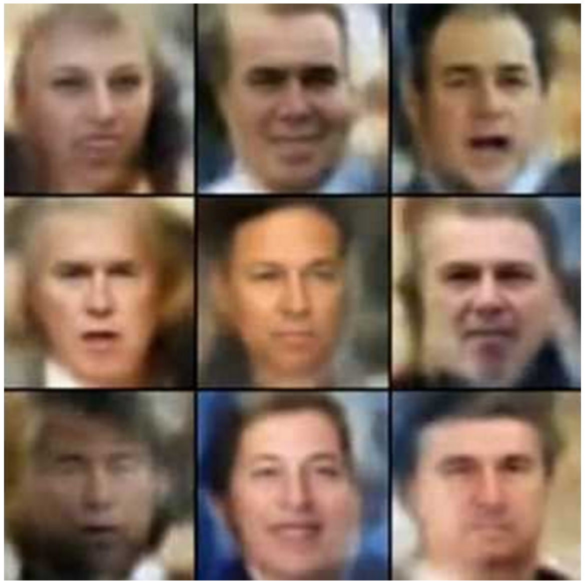
TTIC 31230, Fundamentals of Deep Learning

David McAllester, Autumn 2021

Vector Quantized Variational Autoencoders (VQ-VAEs)

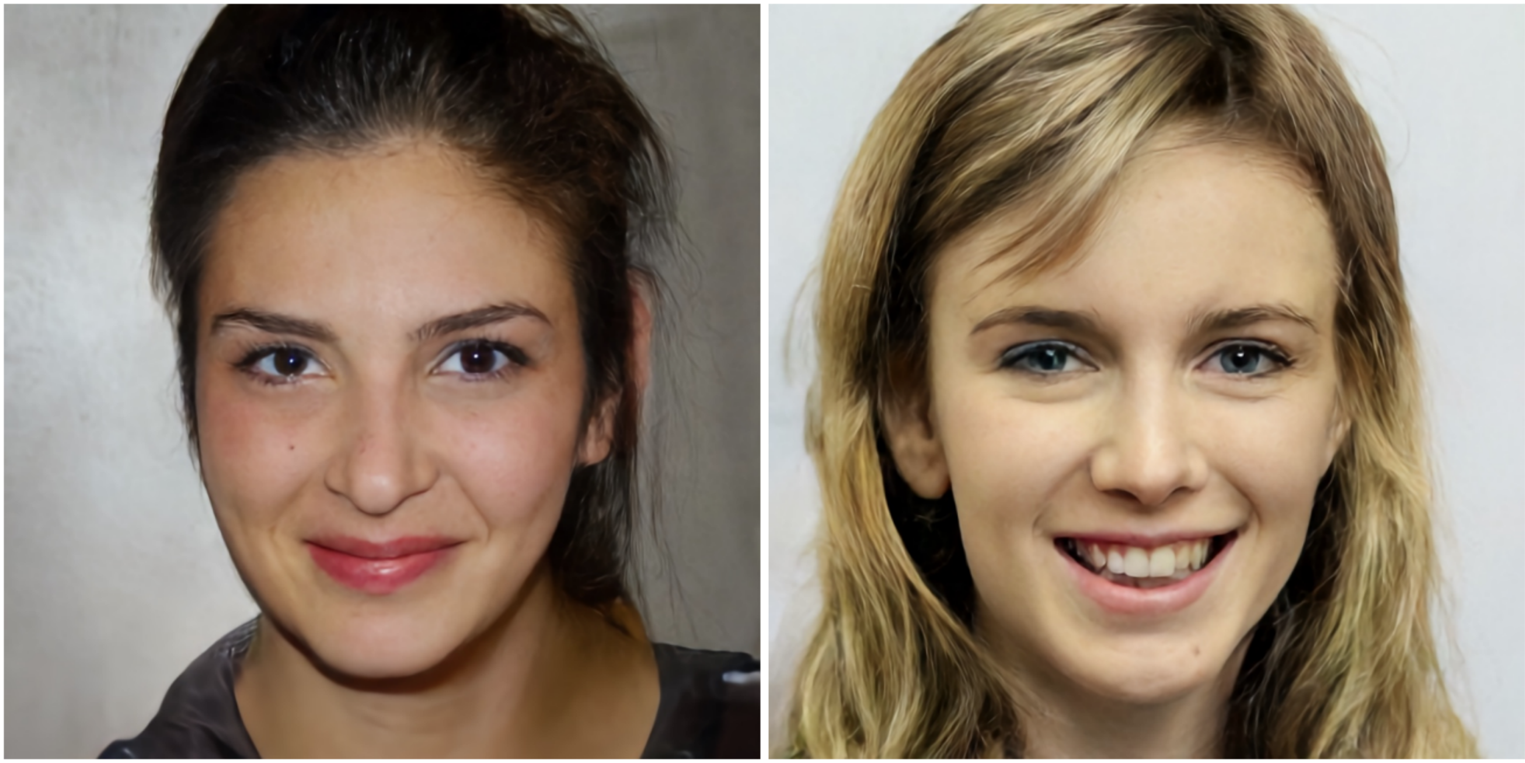
Gaussian VAEs for Faces 2014

We can sample faces from the VAE by sampling noise z from $p_\Phi(z)$ and then sampling an image y from $p_\Phi(y|z)$.



[Alec Radford]

VQ-VAEs 2019



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

VQ-VAEs 2019

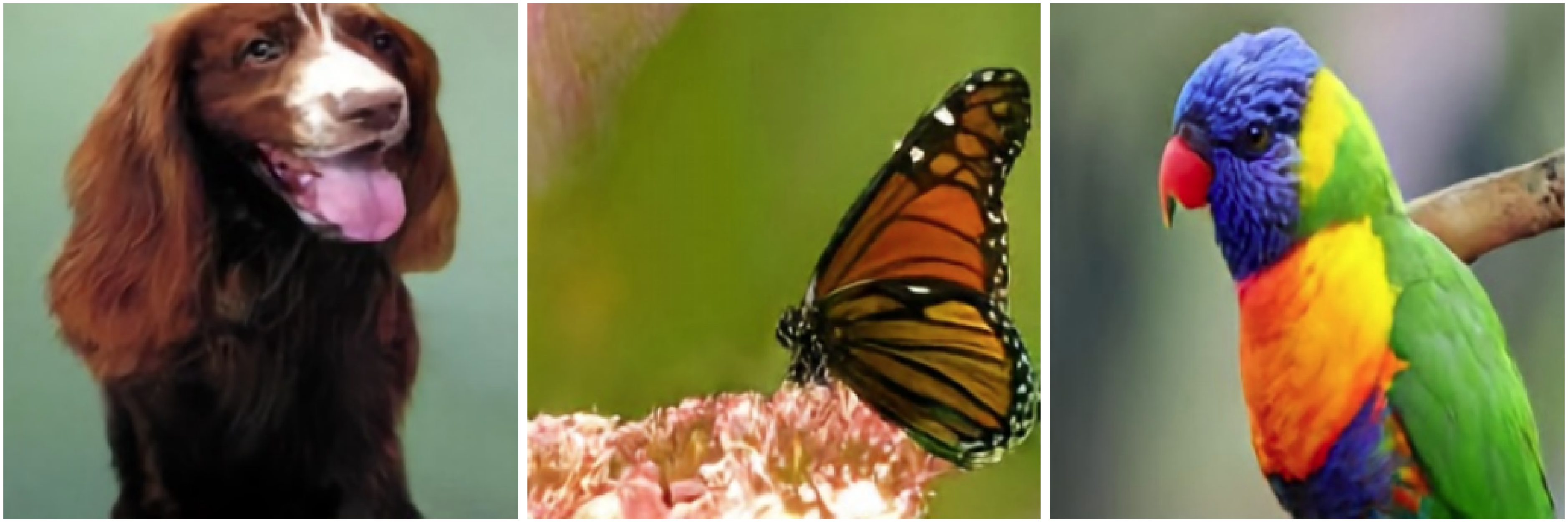
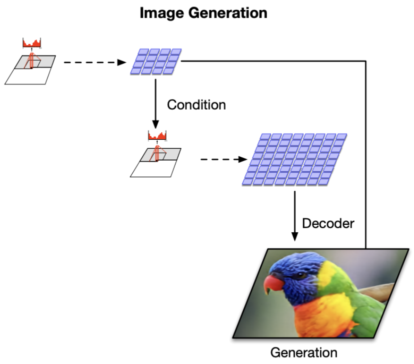


Figure 1: Class-conditional 256x256 image samples from a two-level model trained on ImageNet.

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

VQ-VAE-2 Model



z_1

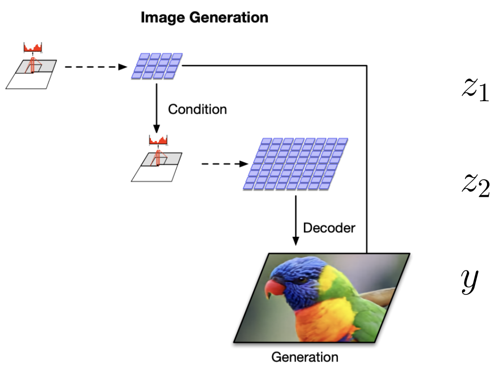
z_2

y

The probability of an image y is defined by the generator.

The generator is top-down and is similar to that of a progressive GAN.

VQ-VAE-2 Model



$$P_{\Phi,\Theta}(y) = \sum_{z_1\ldots,z_n} P_\Phi(z_1,\ldots,z_N)P_\Theta(y|z_1,\ldots,z_N).$$

Each z_i is a "symbolic image": an assignment of an embedded symbol to each pixel.

$P_{\Phi_{i+1}}(z_{i+1} \mid z_1, \ldots z_i)$ is auto-regressive, as in a language model.

VQ-VAE-2 Model

We first describe the case of just one layer.

We then describe a very general approach to multi-layer VAEs.

The Single-Layer VQ-VAE

Let s denote the image (we are using y for an image coordinate).

We have an encoder network, such as a CNN, which produces a layer with a vector at each pixel position.

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

Intuitively we cluster the vectors $L(x,y,I)$ using K-means clustering to produce cluster centers where $C[k,I]$ is the cluster center vector of cluster k.

VQ-VAE Model

We then replace each vector $L[x, y, I]$ by the nearest cluster center to give $\hat{L}[X, Y, I]$.

$$\begin{aligned} z[x, y] &= \underset{k}{\operatorname{argmin}} \; ||L[x, y, I] - C[k, I]|| \\ \hat{L}[x, y, I] &= C[z[x, y], I] \end{aligned}$$

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

VQ-VAE Model

Finally we decode $\hat{L}[X, Y, I]$ to get $\hat{s}_{\Theta}(z)$.

Two-Phase Optimization

Phase 1: Fix the prior $P_\Phi(z)$ at a simple (perhaps uniform) distribution and optimize the encoder $P_\Psi(z|s)$ and the decoder $P_\Theta(s|z)$.

$$\text{for} \quad \Theta^*, \Psi^* \;=\; \underset{\Theta,\Psi}{\mathrm{argmin}} \;\; E_{s\sim \mathrm{Pop}, z\sim P_\Psi(z|s)} \; \ln \frac{P_\Psi(z|s)}{P_\Phi(z)} - \ln P_\Theta(s|z)$$

$$\text{we get} \;\; \Theta^* \;=\; \underset{\Theta}{\mathrm{argmin}} \;\; E_{s\sim \mathrm{Pop}, z\sim P_{\Psi^*}(z|s)} \, [-\ln P_\Theta(s|z)]$$

Phase 2: Train the prior $P_\Phi(z)$ holding the encoder and decoder fixed.

$$\Phi^* \;=\; \underset{\Phi}{\mathrm{argmin}} \;\; E_{s\sim \mathrm{Pop}, z\sim P_{\Psi^*}(z|s)} \, [-\ln P_\Phi(z)]$$

Joint Training of the prior $P_\Phi(z)$ with the encoder $P_\Psi(z|s)$ and the decoder $P_\Theta(s|z)$ is not required.

One Layer VQ-VAE Training Phase 1

We fix the prior on z to be uniform. $P_\Phi(z)$ can then be ignored in phase 1.

The training objective in Phase 1 is then taken to be

$$\Psi^*, \Theta^* \;=\; \underset{\Psi,\Theta}{\mathrm{argmin}}\;\; E_{s\sim\mathrm{Pop},z\sim P_\Psi(z|s)}\left[\frac{\beta}{2}\,||L[X,Y,I]-\hat{L}[X,Y,I]||^2+(s-\hat{s}_\Theta(z))^2\right]$$

Handling Discrete Latents

$$
\begin{aligned}
z[x,y] &= \operatorname*{argmin}_k \; ||L[x,y,I] - C[k,I]|| \\
\hat{L}[x,y,I] &= C[z[x,y],I]
\end{aligned}
$$

Since $z[x,y]$ is discrete we have $z[x,y]$.grad $= 0$. They use "straight-through" gradients and "k-means" gradients.

$$
\begin{aligned}
L[x,y,I].\mathrm{grad} &= \hat{L}[x,y,I].\mathrm{grad} + \beta(L[x,y,I] - C[z[x,y],I]) \\
C[k,I].\mathrm{grad} &= \sum_{z[x,y]=k} \gamma(C[k,I] - L[x,y,I])
\end{aligned}
$$

One Layer VQ-VAE Training Phase 2

Finally we hold the encoder fixed and train the prior $P_\Phi(z)$ to be an auto-regressive model of the symbolic image $z[X, Y]$.

Progressive VAEs

We now consider a general VAE with a sequence of latent variables $z_1, \ldots, z_n$ where we want to model an observable variable y.

In the image VQ-VAE each z_i is a symbolic image.

A progressive VAE has the following structure on the prior and decoder.

$$P_\Phi(z_1, \ldots, z_N) = P_\Phi(z_1) \prod_{i=2}^{N} P_\Phi(z_i \mid z_{i-1})$$

$$P_\Theta(y|z_1, \ldots, z_N) = P_\Theta(y|z_N)$$

This copies the structure of a progressive GAN generator.

Progressive VAE Training Phase 1

We train the progressive VAE in two phases where the first phase trains a series of encoders and decoders.

$$P_\Psi(z_N|y),\ P_\Psi(z_{N-1}|z_N),\ \ldots, P_\Psi(z_1|z_2)$$
$$P_\Theta(y|z_n),\ P_\Theta(z_N|z_{N-1}),\ \ldots, P_\Theta(z_2|z_1)$$

The pair of $P_\Psi(z_N|y)$ and $P_\Theta(y|z_N)$ is trained under a fixed simple prior on z_N.

Each pair of $P_\Psi(z_{i-1}|z_i)$ and $P_\Theta(z_i|z_{i-1})$ is trained under a fixed simple prior on z_{i-1}.

This defines an encoder $P_\Psi(z_1, \ldots, z_N|y)$ and an appropriately strong decoder $P_\Theta(y|z_N)$.

Progressive VAE Training Phase 2

For each y we how have a sequence of latent variables $z_1, \ldots, z_N$.

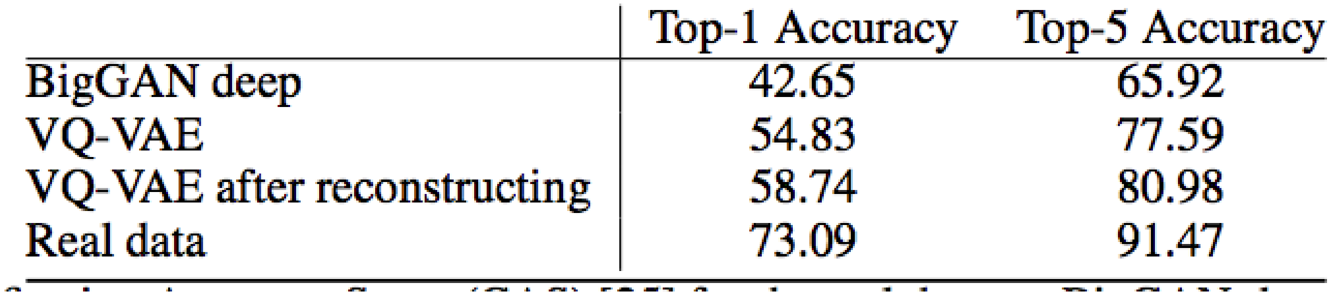
This allows training of a strong prior $P_\Phi(z_1)$ and $P(z_{i+1}|z_i)$.

Quantitative Evaluation

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

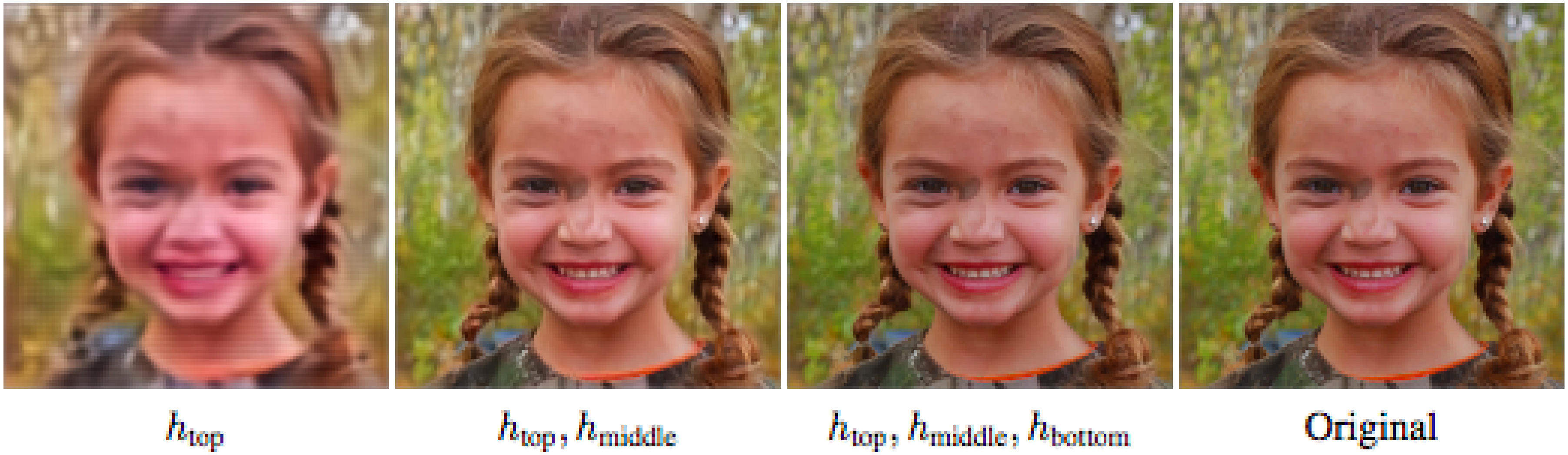
We generate image-class pairs from the generative model trained on the ImageNet training data.

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

Image Compression



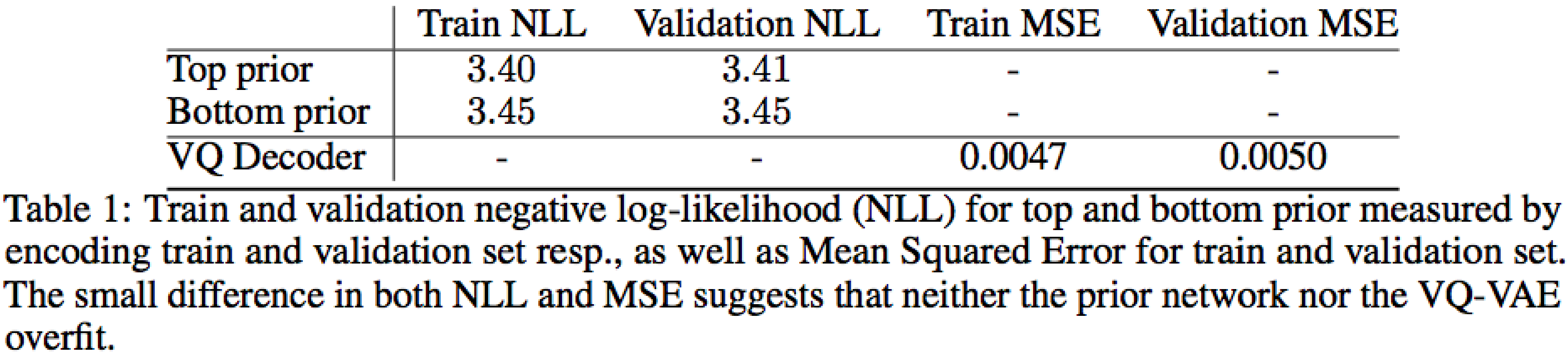
h_{top} $h_{\text{top}}, h_{\text{middle}}$ $h_{\text{top}}, h_{\text{middle}}, h_{\text{bottom}}$ Original

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).

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

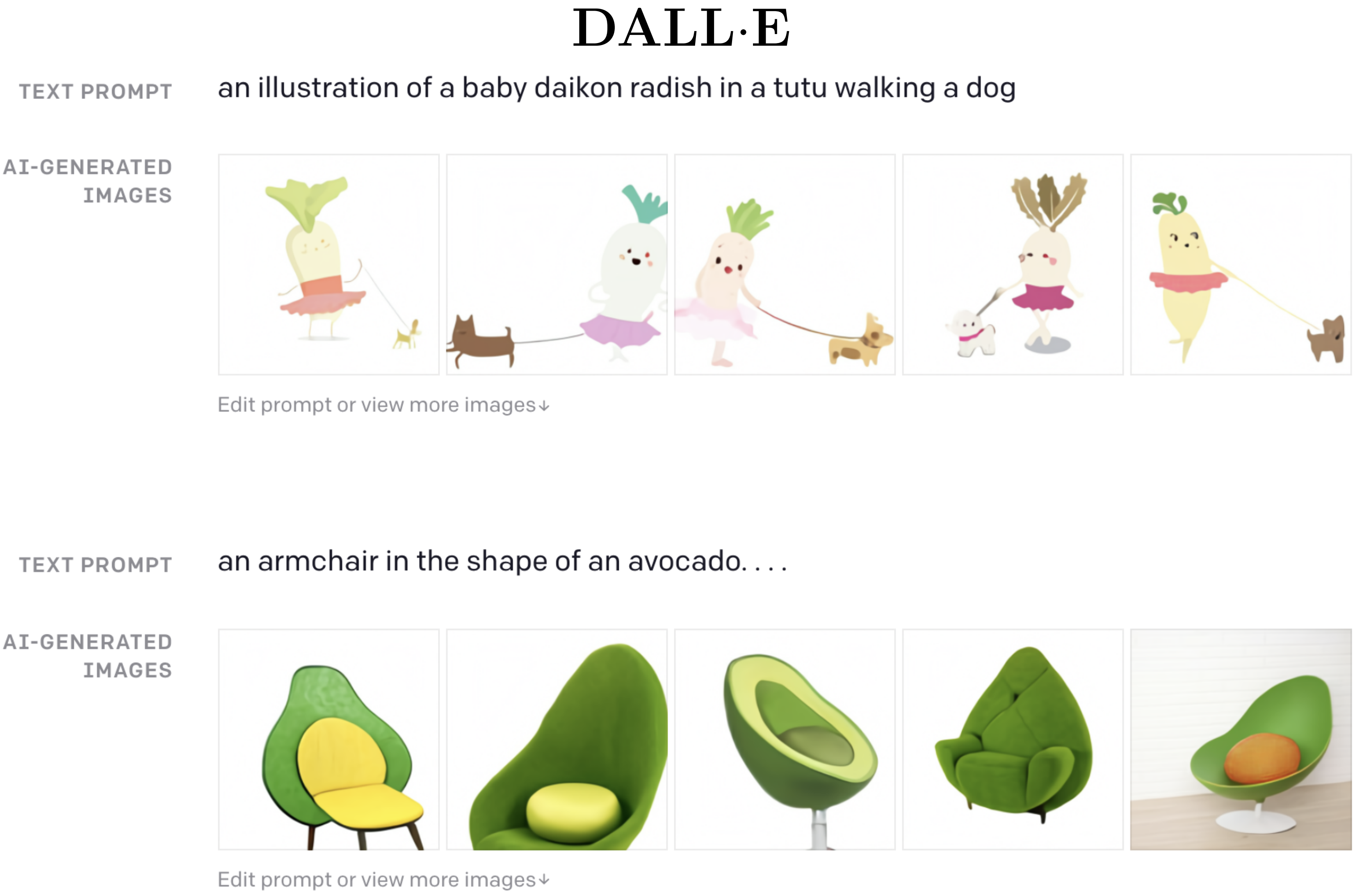
Table 1: Train and validation negative log-likelihood (NLL) for top and bottom prior measured by encoding train and validation set resp., as well as Mean Squared Error for train and validation set. The small difference in both NLL and MSE suggests that neither the prior network nor the VQ-VAE overfit.

DALL·E: A Text-Conditional Image dVAE

DALL·E is a text-conditional VQ-VAE model of images.

The Vector quantization is done independent of the text. However, the model of the probability distribution of the symbolic image $z[x, y]$ is conditioned on text.

Ramesh et al. 2021



DALL·E

TEXT PROMPT

an illustration of a baby daikon radish in a tutu walking a dog

AI-GENERATED IMAGES

Edit prompt or view more images↓

TEXT PROMPT

an armchair in the shape of an avocado. . . .

AI-GENERATED IMAGES

Edit prompt or view more images↓

END