Текстовые диффузионные модели

План

- Недостатки авторегрессионных моделей
- Диффузионные модели: напоминание
- Текстовые диффузионные модели:
  - Дискретные
  - Непрерывные

Авторегрессионные модели

Генерируют по одному токену

Я предсказываю следующий __

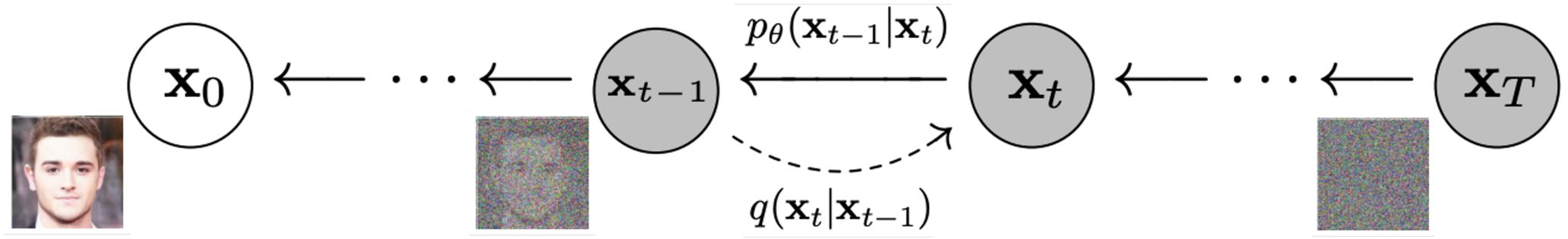
Недостатки:

- Нельзя исправлять уже сгенерированные токены
- Модель не может думать на несколько токенов вперед
- Необходимо выбирать метод семплирования

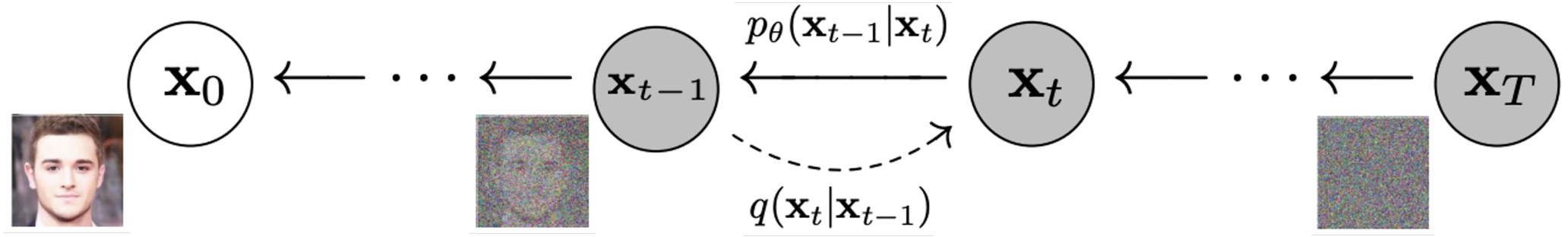
Диффузионные модели

Изначально диффузионные модели были созданы для генерации изображений

Идея: Будем постепенно добавлять шум к объекту в процессе прямой диффузии
Обучим модель восстанавливать исходное изображение из зашумленного



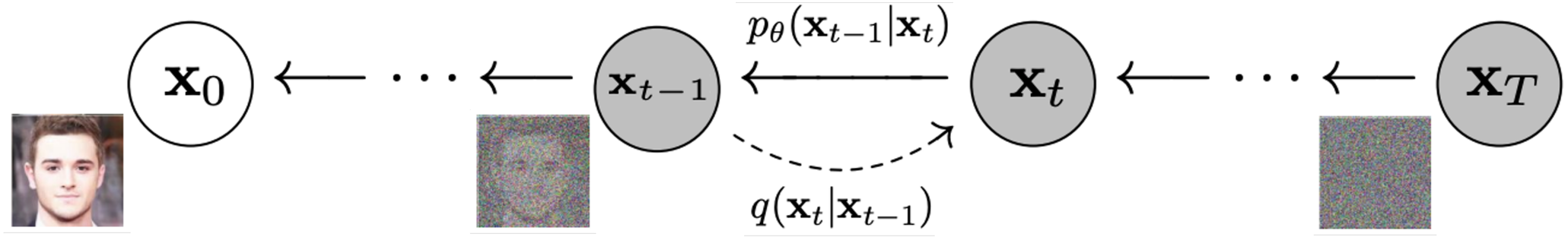
Диффузионные модели



$$q(x_t \mid x_{t-1}) = \mathcal{N}(x_t \mid \sqrt{\alpha_t} x_{t-1}, (1 - \alpha_t) I)$$

$$\alpha_t \in [0, 1]$$

Диффузионные модели



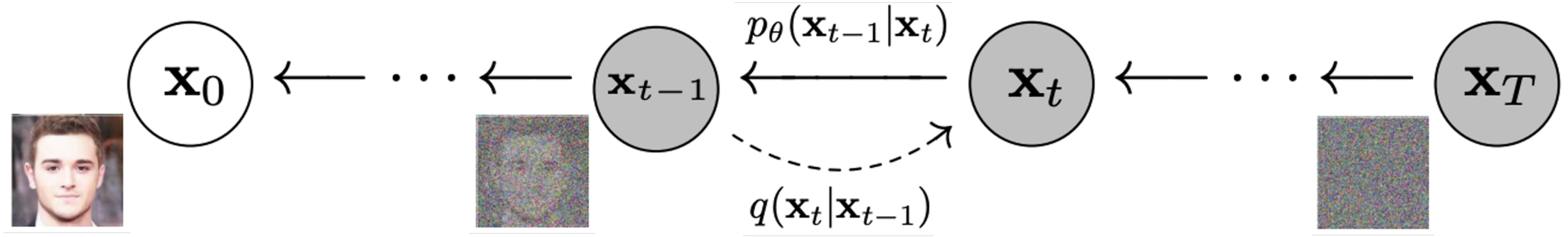
$$q(x_t \mid x_{t-1}) = \mathcal{N}(x_t \mid \sqrt{\alpha_t} x_{t-1}, (1 - \alpha_t) I)$$

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

$$\alpha_t \in [0, 1]$$

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

Диффузионные модели



$$q(x_t \mid x_{t-1}) = \mathcal{N}(x_t \mid \sqrt{\alpha_t} x_{t-1}, (1-\alpha_t) I)$$

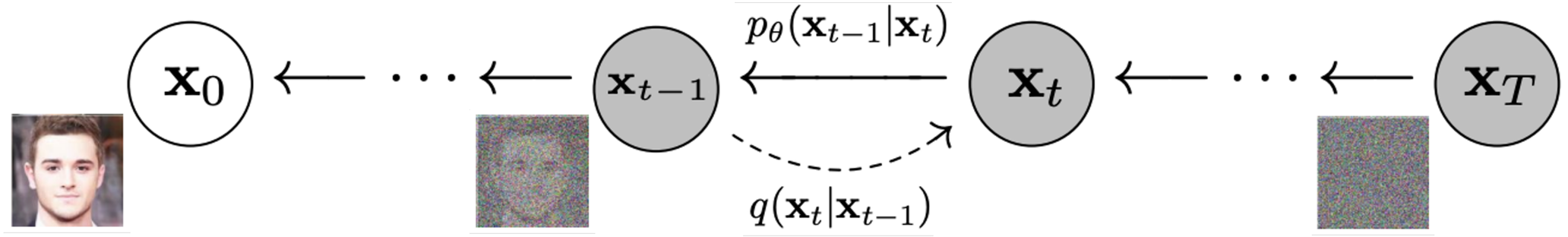
$$\alpha_t \in [0, 1]$$

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

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

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

Диффузионные модели



$$q(x_t \mid x_{t-1}) = \mathcal{N}(x_t \mid \sqrt{\alpha_t} x_{t-1}, (1-\alpha_t) I)$$

$$\alpha_t \in [0, 1]$$

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

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

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

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

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

Обучение и генерация

Algorithm 1 Training

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

Можно также предсказывать $\tilde{\mu}_t$ или x_0

Однако для изображений предсказание ε_t работает лучше

Почему ДМ сложно применять для текста?

Тексты **дискретны** по своей природе

Не понятно, как можно добавлять к ним шум

Почему ДМ сложно применять для текста?

Тексты **дискретны** по своей природе

Не понятно, как можно добавлять к ним шум

Два направления:

- **Дискретная диффузия** – уничтожение информации путем замены одних токенов другими
- **Непрерывная диффузия** – отображение текста в непрерывное пространство и осуществление диффузии уже в нем

Дискретная диффузия

Дискретная диффузия

Идея: Введем стохастическую матрицу Q_t, которая задает вероятности изменения токенов

$$Q_t[i, j] = p(x_t = j \,|\, x_{t-1} = i)$$

В процессе зашумления будем семплировать объекты из категориального распределения

$$q(x_t \,|\, x_{t-1}) = \mathrm{Cat}(x_t \,|\, p = x_{t-1} Q_t)$$

x_t – one-hot вектор

Примеры Q_t

Uniform: интерполяция между вырожденным распределением и равномерным

$$Q_t = (1-\beta_t)\begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \dots & 0 & 1 \end{pmatrix} + \frac{\beta_t}{|V|}\begin{pmatrix} 1 & 1 & \dots & 1 \\ 1 & 1 & & \vdots \\ \vdots & & \ddots & 1 \\ 1 & \dots & 1 & 1 \end{pmatrix}$$

T = 0	The great brown fox hopped over the lazy dog.
T = 10	The vast black fox hopping over the lazy cat.
T = 20	Their vast tripped this jumping upon walked organizations.
T = 25	Bunk scamper tripped this Sanchez walked organizations.

https://arxiv.org/abs/2107.03006

Примеры Q_t

Absorbing: Все токены превращаются в токены [MASK]

$$Q_t[i,j] = \begin{cases} 1, & i = j = m \\ 1 - \beta_t, & i = j \neq m \\ \beta_t, & i \neq m, j = m \end{cases}$$

T = 0	The great brown fox hopped over the lazy dog.
T = 10	The great [MASK] fox hopped over [MASK] lazy dog.
T = 20	The [MASK][MASK] [MASK] ship over [MASK] lazy the.
T = 25	[MASK] [MASK] [MASK] [MASK] [MASK] [MASK] [MASK] [MASK] [MASK]

https://arxiv.org/abs/2107.03006

Обучение дискретной диффузии

Для получения функции ошибки, нам нужно максимизировать правдоподобие $p(x_{t-1} \mid x_t, x_0)$

$$p(x_{t-1} \mid x_t, x_0) \propto q(x_t \mid x_{t-1}) q(x_{t-1} \mid x_0)$$

Обучение дискретной диффузии

Для получения функции ошибки, нам нужно максимизировать правдоподобие $p(x_{t-1} \mid x_t, x_0)$

$$p(x_{t-1} \mid x_t, x_0) \propto q(x_t \mid x_{t-1}) q(x_{t-1} \mid x_0)$$

или максимизировать нижнюю вариационную оценку (VLB), что то же самое

$$\log p_\theta(x_0) = \log \int q(x_{1:T} \mid x_0) \frac{p_\theta(x_{0:T})}{q(x_{1:T} \mid x_0)} dx_{1:T} \geq \mathbb{E}_{q(x_{1:T} \mid x_0)} \left[\log p_\theta(x_{0:T}) - \log q(x_{1:T} \mid x_0) \right]$$

Обучение дискретной диффузии

Для получения функции ошибки, нам нужно максимизировать правдоподобие $p(x_{t-1} \mid x_t, x_0)$

$$p(x_{t-1} \mid x_t, x_0) \propto q(x_t \mid x_{t-1}) q(x_{t-1} \mid x_0)$$

или максимизировать нижнюю вариационную оценку (VLB), что то же самое

$$\log p_\theta(x_0) = \log \int q(x_{1:T} \mid x_0) \frac{p_\theta(x_{0:T})}{q(x_{1:T} \mid x_0)} dx_{1:T} \geq \mathbb{E}_{q(x_{1:T} \mid x_0)} \left[\log p_\theta(x_{0:T}) - \log q(x_{1:T} \mid x_0) \right]$$

Однако на практике чаще всего используют обычную кросс-энтропию

$$L_\theta = - \mathbb{E}_{q(x_t \mid x_0)} \left[\log p_\theta \left(x_0 \mid x_t \right) \right]$$

Concrete Score Matching

Пусть $\mathcal{N}(x)$ – окрестность x, $\mathcal{N}(x) = \{x_{n_1}, \ldots, x_{n_k}\}$

Тогда *concrete score* задается как

$$c_p(x; \mathcal{N}) = \left[\frac{p(x_{n_1})}{p(x)}, \ldots, \frac{p(x_{n_k})}{p(x)}\right] - 1$$

https://arxiv.org/abs/2211.00802

Concrete Score Matching

Пусть $\mathcal{N}(x)$ – окрестность x, $\mathcal{N}(x) = \{x_{n_1}, \ldots, x_{n_k}\}$

Тогда *concrete score* задается как

$$c_p(x; \mathcal{N}) = \left[\frac{p(x_{n_1})}{p(x)}, \ldots, \frac{p(x_{n_k})}{p(x)}\right] - 1$$

Утверждение: Для любого $x \in \mathbb{R}^d$ и $\delta > 0$ пусть $\mathcal{N}_\delta = \{x + \delta \mathbf{e}_i\}_{i=1}^d$. Тогда

$$\lim_{\delta \to 0} \frac{c_p\left(x; \mathcal{N}_\delta\right)}{\delta} = \nabla_x \log p(x)$$

Доказательство:

$$\lim_{\delta \to 0} \left\{ \frac{p(x + \delta \mathbf{e}_i) - p(x)}{\delta \cdot p(x)} \right\}_{i=1}^{d} = \frac{1}{p(x)} \nabla_x p(x)$$

https://arxiv.org/abs/2211.00802

Concrete Score Matching

Оказывается, гораздо лучше научиться предсказывать *concrete score*

$$s_\theta(x, t) \approx \left[\frac{p_t(y)}{p_t(x)}\right]_{x \neq y}$$

Для обучения можно взять MSE

$$L_{\text{CSM}} = \frac{1}{2} \mathop{\mathbb{E}}_{\substack{x_0 \sim p_0 \\ x_t \sim p(\,.\,|\,x_0)}} \left[\sum_{y \neq x}^{|V|} \left(s_\theta(x_t, t)_y - \frac{p_t(y\,|\,x_0)}{p_t(x\,|\,x_0)} \right)^2 \right]$$

https://arxiv.org/abs/2211.00802

Concrete Score Matching

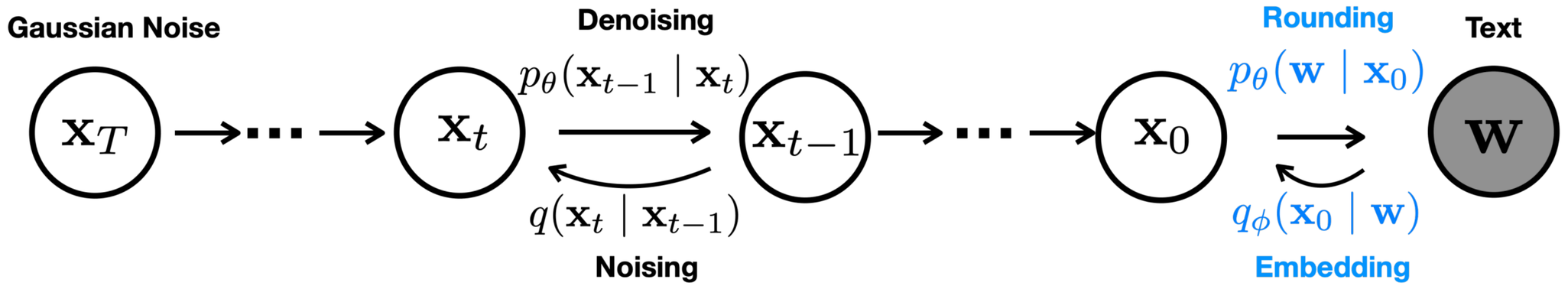
Size	Model	LAMBADA	WikiText2	PTB	WikiText103	1BW
Small	GPT-2	**45.04**	42.43	138.43	41.60	**75.20**
	SEDD Absorb	≤50.92	≤**41.84**	≤**114.24**	≤**40.62**	≤79.29
	SEDD Uniform	≤65.40	≤50.27	≤140.12	≤49.60	≤101.37
	D3PM	≤93.47	≤77.28	≤200.82	≤75.16	≤138.92
	PLAID	≤57.28	≤51.80	≤142.60	≤50.86	≤91.12
Medium	GPT-2	**35.66**	31.80	123.14	31.39	**55.72**
	SEDD Absorb	≤42.77	≤**31.04**	≤**87.12**	≤**29.98**	≤61.19
	SEDD Uniform	≤51.28	≤38.93	≤102.28	≤36.81	≤79.12

Перплексия (↓) для безусловной генерации на наборе датасетов

https://arxiv.org/abs/2310.16834

Непрерывная диффузия

Непрерывная диффузия



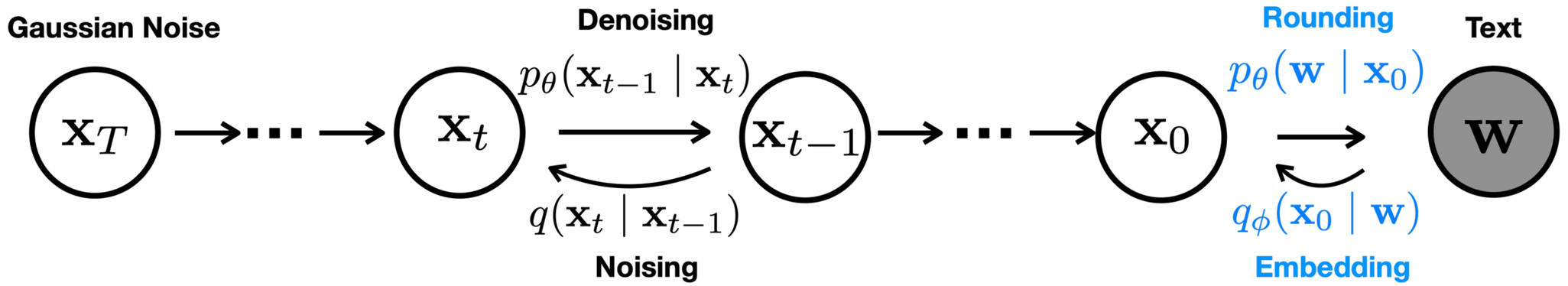
1. Переводим токены в эмбеддинги

2. Выполняем диффузию в пространстве эмбеддингов

3. Округляем сгенерированные эмбеддинги до ближайших токенов

https://arxiv.org/abs/2205.14217

Непрерывная диффузия

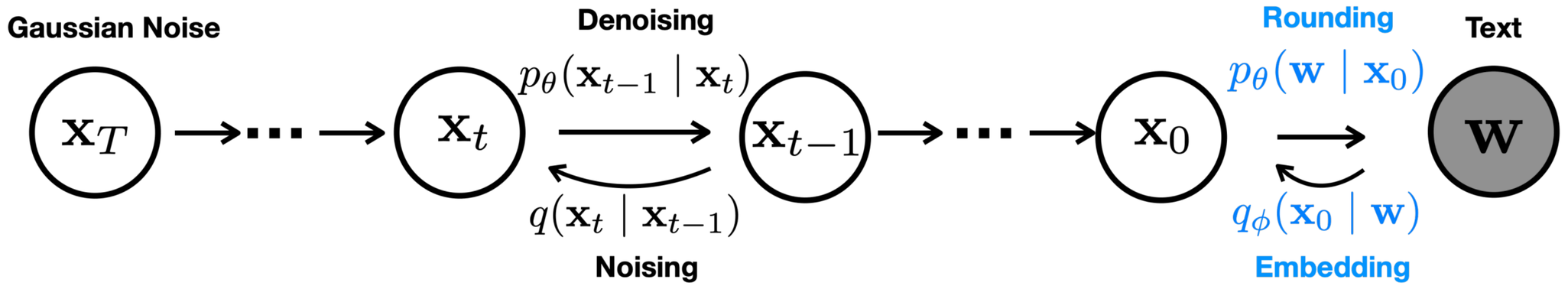


$$\mathbf{x}_0 = \mathrm{Emb}(\mathbf{w}) \qquad \mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\varepsilon_t$$

$$L_{\text{simple}}(\mathbf{x}_0) = \sum_{t=1}^{T} \mathbb{E}_{\mathbf{x}_t} \|f_\theta(\mathbf{x}_t, t) - \mathbf{x}_0\|^2$$

https://arxiv.org/abs/2205.14217

Непрерывная диффузия



$$\mathbf{x}_0 = \mathrm{Emb}(\mathbf{w}) \qquad \mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\varepsilon_t$$

$$L_{\text{simple}}(\mathbf{x}_0) = \sum_{t=1}^{T} \mathbb{E}_{\mathbf{x}_t} \|f_\theta(\mathbf{x}_t, t) - \mathbf{x}_0\|^2$$

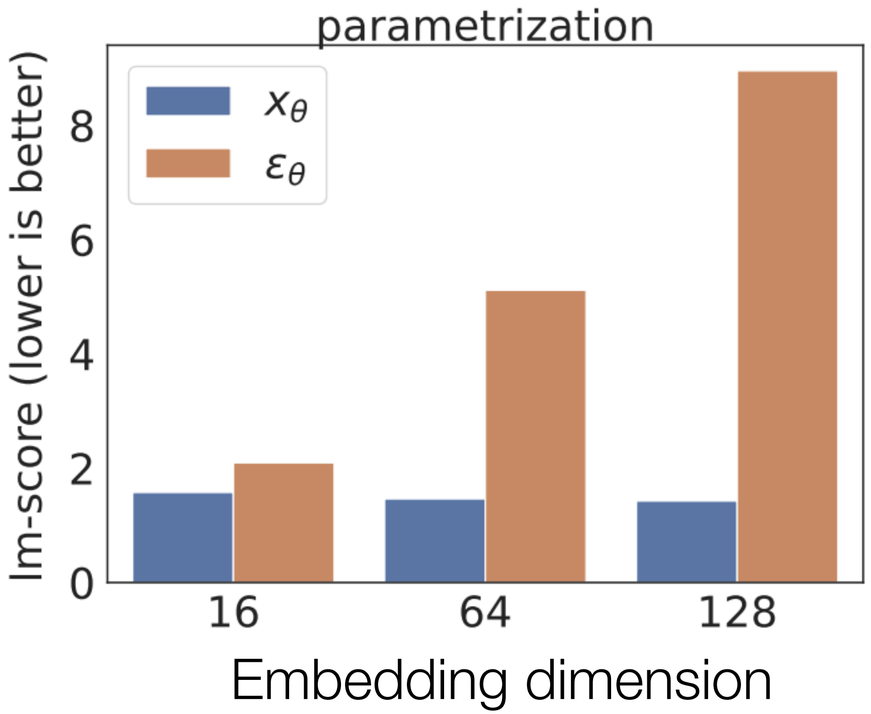
$$L^{\text{e2e}}_{\text{simple}}(\mathbf{w}) = \mathop{\mathbb{E}}_{q_\phi(\mathbf{x}_{0:T} \mid \mathbf{w})} \left[L_{\text{simple}}(\mathbf{x}_0) - \log p_\theta(\mathbf{w} \mid \mathbf{x}_0 + \sigma\varepsilon) \right]$$

Предотвращает схлопывание эмбеддингов

https://arxiv.org/abs/2205.14217

Параметризация модели

Оказывается, текстовую диффузию лучше обучать на предсказание $\mathbf{x}_0$ вместо ε

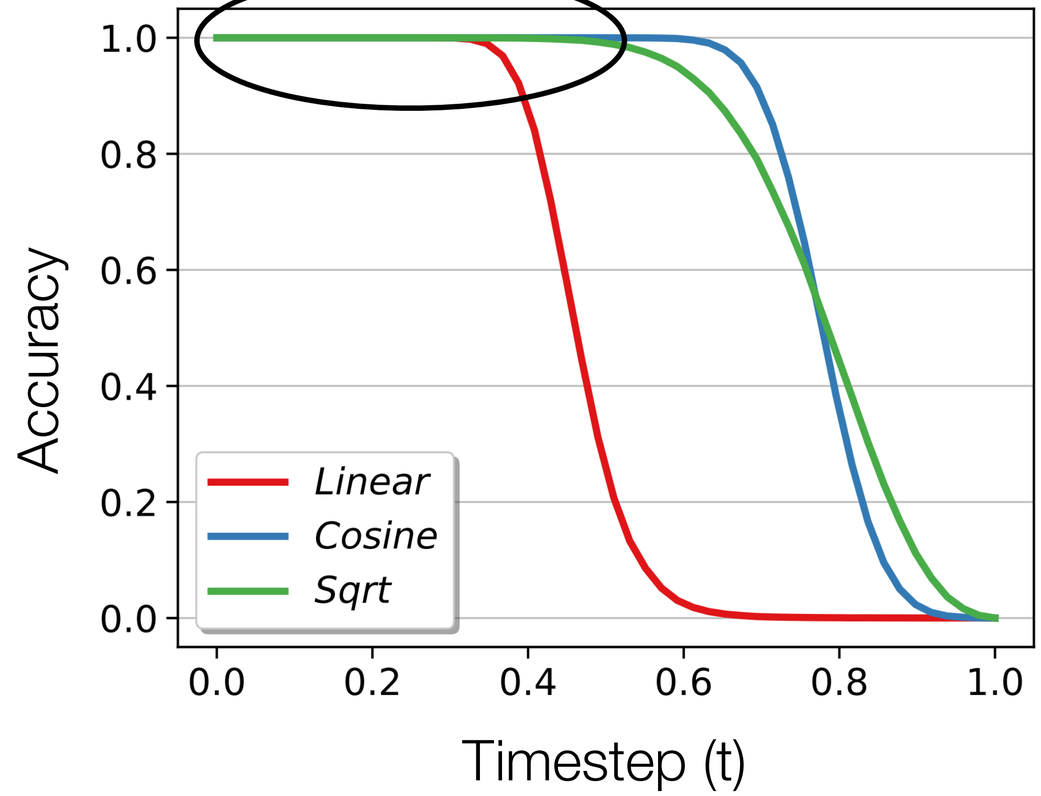


https://arxiv.org/abs/2205.14217

Текстовой диффузии нужно больше шума

Так как эмбеддинги имеют огромную размерность, нужно добавить много шума, чтобы их стало сложно различать

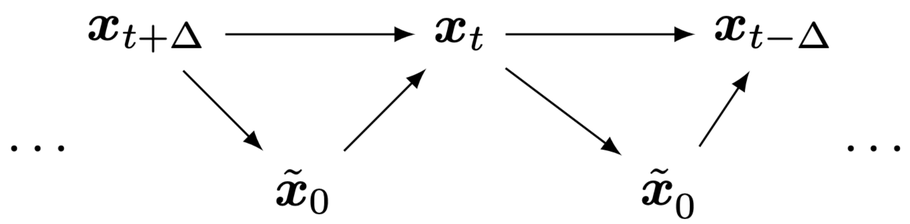
Эти шаги бесполезны для обучения



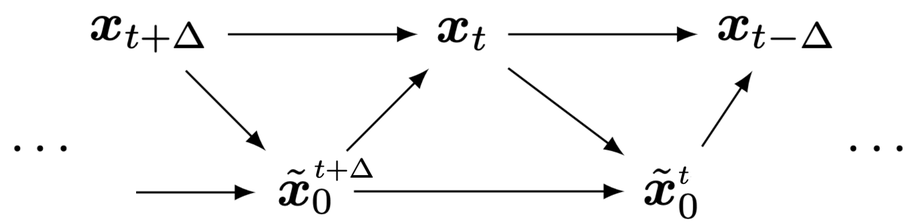
https://arxiv.org/abs/2205.14217

Self-conditioning

Идея: позволим модели обуславливаться на свои собственные предсказания



(a) Standard reverse diffusion steps.



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

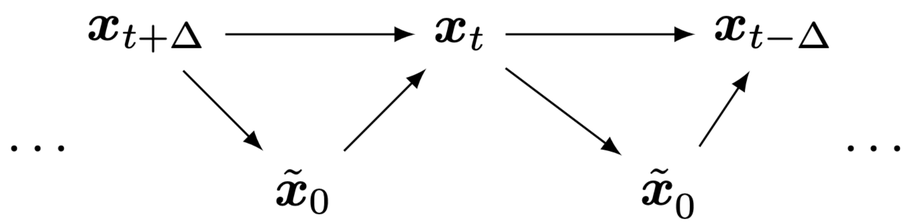
$$\tilde{x}_0^t = f_\theta(x_t, t)$$

$$\tilde{x}_0^t = f_\theta(x_t, t, \tilde{x}_0^{t+\Delta})$$

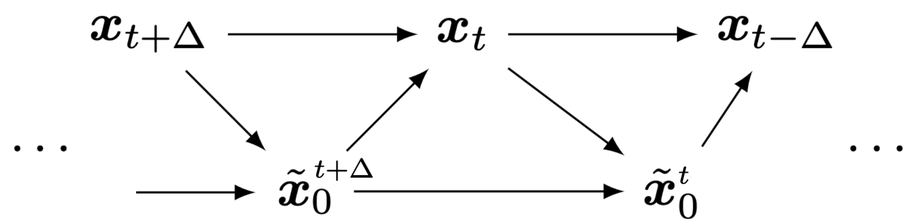
https://arxiv.org/abs/2208.04202

Self-conditioning

Идея: позволим модели обуславливаться на свои собственные предсказания



(a) Standard reverse diffusion steps.



(b) Self-Conditioning on the previous x_0 estimate.

$$\tilde{x}_0^t = f_\theta(x_t, t)$$

$$\tilde{x}_0^t = f_\theta(x_t, t, \tilde{x}_0^{t+\Delta})$$

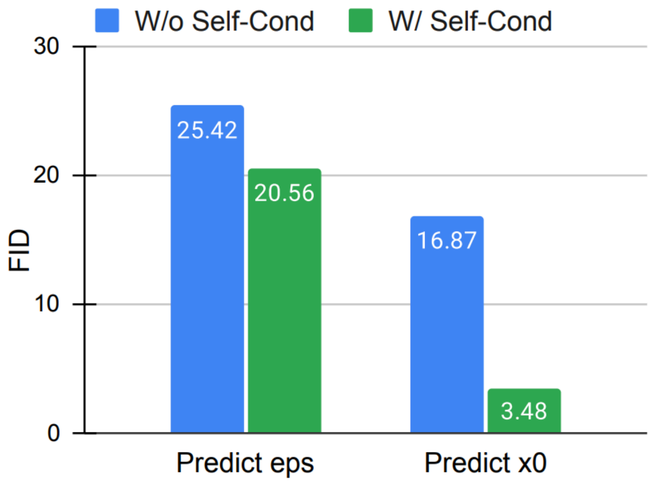
Модификация процесса обучения:

```
if rand(0, 1) > 0.5:
    pred_x_0 = model(x_t, t, 0)
else:
    with no_grad():
        sc_x_0 = model(x_t, t, 0)
    pred_x_0 = model(x_t, t, sc_x_0)
loss = mse(pred_x_0, x_0)
```

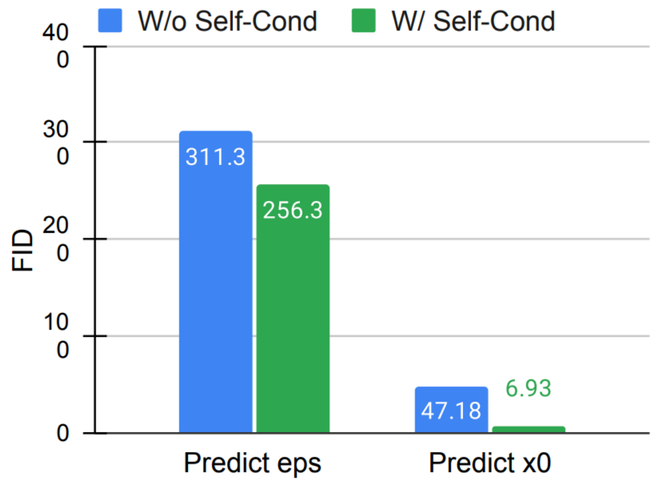
https://arxiv.org/abs/2208.04202

Self-conditioning

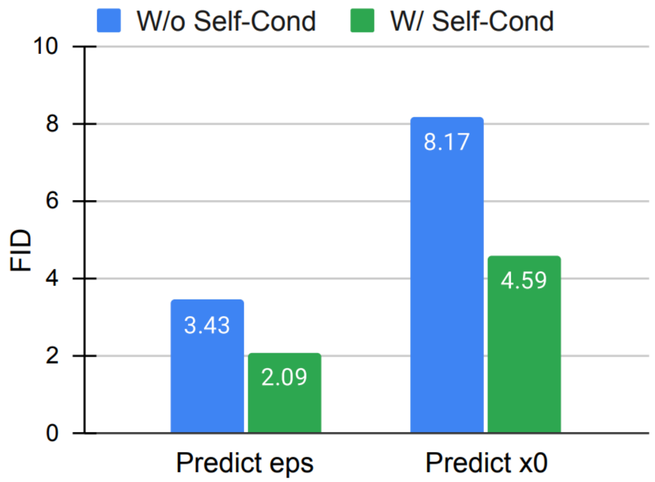
Многие работы экспериментально подтверждают успешность self-conditioning
Однако, объяснений, почему он работает почти нет



(a) CIFAR-10, UINT8.



(b) CIFAR-10, UINT8 (RAND).



(c) IMAGENET 64×64.

https://arxiv.org/abs/2208.04202

Self-conditioning повышает уверенность модели

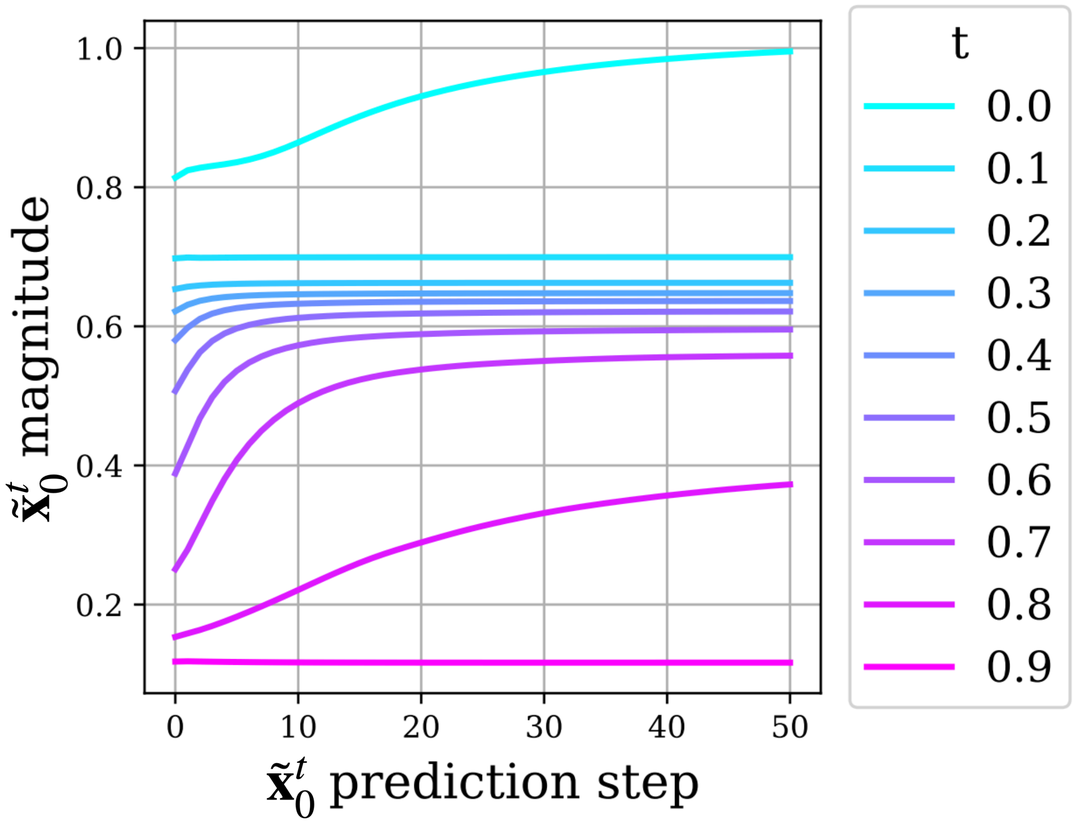
Эксперимент:

Предскажем $\tilde{\mathbf{x}}_0^t$ несколько раз из одного $\mathbf{x}_t$, меняя только условие

```
pred_x_0 = 0
for i in range(K):
    pred_x_0 = model(x_t, t, pred_x_0)
```

Измерим магнитуду предсказаний

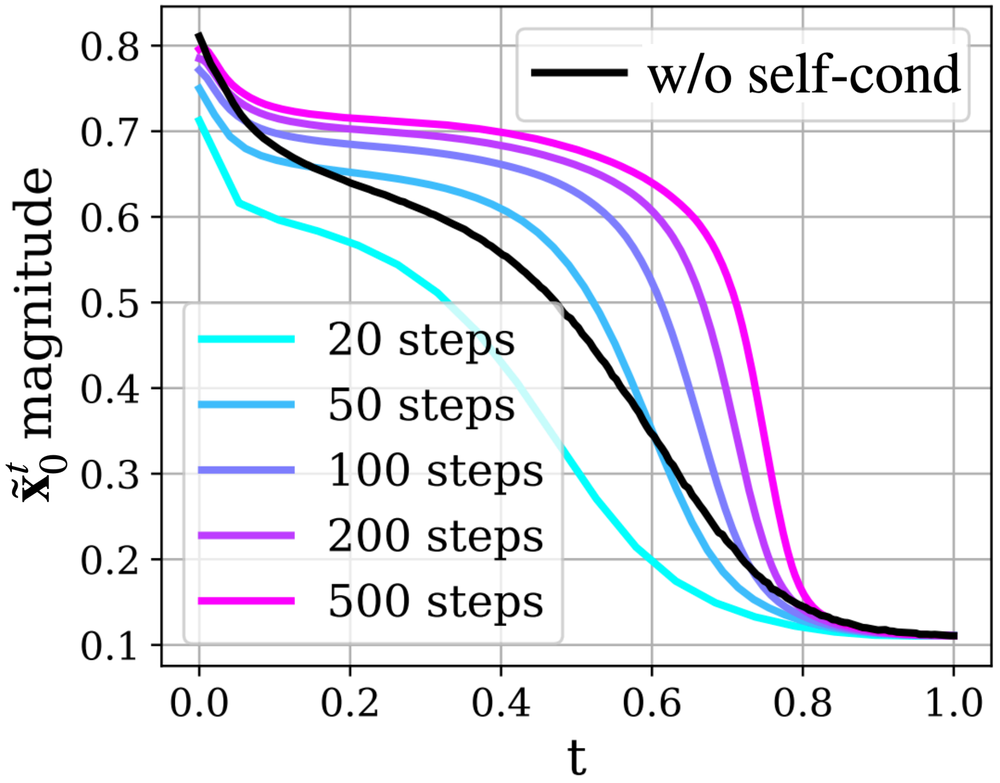
Оказывается, **магнитуда растет**



https://arxiv.org/abs/2402.19097

Self-conditioning меняет динамику генерации

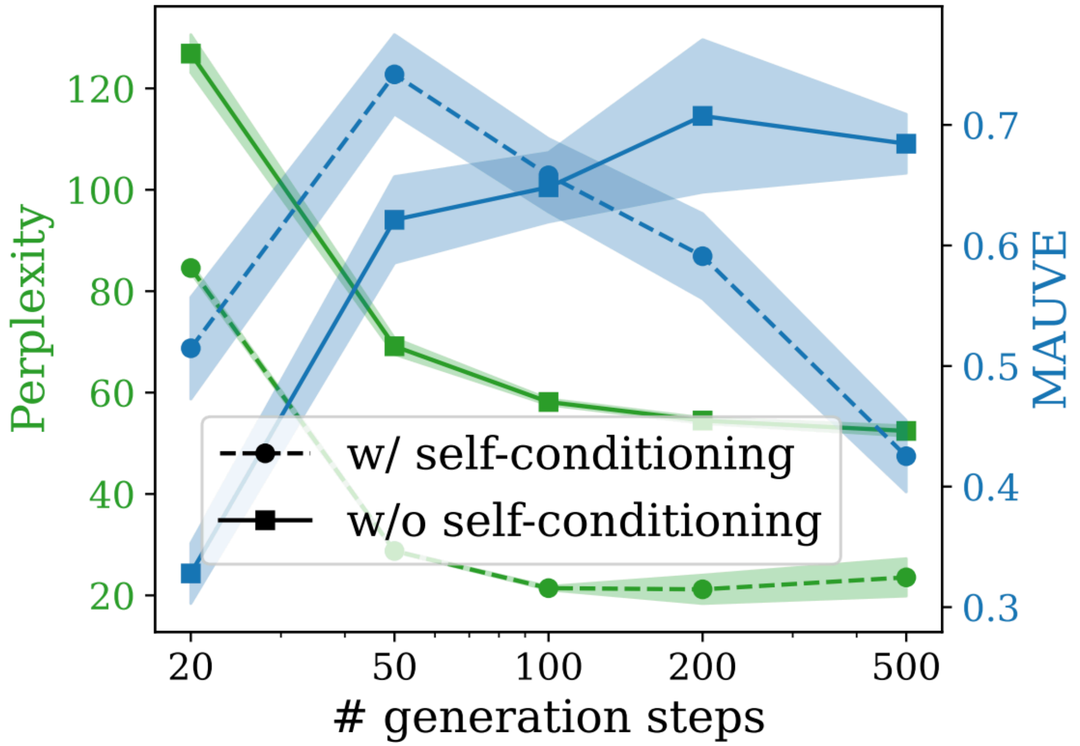
- Мы хотим, чтобы траектория генерации была как можно ближе к траектории прямого процесса
- Увеличивая количество шагов, мы увеличиваем магнитуду предсказаний => Несоответствие между self-conditioning на обучении и генерации
- Из этого графика следует, что оптимально делать 50 шагов генерации



https://arxiv.org/abs/2402.19097

Self-conditioning меняет динамику генерации

Лучше сделать меньше шагов с self-conditioning, чем больше без него



https://arxiv.org/abs/2402.19097

Альтернативные пространства $\mathbf{x}_0$

Текстовая диффузия работает заметно хуже картиночной, хотя отличается от нее только пространством
=> Вид пространства очень важен и его надо выбирать лучше

Альтернативные пространства $\mathbf{x}_0$

Текстовая диффузия работает заметно хуже картиночной, хотя отличается от нее только пространством
=> Вид пространства очень важен и его надо выбирать лучше

Варианты пространств диффузии:

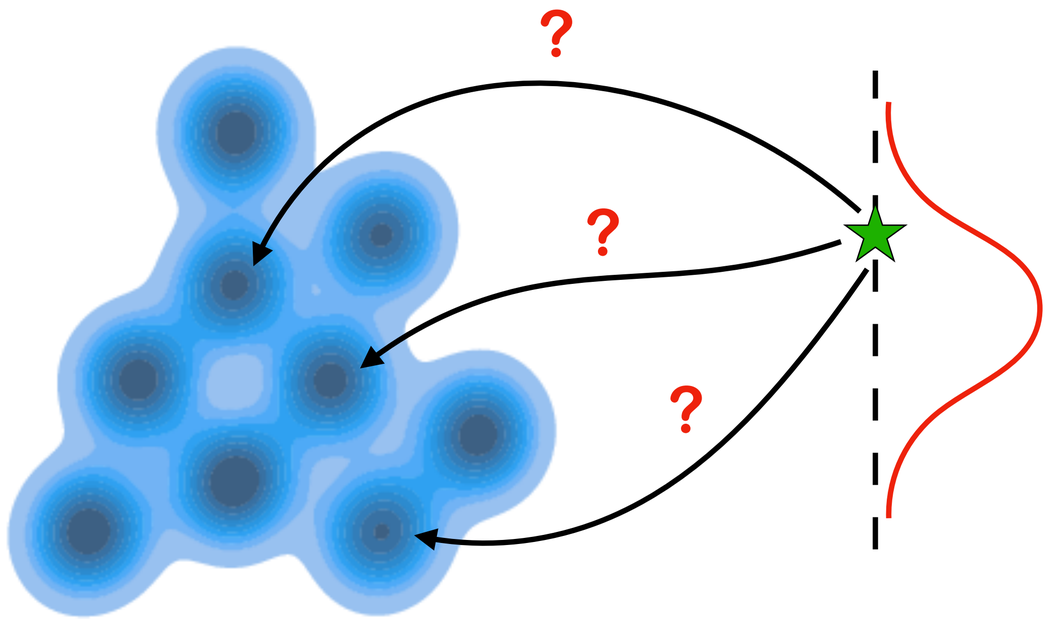
- Латентное пространство
- Пространство выходов языковой модели
- Симплекс

Латентное пространство

Сжимать текст в пространство малой размерности очень сложно из-за огромной мультимодальности

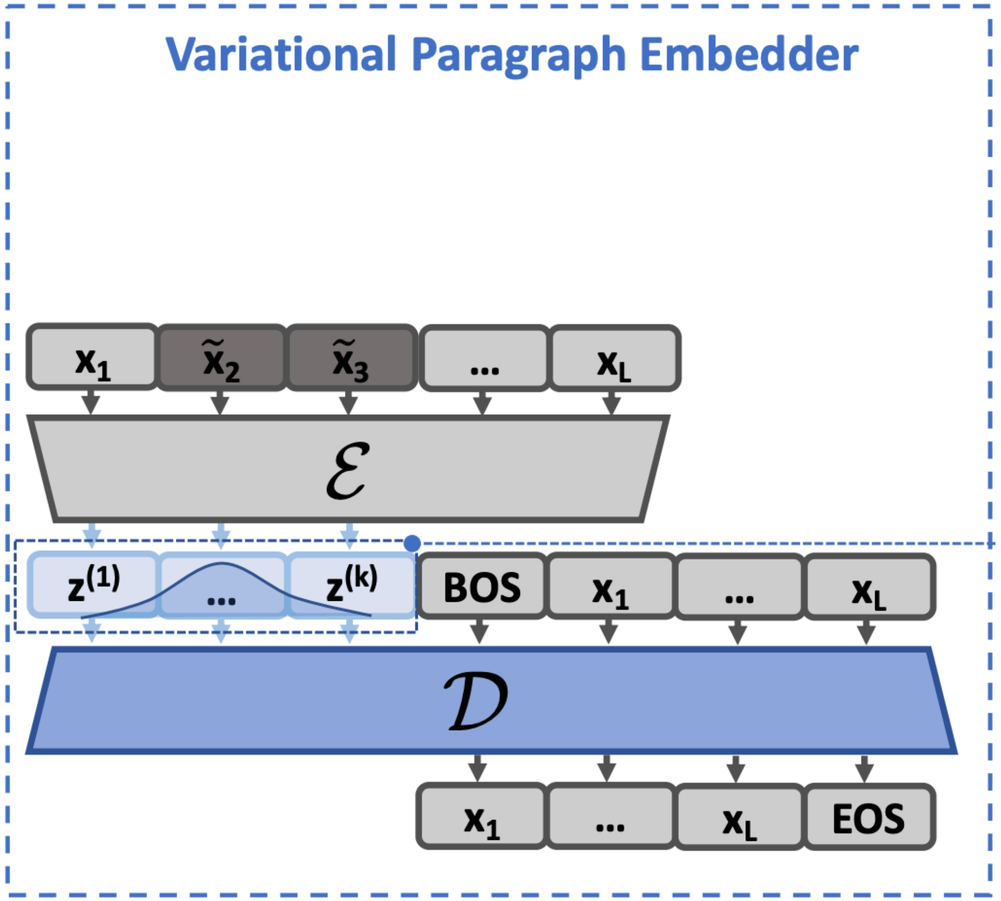
Пример: VAE не учится для текстов

Но хочется, потому что в пространстве малой размерности модели эффективнее работают



PLANNER, 2024

- Построим энкодер E, сжимающий текст в латент фиксированного размера
- И декодер D, декодирующий текст **авторегрессионно**
- Будем приближать латентное пространство к гауссовскому
- Диффузия в гауссовском пространстве отлично работает



$$\mathbf{z} = E(\mathbf{x}) \qquad \hat{\mathbf{x}} = D(\mathbf{z})$$

$$L(E, D; \mathbf{x}) = \mathbb{E}_{q(\mathbf{z}|\mathbf{x})}[\log p_D(\mathbf{x} \mid \mathbf{z})] - \beta \cdot \mathrm{KL}(q_E(\mathbf{z} \mid \mathbf{x}) \| p(\mathbf{z}))$$

https://arxiv.org/abs/2306.02531

LD4LG, 2023

- Будем сжимать текст с помощью Compression Network
- Декодер так же **авторегрессионный**
- Никакие дополнительные требования к пространству не накладываются

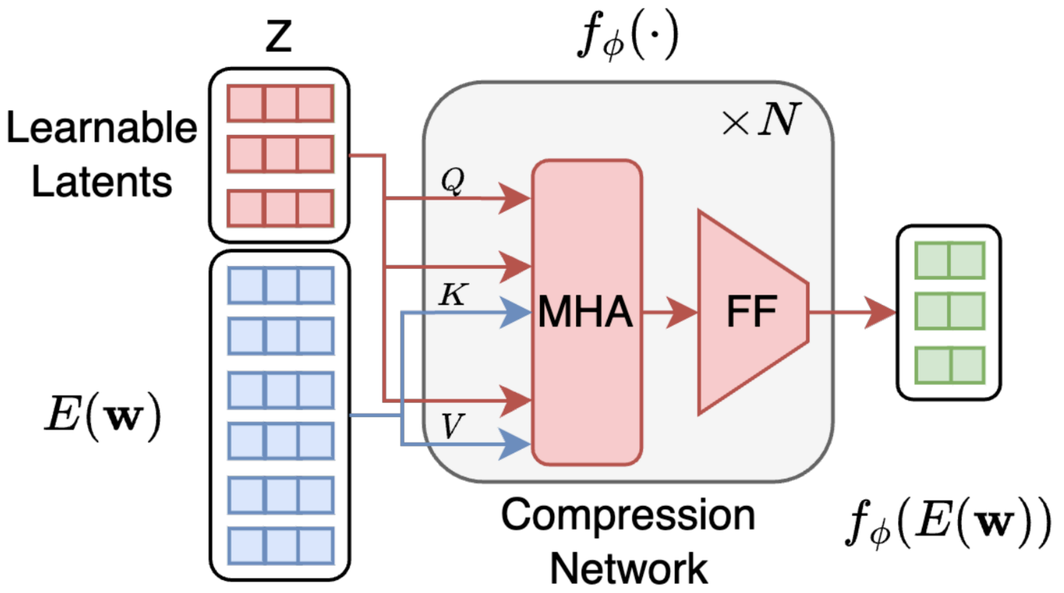
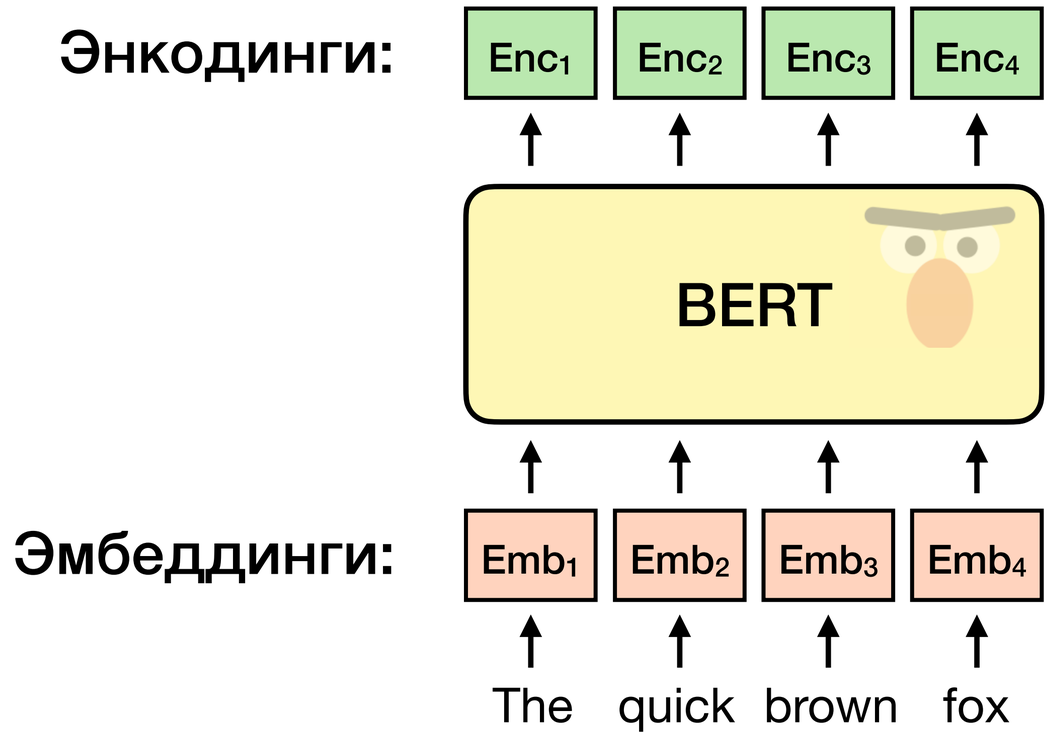


Figure 2: Architecture of our Compression Network.

https://arxiv.org/abs/2212.09462

Пространство выходов языковой модели

- Можно учить диффузию в пространстве выходов BERT
- Оно менее вырожденное, чем пространство эмбедингов и в нем считывается контекстная информация



https://arxiv.org/abs/2402.19097

Пространство выходов языковой модели

По метрикам пространство энкодингов гораздо лучше пространства эмбеддингов

Encoder	ppl ↓	mem ↓	div ↑	mauve ↑
ROCStories				
BERT emb	$48.9_{.36}$	$.371_{.003}$	$.324_{.002}$	$.600_{.016}$
BERT	$29.1_{.89}$	$.453_{.003}$	$.295_{.002}$	$\mathbf{.762}_{.043}$
RoBERTa	$\mathbf{28.3}_{.33}$	$.443_{.003}$	$.302_{.002}$	$.647_{.019}$
T5	$31.3_{.54}$	$.427_{.003}$	$.312_{.004}$	$.706_{.024}$
BART	$34.1_{.52}$	$.441_{.006}$	$.299_{.005}$	$.705_{.030}$
Source text	21.7	.365	.403	.876
Wikipedia				
BERT emb	$156.1_{1.8}$	$.263_{.004}$	$.517_{.002}$	$.378_{.055}$
BERT	$\mathbf{104.4}_{2.1}$	$.286_{.002}$	$.504_{.003}$	$\mathbf{.874}_{.011}$
Source text	37.3	.122	.615	.957

Задача безусловной генерации

https://arxiv.org/abs/2402.19097

Обучение на симплексе

Зададим симплекс в виде

$$\mathbf{S}_0[w, i] = \begin{cases} k, & \text{if } i = w \\ -k, & \text{otherwise} \end{cases}, \quad \mathbf{S}_0 \in \mathbb{R}^{n \times |V|}$$

$$\mathbf{S}_t = \sqrt{\bar{\alpha}_t}\mathbf{S}_0 + \sqrt{1 - \bar{\alpha}_t}\varepsilon_t, \quad \varepsilon \sim \mathcal{N}(0, k^2 I)$$

https://arxiv.org/abs/2305.08379

Обучение на симплексе

Зададим симплекс в виде

$$\mathbf{S}_0[w, i] = \begin{cases} k, & \text{if } i = w \\ -k, & \text{otherwise} \end{cases}, \quad \mathbf{S}_0 \in \mathbb{R}^{n \times |V|}$$

$$\mathbf{S}_t = \sqrt{\bar{\alpha}_t}\mathbf{S}_0 + \sqrt{1 - \bar{\alpha}_t}\varepsilon_t, \quad \varepsilon \sim \mathcal{N}(0, k^2 I)$$

Тогда

$$\mathbf{x}_t = \text{softmax}(\mathbf{S}_t) \cdot \mathbf{E},$$

где $\mathbf{E}$ – матрица эмбеддингов

https://arxiv.org/abs/2305.08379

Обучение на симплексе

Зададим симплекс в виде

$$\mathbf{S}_0[w, i] = \begin{cases} k, & \text{if } i = w \\ -k, & \text{otherwise} \end{cases}, \quad \mathbf{S}_0 \in \mathbb{R}^{n \times |V|}$$

$$\mathbf{S}_t = \sqrt{\bar{\alpha}_t}\mathbf{S}_0 + \sqrt{1 - \bar{\alpha}_t}\varepsilon_t, \quad \varepsilon \sim \mathcal{N}(0, k^2 I)$$

Тогда

$$\mathbf{x}_t = \text{softmax}(\mathbf{S}_t) \cdot \mathbf{E},$$

где $\mathbf{E}$ – матрица эмбеддингов

$$L = \mathbb{E}_{t, q(\mathbf{S}_0), q(\mathbf{S}_t \mid \mathbf{S}_0)} \left[-\sum_{i=1}^{L} \log p_\theta(w_i \mid \mathbf{S}_t, t) \right]$$

Кросс-энтропия показывает себя лучше для дискретных данных

https://arxiv.org/abs/2305.08379

Подведем итоги

- Текстовая диффузия отличается от обычной дискретностью данных
- Два способа борьбы с этим:
  - Дискретная (категориальная) диффузия
  - Непрерывная диффузия в новом пространстве
- Дискретная диффузия вводит матрицу перехода Q для "зашумления"
- Непрерывная диффузия переводит текст в непрерывное (желательно гладкое) пространство
- Self-conditioning – техника, позволяющая модели обуславливаться на свои предсказания во время генерации. Она значительно повышает качество.