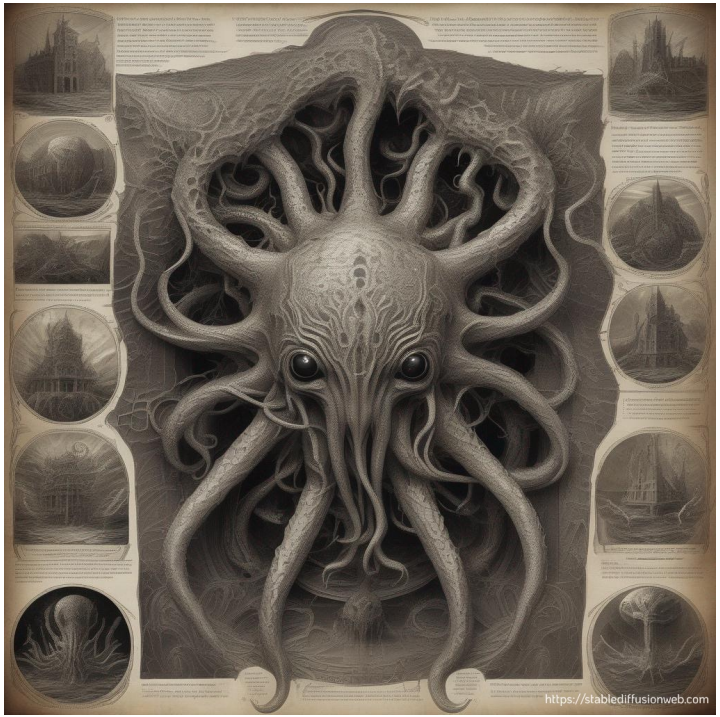
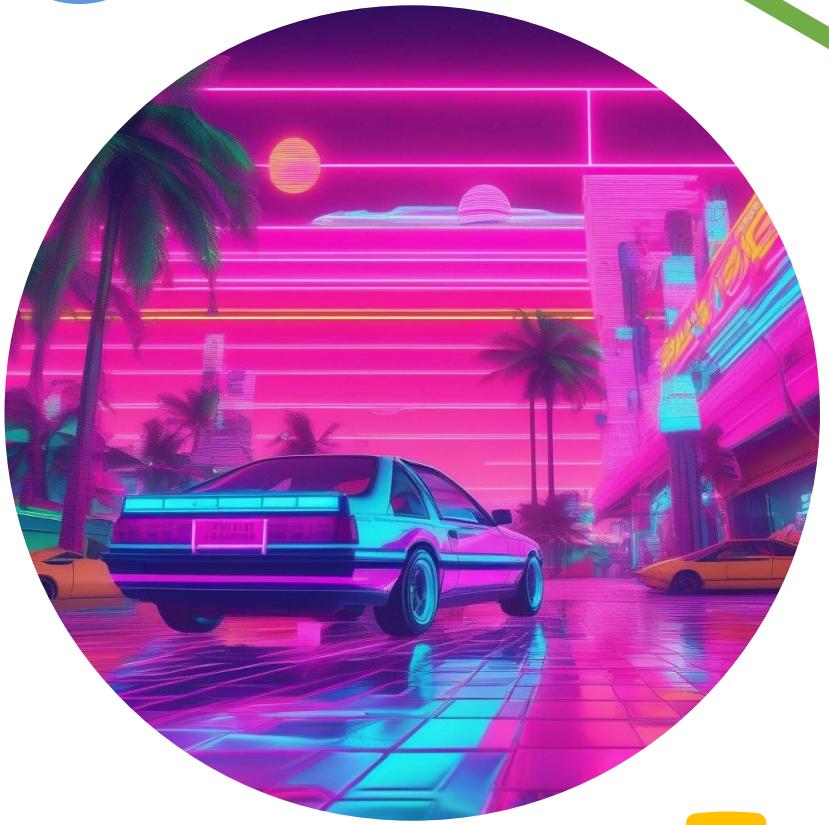
Unblurring from nothing: How do diffusion models work?

Edmund Dable-Heath

Outline

- What is diffusion? Some intuition about what we're trying to do.
- Sampling methods
- U-Nets
- How do we train these models?
- And what if we want to ask it for a particular pretty picture?
- Limitations and ethical quandaries.

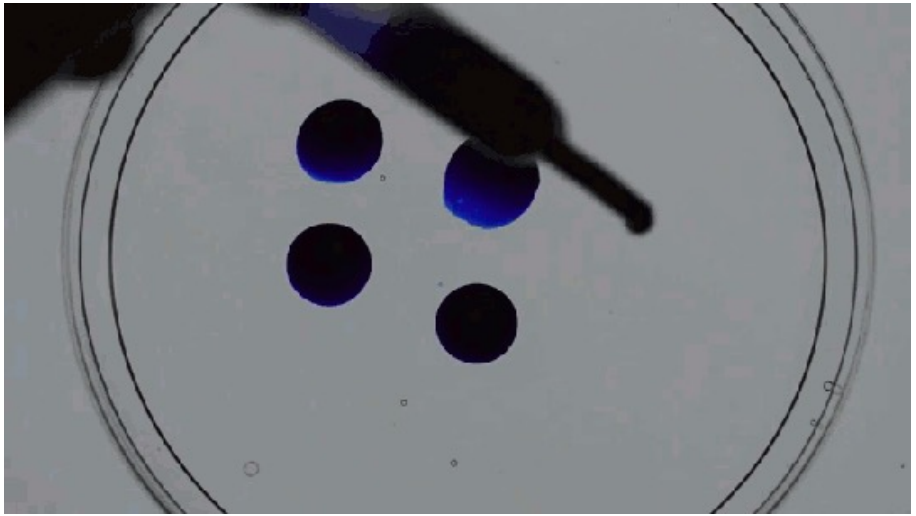




What is diffusion anyway?

Diffusion in the physical world

- Time-dependent random process of something moving from an area of high concentration to low concentration.
- How do we model this?
- If we know exactly how we got to a diffuse state, can we get back to a concentrated one?



$$\mathrm{P}(X_{t+1} = x \mid X_1 = x_1, X_2 = x_2, \ldots, X_t = x_t)$$
$$=$$
$$\mathrm{P}(X_{t+1} = x \mid \mathrm{X}_t = x_t)$$

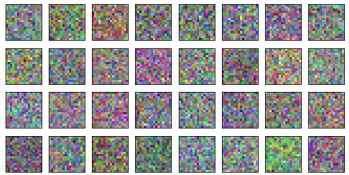
Diffusing Images

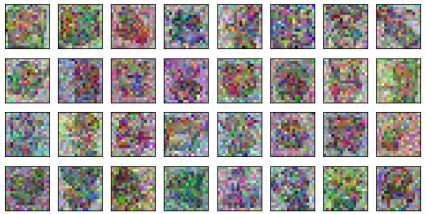
- Consider an image as a concentrated collection of coherent information within pixel space.

- Then we can think of adding noise – or blurring the image – as diffusion.

- Taking a trip through the image library of babel.

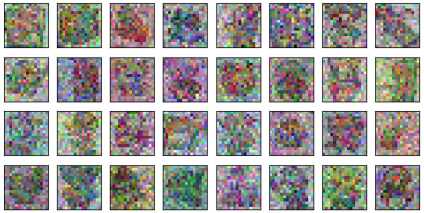
So what do these models look like?



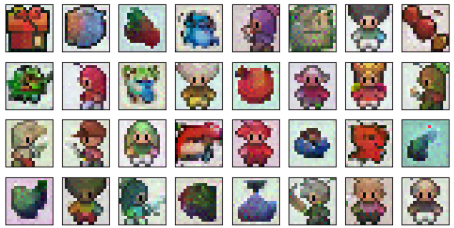


x_T → Neural Net → Remove Noise → x_{T-1}

x_{T-1} → Neural Net → Remove Noise → x_{T-2}



Repeat → x_0



Noise-cancelling sampling

Denoising Diffusion Probabilistic Models

- General idea: learn the information about the scheduling of the variance of the distributions adding noise throughout the process.

- Use this information to run the process backwards from a random sample.

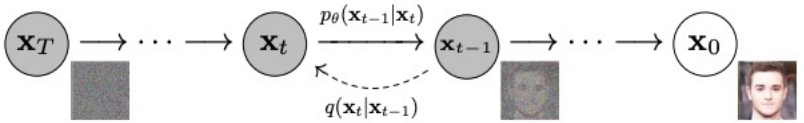


Figure 2: The directed graphical model considered in this work.

Variance schedule: $\beta_1, \dots, \beta_T$

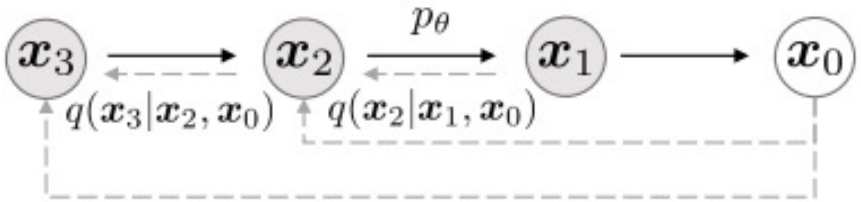
$$q(x_t \mid x_{t-1}) \coloneqq N(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I)$$

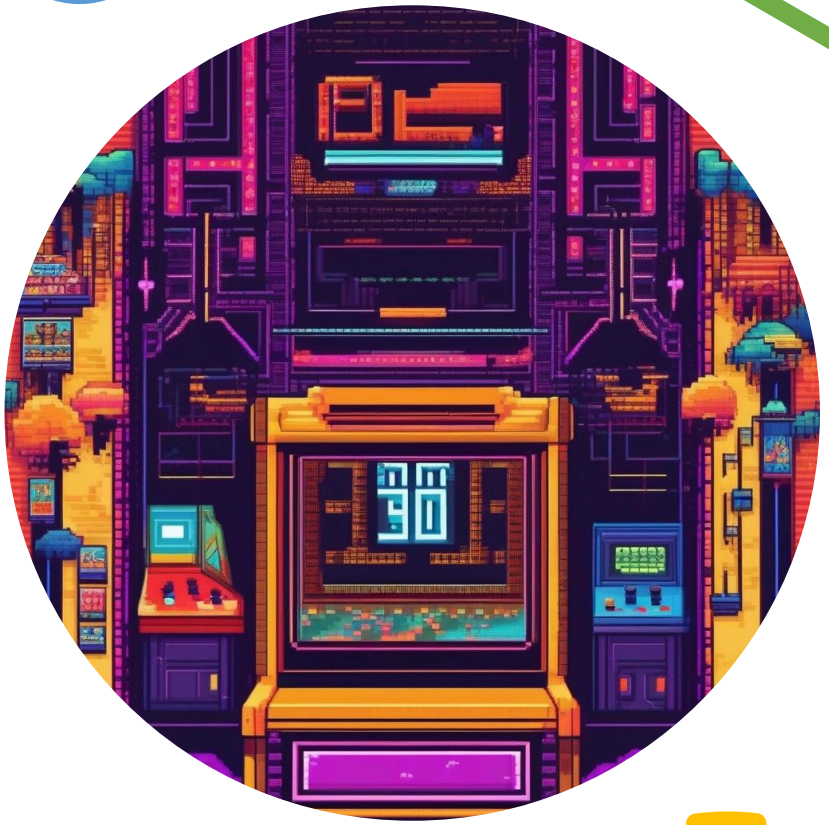
Algorithm 2 Sampling

1: $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
2: **for** $t = T, \dots, 1$ **do**
3: $\quad \mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
4: $\quad \mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\right) + \sigma_t \mathbf{z}$
5: **end for**
6: **return** $\mathbf{x}_0$

Denoising Diffusion Implicit Models

- Instead of learning a random process, this generalizes to non-Markovian processes from which you sample an entire process.
- This allows you to be much faster as it ends up being deterministic.





U-Nets all the way down (and then back up again)

What are U-Nets?

- A U-shaped convolutional neural network.
- Initially created for image segmentation.
- Down-samples images to distill information, before up sampling (with skip connections) to return to original resolution.
- For diffusion models this learns the noise distribution on an image – i.e. ‘segmenting’ the noise from the image.

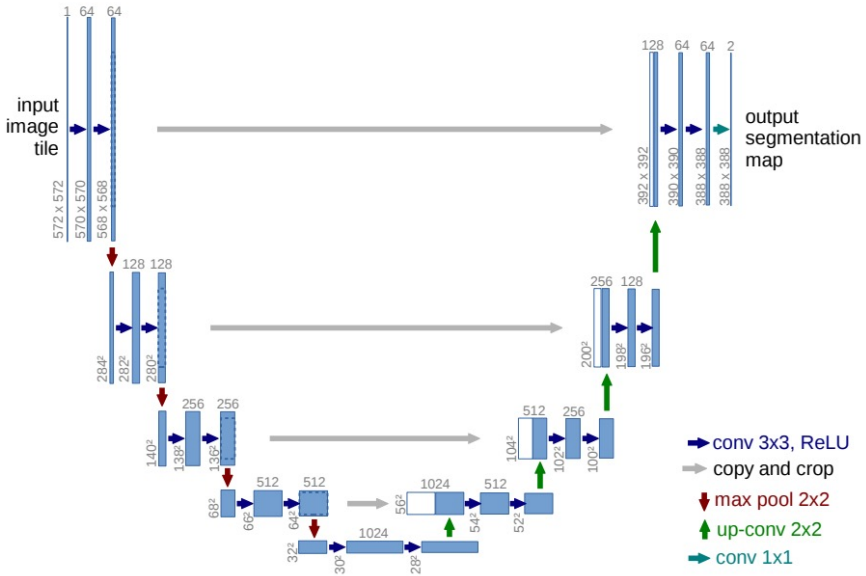
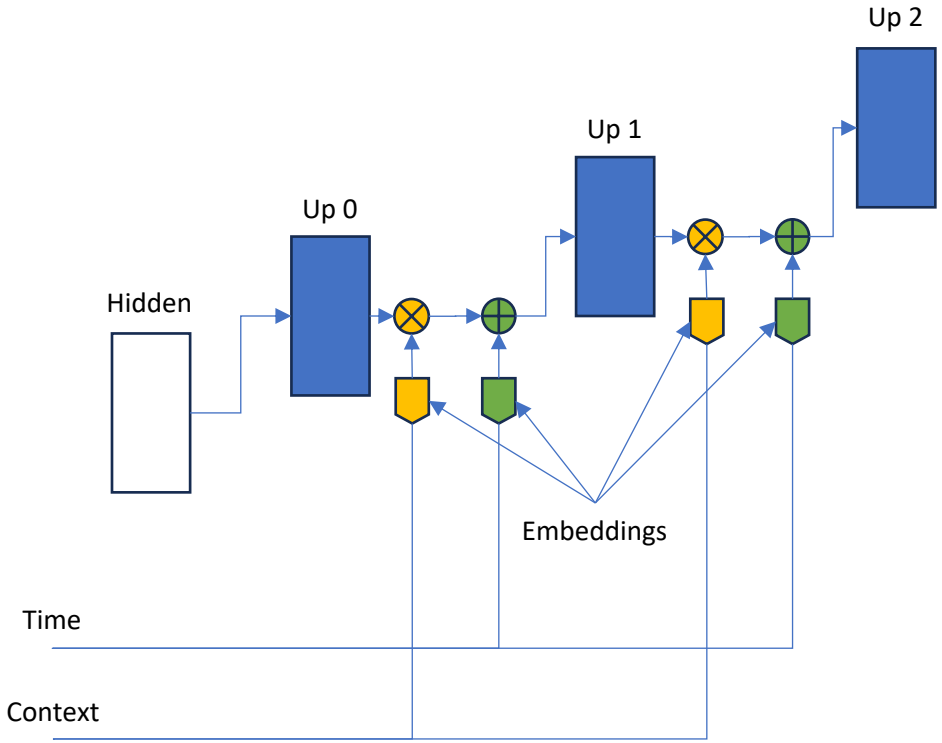


Fig. 1. U-net architecture (example for 32x32 pixels in the lowest resolution). Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations.

Including Additional Information

- In addition to simply learning the noise we want to know the noise in context.
- Noise from which time step?
- Context from user?

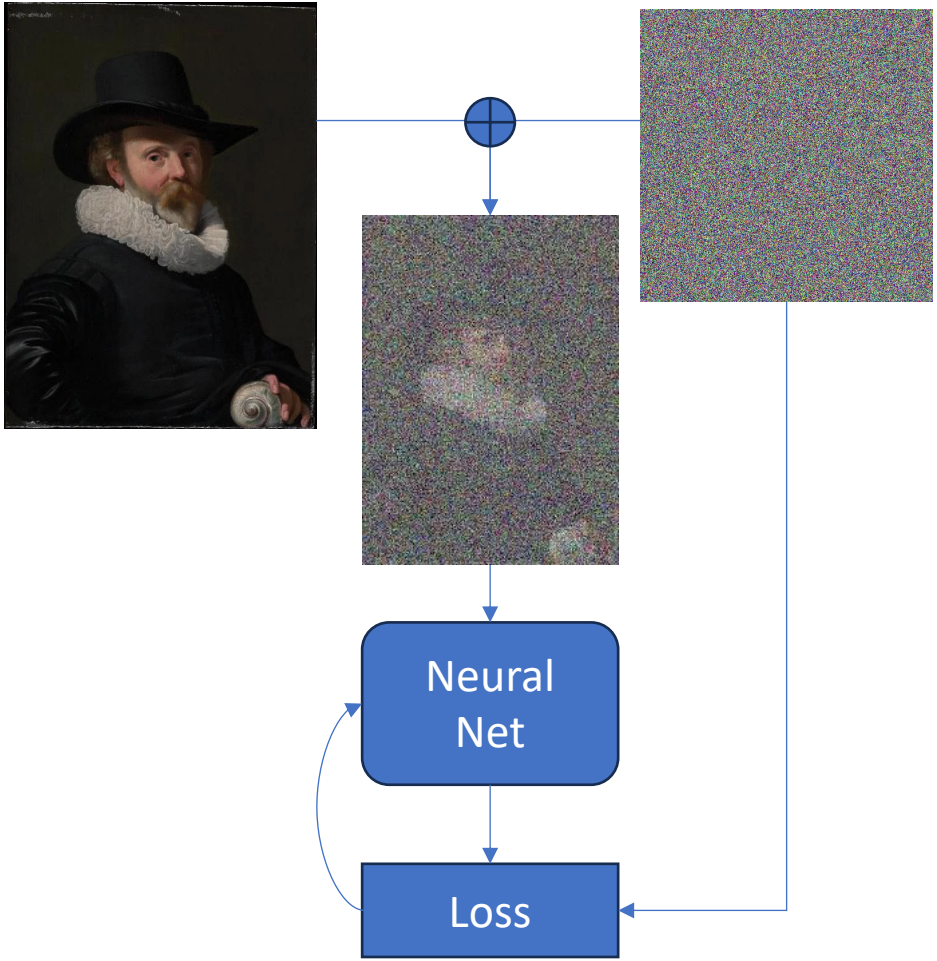


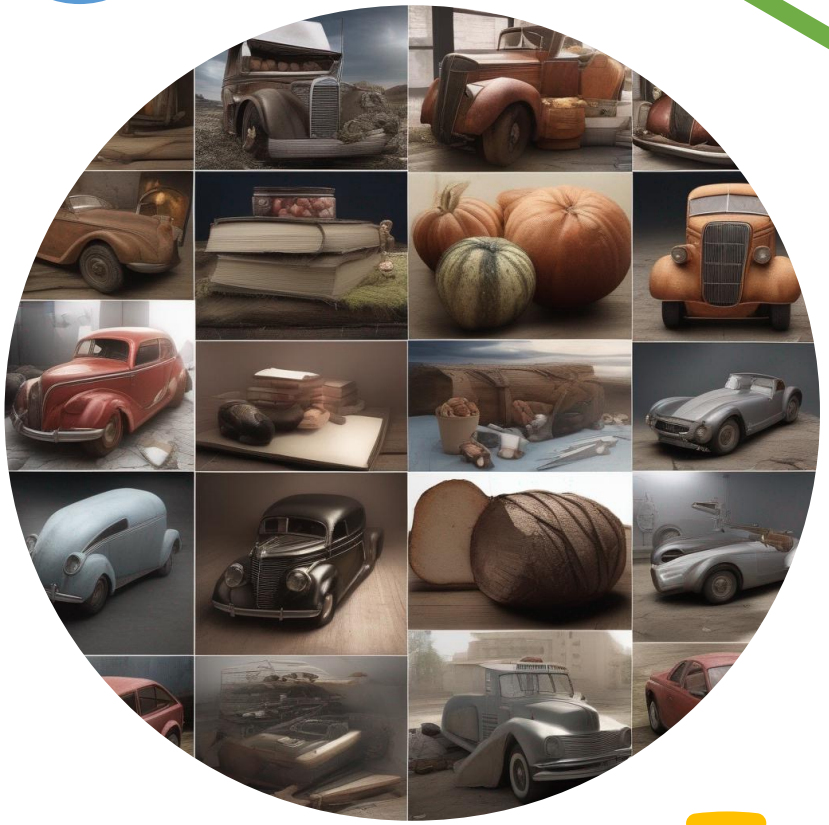
Art lessons for machines: how do we train these models?

What is it actually trying to learn?

- Learns the noise at each time step so it can run the process backwards to a noiseless-image.

- Synthesise data by adding noise different amounts of noise to training images to simulate different time steps.



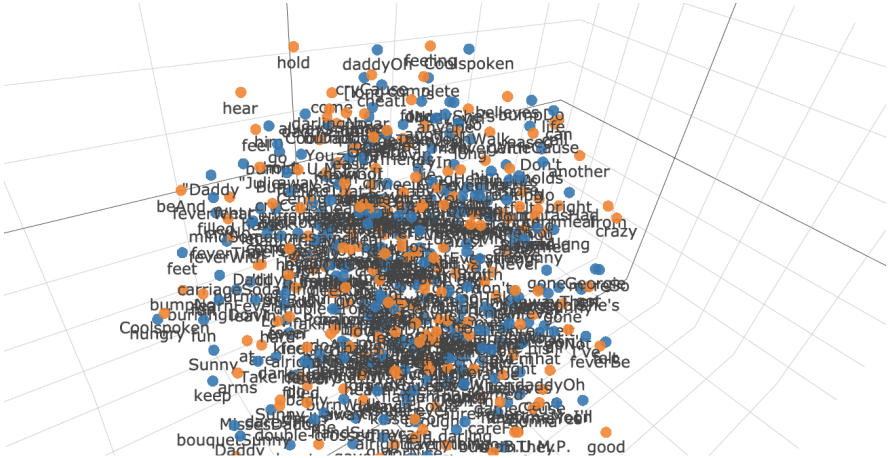


How to ask for pretty pictures: What do you want them to create for you?

Our old friend: Word Embeddings

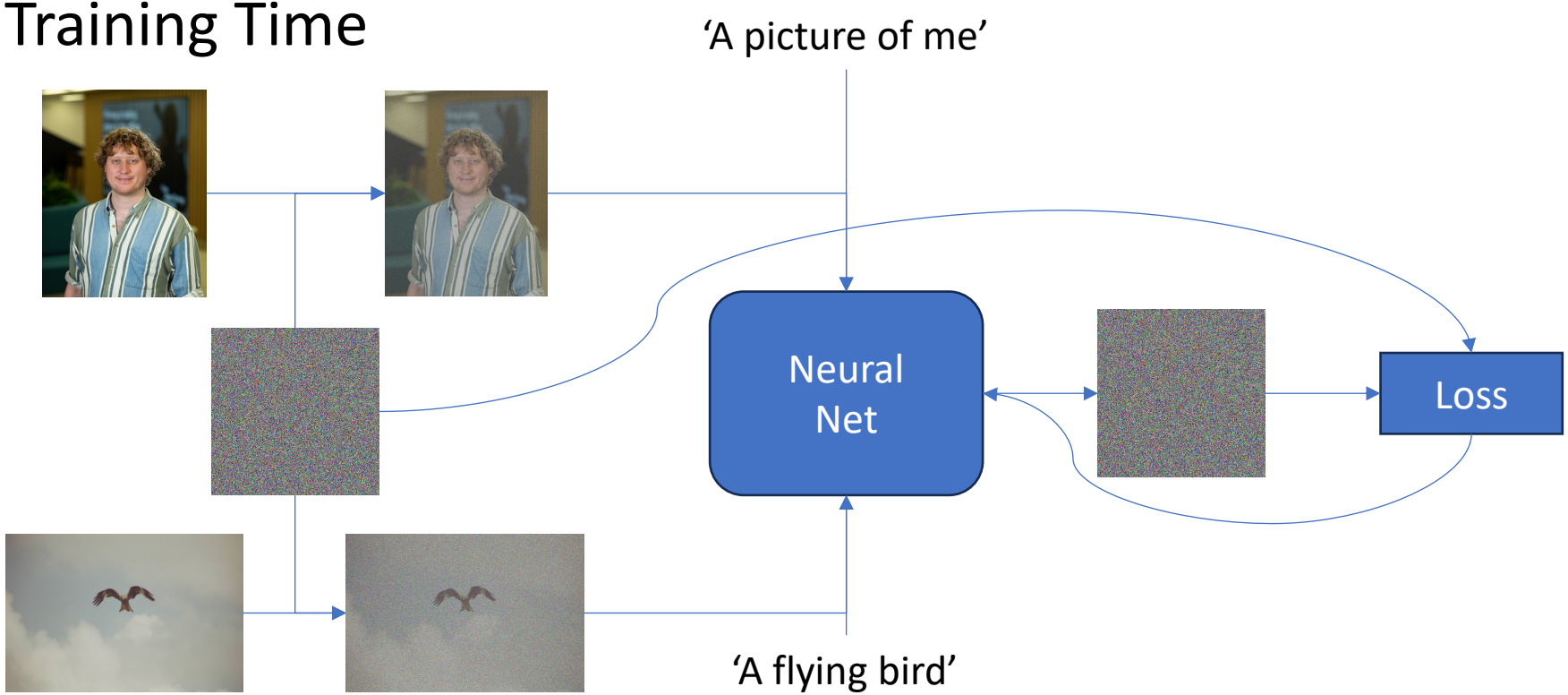
- Given a reference dictionary and a learnt process we turn words (sentences, tokens, etc.) into vectors.

- These vectors need to be of a matching dimension to the images to be incorporated into the model.



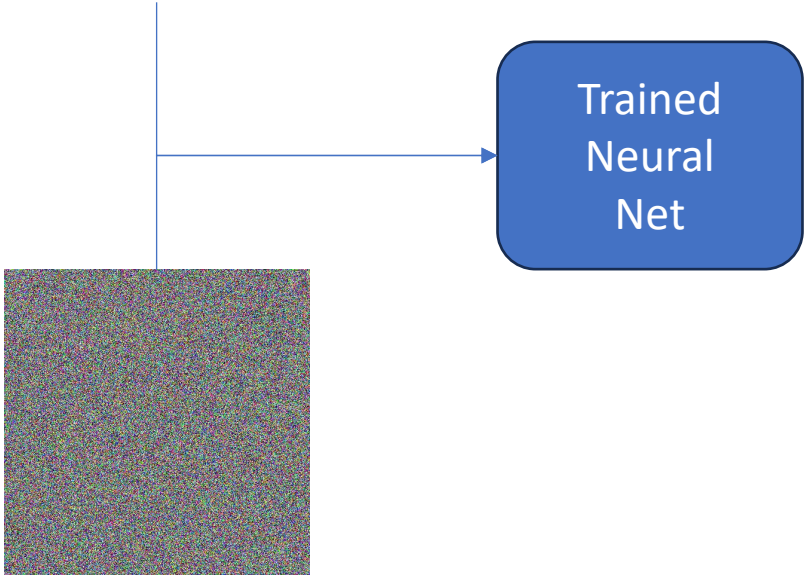
cat =>	1.2	-0.1	4.3	3.2
mat =>	0.4	2.5	-0.9	0.5
on =>	2.1	0.3	0.1	0.4

Adding Context at Training Time



Giving Context at Sample Time

‘A picture of me flying like a bird’

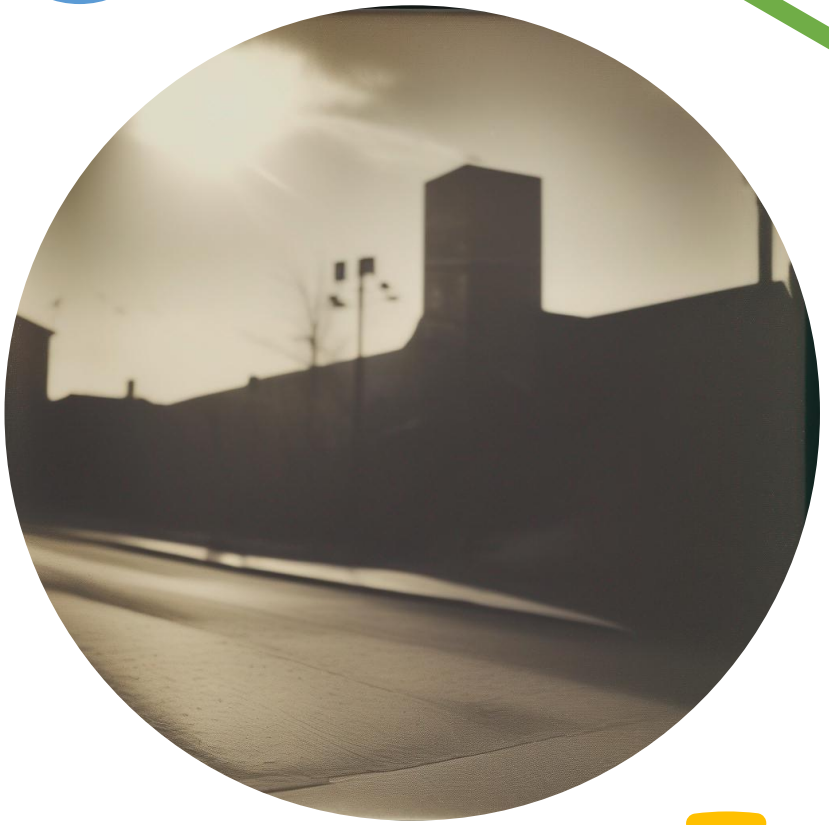


Alternative context to embeddings

Context just needs to be a vector to give to the model.

The source of the context could be text, categorical, other images etc.

Giving the model context it has never seen before will still create something new.



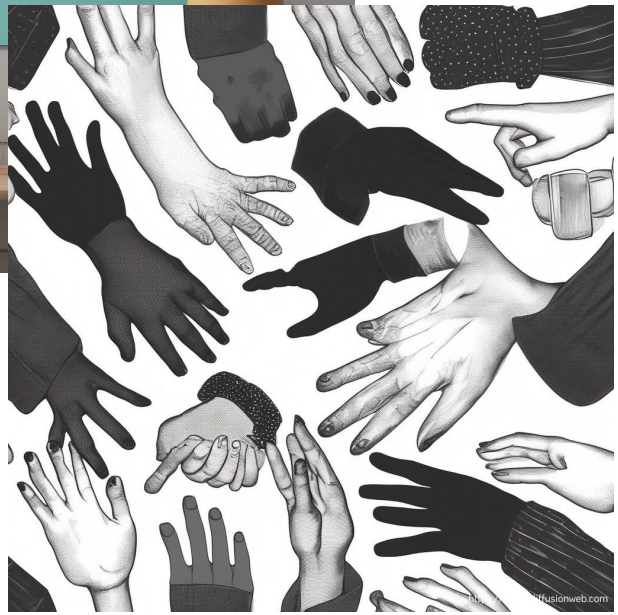
Limitations and ethical quandaries

Limiting Factors

Can’t create new art styles. (Can it be creative?)

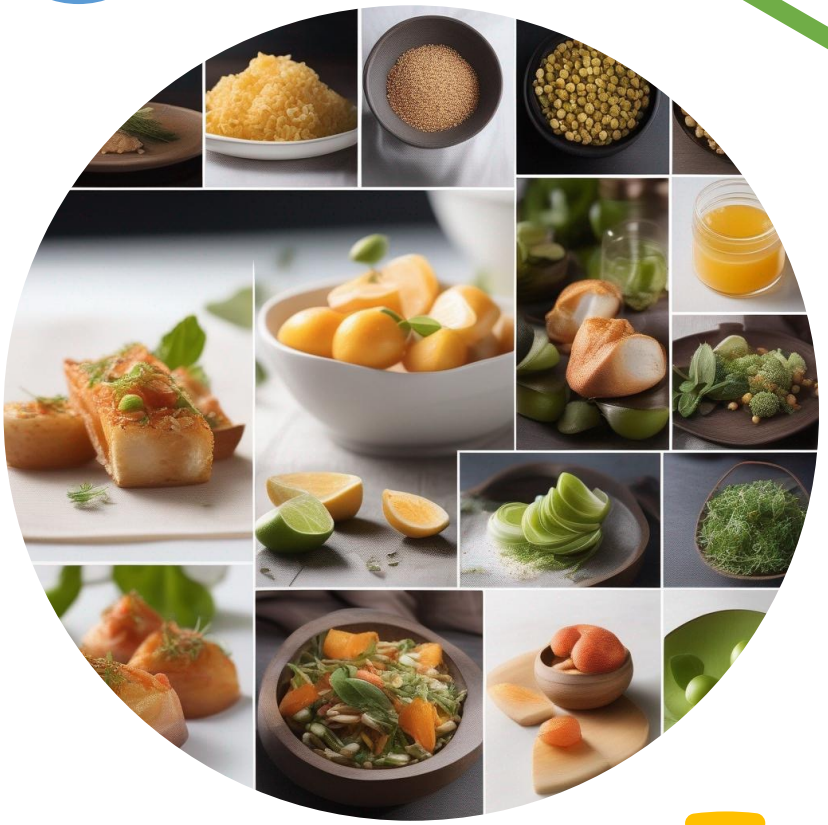
Fairly bad a writing.

Bad at hands???

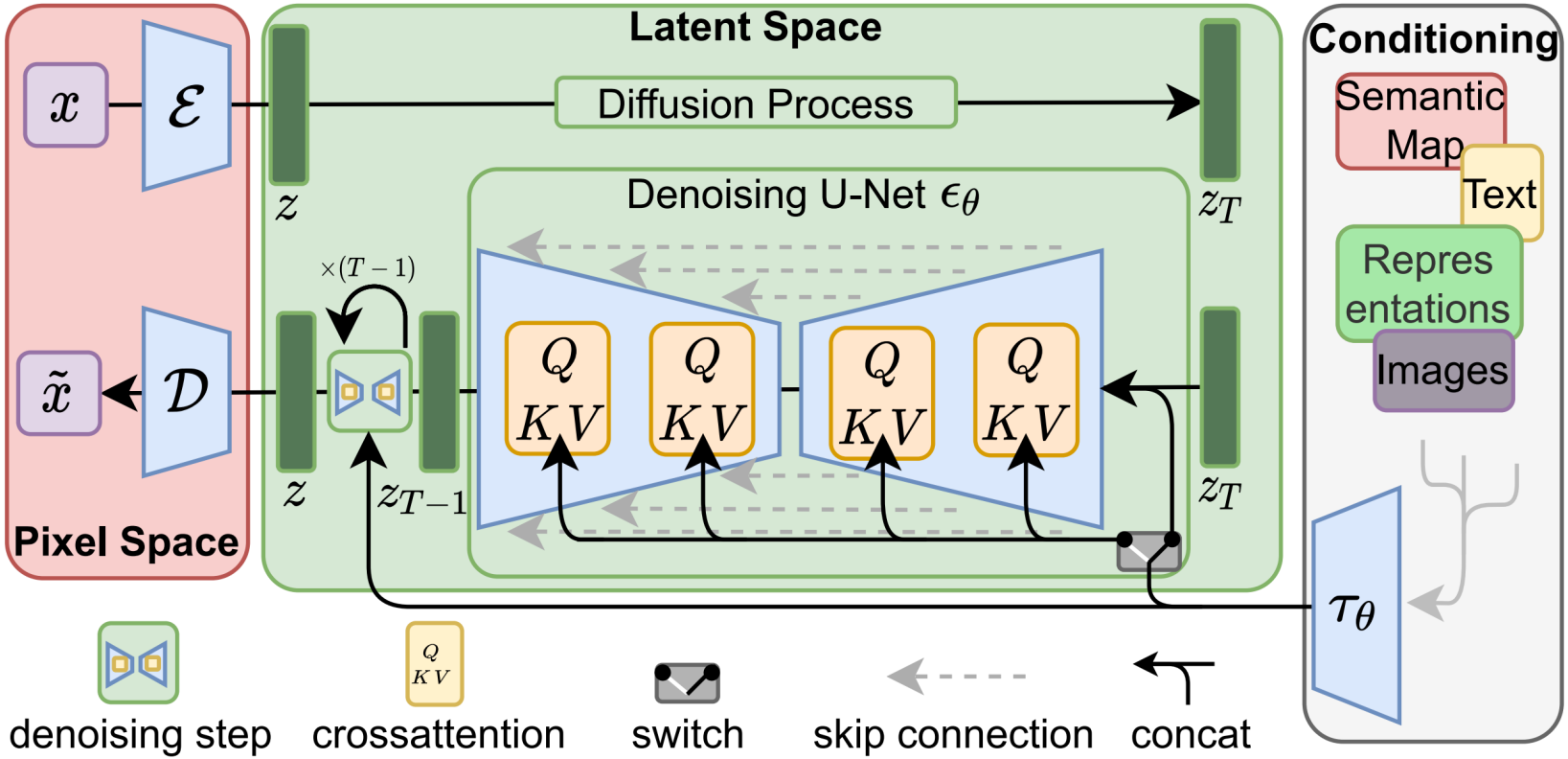


Ethical Quandaries
(deserves it’s own talk)

- Stealing artists work for training data.
- Biases in the dataset.
- Generating images of others without consent.
- Misinformation/misrepresentation.
- Faking artwork???



A sneak peak at stable diffusion


Latent Space
Diffusion Process
Denoising U-Net ϵ_θ
$\times(T-1)$
x
$\mathcal{E}$
z
z_T
$\tilde{x}$
$\mathcal{D}$
z_{T-1}
Q
K V
Pixel Space
Conditioning
Semantic Map
Text
Repres entations
Images
τ_θ
denoising step
crossattention
switch
skip connection
concat

Thanks for Listening!