TTIC 31230, Fundamentals of Deep Learning

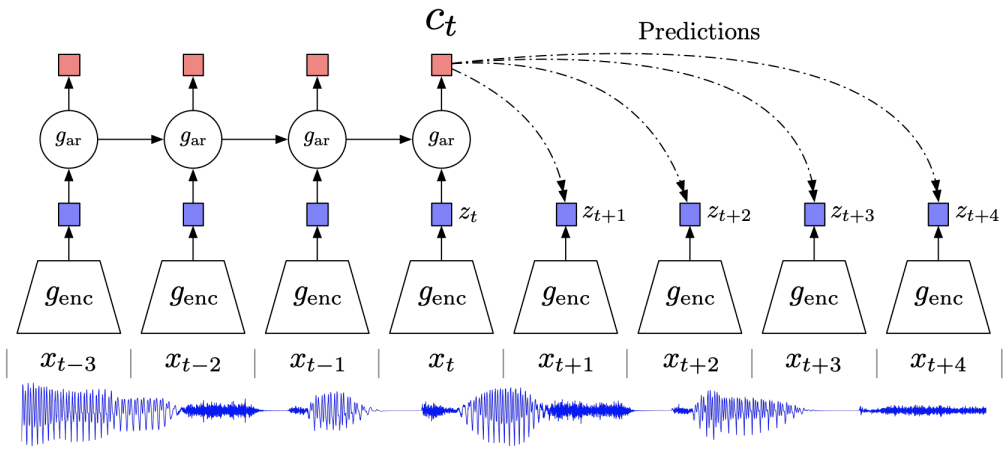
David McAllester, Autumn 2022

Contrastive Coding

Tokenization

Autoregressive Image and Voice Models

Contrastive Coding for Speech

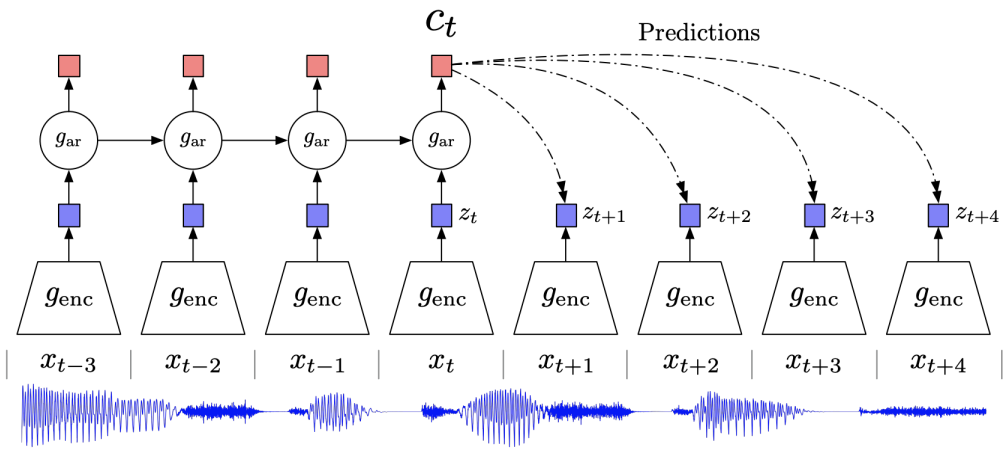


van den Oord, Li and Vinyals,

Representation Learning with Contrastive Predictive Coding, 2018

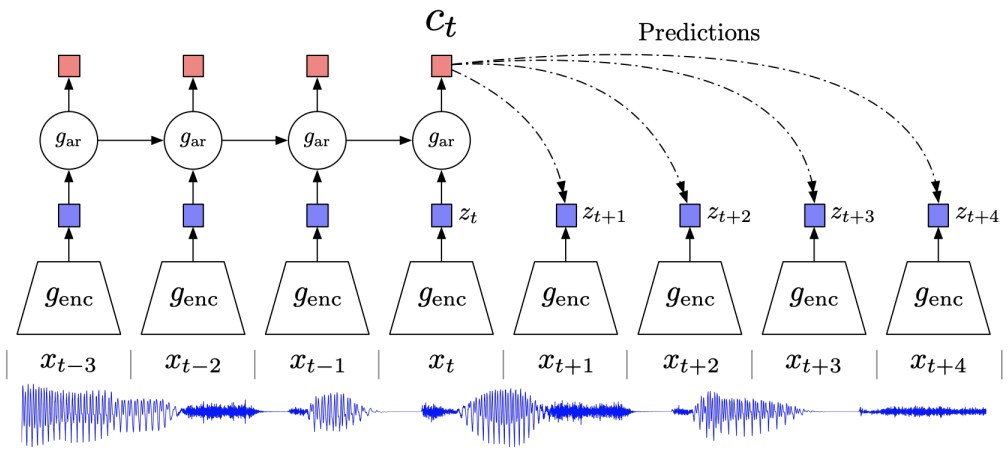
What should we abstract from the past that is relevant to the future?

Contrastive Coding for Speech



Unlike VAEs, contrastive coding is about **capturing mutual information**. Intuitively we want to **separate signal from noise** and avoid modeling noise.

Contrastive Coding for Speech



We abstract this problem to that of capturing the mutual information between any two arbitrary random variables x and y.

Contrastive Coding

We consider a population distribution on pairs (x, y).

For example:

- x might be an image and y might be the text of a caption for image x (CLIP).
- x might be an video frame and y video frame a second later.
- x might be a window of a sound wave and y a later window (Wav2Vec).
- $x = f(z)$ and $y = f(z)$ where f and g are transformation functions on an image z such as translation, rotation, color shift, or cropping. (augmentation) of x. (SimCLR)

Contrastive Coding

We draw pairs $(x_1, y_1), \ldots (x_B, y_B)$ from the population. We then select b uniformly from 1 to B and construct the tuple $(x_b, y_1, \ldots, y_B, b)$.

We then train a model to predict b.

$$\text{enc}_x^*, \text{enc}_y^* = \underset{\text{enc}_x, \text{enc}_y}{\text{argmin}} \; E_{(x, y_1, \ldots, y_B, b)} \left[-\ln P_{\text{enc}_x, \text{enc}_y}(b | x, y_1, \ldots, y_B) \right]$$

$$P_{\text{enc}_x, \text{enc}_y}(b | x, y_1, \ldots, y_B) = \underset{b}{\text{softmax}} \; \text{enc}_x(x)^\top \text{enc}_y(y_b)$$

The Contrastive Coding Theorem

For any distribution on pairs (x, y), with contrastive probabilities computed by

$$P(b|x, y_1, \ldots, y_B) = \underset{b}{\text{softmax}}\ s(x, y_b)$$

we have

$$I(x, y) \geq \ln B - \quad E_{(x,y_1,\ldots,y_B,b)}\left[-\ln P(b|(x, y_1, \ldots, y_B))\right]$$

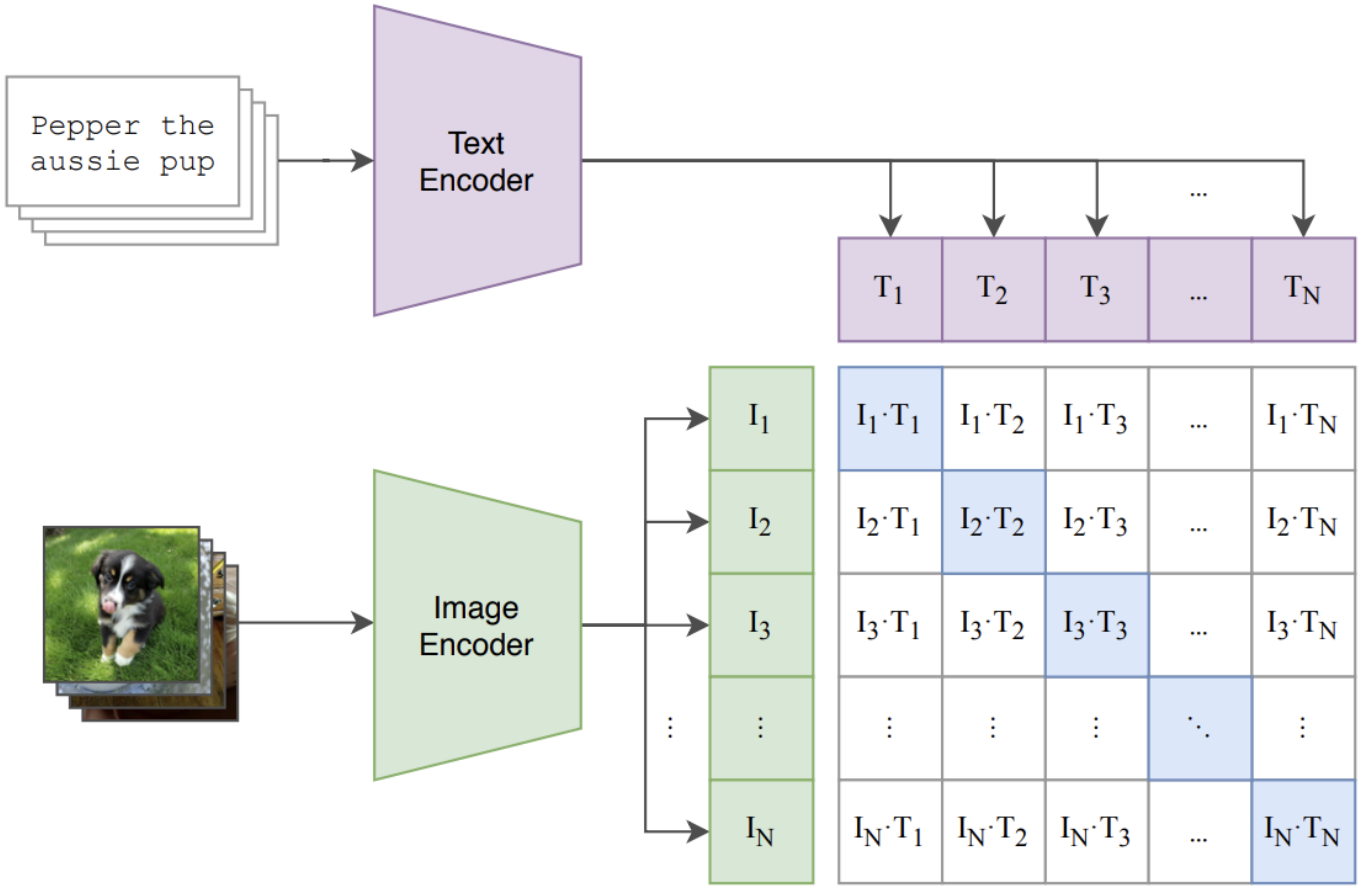
Chen et al., On Variational Bounds of Mutual Information, May 2019.

CLIP, January 2021, OpenAI

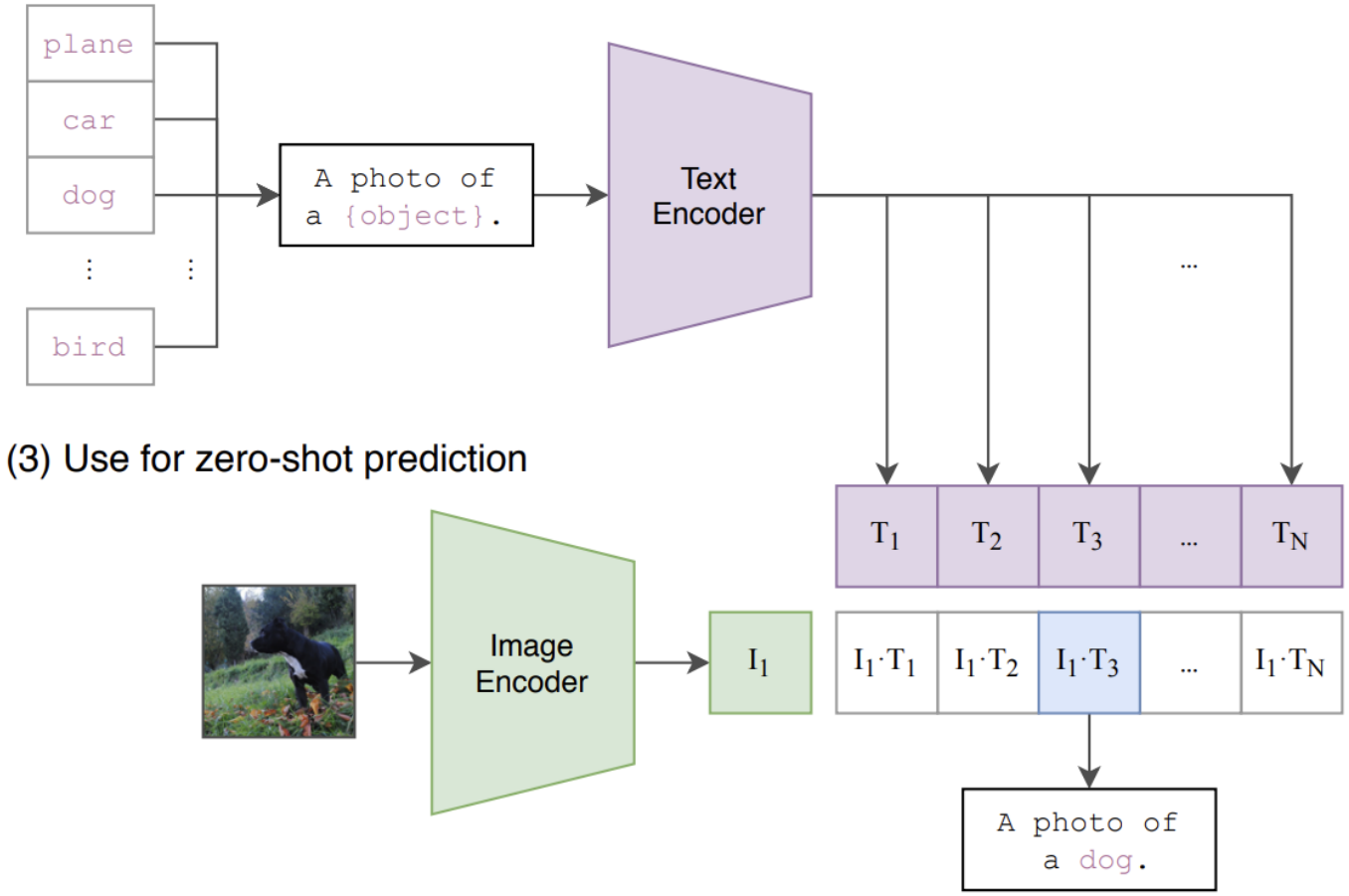
CLIP: Contrastive Language-Image Pre-training.

Trained on images and associated text (such as image captions or hypertext links to images) CLIP computes embeddings of text and embeddings of images ("co-embeddings") trained to capture the mutual information between the two.

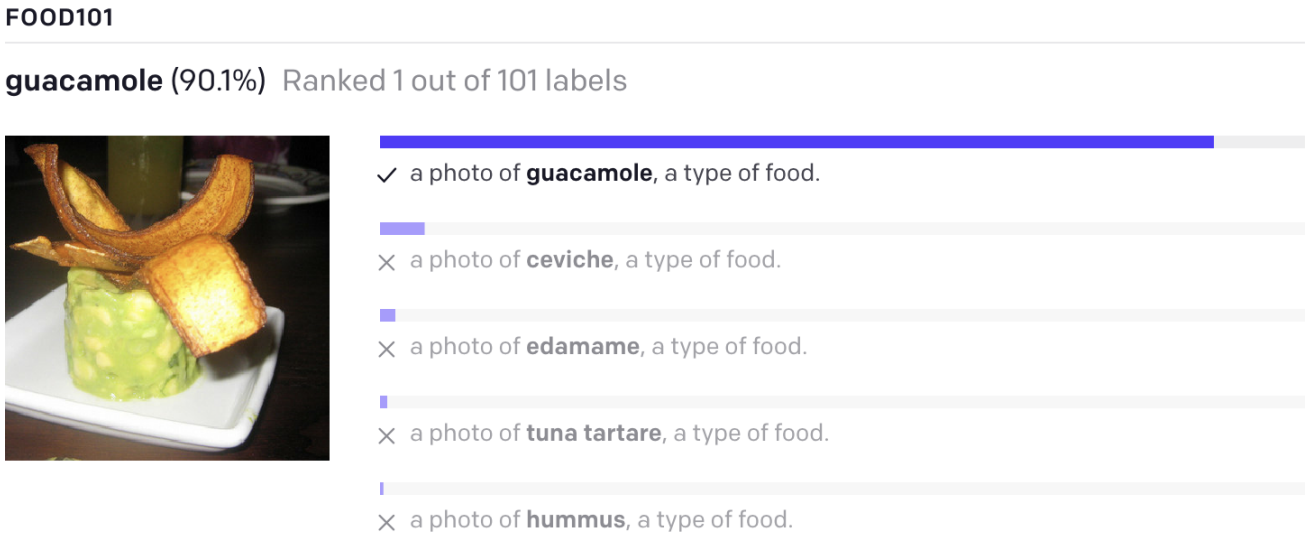
CLIP Constrastive Coding



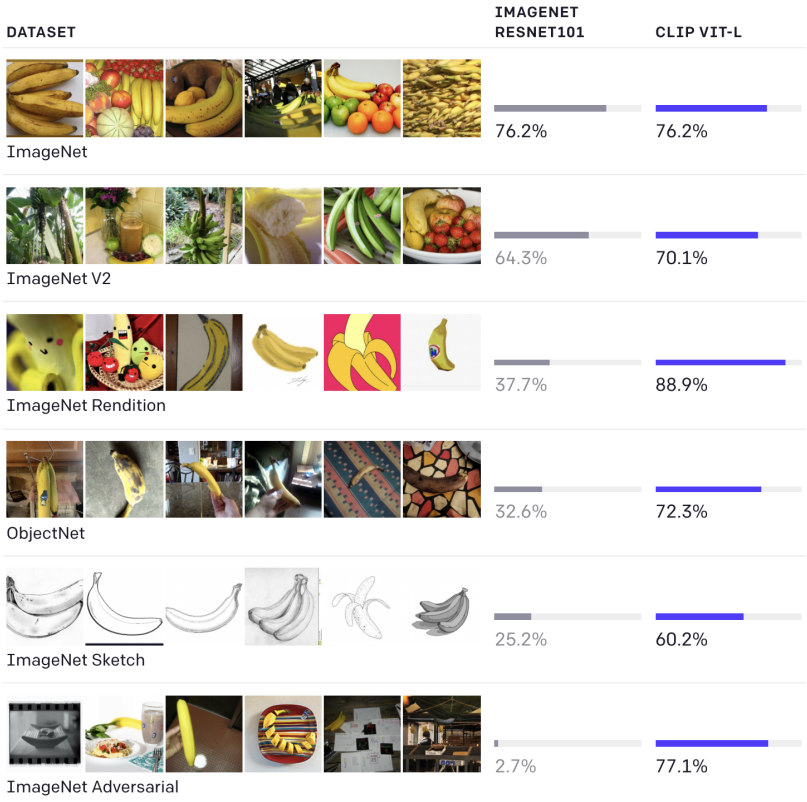
CLIP Image Classification



Zero-Shot Image Classification



Zero-Shot Image Classification



Although both models have the same accuracy on the ImageNet test set, CLIP's performance is much more representative of how it will fare on datasets that measure accuracy in different, non-ImageNet settings. For instance, ObjectNet checks a model's ability to recognize objects in many different poses and with many different backgrounds inside homes while ImageNet Rendition and ImageNet Sketch check a model's ability to recognize more abstract depictions of objects.

A Weakness of Contrastive Coding

$$I(x, y) \geq \ln B - \quad E_{(x, y_1, \ldots, y_B, b)} [-\ln P(b|(x, y_1, \ldots, y_B)]$$

The discrimination problem may be too easy.

The guarantee can never be stronger than $\ln B$ where B is the batch size.

Suppose we have 100 bits of mutual information as seem plausible for translation pairs.

Addresses the Weakness with Large Batch Size

$$I(x, y) \geq \ln B - \quad E_{(x,y_1,\ldots,y_B,b)}[-\ln P(b|(x, y_1, \ldots, y_B)]$$

For CLIP the batch size $B = 2^{15}$ so we can potentially guarantee 15 bits of mutual information.

Tishby's Information Bottleneck

The Information Bottleneck Method
Tishby, Pereira and Bialeck, 1999

Design $P_{\text{enc}}(z|x)$ with the following objective.

$$\text{enc}^* = \underset{\text{enc}}{\text{argmin}} \; I(z,x) - \beta I(z,y)$$

This does not restric $H(z)$.

Balancing Mutual Information with Encoder Enrtopy

$$
\begin{aligned}
\text{enc}^* &= \operatorname*{argmin}_{\text{enc}} H(\text{enc}(x)) - \beta I(\text{enc}(x), y) \\
&= \operatorname*{argmin}_{\text{enc}} \; H(\text{enc}(x)) - \beta(H(y) - H(y|\text{enc}(x))) \\
&= \operatorname*{argmin}_{\text{enc}} \; H(\text{enc}(x)) + \beta H(y|\text{enc}(x))
\end{aligned}
$$

Balancing Mutual Information with Encoder Enrtopy

$$H(\text{enc}(x)) + \beta H(y|\text{enc}(x))$$

$$= \; E_{(x,y)\sim\text{Pop}}[\, -\ln P(\text{enc}(x)) + \beta(-\ln P(y|\text{enc}(x))\,]$$

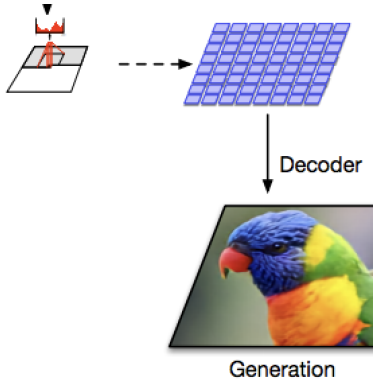
$$\leq \; E_{(x,y)\sim\text{Pop}}[\, -\ln P_{\text{pri}}(\text{enc}(x)) + \beta(-\ln P_{\text{dec}}(y|\text{enc}(x))\,]$$

$$\text{enc}^*, \text{dec}^*, \text{pri}^* = \underset{\text{enc,dec,pri}}{\text{argmin}} \; E_{(x,y)\sim\text{Pop}}[\, -\ln P_{\text{pri}}(\text{enc}(x)) + \beta(-\ln P_{\text{dec}}(y|\text{enc}(x))\,]$$

Autoregressive Image and Voice Modeling

Strong VAE image modeling was first achieved with autoregressive token modeling.

van den Oord, Vinyals and Kavukcuoglu,
Neural Discrete Representation Learning, **2017**



Token Transformer

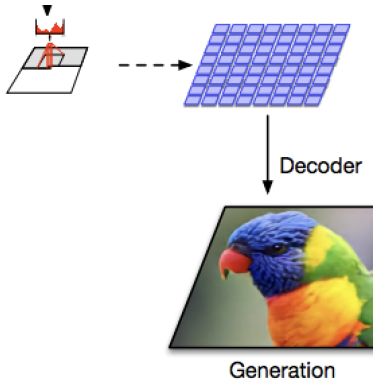
Image-to-Tokens, Tokens-to-Image

VQ-VAE-2, June 2019

Generating Diverse High-Fidelity Images with VQ-VAE-2

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

VAE Tokenization of Images and Voice



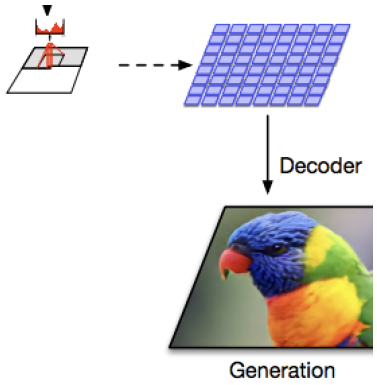
Token Transformer

Image-to-Tokens, Tokens-to-Image

Let y range over a population (such as images or sound waves).

Assume that a given y is encoded as a tensor denoted $z_{\text{enc},p}(y)$ where p is a "position in y" (a pixel in an image tensor or time window in a sound tensor) and $z_{\text{enc},p}(y) \in R^d$ is a vector.

Vector Quantization (Tokenization)



Token Transformer

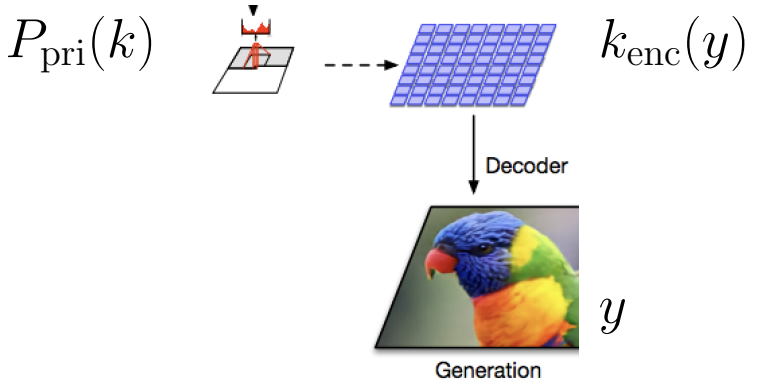
Image-to-Tokens, Tokens-to-Image

Assume a finite set of K "tokens" where token k has an embedding vector $e(k) \in R^d$.

Define $k_{\text{enc},p}(y)$ by

$$k_{\text{enc},p}(y) = \underset{k}{\text{argmin}} \ \frac{1}{2}||z_{\text{enc},p}(y) - e(k)||^2$$

Reconstruction Loss

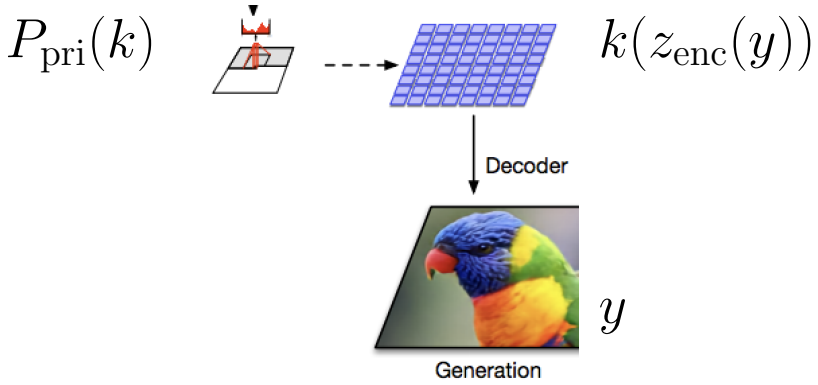


We now have a VAE where the tensor $k_{\text{enc}}(y)$ is the latent variable.

The encoder and decoder are trained jointly.

The prior is a transformer trained after the encoder and decoder are fully trained.

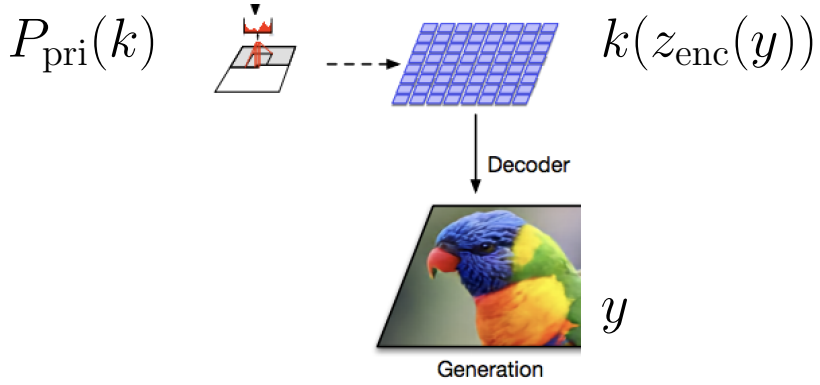
Training the Encoder and Decoder



Taking $P_{\text{dec}}(y|k)$ to be $\mathcal{N}(\hat{y}_{\text{dec}}(k), I)$ we get an L_2 reconstruction loss.

$$\mathcal{L}_{\text{Rec}} = \frac{1}{2}|| \; y - \hat{y}_{\text{dec}}(e(k(z_{\text{enc}}(y)))) \; ||^2$$

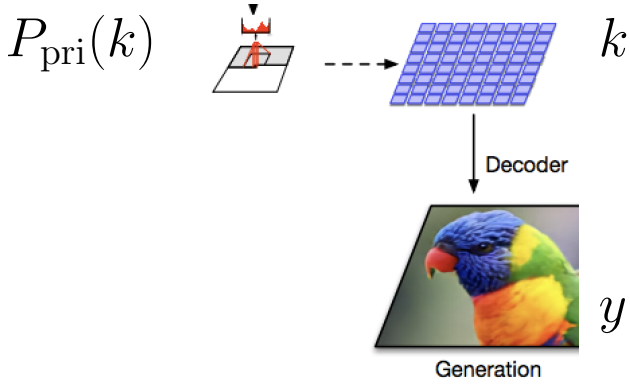
Training the Encoder and Decoder



$$\mathcal{L}_{\text{Rec}} = \frac{1}{2}|| \; y - \hat{y}_{\text{dec}}(e(k(z_{\text{enc}}(y)))) \; ||^2$$

Because the tokens are discrete we do not get any gradient on $z_{\text{enc}}(y)$.

Straight-Through Gradients



$$\mathcal{L}_{\text{Rec}} = \frac{1}{2}|| \; y - \hat{y}_{\text{dec}}(e(k(z_{\text{enc}}(y)))) \; ||^2$$

$$z_{\text{enc},p}(y).\text{grad} = \nabla_{e(k(z_{\text{enc},p}(y)))}\mathcal{L}_{\text{Rec}}$$

K-Means Gradients

We train $z_{\text{enc}}(y)$ and the token embeddings $e(k)$.

$$\mathcal{L}_{\text{Rec}} = \frac{1}{2}|| \; y - \hat{y}_{\text{dec}}(e(k(z_{\text{enc}}(y)))) \; ||^2$$

$$z_{\text{enc},p}(y).\text{grad} = \nabla_{e(k(z_{\text{enc},p}(y)))}\mathcal{L}_{\text{Rec}}$$

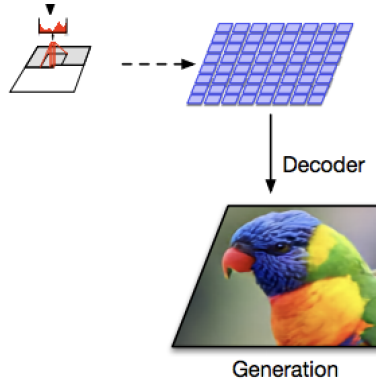
$$\mathcal{L}_{\text{KM}} = \frac{1}{2}||z_{\text{enc},p}(y) - e(k(z_{\text{enc},p}))||^2$$

$$e(k(z_{\text{enc},p}(y))).\text{grad} \; \texttt{+=} \; \beta\nabla_{e(k(z_{\text{enc},p}(y)))}\mathcal{L}_{\text{KM}}$$

β is a hyper-parameter that adjust the relative learning rates.

Transformer Training

Finally we hold the encoder fixed and train the prior $P_{\text{pri}}(z)$ to be an auto-regressive model of the symbolic image $k_{\text{enc}}(s)[X, Y]$.



Tokenization and Gaussian Mixture Models (GMMs)

Consider modeling $P(y|x)$ with $y \in R^d$

A Gaussian model has the form

$$y = \hat{y}(x) + \sigma\epsilon, \quad \epsilon \sim \mathcal{N}(0, I)$$

$$\hat{y} = \underset{\hat{y}}{\text{argmin}} \; E_{x,y} \, ||\hat{y}(x) - y||^2$$

Tokenization and Gaussian Mixture Models (GMMs)

Now consider a tokenizing decoder

$$y = E_{k \sim P_{\text{dec}}(k|x)}[\, e(k) + \sigma\epsilon \,], \quad \epsilon \sim \mathcal{N}(0, I)$$

We get that $P_{\text{dec}}(y|x)$ is a Gaussian mixture model (GMM).

GMMs are significantly more expressive than single Gaussians.

Wav2Vec 2.0, June 2020, Facebook

Trained on 53k hours of unlabeled audio (no text) they convert speech to a sequence of discrete quantized vectors they call "pseudo-text units".

By training on only one hour of human-transcribed audio, and using the Wav2Vec transcription into pseudo-text, they outperform the previous state of the art in word error rate for 100 hours of human-transcribed text.

DALLE-1, January 2021

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↓

GLSM, February 2021, Facebook

Generative Spoken Language Model (GSLM)

They then train a generative model of the sequences of pseudo-text units learned from unlabeled audio.

This model can continue speech from a speech prompt in much the same way that GPT-3 continues text from a text prompt.

Semantic and grammatical structure in a "unit language model" is recovered from speech alone.

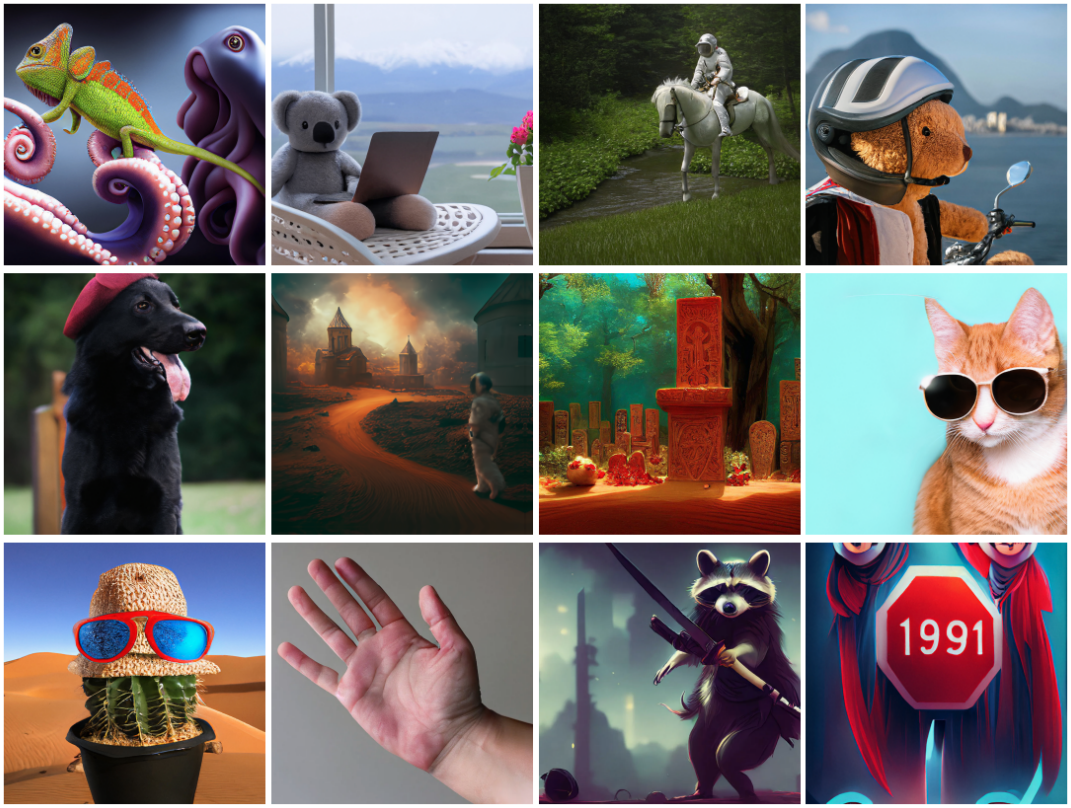
Parti, June 2022

Scaling Autoregressive Models for Content-Rich Text-to-Image Generation
Yu et al.

CM3Leon, September 2023

Scaling Autoregressive Multi-Modal Models: Pretraining and Instruction Tuning

Yu et al.



Voice-Text Language Model (VoxtLM), September 2023

This is similar to CM3Leon but for voice and text rather than images and text.

Voice is tokenized and then a transformer is used to model sequences that alternate voice and text.

END