Hierarchical Text-Conditional Image Generation with CLIP Latents

Authors: Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, Mark Chen

Presented By: Satya Sai Srinath, Debarshi Deka, Prem Abhinav Potta

Table of Contents

A fun motivation/strong limitation

Prompt: "A cartoon photo of a female anthropologist holding magnifying glass"

Prerequisites

- CLIP : Contrastive Language-Image Pre-Training
- Diffusion Models & U-Net
- GLIDE: Guided Language-to-Image Diffusion for Generation and Editing

CLIP

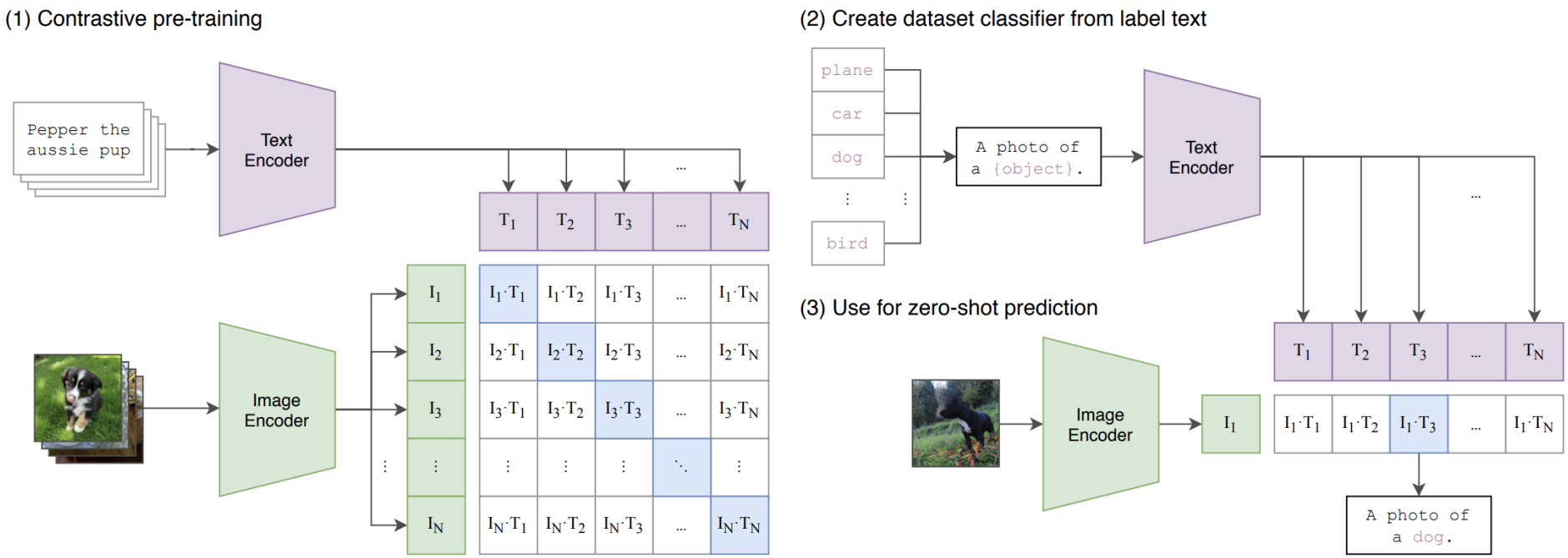
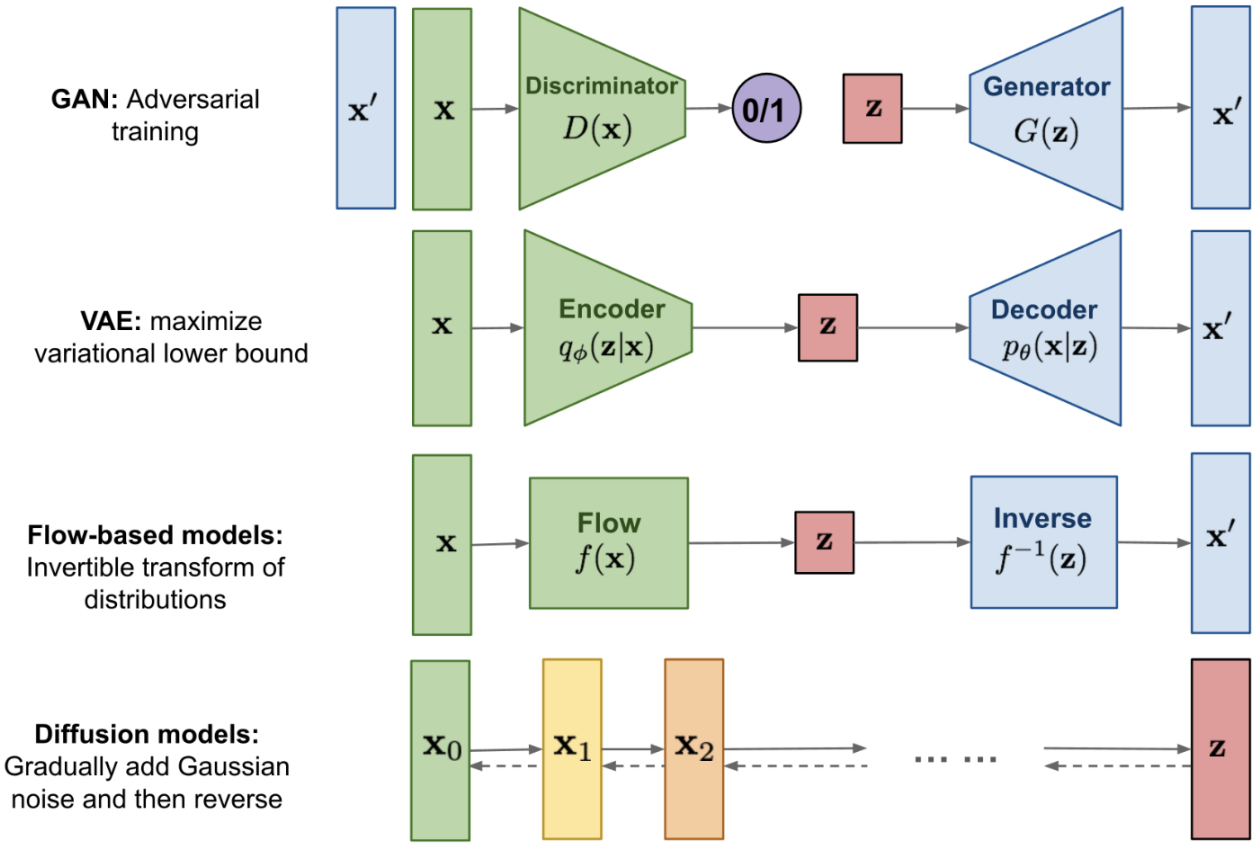


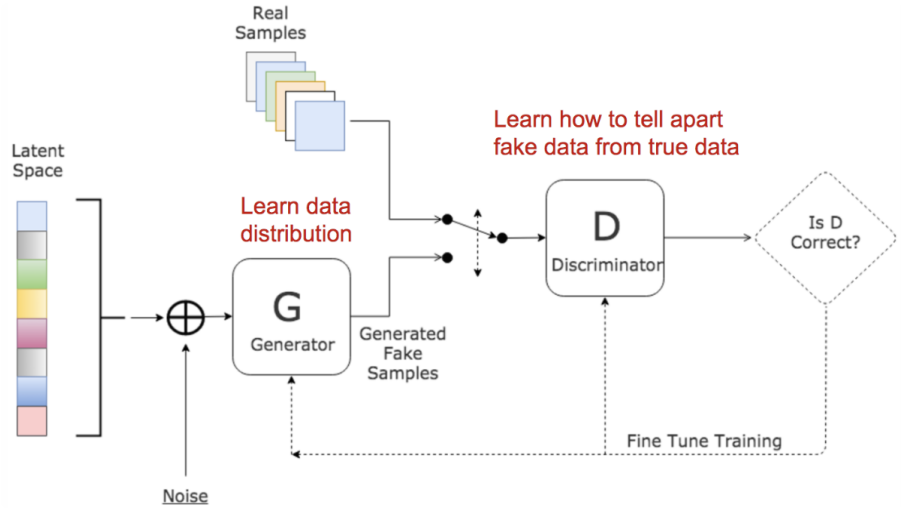
Figure 1. Summary of our approach. While standard image models jointly train an image feature extractor and a linear classifier to predict some label, CLIP jointly trains an image encoder and a text encoder to predict the correct pairings of a batch of (image, text) training examples. At test time the learned text encoder synthesizes a zero-shot linear classifier by embedding the names or descriptions of the target dataset's classes.

Overview of Generative Models

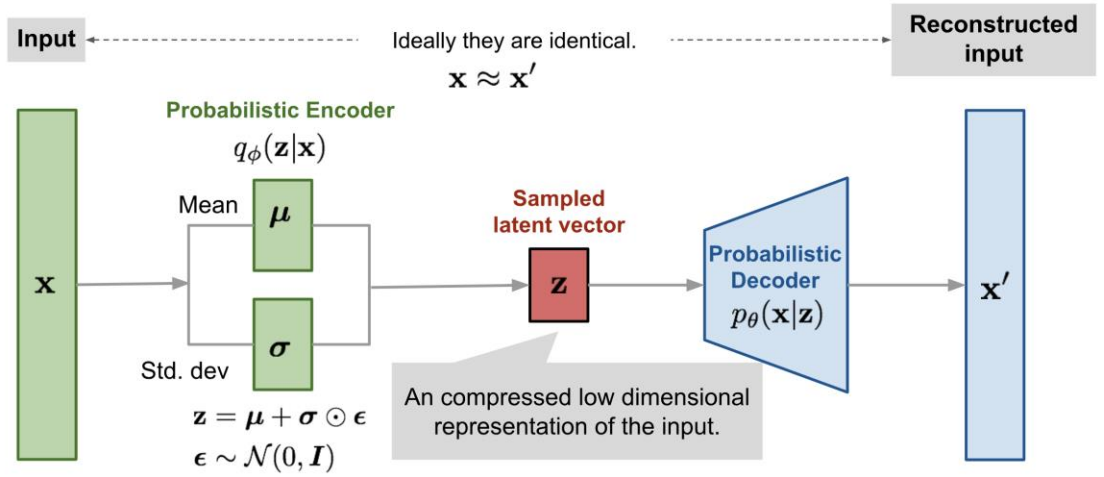


Different types of generative models

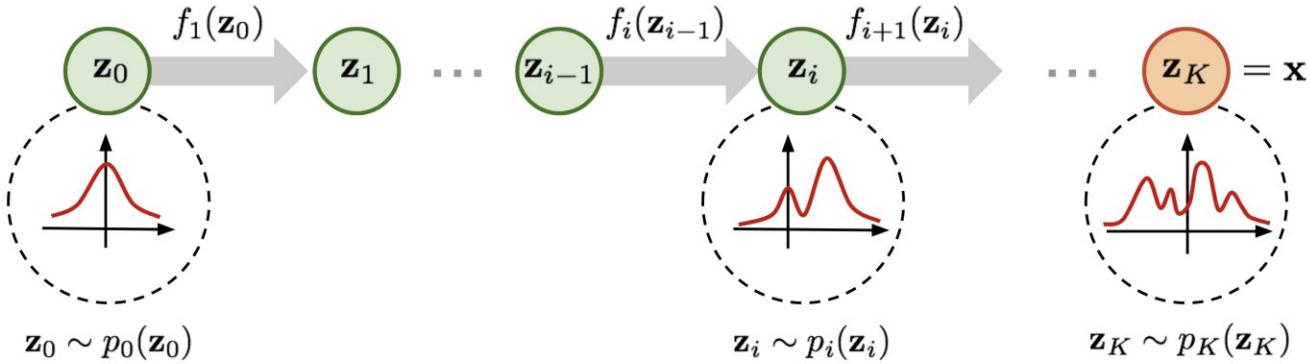
Source: https://lilianweng.github.io/posts/2021-07-11-diffusion-models/



Generative Adversarial Networks

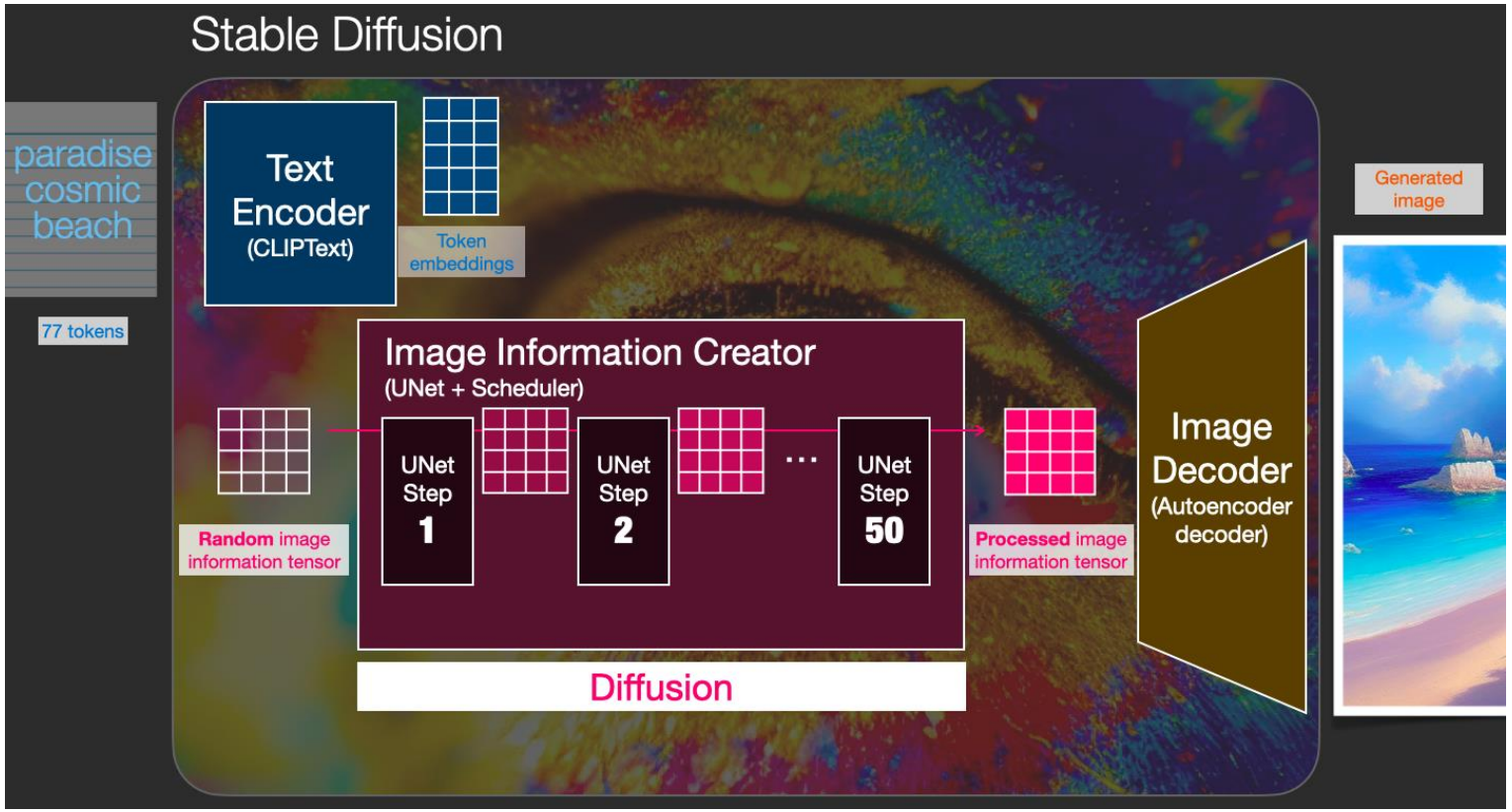


Variational AutoEncoders



Flow Based Models

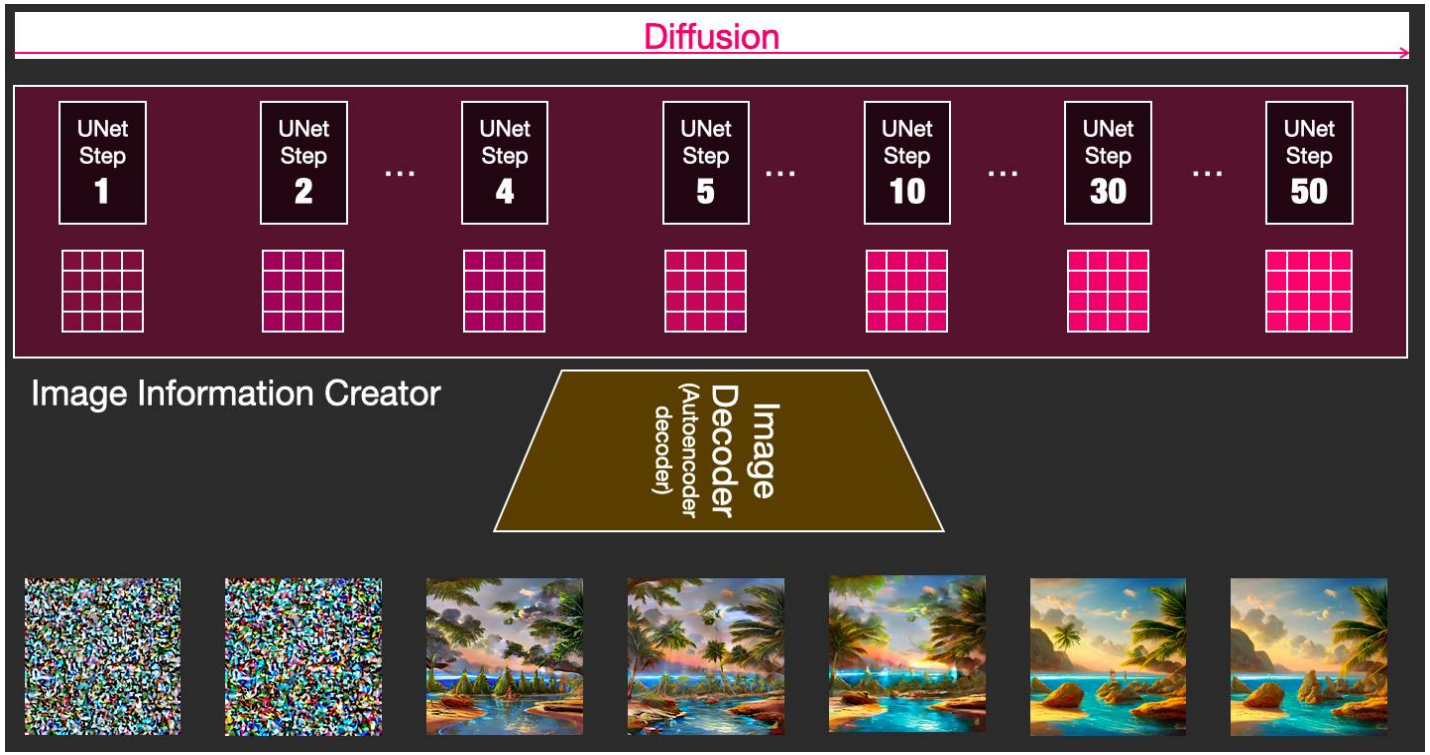
Flow diagram of Stable Diffusion Model



1. Input sentence
2. Tokenize and pass it to CLIP Encoder
3. Pass the token embeddings and random image information to UNet
4. UNet will create some informative image from this input combination (in latent space)
5. Pass the processed latent spaced image information to decoder to generate image

Reference: https://jalammar.github.io/illustrated-stable-diffusion/

U-Net as the denoising model



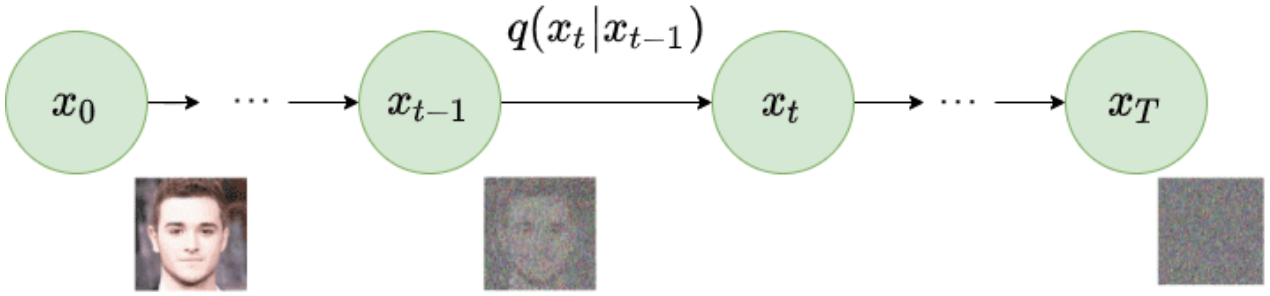
Probing U-Net model at different timesteps

Reference: https://jalammar.github.io/illustrated-stable-diffusion/

Diffusion Models

Forward Diffusion Process

1. Take an image
2. Add small Gaussian noise continuously till the image converts to noise
3. Can be modelled as Markov chain as each step is dependent only on the previous step



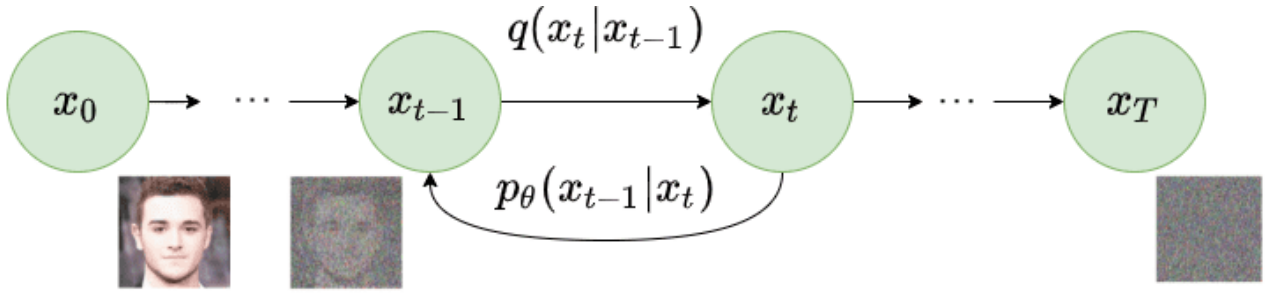
$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}) \quad q(\mathbf{x}_{1:T}|\mathbf{x}_0) = \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1})$$

Reference: https://theaisummer.com/diffusion-models/

Diffusion Models

Reverse Diffusion Process

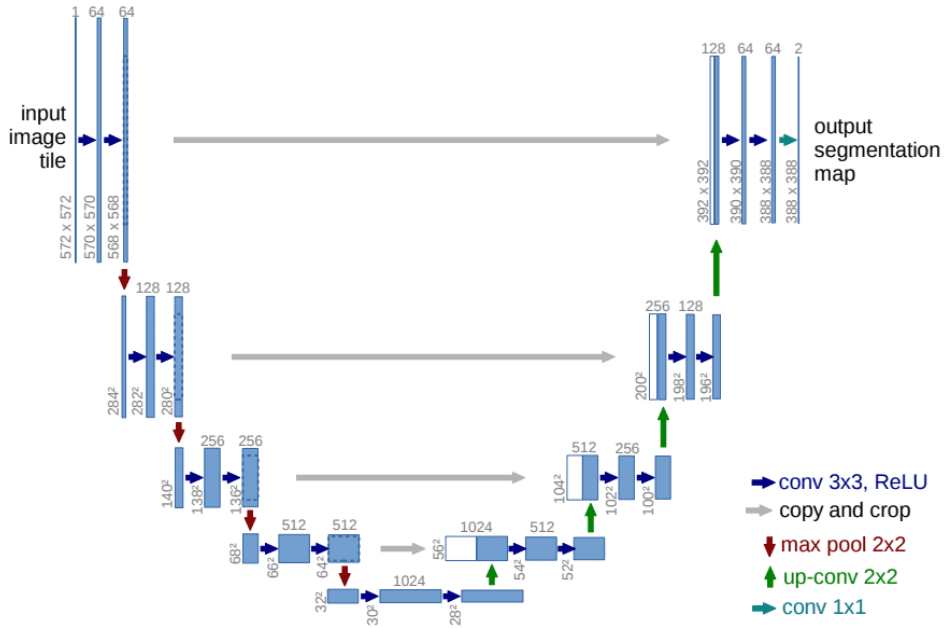
1. Train a model to denoise the image
2. Predict the noise and subtract it to get the unnoised version of the image.
3. If we have ground truth (x0), the reverse diffusion step will be much more easier.



$$p_\theta(\mathbf{x}_{0:T}) = p(\mathbf{x}_T) \prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) \quad p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t))$$

Reference: https://theaisummer.com/diffusion-models/

U-Net



Overview of U-Net architecture
Reference: https://arxiv.org/pdf/1505.04597.pdf

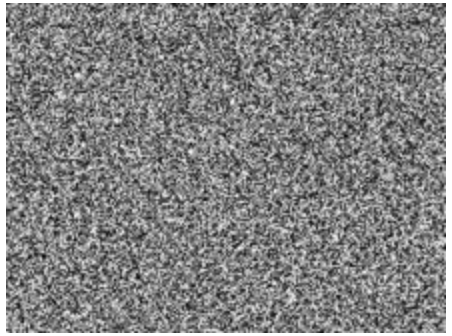
1. Designed for biomedical image segmentation
2. Downsample and then upsample (bottleneck style)
3. Pass information using skip connections
4. Useful when input and output dimensions are same (can modify to behave it differently as well)

GLIDE (DALL E 1.5)

- Guided diffusion

Incorporate image embeddings into the diffusion in order to "guide" the generation

$$p_\theta(\mathbf{x}_{0:T}|y) = p_\theta(\mathbf{x}_T)\prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t, y)$$



Without any "guidance", this white noise can miss out some context!

GLIDE (Contd.)

Classifier guidance

- Train a new neural network to classify the generated image and drive towards a target class (y).

$$f_\phi(y|\mathbf{x}_t, t),$$

Classifier-free guidance

- Replace the label in class-conditioned diffusion model as null with a fixed probability.
- No need to train a new classifier

GLIDE (Contd.)

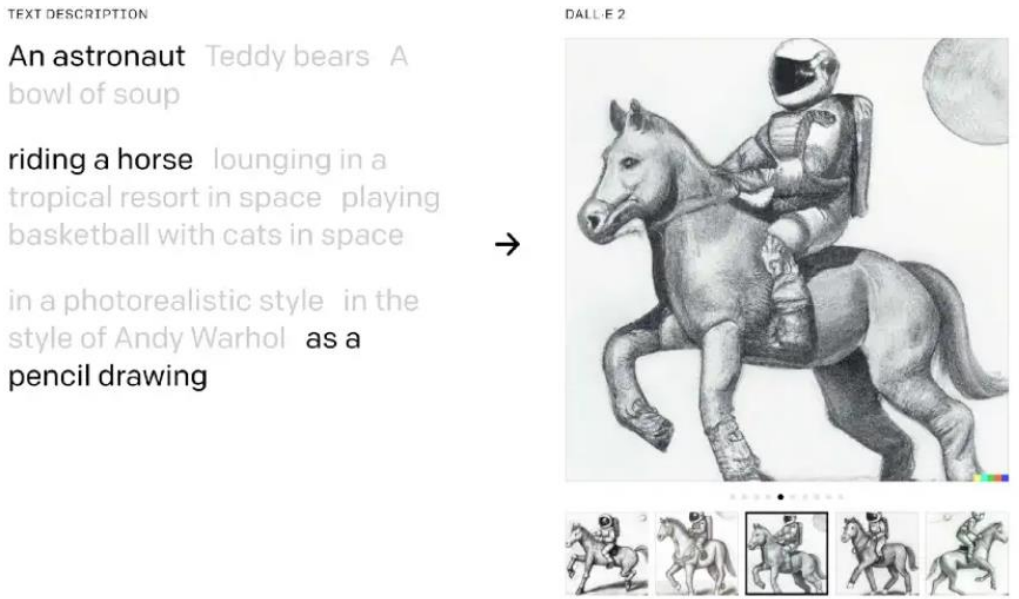
Results from GLIDE paper (**Reference:** https://arxiv.org/pdf/2112.10741v3.pdf)

DALL·E 2

- The DALL·E 2 system significantly improves results over the original DALL·E modes which was based on VQ-VAEs.
- It generates images with 4x greater resolution (compared to original DALL·E and GLIDE), now up to 1024×1024 pixels.
- The model behind the DALL·E 2 system is called unCLIP.

DALL·E 2 (Contd.)

- DALL·E 2 can combine concepts, attributes, and styles



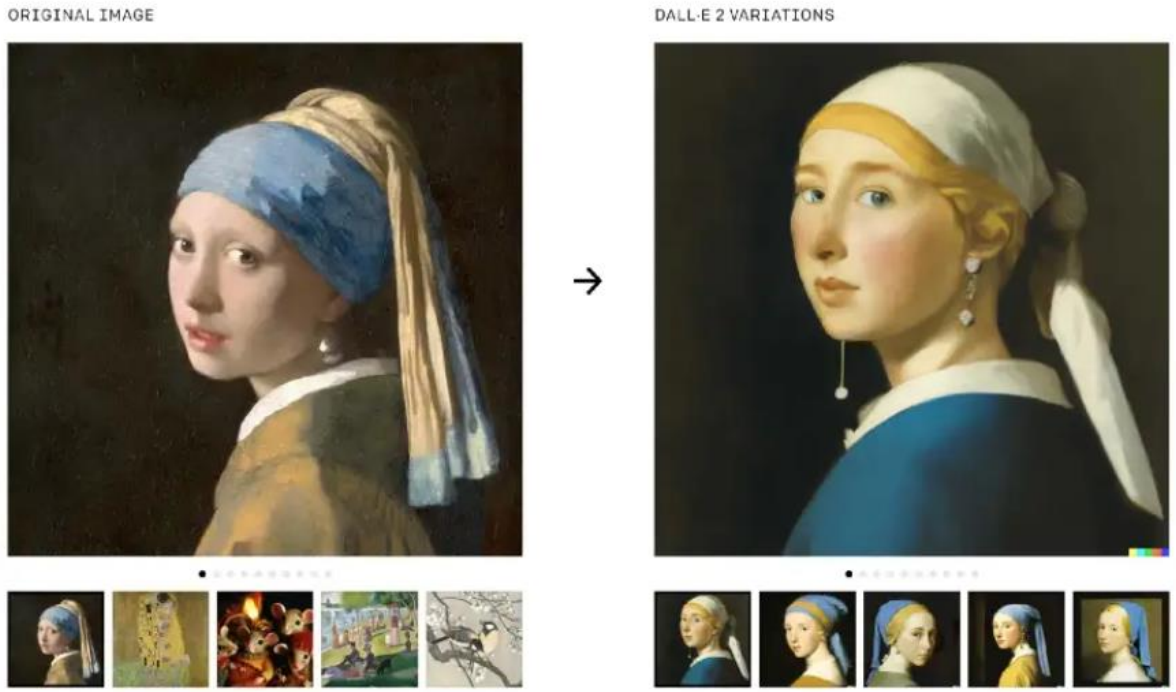
DALL·E 2 (Contd.)

- DALL·E 2 can also perform image editing based on text guidance. It can add and remove elements while taking shadows, reflections, and textures into account.

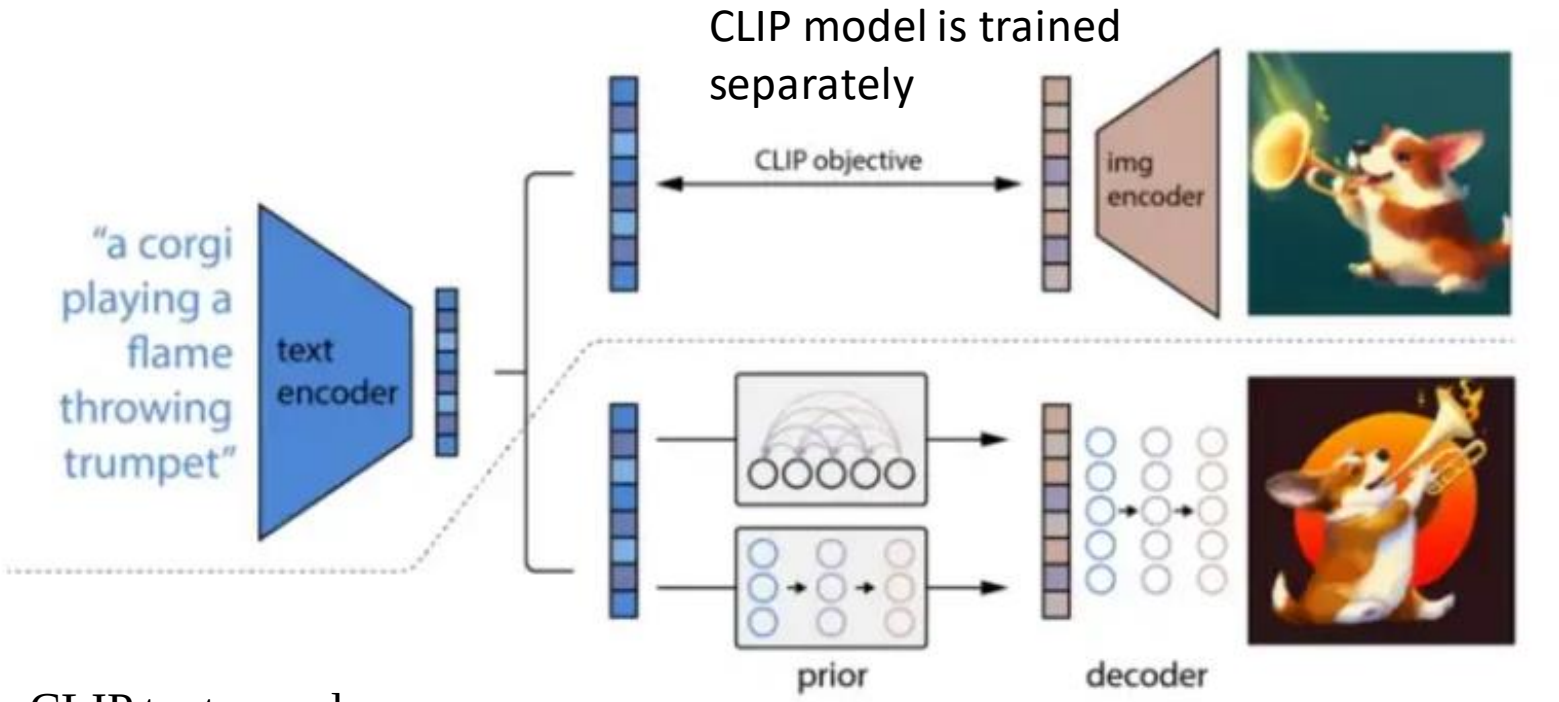
Added a "corgi" at selected location

DALL·E 2 (Contd.)

- DALL·E 2 can be used to generate variations of the original image:



unCLIP - Architecture

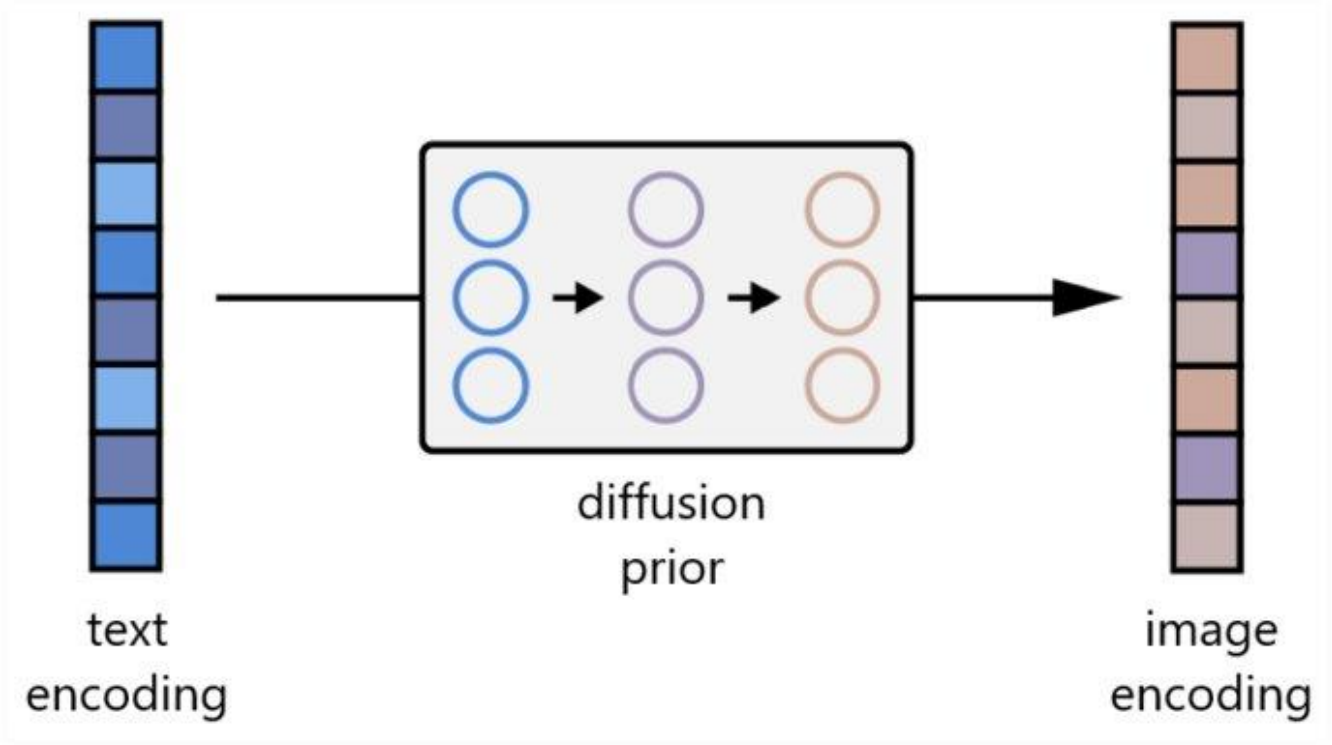


CLIP text encoder generates an embedding for the input text (caption)

A special prior model generates an image embedding based on the text embedding

The diffusion decoder generates an image based on the image embedding.

unCLIP - Prior



unCLIP - Prior (Contd.)

- For the diffusion prior, a decoder-only Transformer with a causal attention mask is trained on a sequence consisting of:
  - the encoded text
  - the **CLIP text embedding**
  - an embedding for the diffusion timestep
  - the **noised CLIP image embedding**
  - a final embedding whose output from the Transformer is used to predict the unnoised CLIP image embedding.

unCLIP - Prior (Contd.)

- The Diffusion Prior is conditioned not only on the CLIP text embedding of the caption, but also the caption itself.
- To improve sample quality, sampling is randomly conducted using classifier-free guidance 10% of the time by dropping the text-conditioning information.
- **To improve quality during sampling time, two image embeddings are generated with the prior and the one with the higher dot product with the text embedding is selected (Why not cosine similarity?)**

unCLIP - Decoder

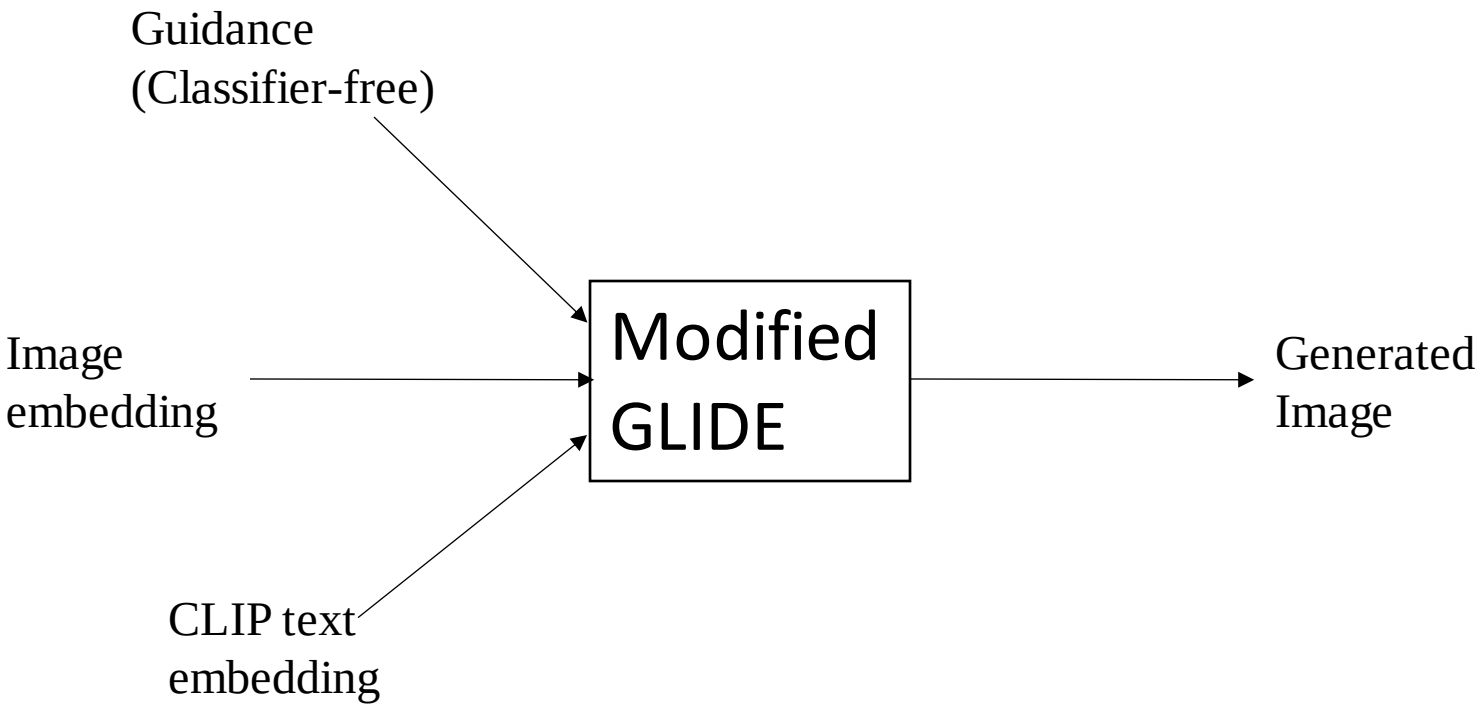


Image Manipulations

- DALL·E 2 can also perform image editing based on text guidance, the feature present in GLIDE. It can add and remove elements while taking shadows, reflections, and textures into account.
- This section discusses some of the unCLIP applications described in the paper:
  - Creating image variations.
  - Making interpolations between images.
  - Language guided image manipulations.

Image Manipulation – Prerequisites

- DDIM is a stochastic diffusion model with shorter sampling time.
- It defines a family of processes, indexed by a 'η' which modulates the stochasticity of the process.
- When $\eta = 0$, the process becomes deterministic i.e. the same original noise leads to the same input image. This is also known as DDIM inversion.

Image Manipulation – Prerequisites

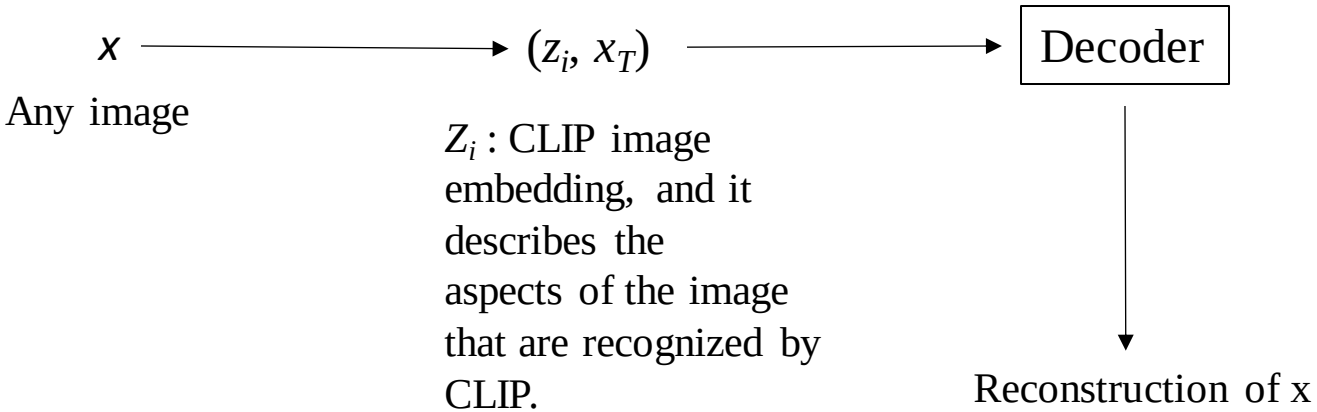


Image Manipulation – Variations

The variations preserve both semantic information like presence of a clock in the painting and the overlapping strokes in the logo, as well as stylistic elements like the surrealism in the painting and the color gradients in the logo, while varying the non-essential details.

Image Manipulation – Variations (Contd.)

- When we sample in the decoder using DDIM with $\eta > 0$, we can create image variations for the given bipartite latent representation (z_i, x_T).
- The larger the η parameter, the larger variations, and we can see what information was captured in the CLIP image embedding and present in all samples.

Image Manipulation – Interpolations

The decoder seed is fixed across each row and the intermediate variations resulting from the *slerp* interpolations naturally blend the content and style from both input images

Image Manipulation – Interpolations (Contd.)

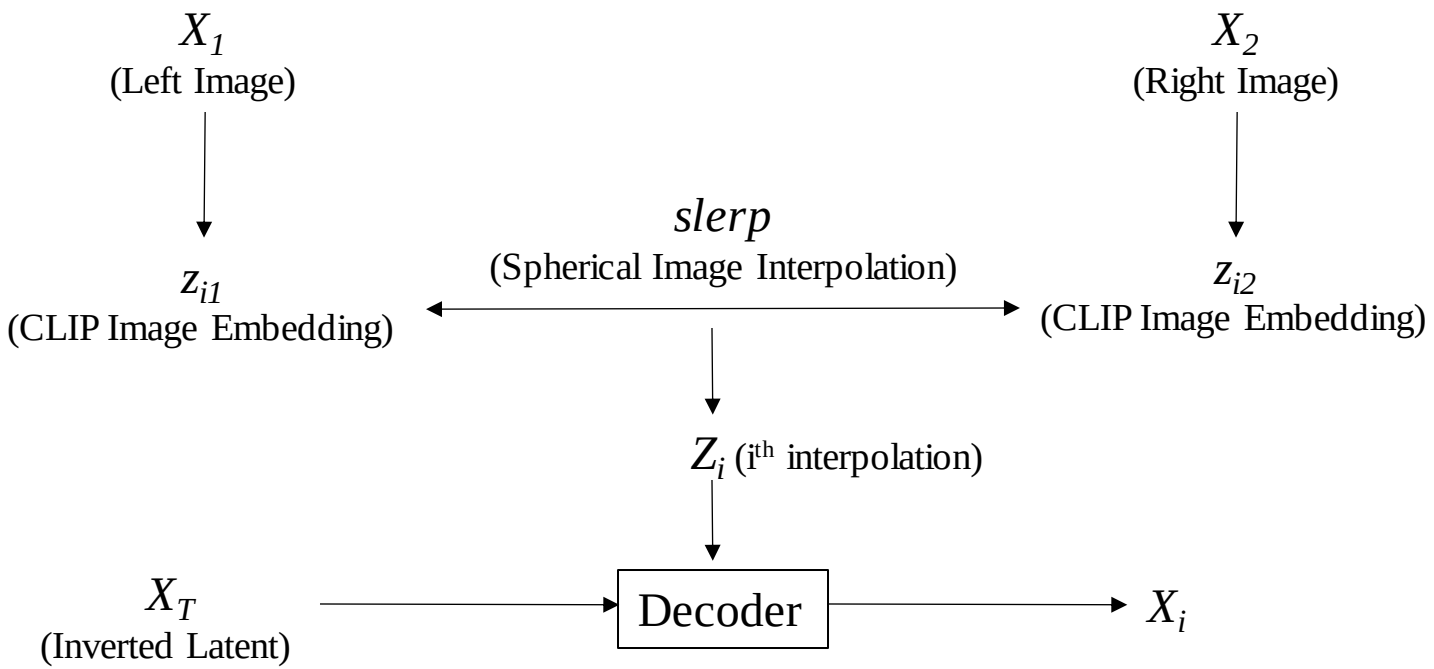


Image Manipulation – Text Diffs

DDIM inversion is applied to obtain a perfect reconstruction (see the first column), then *slerp* transformations are applied across each row.

Image Manipulation – Text Diffs (Contd.)

- In order to modify the image to reflect a new text description y, we first obtain its CLIP text embedding z_t, as well as the CLIP text embedding z_{t0} of a caption describing the current image (for example, it might be a dummy caption like "a photo of ...").
- We compute a text diff vector

  $$z_d = norm(z_t - z_{t0})$$

- Now, we rotate between the image CLIP embedding of z_i and of the text diff vector z_d with *slerp* and generate images with the fixed base DDIM noise x_T obtained from DDIM inversion throughout the entire trajectory.

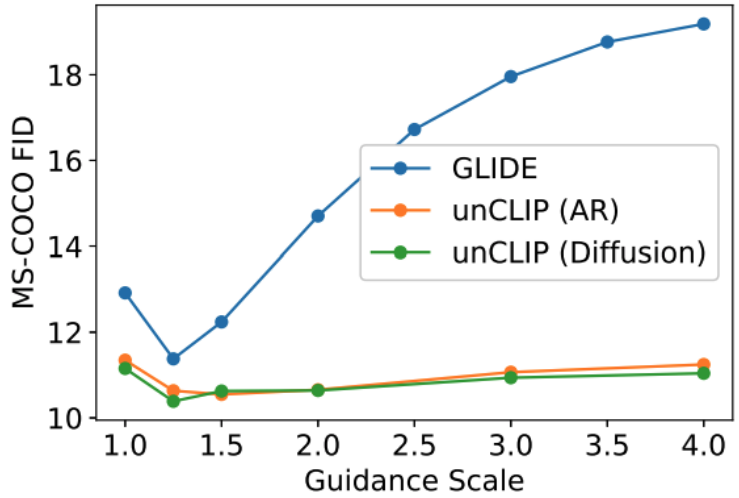
Text-to-Image Generation – Human Evaluation

unCLIP Prior	Photorealism	Caption Similarity	Diversity
AR	$47.1\% \pm 3.1\%$	$41.1\% \pm 3.0\%$	$62.6\% \pm 3.0\%$
Diffusion	$48.9\% \pm 3.1\%$	$45.3\% \pm 3.0\%$	$70.5\% \pm 2.8\%$

Table 1: Human evaluations comparing unCLIP to GLIDE. We compare to both the AR and diffusion prior for unCLIP. Reported figures are 95% confidence intervals of the probability that the unCLIP model specified by the row beats GLIDE. Sampling hyperparameters for all models were swept to optimize an automated proxy for human photorealism evaluations.

Text-to-Image Generation – Human Evaluation

Figure 9: Samples when increasing guidance scale for both unCLIP and GLIDE, using the prompt, "A green vase filled with red roses sitting on top of table." For unCLIP, we fix the latent vectors sampled from the prior, and only vary the guidance scale of the decoder. For both models, we fix the diffusion noise seed for each column. Samples from unCLIP improve in quality (more realistic lighting and shadows) but do not change in content as we increase guidance scale, preserving semantic diversity even at high decoder guidance scales.



Text-to-Image Generation – Comparison on MS-COCO

Model	FID	Zero-shot FID	Zero-shot FID (filt)
AttnGAN (Xu et al., 2017)	35.49		
DM-GAN (Zhu et al., 2019)	32.64		
DF-GAN (Tao et al., 2020)	21.42		
DM-GAN + CL (Ye et al., 2021)	20.79		
XMC-GAN (Zhang et al., 2021)	9.33		
LAFITE (Zhou et al., 2021)	8.12		
Make-A-Scene (Gafni et al., 2022)	**7.55**		
DALL-E (Ramesh et al., 2021)		~ 28	
LAFITE (Zhou et al., 2021)		26.94	
GLIDE (Nichol et al., 2021)		12.24	12.89
Make-A-Scene (Gafni et al., 2022)			11.84
unCLIP (AR prior)		10.63	11.08
unCLIP (Diffusion prior)		**10.39**	**10.87**

Table 2: Comparison of FID on MS-COCO 256×256. We use guidance scale 1.25 for the decoder for both the AR and diffusion prior, and achieve the best results using the diffusion prior.

Text-to-Image Generation – Comparison on MS-COCO

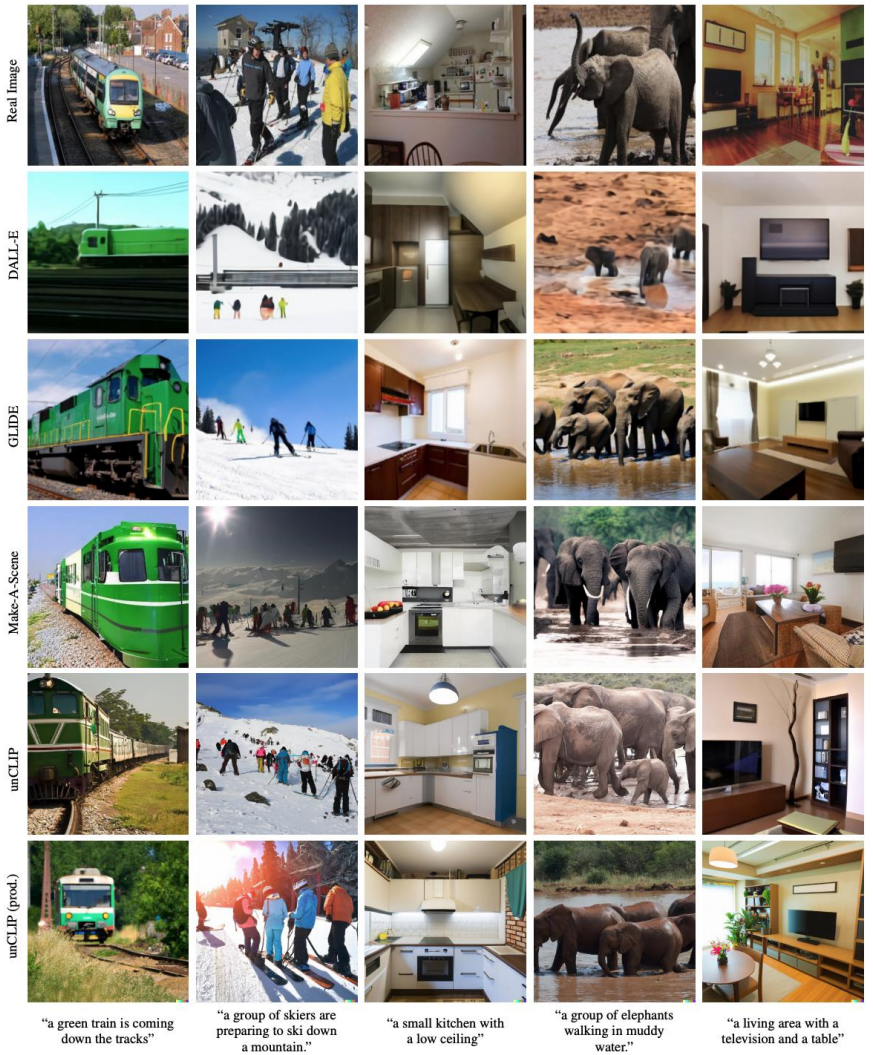


Figure 12: Random image samples on MS-COCO prompts.

Text-to-Image Generation – Aesthetic Quality Comparison

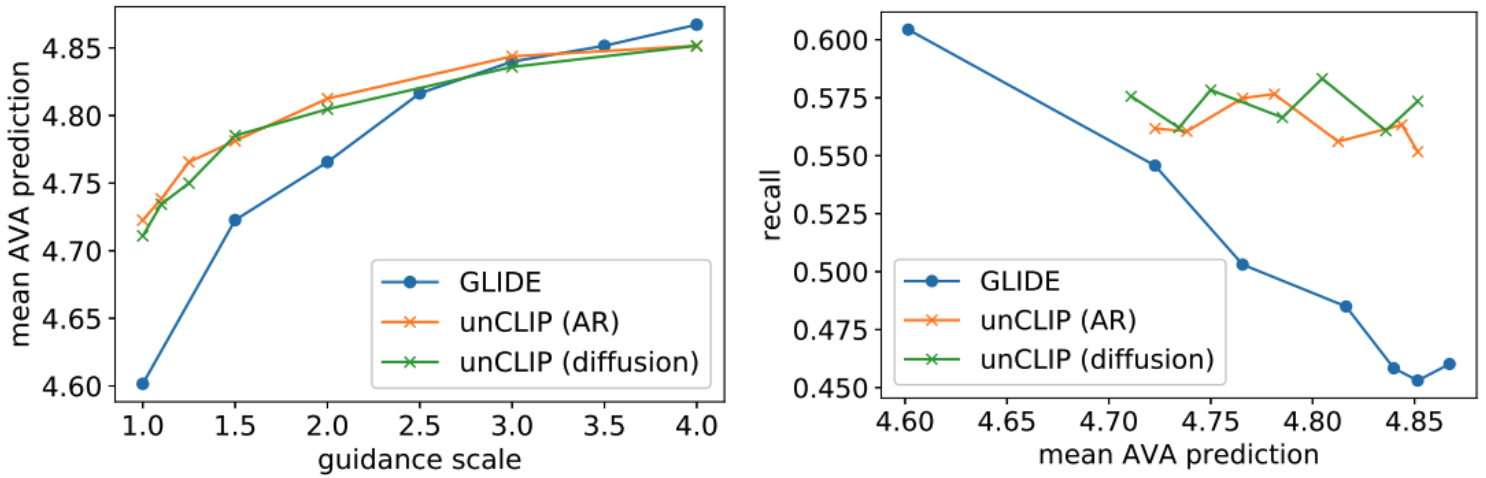


Figure 13: Aesthetic quality evaluations comparing GLIDE and unCLIP using 512 auto-generated artistic prompts. We find that both models benefit from guidance, but unCLIP does not sacrifice recall for aesthetic quality.

Discussion

- How does the model perform when there is no prior?
- How does the model perform when decoder is not there?
- How does the model perform when both are not there?

Discussion

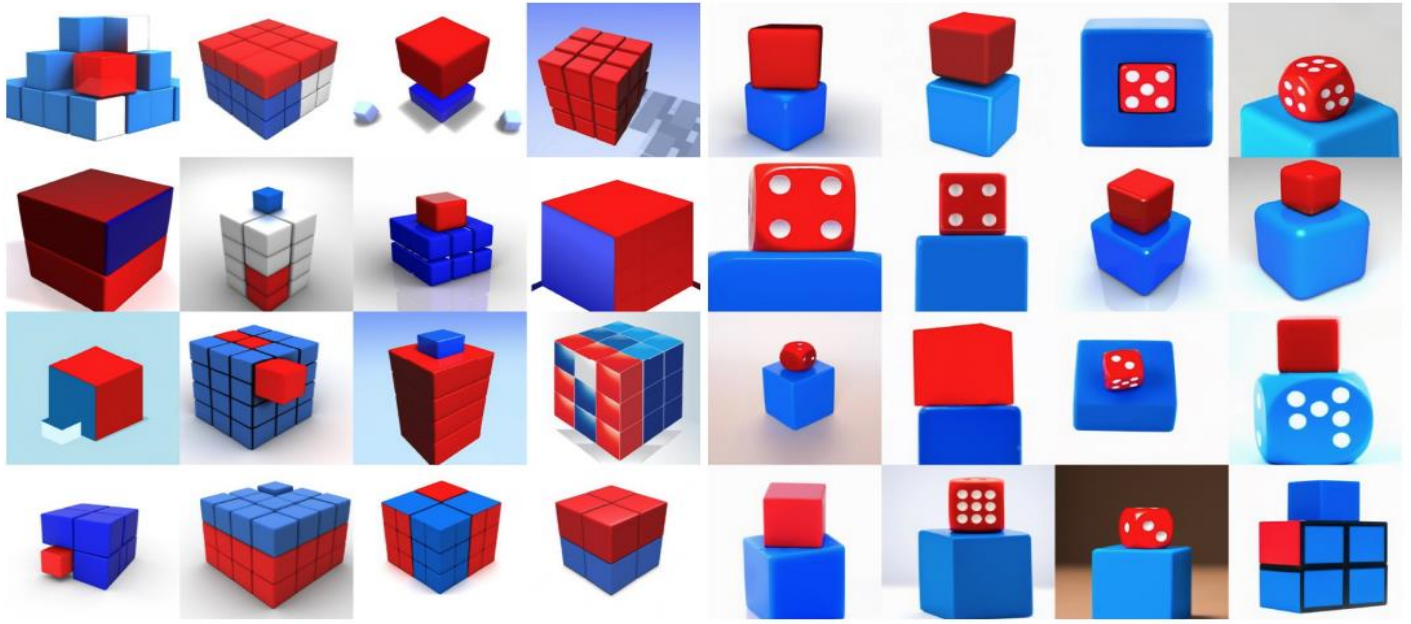
Without Prior and Decoder →

With Prior Without Decoder →

With Prior and Decoder →

Prompts

Limitations



(a) unCLIP (b) GLIDE

Figure 14: Samples from unCLIP and GLIDE for the prompt “a red cube on top of a blue cube”.

Figure 16: Samples from unCLIP for the prompt, “A sign that says deep learning.”

Quiz discussions

1. Which ones did the authors found to computationally more efficient and produce higher-quality samples for prior?

Autoregressive models

Diffusion models – Abstract