

mipt2024s-5-belkov-alexey

Report 30.85.3034 Елена Александровна Давыдова МОС-301С

Задача оптимизации генератора. Задача оптимизации генератора представляет собой задачу оптимизации черного ящика, для её решения понадобятся следующая информация:

- Пространство оптимизируемых параметров
- Оценка качества работы генератора
- Метод оптимизации

Разбор генератора

На данный момент имеется генератор кодов, генератор искажений и набор аутентификаций

- Генерируется изображение кода
- Применяется реперсиве transform (далее RF) со случайными параметрами
- Применяется фиксированный набор аугментаций (по каждой из них уже случайным образом воздействует на изображение)
- Полученное изображение кода вставляется в заранее заданное изображение фона

На выходе получим синтезированное изображение

В качестве пространства параметров на котором будет выполнена оптимизация было выбрано: Параметры распределения для генерации параметров PP + пространство параметров аутистичей
(Так же сюда можно добавить и сами аутистичей, для выявления чрезмерно нереалистичных или совсем неэффективных

Изначально такой метрикой было полярное MSE при генерации батча изображений, с помощью ней даже получалось повысить разнообразие генерируемой выборки, но такая метрика никак не свидетельствует о реалистичности результатов генерации. Для оценки качества генерации выборка бат на две метрики

- Fréchet inception distance (FID)
- Inception score (IS)

- FID позволяет сравнить близость распределения сгенерированной выборки (распределение которое моделирует генератор) с распределением реальных данных
- JS можно интерпретировать как оценку репрезентативности и разнообразия генерации

Будем получать глубокие разбиения изображений из модели детекции обученной `model_ab` or [belikov-ansany](#) либо из модели сегментации `model_t` or [Tiniakov-A-O](#)

Получено 10.05.2012

- Пусть имеется датасет D (для модели эксперимента нам важно лишь что это разнотипный для детекции датасет изображений объектов с кодами на нем)
- Разобьем его (стратифицировано по типу кодов) на два равночисленных, непересекающихся датасета $D1$, $D2$
- Для $D1$ предсним боксы, и внутри этих боксов проведем генерацию кода, т.е. перекрем существующий реальный код сгенерированным
- $D2$ не меняется
- Получим збиддены $E1$ для $D1$ и $E2$ для $D2$ (т.е. $E1$ - множество полученных получением збиддены каждого измененного изображения в $D1$, $E2$ аналогично, с отпечатами лишь в том что $D2$ не был подвергнут изменению)
- Посчитаем FID на основе $E1$ и $E2$, посчитаем IS

Более сильный способ оценки:

- Как и в предыдущем методе из датасета D получим D1 и D2, также вставим в картинку D1 стеганые коды
- Обучим модель на D1 и стабилизируем на D2

В данном случае мы устраиваем классическую проверку обобщаемости модели на реальные данные при обучении на синтезированных.

Так же в последний способ можно добавить, не пилить лезвием и не изменять часть, которую в ПН приваривали!

Метод оптимизации

Будем использовать байесовскую оптимизацию через библиотеку Optuna, оптимизируемый функционал - FID на первой оценке, сам алгоритм оптимизации будет устроен так:

- Орбиты выбирает параметры для итерации
- С этими параметрами проводим процедуру первичной оценки (разбиение датасета заранее фиксировано)
- Оценка FID
- Оптимизационный шаг Орбиты

Мисellaneous results

Удалось собрать 4 публичных датасета:

1. <https://www.kaggle.com/datasets/jonathanimmarvel/barcode-and-gp/data>
2. <https://www.kaggle.com/datasets/whoois/barcode-dated-ion-annotated-dataset/data>
3. <https://www.kaggle.com/datasets/krisandrew/bu-goods-barcodes>
4. <https://www.kaggle.com/datasets/bcd2627/ideal-barcode-mask-alignment>

(1) – (3) – детекция, (4) – сегментация

Эксперименты

На данный момент от \mathcal{E} пришлось отказаться, нет понимания как поменять его для детекционных и сегментационных датасетов

Эксперименты с преданной оценкой и оптимизацией проведены на открытом датасете [3].

Примеры пенализации до-отпускных:



