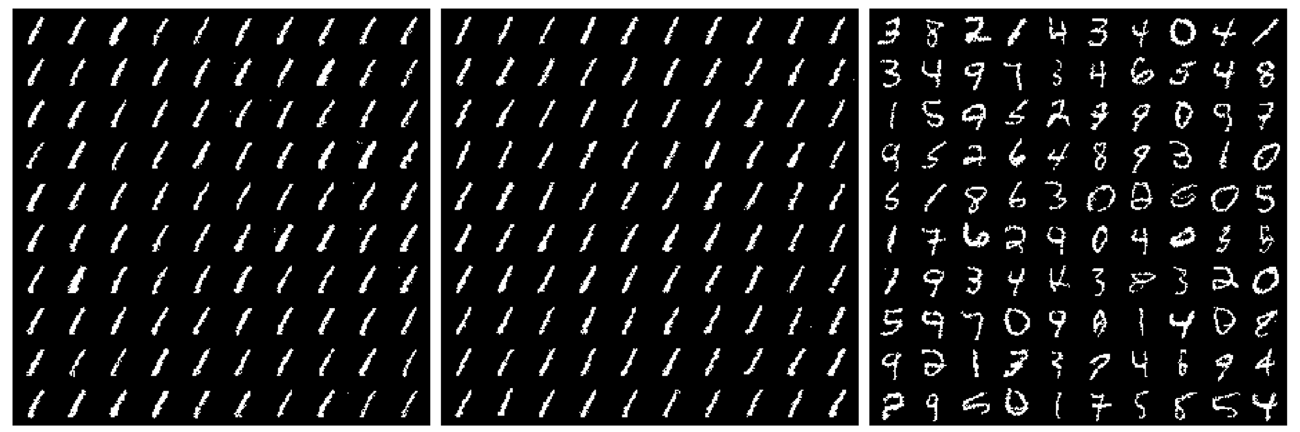
Problem II

Conditional Hierarchical Generator Model. The conditional hierarchical generator models consist of multi-layer latent variables that are organized in a top-down hierarchical structure and modelled to be conditionally dependent on its upper layer, i.e.,

$$p_\theta(\mathbf{z}) = \prod_{i=1}^{L-1} p_{\theta_i}(\mathbf{z}_i|\mathbf{z}_{i+1}) p_0(\mathbf{z}_L)$$

where $p_{\theta_i}(\mathbf{z}_i|\mathbf{z}_{i+1}) \sim \mathcal{N}(\mu_{\theta_i}(\mathbf{z}_{i+1}), \sigma_{\theta_i}(\mathbf{z}_{i+1}))$ and $p_0(\mathbf{z}_L) \sim \mathcal{N}(0, I)$.

Limitation. Such multi-layer latent variables are typically parameterized to be Gaussian, which primarily focuses on modelling the *inter-layer* relation for latent variables while the *intra-layer* relation is largely ignored. This can be less informative in capturing complex abstractions, resulting in limited success in hierarchical representation learning.



Hierarchical sampling on BIVA via the repamaramization trick, i.e., $\mathbf{z}_i = \mu_{\theta_i}(\mathbf{z}_{i+1}) + \sigma_{\theta_i}(\mathbf{z}_{i+1}) \cdot \epsilon_i$. **Left:** sampling ϵ_1, ϵ_2 for bottom layers. **Middle:** sampling ϵ_3, ϵ_4 for middle layers. **Right:** sampling ϵ_5, ϵ_6 for top layers.

Proposed Method

Joint Latent Space EBM Prior Model. We propose a joint latent space EBM prior for multi-layer latent variables, which can capture hierarchical representations by jointly modelling the latent variables of all layers and is also expressive in modelling the *intra-layer* relation among latent units at each layer.

$$p_\alpha(\mathbf{z}) = \frac{1}{\mathbb{Z}(\alpha)} \exp[f_\alpha([\mathbf{z}_1, \ldots, \mathbf{z}_L])] p_0([\mathbf{z}_1, \ldots, \mathbf{z}_L])$$

where latent variables are partitioned into multiple groups and concatenated, i.e., $\mathbf{z} = [\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_L]$.

Architectural Hierarchical Generation Model. The generation model is formulated as

$$p_\beta(\mathbf{x}|[\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_L]) \sim \mathcal{N}(g_\beta([\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_L]), \sigma^2 I_D)$$

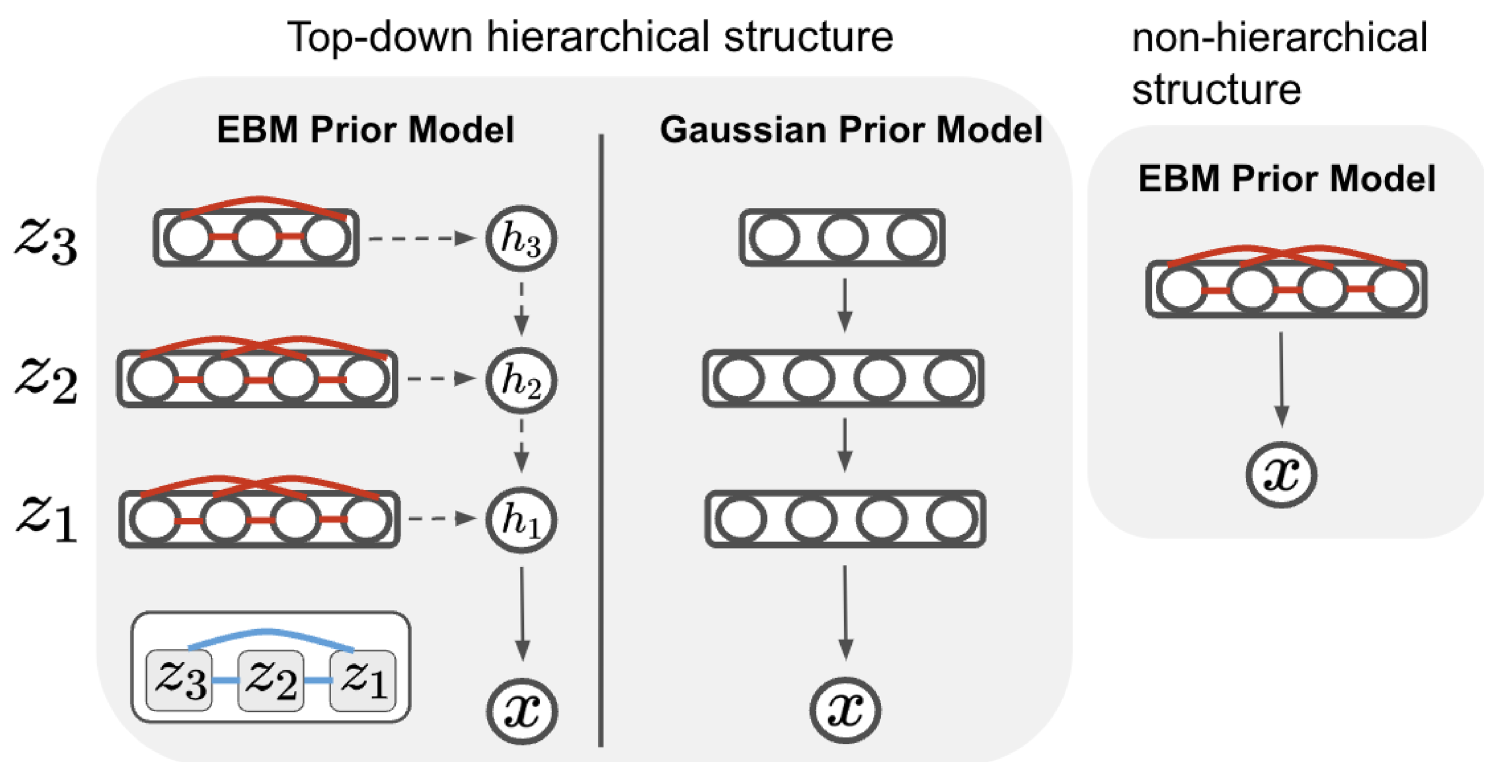
To facilitate the hierarchical representation learning with multi-layer latent variables, we consider multi-layer hierarchical generator network g_β ($= \{g_1, g_2, \ldots, g_L\}$) that is designed to explain the observation $\mathbf{x}$ by integrating data representation from the above layers, i.e.,

$$h_L = g_L(\mathbf{z}_L), \quad h_i = g_i([\mathbf{z}_i, h_{i+1}]), \quad i = 1, 2, \ldots, L-1$$
$$\mathbf{x} \sim \mathcal{N}(h_1, \sigma^2 I_D)$$

in which $\mathbf{z}_L$ is at the top layer, and g_i is a shallow network that decodes latent code $\mathbf{z}_i$ while integrating features from the upper layer.

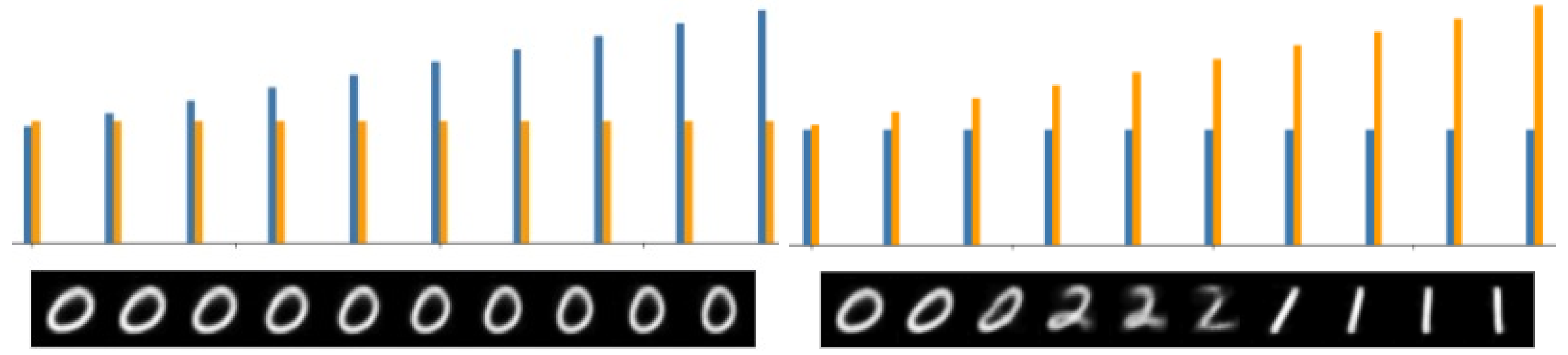
Illustration

Comparison to Gaussian Prior and LEBM.



The illustration of the proposed joint EBM prior model (Left). **Red lines** indicates the modelling of *intra-layer* relation, and **blue lines** indicate *inter-layer* relation. Our joint EBM prior model is capable of modelling the *intra-layer* and *inter-layer* relation of latent variables from all layers, which thus benefits effective hierarchical representation learning.

Experiment: Hierarchical Representation Learning



Visualization for our model by changing each unit of 2-dimensional z. Left: changing the first unit. **Right**: changing the second unit. **Top:** the value of each unit, where the orange color indicates the first unit and the blue color indicates the second unit.

Hierarchical sampling on MNIST. Left: The latent code at bottom layer (z_1) indicates the stroke width. **Center:** the latent code at second layer (z_2) encodes geometric changes among similar digits. **Right:** the latent code at top layer (z_3) learns the digit identity and general structure.

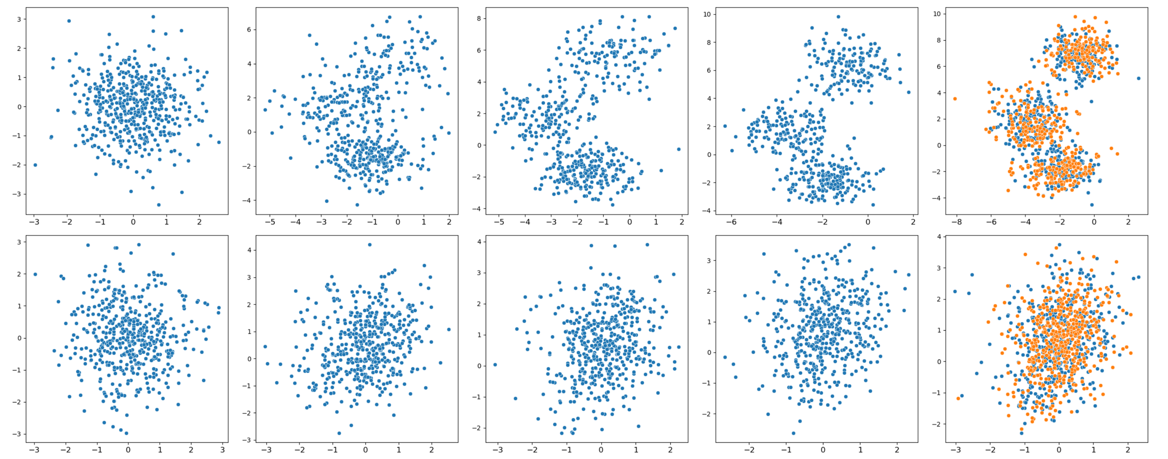
Hierarchical sampling on SVHN.
Left: The latent code at bottom layer ($\mathbf{z}_1$) represents the background light and shading. **Center-left:** the latent code at second bottom layer ($\mathbf{z}_2$) represents the color schemes. **Center-right:** the latent code at second top layer ($\mathbf{z}_3$) encodes the shape variations of the same digit. **Right:** the latent code at top layer ($\mathbf{z}_4$) captures the digit identity and the general structure.

Experiment: Image Modelling

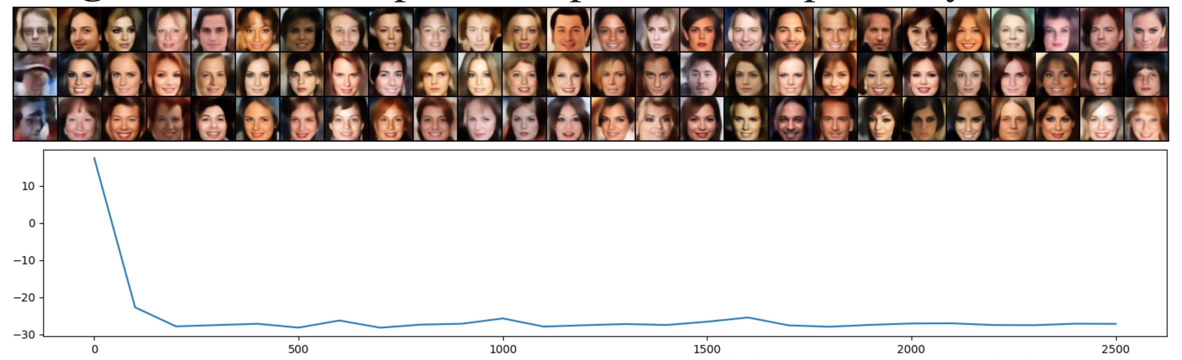
Model	SVHN MSE (↓)	SVHN FID (↓)	CelebA-64 MSE (↓)	CelebA-64 FID (↓)
ABP	-	49.71	-	51.50
LVAE	0.014	39.26	0.028	53.40
BIVA	0.010	31.65	0.010	33.58
SRI	0.011	35.23	0.011	36.84
VLAE	0.016	43.95	0.010	44.05
2s-VAE	0.019	42.81	0.021	44.40
RAE	0.014	40.02	0.018	40.95
NCP-VAE	0.020	33.23	0.021	42.07
Multi-NCP	**0.004**	26.19	0.009	35.38
LEBM	0.008	29.44	0.013	37.87
Ours	0.008	**24.16**	**0.004**	**32.15**

Testing reconstruction by MSE, and generation evaluation by FID on SVHN and CelebA-64.

Experiment: Analysis of Latent Space



Visualization of the latent codes sampled from our EBM prior (Top row: $\mathbf{z}_2$). **Blue, Orange** color indicate prior and posterior, respectively.



Transition of Markov chains initialized from $p_0(\mathbf{z})$ towards $p_\alpha(\mathbf{z})$ for 2500 steps. Top: Trajectory in the CelebA-64 data space for every 100 steps. Bottom: Energy profile over time.