TTIC 31230, Fundamentals of Deep Learning

David McAllester, Winter 2020

Vector Quantized Variational Autoencoders (VQ-VAEs)

Gaussian VAEs 2013

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

[Alec Radford]

VQ-VAEs 2019

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

VQ-VAEs 2019

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.

$$\begin{aligned}
L[X,Y,I] &= \mathrm{Enc}_\Phi(s) \\
\\
z[x,y] &= \operatorname*{argmin}_k \; ||L[x,y,I] - C[k,I]|| \\
\hat{L}[x,y,I] &= C[z[x,y],I] \\
\\
\hat{s} &= \mathrm{Dec}_\Phi(\hat{L}[X,Y,I])
\end{aligned}$$

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

VQ-VAE as an RDA

We will interpret the VQ-VAE as an RDA.

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

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 $\hat{L}[X,Y,I]$.

$$\Phi^* = \operatorname*{argmin}_{\Phi} \; E_s \; \beta||L[X,Y,I] - \hat{L}[X,Y,I]||^2 + ||s - \hat{s}||^2$$

This is a two-level rate-distortion auto-encoder where the rate can be no larger than $XY \ln K$.

Parameter-Specific Learning Rates

$$||L[X,Y,I] - \hat{L}[X,Y,I]||^2 = \sum_{x,y} ||L[x,y,I] - C[z[x,y],I]||^2$$

For the gradient of this they use

```
  for $x, y$  $L[x,y,I]$.grad += $2\beta(L[x,y,I] - C[z[x,y],I])$
for $x, y$  $C[z[x,y],I]$.grad += $2(C[z[x,y],I] - L[x,y,I])$
```

This gives a parameter-specific learning rate for $C[K,I]$.

Parameter-specific learning rates do not change the stationary points (the points where the gradients are zero).

The Relationship to K-means

$$\text{for } x, y \;\; C[z[x,y],I].\mathrm{grad} \texttt{ += } 2(C[z[x,y],I] - L[x,y,I])$$

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).

Straight Through Gradients

The latent variables are discrete so some approximation to SGD must be used.

They use “straight-through” gradients.

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

This assumes low distortion between $L[X, Y, I]$ and $\hat{L}[X, Y, I]$.

A Suggested Modification

The parameter β is paying two roles

- It controls the relative weight of the two distortion losses.

- It controls the learning rate adjustment for the codebook.

Shouldn't we have separate parameters for these two roles?

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 pixel-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.

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.

Image Compression

Vector Quantization (Emergent Symbols)

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

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.

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

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

In practice we use “backtranslation”

$$\Phi^* = \operatorname*{argmin}_{\Phi} \begin{array}{l} E_{y\sim \mathrm{Pop}_y,\; z\sim Q_\Phi(z|y)} \quad -\ln P_\Phi(y|z) \\ \\ +\; E_{z\sim \mathrm{Pop}_z,\; y\sim P_\Phi(y|z)} \quad -\ln Q_\Phi(z|y) \end{array}$$

END