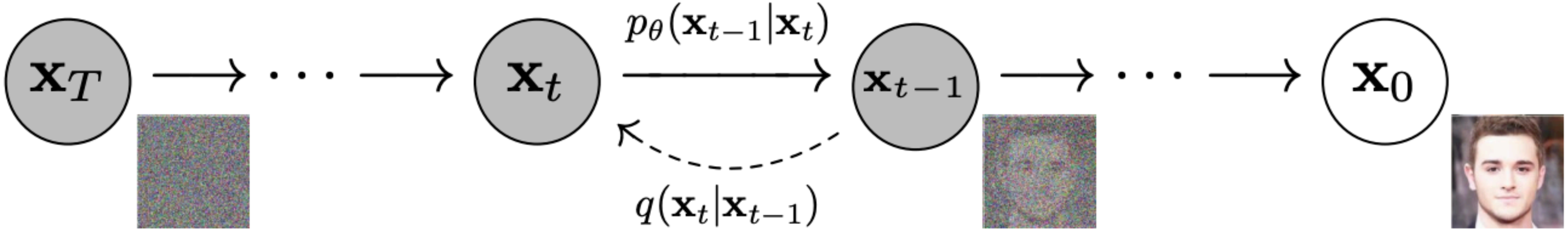
Text diffusion models

ВШЭ ФКН, NLP

Шабалин Александр

Диффузионные модели

- Хотим получить генеративную модель для семплирования из $p(x)$.
- Зададим прямой процесс зашумления и будем учиться его обращать.



Вывод алгоритма

Прямой процесс:

$$q(x_t|x_{t-1}) = \mathcal{N}\left(x_t|\sqrt{\alpha_t}x_{t-1}, (1-\alpha_t)I\right), \quad \alpha_t \in [0,1]$$

Вывод алгоритма

Прямой процесс:

$$q(x_t|x_{t-1}) = \mathcal{N}\left(x_t|\sqrt{\alpha_t}x_{t-1}, (1-\alpha_t)I\right), \quad \alpha_t \in [0,1]$$

Нам пригодится $q(x_t|x_0)$, поэтому выведем ее.

Вывод алгоритма

Прямой процесс:

$$q(x_t|x_{t-1}) = \mathcal{N}\left(x_t|\sqrt{\alpha_t}x_{t-1}, (1-\alpha_t)I\right), \quad \alpha_t \in [0,1]$$

Нам пригодится $q(x_t|x_0)$, поэтому выведем ее.

$$x_t = \sqrt{\alpha_t}x_{t-1} + \sqrt{1-\alpha_t}\varepsilon_{t-1}$$

Вывод алгоритма

Прямой процесс:

$$q(x_t|x_{t-1}) = \mathcal{N}\left(x_t|\sqrt{\alpha_t}x_{t-1}, (1-\alpha_t)I\right), \quad \alpha_t \in [0,1]$$

Нам пригодится $q(x_t|x_0)$, поэтому выведем ее.

$$x_t = \sqrt{\alpha_t}x_{t-1} + \sqrt{1-\alpha_t}\varepsilon_{t-1} =$$

$$= \sqrt{\alpha_t}\left(\sqrt{\alpha_{t-1}}x_{t-2} + \sqrt{1-\alpha_{t-1}}\varepsilon_{t-2}\right) + \sqrt{1-\alpha_t}\varepsilon_{t-1}$$

Вывод алгоритма

Прямой процесс:

$$q(x_t|x_{t-1}) = \mathcal{N}\left(x_t|\sqrt{\alpha_t}x_{t-1}, (1-\alpha_t)I\right), \quad \alpha_t \in [0,1]$$

Нам пригодится $q(x_t|x_0)$, поэтому выведем ее.

$$\begin{aligned} x_t &= \sqrt{\alpha_t}x_{t-1} + \sqrt{1-\alpha_t}\varepsilon_{t-1} = \\ &= \sqrt{\alpha_t}\left(\sqrt{\alpha_{t-1}}x_{t-2} + \sqrt{1-\alpha_{t-1}}\varepsilon_{t-2}\right) + \sqrt{1-\alpha_t}\varepsilon_{t-1} = \\ &= \sqrt{\alpha_t\alpha_{t-1}}x_{t-2} + (\sqrt{\alpha_t(1-\alpha_{t-1})}\varepsilon_{t-2} + \sqrt{1-\alpha_t}\varepsilon_{t-1}) \end{aligned}$$

Вывод алгоритма

Прямой процесс:

$$q(x_t|x_{t-1}) = \mathcal{N}\left(x_t|\sqrt{\alpha_t}x_{t-1}, (1-\alpha_t)I\right), \quad \alpha_t \in [0,1]$$

Нам пригодится $q(x_t|x_0)$, поэтому выведем ее.

$$\begin{aligned}
x_t &= \sqrt{\alpha_t}x_{t-1} + \sqrt{1-\alpha_t}\varepsilon_{t-1} = \\
&= \sqrt{\alpha_t}\left(\sqrt{\alpha_{t-1}}x_{t-2} + \sqrt{1-\alpha_{t-1}}\varepsilon_{t-2}\right) + \sqrt{1-\alpha_t}\varepsilon_{t-1} = \\
&= \sqrt{\alpha_t\alpha_{t-1}}x_{t-2} + (\underbrace{\sqrt{\alpha_t(1-\alpha_{t-1})}\varepsilon_{t-2}}_{\sim \mathcal{N}(0,\alpha_t(1-\alpha_{t-1})I)} + \underbrace{\sqrt{1-\alpha_t}\varepsilon_{t-1}}_{\sim \mathcal{N}(0,(1-\alpha_t)I)})
\end{aligned}$$

Вывод алгоритма

Прямой процесс:

$$q(x_t|x_{t-1}) = \mathcal{N}\left(x_t|\sqrt{\alpha_t}x_{t-1}, (1-\alpha_t)I\right), \quad \alpha_t \in [0,1]$$

Нам пригодится $q(x_t|x_0)$, поэтому выведем ее.

$$\begin{aligned}
x_t &= \sqrt{\alpha_t}x_{t-1} + \sqrt{1-\alpha_t}\varepsilon_{t-1} = \\
&= \sqrt{\alpha_t}\left(\sqrt{\alpha_{t-1}}x_{t-2} + \sqrt{1-\alpha_{t-1}}\varepsilon_{t-2}\right) + \sqrt{1-\alpha_t}\varepsilon_{t-1} = \\
&= \sqrt{\alpha_t\alpha_{t-1}}x_{t-2} + (\underbrace{\sqrt{\alpha_t(1-\alpha_{t-1})}\varepsilon_{t-2}}_{\sim\, \mathcal{N}(0,\alpha_t(1-\alpha_{t-1})I)} + \underbrace{\sqrt{1-\alpha_t}\varepsilon_{t-1}}_{\sim\, \mathcal{N}(0,(1-\alpha_t)I)}) = \\
&= \sqrt{\alpha_t\alpha_{t-1}}x_{t-2} + \sqrt{1-\alpha_t\alpha_{t-1}}\varepsilon
\end{aligned}$$

Вывод алгоритма

Прямой процесс:

$$q(x_t|x_{t-1}) = \mathcal{N}\left(x_t|\sqrt{\alpha_t}x_{t-1}, (1-\alpha_t)I\right), \quad \alpha_t \in [0,1]$$

Нам пригодится $q(x_t|x_0)$, поэтому выведем ее.

$$\begin{aligned}
x_t &= \sqrt{\alpha_t}x_{t-1} + \sqrt{1-\alpha_t}\varepsilon_{t-1} = \\
&= \sqrt{\alpha_t}\left(\sqrt{\alpha_{t-1}}x_{t-2} + \sqrt{1-\alpha_{t-1}}\varepsilon_{t-2}\right) + \sqrt{1-\alpha_t}\varepsilon_{t-1} = \\
&= \sqrt{\alpha_t\alpha_{t-1}}x_{t-2} + (\underbrace{\sqrt{\alpha_t(1-\alpha_{t-1})}\varepsilon_{t-2}}_{\sim\,\mathcal{N}(0,\alpha_t(1-\alpha_{t-1})I)} + \underbrace{\sqrt{1-\alpha_t}\varepsilon_{t-1}}_{\sim\,\mathcal{N}(0,(1-\alpha_t)I)}) = \\
&= \sqrt{\alpha_t\alpha_{t-1}}x_{t-2} + \sqrt{1-\alpha_t\alpha_{t-1}}\varepsilon = \dots = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\varepsilon
\end{aligned}$$

$$\boxed{\bar{\alpha}_t = \prod_{i=1}^{t}\alpha_t}$$

Вывод алгоритма

$$\boxed{\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_t}$$

$$x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t} \varepsilon \;\Rightarrow\; \underline{q(x_t | x_0) = \mathcal{N}(x_t | \sqrt{\bar{\alpha}_t} x_0, (1 - \bar{\alpha}_t) I)}$$

Вывод алгоритма

$$\boxed{\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_t}$$

$$x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1-\bar{\alpha}_t}\varepsilon \;\Rightarrow\; \underline{q(x_t|x_0) = \mathcal{N}(x_t|\sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)I)}$$

Обратный процесс:

$$p(x_{t-1}|x_t) = \mathcal{N}\big(x_t|\mu(x_t,t), \Sigma(x_t,t)\big)$$

Мы не знаем $\mu(x_t, t)$ и $\Sigma(x_t, t)$ так как не знаем распределение $p(x)$.

Вывод алгоритма

$$\boxed{\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_t}$$

$$x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t} \varepsilon \;\Rightarrow\; \underline{q(x_t | x_0) = \mathcal{N}(x_t | \sqrt{\bar{\alpha}_t} x_0, (1 - \bar{\alpha}_t) I)}$$

Обратный процесс:

$$p(x_{t-1} | x_t) = \mathcal{N}\big(x_t | \mu(x_t, t), \Sigma(x_t, t)\big)$$

Мы не знаем $\mu(x_t, t)$ и $\Sigma(x_t, t)$ так как не знаем распределение $p(x)$.

Вместо этого найдем $p(x_{t-1} | x_t, x_0)$

$$p(x_{t-1} | x_t, x_0) = \mathcal{N}(x_t | \tilde{\mu}(x_t, x_0), \tilde{\beta}_t I)$$

Вывод алгоритма

$$p(x_{t-1}|x_t, x_0) = \mathcal{N}(x_t|\tilde{\mu}(x_t, x_0), \tilde{\beta}_t I)$$

Вывод алгоритма

$$p(x_{t-1}|x_t, x_0) = \mathcal{N}(x_t|\tilde{\mu}(x_t, x_0), \tilde{\beta}_t I) = \frac{p(x_t|x_{t-1}, x_0)p(x_{t-1}|x_0)}{p(x_t|x_0)}$$

Вывод алгоритма

$$\boxed{\mathcal{N}(x \mid \mu, \sigma) = \frac{1}{Z} \exp\left(- \frac{(x-\mu)^2}{2\sigma} \right)}$$

$$\mathcal{N}(x_t | \sqrt{\alpha_t} x_{t-1}, (1-\alpha_t)I) \qquad \mathcal{N}(x_{t-1} | \sqrt{\bar{\alpha}_{t-1}} x_0, (1-\bar{\alpha}_{t-1})I)$$

$$\| \qquad\qquad \|$$

$$p(x_{t-1}|x_t, x_0) = \mathcal{N}(x_t|\tilde{\mu}(x_t, x_0), \tilde{\beta}_t I) = \frac{p(x_t|x_{t-1}, x_0)p(x_{t-1}|x_0)}{p(x_t|x_0)}$$

$$\|$$

$$\mathcal{N}(x_t|\sqrt{\bar{\alpha}_t} x_0, (1-\bar{\alpha}_t)I)$$

Вывод алгоритма

$$\boxed{\mathcal{N}(x \mid \mu, \sigma) = \frac{1}{Z}\exp\Big(-\frac{(x-\mu)^2}{2\sigma}\Big)}$$

$$\mathcal{N}(x_t|\sqrt{\alpha_t}x_{t-1}, (1-\alpha_t)I) \qquad \mathcal{N}(x_{t-1}|\sqrt{\bar{\alpha}_{t-1}}x_0, (1-\bar{\alpha}_{t-1})I)$$

$$\|\qquad\qquad\qquad\qquad //$$

$$p(x_{t-1}|x_t, x_0) = \mathcal{N}(x_t|\tilde{\mu}(x_t, x_0), \tilde{\beta}_t I) = \frac{p(x_t|x_{t-1}, x_0)p(x_{t-1}|x_0)}{p(x_t|x_0)} \propto$$

$$\|$$

$$\mathcal{N}(x_t|\sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)I)$$

$$\propto \exp\Big[-\frac{1}{2}\Big(\frac{(x_t-\sqrt{\alpha_t}x_{t-1})^2}{1-\alpha_t} + \frac{(x_{t-1}-\sqrt{\bar{\alpha}_{t-1}}x_0)^2}{1-\bar{\alpha}_{t-1}} - \frac{(x_t-\sqrt{\bar{\alpha}_t}x_0)^2}{1-\bar{\alpha}_t}\Big)\Big] =$$

Вывод алгоритма

$$\boxed{\mathcal{N}(x \mid \mu, \sigma) = \frac{1}{Z}\exp\Big(-\frac{(x-\mu)^2}{2\sigma}\Big)}$$

$$\mathcal{N}(x_t|\sqrt{\alpha_t}x_{t-1}, (1-\alpha_t)I) \qquad \mathcal{N}(x_{t-1}|\sqrt{\bar\alpha_{t-1}}x_0, (1-\bar\alpha_{t-1})I)$$

$$p(x_{t-1}|x_t, x_0) = \mathcal{N}(x_t|\tilde\mu(x_t, x_0), \tilde\beta_t I) = \frac{p(x_t|x_{t-1}, x_0)p(x_{t-1}|x_0)}{p(x_t|x_0)} \propto$$

$$\mathcal{N}(x_t|\sqrt{\bar\alpha_t}x_0, (1-\bar\alpha_t)I)$$

$$\propto \exp\Big[-\frac{1}{2}\Big(\frac{(x_t-\sqrt{\alpha_t}x_{t-1})^2}{1-\alpha_t} + \frac{(x_{t-1}-\sqrt{\bar\alpha_{t-1}}x_0)^2}{1-\bar\alpha_{t-1}} - \frac{(x_t-\sqrt{\bar\alpha_t}x_0)^2}{1-\bar\alpha_t}\Big)\Big] =$$

$$= \exp\Big[-\frac{1}{2}\Big(\Big(\frac{\alpha_t}{1-\alpha_t} + \frac{1}{1-\bar\alpha_{t-1}}\Big)x_{t-1}^2 - 2\Big(\frac{\sqrt{\alpha_t}}{1-\alpha_t}x_t + \frac{\sqrt{\bar\alpha_{t-1}}}{1-\bar\alpha_{t-1}}x_0\Big)x_{t-1} + C(x_t, x_0)\Big)\Big]$$

Вывод алгоритма

$$\boxed{\mathcal{N}(x \mid \mu, \sigma) = \frac{1}{Z} \exp\left(-\frac{(x-\mu)^2}{2\sigma}\right)}$$

$$\mathcal{N}(x_t|\sqrt{\alpha_t}x_{t-1}, (1-\alpha_t)I) \quad \mathcal{N}(x_{t-1}|\sqrt{\bar{\alpha}_{t-1}}x_0, (1-\bar{\alpha}_{t-1})I)$$

$$p(x_{t-1}|x_t, x_0) = \mathcal{N}(x_t|\tilde{\mu}(x_t, x_0), \tilde{\beta}_t I) = \frac{p(x_t|x_{t-1}, x_0)p(x_{t-1}|x_0)}{p(x_t|x_0)} \propto$$

$$\mathcal{N}(x_t|\sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)I)$$

$$\propto \exp\left[-\frac{1}{2}\left(\frac{(x_t - \sqrt{\alpha_t}x_{t-1})^2}{1-\alpha_t} + \frac{(x_{t-1} - \sqrt{\bar{\alpha}_{t-1}}x_0)^2}{1-\bar{\alpha}_{t-1}} - \frac{(x_t - \sqrt{\bar{\alpha}_t}x_0)^2}{1-\bar{\alpha}_t}\right)\right] =$$

$$= \exp\left[-\frac{1}{2}\left(\left(\frac{\alpha_t}{1-\alpha_t} + \frac{1}{1-\bar{\alpha}_{t-1}}\right)x_{t-1}^2 - 2\left(\frac{\sqrt{\alpha_t}}{1-\alpha_t}x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_{t-1}}x_0\right)x_{t-1} + C(x_t, x_0)\right)\right]$$

$$\tilde{\beta}_t = \frac{1}{\frac{\alpha_t}{1-\alpha_t} + \frac{1}{1-\bar{\alpha}_{t-1}}} \qquad \tilde{\mu}(x_t, x_0) = \left(\frac{\sqrt{\alpha_t}}{1-\alpha_t}x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_{t-1}}x_0\right)\tilde{\beta}_t$$

Вывод алгоритма

$$\tilde{\beta}_t = \frac{1}{\frac{\alpha_t}{1-\alpha_t} + \frac{1}{1-\bar{\alpha}_{t-1}}}$$

$$\tilde{\mu}(x_t, x_0) = \left(\frac{\sqrt{\alpha_t}}{1-\alpha_t}x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_{t-1}}x_0\right)\tilde{\beta}_t$$

Вывод алгоритма

$$\tilde{\beta}_t = \frac{1}{\frac{\alpha_t}{1-\alpha_t} + \frac{1}{1-\bar{\alpha}_{t-1}}} = \frac{(1-\alpha_t)(1-\bar{\alpha}_{t-1})}{\alpha_t(1-\bar{\alpha}_{t-1}) + 1 - \alpha_t} = \frac{(1-\alpha_t)(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}$$

$$\tilde{\mu}(x_t, x_0) = \left(\frac{\sqrt{\alpha_t}}{1-\alpha_t}x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_{t-1}}x_0\right)\tilde{\beta}_t$$

Вывод алгоритма

$$\tilde{\beta}_t = \frac{1}{\frac{\alpha_t}{1-\alpha_t} + \frac{1}{1-\bar{\alpha}_{t-1}}} = \frac{(1-\alpha_t)(1-\bar{\alpha}_{t-1})}{\alpha_t(1-\bar{\alpha}_{t-1}) + 1 - \alpha_t} = \frac{(1-\alpha_t)(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}$$

$$\tilde{\mu}(x_t, x_0) = \left(\frac{\sqrt{\alpha_t}}{1-\alpha_t} x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_{t-1}} x_0\right)\tilde{\beta}_t =$$

$$= \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t} x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)}{1-\bar{\alpha}_t} x_0 =$$

Вывод алгоритма

$$\tilde{\beta}_t = \frac{1}{\frac{\alpha_t}{1-\alpha_t} + \frac{1}{1-\bar{\alpha}_{t-1}}} = \frac{(1-\alpha_t)(1-\bar{\alpha}_{t-1})}{\alpha_t(1-\bar{\alpha}_{t-1}) + 1 - \alpha_t} = \frac{(1-\alpha_t)(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}$$

$$\begin{aligned}
\tilde{\mu}(x_t, x_0) &= \left(\frac{\sqrt{\alpha_t}}{1-\alpha_t}x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_{t-1}}x_0\right)\tilde{\beta}_t = \\
&= \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)}{1-\bar{\alpha}_t}x_0 = \\
&\qquad \left[x_0 = \frac{x_t - \sqrt{1-\bar{\alpha}_t}\varepsilon_t}{\sqrt{\bar{\alpha}_t}}\right] \\
&= \frac{1}{\bar{\alpha}_t}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_t\right)
\end{aligned}$$

Почти готово

Что теперь?

Почти готово

Что теперь?

У нас есть явная формула для прямого процесса

$$q(x_t|x_0) = \mathcal{N}(x_t|\sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)I)$$

И для обратного

$$p(x_{t-1}|x_t, x_0) = \mathcal{N}(x_t|\tilde{\mu}(x_t, x_0), \tilde{\beta}_t I)$$

Почти готово

Что теперь?

У нас есть явная формула для прямого процесса

$$q(x_t|x_0) = \mathcal{N}(x_t|\sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)I)$$

И для обратного

$$p(x_{t-1}|x_t, x_0) = \mathcal{N}(x_t|\tilde{\mu}(x_t, x_0), \tilde{\beta}_t I)$$

$\tilde{\mu}(x_t, x_0)$ мы не знаем, поэтому обучим нейронную сеть для аппроксимации.

$$L_t = \|\tilde{\mu}(x_t, x_0) - \tilde{\mu}_\theta(x_t, t)\|^2 \quad \text{или} \quad L_t = \|\varepsilon_t - \varepsilon_\theta(x_t, t)\|^2$$

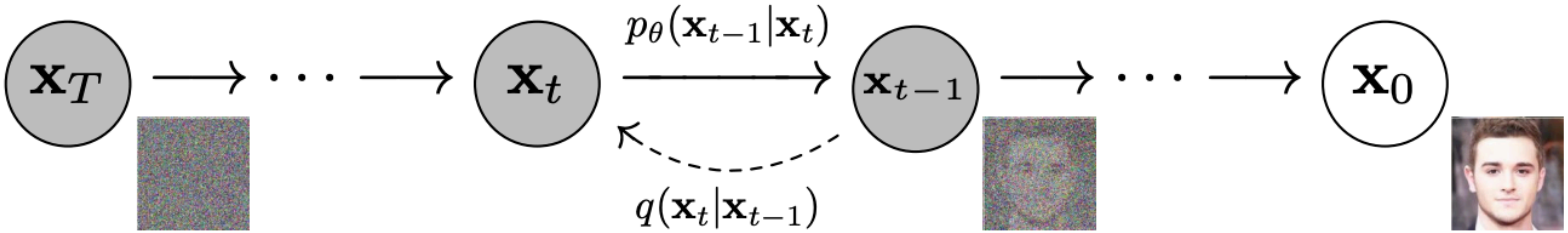
Обучение и инференс

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 $\mathbf{x}_0$



Text diffusion models

Зачем?

- Авторегрессионные модели генерируют каждый токен отдельно. Это дорого.

- Авторегрессионные модели не могут исправлять свои ошибки.

Особенности

- Тексты имеют дискретную структуру => недифференцируемы
- Непонятно, как зашумлять текст
- ~~Текстовая диффузия не работает~~

Два вида текстовых диффузионных моделей

- С дискретным зашумлением
- С гауссовским зашумлением

Дискретный шум

- Можно смотреть на каждый токен, как на семпл из категориального распределение всех возможных токенов.
- В точке x_0 распределение вырожденное.
- Добавление шума – релаксация данного распределения.

Multinomial Diffusion

- Постепенно сводит вырожденное категориальное распределение к равномерному.

$$q(x_t|x_{t-1}) = \mathcal{C}(x_t|\alpha_t x_{t-1} + (1-\alpha_t)/K)$$
$$q(x_t|x_0) = \mathcal{C}(x_t|\bar{\alpha}_t x_{t-1} + (1-\bar{\alpha}_t)/K)$$

$$p(x_{t-1}|x_t, x_0) = \mathcal{C}(x_{t-1}|D_{\text{post}}(x_t, x_0))$$

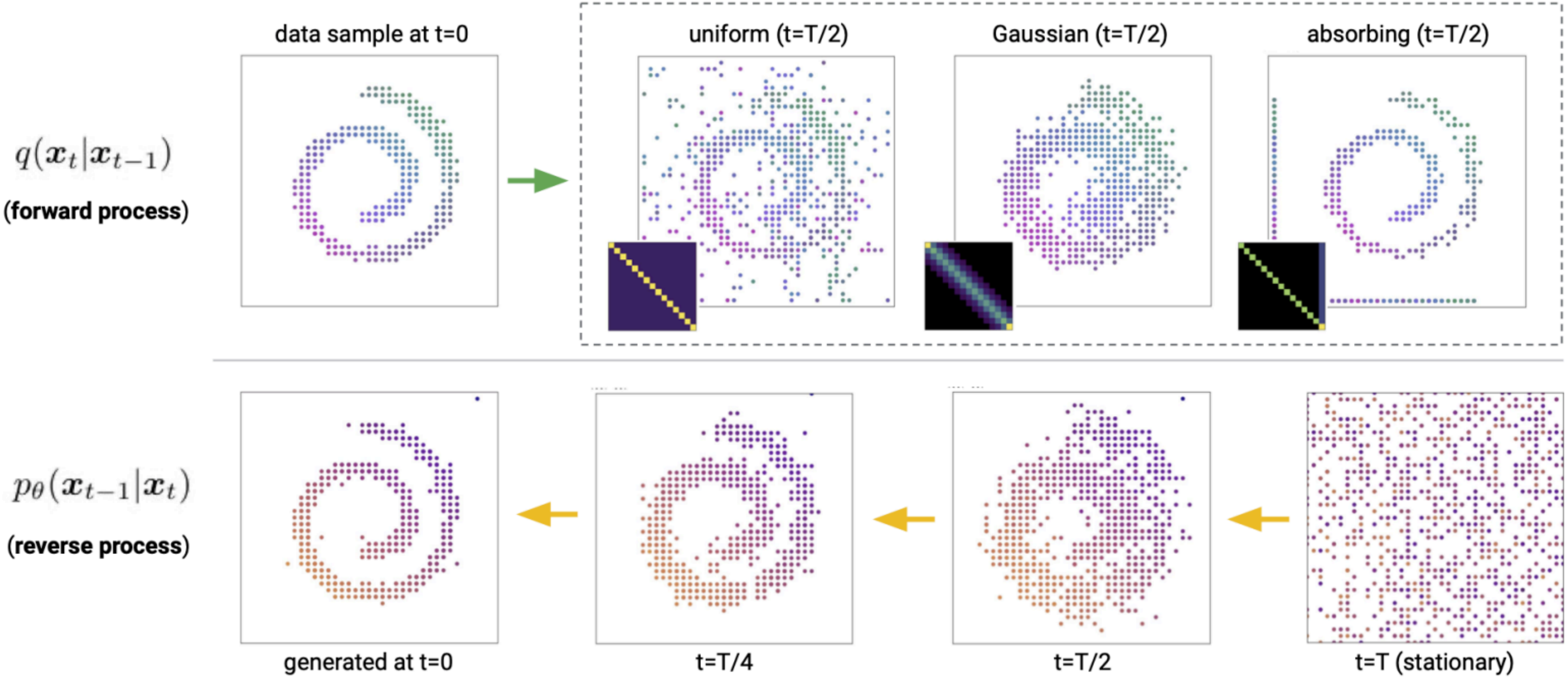
$$D_{\text{post}}(x_t, x_0) = \frac{1}{Z}\left[\alpha_t x_t + (1-\alpha_t)/K\right] \odot \left[\bar{\alpha}_{t-1} x_0 + (1-\bar{\alpha}_{t-1})/K\right]$$

https://arxiv.org/pdf/2102.05379.pdf

D3PM

Discrete Denoising Diffusion Probabilistic Model

- Пробуют разные способы зашумления



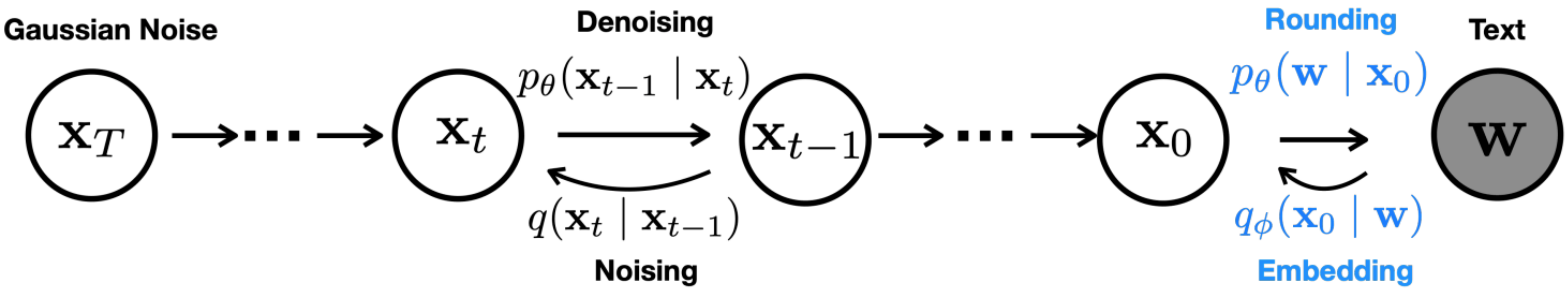
https://arxiv.org/pdf/2107.03006.pdf

Непрерывный шум

- Токены переводятся в пространство эмбеддингов (или энкодингов)
- В этом пространстве учится гауссовская диффузия
- К выходам диффузии применяется декодер
- Работает намного лучше диффузии с дискретным шумом!

Diffusion-LM

- Первая относительно удачная попытка текстовой диффузии.



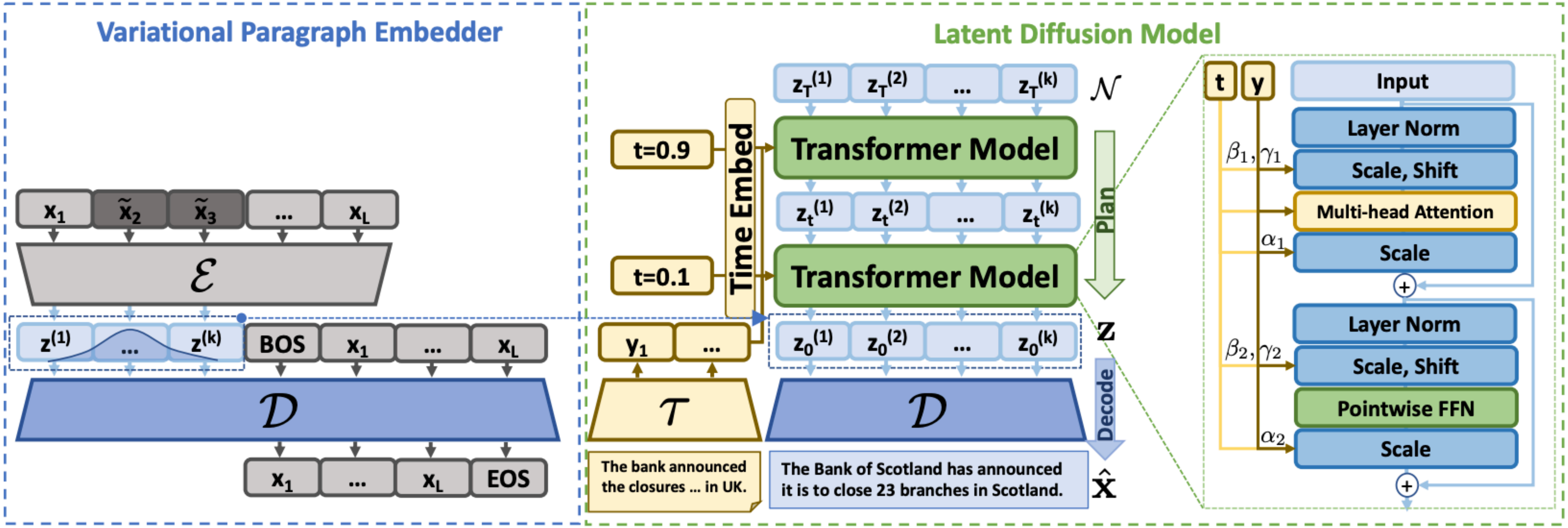
$$\mathcal{L}_{\text{simple}}(\mathbf{x}_0) = \sum_{t=1}^{T} \mathop{\mathbb{E}}_{q(\mathbf{x}_t|\mathbf{x}_0)} ||\mu_\theta(\mathbf{x}_t, t) - \hat{\mu}(\mathbf{x}_t, \mathbf{x}_0)||^2$$

$$\mathcal{L}^{\text{e2e}}_{\text{simple}}(\mathbf{w}) = \mathop{\mathbb{E}}_{q_\phi(\mathbf{x}_{0:T}|\mathbf{w})} \left[\mathcal{L}_{\text{simple}}(\mathbf{x}_0) + ||\text{EMB}(\mathbf{w}) - \mu_\theta(\mathbf{x}_1, 1)||^2 - \log p_\theta(\mathbf{w}|\mathbf{x}_0)\right]$$

https://arxiv.org/pdf/2205.14217.pdf

PLANNER

- VAE + autoregressive decoder

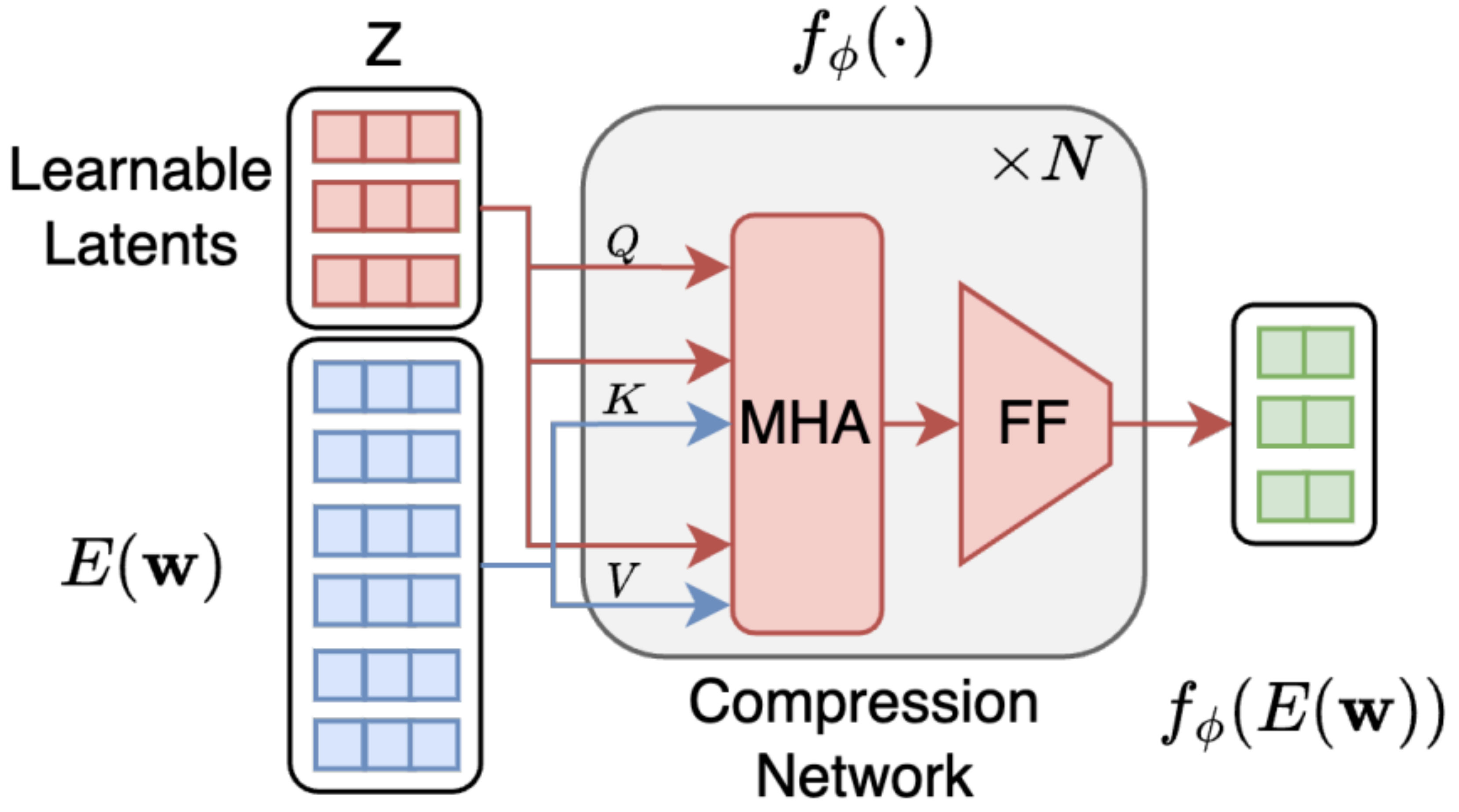


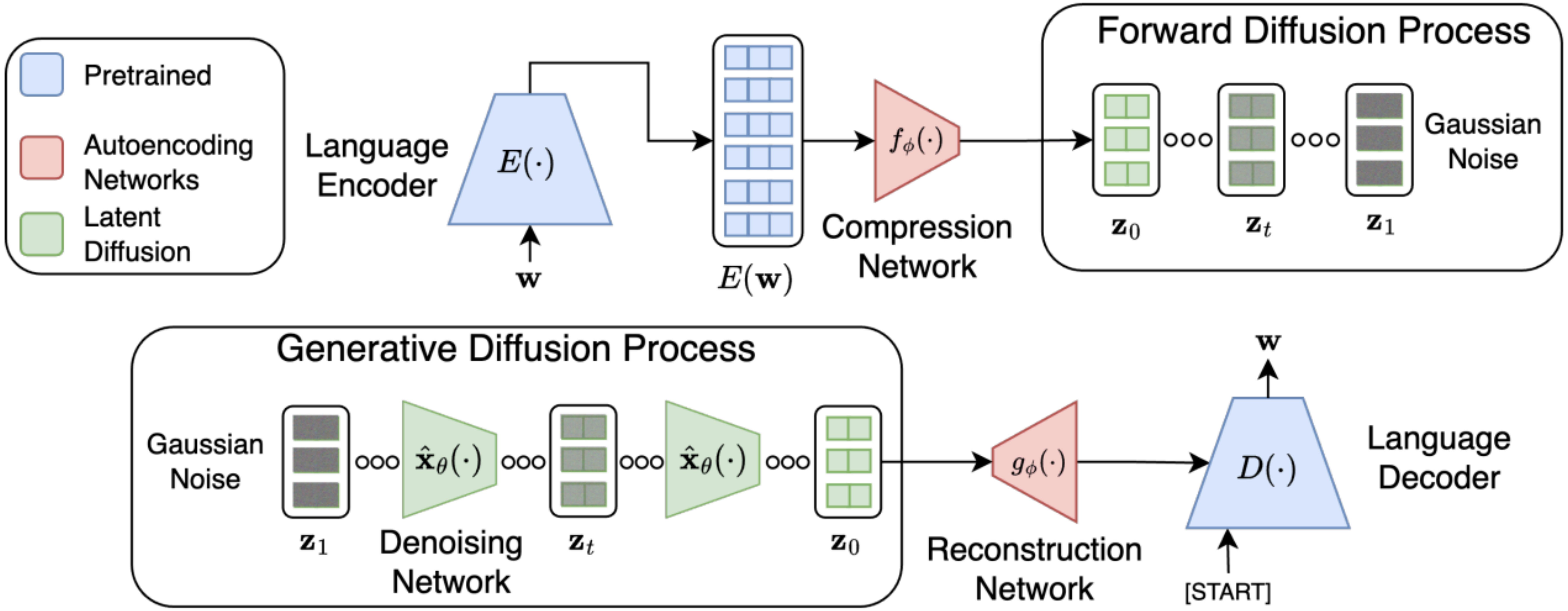
https://arxiv.org/pdf/2306.02531.pdf

LD4LG

Latent Diffusion for Language Generation

- Latent space + autoregressive decoder





https://arxiv.org/pdf/2212.09462.pdf

Трюки для текстовой диффузии

Трюки для обучения

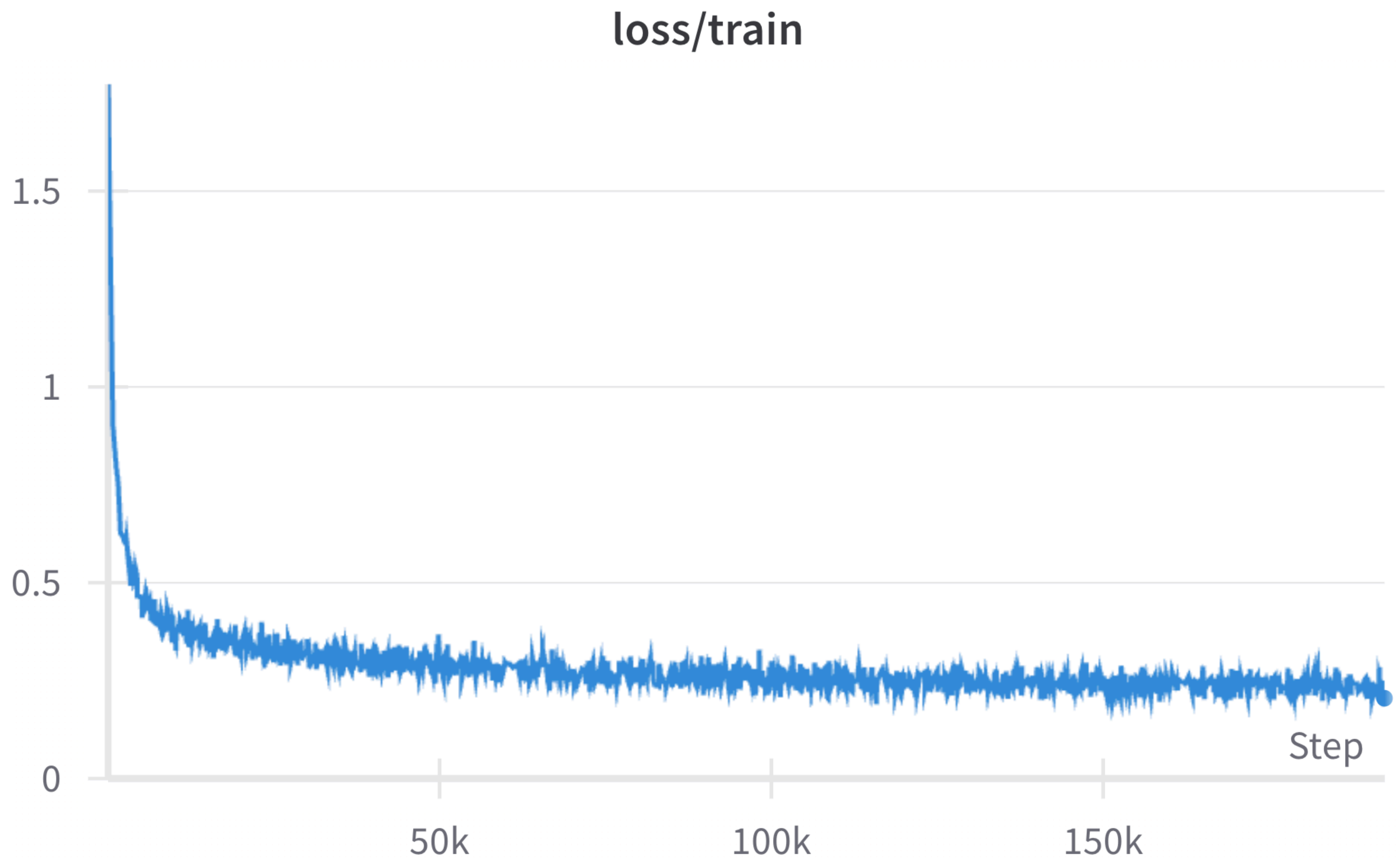
Предсказание x_0 вместо ε_t работает эффективнее для диффузии на дискретных данных.

Никто толком не понимает, почему.

$$x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t} \varepsilon_t$$

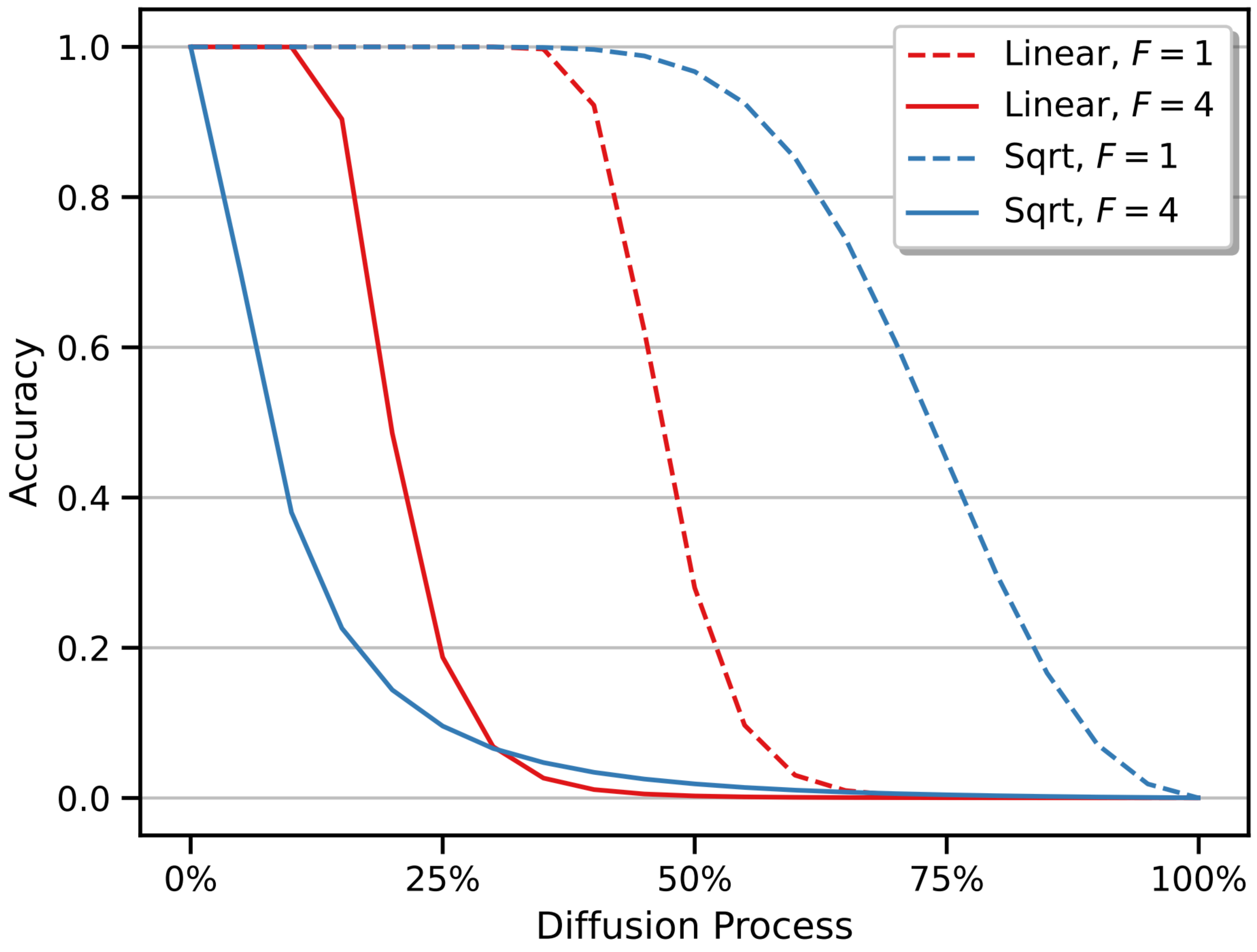
Трюки для обучения

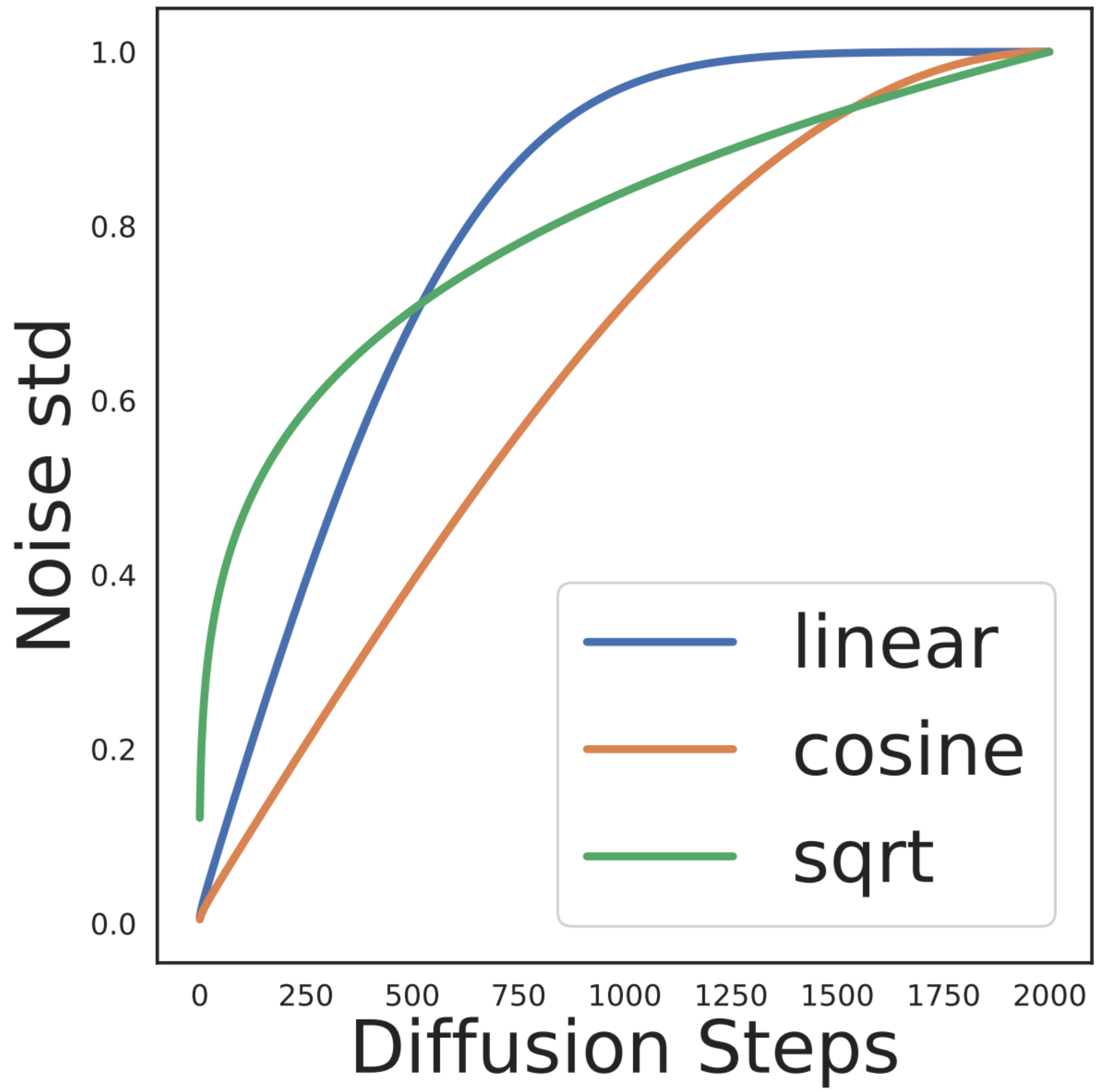
Диффузионные модели надо учить **долго**.
Даже если лосс не падает.



Трюки для обучения

Лучше брать более агрессивное расписание зашумления.





Schedules	**BLEU** $F = 1$	$F = 4$
Linear (Ho et al., 2020)	31.02	31.98
Cosine (Nichol & Dhariwal, 2021)	26.61	32.68
Sqrt (Li et al., 2022)	2.85	34.13
EDM (Karras et al., 2022)	29.16	30.09

Трюки для обучения

Нормализация энкодингов

- x_0 должен иметь единичную дисперсию.
- Нормировать лучше по всем признакам отдельно, так как они отличаются по норме.

Трюки для обучения

Clamping trick

Округление предсказанного $\hat{x}_0$ до ближайшего текста на каждом шаге.

$$\mathbf{x}_{t-1} = \sqrt{\bar{\alpha}} \cdot \mathrm{Clamp}(f_\theta(\mathbf{x}_t, t)) + \sqrt{1-\bar{\alpha}}\epsilon$$

https://arxiv.org/pdf/2205.14217.pdf

Архитектурные трюки: Encoder

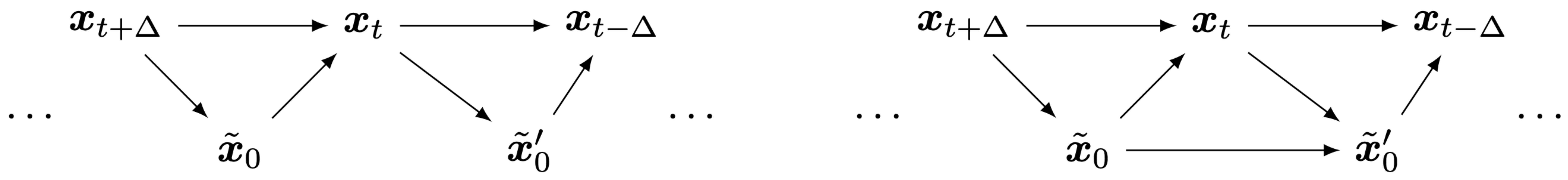
- Диффузия работает лучше, если латентное пространство более гладкое.
  - ➡ Encodings > embeddings
  - ➡ Encoder должен для "похожих" текстов получать похожие латенты
  - ➡ Учить encoder вместе с диффузией – плохая идея

Архитектурные трюки: Decoder

- Диффузионная модель аппроксимирует распределение $p(x)$.
- Качество аппроксимации падает с уменьшением числа шагов.
- При декодировании латентов хочется нивелировать ошибки.

➡ Лучше брать сложный декодер (несколько слоев трансформера)

Архитектурные трюки: Self-condition

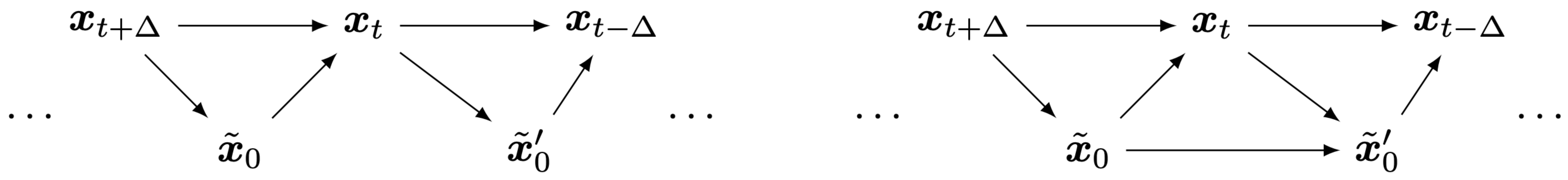


(a) Standard reverse diffusion steps.

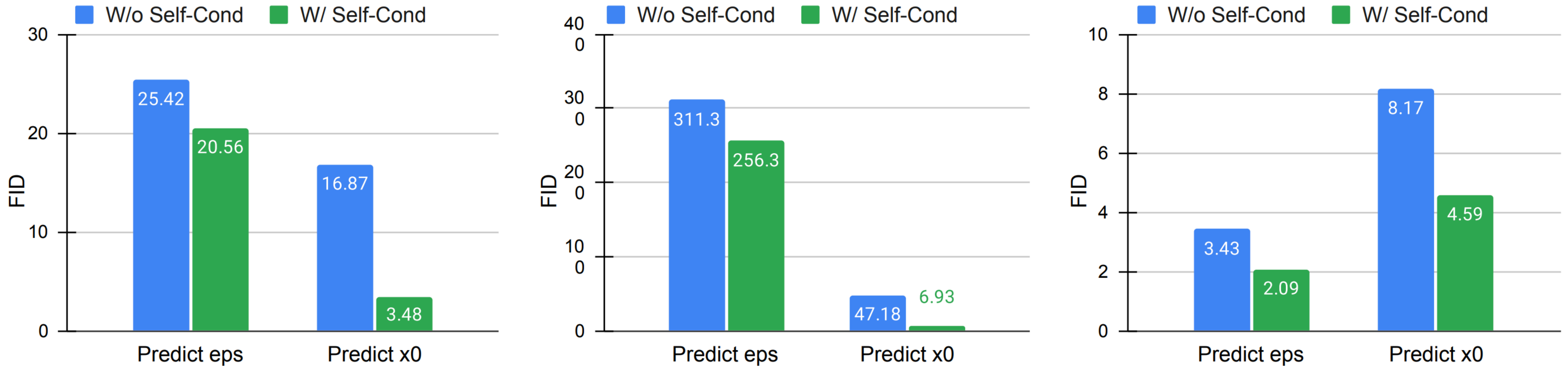
(b) Self-Conditioning on the previous $\boldsymbol{x}_0$ estimate.

https://arxiv.org/pdf/2208.04202.pdf

Архитектурные трюки: Self-condition



(a) Standard reverse diffusion steps. (b) Self-Conditioning on the previous x_0 estimate.



(a) CIFAR-10, UINT8. (b) CIFAR-10, UINT8 (RAND). (c) IMAGENET 64×64.

https://arxiv.org/pdf/2208.04202.pdf