Recent Trends in Machine Learning: A Large-scale Perspective

A Short Introduction to Multi-modal AI Models (Part 3)

Saehoon Kim @ Kakaobrain

Outline of This Course

CLIP
Encoder-only

05/04

DALL-E
Decoder-only

05/11

DALL-E 2
Enc-Dec

05/18

Outline of This Course

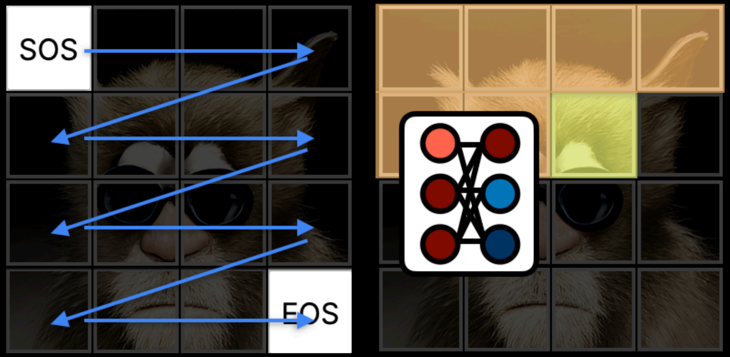
Contrastive Learning

Autoregressive Model

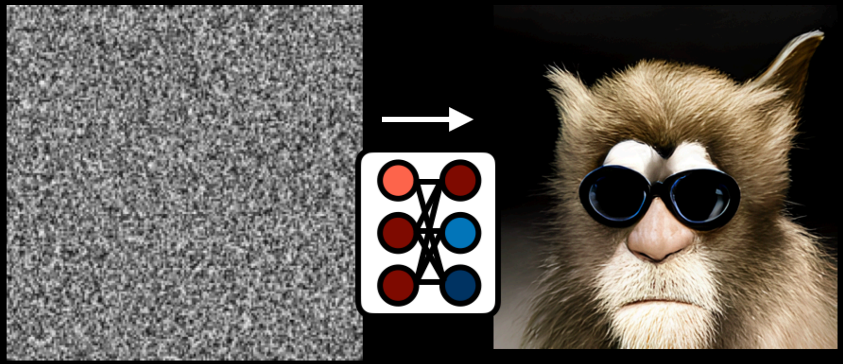
Diffusion Model

AR vs. Diffusion

Autoregressive Model

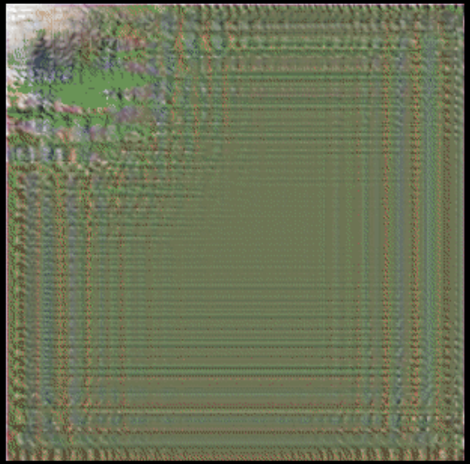


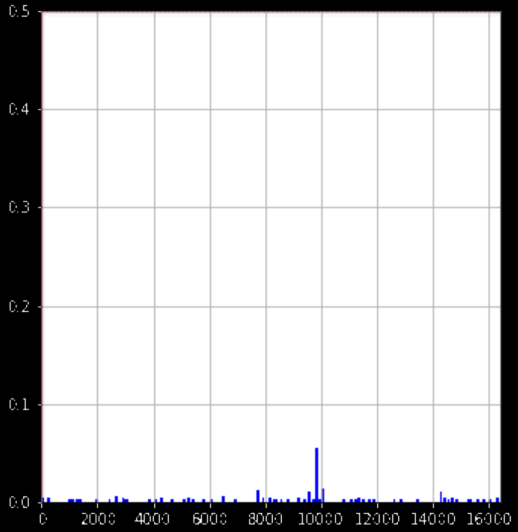
Diffusion Model



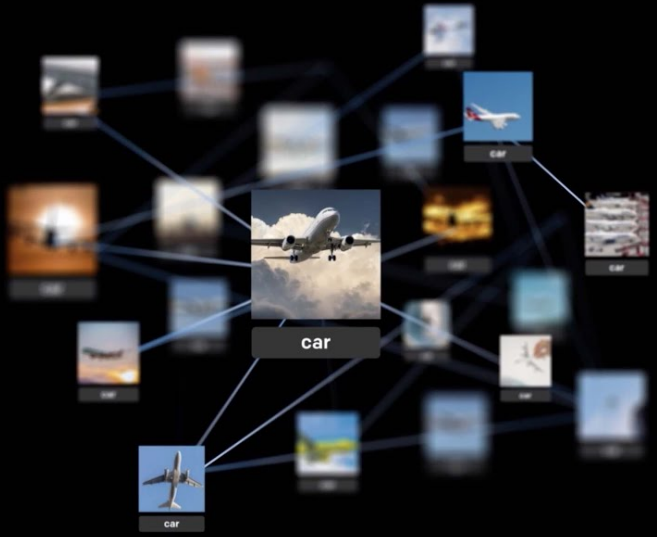
DALL-E 1 (AR Model)

A painting of a cherry blossom tree





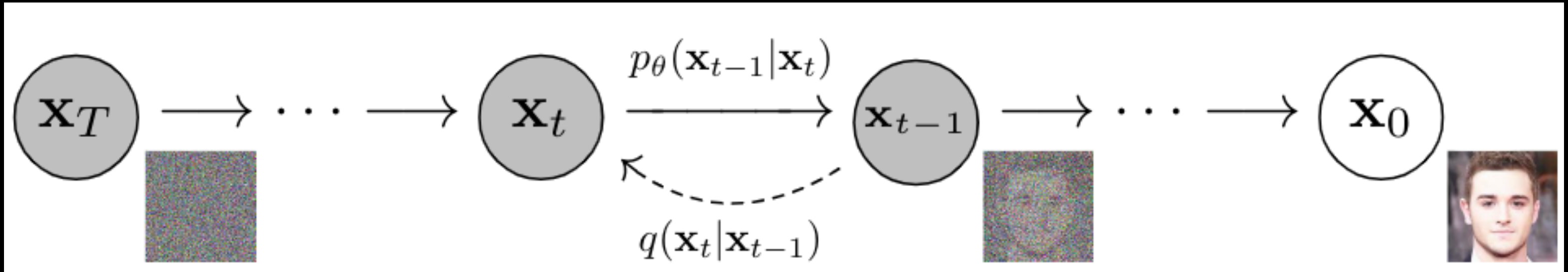
DALL-E 2 (Diffusion Model)



From OpenAI's official page

DDPM: Denoising Diffusion Probabilistic Models

Diffusion models are latent variables models defined by diffusion (forward) process and reverse process



J. Ho, A. Jain and P. Abbeel . Denoising Diffusion Probabilistic Models, NeurIPS'20.

Diffusion Process

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

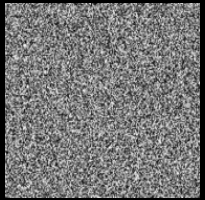
Diffusion Process

When beta is sufficiently small, this forward process can be approximated by a Gaussian distribution in the reverse process

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

Reverse Process

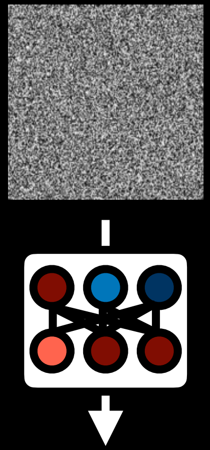
$$p(\mathbf{x}_T) = \mathcal{N}(\mathbf{x}_T | \mathbf{0}, \mathbf{I})$$



Reverse Process

$$p(\mathbf{x}_T) = \mathcal{N}(\mathbf{x}_T | \mathbf{0}, \mathbf{I})$$

$$p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \sigma_t^2 \mathbf{I})$$

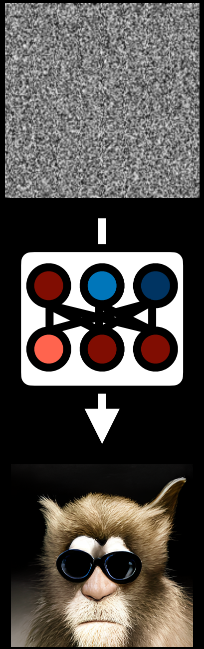


Reverse Process

$$p(\mathbf{x}_T) = \mathcal{N}(\mathbf{x}_T | \mathbf{0}, \mathbf{I})$$

$$p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \sigma_t^2 \mathbf{I})$$

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



Optimization (1/2)

Parameters of reverse process can be learned by optimizing the standard ELBO

$$\mathbb{E}[-\log p_\theta(\mathbf{x}_0)] \geq \mathbb{E}_q \left[-\log \frac{p_\theta(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}\right]$$

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

Optimization (2/2)

Parameters of reverse process can be learned by optimizing the standard ELBO

$$\mathbb{E}_q\Big[\underbrace{D_{\mathrm{KL}}[q(\mathbf{x}_T|\mathbf{x}_0)\|p(\mathbf{x}_T)]}_{L_T}$$

$$+\sum_{t>1}\underbrace{D_{\mathrm{KL}}[q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)\|p(\mathbf{x}_t)]}_{L_{t-1}}\underbrace{-\log p_\theta(\mathbf{x}_0|\mathbf{x}_1)}_{L_0}\Big]$$

Optimization (Simplified Version)

Through its reparmeterization, the objective simplifies to

$$L_{t-1} = \mathbb{E}_{\mathbf{x}_0,\boldsymbol{\epsilon}}\left[\frac{\beta_t^2}{2\sigma_t^2\alpha_t(1-\bar{\alpha}_t)}\|\boldsymbol{\epsilon}-\boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}}\mathbf{x}_0+\sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon},t)\|_2^2\right]$$

Optimization (Simplified Version)

Through its reparmeterization, the objective simplifies to

$$L_{t-1} = \mathbb{E}_{\mathbf{x}_0,\boldsymbol{\epsilon}}\left[\frac{\beta_t^2}{2\sigma_t^2\alpha_t(1-\bar{\alpha}_t)}\|\boldsymbol{\epsilon}-\boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}}\mathbf{x}_0+\sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon},t)\|_2^2\right]$$

Optimization (Simplified Version)

Through its reparmeterization, the objective simplifies to

$$L_{t-1} = \mathbb{E}_{\mathbf{x}_0, \boldsymbol{\epsilon}} \left[\frac{\beta_t^2}{2\sigma_t^2 \alpha_t (1 - \bar{\alpha}_t)} \| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta (\sqrt{\bar{\alpha}} \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \boldsymbol{\epsilon}, t) \|_2^2 \right]$$

Training / Sampling

Algorithm 1 Training

repeat
  $\mathbf{x}_0 \sim q(\mathbf{x}_0)$
  $t \sim \mathrm{Uniform}(\{1, \dots, T\})$
  $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  Take gradient descent step on
  $$\nabla_\theta \left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t) \right\|^2$$
until converged

Algorithm 2 Sampling

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

Training / Sampling

Algorithm 1 Training

repeat
  $\mathbf{x}_0 \sim q(\mathbf{x}_0)$
  $t \sim \mathrm{Uniform}(\{1, \ldots, T\})$
  $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  Take gradient descent step on
    $\nabla_\theta \left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t) \right\|^2$
until converged

Algorithm 2 Sampling

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

Experiments

Compared to AR models, DDPM generates samples in a bi-directional manner!

Table 1: CIFAR10 results. NLL measured in bits/dim.

Model	IS	FID	NLL Test (Train)
Conditional			
EBM [11]	8.30	37.9	
JEM [17]	8.76	38.4	
BigGAN [3]	9.22	14.73	
StyleGAN2 + ADA (v1) [29]	**10.06**	**2.67**	
Unconditional			
Diffusion (original) [53]			≤ 5.40
Gated PixelCNN [59]	4.60	65.93	3.03 (2.90)
Sparse Transformer [7]			**2.80**
PixelIQN [43]	5.29	49.46	
EBM [11]	6.78	38.2	
NCSNv2 [56]		31.75	
NCSN [55]	8.87±0.12	25.32	
SNGAN [39]	8.22±0.05	21.7	
SNGAN-DDLS [4]	9.09±0.10	15.42	
StyleGAN2 + ADA (v1) [29]	**9.74** ± 0.05	3.26	
Ours (L, fixed isotropic $\mathbf{\Sigma}$)	7.67±0.13	13.51	≤ 3.70 (3.69)
Ours (L_{simple})	9.46±0.11	**3.17**	≤ 3.75 (3.72)

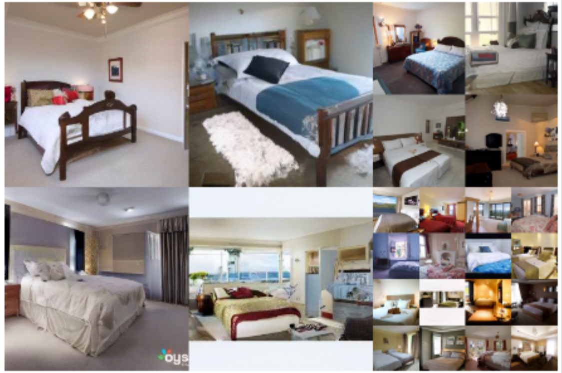


Figure 4: LSUN Bedroom samples. FID=4.90

Experiments

Compared to AR models, DDPM generates samples in a bi-directional manner!

Figure 6: Unconditional CIFAR10 progressive generation ($\hat{\mathbf{x}}_0$ over time, from left to right). Extended samples and sample quality metrics over time in the appendix (Figs. 10 and 14).

GLIDE: Guided Language to Image Diffusion for Generation and Editing

Class-conditional diffusion models can be implemented by classifier guidance

$$\hat{\mu}_\theta(x_t|y) = \mu_\theta(x_t|y) + s \cdot \Sigma_\theta(x_t|y)\nabla_{x_t} \log p_\phi(y|x_t)$$

GLIDE: Guided Language to Image Diffusion for Generation and Editing

Classifier-free guidance for removing the need of a separate classier

$$\hat{\epsilon}_\theta(x_t|y) = \epsilon_\theta(x_t|\emptyset) + s \cdot (\epsilon_\theta(x_t|y) - \epsilon_\theta(x_t|\emptyset))$$

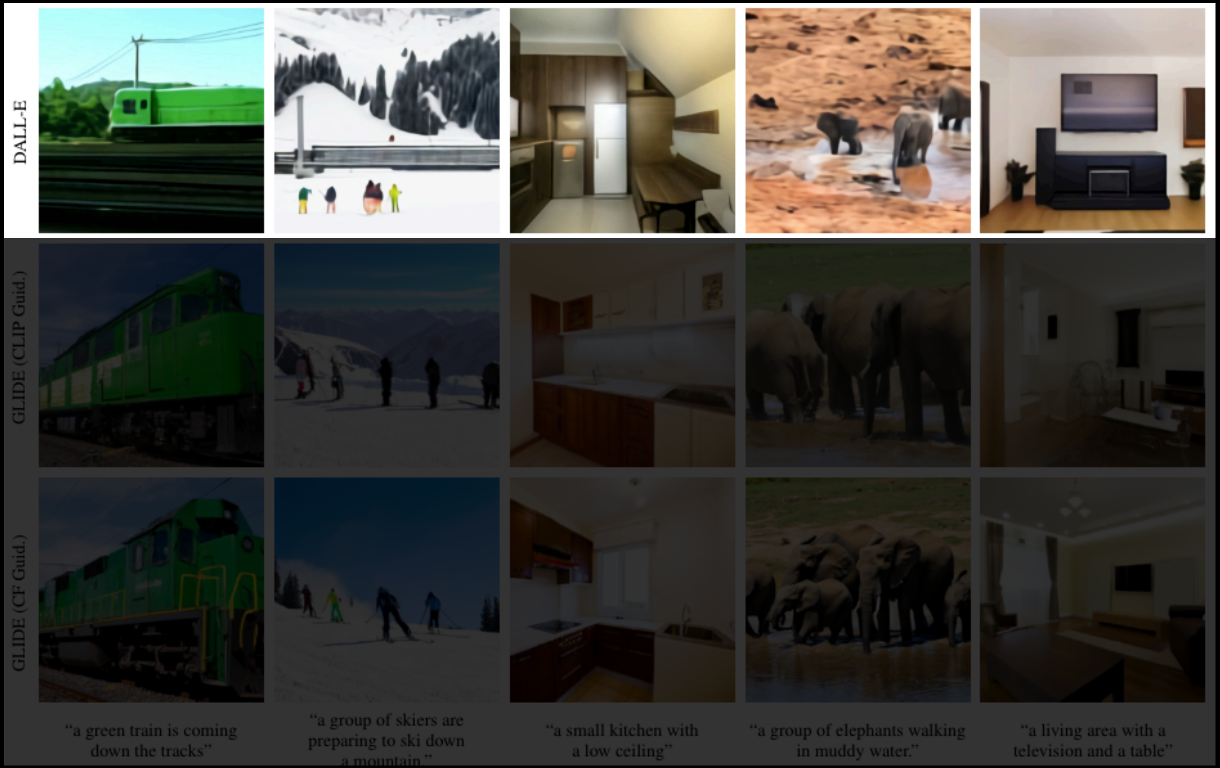
vs.

$$\hat{\mu}_\theta(x_t|y) = \mu_\theta(x_t|y) + s \cdot \Sigma_\theta(x_t|y) \nabla_{x_t} \log p_\phi(y|x_t)$$

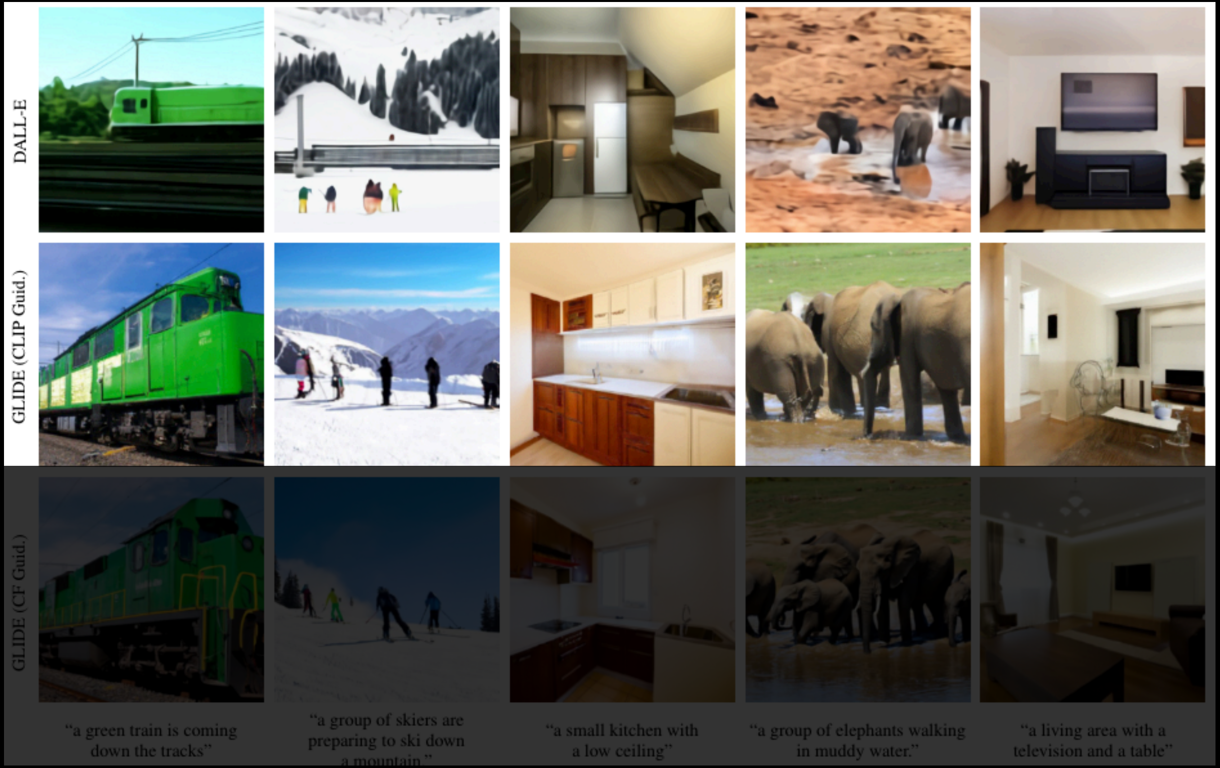
GLIDE (Model)

- Using the ADM model architecture from Guided Diffusion
- Using the same dataset as DALL-E
- Two-stage training
    - For the text encoding, a 1.2B parameter diffusion model is used
    - For upsampling (64×64 → 256×256), a 1.5B parameter diffusion model is used

Experiments (Generation)



Experiments (Generation)



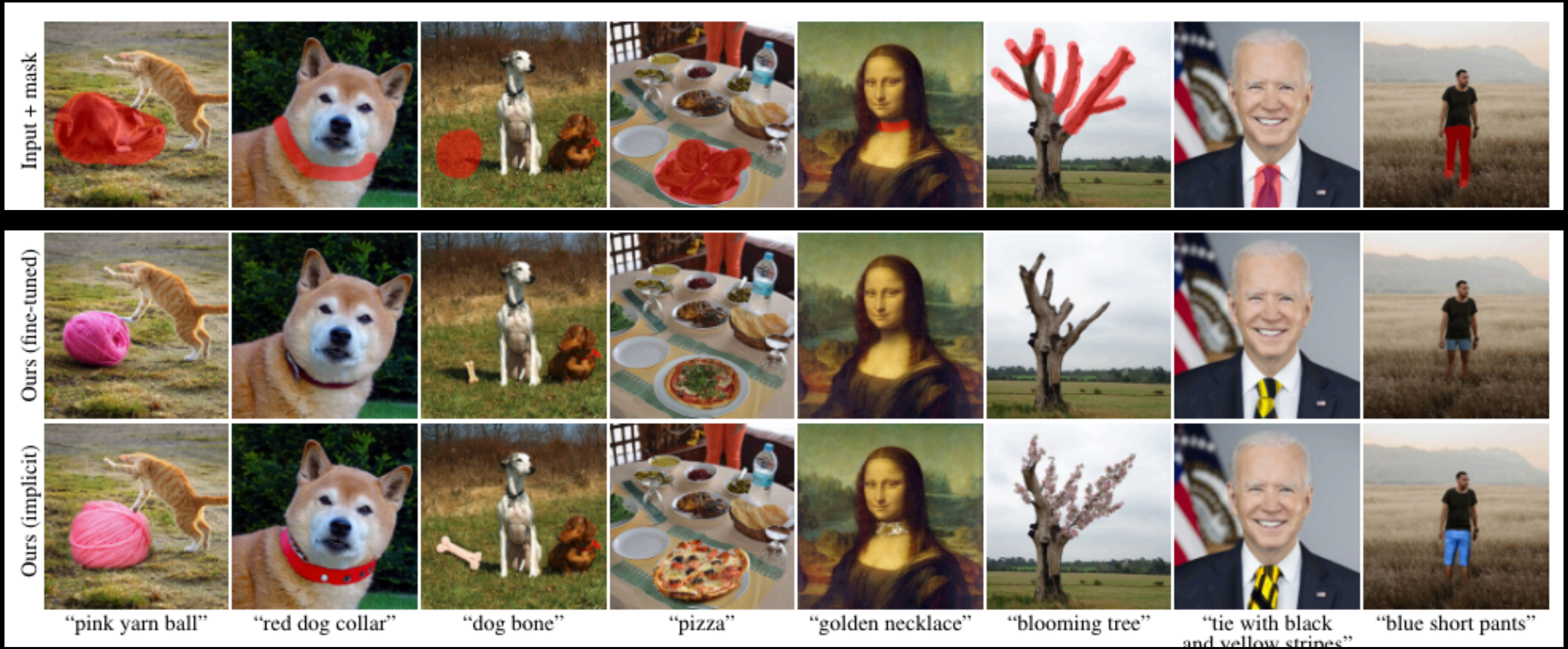
Experiments (Generation)

Experiments (Image Editing)

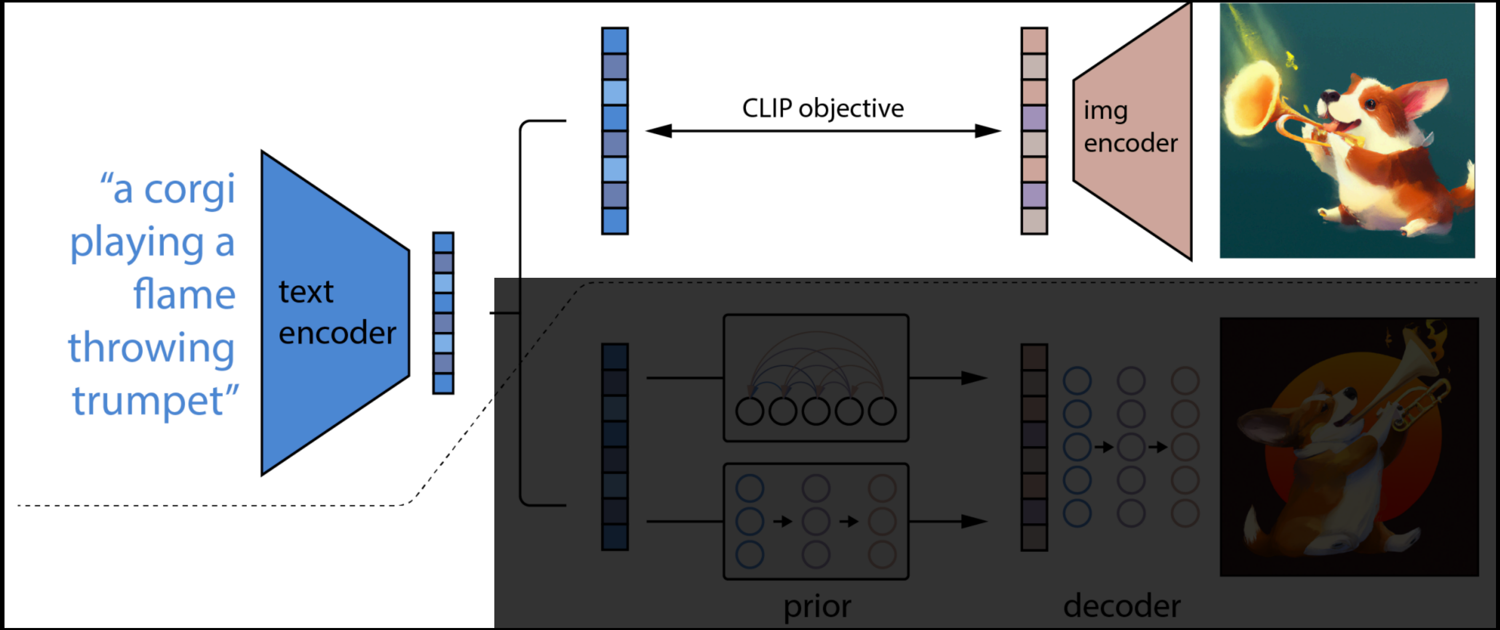
Experiments (Image Editing)

“pink yarn ball” “red dog collar”

Experiments (Image Editing)

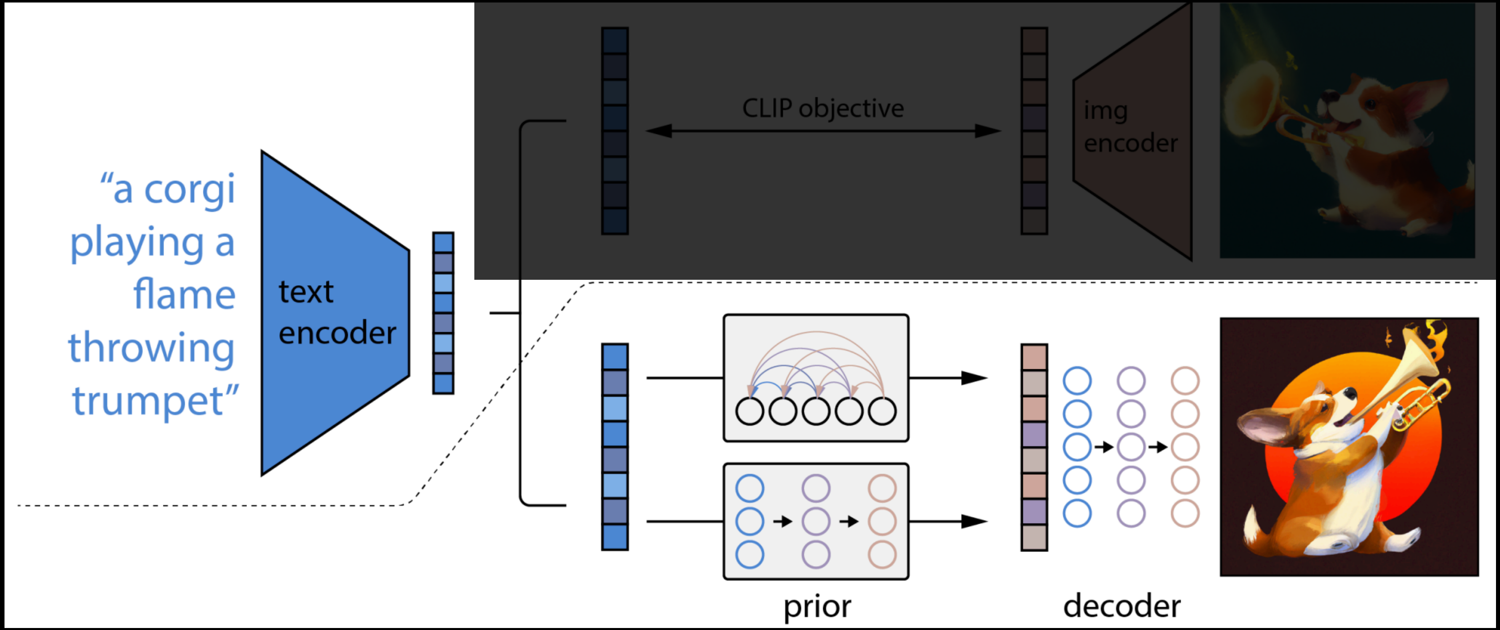


DALLE-2 - Overview

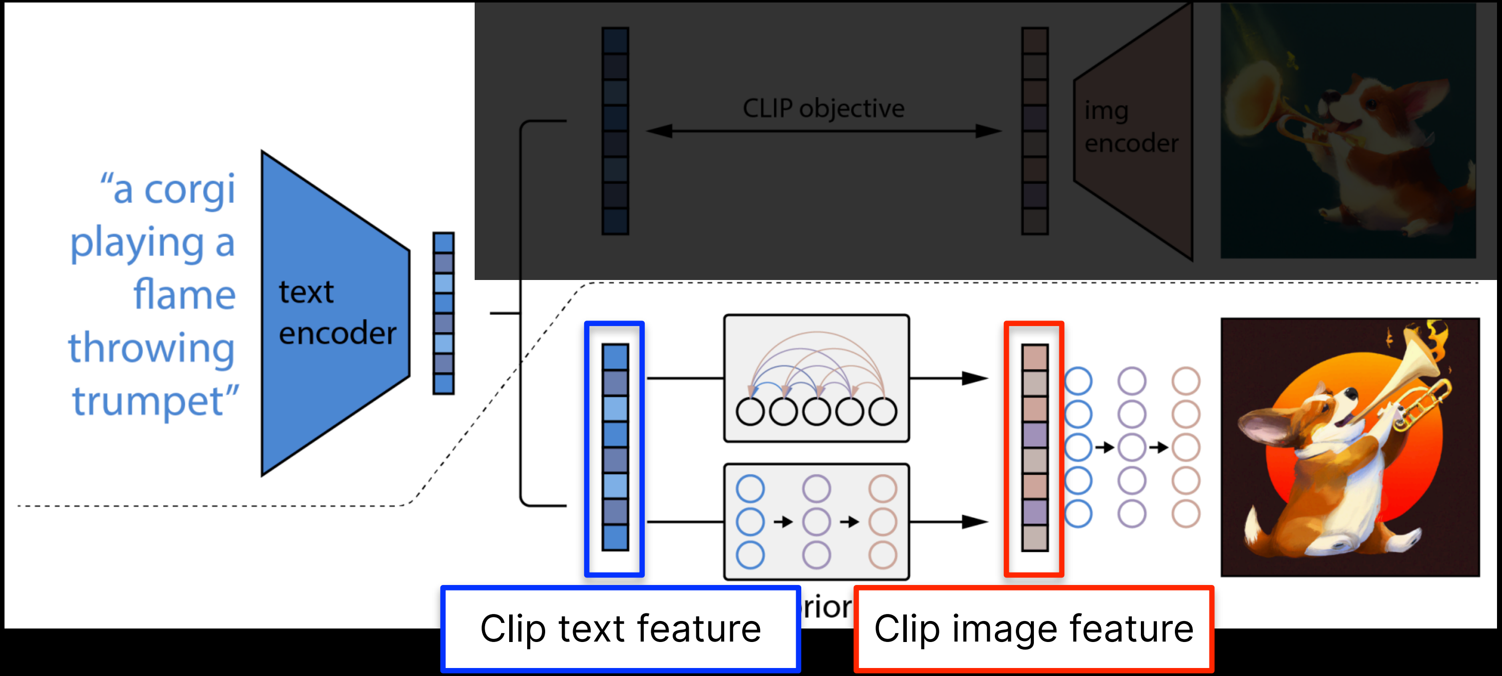


Ramesh et al. "Hierarchical Text-Conditional Image Generation with CLIP Latents", arXiv'22

DALLE-2 - Overview



DALLE-2 - Overview



DALLE-2 - Importance of Prior Model

DALLE-2 - Objective

$$P_\theta(\text{image}|\text{text}) = P_\theta(\text{image}, z|\text{text})$$

DALLE-2 - Objective

$$P_\theta(\text{image}|\text{text}) = P_\theta(\text{image}, z|\text{text})$$

Deterministic variable!

DALLE-2 - Objective

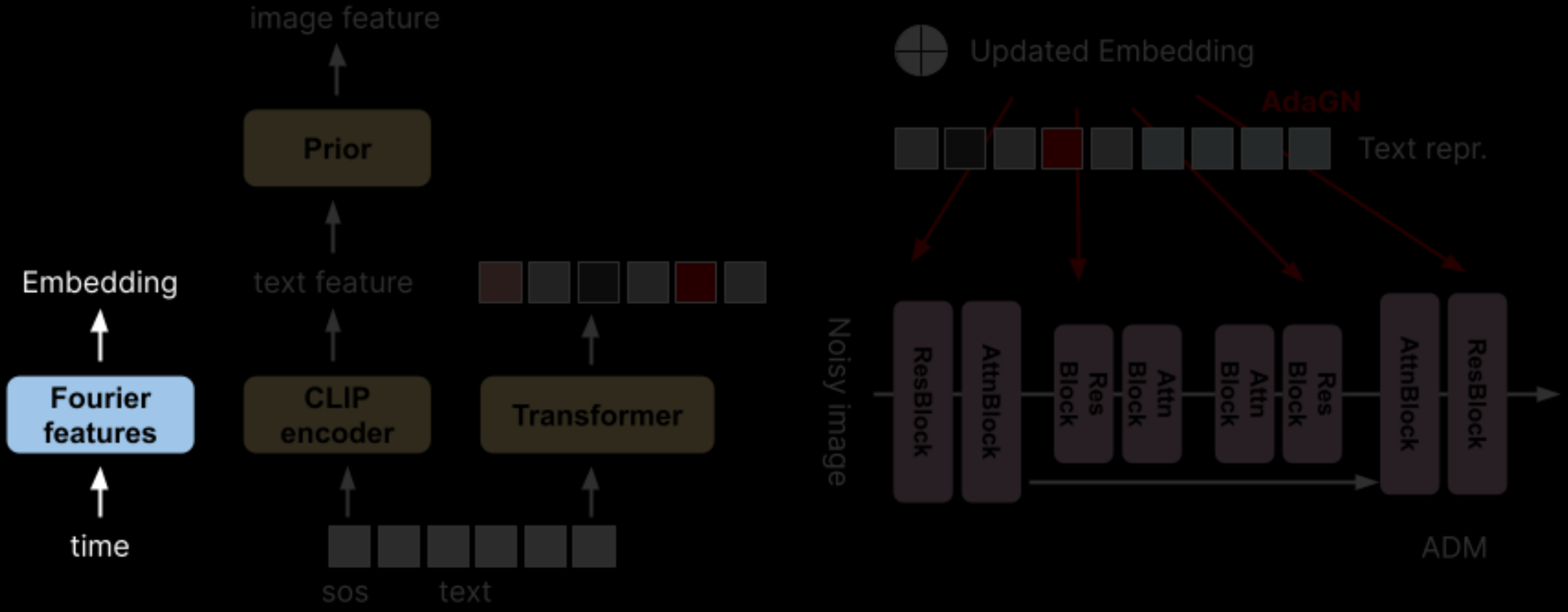
$$P_\theta(\text{image}|\text{text}) = P_\theta(\text{image}, z|\text{text})$$

$$= \underbrace{P_\theta(\text{image}|z, \text{text})}_{\text{Decoder}} \cdot \underbrace{P_\phi(z|\text{text})}_{\text{Prior}}$$

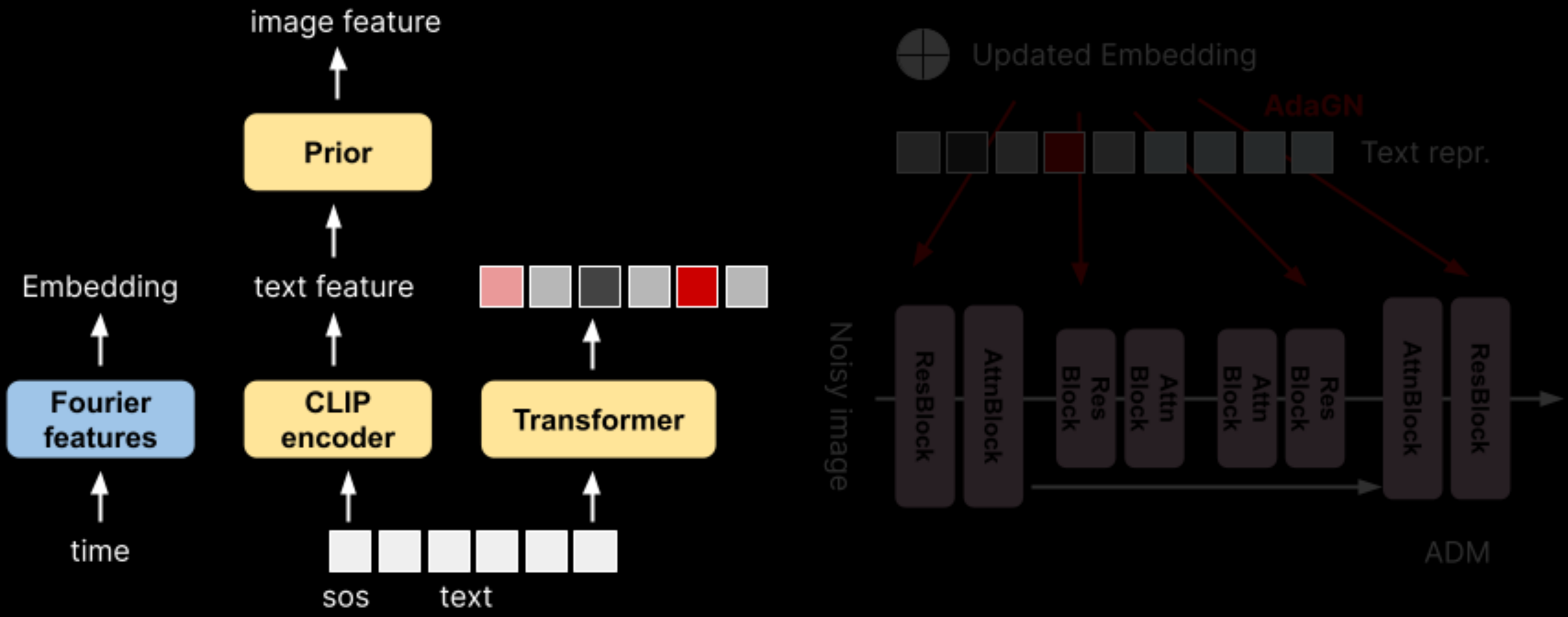
Decoder

Prior

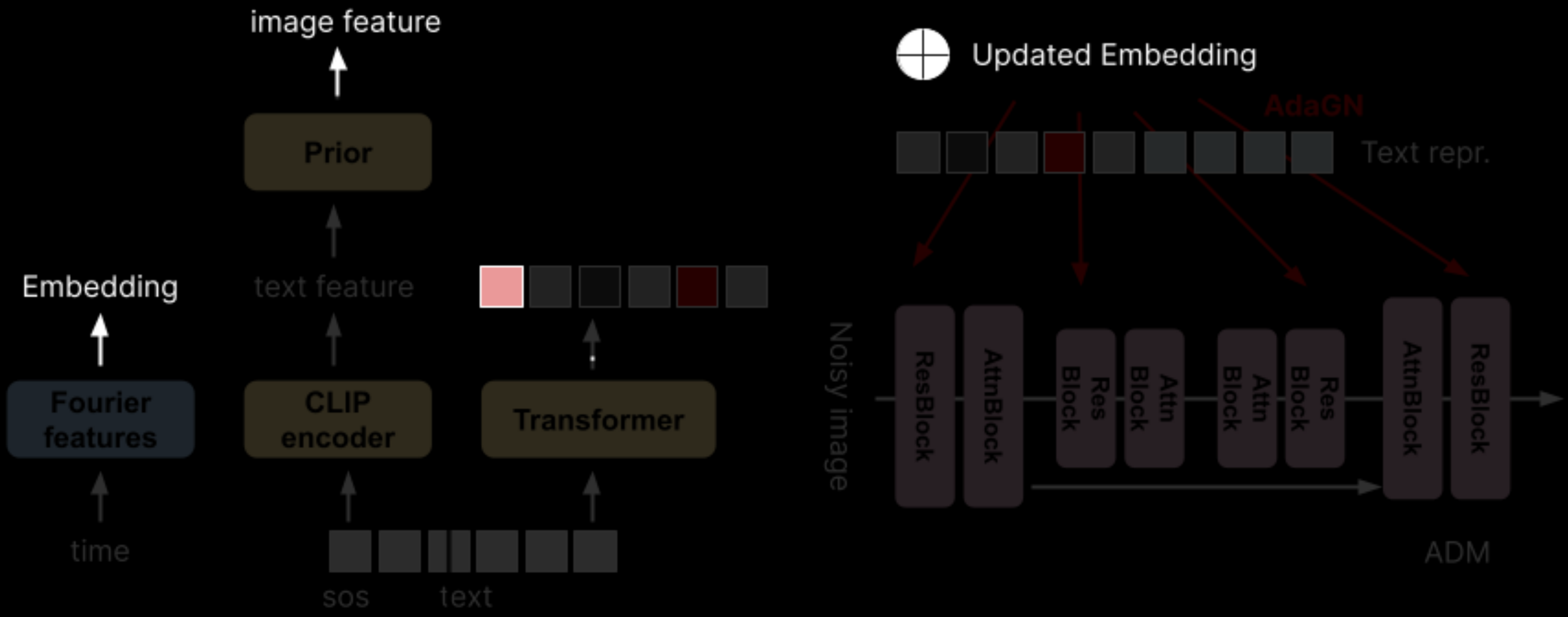
DALLE-2 Architecture - Details



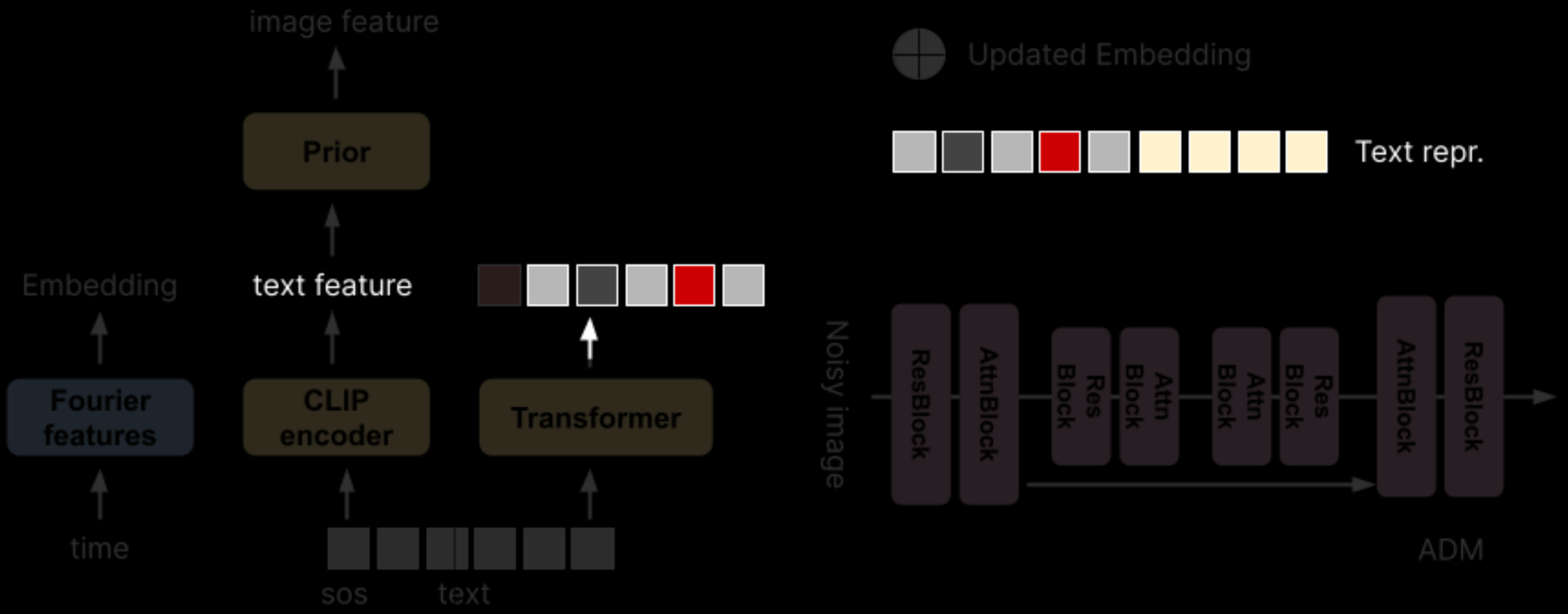
DALLE-2 Architecture - Details



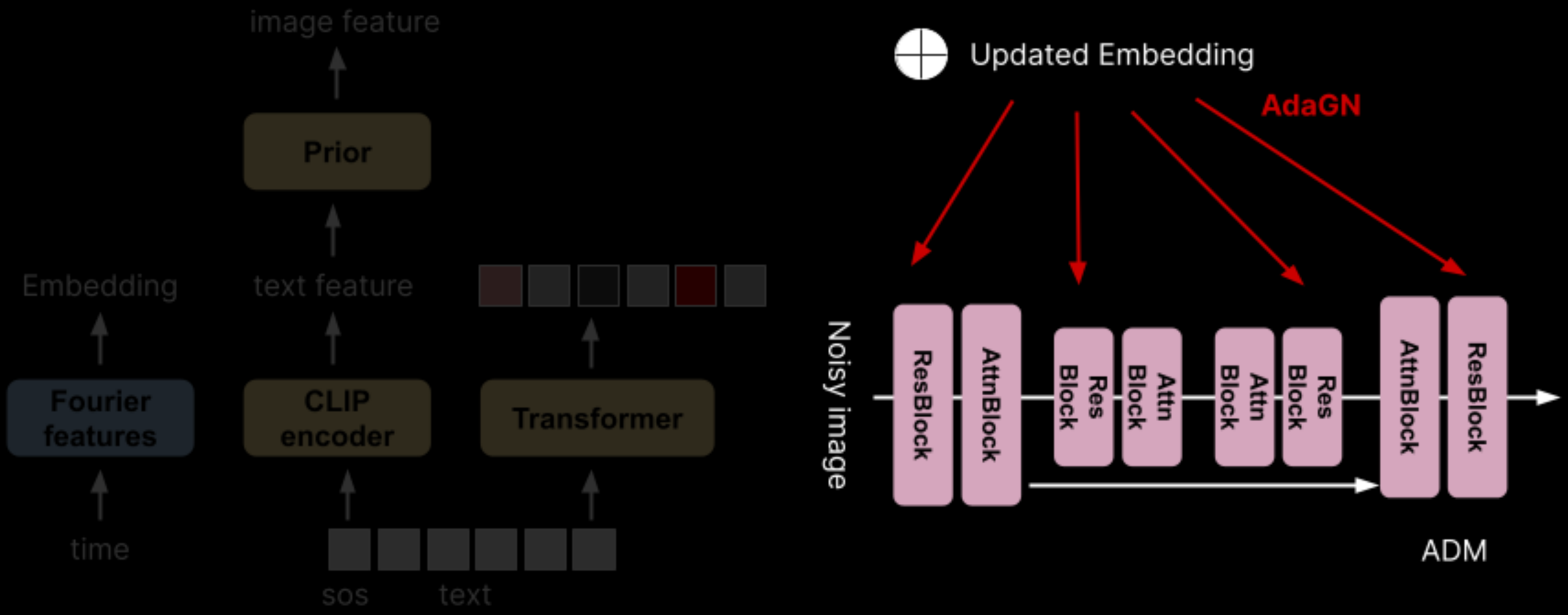
DALLE-2 Architecture - Details



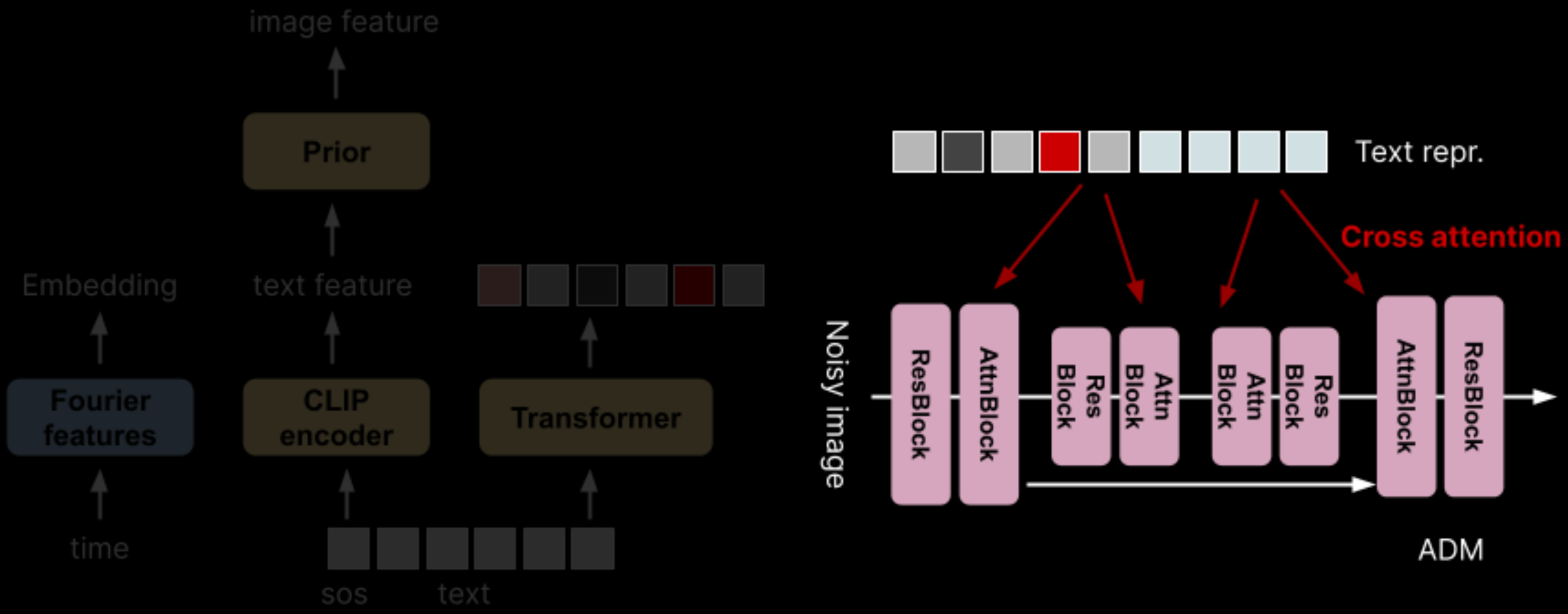
DALLE-2 Architecture - Details



DALLE-2 Architecture - Details



DALLE-2 Architecture - Details



	Diffusion prior	64	$64 \rightarrow 256$	$256 \rightarrow 1024$
Diffusion steps	1000	1000	1000	1000
Noise schedule	cosine	cosine	cosine	linear
Sampling steps	64	250	27	15
Sampling variance method	analytic [2]	learned [34]	DDIM [47]	DDIM [47]
Crop fraction	-	-	0.25	0.25
Model size	1B	3.5B	700M	300M
Channels	-	512	320	192
Depth	-	3	3	2
Channels multiple	-	1,2,3,4	1,2,3,4	1,1,2,2,4,4
Heads channels	-	64	-	-
Attention resolution	-	32,16,8	-	-
Text encoder context	256	256	-	-
Text encoder width	2048	2048	-	-
Text encoder depth	24	24	-	-
Text encoder heads	32	32	-	-
Latent decoder context	-	-	-	-
Latent decoder width	-	-	-	-
Latent decoder depth	-	-	-	-
Latent decoder heads	-	-	-	-
Dropout	-	0.1	0.1	-
Weight decay	6.0e-2	-	-	-
Batch size	4096	2048	1024	512
Iterations	600K	800K	1M	1M
Learning rate	1.1e-4	1.2e-4	1.2e-4	1.0e-4
Adam β_2	0.96	0.999	0.999	0.999
Adam ϵ	1.0e-6	1.0e-8	1.0e-8	1.0e-8
EMA decay	0.9999	0.9999	0.9999	0.9999

Sample Examples (from reddit/Dall-e-2)

Sample)

Posted by u/cench 15 days ago 2

808

An orange cat staring at a drawer filled with socks on fire, high-resolution photo

113 Comments Award Share Save ...

Sampl... 2)


Posted by u/Wiskkey 14 days ago
732
“a painting by Grant Wood of an astronaut couple, american gothic style”
30 Comments
Award
Share
Save

Sample

630

Posted by u/danielbln dalle2 user 7 days ago

happy racoons wearing colourful turtlenecks

32 Comments Award Share Save ...

DALLE-2 Architecture - Limitation

(a) A high quality photo of a dog playing in a green field next to a lake.

(b) A high quality photo of Times Square.

Conclusion

Diffusion Models (DDPM, GLIDE, DALL-E 2)