# Stable Diffusion finetune

# Links

https://dreambooth.github.io/

https://huggingface.co/blog/dreambooth

https://github.com/huggingface/diffusers/tree/main/examples/dreambooth

https://arxiv.org/pdf/2208.01618.pdf

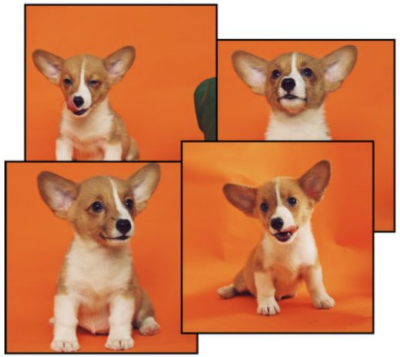
https://arxiv.org/abs/2106.09685

https://stable-diffusion-art.com/lora/

# Методы дообучения модели

- Dreambooth
- Textual Inversion
- LoRA
- Hypernetworks

# DreamBooth



Input images

in the Acropolis

swimming

sleeping

in a doghouse

in a bucket

getting a haircut

Input images

A [V] teapot floating in milk

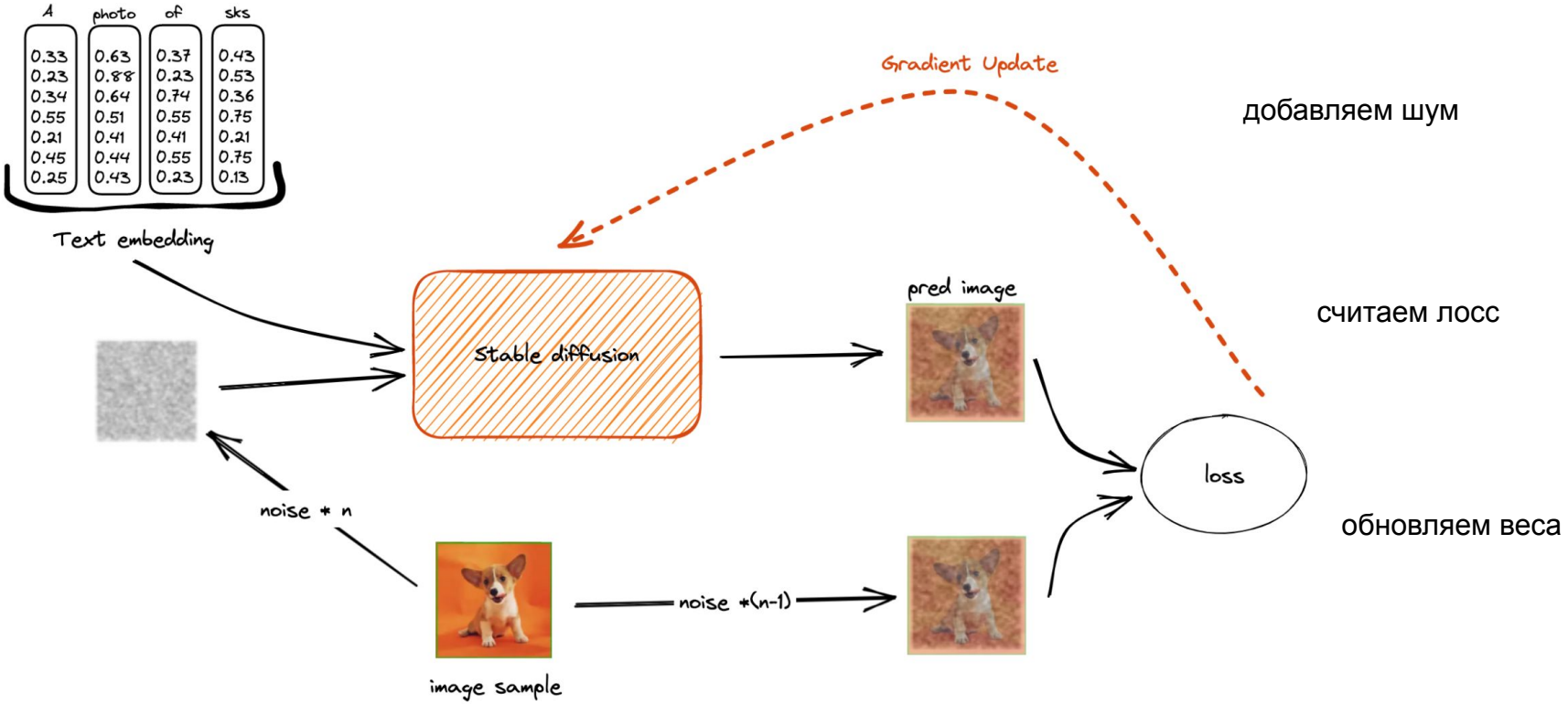
A transparent [V] teapot with milk inside

A [V] teapot pouring tea

A [V] teapot floating in the sea

# DreamBooth

Связываем наш объект с конкретным токеном



# DreamBooth

$$\mathbf{x}_{\text{gen}} = \hat{\mathbf{x}}_\theta(\boldsymbol{\epsilon}, \mathbf{c})$$

ε - шум; c - conditional вектор; $x_\theta$ - модель

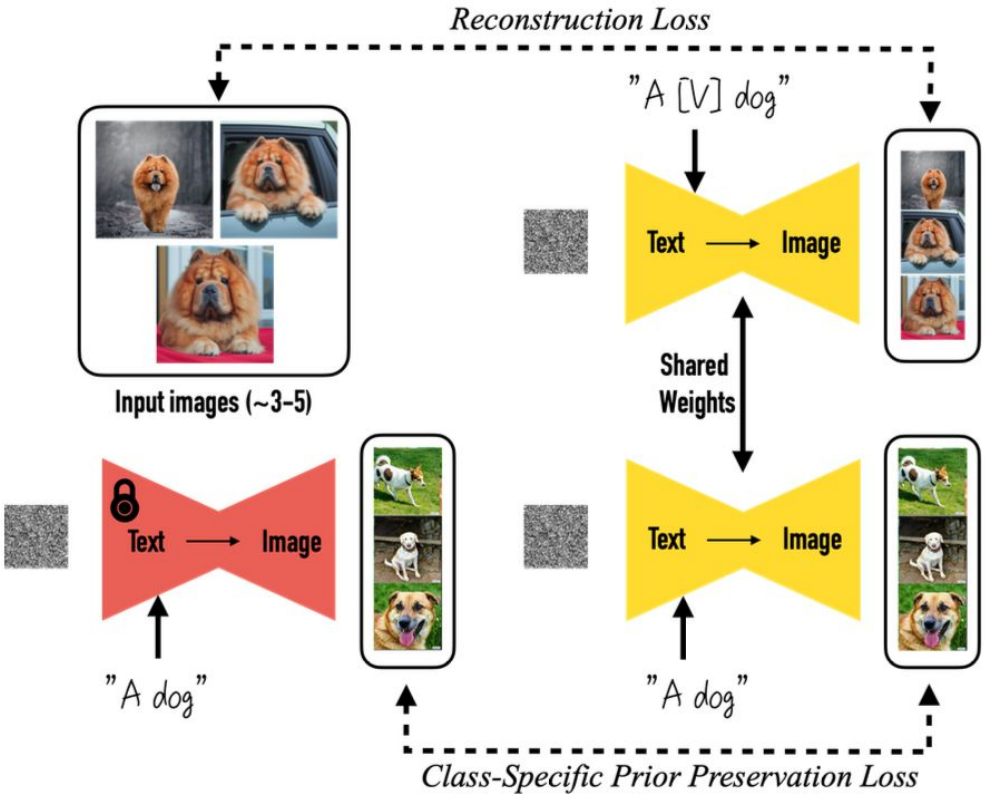
Считаем MSE loss для шума из разницы изображений

функция потерь

$$\mathbb{E}_{\mathbf{x},\mathbf{c},\boldsymbol{\epsilon},t}\left[w_t \|\hat{\mathbf{x}}_\theta(\alpha_t \mathbf{x} + \sigma_t \boldsymbol{\epsilon}, \mathbf{c}) - \mathbf{x}\|_2^2\right]$$

функция потерь с регуляризацией

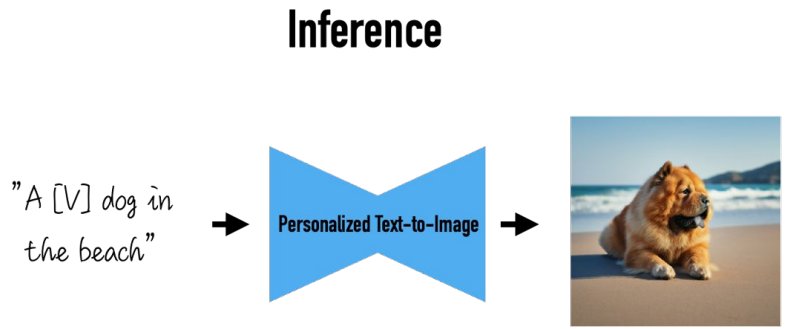
$$\mathbb{E}_{\mathbf{x},\mathbf{c},\boldsymbol{\epsilon},\boldsymbol{\epsilon}',t}[w_t \|\hat{\mathbf{x}}_\theta(\alpha_t \mathbf{x} + \sigma_t \boldsymbol{\epsilon}, \mathbf{c}) - \mathbf{x}\|_2^2 + \lambda w_{t'} \|\hat{\mathbf{x}}_\theta(\alpha_{t'} \mathbf{x}_{\text{pr}} + \sigma_{t'} \boldsymbol{\epsilon}', \mathbf{c}_{\text{pr}}) - \mathbf{x}_{\text{pr}}\|_2^2],$$



# DreamBooth

**Цель:** добавить в модель конкретный объект при этом не потерять накопленные знания

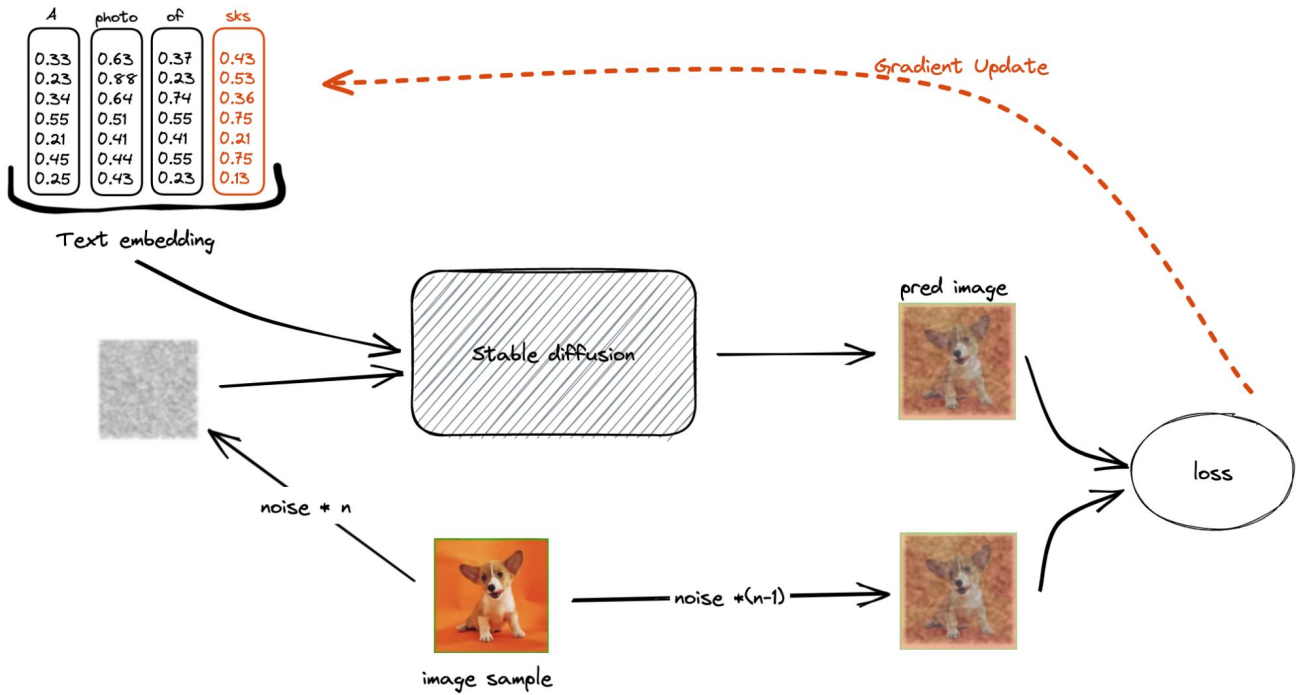
1. расширяем словарь и соотносим объект с непопулярным токеном[sks] чтобы не было *language drift*
2. Плохая идея выбрать рандомный набор символов “xxy5syt00” токенайзер может их разбить на разные токены
3. внедряем пару  “a [identifier] [class noun]”  (e.g. cat, dog, watch, etc.)
4. На вход достаточно 5 картинок на разном фоне и с разных ракурсов (максимально разнообразные)
5. тренируем модель целиком, приходится  сохранять веса целиком



# Textual inversion

# Textual inversion

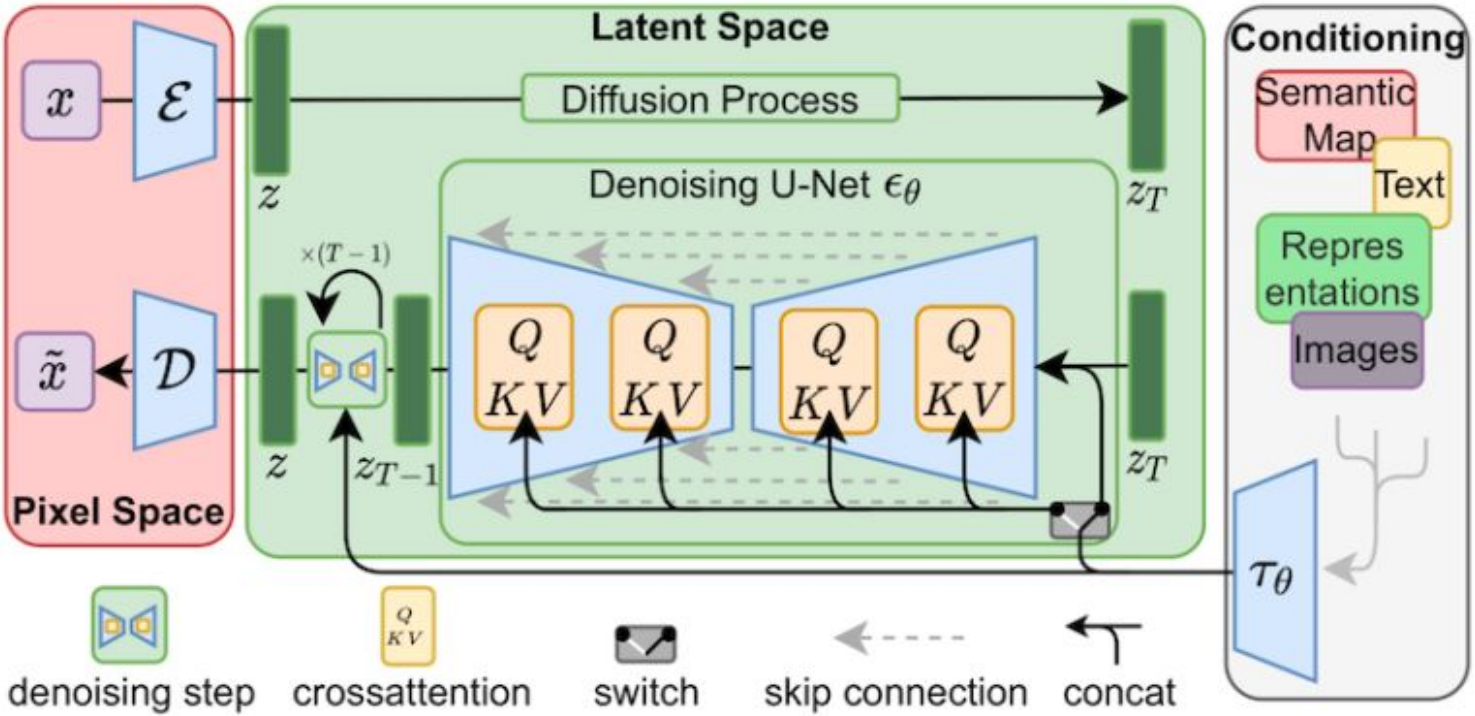
вместо целой модели “обучаем” наш текстовый embedding



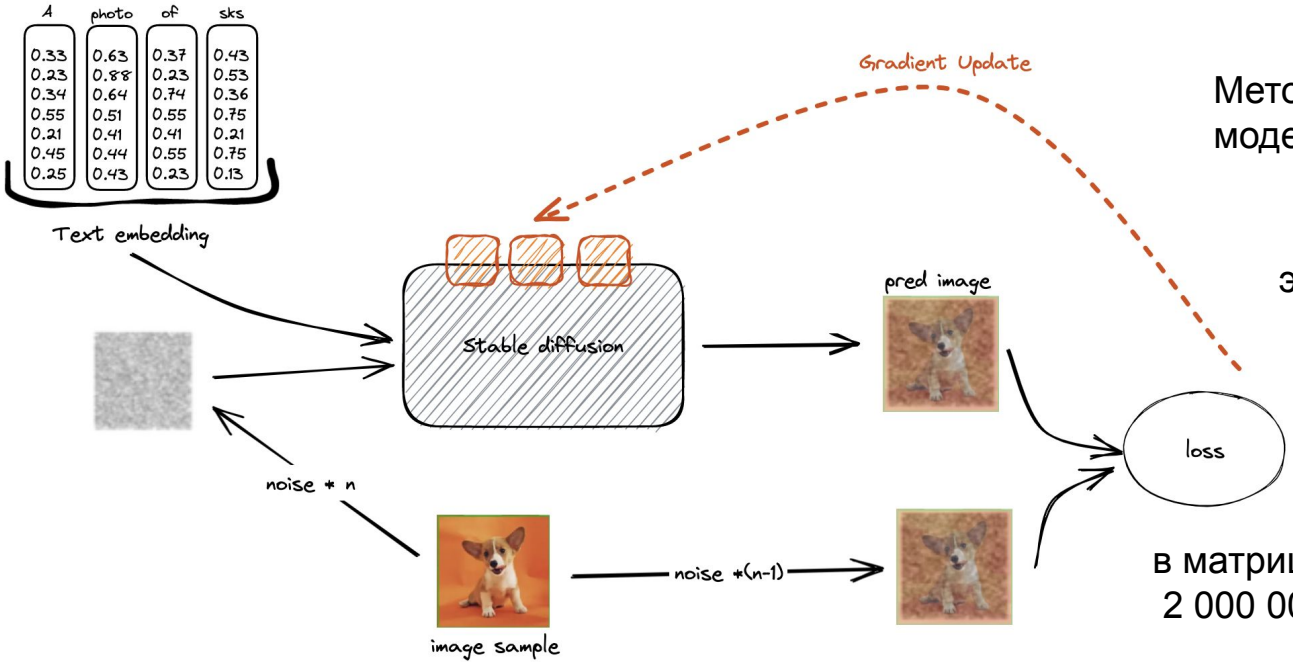

Шаблоны

- “a photo of a $S_*$.”,
- “a rendering of a $S_*$.”,
- “a cropped photo of the $S_*$.”,
- “the photo of a $S_*$.”,
- “a photo of a clean $S_*$.”,
- “a photo of a dirty $S_*$.”,
- “a dark photo of the $S_*$.”,
- “a photo of my $S_*$.”,

# Lora Low-Rank Adaptation



# Lora Low-Rank Adaptation




Метод для больших языковых моделей

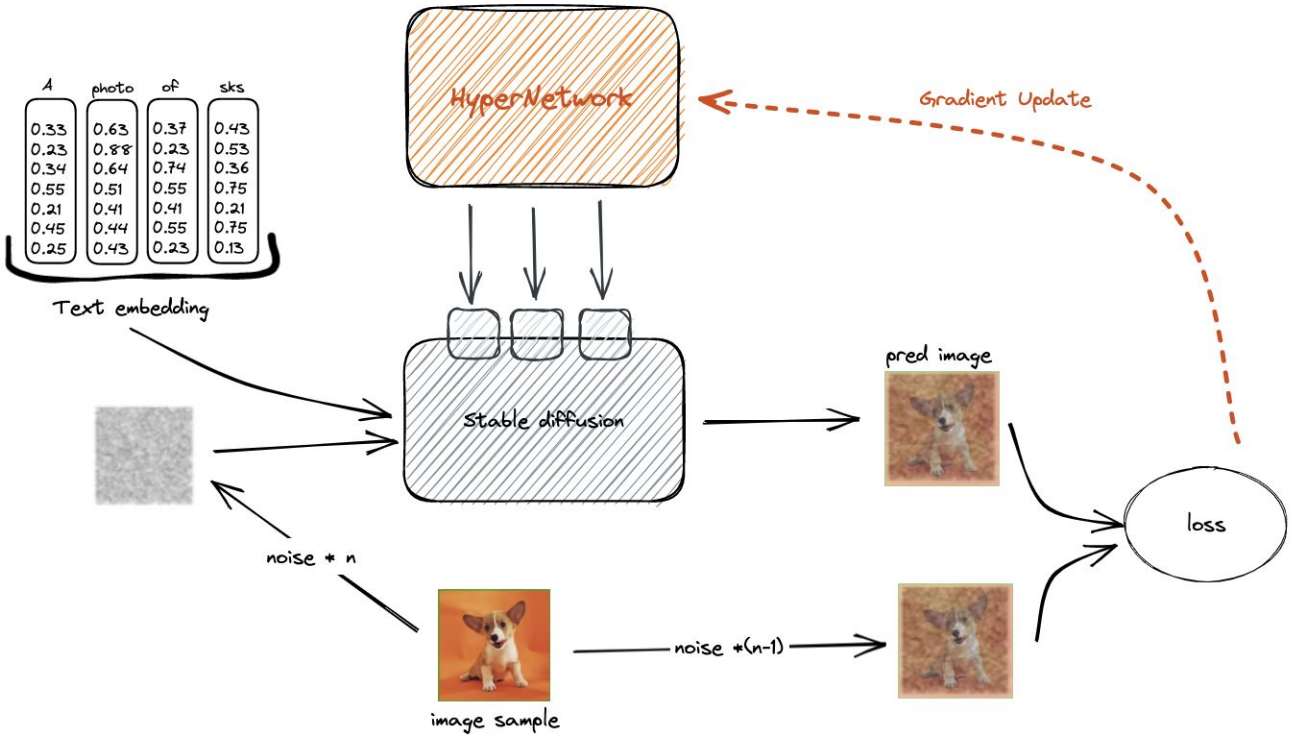
экономит ресурсы и место

в матрице размером 1000*2000
2 000 000 параметров

$$\mathbf{C} = \mathbf{A} \cdot \mathbf{B} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

в матрицах размером 1000*2 и 2000*2
6 000 параметров

# HeperNetwork




That's all Folks!