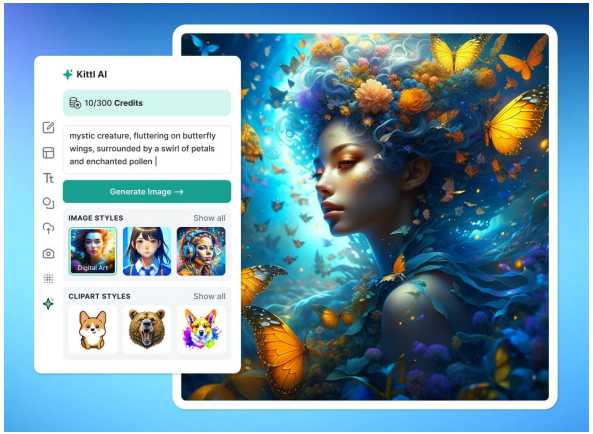
Latent Diffusion Models

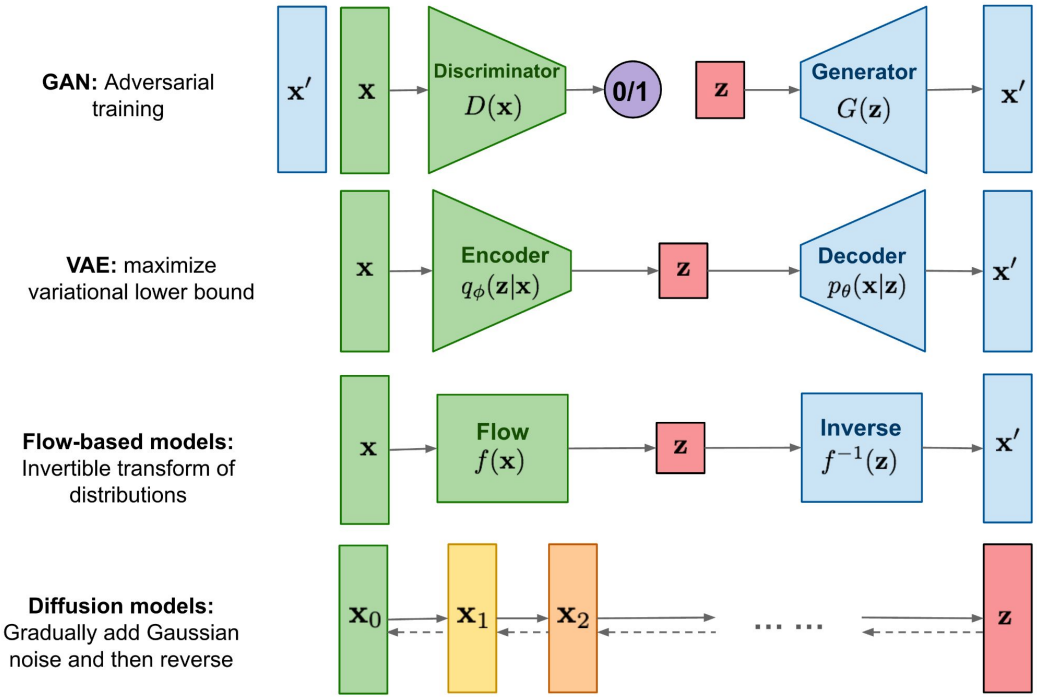
Описание задачи

Хотим генерировать качественные изображения, которые соответствуют текстовому описанию

- Как генерировать изображение?
- Как сделать так, чтобы оно соответствовало тексту

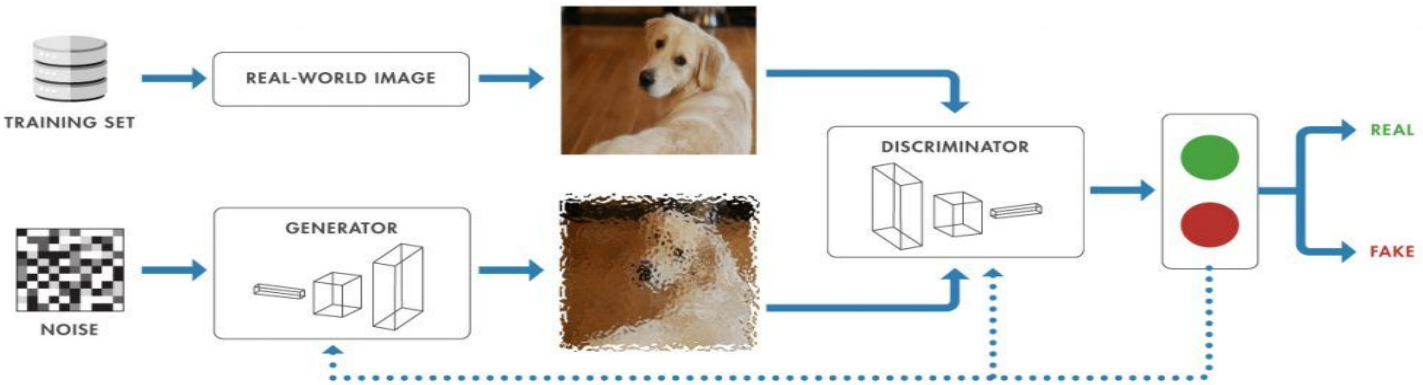


SOTA подходы, до появления LDM



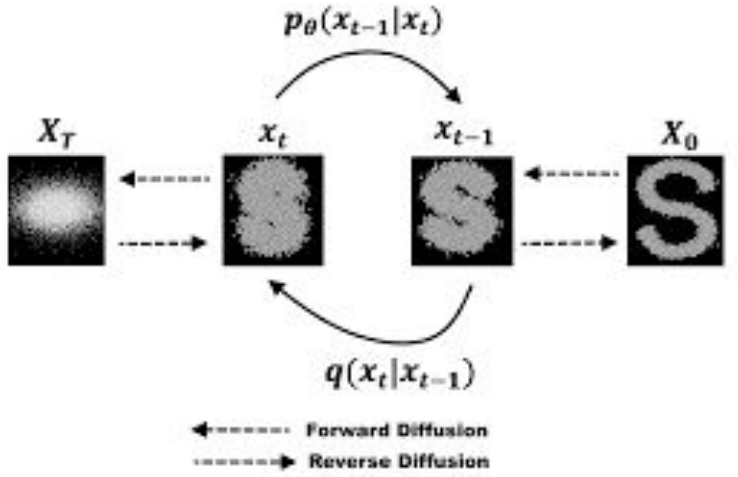
Generative Adversarial Networks (GAN)

- Генерируют качественно
- Сложно оптимизировать
- Плохо фиксируют распределение



Diffusion Probabilistic Models

- Лучшая оценка плотности
- Лучшее качество изображений
- Генерирует долго
- Требует больших мощностей



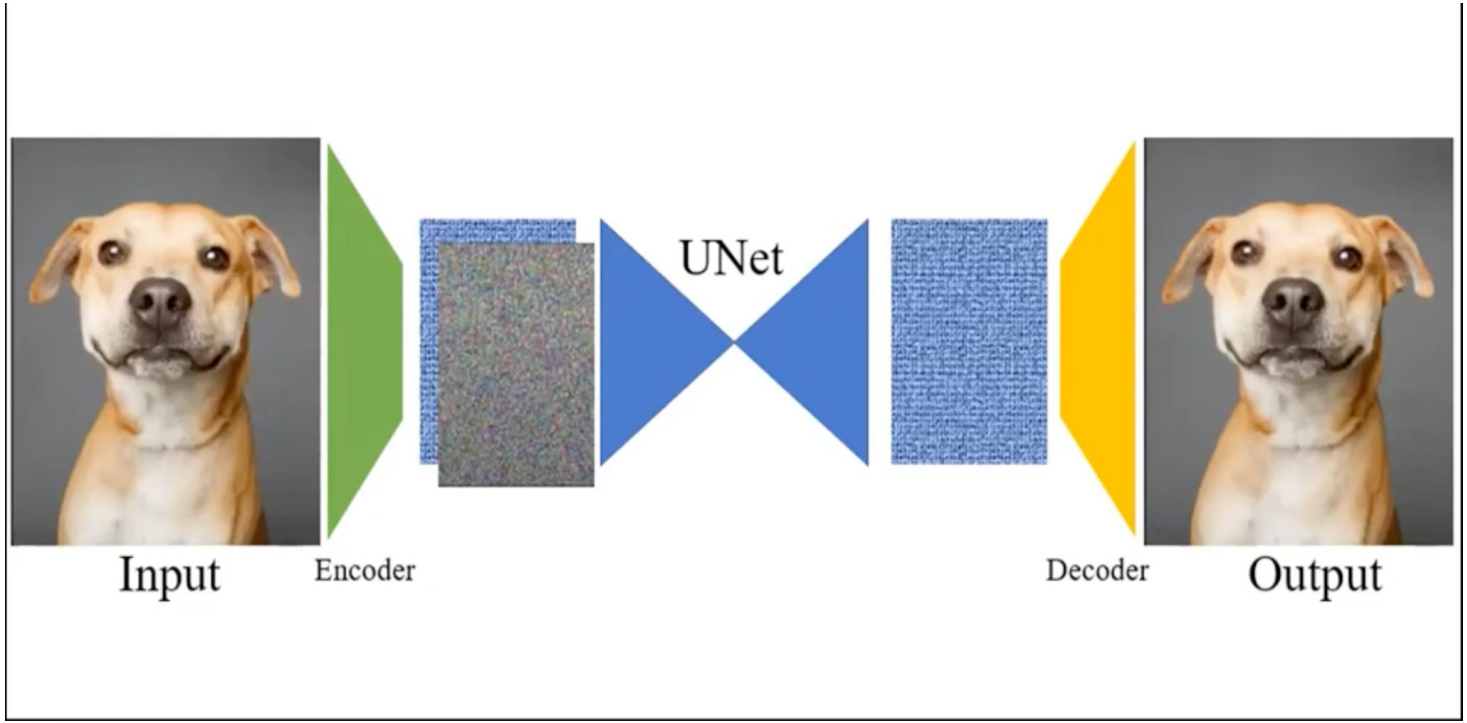
Two-Stage Image Synthesis

Попытаемся учить в две стадии, чтобы избежать недостатков индивидуальных моделей

- VQ-VAEs (autoregressive models)
- VQGANs (autoregressive transformers)

- Миллиарды параметров для обучения

Latent Diffusion Models



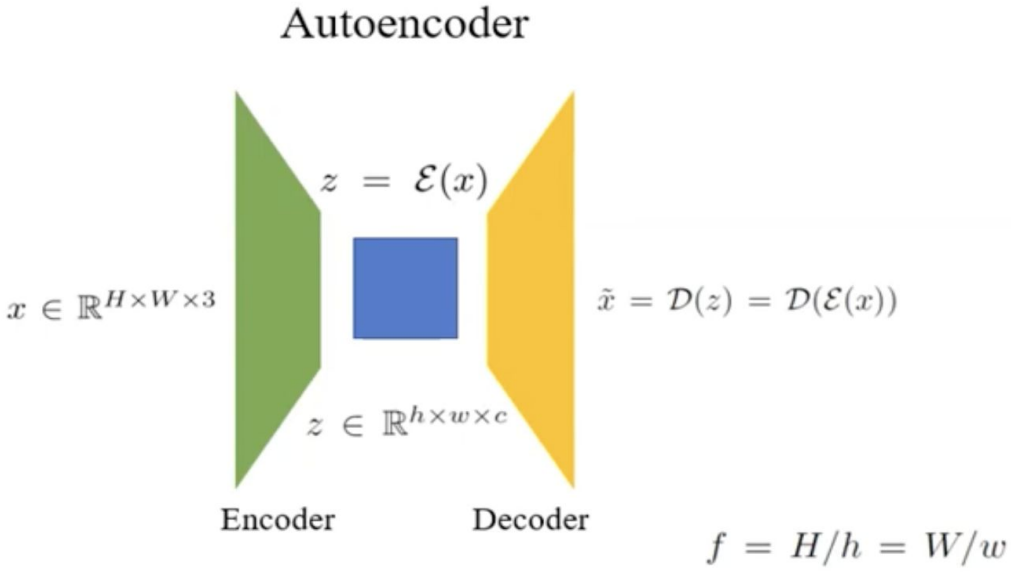
Latent Diffusion Models

- Уйдя от обучения входной картинки высокого разрешения, мы снижаем требование высокой производительности, поскольку обучаем модель в низкоразмерном пространстве.
- Все еще сохраняем качество генерации из-за архитектуры диффузионной модели

Perceptual Image Compression

- Perceptual Loss
- Patch-based adversarial objective

Этот подход заставляют модель делать изображения реалистичными и избегать блюра



Latent Diffusion Models

$$L_{DM} = \mathbb{E}_{x,\epsilon\sim\mathcal{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta(x_t, t)\|_2^2\right],$$

$$L_{LDM} := \mathbb{E}_{\mathcal{E}(x),\epsilon\sim\mathcal{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta(z_t, t)\|_2^2\right].$$

- Фокус на важных семантических фрагментах данных
- Обучение в пространстве меньшей размерности, гораздо более эффективном с вычислительной точки зрения.

Conditioning Mechanisms

$\epsilon_\theta(z_t, t, y)$ cross-attention mechanism

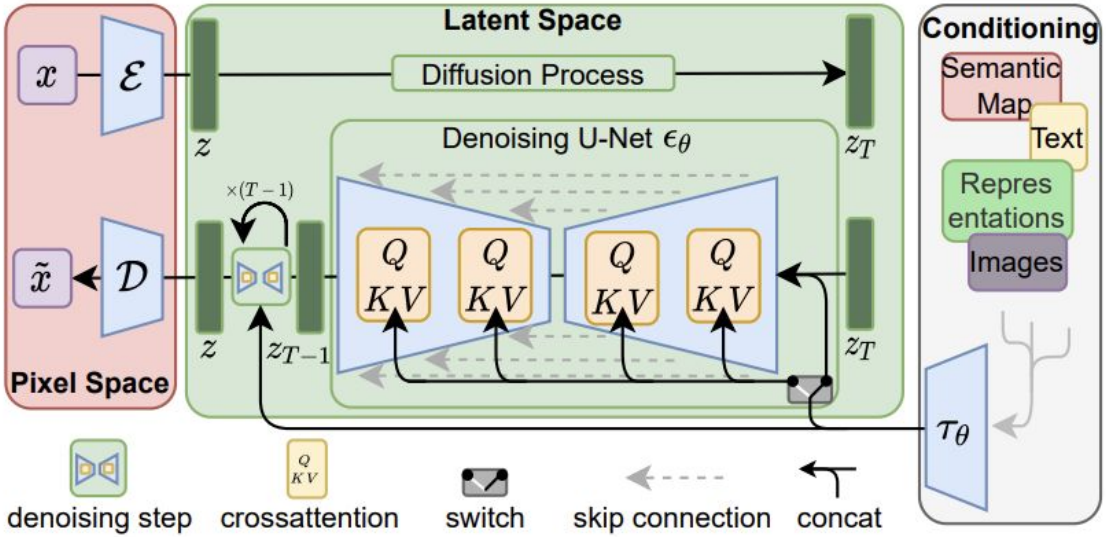
Закодируем наше условие с помощью энкодера - получим r(y).

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right) \cdot V, \text{ with}$$

$$Q = W_Q^{(i)} \cdot \varphi_i(z_t),\ K = W_K^{(i)} \cdot \tau_\theta(y),\ V = W_V^{(i)} \cdot \tau_\theta(y).$$

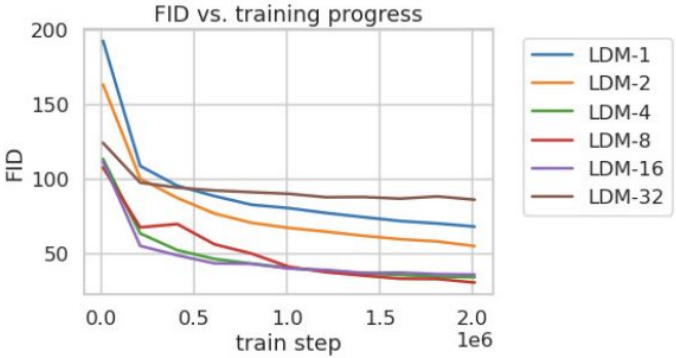
Here, ϕi denotes a (flattened) intermediate representation of the UNet

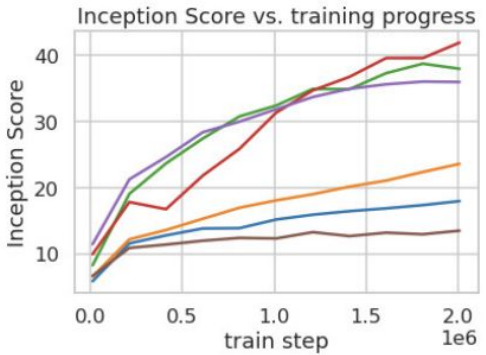
Latent Diffusion Models

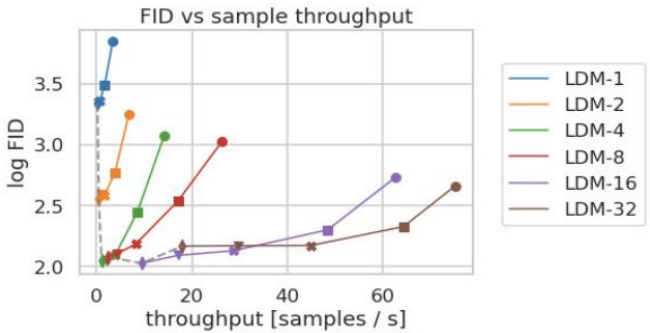


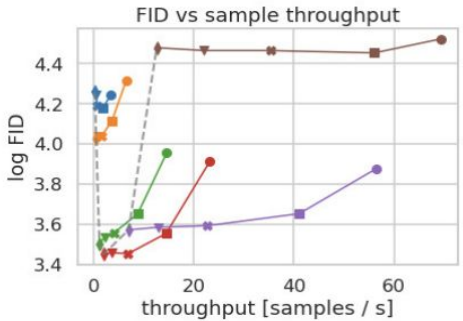
$$L_{LDM} := \mathbb{E}_{\mathcal{E}(x),y,\epsilon\sim\mathcal{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta(z_t, t, \tau_\theta(y))\|_2^2\right]$$

Тесты


FID vs. training progress
FID
200
150
100
50
0.0
0.5
1.0
1.5
2.0
1e6
train step
LDM-1
LDM-2
LDM-4
LDM-8
LDM-16
LDM-32


Inception Score vs. training progress
Inception Score
40
30
20
10
0.0
0.5
1.0
1.5
2.0
1e6
train step


FID vs sample throughput
log FID
3.5
3.0
2.5
2.0
0
20
40
60
throughput [samples / s]
LDM-1
LDM-2
LDM-4
LDM-8
LDM-16
LDM-32

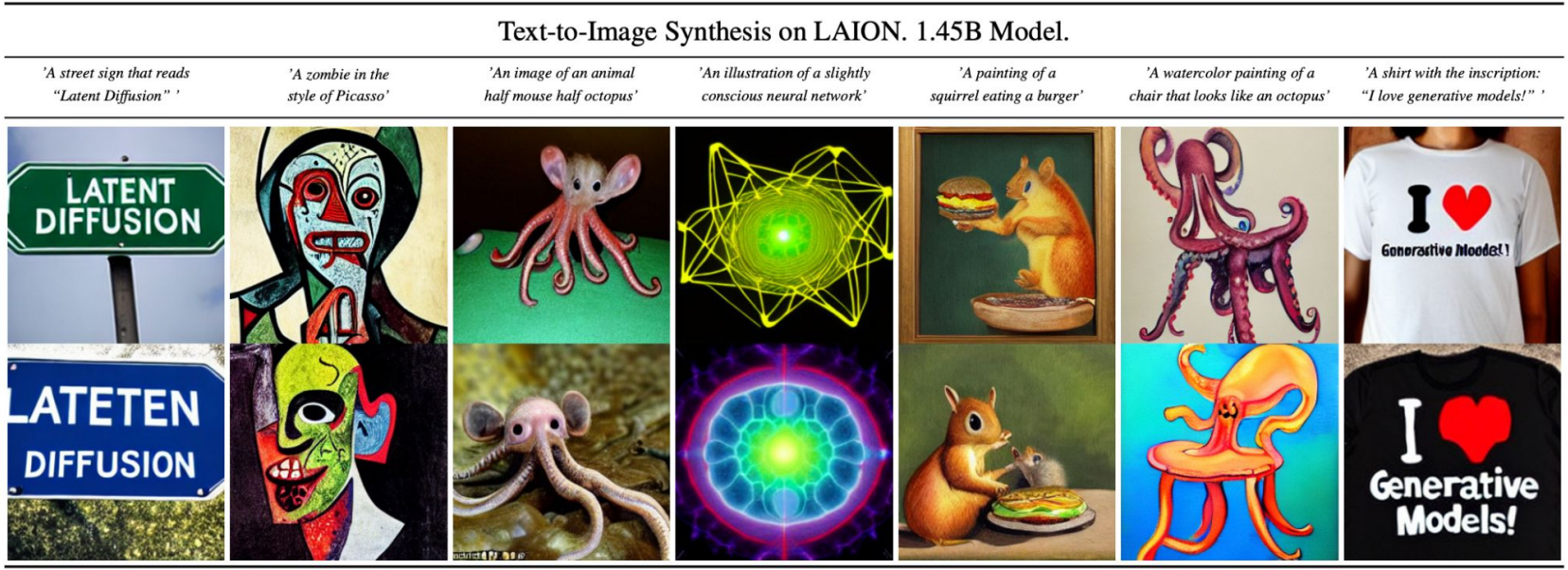

FID vs sample throughput
log FID
4.4
4.2
4.0
3.8
3.6
3.4
0
20
40
60
throughput [samples / s]

Txt2img

Text-Conditional Image Synthesis				
Method	FID ↓	IS↑	Nparams	
CogView† [17]	27.10	18.20	4B	self-ranking, rejection rate 0.017
LAFITE† [109]	26.94	26.02	75M	
GLIDE* [59]	12.24	-	6B	277 DDIM steps, c.f.g. [32] $s = 3$
Make-A-Scene* [26]	**11.84**	-	4B	c.f.g for AR models [98] $s = 5$
LDM-KL-8	23.31	$20.03_{\pm 0.33}$	1.45B	250 DDIM steps
*LDM-KL-8-G**	12.63	$\mathbf{30.29}_{\pm 0.42}$	1.45B	250 DDIM steps, c.f.g. [32] $s = 1.5$

Получаем кодирование условия при помощи BERT

Text-to-Image Synthesis on LAION. 1.45B Model.



В итоге

- Все еще медленнее ганов
- Но намного лучше для модифицирования чем ганы