Text to image generation with diffusion models

Andres Felipe Cruz-Salinas

panda mad scientist mixing sparkling chemicals, artstation

Outline

- Auto-regressive models
- CLIP
- Intro to DALL·E models
- Diffusion models (generative)
- Diffusion models (conditioned)
- Before DALL·E 2 there was GLIDE
- DALL·E 2
- Other diffusion-based models
- Other text to image models

vibrant portrait painting of Salvador Dalí with a robotic half face

Auto-regressive (or left-to-right) models

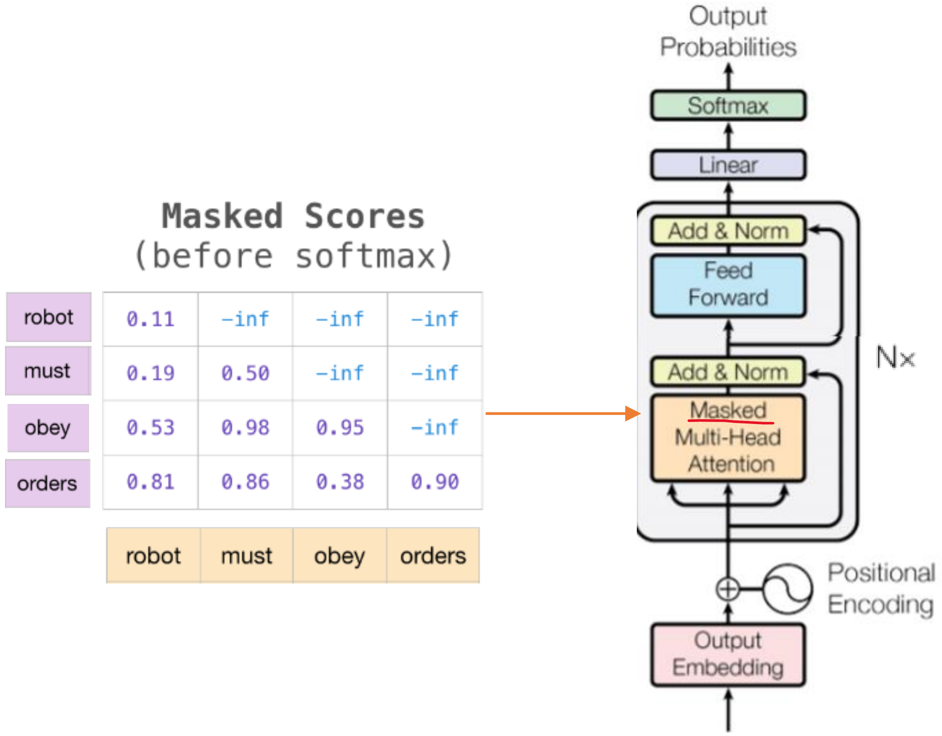
Auto-regressive DL models are trained to predict an element in a sequence based on its causal context:

$$L_1(\mathcal{U}) = \sum_i \log P(u_i | u_{i-k}, \ldots, u_{i-1}; \Theta)$$

Given an unsupervised corpus of tokens $\mathcal{U} = \{u_1, \ldots, u_n\}$

A token can be anything (word, image patches, speech, video frames, etc)

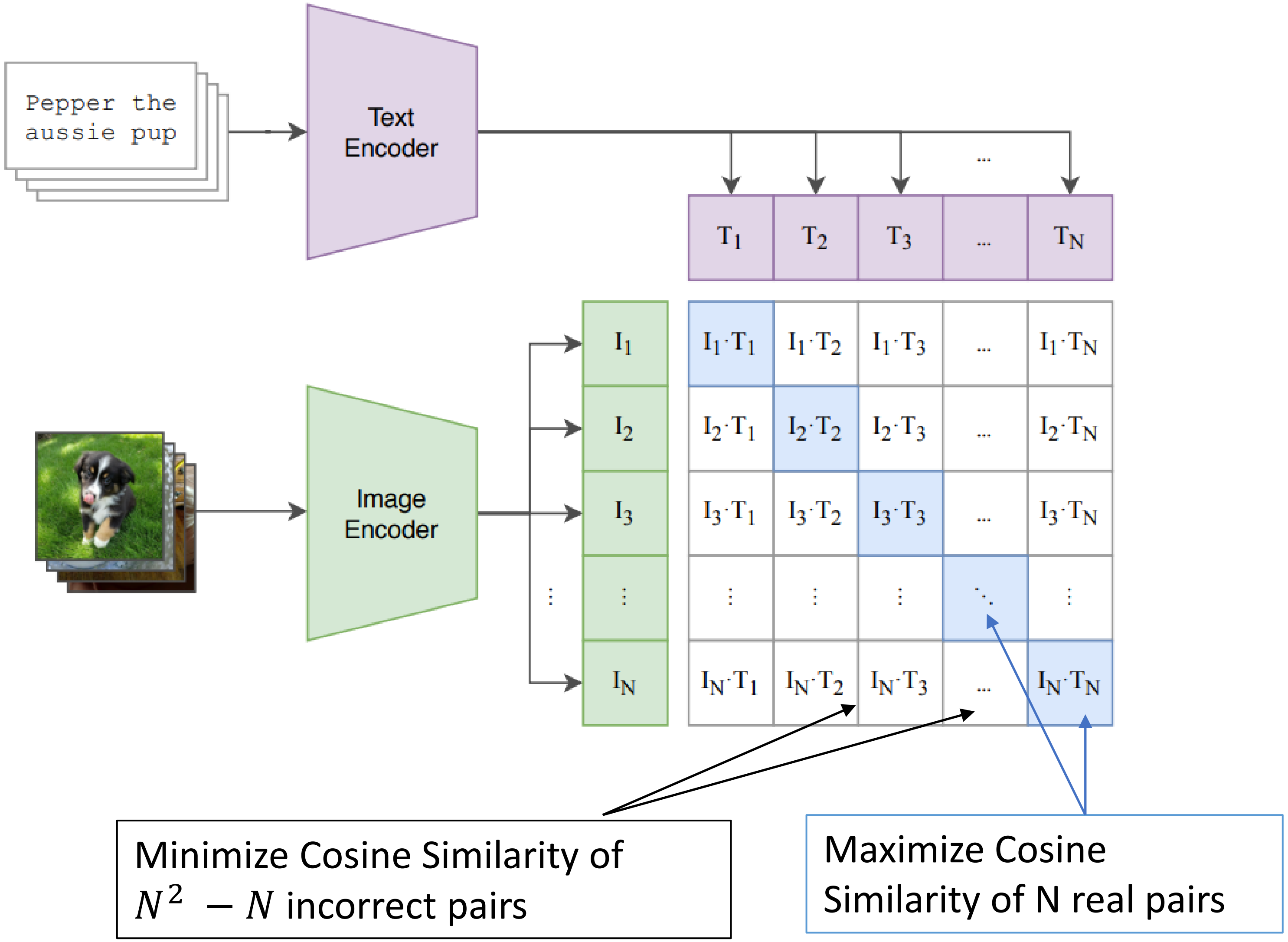
Modern versions of these models typically use a decoder-only transformer model with casual masking.



https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf

https://jalammar.github.io/illustrated-gpt2/

CLIP (Contrastive Language-Image Pre-Training)



- <u>Pretraining maps images and text to the same embedding space.</u>
- Trained of 400M crawled image-text pairs.
- Text encoder is a regular decoder-only transformer.
- The image encoder has different variants: Resnet and Vision transformers (best).
- Zero-shot classification for “free”
- Powerful model for multimodal search, re-ranking, and more…
- Weights are open source!

[2103.00020] Learning Transferable Visual Models From Natural Language Supervision (arxiv.org)

Intro to DALL·E models

DALL·E “1” was introduced in 2021 by OpenAI, a transformer generates directly image tokens from both text and image tokens (more or less).

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

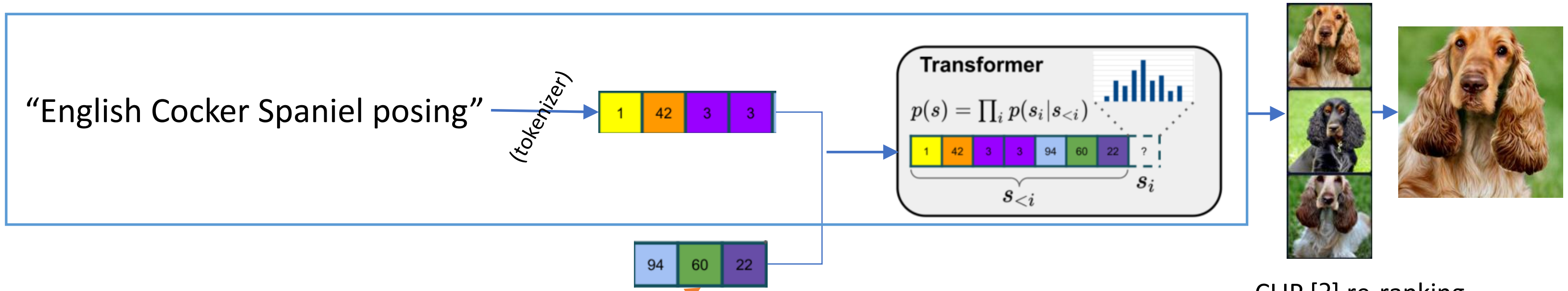
DALL·E “2” was released in 2022, *it’s more sophisticated*, and better at both quality and diversity.

“A teddybear on a skateboard in Times Square.”

Today’s topic

Intro to DALL·E models

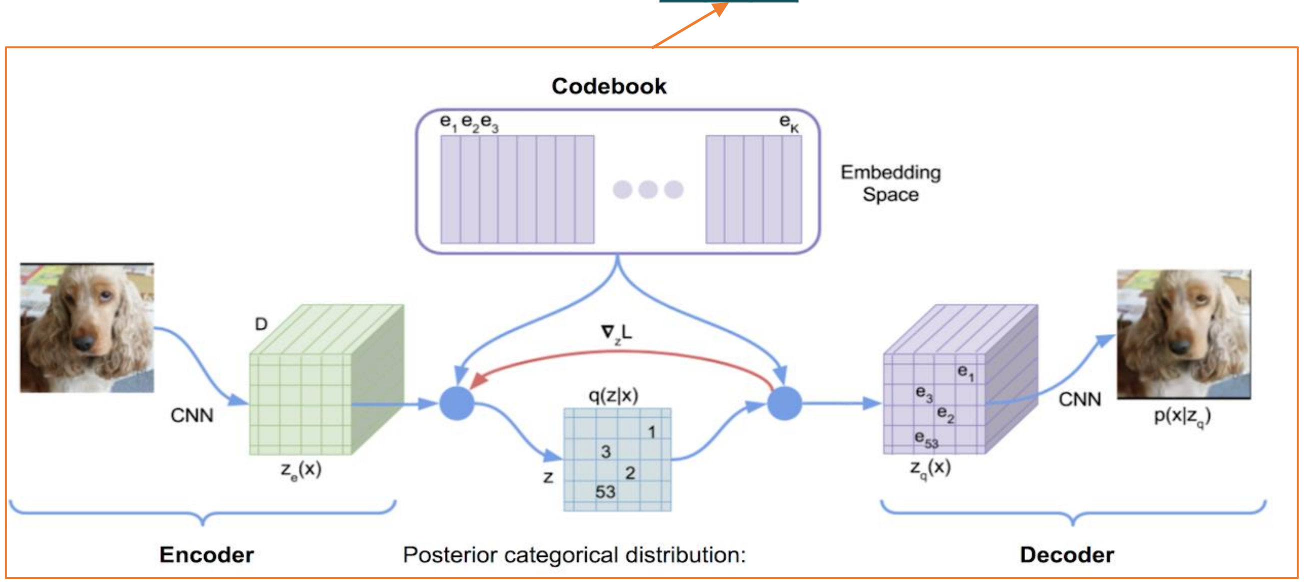
DALL·E “1” was introduced in 2021 by OpenAI, it receives text and image tokens in a single sequence, and is trained to generate image tokens auto-regressively (12b transformer).



CLIP [?] re-ranking

Stage 2:

- A transformer is trained to predict image tokens from text *and* image tokens.



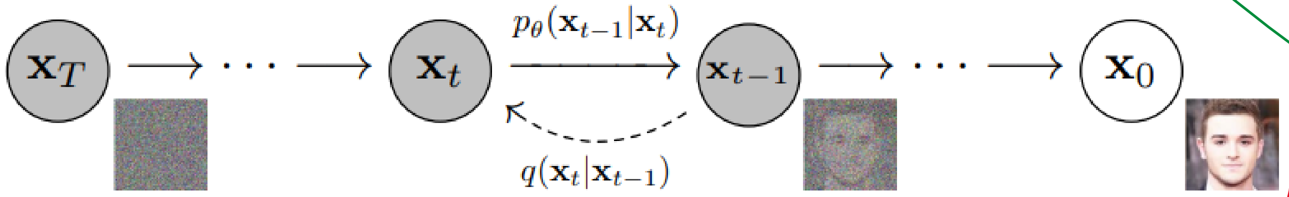
Stage 1:

- It uses a discrete VAE to compress images into a 32x32 grid of 8192 possible tokens.

- Craiyon, formerly DALL-E mini
- [2102.12092] Zero-Shot Text-to-Image Generation (arxiv.org)
- Taming Transformers for High-Resolution Image Synthesis (compvis.github.io)
- How is it so good ? (DALL-E Explained Pt. 2) - ML@B Blog (berkeley.edu)

Denoised Diffusion Probabilistic Models

Given a data distribution $x_0 \sim q(x_0)$ we design a noising process which produces *latents* $x_1 \dots x_T$



Forward: is defined such that x_T is a nearly isotropic Gaussian Distribution (cov matrix is $\sigma^2 I$):

$$q(x_t|x_{t-1}) := \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t \mathbf{I})$$

Reverse: We can start from $x_T \sim \mathcal{N}(0, \boldsymbol{I})$, but $q(x_t|x_{t-1})$ depends on the entire distribution, so we approximate it with a model:

$$p_\theta(x_{t-1}|x_t) := \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$$

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t, t)\right)$$

$$\alpha_t := 1 - \beta_t$$

Fixed to $\beta_t \boldsymbol{I}$ in [1]

[1] trains the model θ to predict ε, instead of $\mu_\theta(x_t, t)$

How do we learn this?

Few tricks:

- x_t can be sampled in closed-form using the reparameterization trick (conditioned on x_0).
- β_t defines a noise schedule, [1] uses a simple linear one.
- [2] found that learning $\Sigma_\theta(x_t, t)$ instead of fixing it to $\beta_t \boldsymbol{I}$ results in a model that requires less step for sampling.

[1] [2006.11239] Denoising Diffusion Probabilistic Models (arxiv.org)
[2] [2102.09672] Improved Denoising Diffusion Probabilistic Models (arxiv.org)

Denoised Diffusion Probabilistic Models

- **Problem**: how do we learn $p_\theta(x_{t-1}|x_t)$?
- Computing directly $p_\theta(x_0)$ requires to consider all possible forward-reversed trajectories - not feasible.
- Remember that $x_1 \ldots x_T$ are latent variables, similar to latent **Z** in VAE models
- What VAE optimizes: <u>variational lower bound</u> $\leq p_\theta(x_0)$

$$\mathbb{E}\left[-\log p_\theta(\mathbf{x}_0)\right] \leq \mathbb{E}_q\left[-\log p(\mathbf{x}_T) - \sum_{t\geq 1} \log \frac{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_t|\mathbf{x}_{t-1})}\right] =: L$$

Comes from KL divergence between q and p_θ

In practice we optimize with randomly sampled pairs of x_{t-1} and x_t

Can be sampled in closed-form

This can be further re-arranged to reduce variance during training

[2006.11239] Denoising Diffusion Probabilistic Models (arxiv.org)
What are Diffusion Models?

Comparison to other generative models

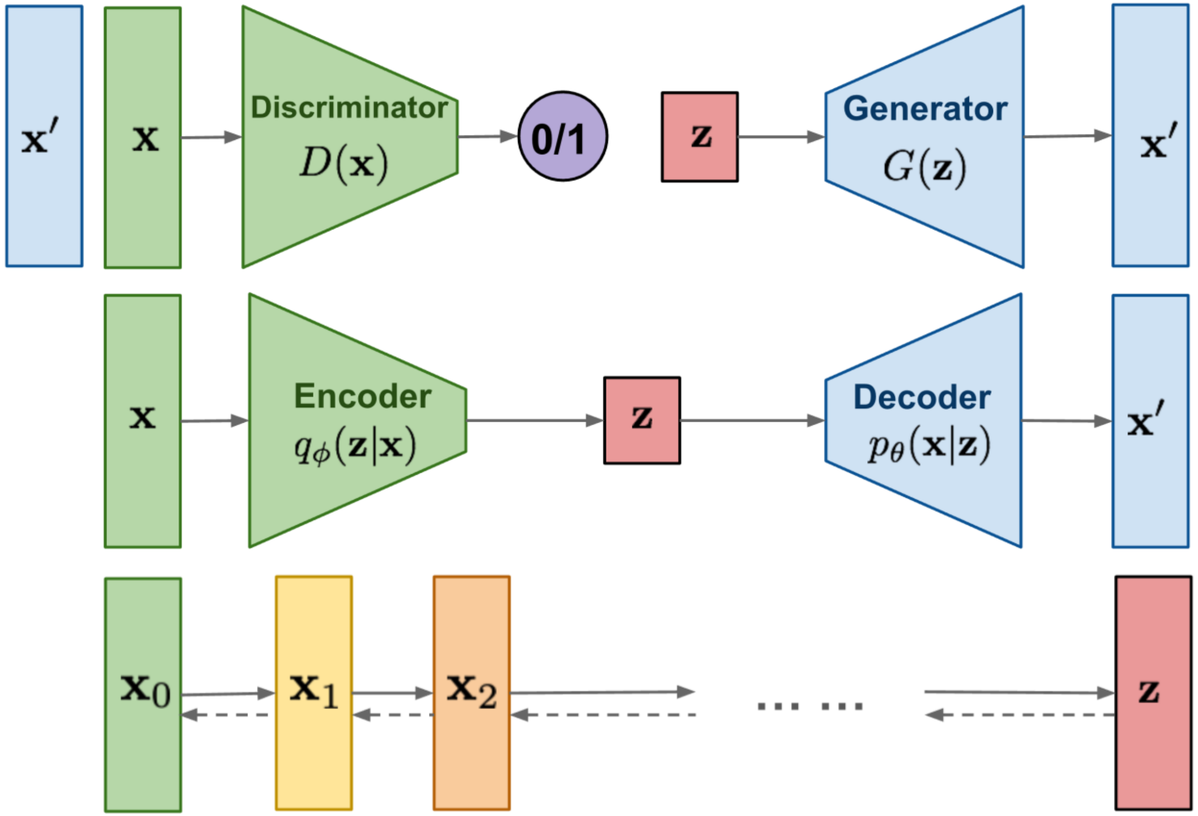
Optimizing GANs is hard™, usually confined to limited data distributions (eg faces). Still, they have high visual fidelity.

VAE uses **two** networks to generate the latent Z and to generate. In contrast, diffusion models use only **one** network for generation, and a fixed forward process for generating the latents (sequential noise).

GAN: Adversarial training

VAE: maximize variational lower bound

Diffusion models: Gradually add Gaussian noise and then reverse



[What are Diffusion Models? | Lil'Log (lilianweng.github.io)](lilianweng.github.io)

Classifier Guidance – adding text conditioning

- [1] used an auxiliary image classifier p_ϕ (trained on noised ImageNet) to guide the generative process using its gradients.
- [1] beats BigGAN on FiD scores, with more 'diverse' samples. This is also done by tweaking the UNet architecture.

Algorithm 1 Classifier guided diffusion sampling, given a diffusion model $(\mu_\theta(x_t), \Sigma_\theta(x_t))$, classifier $p_\phi(y|x_t)$, and gradient scale s.

Input: class label y, gradient scale s → y is our class to guide the generation. For example "flamingo"
$x_T \leftarrow$ sample from $\mathcal{N}(0, \mathbf{I})$
for all t from T to 1 **do**
  $\mu, \Sigma \leftarrow \mu_\theta(x_t), \Sigma_\theta(x_t)$ → Sample from the model (sometimes named the "diffusion score")
  $x_{t-1} \leftarrow$ sample from $\mathcal{N}(\mu + \underbrace{s}_{\text{Scale factor}}\Sigma \underbrace{\nabla_{x_t} \log p_\phi(y|x_t)}_{\text{Classifier gradient}}, \Sigma)$
end for
return x_0

BigGAN

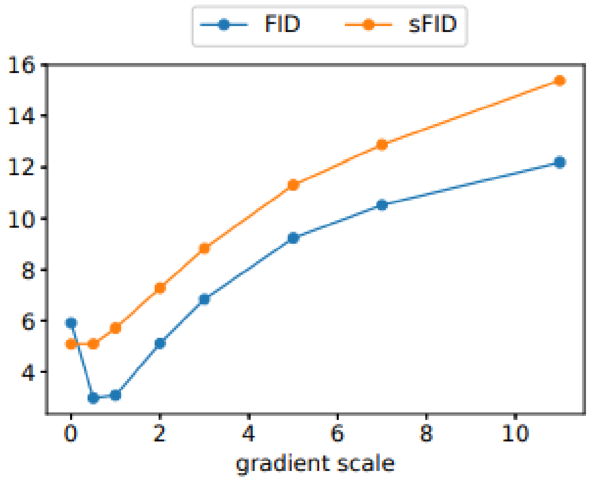
Diffusion

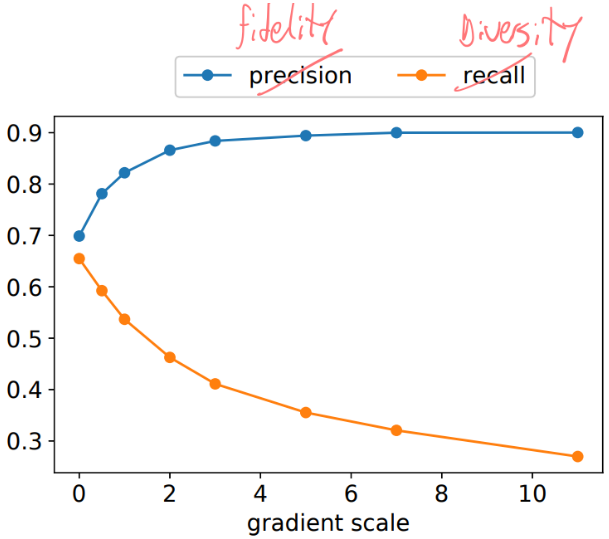
Training set

[1] [2105.05233] Diffusion Models Beat GANs on Image Synthesis (arxiv.org)

Classifier Guidance - adding text conditioning

- Increasing the classifier gradients hits a trade-off between fidelity and diversity.
- The ‘optimal’ value for FiD and sFID scores is in between.





A note in eval metrics:

- FID [3] compares the distribution of generated images (given by the Inception model) and the images in the training set.
- Precision and Recall here refer to [2]:

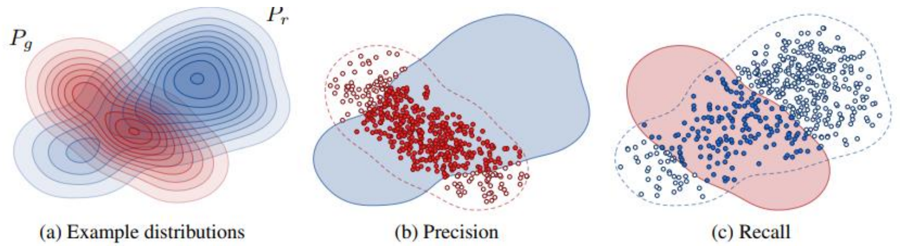


Figure 1: Definition of precision and recall for distributions [25]. (a) Denote the distribution of real images with P_r (blue) and the distribution of generated images with P_g (red). (b) Precision is the probability that a random image from P_g falls within the support of P_r. (c) Recall is the probability that a random image from P_r falls within the support of P_g.

BigGAN

Diffusion

Training set

[1] [2105.05233] Diffusion Models Beat GANs on Image Synthesis (arxiv.org)
[2] [1904.06991] Improved Precision and Recall Metric for Assessing Generative Models (arxiv.org)
[3] [1706.08500] GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium (arxiv.org)

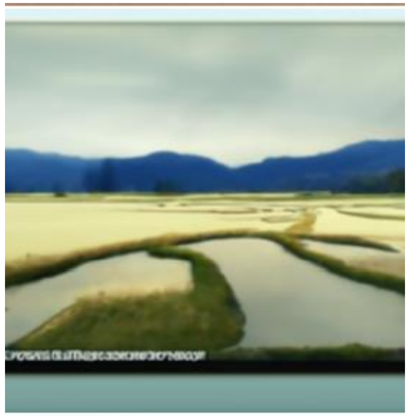
Classifier Guidance – CLIP (image and text aligned embeddings)

- CLIP naturally resembles a ‘zero-shot’ classifier.
- Although in theory guidance requires a classifier trained in noised data, [2] used CLIP public models with some level of success.
- [2] and [3] are important since were projects developed “on the open”, arguably influenced Stable Diffusion.

"A painting of an apple“
(CLIP-guided diffusion)

"a futuristic city in synthwave style“
VQGAN-CLIP

“Rice farming by Hokusai Gogh”



GLIDE (CLIP-guided)

GLIDE (guided without CLIP)

VQGAN-CLIP

We will talk about classifier-free guidance next

[2] CLIP Guided Diffusion HQ 256x256.ipynb - Colaboratory (google.com)
[3] [2204.08583v1] VQGAN-CLIP: Open Domain Image Generation and Editing with Natural Language Guidance (arxiv.org)
[4] GitHub - nerdyrodent/CLIP-Guided-Diffusion: Just playing with getting CLIP Guided Diffusion running locally, rather than having to use colab.

Classifier-free guidance (still text-conditioned)

- Classifier guidance was a great fix to be able to trade-off diversity by fidelity.
- Authors argue classifier guidance boosts FiD scores more-and-less artificially.
- [1] proposes a simple trick to still control diversity/fidelity without another model:
  - Used paired data (x, y) during training, typically image captions
  - Train an **unconditional** diffusion model $p_\theta(x)$, and a **conditional** diffusion model $p_\theta(x|y)$
  - Use a single network to represent $p_\theta(x)$ and $p_\theta(x|y)$. Note that p_θ is simply the same diffusion model.
  - Practically speaking, $p_\theta(x|y)$ is trained and periodically y is simply discarded (set to zeroes)
  - The parametrized outputs ($\boldsymbol{\epsilon}_\theta$) of $p_\theta(x)$ and $p_\theta(x|y)$ are weighted:

$$\tilde{\boldsymbol{\epsilon}}_\theta(\mathbf{z}_\lambda, \mathbf{c}) = (1 + w)\boldsymbol{\epsilon}_\theta(\mathbf{z}_\lambda, \mathbf{c}) - w\boldsymbol{\epsilon}_\theta(\mathbf{z}_\lambda)$$

Increase guidance,
more fidelity

Decrease guidance,
More diversity

[1] [2207.12598] Classifier-Free Diffusion Guidance (arxiv.org)

GLIDE – a model before DALLE-2 *with diffusion*

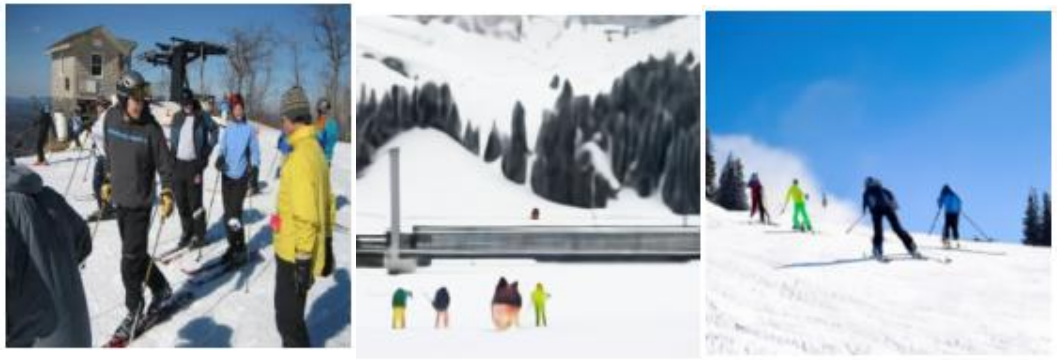
- Rivals DALLE "1": 3b parameters vs DALLE's 12b.
- GLIDE reports that classifier-free is preferred over CLIP-guidance by *human evaluators*.
- The condition on text is done via attention and token embeddings from a text-transformer.
- GLIDE's samples are preferred over DALLE's by *human evaluators* (89% in photorealism, and 69% in caption similarity).
- Training details:
  - 3.5 billion parameter text-conditional diffusion model, at 64x64 resolution
    - 2.3b for the visual part
    - 1.5b for a transformer encoding the txt
  - 1.5 billion parameter up-sampling diffusion model, at 256x256 resolution
  - Same dataset as DALLE-1, and roughly the same compute
  - Extra fine-tuning for *unconditional* image generation, and for *inpainting* (random masks added in a 4^{th} channel)

[1] [2112.10741] GLIDE: Towards Photorealistic Image Generation and Editing with Text-Guided Diffusion Models (arxiv.org)

GLIDE – a model before DALLE-2

Conditional generation:

Real image **DALLE-1** **GLIDE**

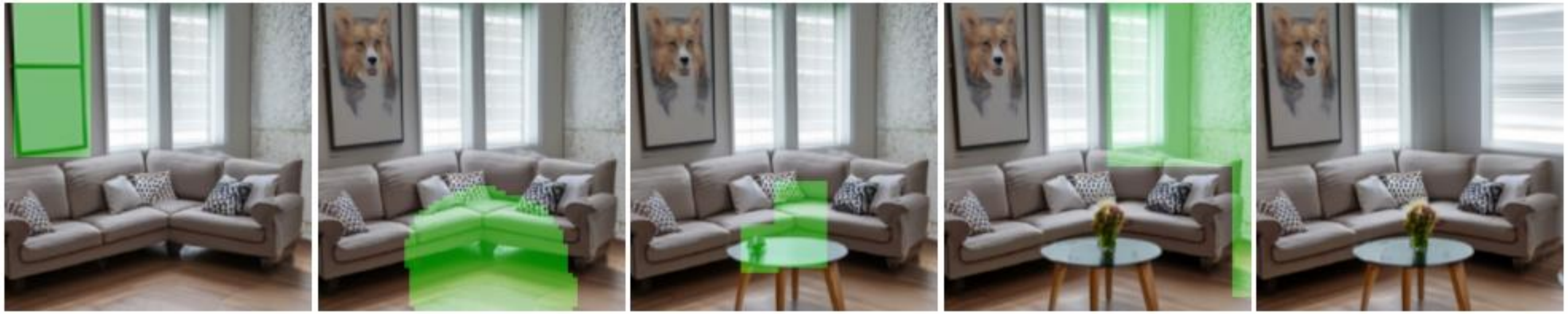


"a group of skiers are preparing to ski down a mountain."

Real image **DALLE-1** **GLIDE**

"a small kitchen with a low ceiling"

Inpainting: iteratively add a mask to the model. The first image is generated from the prompt alone.



"a cozy living room" "a painting of a corgi on the wall above" "a round coffee table in front of a couch" "a vase of flowers on a coffee table" "a couch in the corner of a room"

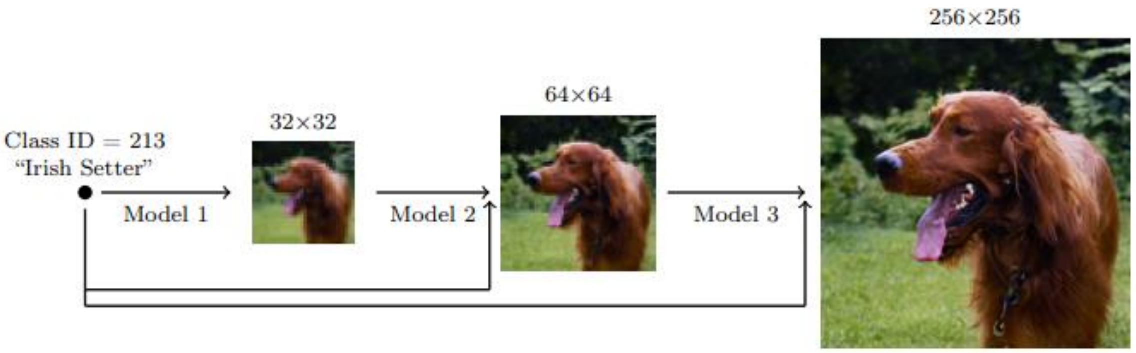
[1] [2112.10741] GLIDE: Towards Photorealistic Image Generation and Editing with Text-Guided Diffusion Models (arxiv.org)

Summary so far – before DALLE-2

- Diffusion models are generative models that generate noised latents, and then are denoised iteratively.
- Conditional generation can be enabled with classifiers (CLIP, etc)…
- … However, classifier-free guidance is a established trick to do so without extra classifiers.
- DALLE-1 uses a discrete VAE and a transformer to generate image tokens.
- Diffusion-based conditional models beat GANs
- GLIDE, a diffusion-based model with the ‘tricks’ above, beats GANs and DALLE-1. <u>However, it’s resource-intensive at inference.</u>

DALL·E 2 (finally) - Basics

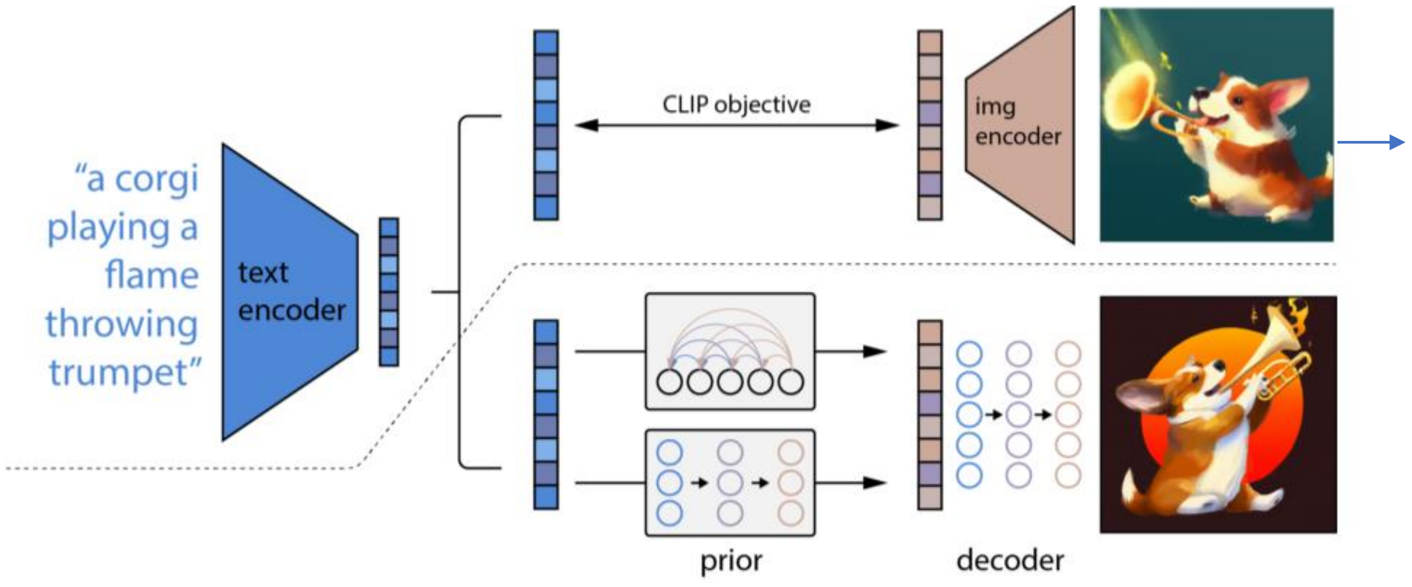
- **CLIP**: a model that maps text and images to the same embedding space
- **Auto-regressive model (AR)**: A sequence model that generate tokens causally.
- **Diffusion model (DM)**: a generative model which uses iterative denoising to learn a data distribution (optionally conditioned)
- **Upsampling:** cascaded diffusion models can increase resolution, typically using U-net:



[2106.15282] Cascaded Diffusion Models for High Fidelity Image Generation (arxiv.org)

DALL·E 2 - unCLIP

unCLIP is a two stages model. The goal is to “invert” the CLIP text embeddings.



Stage 1: Generate a CLIP image embedding

Stage 2: Generate the image from the embedding.

CLIP Pretraining

CLIP’s training objective only forces the embeddings to match an image to a caption, but it does not necessarily captures image features describable by text, relative positions, etc.

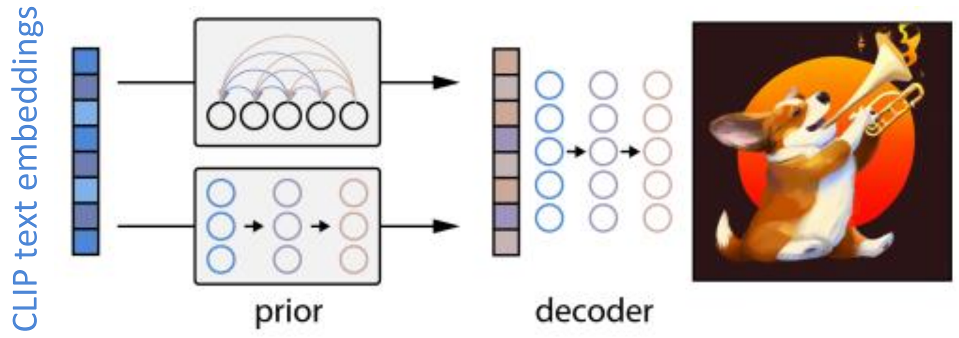
By training the prior now the produced embeddings should capture salient features of the image, rather than just its relationship to the caption.

The decoder has 2 upsamplers to go from 64x64->256x256->1024x1024

DALL·E 2 - unCLIP

Given image-caption pairs (x, y):

- z_i = CLIP image embeddings
- z_t = CLIP text embeddings



- **Prior**: $P(z_i, y)$ produces z_i , conditioned on captions y.
- **Decoder**: $P(x|z_i, y)$ produces an image x conditioned on z_i and (optionally) y.

Note that even though not explicitly mentioned here, the prior can be conditioned on z_t because z_t is a deterministic function of y.

Model: $P(x|y) = P(x, z_i \mid y) = P(x|z_i, y)P(z_i, y)$

[[2204.06125] Hierarchical Text-Conditional Image Generation with CLIP Latents (arxiv.org)](https://arxiv.org/abs/2204.06125)

DALL·E 2 – the prior

Why do we need a prior?
CLIP's objective is not enough to align the embeddings for image generation, the prior aligns them correctly for this.

Two models were tried in the paper for the Prior (only one is used):

- **Auto-regressive (AR)**: The embeddings are discretized and generated left-ot-right conditioned on the captions.
- **Diffusion (DM)**: the embedding z_i is directly produced by a diffusion model conditioned on y.

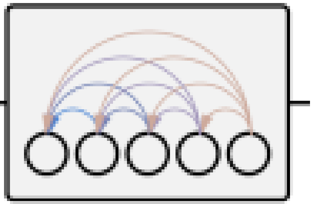
The DM prior outperforms AR in human eval, FID (MS-COCO), and "aesthetic".

AR: dot product is added during training and in inferencing a fixed value is hardcoded in the input.

$z_i \rightarrow$ PCA $\rightarrow$ quantization

Quantized dot product | Caption | Text embedding

$[z_i \cdot z_t]$ + [a, corgi] + $[z_t]$

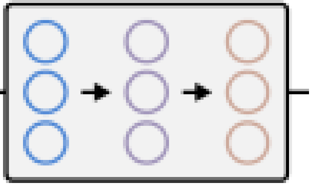


$\hat{z}_i$

DM: instead of adding dot product, two z_i are sampled and the best is chosen.

Caption | Text embedding | Diffusion Step embedding | Diffusion step latent | Final embedding

[a, corgi] + $[z_t]$ + $[h_t]$ + $[x_t]$ + $[r]$



$\hat{z}_i$

$decoder(y)$
no prior

$decoder(y, z_t)$
no prior

$decoder(y, \hat{z}_i)$
with prior

"an oil painting of a corgi wearing a party hat"

[2204.06125] Hierarchical Text-Conditional Image Generation with CLIP Latents (arxiv.org)

DALL·E 2 – the decoder

The decoder is very similar to GLIDE, except that it includes the CLIP embeddings $\widehat{Z}_i$ <u>generated by the prior</u>.

$\hat{z}_i W_1$

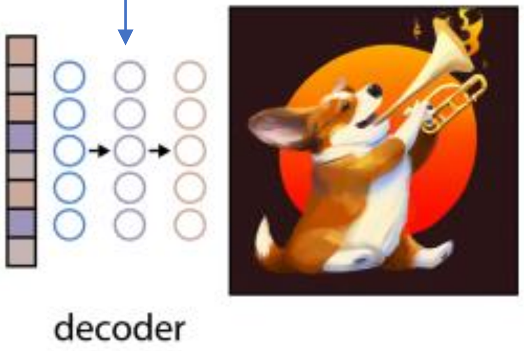
[t1, t2, t3, t4] + encode([a, corgi])

Classifier-free guidance:

- The CLIP embeddings are dropped 10% of the time.
- The text caption is dropped 50% of the time.

$\hat{z}_i W_2$

The projected image embedding is added to the diffusion step embedding too.

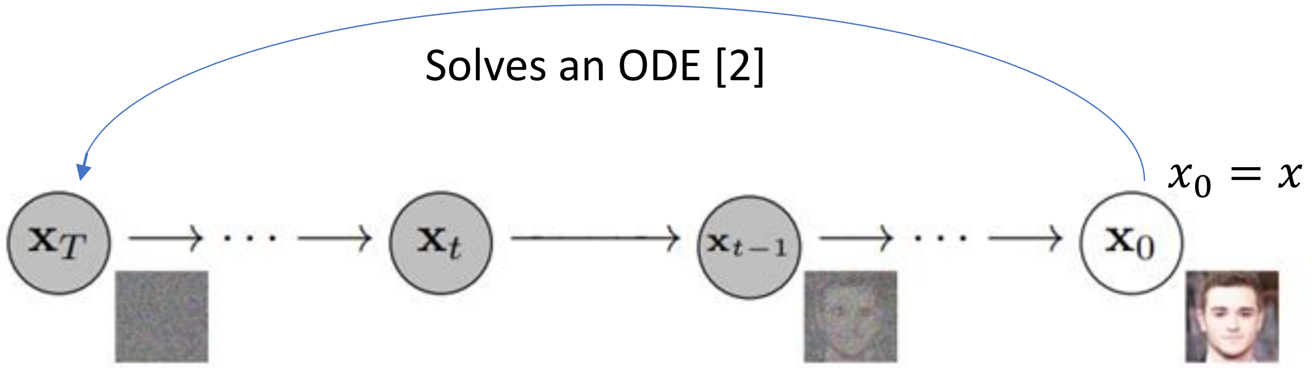


[[2204.06125] Hierarchical Text-Conditional Image Generation with CLIP Latents (arxiv.org)](https://arxiv.org/abs/2204.06125)

Image manipulations – Inverse DDIM

- x = any image
- $z_i = \text{CLIP\_image\_encoder}(x)$
- x_T: latent given X, computed as:

We can now generate images conditioned on z_i and also manipulate the latent x_T (next).



The unclip decoder inverse DDIM

[1] [2204.06125] Hierarchical Text-Conditional Image Generation with CLIP Latents (arxiv.org)
[2] [2105.05233] Diffusion Models Beat GANs on Image Synthesis (arxiv.org)

Image manipulations – Inverse DDIM

Variation: fix z_i, change noise in x_T:

Interpolation: Interpolate between
$z_i \rightarrow z'_i$ and $x_T \rightarrow x'_T$:

Diffs: get normalized difference of captions, and use it to interpolate with z_i, while varying x_T

a photo of a landscape in winter $\longrightarrow$ a photo of a landscape in fall

[1] [2204.06125] Hierarchical Text-Conditional Image Generation with CLIP Latents (arxiv.org)

How good is DALLE-2? - Metrics

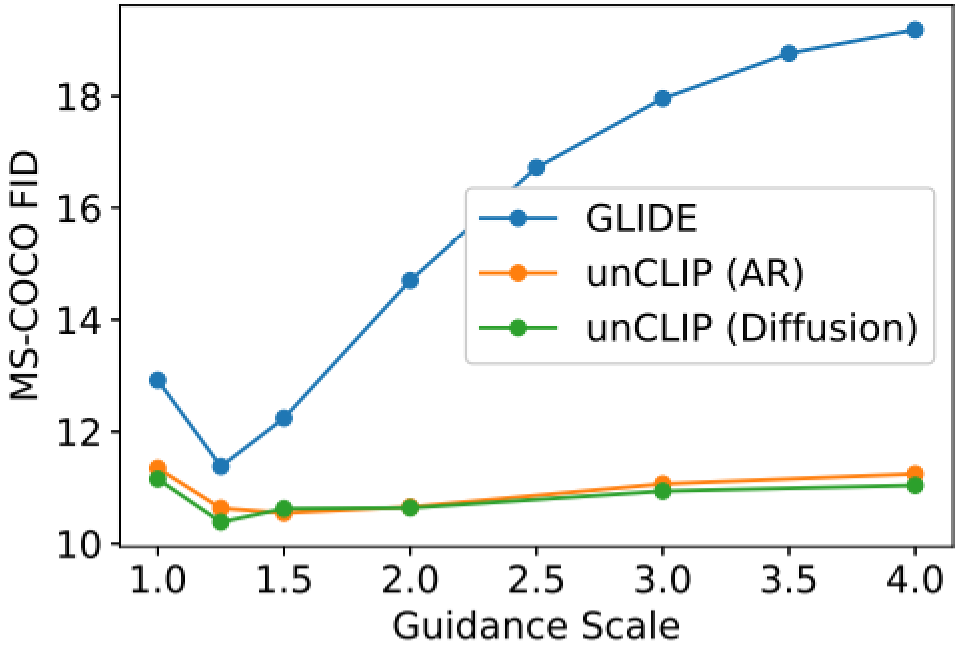
% of human evaluators that prefer unCLIP over GLIDE

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

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.

FiD score (lower is better, more 'realistic')



Limitations

- DALLE-2 has a hard time with variable binding.
- CLIP's inductive bias doesn't bind linguistic properties from image to text, and neither does unCLIP.
- The authors hoped that conditioning on encoded text in diffusion would help, but it didn't

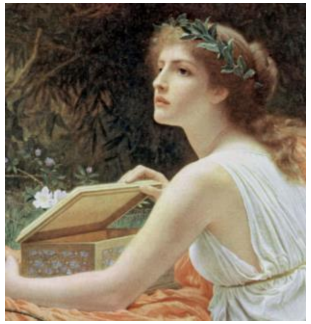
"A corgi with a green bow tie and a red party hat"

"a sign that says deep learning" . It's possible the text tokenization procedure makes it very difficult for clip.

High-level details are hard, likely due to the resolution up-sampling.

(b) A high quality photo of Times Square.

Risks, datasets, and copyright



- As usual with large-pretrained models, the datasets used are very large and with a lot of problematic content (hate-speech, stereotypes, etc), usually with poor/none filtering. These models can be easily/cheaply weaponized. Images have higher impact than text?

- Available models (DALLE-2, SD) "patch" this with a binary classifier to prevent misuse, it's effectiveness is debatable.

- There's a growing concern on artists about having their work used on these massive models without consent -> legal loophole. Some US rulings argue scrapped datasets are fair-use [2].

Very important problems, **no effective solutions thus far**, *models are in the open now…*

[1] Multimodal datasets: misogyny, pornography, and malignant stereotypes

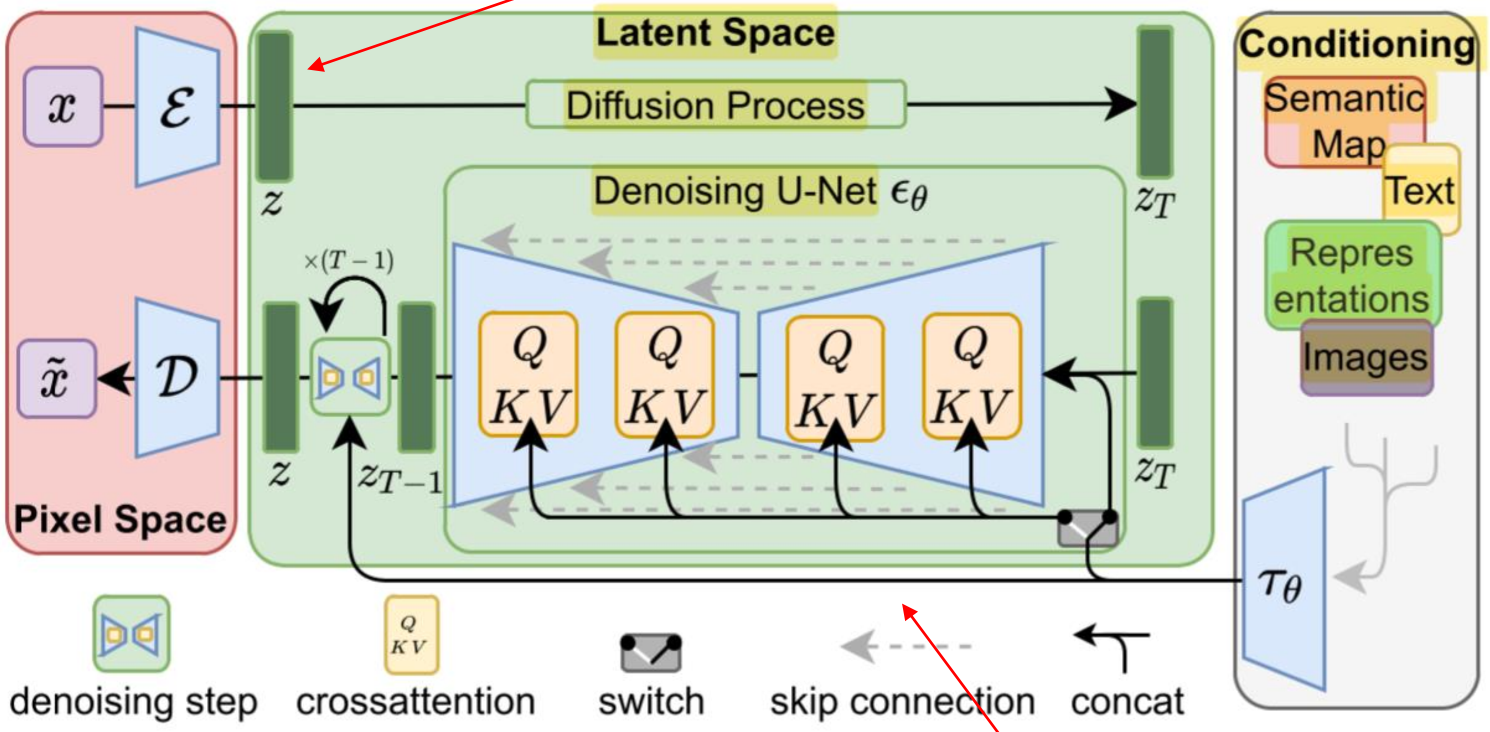
[2] https://en.wikipedia.org/wiki/Authors_Guild,_Inc._v._Google,_Inc.

~~Stable~~ Latent Diffusion (SD)

4 in the paper, 8 in the widely released "stable diffusion"

- Trained on "LAION Aesthetics"
- Re-introduces latent spaces, similar to the codebook of DALLE-1, which used a dVAE.
- The encoder $\mathcal{E}$ has a strong image inductive bias (~VQGAN).
- The latent now can exploit these inductive biases to do diffusion more effectively, instead of "naïve" AR (DALLE-1).
- Models are 0.8b – 1.45B parameters, noticeably smaller than DALLE (3.5b and 12b).
- **Model weights and code are open-source.**

1) Diffusion is done **in a compressed** space (down sampling factor of 4), not pixel space. This is why SD is noticeably faster.



2) Cross-attention coming from the output of a domain-specific encoder τ_θ and the UNet backbone.

~~Stable~~ Latent Diffusion (SD)

Conditioning in Latent spaces:

[quant_top_left, quant_bottom, class]

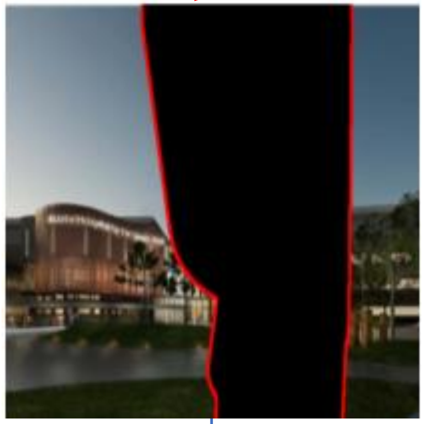
↓

τ_θ Transformer

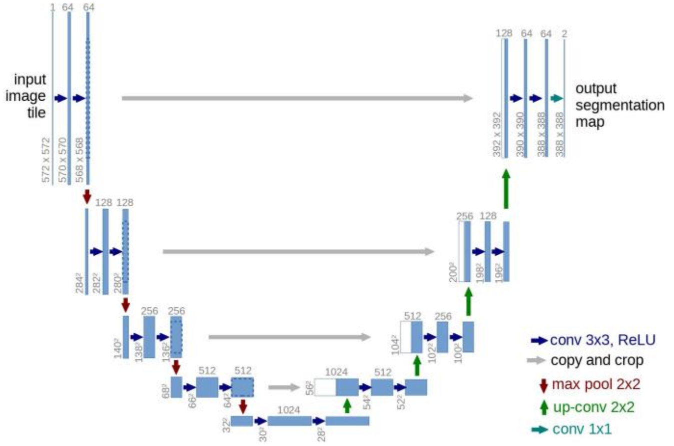
↓

Cross-attention in UNet

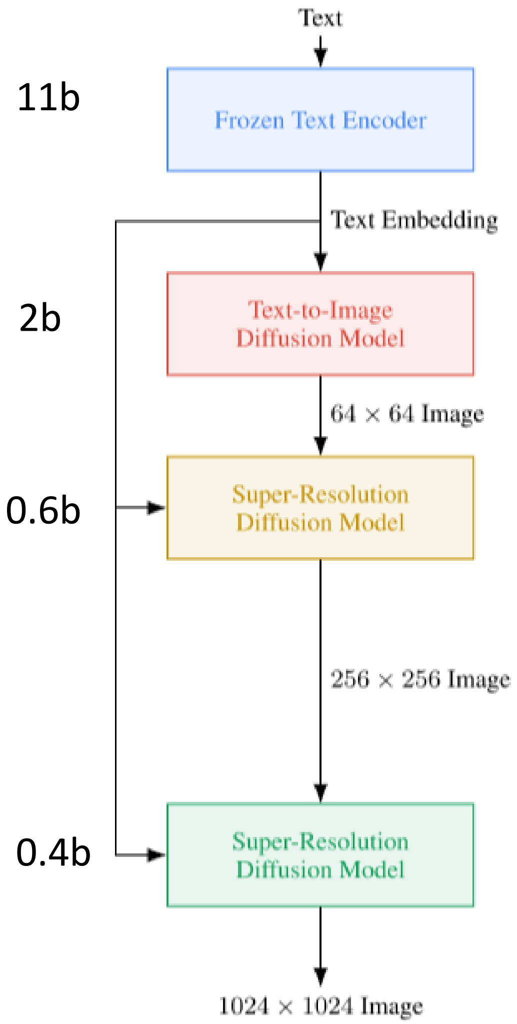
"spatially aligned information" it's simply concatenated to the input of the UNet network (which already has the diffusion latent).

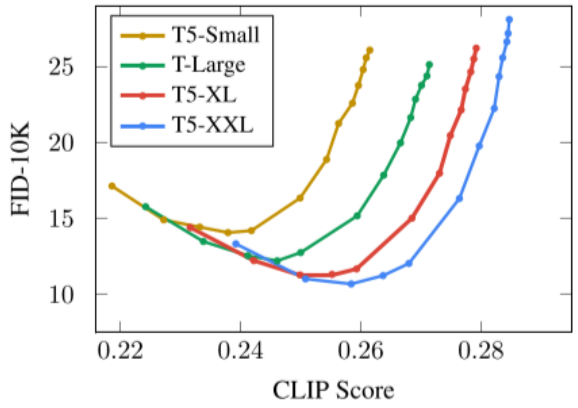


The Unet architecture. Cross-attention is applied on intermediate <u>flattened</u> representations.

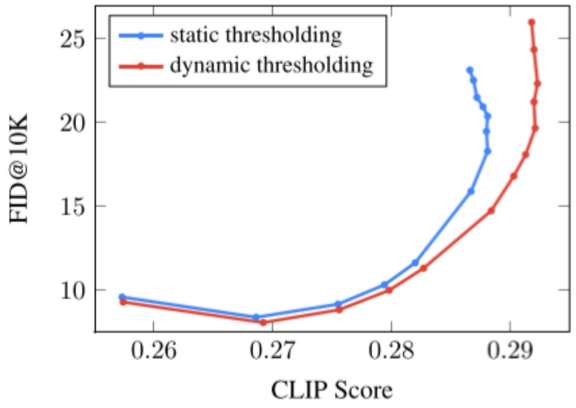


SOTA on Diffusion Models – Google's Imagen



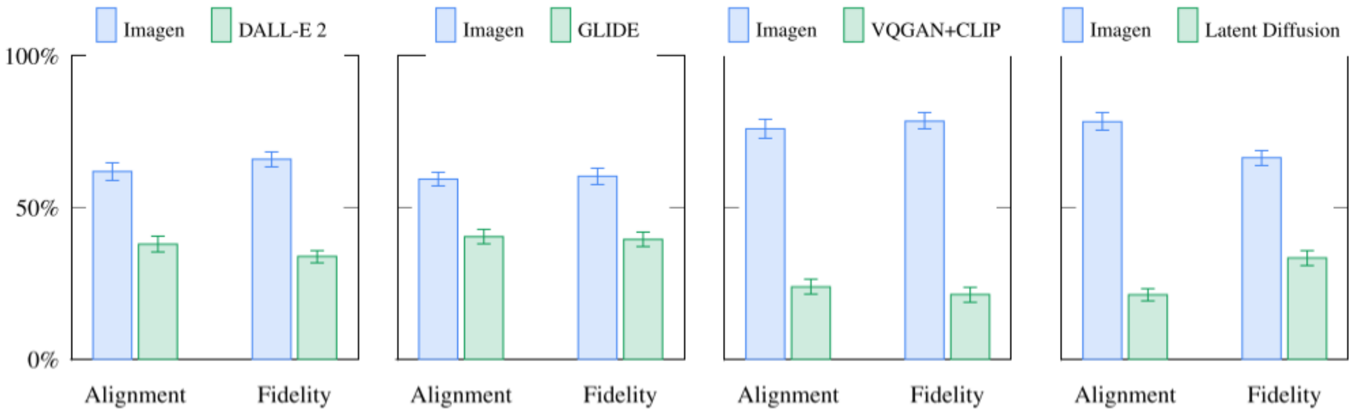


(a) Impact of encoder size.



(c) Impact of thresholding.

Dramatically simpler architecture, it simply uses a very big text encoder followed by diffusion for super-resolution, for a total of 14b parameters.



Beats GLIDE and DALL-E 2 in human evaluation benchmarks and FiD.

SOTA on Diffusion Models – Google's Imagen

Imagen DALLE-2

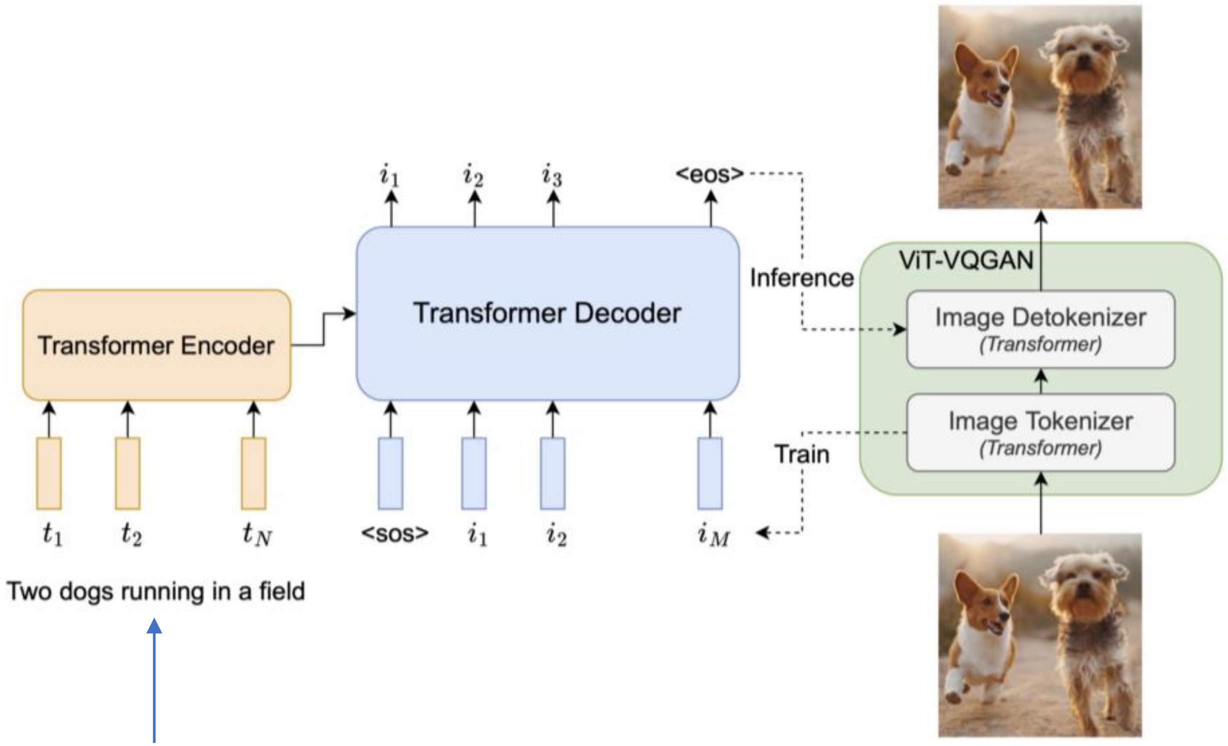
A storefront with Text to Image written on it.

Imagen DALLE-2

A panda making latte art.

Parti – scratch Diffusion Models

The bitter lesson strikes again: scratch diffusion models and use a simpler, and scaled-up architecture (20B model):



Pre-Trained with BERT's loss and contrastive image-text data

- Stage 1: tokenize images using ViT-VQGAN (30M parameters).
- Stage 2: Train an encoder-decoder model to generate image tokens given text tokens.
- Stage 3: Scale up the image detokenizer to 600M parameters and higher resolution.
- Add a super-resolution model on top to reach 1024x1024.
- Also uses classifier-free guidance.

Parti – scratch diffusion models

Approach	Model Type	MS-COCO Zero-shot	Fi	
Random Train Images [10]	-		2.47	
Retrieval Baseline	-	17.97		
TReCS [46]	GAN	-		
XMC-GAN [47]	GAN	-		
DALL-E [2]	Autoregressive	~28		
CogView [3]	Autoregressive	27.1		
CogView2 [61]	Autoregressive	24.0		
GLIDE [11]	Diffusion	12.24		12b
Make-A-Scene [10]	Autoregressive	11.84		
DALL-E 2 [12]	Diffusion	10.39		3b
Imagen [13]	Diffusion	**7.27**		11b
Parti	Autoregressive	**7.23**		20b

"a propaganda poster depicting a cat dressed as french emperor napoleon holding a piece of cheese"

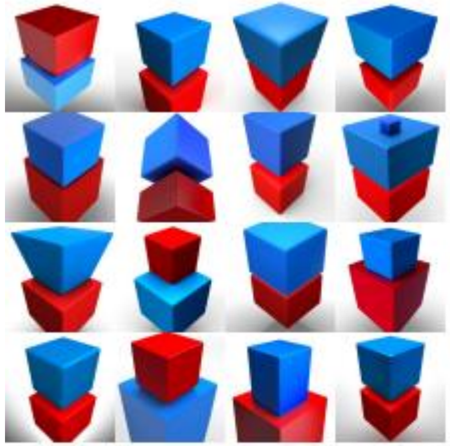
Can you tell which one is Parti and which one is DALLE-2? ☺

Takeaway: scale matters! But diffusion is *probably* a good generic compression technique, time will tell (as with the transformer… or with GANs)

Summary

- Diffusion models are a powerful compression/denoising technique
- Small diffusion models seem to be very good, but scaling up still pays off.
- Classifier-free guidance > CLIP-guidance
- Aligning text and image embedding spaces is probably easier than what we thought before.
- No one has solved variable binding, counting, negation, spatial relations, grounding, etc… Language is still unsolved ☺

A red cube **on top** of a blue cube



a plate that has **no bananas** on it. there is a glass **without** orange juice next to it

two baseballs to the left of **three** tennis balls

rhino beetle this size of a tank grapples a real life passenger airplane on the tarmac