Recent Advances of Variational Auto-Encoders
(focused on poster collapse problem)

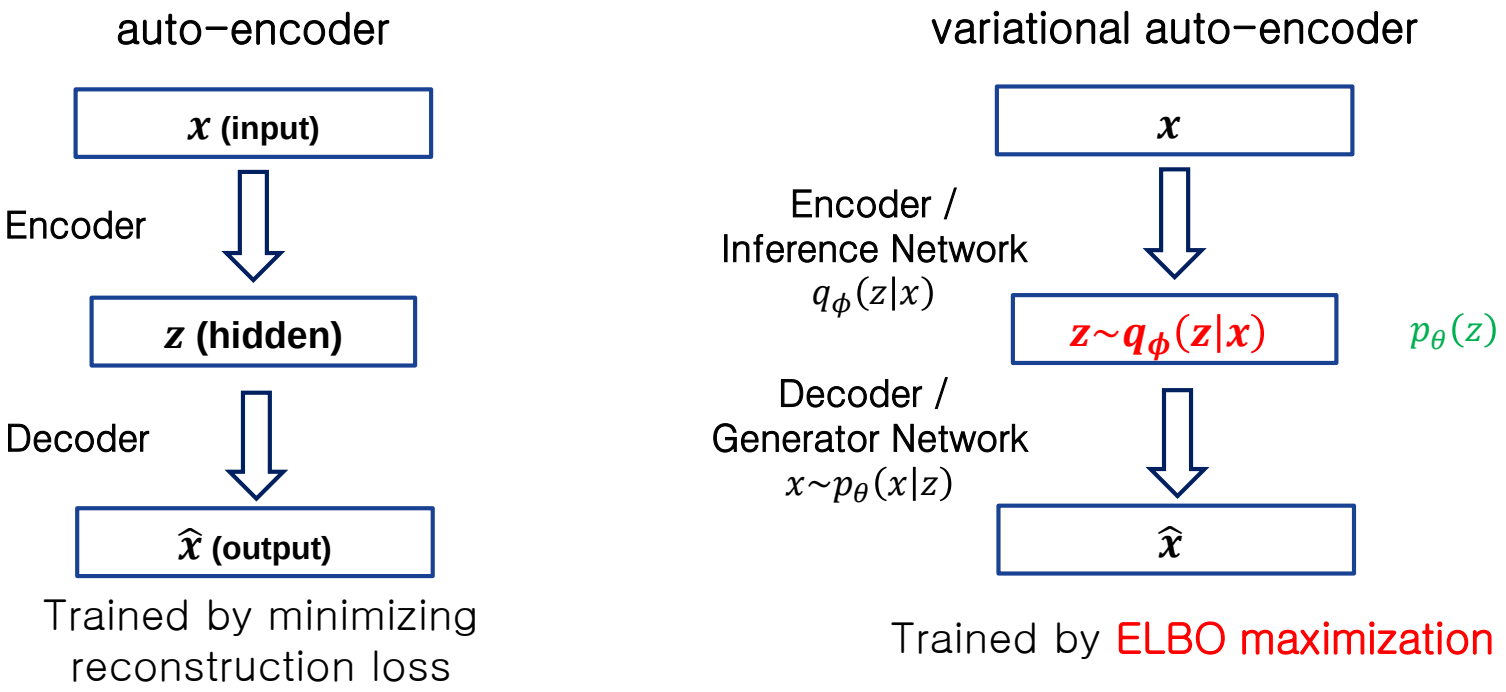
Injung Kim
Handong Global University

Agenda

- Introduction to VAE
- Posterior Collapse Problem
- Preventing Posterior Collapse
- Q&A

Variational Auto-Encoder (VAE)

- An extension of auto-encoder that learns distribution of data using latent variable and generates diverging samples



Properties of VAE

- Likelihood-based generative model

$$\theta^* = \text{argmax}_\theta \, P_\theta(x) \approx \boldsymbol{argmax_\theta \, [ELBO]}$$

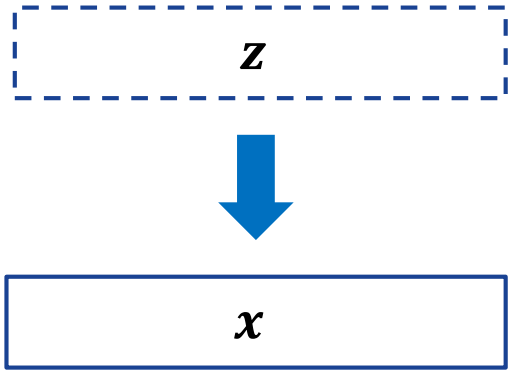
- Pros
  - Explicitly provides $P_\theta(x)$
  - High diversity
    - In principle, captures all modes of the data not suffering from mode collapse
  - Can learn disentangled latent representation
  - Compared with GAN, less suffers from training instability

- Cons
  - Posterior collapse problem
  - No intrinsic incentive to focus on global structure
  - Output quality is not as good as GAN, but rapidly improving

Latent Variable Model

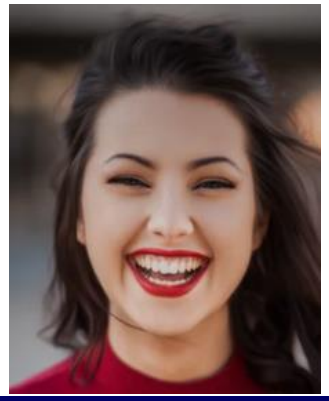
- Latent variable model aims to explain observed variables in terms of hidden latent variables
  - Learn inferencing hidden attribute, underlying causal factors, or source of variation of observed data in an unsupervised manner

Latent variable
(hidden factor)



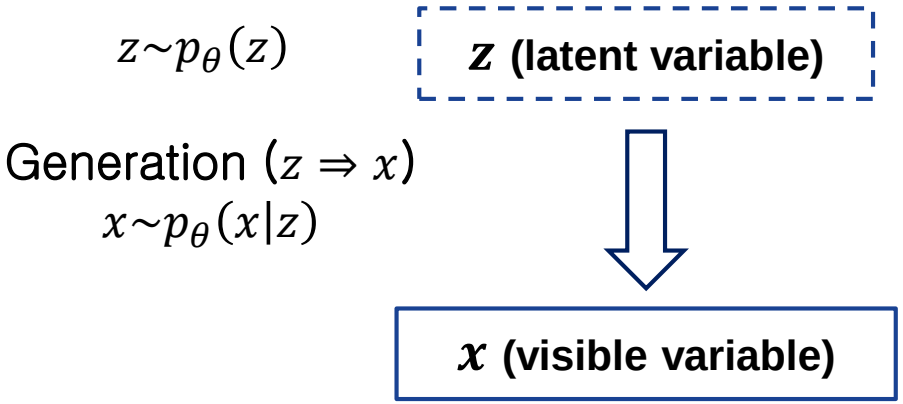
Visible variable
(observed data)

identity, gender, pose,
hairstyle, expression, etc.



VAE as a Latent Variable Model

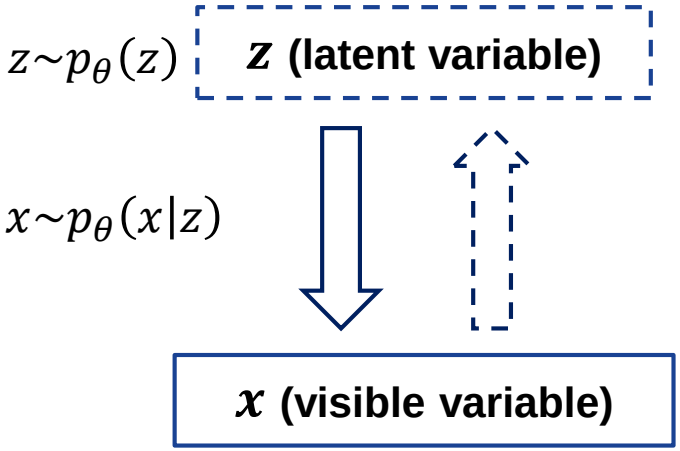
- Sample generation using a latent variable $\boldsymbol{z}$
  - Hopefully, each element of $\boldsymbol{z}$ represents an attribute of $\boldsymbol{x}$
  1. Sample z from prior distribution $P(z)$
     - Assumed a tractable distribution (e.g., $N(0, I)$)
  2. Produce x from z using likelihood $P(x|z)$
     - Modeled by a parametric function (e.g., a neural network)



$\boldsymbol{p_\theta(z|x)}$ and $\boldsymbol{p_\theta(x)}$ are intractable!

Variational Learning

- Inference
  - Approximate $p_\theta(z|x)$ by a variational distribution $q_\phi(z|x)$
- Learning by MLE
  - Maximize ELBO instead of $\log p_\theta(x)$
    - ELBO (Evidence Lower BOund) = variational lower bound of $\log p_\theta(x)$



Inference ($x \Rightarrow z$)

$z \sim p_\theta(z|x) \approx q_\phi(z|x)$

Learning

$argmax_\theta \log p_\theta(x) \Rightarrow argmax_{\theta,\phi} ELBO$

Variational Lower Bound

- Variational learning

$$p_\theta(z|x^{(i)}) = \frac{p_\theta(x^{(i)}|z)p_\theta(z)}{p_\theta(x^{(i)})}$$

$$p_\theta(x^{(i)}) = \frac{p_\theta(x^{(i)}|z)p_\theta(z)}{p_\theta(z|x^{(i)})}$$

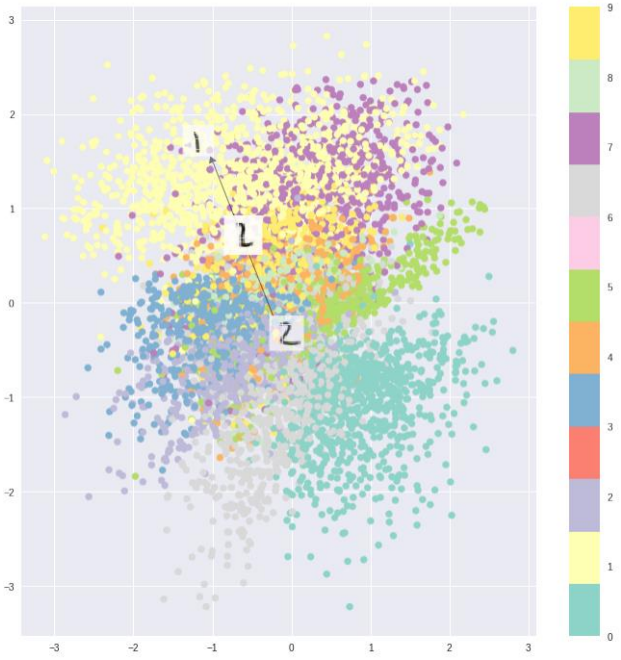
$$\begin{aligned}
\log p_\theta(x^{(i)}) &= \mathbf{E}_{z\sim q_\phi(z|x^{(i)})}\left[\log p_\theta(x^{(i)})\right] && (p_\theta(x^{(i)}) \text{ Does not depend on } z) \\
&= \mathbf{E}_z\left[\log \frac{p_\theta(x^{(i)} \mid z)p_\theta(z)}{p_\theta(z \mid x^{(i)})}\right] && \text{(Bayes' Rule)} \\
&= \mathbf{E}_z\left[\log \frac{p_\theta(x^{(i)} \mid z)p_\theta(z)}{p_\theta(z \mid x^{(i)})}\frac{q_\phi(z \mid x^{(i)})}{q_\phi(z \mid x^{(i)})}\right] && \text{(Multiply by constant)} \\
&= \mathbf{E}_z\left[\log p_\theta(x^{(i)} \mid z)\right] - \mathbf{E}_z\left[\log \frac{q_\phi(z \mid x^{(i)})}{p_\theta(z)}\right] + \mathbf{E}_z\left[\log \frac{q_\phi(z \mid x^{(i)})}{p_\theta(z \mid x^{(i)})}\right] && \text{(Logarithms)} \\
&= \underbrace{\mathbf{E}_z\left[\log p_\theta(x^{(i)} \mid z)\right] - D_{KL}(q_\phi(z \mid x^{(i)}) \,||\, p_\theta(z))}_{ELBO \;\; \mathcal{L}(x^{(i)}, \theta, \phi) \le \log p_\theta(x)} + \underbrace{D_{KL}(q_\phi(z \mid x^{(i)}) \,||\, p_\theta(z \mid x^{(i)}))}_{\ge 0}
\end{aligned}$$

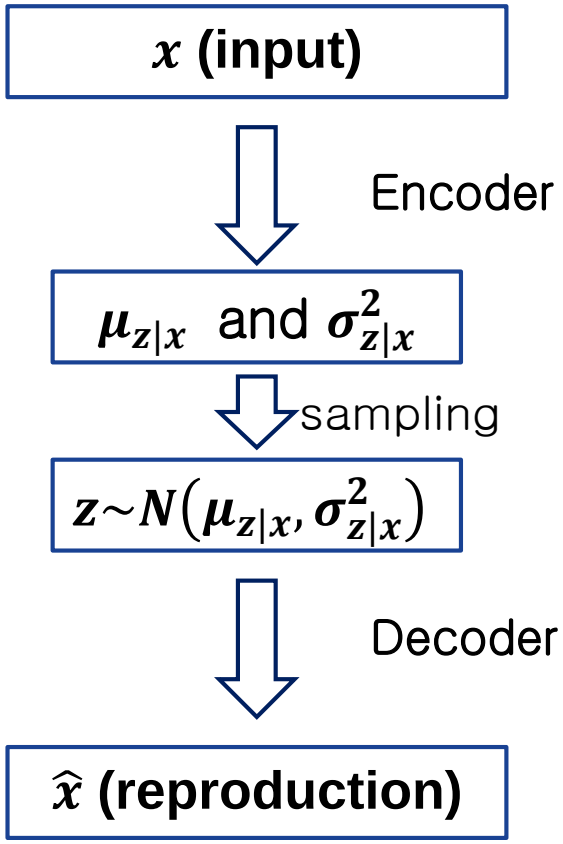
- 1st term: reconstruction loss
- 2nd term: KL-loss that encourages $q_\phi(z|x) \approx p_\theta(z)$

Latent Space of VAE

- VAE learns smooth and disentangled latent space
  - Sampling z from $\boldsymbol{N(\mu_{z|x}, \sigma^2_{z|x})}$
  - KL-divergence in ELBO

$$\mathbf{E}_z\left[\log p_\theta(x^{(i)} \mid z)\right] - D_{KL}(q_\phi(z \mid x^{(i)}) \,||\, p_\theta(z))$$

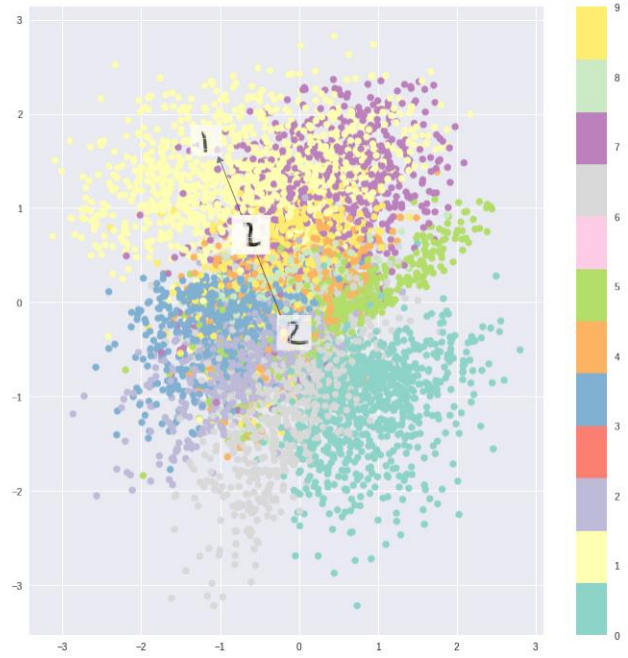




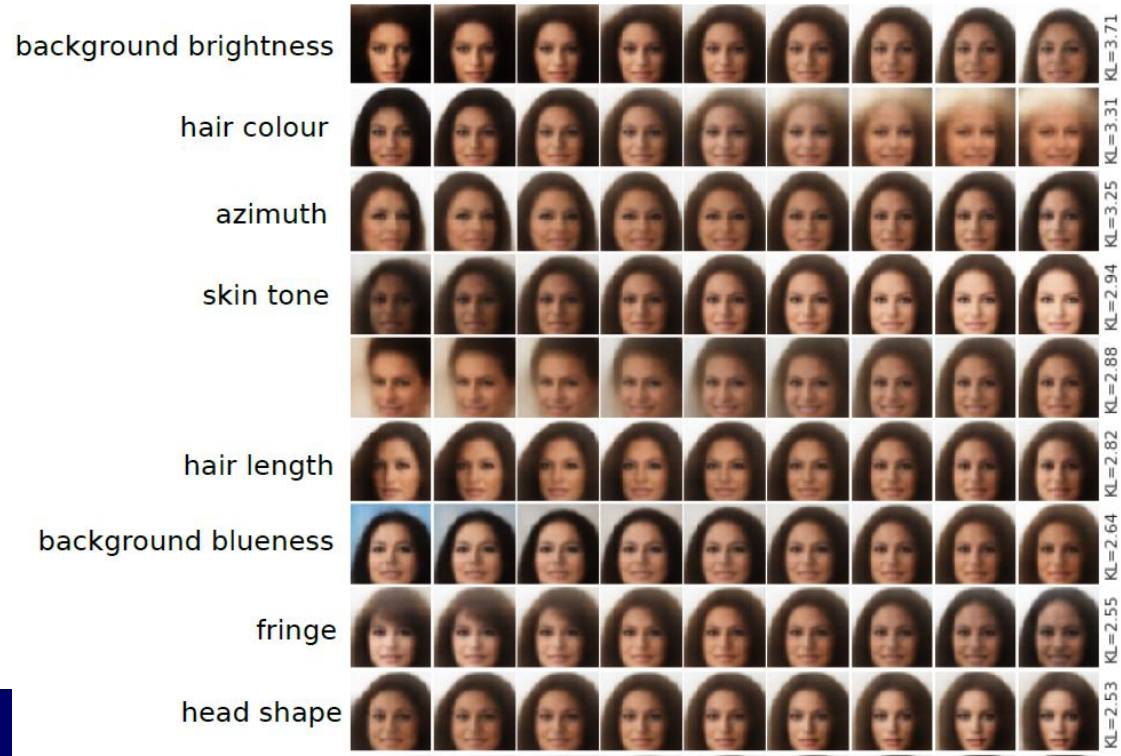
Latent Space of VAE

- VAE learns smooth and disentangled latent space
  - Sampling z from $\boldsymbol{N(\mu_{z|x}, \sigma^2_{z|x})}$
  - KL-divergence in ELBO

$$\mathbf{E}_z\left[\log p_\theta(x^{(i)} \mid z)\right] - D_{KL}(q_\phi(z \mid x^{(i)}) \,||\, p_\theta(z))$$



Factor VAE



Agenda

- Introduction to VAE
- Posterior Collapse Problem
- Preventing Posterior Collapse
- Q&A

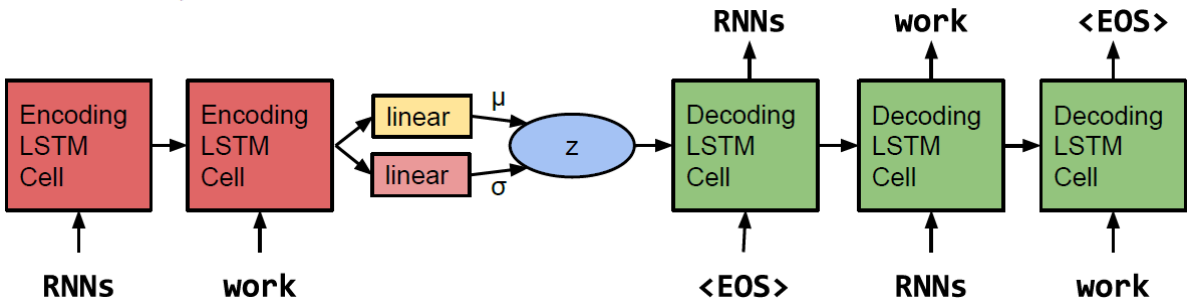
Posterior Collapse Problem

- A generative model with a powerful autoregressive decoder often ignores latent variable.
  - Produce samples independent of latent variable

➔ Hard to combine VAE and autoregressive decoder to process complex data (e.g., text, speech)

  - Hard to learn disentangled representation behind complex data
  - Hard to synthesize complex data with a specific content or style

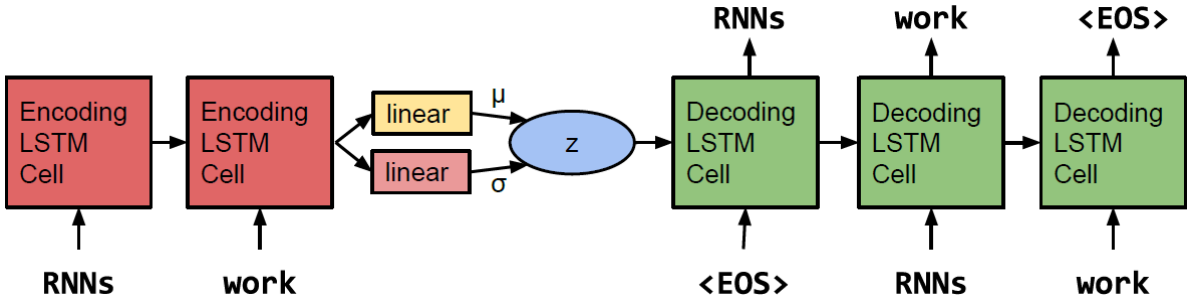
$$\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - \quad D_{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z}))$$



Posterior Collapse Problem

- Bowman et al., Sentence generation from continuous latent variable using VAE, 2016
- Optimization challenges (= posterior collapse)
  - In learning, the KL-divergence loss quickly vanishes to zero.
  - Produces x independent of latent variables z
    - VAE reduces to language model

$$\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - \quad D_{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z}))$$



- Mitigated by KL-annealing and word dropout

Why Posterior Collapses?

- Autoregressive decoder does not require latents for generation
  - $P(x_t|x_{\leq t})$ is sufficient for sample generation and $P(x_t|x_{\leq t}, z)$ is not essential
- Representation learning by a generative model is ill-posed
  - Multiple different latents that can produce the same data
  - Existence of solutions (θ, ϕ) that provide high ELBO values while ignoring z
- Information preference
  - VAE prefer to learn locally rather than relying on global latents
- Lack of good latent codes (in early stages)

Why Posterior Collapses?

- Simple uninformative prior $N(0, I)$
  - Learning minimizes $KL(q_\phi(z|x)||p_\theta(z))$, where $p_\theta(z) = N(0, I)$
- Lack of dispersion in encoder features in sequence VAE
- The approximate posterior $q_\phi(z|x)$ lags behind the true model posterior $p_\theta(z|x)$
- Gap between true model evidence and ELBO

How Does Posterior Collapse?

- The process of posterior collapse.
  - At the beginning of training, z and x are nearly independent
    - z does not carry useful information for generating x.
  - $q_\phi(z|x)$ quickly approaches the uninformative prior $(N(0, I))$
  - The powerful autoregressive decoder learns generation ignoring latent variables.
  - VAE achieves high ELBO not depending on latent variables
  - Gradient-based learning fails to make further progress

Agenda

- Introduction to VAE
- Posterior Collapse Problem
- Preventing Posterior Collapse
- Q&A

Remedies of Posterior Collapse

- Tweak ELBO
  - KL-annealing, cyclical annealing, δ-VAE
- Weaken decoder
  - Dropout, limit receptive fields of decoder
- Informative prior
  - Learning prior (AF, PixelCNN, self-attention)
  - Prior conditioned on neighbors (ACN)
- Discrete latent representation
  - VQ-VAE, VQ-VAE2, DB-VAE, etc.

Remedies of Posterior Collapse

- Reducing amortization gap
  - SA-VAE
- Enhancing dispersion in encoder feature
  - Cosine regularization
  - Sequence encoder based on pooling
- Tweak learning algorithm
  - Aggressive learning of variational posterior

Variational Lossy Auto-Encoder [Chen16]

- Representation learning via generative modeling is ill-posed
  - The problem does not just due to optimization challenges.
  - Even if we can solve the optimization problems exactly, the latent code should still be ignored.

- Information preference (=posterior collapse)
  - Information that can be modeled locally by decoding distribution $p(x|z)$ without access to z will be encoded locally
  - Only the remainder will be encoded in z.

Variational Lossy Auto-Encoder [Chen16]

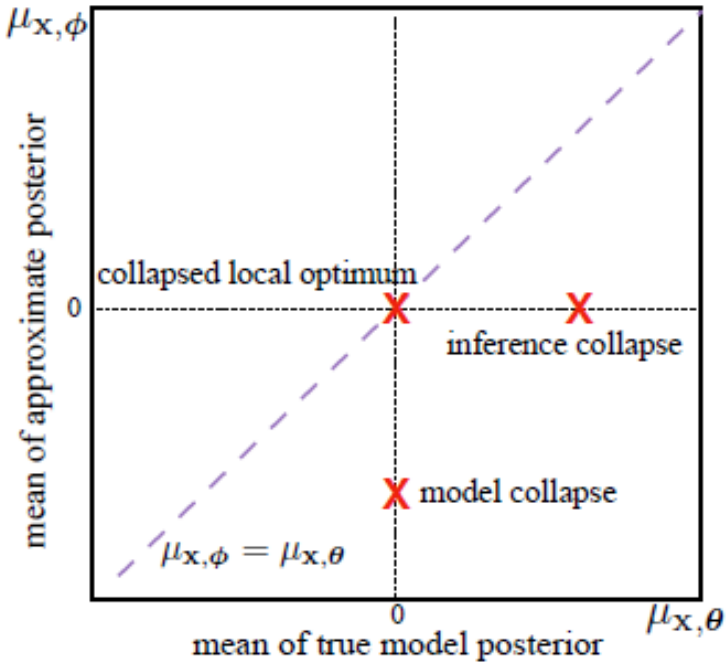
- Learning global representation by limiting receptive field of decoder
  - Ordinary autoregressive decoder: $p(x|z) = \prod_t p(x_t|z, x_{\leq i})$
  - Autoregressive decoder with limited receptive field

$$p_{local}(x|z) = \prod_t p(x_t|z, x_{WindowsAround(t)})$$

- Learn informative prior using autoregressive flow (AF)
  - Learn invertible transform from a simple distribution $u(\epsilon)$ to a complex distribution $p(z)$
    - $u(\epsilon)$: noise source
    - $p(z)$: target distribution

Aggressive Training of Inference Networks [He19]

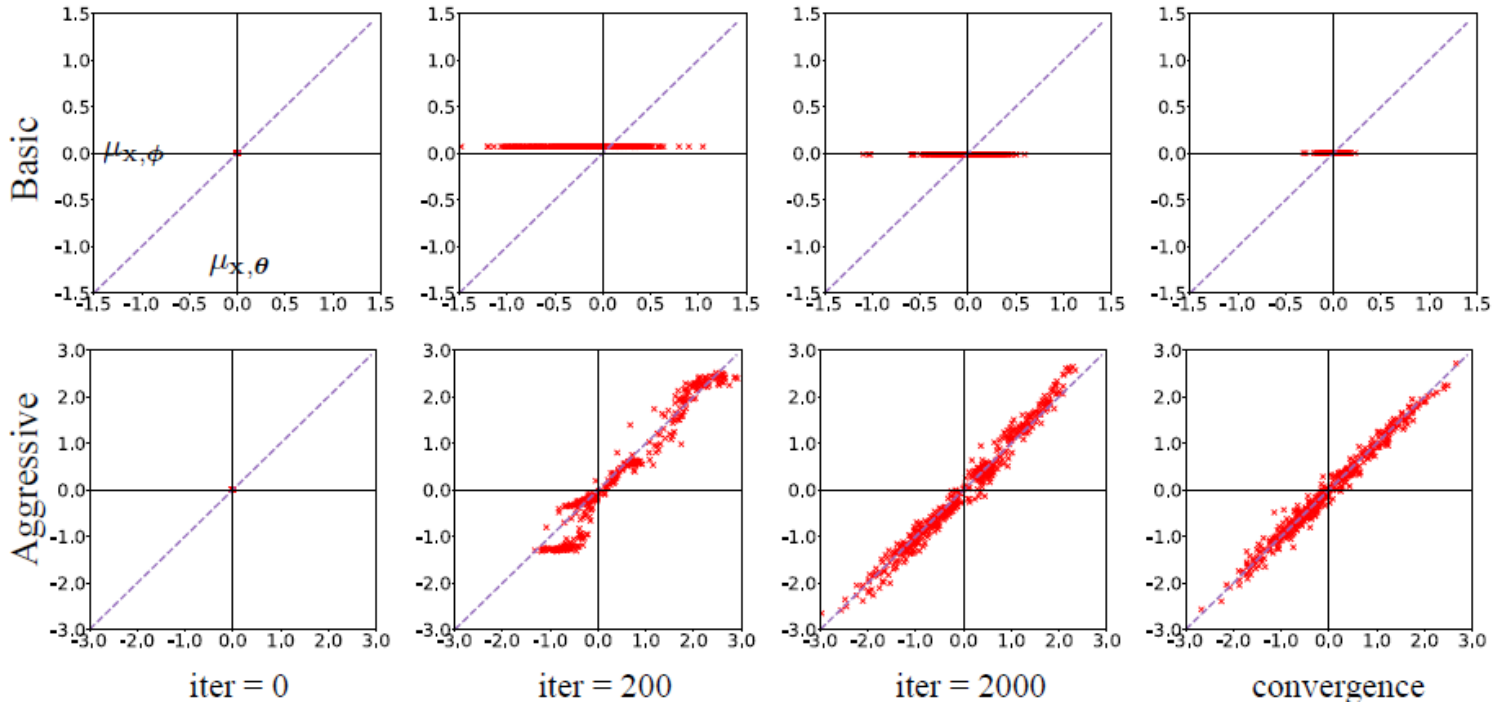
- Analysis posterior collapse from perspective of training dynamics
  - Posterior collapse: $q_\phi(z|x) = p_\theta(z|x) = p(z)$
    - $p(z) = N(0, I)$
  - Model collapse: $p_\theta(z|x) = p(z)$
    - Vertical line in figure
  - Inference collapse: $q_\phi(z|x) = p(z)$
    - Horizontal line in figure



(b) Posterior mean space

Aggressive Training of Inference Networks

- In VAE, $q_\phi(z|x)$ lags far behind $p_\theta(z|x)$



==> Aggressively training of $q_\phi(z|x)$ can prevent posterior collapse

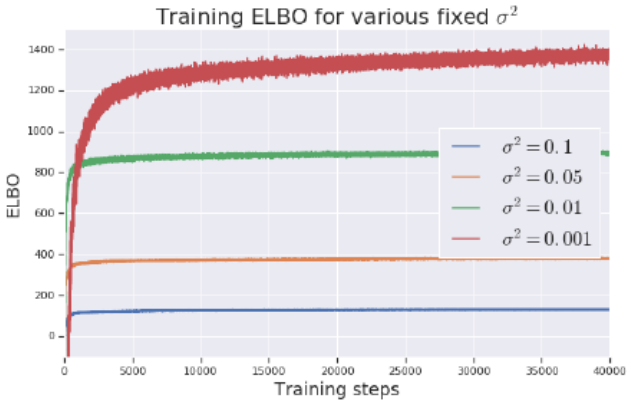
Analysis of Linear VAE via pPCA [Lucas19]

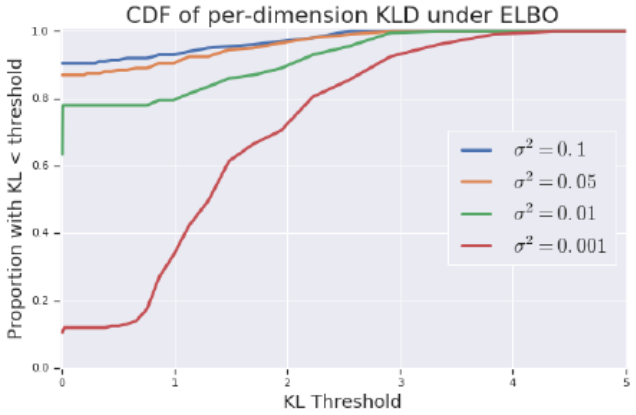
- Analyze VAE using probabilistic PCA
  - Linear VAE and pPCA has a uniquely identifiable global maximum (principal component direction)

- Findings
  - The ELBO objective does not introduce any additional local maxima to the pPCA model.
    - Posterior collapse is not entirely due to the KL-loss
    - The marginal log-likelihood $\log p(x)$ itself can encourage posterior collapse.
  - Small σ^2 prevents posterior collapse
  - KL-annealing temporarily lead to small σ^2, but insufficient to prevent posterior collapse

Analysis of Linear VAE via pPCA [Lucas19]

- Training a nonlinear VAE with fixed σ^2
  - Small σ^2 prevents posterior collapse and leads to higher ELBO





  - After training, turn σ^2 while keeping all other parameters

Model	ELBO	σ^2-tuned ELBO
$\sigma^2 = 0.1$	130.3	1302.9
$\sigma^2 = 0.05$	378.7	1376.0
$\sigma^2 = 0.01$	893.6	1435.1
$\sigma^2 = 0.001$	1379.0	1485.9

Posterior Collapse and Feature Dispersion

[Long20]

- Posterior collapse is caused in part by the lack of dispersion in encoder feature
  - In sequence VAE, input representations (the last hidden state values) are close to each other
  - Approximate posterior for each sequence concentrate in a small region carrying little information.
  - Optimization pushes approximate posteriors to prior to avoid paying the cost of the KL term

  ➔ Posterior collapse

ACN [Graves18]

- Analyze posterior collapse from the perspective of sequential data compression
  - In VAE, (benefit of using code) ≤ (coding cost)
    - Latent code is stochastic, while decoder is deterministic

  ➔ Utilizing latent code gives no benefit

- Associative compression network (ACN)
  - Motivate to utilize latent code by reducing coding cost
  - A single concept collectively associated with multiple low-level data

    Ex) first sort digits according digit class, encyclopedia
  - Lead prior to only learn local variation

ACN [Graves18]

- Associative compression network (ACN)
  - Condition the prior on a code chosen from KNN in latent space
    - $p(z|\hat{c})$ instead of $p(z)$
    - $\hat{c}$ is randomly picked from $KNN(x)$ in latent space

➔ Greatly reduces coding cost

- Only account for local variations

$$L^{ACN}(x) = \mathop{\mathbb{E}}_{\hat{c}\sim \mathrm{KNN}(x)}[KL(q(z|x)|p(z|\hat{c})] - \mathop{\mathbb{E}}_{z\sim q}[\log r(x|z)]$$

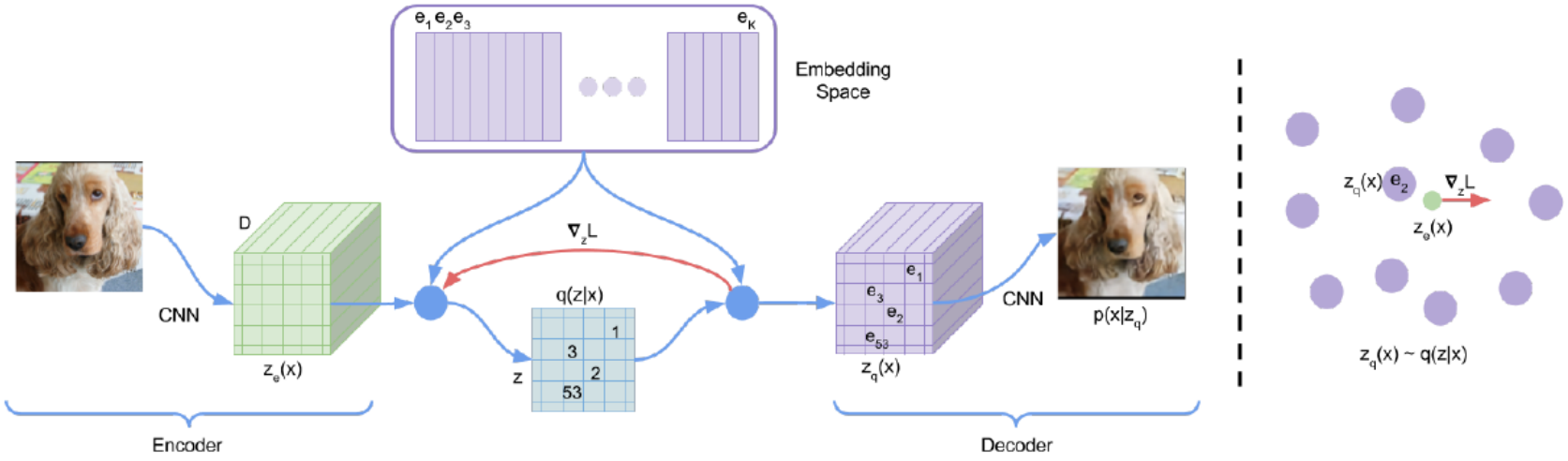
Learn $p(z|\hat{c})$ using a mixture density network

$$p(z|c) = \prod_{d=1}^{D}\sum_{m=1}^{M}\pi_m^d \mathcal{N}(z^d|\mu_m^d, \sigma_m^d)$$

VQ-VAE [Oord17]

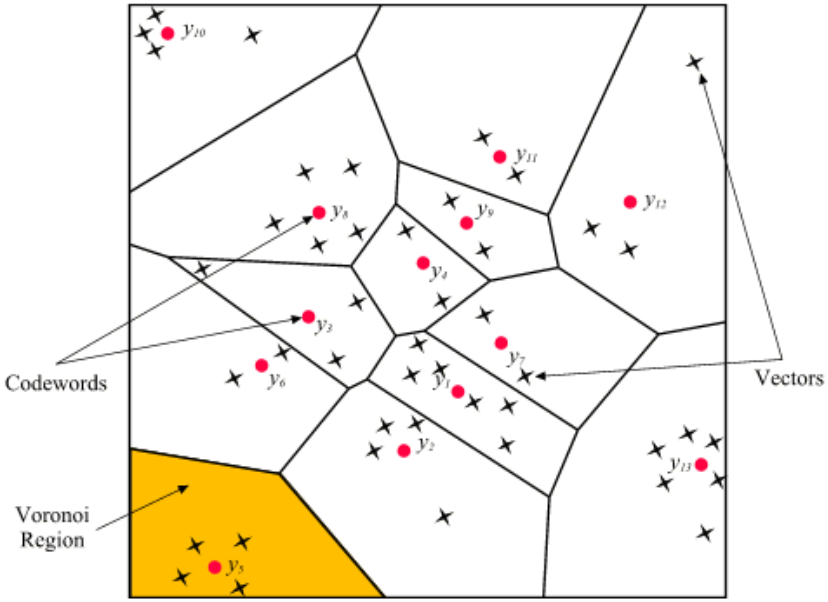
- Vector quantization VAE
  - The encoder outputs discrete codes for input data
  - Learned prior
    - Pair the discrete representations with an autoregressive prior

    ➔ Generates high quality images, sound, and videos
  - Circumvent the issue of posterior collapse



Vector Quantization

- A vector quantizer maps N−dimensional vectors in the vector space R^N into a finite set of vectors $Y = \{y_i: i = 1, 2, \ldots, K\}$.
  - Each vector y_i is called a code vector or a codeword.
  - The set of all the codewords is called a codebook.



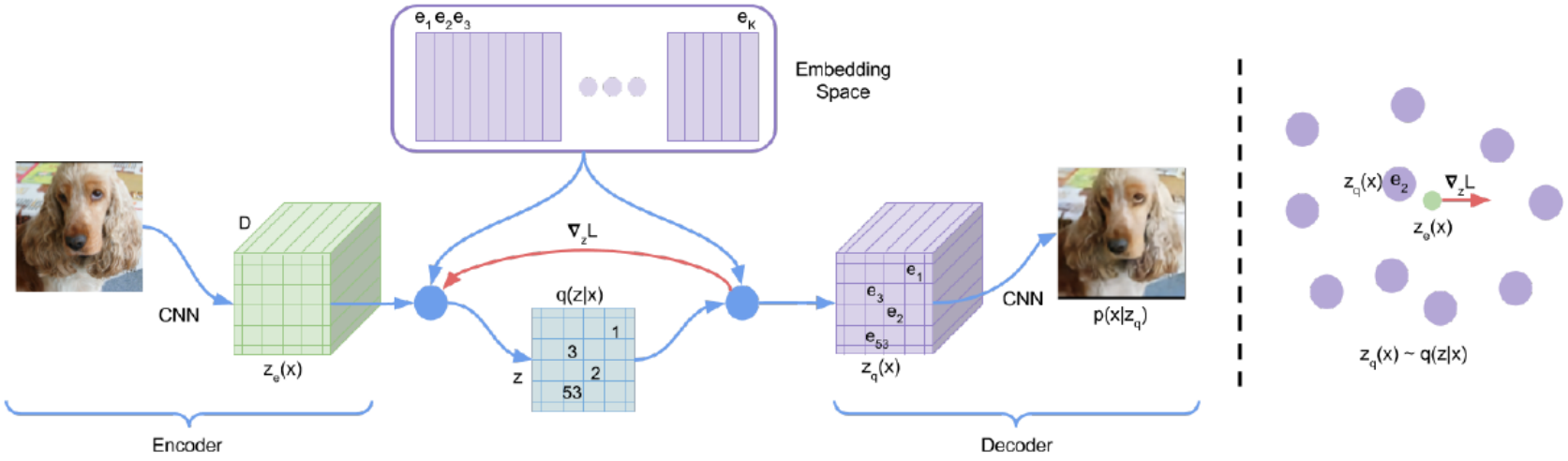
VQ-VAE [Oord17]

- Assigning input to an embedding

$$q(z=k|x) = \begin{cases} 1 & \text{for } k = \text{argmin}_j \|z_e(x) - e_j\|_2 \\ 0 & \text{otherwise} \end{cases}$$

- Quantized representation of feature map

$$z_q(x) = e_k, \quad \text{where} \quad k = \text{argmin}_j \|z_e(x) - e_j\|_2$$

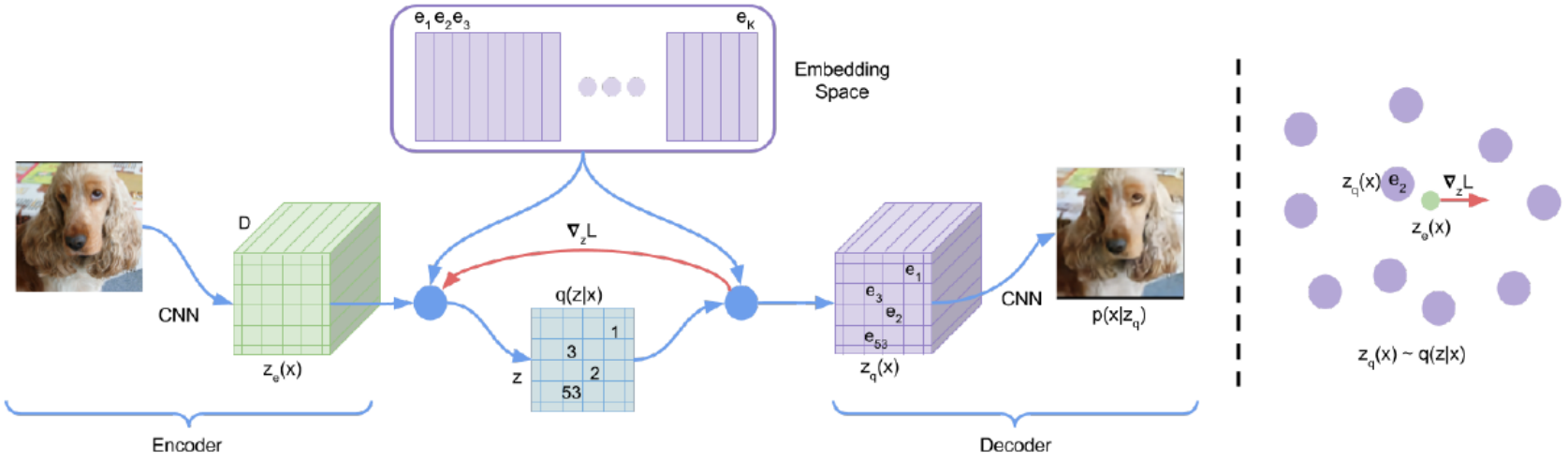


VQ-VAE [Oord17]

- Learning objective

$$L = \underbrace{\log p(x|z_q(x))}_{\text{Reconstruction loss}} + \underbrace{\|\text{sg}[z_e(x)] - e\|_2^2}_{\text{codebook loss to learn embeddings}} + \underbrace{\beta\|z_e(x) - \text{sg}[e]\|_2^2}_{\text{commitment loss to learn encoder}}$$

  - Codebook loss can be replaced by moving average of $z_e(x)$



VQ-VAE [Oord17]

- Learning prior distribution over discrete latents $p(z)$
  1. Training VQ-VAE using constant uniform prior
  2. Learn a autoregressive distribution over z (PixelCNN)

  ➔ Improves output quality

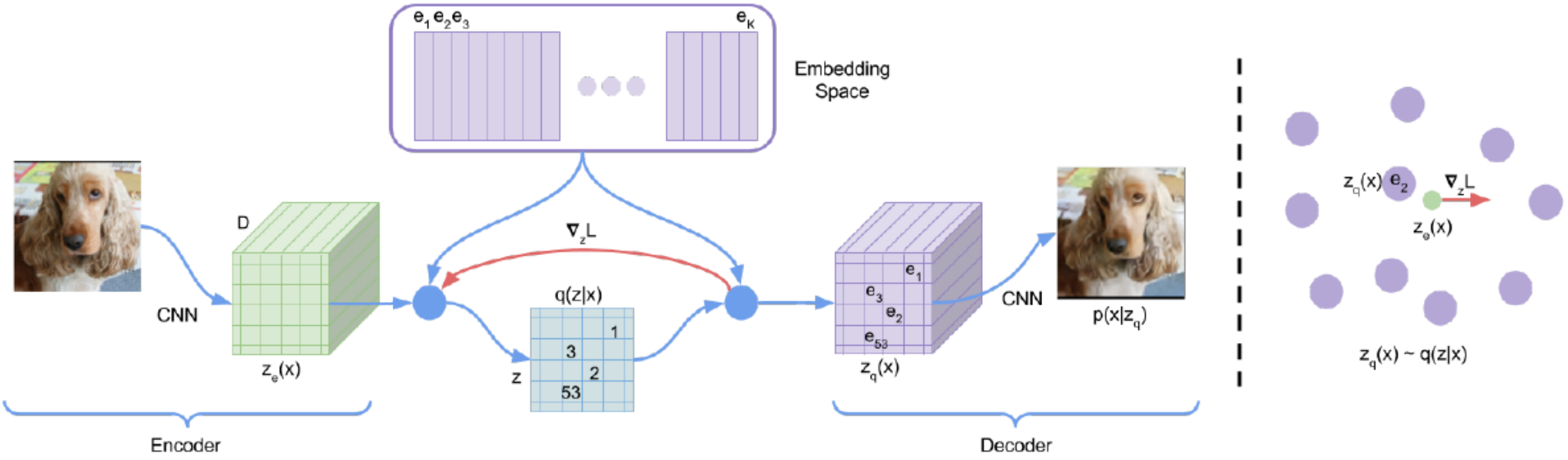


Image Synthesis using VQ-VAE

- Reconstruction using VQ-VAE and PixelCNN decoder
  - Latents are meaningfully used
    - 21x21 latents represent global structure
    - PixelCNN decoder generates texture

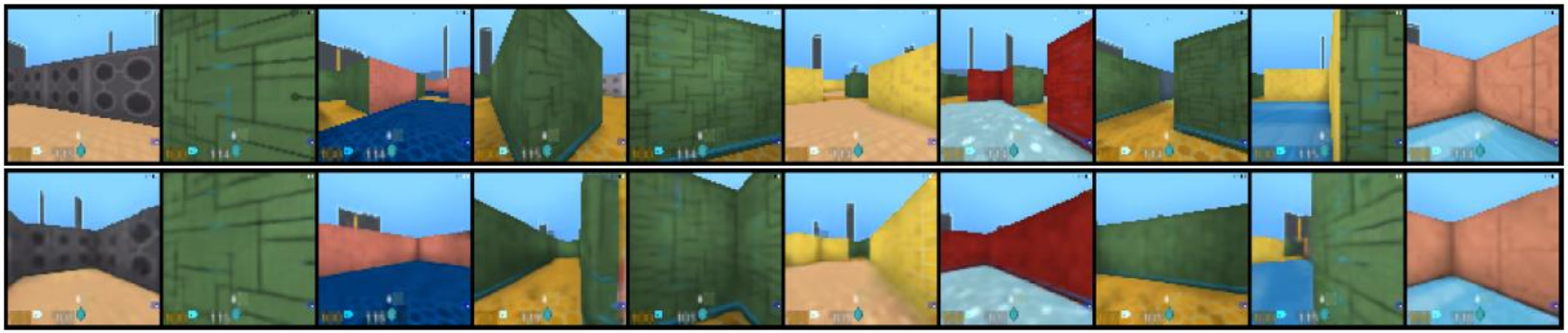
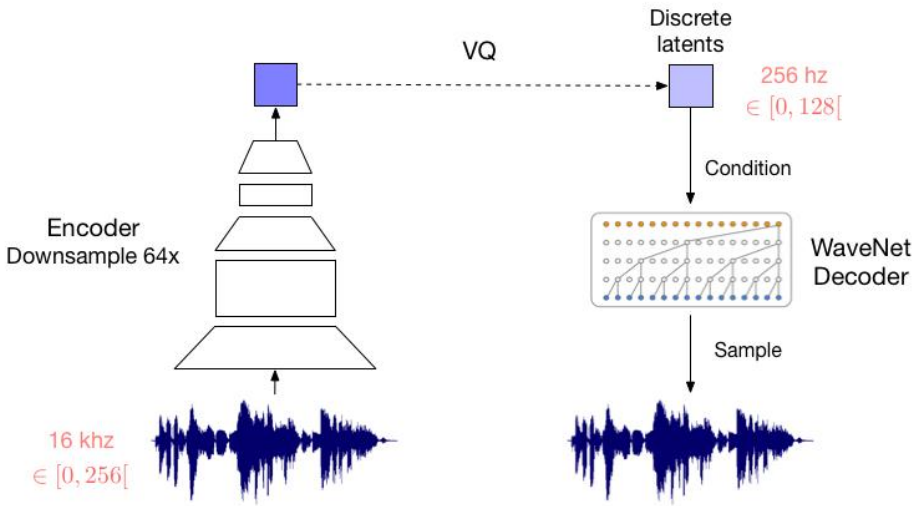


Figure 5: Top original images, Bottom: reconstructions from a 2 stage VQ-VAE, with 3 latents to model the whole image (27 bits), and as such the model cannot reconstruct the images perfectly. The reconstructions are generated by sampled from the second PixelCNN prior in the 21x21 latent domain of first VQ-VAE, and then decoded with standard VQ-VAE decoder to 84x84. A lot of the original scene, including textures, room layout and nearby walls remain, but the model does not try to store the pixel values themselves, which means the textures are generated procedurally by the PixelCNN.

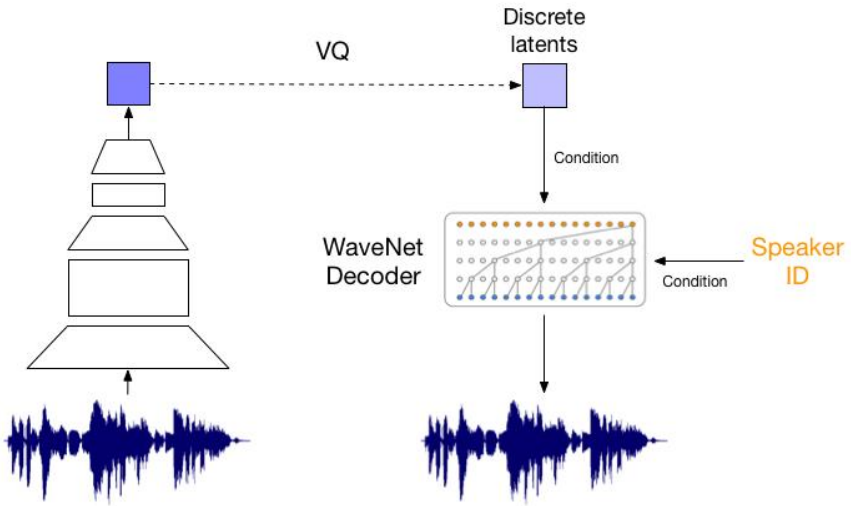
Voice Synthesis using VQ-VAE

- VQ-VAE discovers discrete latent codes similar to phonemes

Voice reconstruction

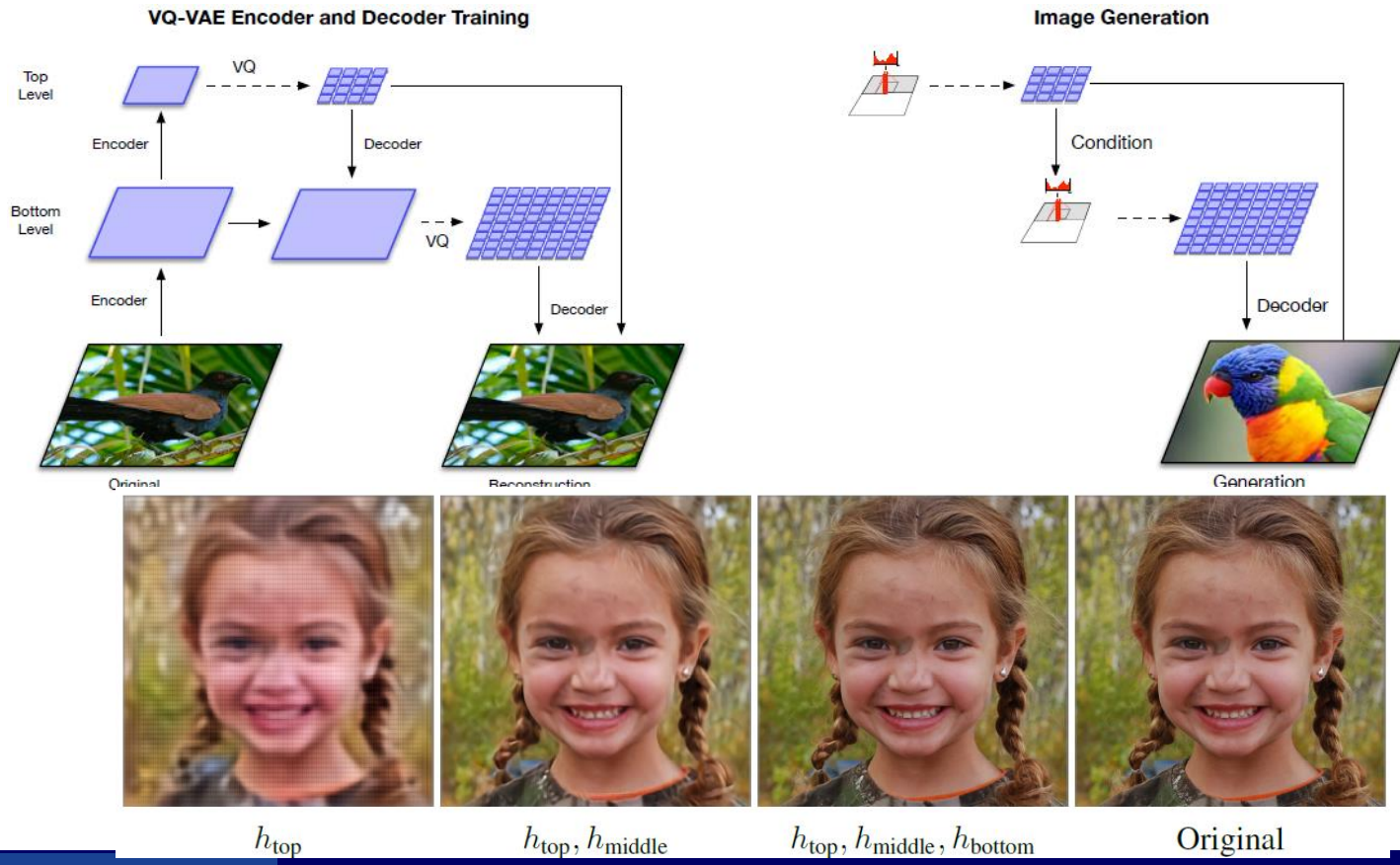


Voice style transfer



VQ-VAE2 [Razavi19]

- Learning hierarchical latent codes for large images
  - Separately learn global and local information



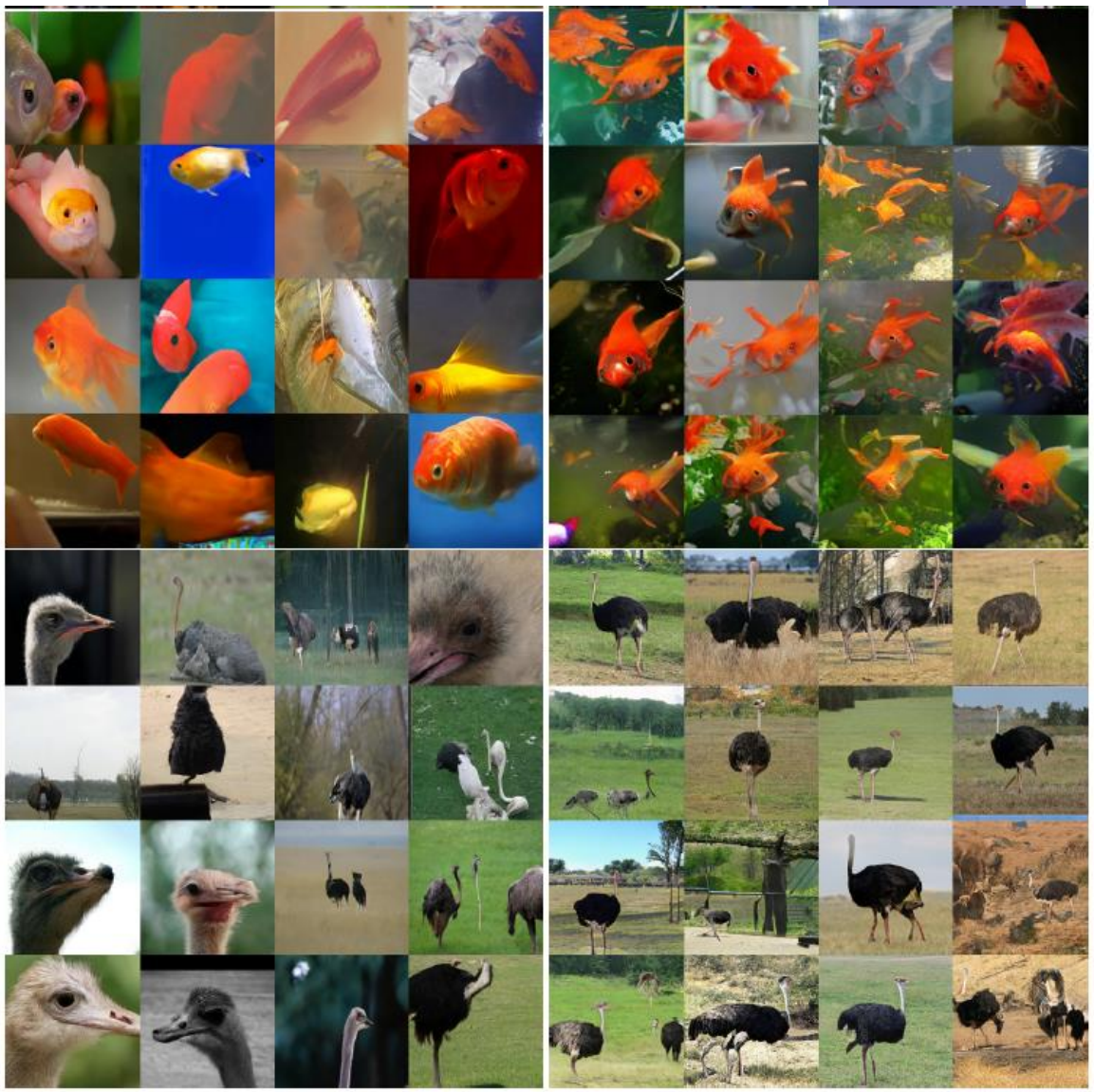
VQ-VAE2 [Razavi19]

1024x1024 images from three level hierarchical VQ-VAE

The model generates realistic looking faces that respect long-range dependencies such as matching eye colour or symmetric facial features, while covering lower density modes of the dataset (e.g., green hair).

VQ-VAE2 [Razavi19]

More diversity than BigGAN



VQ-VAE (Proposed) BigGAN deep

Discrete Latent Representation

- Discrete latents improve sample quality and prevent posterior collapse
  - In many takes, underlying causal factors are discrete
    - Class, text, action, phoneme, speaker, etc.
  - Effective and efficient
    - Motivates the model to utilize latents
  - Prevents the KL-term in ELBO from vanishing
  - Relieves the model from learning negligible information
    - Information bottleneck
    - Learns latents focusing on global information

    c.f. Local detail is learned by a powerful decoder
  - Easy to learn informative prior

Recent Work based on Vector Quantization

- Zhao et al., Discretized Bottleneck: Posterior-Collapse-Free Sequence-to-Sequence Learning (DB-VAE), 2020
- Baevski et al., vq-wav2vec: Self-Supervised Learning of Discrete Speech Representations, 2019
- Niekerk et al., Vector-quantized neural networks for acoustic unit discovery in the ZeroSpeech 2020 challenge, 2020
- Wu et al., VQVC+: One-Shot Voice Conversion by Vector Quantization and U-Net architecture, 2020
- Prato et al., Fully Quantized Transformer for Machine Translation, 2020

Q&A

Thank you
for your attention!