TTIC 31230, Fundamentals of Deep Learning

David McAllester, Winter 2019

Variational Autoencoders (VAEs)

Latent Variable Models

$$P_\Phi(y) = \sum_z P_\Phi(z) P_\Phi(y|z) = E_{z \sim P_\Phi(z)} P_\Phi(y|z)$$

Or

$$P_\Phi(y|x) = \sum_z P_\Phi(z|x) P_\Phi(y|z,x) = E_{z \sim P_\Phi(z|x)} P_\Phi(y|z,x)$$

Here z is a latent variable.

Latent Variable Models

$$P_\Phi(y) = \sum_z P_\Phi(z) P_\Phi(y|z) = E_{z \sim P_\Phi(z)} P_\Phi(y|z)$$

Here we often think of z as the causal source of y.

For example z might be a physical scene causing image y.

Or z might be the intended utterance causing speech signal y.

In these situations a latent variable model should more accurately represents the distribution on y.

Latent Variable Models

$$P_\Phi(y) = \sum_z P_\Phi(z) P_\Phi(y|z) = E_{z \sim P_\Phi(z)} P_\Phi(y|z)$$

$P_\Phi(z)$ is called the prior.

Given an observation of y (the evidence) $P_\Phi(z|y)$ is called the posterior.

Variational Bayesian inference involves approximating the posterior.

Colorization with Latent Segmentation

x $\hat{y}$ y

Larsson et al., 2016

Colorization is a natural self-supervised learning problem — we delete the color and then try to recover it from the grey-level image.

Can colorization be used to learn segmentation?

Segmentation is latent — not determined by the color label.

Colorization with Latent Segmentation

x $\hat{y}$ y

Larsson et al., 2016

x is a grey level image.

y is a color image drawn from $\text{Pop}(y|x)$.

$\hat{y}$ is an arbitrary color image.

$P_\Phi(\hat{y}|x)$ is the probability that model Φ assigns to the color image $\hat{y}$ given grey level image x.

Colorization with Latent Segmentation

Input | Our Method | Ground-truth

x $\hat{y}$ y

$$P_\Phi(\hat{y}|x) = \sum_z P_\Phi(z|x) P_\Phi(\hat{y}|z,x).$$

input x

$P_\Phi(z|x) = \ldots$ semantic segmentation

$P_\Phi(\hat{y}|z,x) = \ldots$ segment colorization

Assumptions

We assume models $P_\Phi(z)$ and $P_\Phi(y|z)$ are both samplable and computable.

In other words, we can sample from these distributions and for any given z and y we can compute $P_\Phi(z)$ and $P_\Phi(y|z)$.

These are nontrivial assumptions.

A loopy graphical model is neither (efficiently) samplable nor computable.

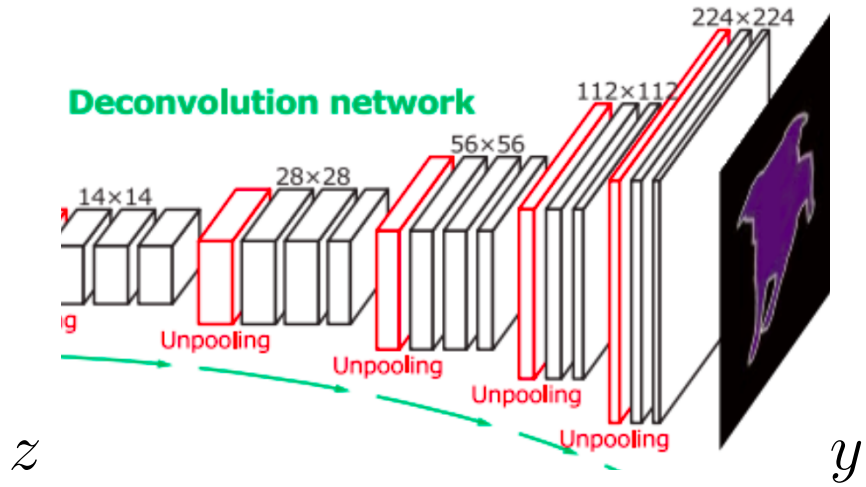
Cases Where the Assumptions Hold

In CTC we have that z is the sequence with blanks and y is the result of removing the blanks from z.

In a hidden markov model z is the sequence of hidden states and y is the sequence of emissions.

An autoregressive model, such as an autoregressive language model, is both samplable and computable.

Image Generators



We can generate an image y form noise z where $p_\Phi(z)$ and $p_\Phi(y|z)$ are both samplable and computable.

Typically $p_\Phi(z)$ is $\mathcal{N}(0, I)$ reshaped as $z[X, Y, J]$

Assumptions

Even when $P_\Phi(z)$ and $P_\Phi(y|z)$ are samplable and computable we cannot typically compute $P_\Phi(y)$.

Specifically, for $P_\Phi(y)$ defined by a GAN generator we cannot compute $P_\Phi(y)$ for a test image y.

Hence it is not obvious how to optimize the fundamental equation.

$$\Phi^* = \underset{\Phi}{\operatorname{argmin}} \; E_{y \sim \mathrm{Pop}} \; - \ln P_\Phi(y)$$

The Evidence Lower Bound (ELBO)

$$
\begin{aligned}
P_\Phi(y) &= \frac{P_\Phi(y)P_\Phi(z|y)}{P_\Phi(z|y)} \\
&= \frac{P_\Phi(z)P_\Phi(y|z)}{P_\Phi(z|y)} \\
\ln P_\Phi(y) &= \ln \frac{P_\Phi(z)P_\Phi(y|z)}{P_\Phi(z|y)} \\
&= E_{z\sim Q_\Phi(z|y)} \ln \frac{P_\Phi(z)P_\Phi(y|z)}{P_\Phi(z|y)}
\end{aligned}
$$

The Evidence Lower Bound (The ELBO)

We introduce a samplable and computable model $Q_\Phi(z|y)$ to approximate $P_\Phi(z|y)$.

$$
\begin{aligned}
\ln P_\Phi(y) &= E_{z\sim Q_\Phi(z|y)} \ln \frac{P_\Phi(z)P_\Phi(y|z)}{P_\Phi(z|y)} \\
&= E_{z\sim Q_\Phi(z|y)} \left(\ln \frac{P_\Phi(z)P_\Phi(y|z)}{Q_\Phi(z|y)} + \ln \frac{Q_\Phi(z|y)}{P_\Phi(z|y)} \right) \\
&= \left(E_{z\sim Q_\Phi(z|y)} \ln \frac{P_\Phi(z)P_\Phi(y|z)}{Q_\Phi(z|y)} \right) + KL(Q_\Phi(z|y), P_\Phi(z|y)) \\
&\geq E_{z\sim Q_\Phi(z|y)} \ln \frac{P_\Phi(z)P_\Phi(y|z)}{Q_\Phi(z|y)} \quad \text{The ELBO}
\end{aligned}
$$

The Variational Autoencoder (VAE)

$$\Phi^* = \underset{\Phi}{\operatorname{argmin}}\, E_{y\sim \mathrm{Pop},\, z\sim Q_\Phi(z|y)} \quad -\ln \frac{P_\Phi(z,y)}{Q_\Phi(z|y)}$$

VAE generalizes EM

Expectation Maximimization (EM) applies in the (highly special) case where the exact posterior $P_\Phi(z|y)$ is samplable and computable. EM alternates exact optimization of Q and P.

$$\text{VAE: } \Phi^* = \text{argmin}_\Phi \, E_{y\sim\text{Train},\, z\sim Q_\Phi(z|y)} \quad -\ln \frac{P_\Phi(z,y)}{Q_\Phi(z|y)}$$

$$\text{EM: } \Phi^{t+1} = \text{argmin}_\Phi \;\; E_{y\sim\text{Train}} \, E_{z\sim P_{\Phi^t}(z|y)} \;\; -\ln P_\Phi(z,y)$$

Update (M Step)	Inference (E Step)
Hold Q fixed	$Q(z\|y) = P_{\Phi^t}(z\|y)$

Hard EM relies on Closed Form Q^*

EM: $\Phi^{t+1} = \text{argmin}_\Phi \; E_{y,z\sim P_{\Phi^t}(z|y)} \; -\ln P_\Phi(z,y)$

Hard EM: $\Phi^{t+1} = \text{argmin}_\Phi \; E_{y,z=\text{argmax}_z P_{\Phi^t}(z|y)} \; -\ln P_\Phi(z,y)$

This relies on $P_{\Phi^t}(z|y)$ being exactly computable so that the optimization over Q in VAE has a closed form solution.

There does not seem to be a sensible "hard VAE".

The Reparameterization Trick

$$-\ln P_\Phi(y) \leq E_{z\sim Q_\Phi(z|y)} \; -\ln \frac{P_\Phi(z)P_\Phi(y|z)}{Q_\Phi(z|y)}$$

$$= E_\epsilon \; -\ln \frac{P_\Phi(z)P_\Phi(y|z)}{Q_\Phi(z|y)} \quad z := f_\Phi(y,\epsilon)$$

ϵ is parameter-independent noise.

This supports SGD: $\nabla_\Phi \, E_{y,\epsilon} \, [\ldots] = E_{y,\epsilon} \, \nabla_\Phi \, [\ldots]$

Posterior Collapse

Assume Universal Expressiveness for $P_\Phi(y|z)$.

This allows $P_\Phi(y|z) = \text{Pop}(y)$ independent of z.

We then get a completely optimized model with z taking a single (meaningless) determined value.

$$Q_\Phi(z|y) = P_\Phi(z|y) = 1$$

Colorization with Latent Segmentation

x $\hat{y}$ y

Larsson et al., 2016

Can colorization be used to learn latent segmentation?

We introduce a latent segmentation into the model.

In practice the latent segmentation is likely to "collapse" because the colorization can be done just as well without it.

Optimizing the VAE leaves $I(y, z)$ undetermined

Complete optimization gives $P_\Phi(y) = \mathrm{Pop}(y)$. But this does not determine $Q_\Phi(z|y)$.

At complete optimization the value of the objective function is $H(y)$. But we have

$$H(y) = I(y, z) + H(y|z)$$

The VAE operates on the joint distribution on y and z determined by $\mathrm{Pop}(y)$ and $Q_\Phi(z|y)$.

Posterior collapse is the case of $I(y, z) = 0$.

The β-VAE

β-VAE: Learning Basic Visual Concepts With A Constrained Variational Framework, Higgins et al., ICLR 2017.

The VAE:

$$\Phi^* = \underset{\Phi}{\operatorname{argmin}}\, E_{y,\epsilon} \; - \ln \frac{P_\Phi(z) P_\Phi(y|z)}{Q_\Phi(z|y)}$$

The β-VAE:

$$\Phi^* = \underset{\Phi}{\operatorname{argmin}}\, E_{y,\epsilon} \quad - \beta \ln \frac{P_\Phi(z)}{Q_\Phi(z|y)} - \ln P_\Phi(y|z)$$

The β-VAE

The β-VAE:

$$\Phi^* = \underset{\Phi}{\operatorname{argmin}}\, E_{y,\epsilon} \quad -\beta \ln \frac{P_\Phi(z)}{Q_\Phi(z|y)} - \ln P_\Phi(y|z)$$

The paper claims that taking $\beta > 1$ can prevent posterior collapse. More Later.

Noisy-Channel RDAs vs. β-VAEs

Noisy-Channel RDA

$$\Phi^* = \underset{\Phi}{\operatorname{argmin}}\, E_{y,\epsilon} \; -\ln \frac{q_\Phi(z)}{p_\Phi(z|y)} + \lambda \mathrm{Dist}(y, y_\Phi(z)) \quad z := f_\Phi(y, \epsilon)$$

β-VAE

$$\Phi^* = \underset{\Phi}{\operatorname{argmin}}\, E_{y,\epsilon} \quad -\beta \ln \frac{p_\Phi(z)}{q_\Phi(z|y)} - \ln p_\Phi(y|z) \quad z := f_\Phi(y, \epsilon)$$

L_2 Distortion and Gaussian Image Noise

Using L_2 distortion and $p_\Phi(y|z) \propto \exp(-||y - y_\Phi(z)||/(2\sigma^2))$

Noisy-Channel RDA

$$\Phi^* = \underset{\Phi}{\operatorname{argmin}}\, E_{y,\epsilon} \; - \ln \frac{p_\Phi(z|y)}{q_\Phi(z)} + \lambda ||y - y_\Phi(z)||^2$$

β-VAE

$$\Phi^* = \underset{\Phi,\sigma}{\operatorname{argmin}}\, E_{y,\epsilon} \quad - \beta \ln \frac{p_\Phi(z)}{q_\Phi(z|y)} + \left(\frac{1}{2\sigma^2}\right) ||y - y_\Phi(z)||^2 + \ln \sigma$$

Gaussian Image Noise

β-VAE:

$$\Phi^*, \sigma^* = \underset{\Phi,\sigma}{\operatorname{argmin}} \, E_{y,\epsilon} - \beta \ln \frac{p_\Phi(z)}{q_\Phi(z|y)} + \left(\frac{1}{2\sigma^2} \right) ||y - y_\Phi(z)||^2 + \ln \sigma$$

where

$$\sigma^* = \sqrt{E_{y,\epsilon} \, ||y - y_\Phi(z)||^2}$$

Partial Optimization

Noisy-Channel RDA

$$\Phi^* = \operatorname*{argmin}_{\Phi} E_{y,\epsilon} \; - \ln \frac{q_\Phi(z)}{p_\Phi(z|y)} + \lambda \mathrm{Dist}(y, y_\Phi(z))$$

$$q^*(z) = p_{\mathrm{pop}}(z) = E_{y \sim \mathrm{pop}} \, p_\Phi(z|y)$$

β-VAE

$$\Phi^* = \operatorname*{argmin}_{\Phi} E_{y,\epsilon} \quad - \beta \ln \frac{p_\Phi(z)}{q_\Phi(z|y)} - \ln p_\Phi(y|z)$$

$$p^*(z) = p_{\mathrm{pop}}(z) = E_{y \sim \mathrm{pop}} \, q_\Phi(z|y)$$

Inserting the Optima

Noisy-Channel RDA

$$\Phi^* = \underset{\Phi}{\operatorname{argmin}} \quad I(y,z) + \lambda E_{y,\epsilon} \operatorname{Dist}(y, y_\Phi(z))$$

β-VAE

$$\Phi^* = \underset{\Phi}{\operatorname{argmin}} \quad \beta I(y,z) + E_{y,\epsilon} \; -\ln p_\Phi(y|z)$$

where the joint distribution on y and z is determined by $\operatorname{pop}(y)$ and the respective encoder distributions.

Semantics of the β-VAE seems unclear

$$\Phi^* = \underset{\Phi}{\operatorname{argmin}} \quad \beta I(y, z) + E_{y,\epsilon} \; - \ln p_\Phi(y|z)$$

We are minimizing the mutual information term. To encourage large mutual information we should take $\beta < 1$ not $\beta > 1$ as recommended.

Gaussian VAEs

$$\Phi^* = \operatorname*{argmin}_{\Phi} E_{y,\epsilon} \; -\ln \frac{p_\Phi(z)p_\Phi(y|z)}{q_\Phi(z|y)}$$

$$z[i] = z_\Phi(y)[i] + \sigma_\Phi(y)\epsilon[i] \quad \epsilon[i] \sim \mathcal{N}(0,1)$$

$$q_\Phi(z[i]|y) = \mathcal{N}(z_\Phi(y)[i], \sigma_\Phi(y)[i])$$ → approximate $p(z|y)$

$$p_\Phi(z[i]) = \mathcal{N}(\mu_p[i], \sigma_p[i]) \;\; \text{WLOG} = \mathcal{N}(0,1)$$ → prior

$$p_\Phi(y[i]|z) = \mathcal{N}(y_\Phi(z)[i], \; \sigma_\Phi(z)[i])$$ → likelihood

VAEs 2013

Sample $z \sim \mathcal{N}(0, I)$ and compute $y_\Phi(z)$

[Alec Radford]

VQ-VAE-2, June 2019

VQ-VAE-2, Razavi et al. June, 2019

VQ-VAE-2, June 2019

Figure 1: Class-conditional 256x256 image samples from a two-level model trained on ImageNet.

VQ-VAE-2, Razavi et al. June, 2019

Vector Quantized VAEs (VQ-VAE)

VQ-VAEs effectively perform k-means on vectors in the model so as to represent vectors by discrete cluster centers.

For concreteness we will consider VQ-VAEs on images with a single layer of quantization.

We use x and y for spatial image coordinates and use s (for signal) to denote images.

VQ-VAE Encoder-Decoder

We train a dictionary $C[K, I]$ where $C[k, I]$ is the center vector of cluster k.

$$L[X, Y, I] = \text{Enc}_\Phi(s)$$

$$z[x, y] = \underset{k}{\text{argmin}} \; ||L[x, y, I] - C[k, I]||$$

$$\hat{L}[x, y, I] = C[z[x, y], I]$$

$$\hat{s} = \text{Dec}_\Phi(\hat{L}[X, Y, I])$$

The "symbolic image" $z[X, Y]$ is the latent variable.

VQ-VAE Training Loss

We will interpret the VQ-VAE as a noisy-channel RDA.

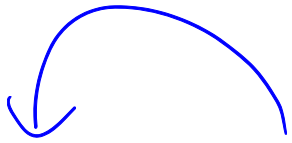
$$\Phi^* = \underset{\Phi}{\operatorname{argmin}}\, E_s\, I(s, z) + \lambda \mathrm{Dist}(s, \hat{s})$$

The mutual information $I(s, z)$ is limited by the entropy of $z[X, Y]$ which can be no larger than $\ln K^{XY} = XY \ln K$.

Maximizing $I(s, z)$ subject to this upper bound should reduce the distortion by providing the decoder with adequate information about the image.

VQ-VAE Training Loss

We preserve information about the image s by minimizing the distortion between $L[X, Y, I]$ and its reconstruction $\hat{L}[X, Y, I]$.



$$\Phi^* = \underset{\Phi}{\text{argmin}} \; E_s \sum_{x,y} \beta ||L[x, y, I] - \hat{L}[x, y, I]||^2 + ||s - \hat{s}||^2$$

This is a two-level rate-distortion auto-encoder where the rate is ultimately governed by the size K of the codebook $C[K, I]$.

VQ-VAE Training Loss

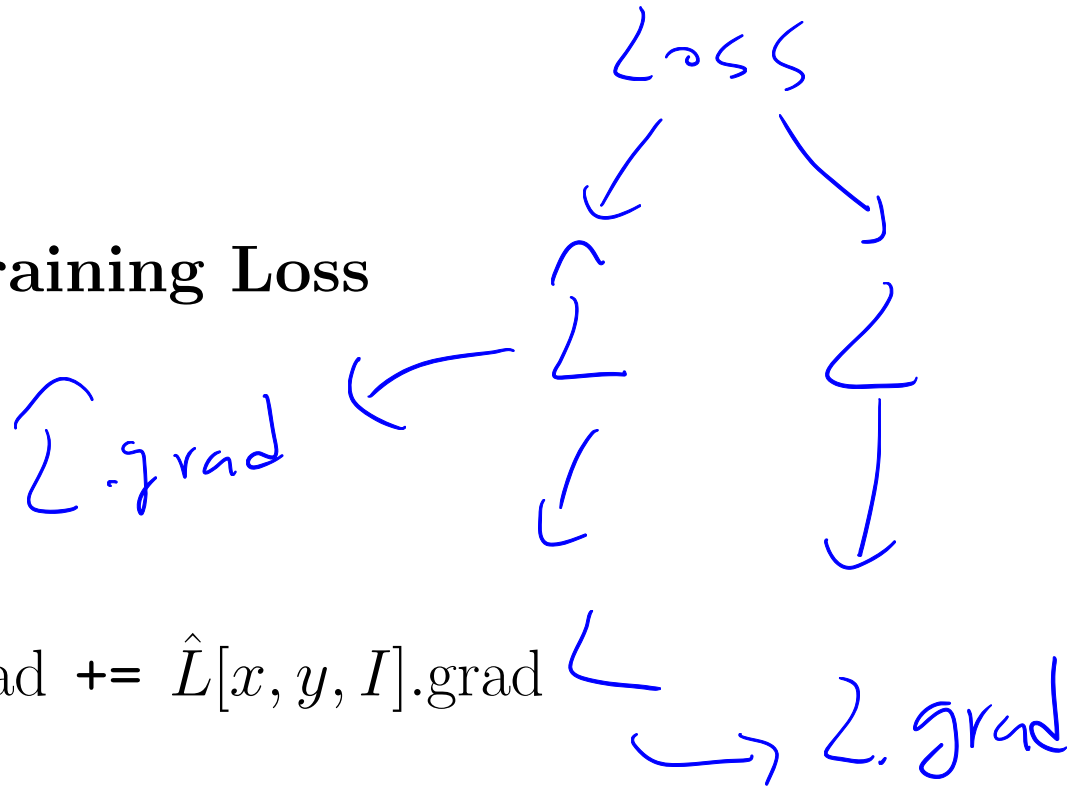
Unfortunately the latent variable $z[X, Y]$ is discrete and has no gradient. Hence some approximation must be used. They use:

$$\text{for } x, y \;\; L[x, y, I].\text{grad} \;\texttt{+=}\; \hat{L}[x, y, I].\text{grad}$$

$$\text{for } x, y \;\; L[x, y, I].\text{grad} \;\texttt{+=}\; \beta(L[x, y, I] - C[z[x, y], I])$$

$$\text{for } x, y \;\; C[z[x, y], I].\text{grad} \;\texttt{+=}\; C[z[x, y], I] - L[x, y, I]$$

VQ-VAE Training Loss



for x, y $L[x, y, I]$.grad += $\hat{L}[x, y, I]$.grad

This is the "straight through" gradient. This makes sense for low distortion between $L[X, Y, I]$ and $\hat{L}[X, Y, I]$.

VQ-VAE Training Loss

$$\text{for } x, y \;\; L[x, y, I].\text{grad} \;\texttt{+=}\; \beta(L[x, y, I] - C[z[x, y], I])$$
$$\text{for } x, y \;\; C[z[x, y], I].\text{grad} \;\texttt{+=}\; C[z[x, y], I] - L[x, y, I]$$

This is the gradient of the loss term $\beta||L[x, y, I] - \hat{L}[x, y, I]||^2$ which equals $\beta||L[x, y, I] - C[z[x, y], I]||^2$.

The absense of the weight β in the gradient update for $C[K, I]$ is equivalent to using a different learning rate for these parameters. Giving different parameters different learning rates does not change the optimum (the stationary points).

VQ-VAE Training Loss

$$\text{for } x, y \;\; L[x, y, I].\text{grad } \texttt{+=} \; \beta(L[x, y, I] - C[z[x, y], I])$$

$$\text{for } x, y \;\; C[z[x, y], I].\text{grad } \texttt{+=} \; C[z[x, y], I] - L[x, y, I]$$

At a stationary point we get that $C[k, I]$ is the mean of the set of vectors $L[x, y, I]$ with $z[x, y] = k$ (as in k-means).

Training Phase II

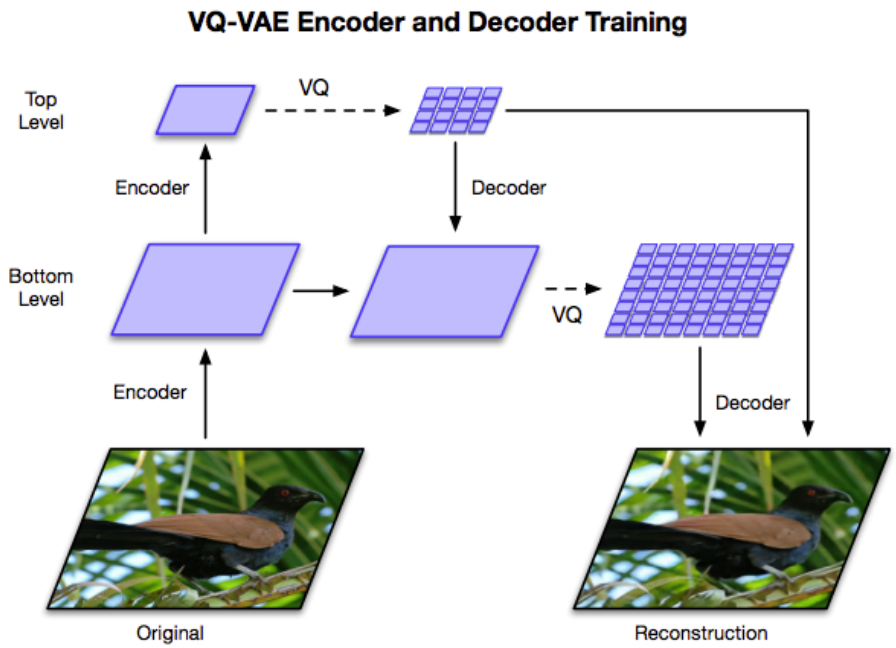
Once the model is trained we can sample images s and compute the "symbolic image" $z[X, Y]$.

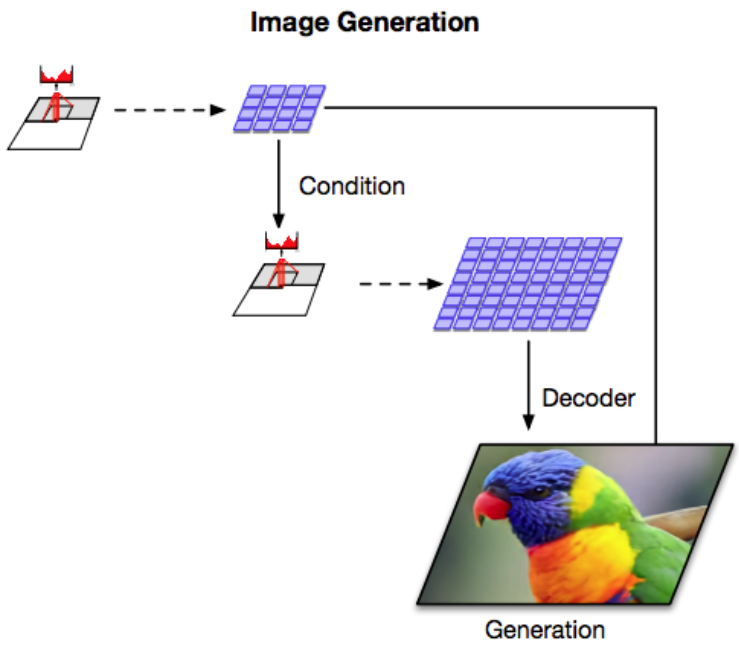
Given samples of symbolic images $z[X, Y]$ we can learn an auto-regressive model of these symbolic images using a pixal-CNN.

This yields a prior probability distribution $P_\Phi(z[X, Y])$ which provides a tighter upper bound on the rate.

We can then measure compression and distortion for test images. This is something GANs cannot do.

Multi-Layer Vector Quantized VAEs





Quantitative Evaluation

The VQ-VAE2 paper reports a classification accuracy score (CAS) for class-conditional image generation.

We generate image-class pairs from the generative model trained on the ImageNet training data.

We then train an image classifier from the generated pairs and measure its accuracy on the ImageNet test set.

	Top-1 Accuracy	Top-5 Accuracy
BigGAN deep	42.65	65.92
VQ-VAE	54.83	77.59
VQ-VAE after reconstructing	58.74	80.98
Real data	73.09	91.47

Direct Rate-Distortion Evaluation.

Rate-distortion metrics for image compression to discrete representations support unambiguous rate-distortion evaluation.

Rate-distortion metrics also allow one to explore the rate-distortion trade-off.

	Train NLL	Validation NLL	Train MSE	Validation MSE
Top prior	3.40	3.41	-	-
Bottom prior	3.45	3.45	-	-
VQ Decoder	-	-	0.0047	0.0050

Table 1: Train and validation negative log-likelihood (NLL) for top and bottom prior measured by encoding train and validation set resp., as well as Mean Squared Error for train and validation set. The small difference in both NLL and MSE suggests that neither the prior network nor the VQ-VAE overfit.

Image Compression

h_{top} $h_{\text{top}}, h_{\text{middle}}$ $h_{\text{top}}, h_{\text{middle}}, h_{\text{bottom}}$ Original

Figure 3: Reconstructions from a hierarchical VQ-VAE with three latent maps (top, middle, bottom). The rightmost image is the original. Each latent map adds extra detail to the reconstruction. These latent maps are approximately 3072x, 768x, 192x times smaller than the original image (respectively).

Vector Quantization (Emergent Symbols)

Vector quantization represents a distribution (or density) on vectors with a discrete set of embedded symbols.

world objects are discrete

Vector quantization optimizes a rate-distortion tradeoff for vector compression.

The VQ-VAE uses vector quantization to construct a discrete representation of images and hence a measurable image compression rate-distortion trade-off.

Symbols: A Better Learning Bias

Do the objects of reality fall into categories?

If so, shouldn't a learning architecture be designed to categorize?

Whole image symbols would yield emergent whole image classification.

Symbols: Improved Interpretability

Vector quantization shifts interpretation from linear threshold units to the emergent symbols.

This seems related to the use of t-SNE as a tool in interpretation.

Symbols: Unifying Vision and Language

Modern language models use word vectors.

Word vectors are embedded symbols.

Vector quantization also results in models based on embedded symbols.

Symbols: Addressing the "Forgetting" Problem

When we learn to ski we do not forget how to ride a bicycle.

However, when a model is trained on a first task, retraining on a second tasks degrades performance on the first (the model "forgets").

But embedded symbols can be task specific.

The embedding of a task-specific symbol will not change when training on a different task.

Symbols: Improved Transfer Learning.

Embedded symbols can be domain specific.

Separating domain-general parameters from domain-specific parameters may improve transfer between domains.

Unsupervised Machine Translation

We can treat the German sentence z as a latent variable in a probability model of a English sentence y.

$$\Phi^* = \underset{\Phi}{\operatorname{argmin}} \; E_{y,z\sim Q_\Phi(z|y)} \quad -\ln \frac{P_\Phi(z)P_\Phi(y|z)}{Q_\Phi(z|y)}$$

Here $P_\Phi(z)$ can be a trained language model for German and $P_\Phi(y|z)$ and $Q_\Phi(z|y)$ are translation models.

Unsupervised Machine Translation

$$\Phi^* = \underset{\Phi}{\operatorname{argmin}} \; E_{y,z \sim Q_\Phi(z|y)} \quad - \ln \frac{P_\Phi(z) P_\Phi(y|z)}{Q_\Phi(z|y)}$$

This will be subject to mode collapse. $P_\Phi(y|z)$ will collapse to an unconditional language model of English.

We can model the distribution of English directly better than we can model it with German as a latent variable.

Unsupervised Machine Translation

In practice we use "backtranslation"

$$\Phi^* = \underset{\Phi}{\operatorname{argmin}} \begin{array}{ll} & E_{y,z\sim Q_\Phi(z|y)} \quad -\ln P_\Phi(y|z) \\ + & E_{y,z\sim P_\Phi(z|y)} \quad -\ln Q_\Phi(y|z) \end{array}$$

Is there some theoretical justification for this?

Final Thought: Attention and Latent Variables

In machine translation attention is used to handle a latent alignment between the input sentence and the gold label translation.

In general, attention can be viewed as defining a probability distribution over a latent choice.

The precise relationship between attention and latent variables is unclear.

END