Denoising Diffusion Probabilistic Models

Докладчик: **Асланов Алишер**
Рецензент-исследователь: **Поздеев Дмитрий**
Хакер: **Демитраки Елизавета**

НИС МОП 2, 19 октября 2022

Синтез изображений

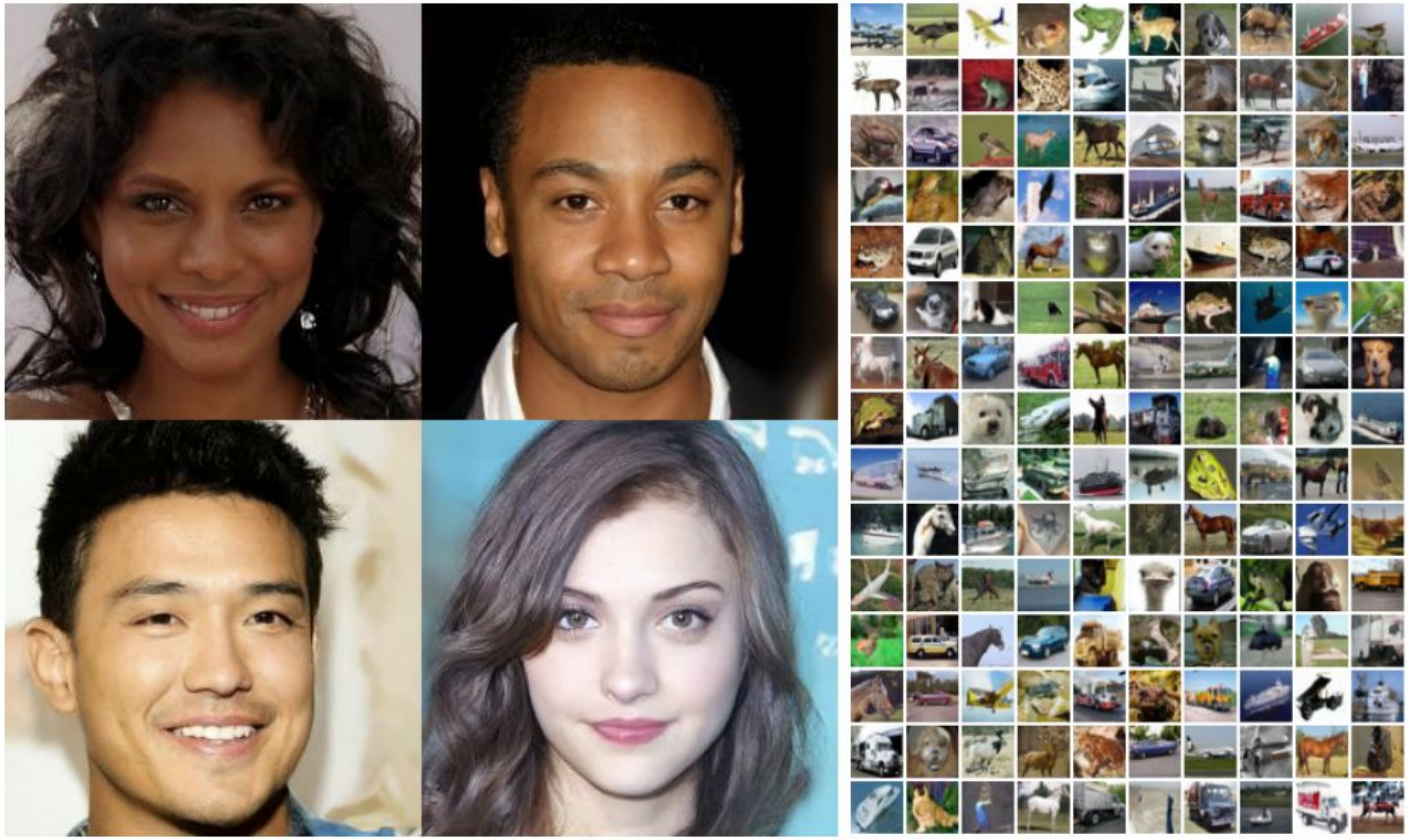
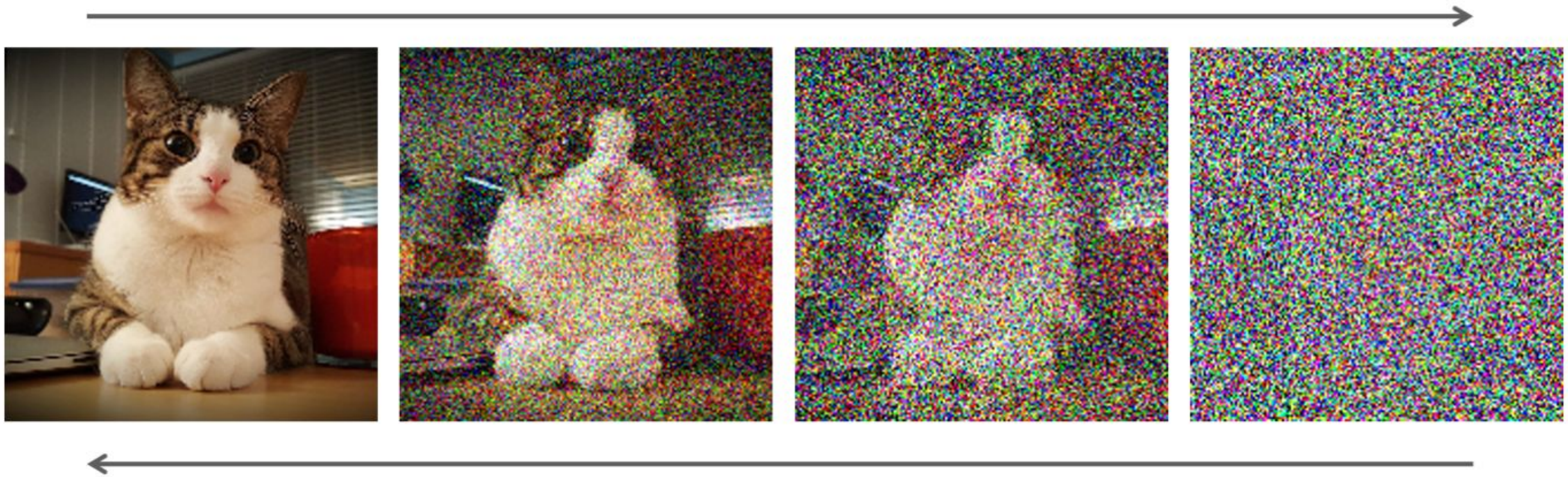


Figure 1: Generated samples on CelebA-HQ 256 × 256 (left) and unconditional CIFAR10 (right)

[https://arxiv.org/pdf/2006.11239.pdf]

Идея DDPM

- Forward diffusion process — зашумление
- Reverse diffusion process — удаление шума
- Обучим модель удалять шум => неявно выучим распределение на исходных данных



[https://developer.nvidia.com/blog/improving-diffusion-models-as-an-alternative-to-gans-part-2/]

Идея DDPM: генерация

Применим расшумляющую модель к **случайному шуму** и получим сгенерированный сэмпл

[https://webbigdata.jp/ai/post-14457]

Формализация идеи

x_t — изображение после **t** итераций добавления шума
x_0 — исходная картинка
T — число итераций (**T = 1000**)

Forward diffusion process:

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

$$\mathbf{x}_0 \sim q(\mathbf{x}_0)$$

Variance schedule:

$$\beta_1, \ldots, \beta_T \in [0, 1]$$

Reverse diffusion process:

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) := \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t))$$

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

[https://arxiv.org/pdf/2006.11239.pdf]

Формализация идеи

$$\alpha_t = 1 - \beta_t \qquad \bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$$

$$\begin{aligned}
\mathbf{x}_t &= \sqrt{\alpha_t}\mathbf{x}_{t-1} + \sqrt{1-\alpha_t}\boldsymbol{\epsilon}_{t-1} & \text{;where } \boldsymbol{\epsilon}_{t-1}, \boldsymbol{\epsilon}_{t-2}, \dots \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \\
&= \sqrt{\alpha_t \alpha_{t-1}}\mathbf{x}_{t-2} + \sqrt{1-\alpha_t \alpha_{t-1}}\bar{\boldsymbol{\epsilon}}_{t-2} & \text{;where } \bar{\boldsymbol{\epsilon}}_{t-2} \text{ merges two Gaussians (*).} \\
&= \dots \\
&= \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon} \\
q(\mathbf{x}_t|\mathbf{x}_0) &= \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I})
\end{aligned}$$

Прямой процесс сходится к стандартному нормальному распределению

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

Формализация идеи

При заданном **x_0** *точный* обратный процесс может выражен с помощью теоремы Байеса через нормальное распределение:

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_{t-1}; \tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0), \tilde{\beta}_t \mathbf{I}),$$

$$\text{where} \quad \tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) := \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\mathbf{x}_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{x}_t \quad \text{and} \quad \tilde{\beta}_t := \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$$

$q(\mathbf{x}_{t-1}|\mathbf{x}_t)$ расписать аналогично не выйдет, т.к. оно зависит от распределения на **x_0**, которого мы не знаем

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

Функционал ошибки

Обучаемся методом максимального правдоподобия. Само по себе правдоподобие на практике не вычислимо => перейдем к вариационной нижней оценке

$$
\begin{aligned}
-\log p_\theta(\mathbf{x}_0) &\leq -\log p_\theta(\mathbf{x}_0) + D_{\mathrm{KL}}(q(\mathbf{x}_{1:T}|\mathbf{x}_0) \| p_\theta(\mathbf{x}_{1:T}|\mathbf{x}_0)) \\
&= -\log p_\theta(\mathbf{x}_0) + \mathbb{E}_{\mathbf{x}_{1:T}\sim q(\mathbf{x}_{1:T}|\mathbf{x}_0)}\Big[\log \frac{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}{p_\theta(\mathbf{x}_{0:T})/p_\theta(\mathbf{x}_0)}\Big] \\
&= -\log p_\theta(\mathbf{x}_0) + \mathbb{E}_q\Big[\log \frac{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}{p_\theta(\mathbf{x}_{0:T})} + \log p_\theta(\mathbf{x}_0)\Big] \\
&= \mathbb{E}_q\Big[\log \frac{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}{p_\theta(\mathbf{x}_{0:T})}\Big]
\end{aligned}
$$

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

Функционал ошибки

Полученную оценку можно привести к следующему виду:

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

$$L_{t-1}=\mathbb{E}_q\left[\frac{1}{2\sigma_t^2}\|\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t,\mathbf{x}_0)-\boldsymbol{\mu}_\theta(\mathbf{x}_t,t)\|^2\right]+C$$

[https://arxiv.org/pdf/2006.11239.pdf]

Функционал ошибки

$$\tilde{\boldsymbol{\mu}}_t = \frac{1}{\sqrt{\alpha_t}}\Big(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_t\Big)$$

$$\begin{aligned}
L_t &= \mathbb{E}_{\mathbf{x}_0,\boldsymbol{\epsilon}}\Big[\frac{1}{2\|\boldsymbol{\Sigma}_\theta(\mathbf{x}_t,t)\|_2^2}\|\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t,\mathbf{x}_0) - \boldsymbol{\mu}_\theta(\mathbf{x}_t,t)\|^2\Big] \\
&= \mathbb{E}_{\mathbf{x}_0,\boldsymbol{\epsilon}}\Big[\frac{1}{2\|\boldsymbol{\Sigma}_\theta\|_2^2}\|\frac{1}{\sqrt{\alpha_t}}\Big(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_t\Big) - \frac{1}{\sqrt{\alpha_t}}\Big(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t,t)\Big)\|^2\Big] \\
&= \mathbb{E}_{\mathbf{x}_0,\boldsymbol{\epsilon}}\Big[\frac{(1-\alpha_t)^2}{2\alpha_t(1-\bar{\alpha}_t)\|\boldsymbol{\Sigma}_\theta\|_2^2}\|\boldsymbol{\epsilon}_t - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t,t)\|^2\Big] \\
&= \mathbb{E}_{\mathbf{x}_0,\boldsymbol{\epsilon}}\Big[\frac{(1-\alpha_t)^2}{2\alpha_t(1-\bar{\alpha}_t)\|\boldsymbol{\Sigma}_\theta\|_2^2}\|\boldsymbol{\epsilon}_t - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_t, t)\|^2\Big]
\end{aligned}$$

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

Обучение и применение

Упрощенный функционал ошибки:

$$L_t^{\text{simple}} = \mathbb{E}_{t\sim[1,T],\mathbf{x}_0,\boldsymbol{\epsilon}_t}\Big[\|\boldsymbol{\epsilon}_t - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\|^2\Big]$$

Algorithm 1 Training

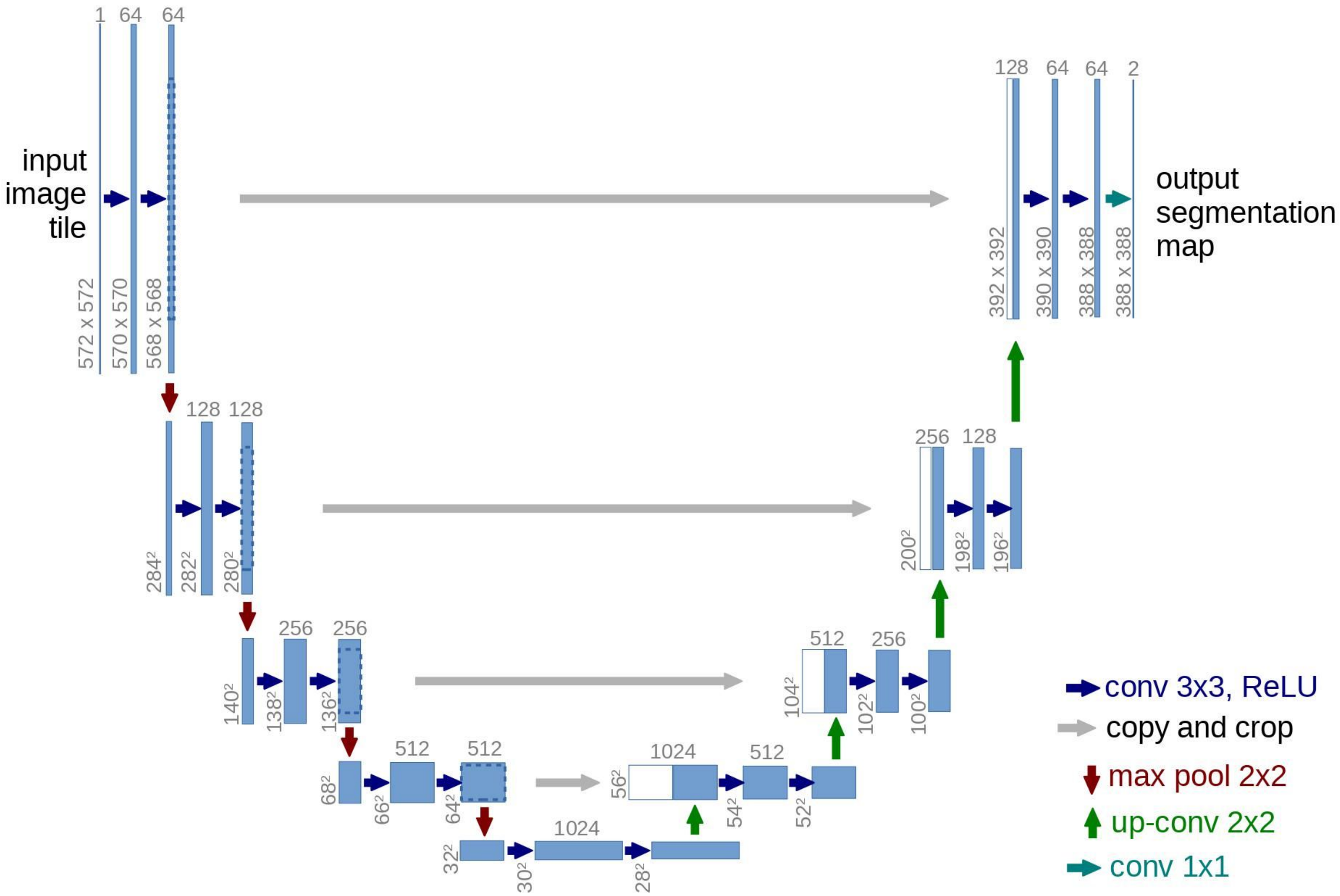
1: **repeat**
2: $\quad \mathbf{x}_0 \sim q(\mathbf{x}_0)$
3: $\quad t \sim \text{Uniform}(\{1,\ldots,T\})$
4: $\quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0},\mathbf{I})$
5: $\quad$ Take gradient descent step on
$$\nabla_\theta \left\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0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

Algorithm 2 Sampling

1: $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0},\mathbf{I})$
2: **for** $t = T,\ldots,1$ **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** $\mathbf{x}_0$

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

Архитектура сети: U-Net



[https://lmb.informatik.uni-freiburg.de/people/ronneber/u-net/]

Интерполяция картинок

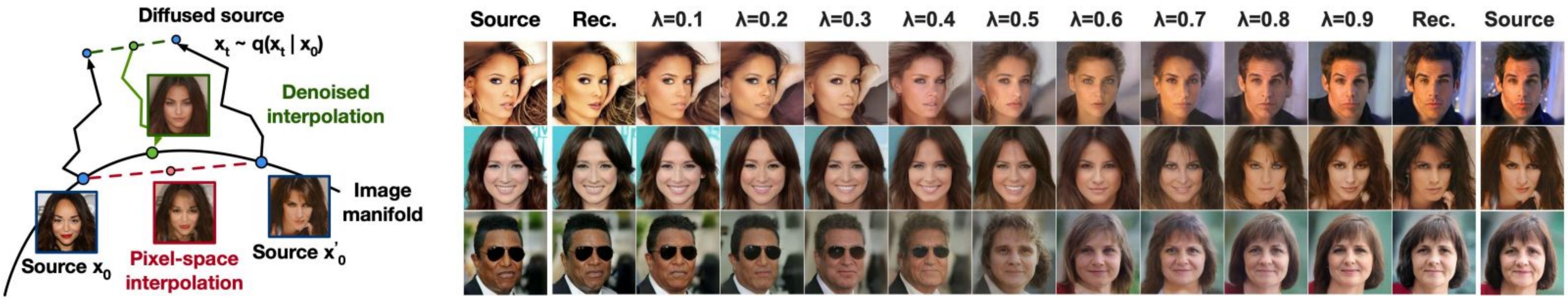


Figure 8: Interpolations of CelebA-HQ 256x256 images with 500 timesteps of diffusion.

[https://arxiv.org/pdf/2006.11239.pdf]

Список источников

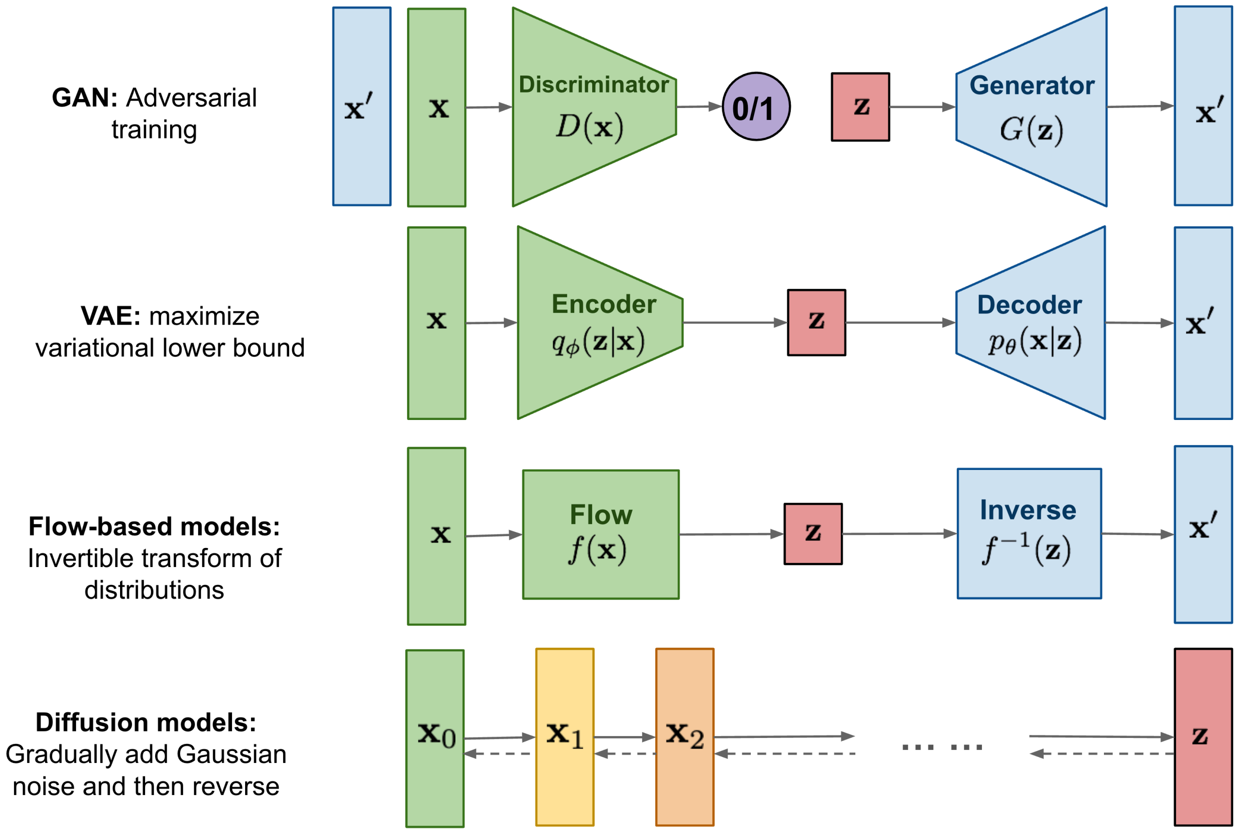
- https://arxiv.org/abs/2006.11239
- https://lilianweng.github.io/posts/2021-07-11-diffusion-models/

- https://www.youtube.com/watch?v=HoKDTa5jHvg
- https://www.youtube.com/watch?v=2Y2Qbsgnfiw
- https://www.youtube.com/watch?v=XTs7M6TSK9I
- https://www.youtube.com/watch?v=y7J6sSO1k50

Denoising Diffusion Probabilistic Models

Prepared by the student of group 192:
Pozdeev Dmitrii

Overview



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

Timeline

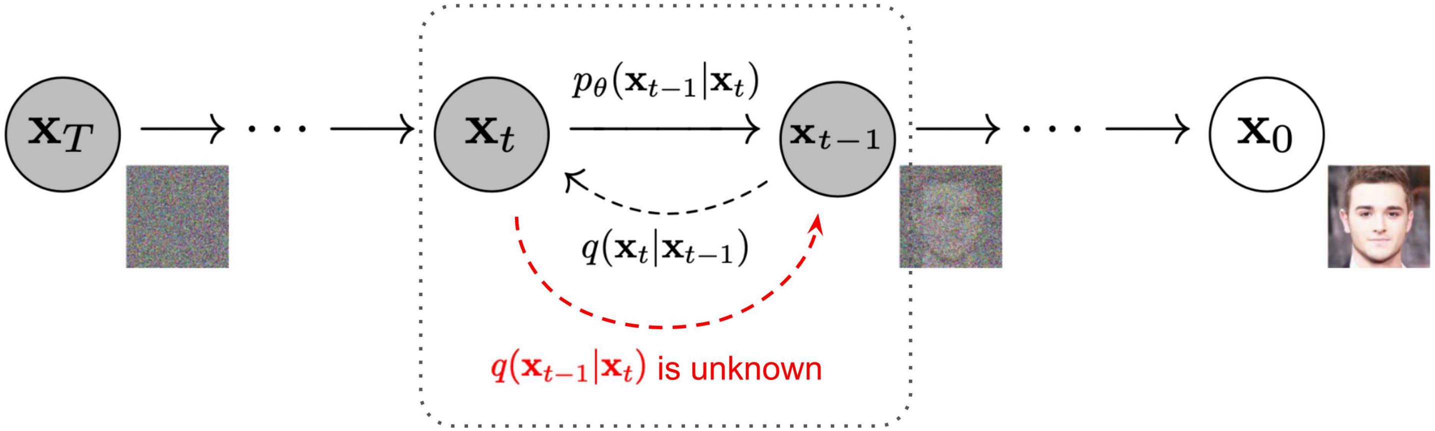
- 2015 Start point
- 2015-2020 Score matching approach

Timeline

- 2015 Start point
- 2015-2020 Score matching approach
- **2020** DDPM
- 2021 Improved DDPM
- 2021 Diffusion Models Beat GANs

DDPM Loss

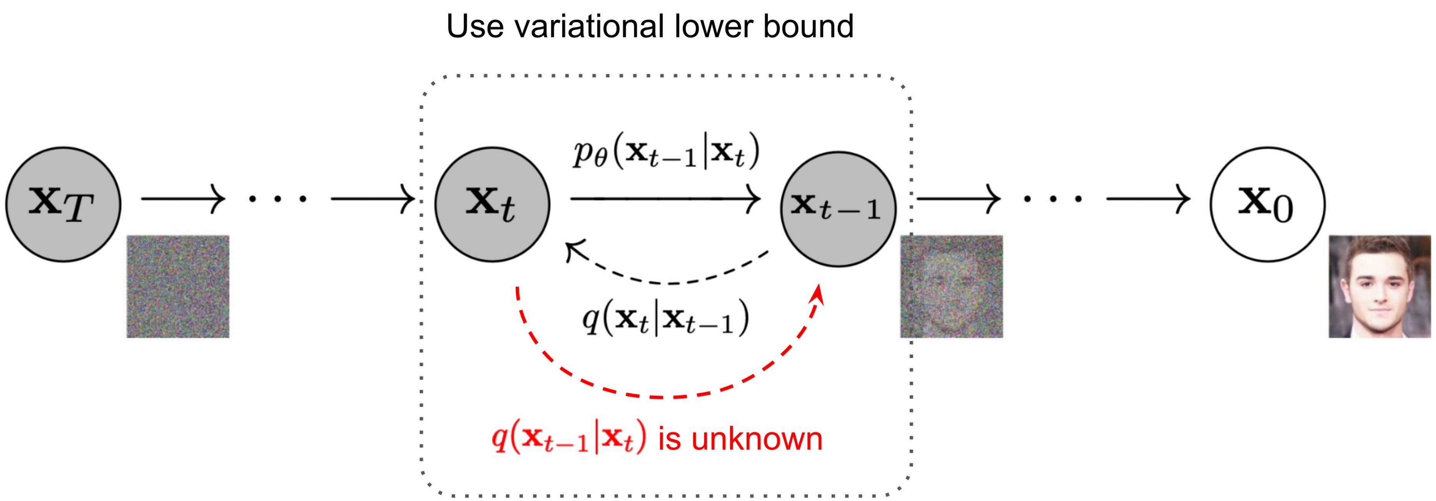
Use variational lower bound



[https://arxiv.org/pdf/2006.11239.pdf]

$$
\begin{aligned}
L_{\text{vlb}} &:= L_0 + L_1 + \ldots + L_{T-1} + L_T \\
L_0 &:= -\log p_\theta(x_0|x_1) \\
L_{t-1} &:= D_{KL}(q(x_{t-1}|x_t, x_0) \;||\; p_\theta(x_{t-1}|x_t)) \\
L_T &:= D_{KL}(q(x_T|x_0) \;||\; p(x_T))
\end{aligned}
$$

DDPM Loss



[https://arxiv.org/pdf/2006.11239.pdf]

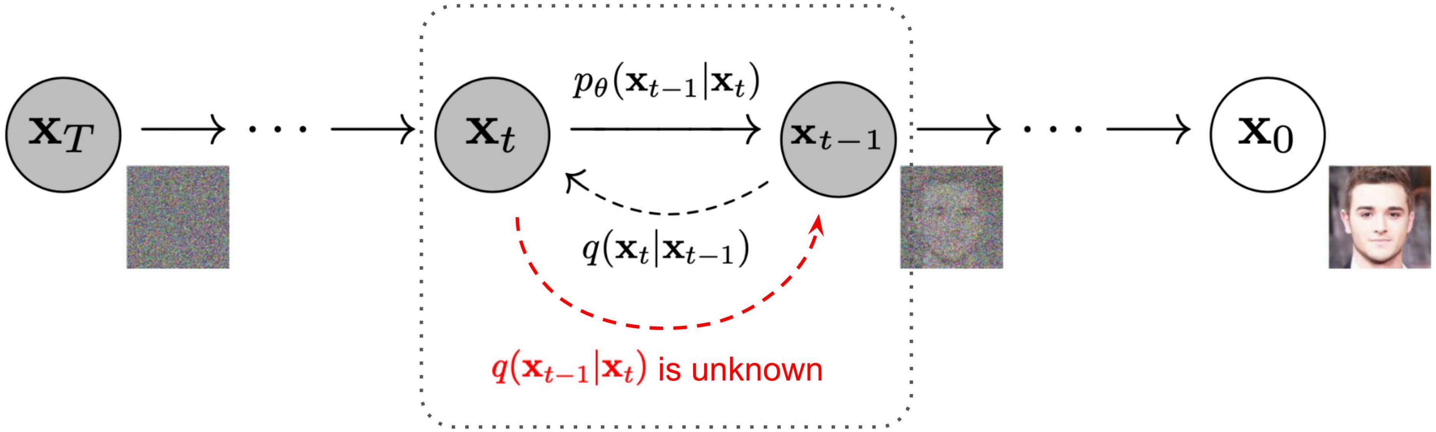
$$\mathbb{E}_{\mathbf{x}_0,\boldsymbol{\epsilon}}\left[\frac{\beta_t^2}{2\sigma_t^2\alpha_t(1-\bar{\alpha}_t)}\left\|\boldsymbol{\epsilon}-\boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0+\sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon},t)\right\|^2\right]$$

$$\alpha_t := 1-\beta_t \text{ and } \bar{\alpha}_t := \prod_{s=1}^t \alpha_s$$

$$L_{\text{simple}} = E_{t,x_0,\epsilon}\left[||\epsilon-\epsilon_\theta(x_t,t)||^2\right]$$

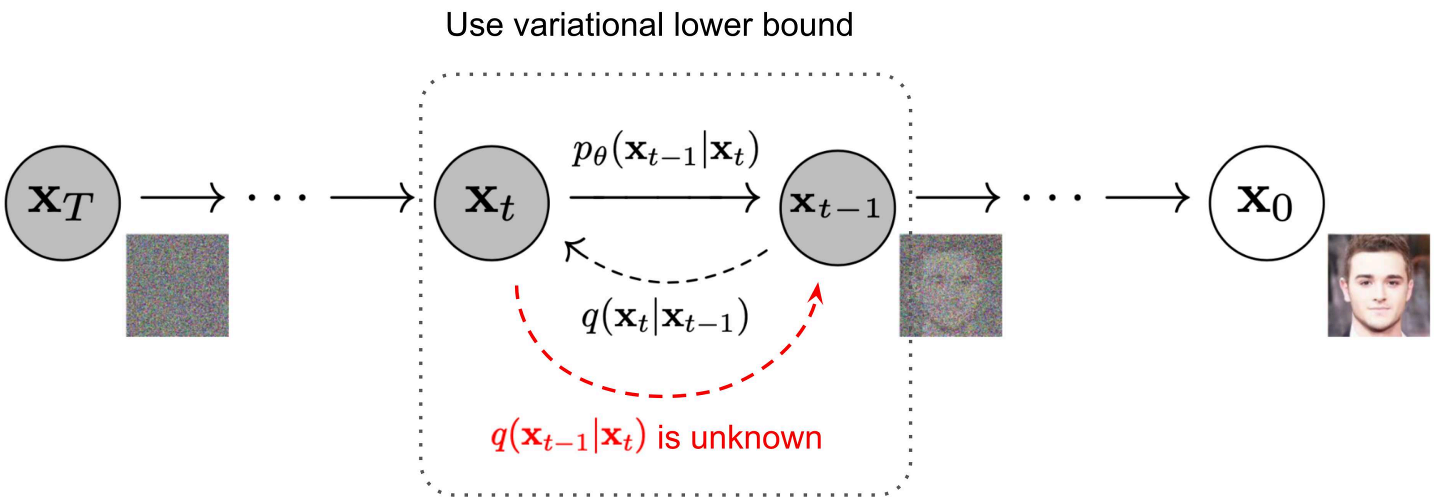
Improved DDPM

Use variational lower bound



[https://arxiv.org/pdf/2006.11239.pdf]

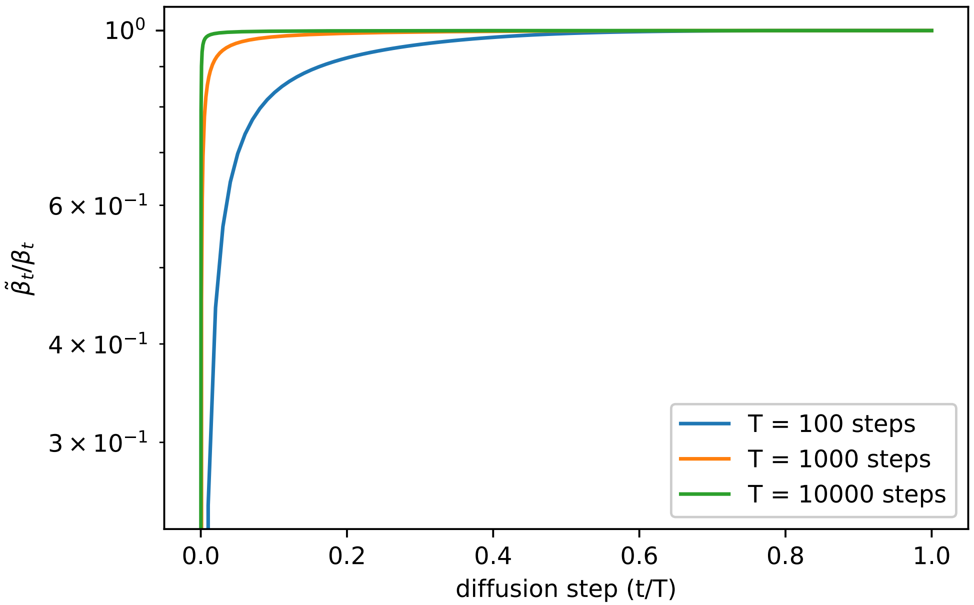
Improved DDPM



[https://arxiv.org/pdf/2006.11239.pdf]

- $\beta_1, \ldots, \beta_T$ decay schedule is linear
- Variances of p_θ are fixed

Learn variances

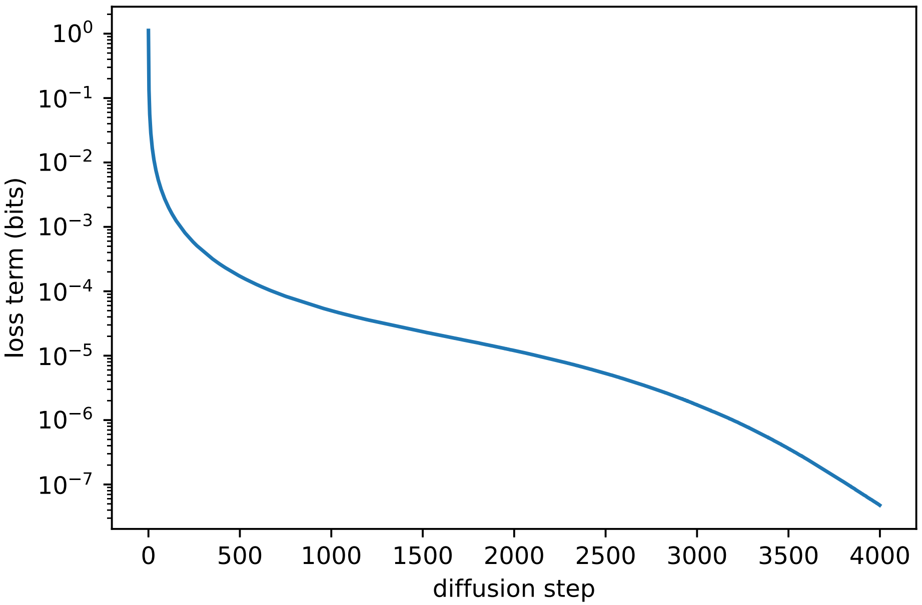


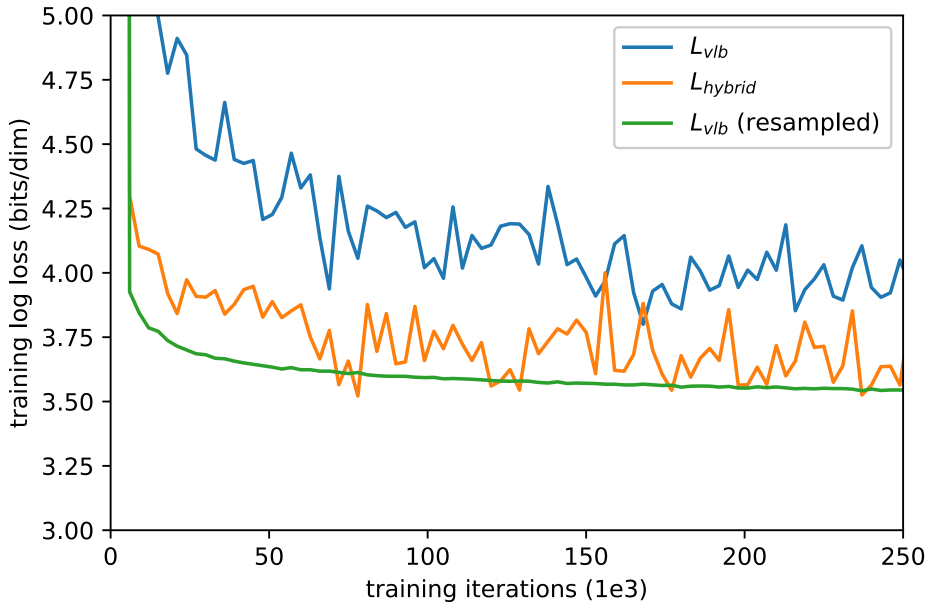
[https://arxiv.org/pdf/2102.09672.pdf]

$$\Sigma_\theta(x_t, t) = \exp(v \log \beta_t + (1 - v) \log \tilde{\beta}_t)$$

$$L_{\text{hybrid}} = L_{\text{simple}} + \lambda L_{\text{vlb}}$$

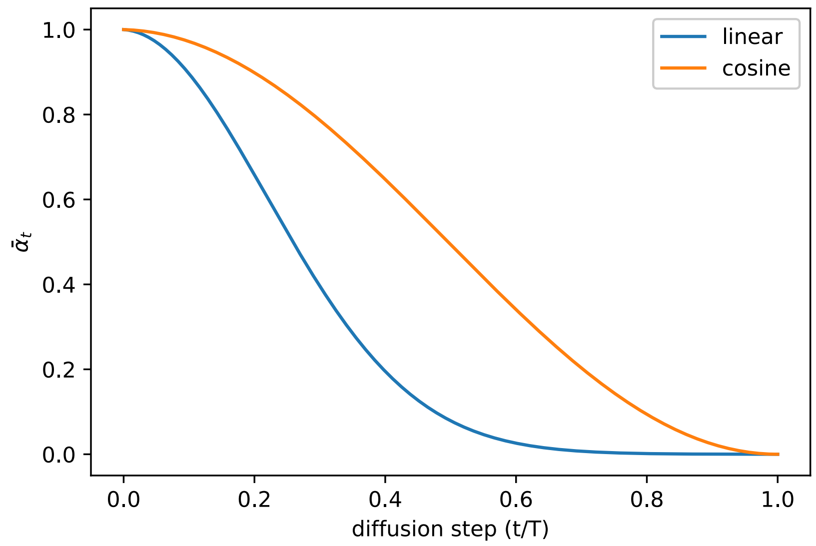
Improved DDPM





[https://arxiv.org/pdf/2102.09672.pdf]

Improved DDPM



[https://arxiv.org/pdf/2102.09672.pdf]

Results

Table 1. Ablating schedule and objective on ImageNet 64×64.

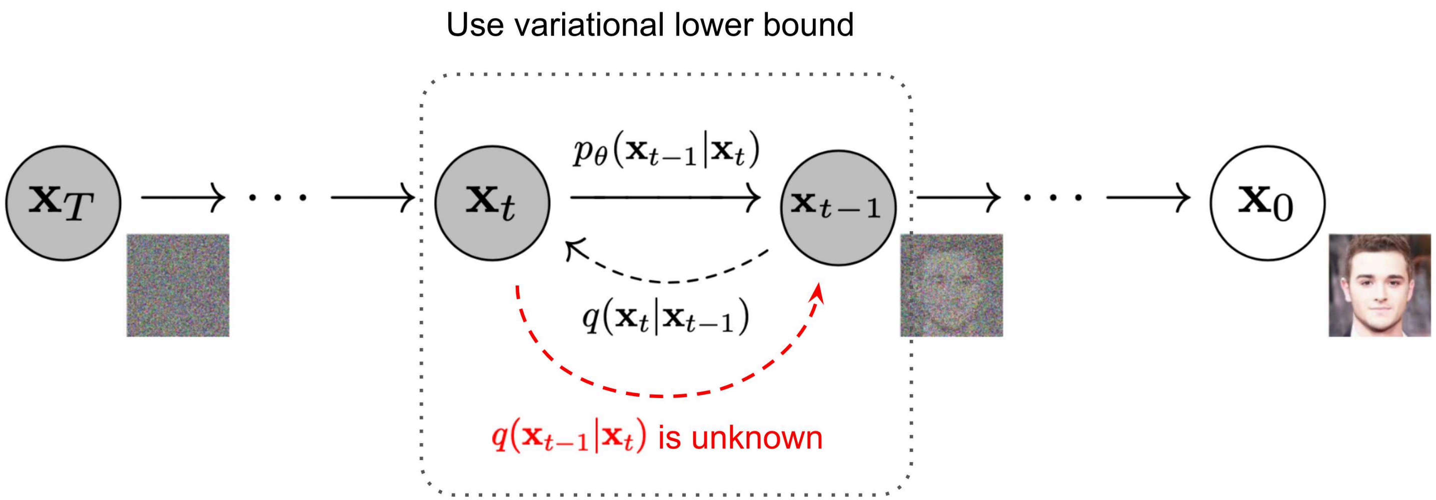
Iters	T	Schedule	Objective	NLL	FID
200K	1K	linear	L_{simple}	3.99	32.5
200K	4K	linear	L_{simple}	3.77	31.3
200K	4K	linear	L_{hybrid}	3.66	32.2
200K	4K	cosine	L_{simple}	3.68	**27.0**
200K	4K	cosine	L_{hybrid}	3.62	28.0
200K	4K	cosine	L_{vlb}	**3.57**	56.7
1.5M	4K	cosine	L_{hybrid}	3.57	**19.2**
1.5M	4K	cosine	L_{vlb}	**3.53**	40.1

Table 2. Ablating schedule and objective on CIFAR-10.

Iters	T	Schedule	Objective	NLL	FID
500K	1K	linear	L_{simple}	3.73	3.29
500K	4K	linear	L_{simple}	3.37	**2.90**
500K	4K	linear	L_{hybrid}	3.26	3.07
500K	4K	cosine	L_{simple}	3.26	3.05
500K	4K	cosine	L_{hybrid}	3.17	3.19
500K	4K	cosine	L_{vlb}	**2.94**	11.47

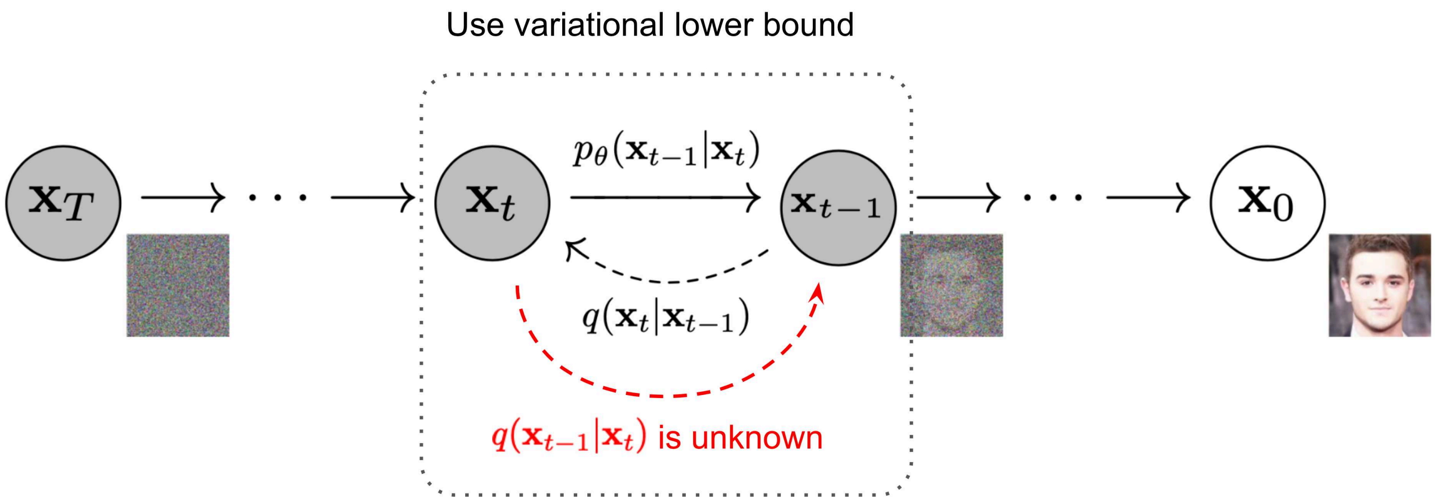
[https://arxiv.org/pdf/2102.09672.pdf]

Speed up diffusion



[https://arxiv.org/pdf/2006.11239.pdf]

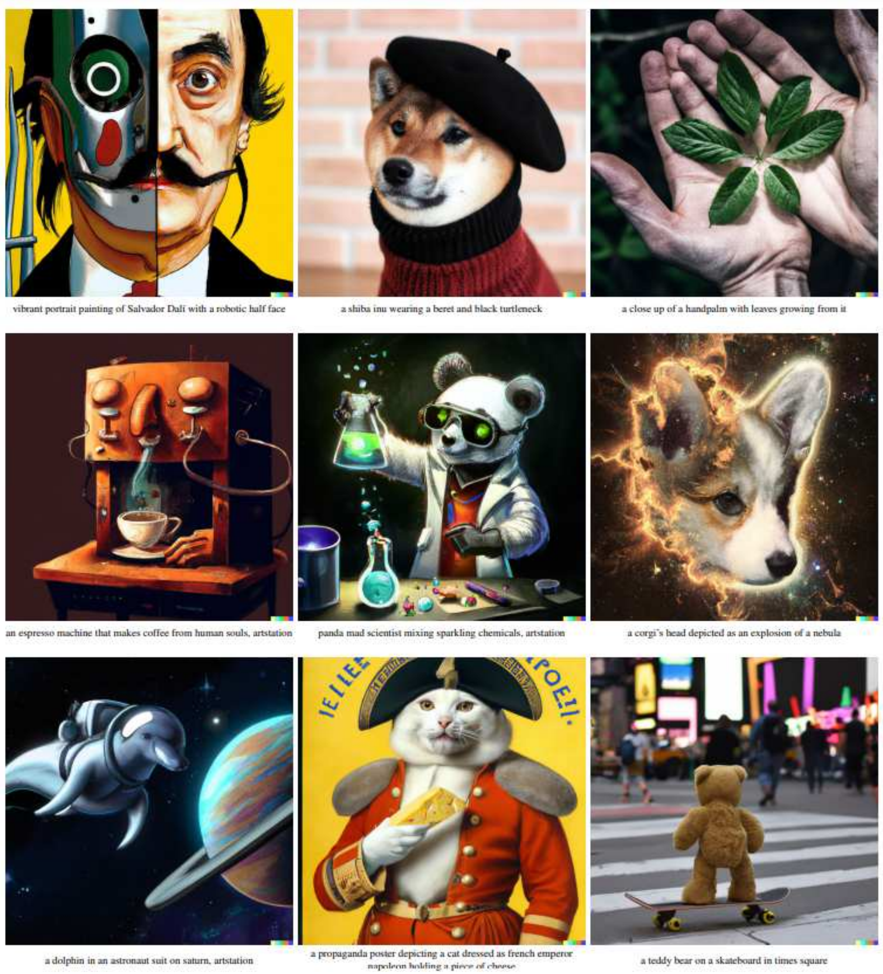
Speed up diffusion



[https://arxiv.org/pdf/2006.11239.pdf]

- Skip some steps
- 10x to 50x faster sampling [3]

Now



[https://cdn.openai.com/papers/dall-e-2.pdf]

Gallery


A small domesticated carnivorous mammal with soft fur, a short snout, and retractable claws


A golden retriever eating ice cream on a beautiful tropical beach at sunset, high resolution


A musk ox grazing on beautiful wildflowers


A blue unicorn flying over a mystical land


Humans building a highway on mars, highly detailed


Sailboat sailing on a sunny day in a mountain lake, highly detailed


A confused grizzly bear in calculus class


A ballerina performs a beautiful and difficult dance on the roof of a very tall skyscraper; the city is lit up and glowing behind her


A knight riding on a horse through the countryside


A panda playing on a swing set

[https://makeavideo.studio/Make-A-Video.pdf]

Applications

- Diffusion repo

Conclusions

- Diffusion models are analytically tractable and flexible

Conclusions

- Diffusion models are analytically tractable and flexible
- But sampling can be expensive

Conclusions

- Diffusion models are analytically tractable and flexible
- But sampling can be expensive
- Be prepared for yet another diffusion paper

References

- [1] main article DDPM
- [2] improved DDPM
- [3] fast sampling
- Awesome blogpost