

# Создание персонализированных генераций изображений

Степанов Илья Казистова Кристина

Московский физико-технический институт

Курс: Научный трек иннпрака ФПМИ

Эксперт: Филатов Андрей Викторович

2024

# Цель исследования

## Задача

Сгенерировать изображения человека в различных вариациях в высоком разрешении.

## Цель

Повысить качество изображений, генерируемых с помощью диффузионных моделей.

## Проблема

Низкая точность генерации, неполное соответствие сгенерированных изображений текстовым описаниям, недостаточно высокое качество получаемого изображения.

## Постановка задачи

Определим датасет как  $\mathcal{D} = \{(x_i, \tau_i) : i = 1, \dots, n\}$ ,  $x_i$  – изображение,  $\tau_i$  – соответствующий текстовый промпт.

Рассматривается модель  $\epsilon_\theta$  из класса диффузионных моделей. На этапе обучения на каждом шаге из  $\mathcal{D}$  удаляется изображение  $x_j, j \sim \mathcal{U}\{1, \dots, n\}$ .

## Постановка задачи

Определим функцию потерь:

$$\mathcal{L}(\epsilon, \epsilon_\theta) = \mathbb{E}_{\epsilon \sim N(0, I), \mathbf{c}_\tau, \mathbf{c}_i, t, \mathbf{c}_t^j} \|\epsilon - \epsilon_\theta(\mathbf{c}_\tau, \mathbf{c}_i, t, \mathbf{c}_t^j)\|^2, \quad (1)$$

где  $\mathbf{c}_\tau = \Gamma_\tau(\tau_j)$  — текстовые признаки удаленного изображения;  $\Gamma_\tau$  — текстовый энкодер; текстовый промпт  $\tau_j$ ;  
 $\mathbf{c}_i = G(\Gamma_i(x_1), \dots, \Gamma_i(x_{j-1}), \Gamma_i(x_{j+1}), \dots, \Gamma_i(x_n))$  — признаки оставшихся изображений;  $G$  — агрегирующая функция;  $\Gamma_i$  — image-энкодер;  $\mathbf{c}^j = \Gamma_i(x_j)$  — признаки удаленного изображения;  
 $t \in [0, T]$  — временной шаг диффузионного процесса;  $\mathbf{c}_t^j = \alpha_t \mathbf{c}^j + \sigma_t \epsilon$  — зашумленные данные удаленного изображения на шаге  $t$ ;  $\alpha_t, \sigma_t$  — предопределенные функции от  $t$ , определяющие диффузионный процесс.

## Постановка задачи

Решается следующая оптимизационная задача:

$$\epsilon_{\theta}^{*} = \arg \min_{\epsilon_{\theta}} \mathcal{L}(\epsilon, \epsilon_{\theta}), \quad (2)$$

## Постановка задачи

Для определения качества модели введем метрики качества генерации Frechet Inception Distance (FID) и Inception Score (IS):

$$FID = \|\mu_p - \mu_q\|^2 + \text{Tr}(\Sigma_p + \Sigma_q - 2(\Sigma_p \Sigma_q)^{1/2}) \quad (3)$$

где  $\mu_p$  и  $\mu_q$  — средние значения признаков в реальных и сгенерированных изображениях соответственно,  $\Sigma_p$  и  $\Sigma_q$  — ковариационные матрицы для распределений признаков на реальных и сгенерированных изображениях соответственно.

$$IS(x) = \exp(\mathbb{E}_x [D_{KL}(p(y|x) || p(y))]) \quad (4)$$

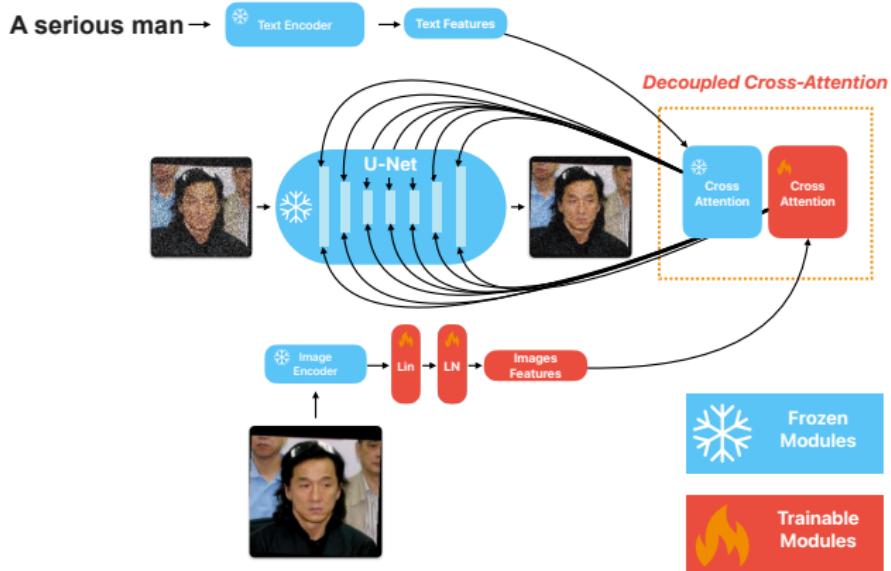
где  $D_{KL}$  - дивергенция Кульбака-Лейблера для двух распределений;  $p(y|x)$  - вероятность класса  $y$  для изображения  $x$ ;  $p(y)$  - равномерное распределение на множестве классов.

# DreamBooth



- ▶ Принимает на вход несколько изображений одного объекта вместе с соответствующим названием класса;
- ▶ Возвращает специальный токен, идентифицирующий объект;
- ▶ Токен встраивается в текстовую подсказку, по которой генерируется желаемое изображение.

# IP-Adapter



- ▶ Энкодер для извлечения признаков изображения;
- ▶ Адаптированные модули с механизмом перекрестного внимания.

## Decoupled Cross-Attention

Выход слоя Cross-Attention для текстовых признаков  $\mathbf{c}_t$ :

$$\mathbf{Z}' = \text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d}}\right)\mathbf{V}, \quad (5)$$

где  $\mathbf{Z}$  — признаки запроса,  $\mathbf{Q} = \mathbf{Z}\mathbf{W}_q$ ,  $\mathbf{K} = \mathbf{c}_t\mathbf{W}_k$ ,  $\mathbf{V} = \mathbf{c}_t\mathbf{W}_v$  — матрицы запросов, ключей и значений механизма внимания для текстовых признаков соответственно, а  $\mathbf{W}_q$ ,  $\mathbf{W}_k$ ,  $\mathbf{W}_v$  — соответствующие матрицы весов.

Выход слоя Cross-Attention для признаков изображения  $\mathbf{c}_i$ :

$$\mathbf{Z}'' = \text{Attention}(\mathbf{Q}, \mathbf{K}', \mathbf{V}') = \text{Softmax}\left(\frac{\mathbf{Q}(\mathbf{K}')^T}{\sqrt{d}}\right)\mathbf{V}', \quad (6)$$

где  $\mathbf{Q} = \mathbf{Z}\mathbf{W}_q$ ,  $\mathbf{K}' = \mathbf{c}_i\mathbf{W}'_k$ ,  $\mathbf{V}' = \mathbf{c}_i\mathbf{W}'_v$  — матрицы запросов, ключей и значений механизма внимания для признаков изображения соответственно, а  $\mathbf{W}'_k$ ,  $\mathbf{W}'_v$  — соответствующие матрицы весов.

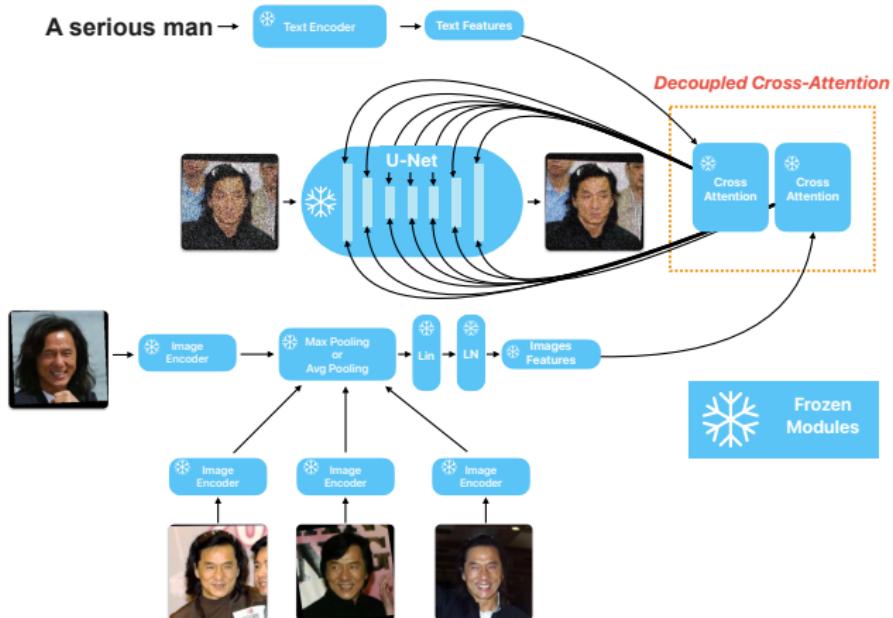
## Decoupled Cross-Attention

Выход слоя Decoupled Cross-Attention:

$$\mathbf{Z}^{new} = \text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) + \lambda \cdot \text{Attention}(\mathbf{Q}, \mathbf{K}', \mathbf{V}'), \quad (7)$$

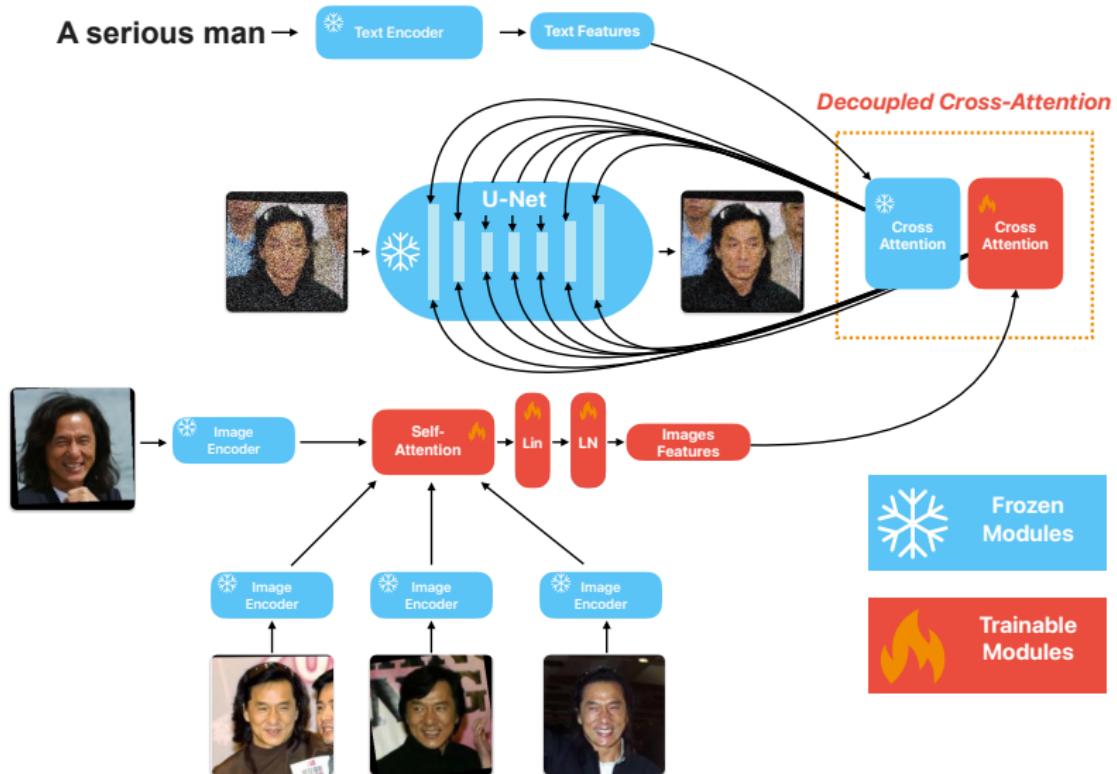
где  $\lambda$  — весовой коэффициент.

# IP-Adapter + агрегирующая функция



- ▶ Принимает на вход несколько изображений;
- ▶ К эмбеддингам входных изображений применяется Max Pooling или Avg Pooling.

# IP-Adapter + Self-Attention



Процедура обучения:

- ▶ Входные данные: 10 изображений одного объекта с текстовыми промптами;
- ▶ Случайно выбранное изображение удаляется из рассмотрения;
- ▶ Модель учится предсказывать выброшенное изображение по его текстовому промпту и эмбеддингам оставшихся изображений.

## Датасет

LFW Deep Funneled — набор изображений лиц людей вместе с их именами. В этом наборе данных 100 людям соответствует не меньше десяти разных фотографий.



## Результаты экспериментов

Метод	IS ↑	FID ↓
IP-Adapter	15.37	8.92
DreamBooth	17.64	9.61
IP-Adapter + Max Pooling	14.12	10.10
IP-Adapter + Avg Pooling	13.56	11.82
IP-Adapter + Self-Attention	<b>18.72</b>	<b>7.56</b>

# Результаты генерации

DreamBooth



IP-Adapter



IP-Adapter +  
Max Pooling



IP-Adapter +  
Avg Pooling



IP-Adapter +  
Self-Attention



## Заключение

- ▶ Модификация метода IP-Adapter с использованием Self-Attention показала наилучший результат по метрикам качества IS и FID;
- ▶ Далее можно модифицировать наши методы посредством использования LoRa, FaceNet и др.