ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»
Факультет Санкт-Петербургская школа физико-математических и
компьютерных наук

Воробьев Тихон Михайлович

ДООБУЧЕНИЕ ДИФФУЗИОННОЙ МОДЕЛИ ДЛЯ ГЕНЕРАЦИИ ПЕРСОНАЛИЗИРОВАННЫХ ПОРТРЕТНЫХ ИЗОБРАЖЕНИЙ

Выпускная квалификационная работа
по направлению подготовки 01.04.02 Прикладная математика и информатика
образовательная программа «Машинное обучение и анализ данных»

Рецензент
магистр, PicsArt, Inc.,
ML Scientist

_____
А.А. Котов

Научный руководитель
к. т. н., старший преподаватель,
департамент информатики

_____
А.А. Шпильман

Санкт Петербург - 2023

The Government of the Russian Federation
Federal State Autonomous Institution for Higher Education
National Research University Higher School of Economics
St. Petersburg Branch
St. Petersburg School of Physics, Mathematics and Computer Science

Vorobev Tikhon

Fine tuning the diffusion model to generate personalized portrait images

Master dissertation
Area of studies 01.04.02 «Applied Mathematics and Informatics»
Master Program «Machine Learning and Data Analysis»

Reviewer	Research Supervisor
Master, PicsArt, Inc.,	PhD, head of AI educational programs
ML Scientist	department of informatics
_____	_____
Artem Kotov	Aleksei Shpilman

Saint Petersburg – 2023

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

В течение многих веков люди заказывали свои портреты у художников, а теперь делают фотосессии. Для создания таких картин или фотографий необходимы специальные навыки и много времени, однако их достаточно просто описать при помощи текста. Поэтому инструмент, способный генерировать реалистичные изображения по текстовым описаниям, дает возможность легко и быстро создавать разнообразный контент, тем самым уменьшая как финансовые, так и временные затраты.

Генерация персонализированных портретных изображений в различных сценах, позах, видах и условиях освещения, которые не появляются на исходных изображениях, представляет собой сложную задачу, которая ранее решалась многими способами. Однако большинство решений не позволяют достичь разнообразной генерации или сильно изменяют идентичность пользователя.

Диффузионные модели преобразования текста в изображение произвели революцию в создании высококачественных изображений. Такие модели могут генерировать самые разные объекты, стили и сцены. Но, если пользователь захочет сгенерировать свой портрет в конкретном стиле или окружении, у него не получится достаточно точно воспроизвести собственную идентичность, как бы он подробно себя не описывал. Это связано с тем, что изображения пользователя не находятся в обучающем наборе данных. Повторное обучение модели с расширенным набором данных под каждого пользователя является непомерно дорогим и долгим процессом. Данную проблему прекрасно решают недавно появившиеся методы персонализации диффузионных моделей, такие как Text Inversion [9], DreamBooth [43] или Custom Diffusion [25]. Однако все они недостаточно вычислительно эффективные, а также требуют для получения результатов много промежуточных шагов, что не подходит для внедрения в рабочий продукт.

Целью данной работы является разработка эффективного алгоритма дообучения диффузионной модели для получения персонализированных портретных изображений.

Задачи исследования:

1. Подготовка персонализированных данных для дообучения.
2. Дообучение диффузионной модели.
3. Реализация метода контролируемой генерации.
4. Создание целостного прототипа для получения персонализированных портретных изображений.

1. Обзор предметной области

Генерация персонализированных портретных изображений в различных контекстах представляет собой сложную задачу, которая ранее решалась многими способами.

Для генерации объекта в разных позах можно использовать современные методы 3D-реконструкции, основанные на подходе Nerf [31]. Однако для таких подходов необходимо большое количество фотографий объекта с разных точек обзора со согласованной позой, а также создание трехмерных сцен для смены контекста изображений.

Данную задачу можно решать при помощи генеративных состязательных моделей GAN [12]. Например, метод композиции изображений Gp-gan [52] направлен на клонирование заданного объекта на новый фон таким образом, чтобы объект плавно сливался со сценой. A DRB-GAN [53] решает задачу генерации изображения, которое объединяет изначальную структуру со стилем, заданным вторым изображением. Тем не менее, этими методами сложно управлять результатами генерации, а также они требуют для результата два изображения, что сужает их вариативность. Эти проблемы помогают решать модели, основанные на подходах Pix2Pix [18] или CycleGAN [57], которые переводят пользовательские изображения в выученный стиль. Ограничение этих подходов в том, что под каждый стиль необходимо учить новую модель. Так как на пользовательских фотографиях изображены люди, то данную задачу можно решать манипулированием скрытого представления StyleGAN [22], полученного в результате инверсии [1, 50]. Мультимодальные походы, такие как StyleClip [36] или StyleGAN-NADA [10], позволяют производить эти манипуляции при помощи текста, используя модель CLIP [38], что дает более гибкий инструмент управления. Однако при использовании таких методов часто теряется идентичность пользователя, а также возникают сложности с изменением окружения. Таким образом генеративные состязательные модели, несмотря

на свою эффективность и мультимодальность, однодоменные. Совсем недавно стали появляться мультидоменные модели, например GigaGAN [20]. К сожалению, у них нет открытого доступа, а обучать такую модель с нуля требует огромных вычислительных мощностей.

На текущий момент диффузионные модели [19, 40, 44, 45, 49] произвели революцию в создании высококачественных изображений. Такие модели могут генерировать самые разные объекты, стили и сцены. Но если пользователь захочет сгенерировать самого себя в конкретном стиле или окружении, у него не получится достаточно точно воспроизвести собственную идентичность, как бы он подробно себя не описывал. Это связано с тем, что изображения пользователя не находятся в обучающем наборе данных. Повторное обучение модели с расширенным набором данных под каждого пользователя является непомерно дорогим и долгим процессом. В DALL-E 2 [44], Kandinsky 2.1 [19] или Stable Diffusion Variations [26] есть возможность при генерации смешивать латентные представления изображений и текстовых описаний, но это приводит к потери идентичности пользователя. Наподобие с генеративными состязательными моделями можно сделать инверсию исходного изображения. Например, за счет простого добавления шума к исходному изображению, а затем удаления шума через модель. Однако этот процесс имеет тенденцию значительно изменять содержание изображения, что приводит к потери идентичности пользователя. Для решения этой проблемы можно подключить легко переносимую сеть ControlNet [55] или T2I-Adapter [33], которая при генерации дополнительно учитывает границы, скелет или карту глубины. Но чем больше добавляется контроля, тем больше сохраняется идентичность пользователя и тем менее выразительные результаты. Также появляются подходы, которые позволяют изменять исходное изображение при помощи коротких инструкций InstuctPix2Pix [4], но они плохо сохраняют идентичность. Процесс инверсии можно осуществить за счет детерминированности, а значит обратимости,

таких методов расшумления, как DDIM [48] или Euler [21]. Несмотря на то, что это улучшает сохранение исходной информации, такие подходы, как отмечают в Null-text Inversion [32], плохо работают вместе с подходом Classifier-Free Guidance [14], который используется во всех передовых подходах генерации изображений. Сам подход Null-text Inversion [32], который улучшает подход инверсии DDIM [48], дает прекрасные результаты редактирования, однако он не позволяет изменять позу пользователя на изображении.

Большой прорыв в персонализированной генерации достигли подходы LoRA [16], HyperNetwork [35], Imagic [23], Text Inversion [9], DreamBooth [43] и Custom Diffusion [25]. Все эти подходы направлены на дообучение определенных частей предварительно обученной диффузионной модели на небольшом наборе из пользовательских изображений. В Imagic [23] диффузионная модель дообучается под каждую новую текстовую подсказку, что ограничивает скорость генерации объекта в разных контекстах. LoRA [16] обучает добавочную низкоранговую матрицу к матрицам перекрестного внимания [41], а HyperNetwork [28] обучает дополнительные линейные слои после проекций матриц перекрестного внимания [41]. Эти методы хорошо применимы для обучения генерации нового стиля, а не объекта, так как не предоставляют возможность управления генерацией текстовым описанием. Text Inversion [7] решает эту проблему добавляя новое псевдослово в токенизатор, которое будет соответствовать новой концепции, и обучая соответствующий вектор в текстовом энкодере. Custom Diffusion [25] или DreamBooth [43] расширяют подход Text Inversion [9] и дополнительно с вектором дообучают матрицы перекрестного внимания [41] или всю модель соответственно. Однако все эти методы недостаточно вычислительно эффективны, а также требуют для получения результатов много промежуточных шагов, что не подходит для внедрения их в рабочий продукт.

Для реализации поставленной цели подходы DreamBooth [43] и Text Inversion [9] были взяты за бейзлайн.

1.1. Диффузионные модели

Принцип работы диффузионных моделей был описан в работе 2015 года [47] и улучшен в работе DDPM [15]. Взяв элемент из распределения данных $x_0 \sim q(x_0)$ запускается процесс прямой диффузии, который создает последовательность элементов $x_1, ..., x_T$ за счет прогрессивного добавления Гауссовского шума:

$$q(x_t|x_{t-1}) = N(x_t; \sqrt{\alpha_t}, (1 - \alpha_t)I) \quad q(x_{1:T}|x_0) = \prod_{t=1}^{T} q(x_t|x_{t-1}) \quad \alpha_t \in (0, 1)$$

Приятным свойством выше описанного процесса является то, что можно выразить x_t через x_0:

$$q(x_t|x_0) = N(x_t; \sqrt{\hat{\alpha}_t}x_0, (1 - \hat{\alpha}_t)I) \quad \hat{\alpha}_t = \prod_{i=1}^{t} \alpha_i$$

Получается, что при больших значения T, x_T можно аппроксимировать $N(0, I)$. Диффузионные модели обучаются аппроксимировать апостериорное распределение $q(x_{t-1}|x_t)$:

$$p_\Theta(x_{t-1}|x_t) = N(x_{t-1}; \mu_\Theta(x_t, t), \Sigma_\Theta(x_t, t))$$

$$p_\Theta(x_{0:T}) = p_\Theta(x_T)\prod_{t=1}^{T} p_\Theta(x_{t-1}|x_t) \quad p_\Theta(x_T) = N(x_T; 0, I)$$

В работе DDPM [15], выводят нижнюю вариационную границу для $logp_\Theta(x_0)$. Для ее оптимизации генерируются $x_t \sim q(x_t|x_0)$ путем добавления Гауссовского шума ϵ к x_0, а затем модель ϵ_Θ обучается предсказывать добавленный шум используя следующую функцию потерь:

$$L = E_{x,\epsilon\sim N(0,1),t\in[0,T]}[\|\epsilon - \epsilon_\Theta(x_t, t)\|_2^2]$$

Таким образом, диффузионные модели позволяют генерировать данные $x_0 \sim p_\Theta(x_0)$ взяв в качестве начальной точки $x_T \sim N(0, I)$ и последовательно предсказывая шум, который необходимо удалить (Рис. 1.)

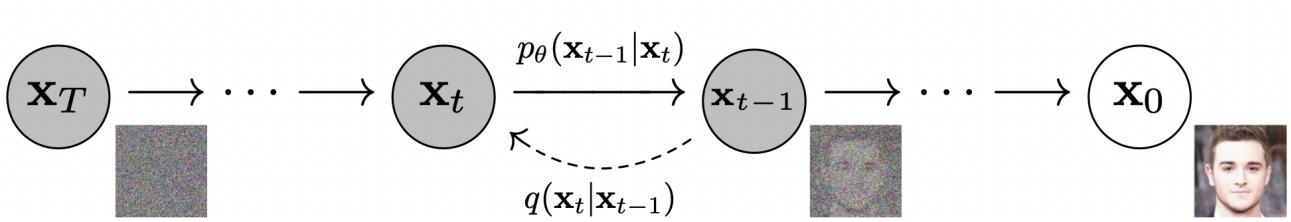


Рис 1. Процесс зашумления и расшумления изображений [15]

Впоследствии стали появляться работы, которые предлагают методы управления генерации за счет дополнительного условия c для предсказания добавленного шума, то есть функция потерь приобретает вид:

$$L = E_{x,\epsilon\sim N(0,1),t\in[0,T],c}[\|\epsilon - \epsilon_\Theta(x_t, t, c)\|_2^2]$$

Так в работе Classifier Guidance [8] авторы продемонстрировали превосходство диффузионных моделей над генеративными состязательными сетями, добавляя градиенты дополнительного обученного классификатора $f_\phi(c|x_t, t)$ для генерации объектов определенного класса:

$$\hat{\epsilon}_\Theta(x_t, t, c) = \epsilon_\Theta(x_t, t) - w\sqrt{1 - \hat{\alpha}_t}\nabla_{x_t} log f_\phi(c|x_t)$$

В работе Classifier-Free Guidance [14] устраняют необходимость обучать дополнительный классификатор. Они предлагают в качестве модели с условием взять $\epsilon_\Theta(x_t, t, c)$, а для обучения модели безусловной генерации с фиксированной вероятностью подавать пустое условие, то есть $\epsilon_\Theta(x_t, t) = \epsilon_\Theta(x_t, t, c = \emptyset)$. Таким образом, объект определенного класса генерируется по следующей формуле:

$$\hat{\epsilon}_\Theta(x_t, t, c) = \epsilon_\Theta(x_t, t, c = \emptyset) + w(\epsilon_\Theta(x_t, t, c) - \epsilon_\Theta(x_t, t, c = \emptyset))$$

В работе CLIDE [34] авторы расширяют подход Classifier-Free Guidance [14], предлагая технику позволяющую в качестве условия брать текст, а не просто класс. На этой работе основаны все последующие диффузионные модели преобразования текста в изображение.

1.2. Выбор модели

На данный момент самыми передовыми диффузионными моделями являются DALL-E 2 [44], Imagen [45], а также Stable Diffusion [49] и Kandinsky 2.1 [19] на основе LDM [40]. Большим плюсом модели LDM [49] является то, что она имеет открытый доступ по сравнению с DALL-E 2 [44], Imagen [45] и диффузионный процесс происходит в латентном пространстве, а не пиксельном, что позволяет эффективно работать с изображениями большого разрешения. Модель Kandinsky 2.1 [19] имеет больший размер, чем Stable Diffusion [49], так как поддерживает мультиязыковые запросы, для данного решение это не является необходимостью, поэтому в качестве диффузионной модели была выбрана Stable Diffusion [49].

Архитектура модели LDM [40], представленная на Рис. 2, состоит из трех основных составляющих:

1. Вариационный автокодировщик VAE [24], который переводит пиксельное пространство изображений в латентное и наоборот.
2. Текстовый энкодер, в данном случае модель CLIP [38], который переводит текстовый запрос в текстовое латентное пространство.
3. Диффузионная U-net - подобная модель [42].

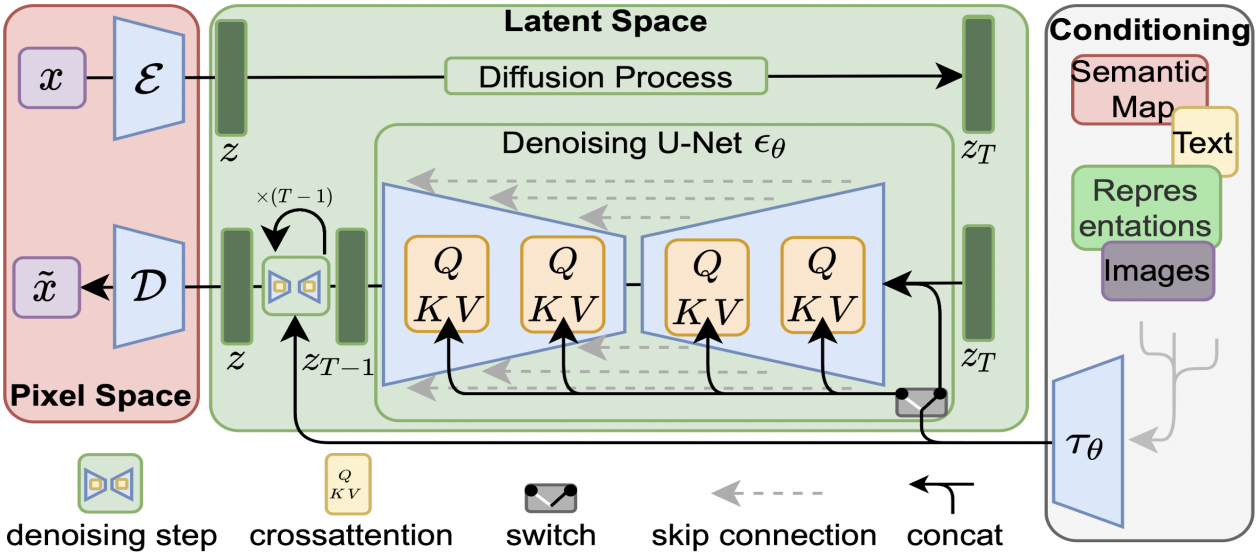


Рис. 2. Архитектура модели LDM [40]

Таким образом, VAE [24] и CLIP [38] переводят изображение и текстовое описание, соответственно, в латентное пространство. Затем диффузионная модель последовательно предсказывет шум, который необходимо удалить, дополнительно обуславливаясь при помощи механизма внимания [51] на номер шага и текстовые латенты. В конце VAE [24] переводит полученное латентное представление в изображение.

1.3. Метрики оценки качества

В Text Inversion [9] предлагают для оценки качества модели использовать такие метрики как Image Alignment и Text Alignment.

Image Alignment показывает насколько похожи сгенерированные изображения на пользовательские, и вычисляется как среднее значение попарных скалярных произведений между визуальными представлениями CLIP [38] сгенерированных изображений и исходных. Генерируются изображения по текстовому запросом «A photo of S_*», где S_* псевдослово соответствующее новой концепции.

Text Alignment показывает насколько модель сохранила возможность изменять новую концепцию с помощью текстовых описаний. Для этого генерируются изображения по созданному набору текстовых описаний, которые проверяют возможность изменения фона («A photo of S_* on the moon»), стиля («An oil painting of S_*») и композиции (»Elmo holding a S_*»). Затем вычисляется среднее значение скалярных произведений в пространстве CLIP [8] между сгенерированными изображениями и соответствующими текстовыми описаниями, в которых удалили S_*.

Также для оценки похожести сгенерированных изображений на пользовательские была рассмотрена метрика LPIPS [56].

2. Метод

2.1. Подготовка набора данных

Диффузионные модели обучаются на парном наборе данных из изображений и соответствующим им текстовых описаний.

2.1.1. Подготовка изображений

Модели Stable Diffusion [49] преимущественно дообучаются на изображениях размера 512 на 512 пикселей. Так как цель работы заключается в генерации персонализированных портретов, а Stable Diffusion [49] хорошо справляется с генерацией лиц только тогда, когда они находятся на большей части изображения, было принято решение настраивать модель на лицах пользователя. Таким образом пользователь загружает в мобильное приложение [11] свои фотографии, и при помощи существующих методов поиска лица [3, 7] из них вырезаются лица, которые преобразуются к размеру 512 на 512 пикселей, при необходимости дублируются пиксели с границы, чтобы было равное соотношение сторон. Изначально лицо может занимать маленькую область на изображении, поэтому сильное изменение размера может привести к потери изначальной информации. Для решения этой проблемы можно использовать не классические методы повышения разрешения изображения, а нейросетевые [51].

Во время загрузки фотографий в приложение пользователь может допустить ошибку и загрузить изображение без человека на нем, поэтому такие кадры необходимо предварительно отсеять. Также не учитываются фотографии, на которых больше одного человека. В дальнейшем данную проблему можно решить путем подсчета дескрипторов лиц на фотографиях и выбора мажорирующего лица при помощи специальных метрик [6, 56]. Для поиска человека на изображении можно использовать существующие методы поиска лица [3, 7] или детекции объектов основанных на подходах [29, 39], с проверкой есть ли человек среди обнаруженных объектов. Оба подхода

показали высокую полноту (recall), но первый имеет низкую точность (precision), то есть оба подхода хорошо справляются с задачей поиска человека на фотографии, но метод поиск лица склонен к ложному обнаружению. В итоге был выбрана последняя модификация подхода [39] по обнаружению объектов.

Пример обработки пользовательских изображений представлен на Рис. 3.

А Б

Рис 3. А - Исходные фотографии, Б - Отфильтрованные фотографии 512 на 512 пикселей

2.1.2. Подготовка текстовых описаний

В подходах по персонализации диффузионной модели [9, 25, 43] вводится новая концепция - псевдослово, будем его называть instance_token. То есть во всех этих методах модель дообучается на парном наборе данных из изображений новой концепции и описаний, содержащих instance_token.

Таким образом у модели появляется возможность генерировать выученную концепцию, если текстовое описания содержит instance_token.

В Dreambooth и Custom Diffusion [43, 25] текстовые подсказки имеют следующий вид: «context instance_token class_token», где instance_token соответствует новой концепции и равен случайному малочастотному токену из словаря текстового энкодера, а class_token является грубым описанием домена пользовательских фотографий, например, «person». В Dreambooth context равен строке «a photo of». В Custom Diffusion [25] предлагают в качестве context брать фразы «very small», «far away», «zoomed in», «close up» и, соответственно, context изменять размера объекта на изображении.

В Text Inversion [9] текстовые описания имеют вид «context instance_token», где instance_token это новый добавленный токен в словарь токенизатора, а вектор в текстовом энкодере соответствующий новому токену инициализируются вектором токена домена пользовательских фотографий, то есть instance_token = class_token в начале дообучения, context - случайно выбранная фраза из списка шаблонов, предлагаемых в CLIP [38].

В Dreambooth [43] отмечают, что текстовое описание, состоящие из instance_token без class_token увеличивает время обучения, поэтому было принято решение выбрать текстовое описание, которое строится по следующему шаблону: «context instance_token class_token».

2.2. Инициализация псевдослов

Текстовые описания передаются в токенизатор, который разбивает предложение на токены. Каждый токен связан с уникальным вектором, который хранится в словаре текстового энкодера. Поскольку шаблон текстовых описаний имеет вид «context instance_token class_token», то необходимо в качестве context, instance_token и class_token взять существующие слова или создать новые токены и проинициализировать их.

В подходах по персонализации диффузионной модели [9, 25, 43] в качестве class_token берется описание домена новой концепции, но его задает

пользователь. Инициализация instance_token очень важна и напрямую влияет на скорость и качество дообучения диффузионной модели [5]. Есть две стратегии инициализации instance_token, которые предлагают Dreambooth [43] и Text Inversion [9].

2.2.1. Инициализация домена

Для уменьшения количества кликов в приложении [11] и ошибок неправильного написания домена был разработан алгоритм автоматической инициализации class_token. Так как наш домен - это фотографии людей, то в качестве class_token можно взять такие слова как «human» или «person». Было замечено, что выбор class_token как слово из списка [girl, man, boy, baby], то есть с учетом пола и возраста, дает более качественные результаты дообучения и генерации**.** Для решения этой задачи переиспользовалась модель CLIP [38], используемая в Stable Diffusion [49], так как эта модель обучалась так, чтобы текстовые представления были похожи на визуальные представления изображений. Таким образом, class_token определяется как слово из списка [girl, man, boy, baby], у которого наименьшее среднее косинусное расстояние между его текстовым представлением и визуальными представлениями пользовательских фотографий.

2.2.2. Инициализация нового псевдослова

В Dreambooth и Custom Diffusion [43, 25] в качестве instance_token берут существующий токен из словаря. Авторы не рекомендуют брать в качестве instance_token такие слова, как «unique» или «special», так как эти слова часто встречаются в обучающем наборе модели, то есть модель должна сначала забыть их значения, а потом выучить новую концепцию, это приводит к менее качественной и более долгой настройке модели. Также не стоит выбирать слово в виде случайного набора символов, так как токенизатор разобьет такое слово на токены частотных букв. Самым лучшим

решением авторы считают случайный выбор среди малочастотных токенов состоящих из одного-трех символов.

В Text Inversion [9] предлагают просто добавить instance_token в словарь токенизатора как новый токен, и сопоставить ему вектор в текстовом энкодере, который соответствует вектору class_token, то есть class_token - это слово, которое является токеном. Так же, вдохновляясь инверсией генеративной состязательных моделей, авторы экспериментируют, показывая, что будет если instance_token сопоставить сразу несколько токенов [1] или делать прогрессивное обучение [50] при помощи нескольких токенов, то есть первые шаги оптимизируют первый токен, потом второй и так далее. В итоге авторы показывают, что инициализация instance_token одним токеном показывает лучшие результаты.

В ходе экспериментов было замечено, что нет разницы между взять instance_token как малочастотный токен или добавить instance_token в словарь как новый токен и инициализировать его малочастотным токеном. Второй способ является более гибким, так как дает возможность инициализации instance_token частотными словами без изменения их в изначальной модели.

Главная задача разрабатываемого алгоритма – инициализировать instance_token так, чтобы модель быстро и качественно выучила идентичность пользователя. Было рассмотрено несколько вариантов того, как инициализировать вектор, который будет соответствовать новому псевдослову instance_token:

1. Нормальный шум.
2. Малочастотный токен, как предлагают в Dreambooth [43].
3. Нет смысла инициализировать instance_token описанием домена пользовательских фотографий, как делают в Text Inversion [9], так как шаблон текстовых описаний «context instance_token class_token» уже

содержит class_token, поэтому была рассмотрена инициализация словом «face».

4. Проинициализировать instance_token похожей на пользователя знаменитостью, которая точно присутствовала в обучающем наборе модели. Для этого был собран список из 100 самых знаменитых людей из интернета, чьи имена и фамилии являются двумя отдельными токенами в словаре токенизатора. Ближайшая медийная личность находилась также как и поиск class_token, описанный выше, только список текстовых описаний состоял из имен знаменитостей. В данном подходе instance_token это два токена «instance_token_0 instance_token_1», которые соответствуют имени и фамилии.

2.2.3. Выбор контекста

Так же было исследовано влияние выбора context:

1. «A photo of» как в Dreambooth [43].
2. Случайное словосочетание из шаблонов CLIP [38], как в Text Inversion [9].
3. Является уникальным токеном background_token_i, который соответствует конкретному пользовательскому изображению [9]. Каждый background_token_i добавляется в токенизатор и случайно инициализируется.
4. Сгенерированное текстовое описание для каждого изображения при помощи модели BLIP [28].

2.3. Дообучение диффузионной модели

Тонко настроенная диффузионная модель должна уметь обобщать и комбинировать новую концепцию с уже существующими выученными концепциями, чтобы не потерять возможность создавать разнообразные генерации новой концепции. Как отмечают в Dreambooth и Custom Diffusion [43, 25], дообучение большой диффузионной модели на маленьком наборе

пользовательских фотографий может привести к двум проблемам. Во-первых, модель может переобучиться под данные, что приведет к потере выразительной способности при генерации, то есть модель сможет генерировать новую концепцию только в тех позах, которые представлены на пользовательских фотографиях, а в худшем случае потеряет возможность генерировать что-то кроме новой концепции. Во-вторых модель может забыть [27, 30], как генерировать другие объекты домена новой концепции, что приведет к потере встроенных знаний о разнообразии и естественных положений объектов этого домена.

Для решения второй проблемы Dreambooth и Custom Diffusion [43, 25] предлагают взять регуляризационные изображения, которые встраиваются в обучающий набор данных. В Dreambooth [43] предлагают создать набор при помощи генерации исходной моделью изображений домена пользовательских фотографий. Custom Diffusion [25] отмечают, что лучше не генерировать, а выбрать реальные изображения из набора данных LAION-400M [46] с соответствующими подписями, которые имеют высокое сходство с выбранными текстовыми описаниями. Из-за того, что эти модели дообучаются на смешанном наборе данных, им необходимо делать больше обучающих шагов, чтобы выучить новую концепцию. Стоит отметить, что потеря возможности генерировать портреты обычных людей не критична для разрабатываемого алгоритма, главное сохранить обобщающую способность генерировать пользовательские портреты в разных стилях и окружениях, поэтому для повышения вычислительной эффективности данная регуляризация не использовалась.

Модель Text Inversion [9] меньше склонна к переобучению или забыванию значений слов, так как веса исходной модели заморожены. Но из-за этого ей необходимо делать в разы больше шагов оптимизации, что негативно влияет на время дообучения.

Для решения проблемы переобучения было рассмотрено поэтапное обучение отдельных частей модели с целью найти оптимальное количество шагов и процентное соотношение этапов при тонкой настройке модели. Первый этап заключается в обучении вектора соответствующего instance_token без/вместе с U-net [42]. Второй этап - обучение всей текстовой модели CLIP [38] без/вместе с U-net [42]. Третий этап - обучение U-net [42]. Результаты поэтапного обучения сравнивались с обучением Text Inversion [9] и DreamBooth [43]. Так как DreamBooth [43] является закрытой моделью, была реализована ее имплементация.

Было замечено, что если пользовательские фотографии сделаны в одном окружении, то модели в ходе дообучения сложно разделить, какие признаки относятся к самому пользователю, а какие к его окружению. Поэтому если все фотографии, например, были сделаны в комнате с зелёными стенами, то при генерации новой концепции преобладал зеленый цвет на фоне. Для разделения информации о пользователе и его окружении были предприняты следующие шаги:

1. Инициализация контекста при помощи BLIP или добавления уникальных токенов для каждого изображения, как описано в главе 2.2.3.
2. Вдохновившись работой Prompt-to-Prompt [13], к основной функции потери добавлялась взвешенная маскированная бинарная кросс-энтропия между усреднением всех матриц перекрестного внимания размера 16 на 16 пикселей, соответствующих instance_token и нулевой матрицей. В качестве маски выступает инвертированная маска объекта, которая находилась сегментационной моделью [37]. Таким образом, instance_token штрафуется за то, что он влияет на что-то, кроме объекта. Такой штраф чем-то похож на идею Attend-and-Excitex [5], где авторы штрафуют во время генерации за то, что у выбранных

токенов максимальное значение усредненной матриц перекрестного внимания маленькое.

Так как в мобильное приложение [11] может параллельно прийти несколько запросов на генерацию персонализированных портретов, была рассмотрена возможность дообучения модели сразу на нескольких пользователях с целью оптимизации вычислительных ресурсов.

2.4. Генерация результатов

На текущий момент существует много разных способов генерации изображений из обученной диффузионной модели, например DDPM [15], DDIM [48], Euler или Heun [21]. Также авторы [21] показывают, что для генерации результатов можно использовать стратегию, которая отличается от стратегии при обучении, главное, чтобы модель в ходе генерации делала шаги в сторону более высокой плотности данных при текущем уровне шума. В ходе экспериментов был выбран метод Euler [21], который за минимальное количество шагов расшумления получал качественные изображения новой концепции.

2.4.1. Контролируемая генерация

Несмотря на то, что дообучение происходило на фотографиях, на которых голова пользователя полностью присутствует, модель часто в ходе генерации отсекала часть портрета. Эту проблему также описывают авторы блога [35], она связана с тем, что изначальная модель обучалась на данных с центральным вырезанием, а не случайным. Несмотря на то что дообучение модели происходит на квадратных изображениях, это не ограничивает возможность генерировать изображения в формате 9:16, который необходим для корректного размещения в различных социальных сетях. Но на таких генерациях голова либо неестественно растягивалась или дублировалась. Для решения вышеперечисленных проблем можно подключить легко переносимую сеть ControlNet [55], которая при генерации дополнительно

учитывает границы, скелет или карту глубины. Также появляется возможность управлять положением головы не словами, а изображениями, или решать задачу по замене лиц [54], если просить пользователя дополнительно загружать изображение, на котором он хочет заменить лицо на свое. Однако использование ControlNet [55] вводит ограничение на создание фиксированного набора условных изображений.

2.4.2. Улучшение результатов генерации

Для пользователя генерируется множество изображений, и не все из них получаются качественными из-за неудачной инициализации. Поэтому возникает задача ранжирования результатов генерации. Для этого было реализовано четыре способа упорядочивания множества сгенерированных изображений, которые использовали следующие метрики:

1. Среднее значение LPIPS [56] между сгенерированным изображением и пользовательскими фотографиями.
2. Среднее значение косинусного расстояния между визуальными представлениями CLIP [38] сгенерированного изображения и пользовательских фотографий.
3. Значение косинусного расстояния в пространстве CLIP [38] между визуальным представлением сгенерированного изображения и текстовым представлением описания, по которому это изображение было сгенерировано.
4. Значение модели aesthetic-predictor [2] для визуальных представлений CLIP сгенерированных изображений.

Модель обучалась на изображениях 512 на 512 пикселей, поэтому лучше всего генерировать изображения такого же разрешения. Некоторые социальные сети требуют разрешение изображения 1024 на 1024 пикселей, то есть при загрузке сгенерированные изображения будут растянуты, что приведет к потере качества. Можно генерировать изображения 1024 на 1024 пикселей, но тогда возникают проблемы, описанные в главе 2.4.1., а также это

сильно повышает вычислительную нагрузку. Поэтому разрешение увеличивалось в 2 раза при помощи Real-ESRGAN [51].

Как отмечают авторы модели Kandinsky 2.1 [19], при создании новой версии изменение архитектуры модели VAE [24] привело к сильному приросту качества. В ходе экспериментов было обнаружено, что простое изменение стандартных весов VAE [24] на дообученные [17] приводит к большому приросту качества генераций глаз.

3. Результаты

Все результаты получены при дообучении диффузионной модели с одинаковым количеством шагов и при фиксированном зерне случайных чисел.

3.1. Создание персонализированного набора данных

В процессе подготовки персонализированного набора данных найденные изображения лиц пользователей преобразовывались к размеру 512 на 512 пикселей. Однако изначально лицо может занимать маленькую область на пользовательском изображении, поэтому сильное изменение размера может привести к значительной потере информации. Для решения этой проблемы можно использовать не классические методы повышения разрешения изображения, а нейросетевые, например Real-ESRGAN [51]. На Рис. 4 продемонстрировано сравнение результатов дообучения диффузионной модели на исходных лицах и на лицах, к которым был применен Real-ESRGAN.

А Б С

Рис. 4. А - пример лица из обучающей выборки. Б - результаты диффузионной модели дообученной на исходных лицах. В. - результаты диффузионной модели дообученной на улучшенных Real-ESRGAN исходных лицах

Можно заметить, что на представленных изображениях значительных изменений нет. Скорее всего это связанно с тем, что перевод изображений в латентное представление при помощи модели VAE [31], которая уменьшает

размер изображения в 8 раз, инвариантно к небольшим шумам, возникающим во время классического увеличения разрешения изображений.

Как отмечалось в главе 2.2.1., домен пользовательских фотографий – это люди. Поэтому в качестве class_token можно взять слова «human» или «person». Было замечено, что инициализация class_token гендером пользователя повышало скорость сходимости при одинаковом числе шагов дообучения. Однако, как видно на Рис. 5, если в качестве домена для молодой девушки взять слово «woman», а не «girl», то это приведет к генерациям, которые старят девушку. Поэтому был специально подобран список доменов [girl, man, boy, baby], который в большинстве случаев обеспечивал хорошие результаты на пользователях разных возрастов. В ходе экспериментов CLIP показал хорошее качество определение домена, таким образом его использование не требует поддержки отдельной модели, он инвариантен к размеру изображения и его предсказания легко расширяемые.

А Б

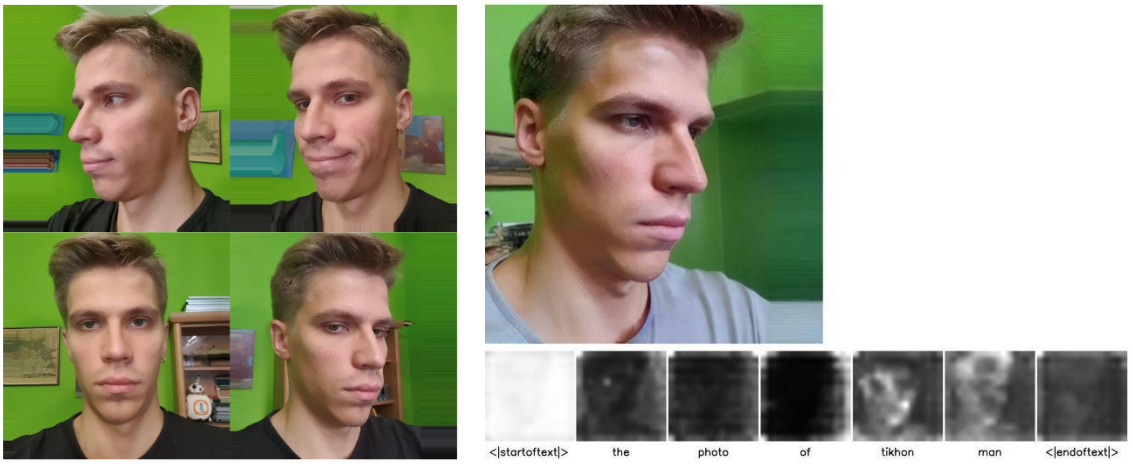
Рис. 5. А - примеры лиц из обучающей выборки. Б - результаты генерации по текстовому запросу «A photo of instance_token class_token» дообученной диффузионной модели при одинаковым числе шагов и разных инициализациях class_token

Были проведены эксперименты по влиянию разных инициализаций instance_token. На Рис. 6 можно увидеть, что инициализация через ближайшую к пользователю знаменитость приводит к смешиванию двух идентичностей. Случайная инициализация через нормальный шум или инициализация через малочастотный токен, как предлагают в DreamBooth, визуально дает результаты менее похожие на пользователя, чем инициализация словом «face». Поэтому в качестве инициализации instance_token было выбрано слово «face».

А Б В Г Д

Рис 6. А - пример лица из обучающей выборки, Б - результаты дообученной диффузионной модели при инициализации instance_token нормальным шумом, В - результаты дообученной диффузионной модели при инициализации instance_token малочастотным токеном, Г - результаты дообученной диффузионной модели при инициализации instance_token словом «face», Д - результаты дообученной диффузионной модели при инициализации instance_token именем ближайшей к пользователю знаменитости

Как было отмечено в главе 2.3., диффузионной модели сложно разделить, какие признаки относятся к самому пользователю, а какие к его окружению, если оно не изменяется на всех изображениях. На Рис. 7, на котором представлены усредненные матрицы перекрестного внимания для каждого токена, видно, что instance_token («tikhon») влияет не только на внешность пользователя.



А Б

Рис. 7. А - пользовательские изображения, Б - результат генерации и усредненные матрицы перекрестного внимания для каждого токена в текстовым запросе «the photo of tikhon man», где «tikhon» - instance_token и «man» - class_token

Для решения этой проблемы были проведены эксперименты по инициализации context. На Рис. 8 продемонстрировано, что инициализация context текстовым описанием, полученным при помощи модели BLIP или индивидуальным background_token_i для каждого изображения, убирает зелёный цвет фона на генерациях, который преобладает на исходных изображениях пользователя.

А Б В Г

Рис. 8. А - пользовательские изображения, Б - результаты дообученной диффузионной модели при инициализации context фразой «a photo of», В - результаты дообученной диффузионной модели при инициализации context текстовым описанием, полученные при помощи модели BLIP, Г - результаты дообученной диффузионной модели при инициализации context индивидуальным background_token_i

Процесс генерации текстового описания моделью BLIP для изображения схематично представлен на Рис. 9.

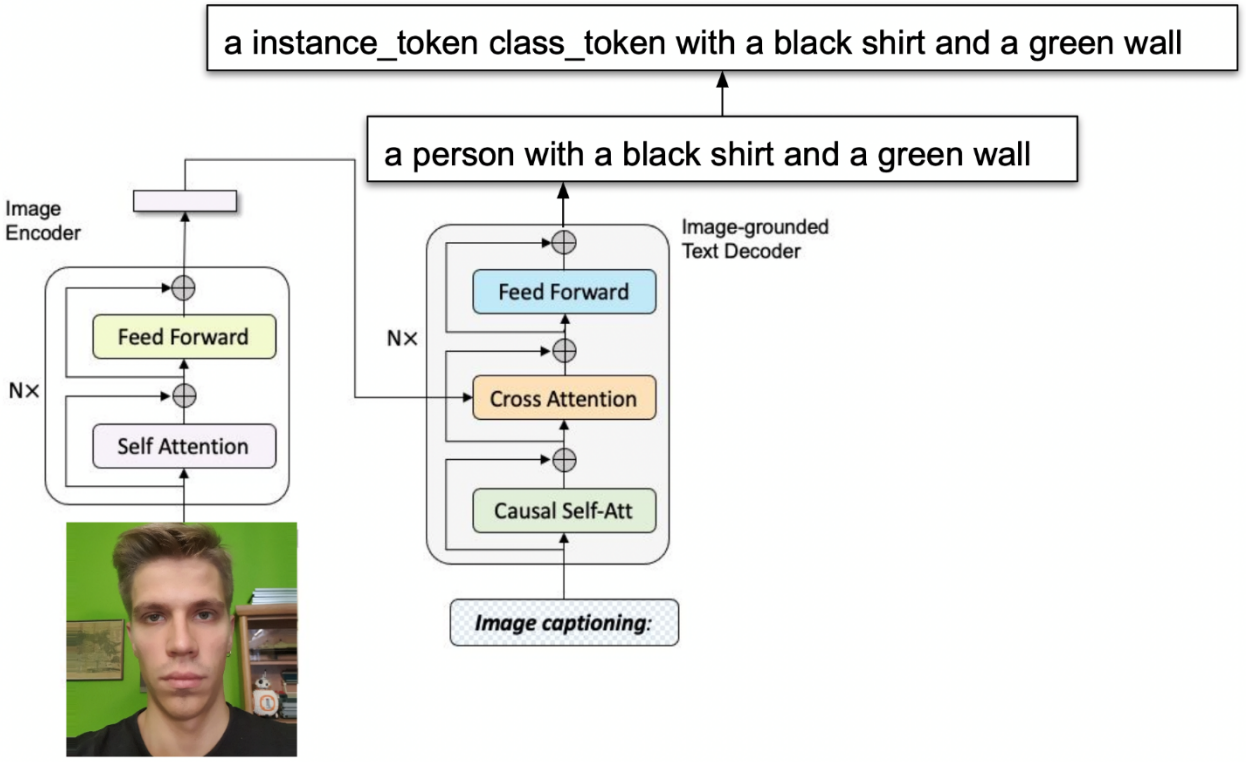


Рис. 9. Схема генерации текстового описания моделью BLIP

В полученном текстовом описании слова «man», «woman», «person» и т.п. заменялись при помощи регулярного выражения на «instance_token class_token», если они не были найдены, то «instance_token class_token» добавлялось в начало описания.

Использование шаблонов, которые предлагаются в Text Inversion, ухудшали сходимость. В итоге была выбрана стратегия создания индивидуальных токенов, так как это вычислительно эффективнее, чем для каждого изображения находить описание.

Добавление маскированной бинарной кросс-энтропии для матриц перекрестного внимания к основной функции потери практически в два раза повышала сходимость, однако модель теряла выразительную способность и не могла генерировать ничего, кроме пользовательских изображений.

3.2. Дообучение диффузионной модели

Для решения проблемы переобучения диффузионной модели под маленький набор пользовательских фотографий были использованы разные поэтапные стратегии дообучения. Первый этап заключался в обучении вектора, соответствующего instance_token без/вместе с Unet, второй этап - обучение всей текстовой модели CLIP без/вместе с Unet, третий этап - обучение только Unet. В ходе выбора лучшей стратегии было замечено, что обучение только вектора, соответствующего instance_token, или замораживание U-net на первых этапах приводит к ухудшению сходимости. Обучение же всей модели приводило к быстрой сходимости, однако она вырождалась. Поэтому была выбрана стратегия обучения сначала CLIP с U-net, а затем дообучение только U-net. Это приводило к результатам, похожим на дообучение всей модели, при этом сохранялась ее выразительная способность. Визуальные результаты поэтапного дообучения представлены на Рис. 10.

А Б В Г Д Е

Рис. 10. Результаты дообучения диффузионной модели при различных стратегиях. А - пример лица из обучающей выборки. Б - обучение вектора instance_token без U-net + U-net. В - обучение вектора instance_token с U-net + U-net. Г - обучение CLIP без U-net + U-net. Д - обучение CLIP с U-net + U-net. Е - обучение CLIP с U-net

Было произведено сравнение выбранной лучшей стратегии с Text Inversion и собственной имплементацией Dreambooth (Рис. 11).

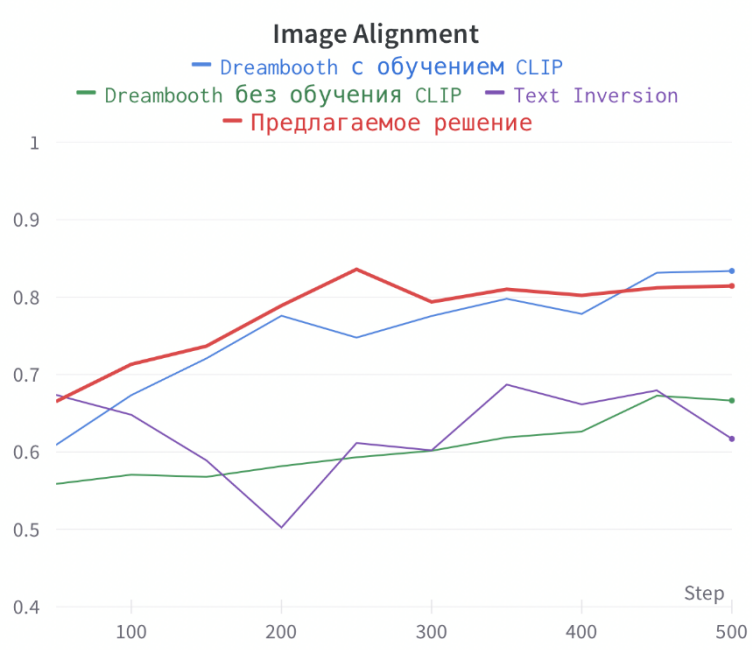
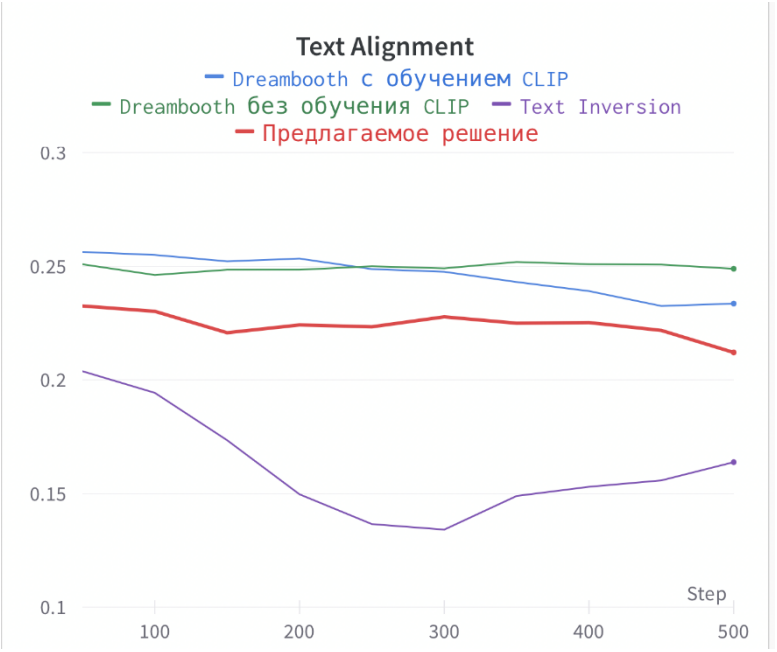
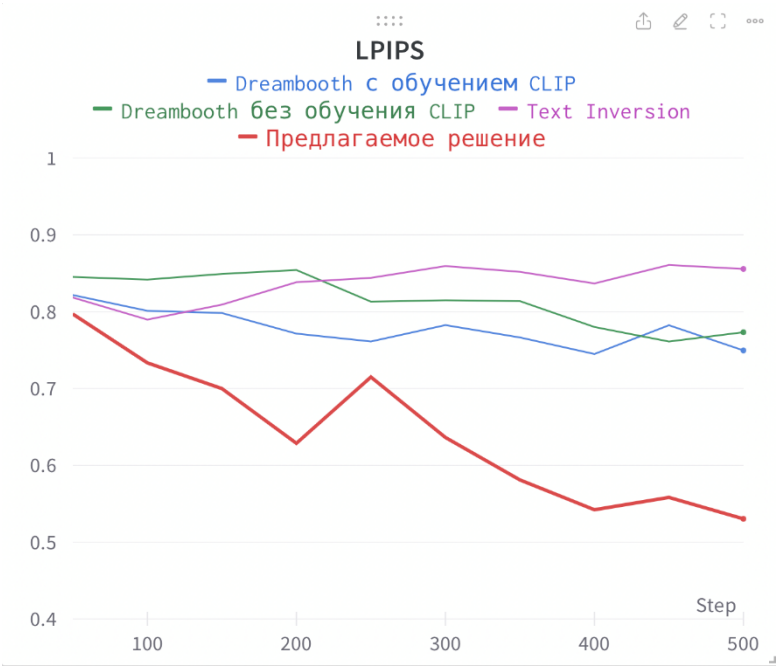


Image Alignment
Dreambooth с обучением CLIP
Dreambooth без обучения CLIP
Text Inversion
Предлагаемое решение
1
0.9
0.8
0.7
0.6
0.5
0.4
Step
100
200
300
400
500

А


Text Alignment
Dreambooth с обучением CLIP
Dreambooth без обучения CLIP
Text Inversion
Предлагаемое решение
0.3
0.25
0.2
0.15
0.1
Step
100
200
300
400
500

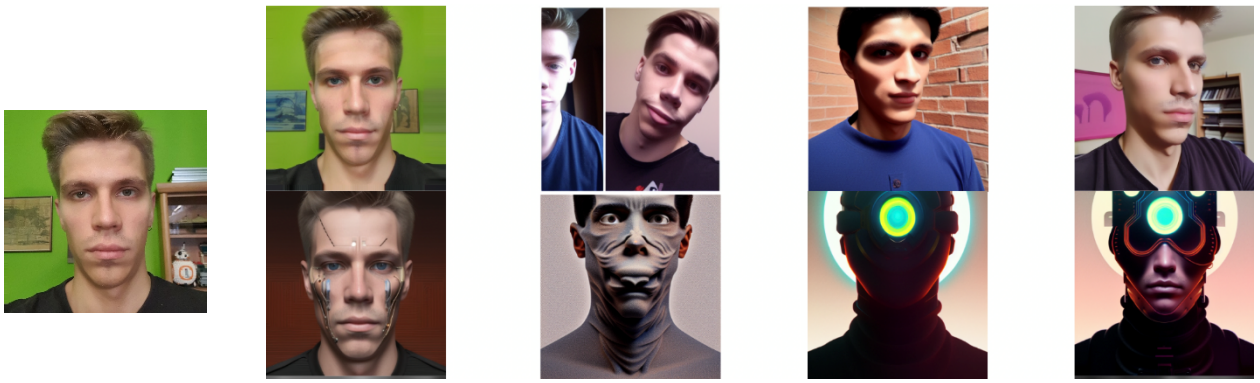
Б



В

Рис. 11. Замеры метрик каждые 50 шагов. А - Image Alignment, Б - Text Alignment, В - LPIPS

Из представленных графиков видно, что предлагаемое решение быстрее начинает генерировать изображения, похожие на пользовательские, так как значения метрики LPIPS быстрее падают, чем у других подходов, а Image Aligment - растут. При этом метрика Text Aligment показывает, что предлагаемое решение дообучения не сильно деградирует модель и остается на том же уровне, что и другие подходы. На Рис 12. представлено визуальное сравнение перечисленных подходов.



А Б В Г Д

Рис. 12. Результаты генерации дообученной диффузионной модели по текстовому запросу «a photo of instance_token class_token» и «a photo of instance_token class_token as cyborg». А - пример лица из обучающей выборки. Б - предлагаемое решение. В - Text Inversion. Г - DreamBooth без обучения CLIP. Д - DreamBooth с обучением CLIP

Как видно из представленных данных, предлагаемое решение намного качественнее восстанавливает изображение пользователя, при этом сохраняет возможность генерировать его персонализированные портреты в разных стилях.

Так как в мобильное приложение может параллельно прийти несколько запросов на генерацию персонализированных портретов, была рассмотрена возможность дообучения модели сразу на нескольких пользователях с целью оптимизации вычислительных ресурсов. Это осуществлялось при помощи объединения пользовательских фотографий в один набор данных для последующего обучения. Результаты дообучения модели представлены на Рис. 13. Как видно из представленных данных, одинаковая инициализация двух новых псевдослов приводит к смешиванию идентичностей, а случайная инициализация приводит к неоднородной скорости обучения новым концепциям. Также в Custom Diffusion говорится о том, что подходы, в которых обучается вся сеть, плохо умеют объединять разные концепции. Поэтому запрос каждого пользователя обрабатывается отдельно.

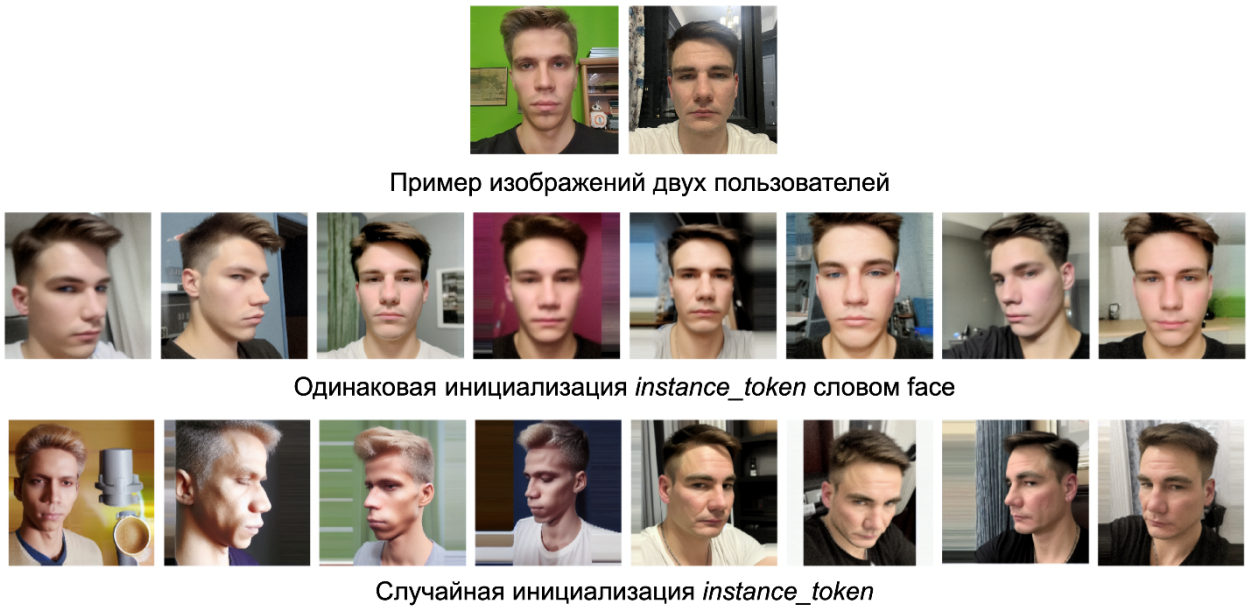
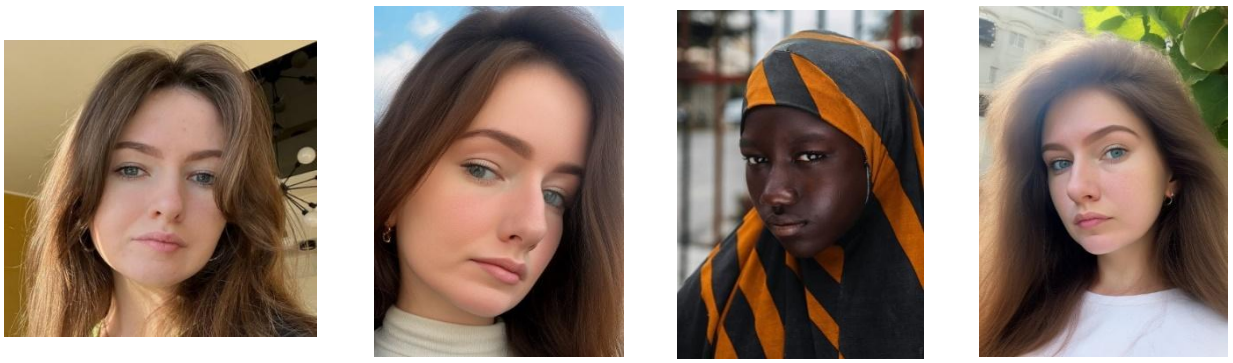


Рис. 13. Результаты дообучения диффузионной модели под двух пользователей

3.3. Улучшение результатов генерации

Было исследовано подключение модели ControlNet для устранения проблем генераций изображений, на которых портрет пользователя обрезался или неестественно растягивался. На Рис. 14 видно, как дополнительное условие на позу, скелет или карту глубины решает эту проблему (добавление позы девушки в платке позволило сгенерировать корректный пользовательский портрет). Также использование ControlNet позволяет решать задачу по замене лиц (Рис. 15).



А Б В Г

Рис. 14. А - пример лица из обучающей выборки. Б - результат генерации по текстовому описанию «a photo of instance_token class_token». В - изображение, которому строится условие в виде позы. Г - результат генерации по текстовому описанию «a photo of instance_token class_token» и дополнительному условию на позу

Рис. 15. Решение задачи замены лица при помощи модели ControlNet

Для пользователя генерируется множество изображений. Иногда из-за неудачной инициализации начальной точки качество генерации сильно падает. Для этого было реализовано четыре способа упорядочивания множества сгенерированных изображений с помощью метрик, описанных в главе 2.4.2. (Рис. 16).

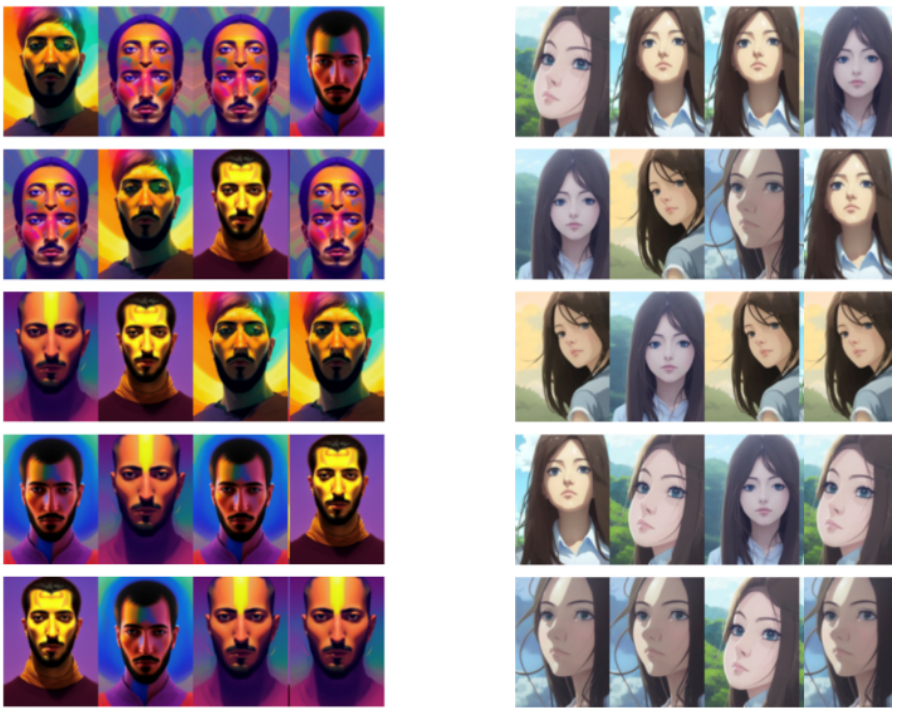


Рис. 16. Результаты ранжирования сгенерированных изображений, где столбец соответствует метрике, а строка рангу от худшего к лучшему

Был проведён внутренний опрос среди сотрудников компании. По результатам голосования был выбран метод ранжирования на основе модели aesthetic-predictor. Таким образом пользователь получает k лучших из n генераций.

Для публикации сгенерированных изображений в социальных сетях они должны иметь высокое разрешение, чтобы после загрузки не произошло потери качества. Однако линейное повышение разрешения ведет к квадратичному увеличению времени генерации. Поэтому для повышения разрешения сгенерированных изображений к ним применялась модель Real-ESRGAN (Рис. 17).

А Б

Рис. 17. А - результат генерации. Б - применение Real-ESRGAN к результату генерации

Данный подход дает вычислительно эффективное повышение разрешения изображений до 4 раз практически без потери деталей.

На основании результатов исследовательской работы был разработан прототип в виде телеграмм бота. В дальнейшем представленный алгоритм дообучения диффузионной модели для генерации персонализированных портретов был внедрен в мобильное приложение GLAM.

ЗАКЛЮЧЕНИЕ

Проведенное исследование в рамках данной работы позволило разработать эффективный алгоритм дообучения диффузионной модели, который по скорости сходимости превосходит существующие аналогичные методы, а также создать полноценный прототип, который по необработанным пользовательским изображениям генерирует их высококачественные персонализированные портреты в разных стилях и сценах. Дополнительным преимуществом разработанного алгоритма являются хорошие результаты генерации человеческих лиц, что выделяет его на фоне других методов, не демонстрирующих свои результаты на домене людей.

Для достижения этих результатов были решены следующие задачи:

- Разработан алгоритм обработки пользовательских фотографий
- Исследовано влияние выбора текстового описания и инициализации новых псевдослов
- Найдена стратегия дообучения, которая позволила достигнуть быстрой сходимости без сильной потери выразительной способности исходной модели
- Проведено сравнение с моделями, выбранными в качестве бейзлайна
- Исследовано подключение модели ControlNet для устранения проблем генераций изображений, на которых портрет пользователя неестественно изменялся
- Разработан алгоритм ранжирования, чтобы минимизировать число неудачных генераций в итоговых изображениях
- Предложен вычислительно эффективный способ повышения разрешения сгенерированных изображений

Полученное решение было интегрировано в мобильное приложение GLAM. Его share rate составил 50%.

СПИСОК ЛИТЕРАТУРЫ

1. Abdal, R. et al. Image2StyleGAN: How to Embed Images Into the StyleGAN Latent Space? //arXiv:1904.03189. – 2019. URL:https://arxiv.org/abs/1904.03189
2. Aesthetic-predictor. URL:https://github.com/LAION-AI/aesthetic-predictor
3. Bazarevsky, V. et al. BlazeFace: Sub-millisecond Neural Face Detection on Mobile GPUs // arXiv:1907.05047. – 2019. URL: https://arxiv.org/abs/1907.05047
4. Brooks, T . et al. InstructPix2Pix: Learning to Follow Image Editing Instruction //arXiv:2211.09800. – 2022. URL:https://arxiv.org/abs/2211.09800
5. Chefer, H. et al. Attend-and-Excite: Attention-Based Semantic Guidance for Text-to-Image Diffusion Models //arXiv:2301.13826. – 2023. URL:https://arxiv.org/abs/2301.13826
6. Deng, J. et al. ArcFace: Additive Angular Margin Loss for Deep Face Recognition //arXiv:1801.07698 – 2018. URL:https://arxiv.org/abs/1801.07698
7. Deng, J. et al. RetinaFace: Single-stage Dense Face Localisation in the Wild // arXiv:1905.00641. – 2019. URL: https://arxiv.org/abs/1905.00641
8. Dhariwal, P. et al. Diffusion Models Beat GANs on Image Synthesis //arXiv:2105.05233. – 2021. URL: https://arxiv.org/abs/2105.05233
9. Gal, R. et al. An Image is Worth One Word: Personalizing Text-to-Image Generation using Textual Inversion //arXiv:2208.01618. – 2022. URL:https://arxiv.org/abs/2208.01618
10. Gal, R. et al. StyleGAN-NADA: CLIP-Guided Domain Adaptation of Image Generators //arXiv:2108.00946. – 2021. URL:https://arxiv.org/abs/2108.00946
11. Glam: AI photo & video editor [Мобильное приложение]. URL: https://apps.apple.com/th/app/glam-ai-photo-video-editor/id1545593132.
12. Goodfellow, I J. et al. Generative Adversarial Networks //arXiv:1406.2661. – 2014. URL: https://arxiv.org/abs/1406.2661
13. Hertz, A. et al. Prompt-to-Prompt Image Editing with Cross Attention Control //arXiv:2208.01626. – 2022. URL:https://arxiv.org/abs/2208.01626

14. Ho, J. et al. Classifier-Free Diffusion Guidance //arXiv:2207.12598. – 2022. URL:https://arxiv.org/abs/2207.12598
15. Ho, J. et al. Denoising Diffusion Probabilistic Models //arXiv:2006.11239. – 2020. URL:https://arxiv.org/abs/2006.11239
16. Hu, E J. et al. LoRA: Low-Rank Adaptation of Large Language Models //arXiv:2106.09685. – 2021. URL:https://arxiv.org/abs/2106.09685
17. Hugging Face [сайт] URL: https://huggingface.co/stabilityai/sd-vae-ft-mse-original
18. Isola, P. et al. Image-to-Image Translation with Conditional Adversarial Networks //arXiv:1611.07004. – 2016. URL: https://arxiv.org/abs/1611.07004
19. Kandinsky 2.1. URL:https://github.com/ai-forever/Kandinsky-2
20. Kang, M. et al. Scaling up GANs for Text-to-Image Synthesis //arXiv:2303.05511. – 2023. URL: https://arxiv.org/abs/2303.05511
21. Karras, T. et al. Elucidating the Design Space of Diffusion-Based Generative Models //arXiv:2206.00364. – 2020. URL:https://arxiv.org/abs/2206.00364
22. Karras, T. et al. A Style-Based Generator Architecture for Generative Adversarial Networks //arXiv:1812.04948. – 2018. URL:https://arxiv.org/abs/1812.04948
23. Kawar, B. et al. Imagic: Text-Based Real Image Editing with Diffusion Models //arXiv:2210.09276. – 2022. URL:https://arxiv.org/abs/2210.09276
24. Kingma, D P. et al. Auto-Encoding Variational Bayes //arXiv:1312.6114. – 2013. URL:https://arxiv.org/abs/1312.6114
25. Kumari, N. et al. Multi-Concept Customization of Text-to-Image Diffusion//arXiv:2212.04488 – 2022. URL:https://arxiv.org/abs/2212.04488
26. Lambda Diffusers. URL:https://github.com/LambdaLabsML/lambda-diffusers
27. Lee, J. et al. Countering Language Drift via Visual Grounding //arXiv:2208.01618. – 2019. URL:https://arxiv.org/abs/1909.04499
28. Li, J. et al. BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation //arXiv:2201.12086. – 2022.

URL:https://arxiv.org/abs/2201.12086
29. Liu, W. et al. SSD: Single Shot MultiBox Detector //arXiv:1512.02325. – 2015. URL:https://arxiv.org/abs/1512.02325 /
30. Lu, Y. et al. Countering Language Drift with Seeded Iterated Learning //arXiv:2003ю12694. – 2020. URL:https://arxiv.org/abs/2003.12694
31. Mildenhall, B. et al. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis //arXiv:2003.08934. – 2020. URL:https://arxiv.org/abs/2003.08934
32. Mokady, R . et al. Null-text Inversion for Editing Real Images using Guided Diffusion Models //arXiv:221.09794. – 2022.
URL:https://arxiv.org/abs/2211.09794
33. Mou, C. et al. T2I-Adapter: Learning Adapters to Dig out More Controllable Ability for Text-to-Image Diffusion Models //arXiv:2302.08453. – 2023.
URL:https://arxiv.org/abs/2302.08453
34. Nichol, A. et al. GLIDE: Towards Photorealistic Image Generation and Editing with Text-Guided Diffusion Models //arXiv:2112.10741. – 2021. URL: https://arxiv.org/abs/2112.10741
35. NovelAI Improvements on Stable Diffusion[сайт].
URL:https://blog.novelai.net/novelai-improvements-on-stable-diffusion-e10d38db82ac
36. Patashnik, O. et al. StyleCLIP: Text-Driven Manipulation of StyleGAN Imagery //arXiv:2103.17249. – 2021. URL:https://arxiv.org/abs/2103.17249
37. Qin, X. et al. Highly Accurate Dichotomous Image Segmentation //arXiv:2203.03041. – 2022. URL:https://arxiv.org/abs/2203.03041
38. Radford, A. et al. Learning Transferable Visual Models From Natural Language Supervision //arXiv:2103.00020. – 2021.
URL:https://arxiv.org/abs/2103.00020
39. Redmon, J. et al. You Only Look Once: Unified, Real-Time Object Detection //arXiv:1506.02640. – 2015. URL:https://arxiv.org/abs/1506.02640

40. Rombach, R. et al. High-Resolution Image Synthesis with Latent Diffusion Models //arXiv:2112.10752. – 2021. URL:https://arxiv.org/abs/2112.10752
41. Ronneberger, O. et al. Attention is all you need //arXiv:1706.03762. – 2017. URL:https://arxiv.org/abs/1706.03762
42. Ronneberger, O . et al.U-Net: Convolutional Networks for Biomedical Image Segmentation//arXiv:1505.04597. – 2015. URL:https://arxiv.org/abs/1505.04597
43. Ruiz, N. et al. DreamBooth: Fine Tuning Text-to-Image Diffusion Models for Subject-Driven Generation // arXiv:2208.12242. – 2022. URL:https://arxiv.org/abs/2208.12242
44. Saharia, C. et al. Hierarchical Text-Conditional Image Generation with CLIP Latents //arXiv:2204.06125. – 2022. URL:https://arxiv.org/abs/2204.06125
45. Saharia, C. et al. Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding //arXiv:2205.11487. – 2022. URL: https://arxiv.org/abs/2205.11487
46. Schuhmann, C . et al. LAION-400M: Open Dataset of CLIP-Filtered 400 Million Image-Text Pairs //arXiv:2111.02114. – 2021. URL:https://arxiv.org/abs/2111.02114
47. Sohl-Dickstein, J. et al.Deep Unsupervised Learning using Nonequilibrium Thermodynamics //arXiv:1503.03585. – 2015. URL:https://arxiv.org/abs/1503.03585
48. Song, J. et al. Denoising Diffusion Implicit Models// arXiv:2010.02502. – 2020. URL:https://arxiv.org/abs/2010.02502
49. Stable Diffusion[сайт]. URL:https://github.com/CompVis/stable-diffusion
50. Tov, O. et al. Designing an encoder for stylegan image manipulation //arXiv:2102.02766. – 2021. URL:https://arxiv.org/abs/2102.02766
51. Wang, X. et al. Real-ESRGAN: Training Real-World Blind Super-Resolution with Pure Synthetic Data //arXiv:2107.10833. – 2021. URL:https://arxiv.org/abs/2107.10833
52. Wu, H . et al. Gp-gan: Towards realistic high-resolution image blending

//arXiv:1703.07195. – 2017. URL: https://arxiv.org/abs/1703.07195
53. Xu, W. et al. DRB-GAN: A Dynamic ResBlock Generative Adversarial Network for Artistic Style Transfer //arXiv:2108.07379. – 2021. URL: https://arxiv.org/abs/2108.07379
54. Yu, H. et al. Migrating Face Swap to Mobile Devices: A lightweight Framework and A Supervised Training Solution //arXiv:2204.08339. – 2022. URL:https://arxiv.org/abs/2204.08339
55. Zhang, L. et al. Adding Conditional Control to Text-to-Image Diffusion Models //arXiv:2302.05543. – 2023. URL:https://arxiv.org/abs/2302.05543
56. Zhang, R. et al. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric //arXiv:1801.03924 – 2018. URL:https://arxiv.org/abs/1801.03924
57. Zhu, J . et al. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks //arXiv:1703.10593. – 2017. URL: https://arxiv.org/abs/1703.10593