

Ссылка на оригинальную статью: <https://arxiv.org/pdf/1511.06434.pdf>.

В статье предлагается методология для обучения сверточных GAN гораздо более стабильным способом.

Дискриминатор, который появляется после обучения DCGAN, может быть использован для задач классификации изображений.

Авторы показывают, что генератор может научиться рисовать конкретные объекты.

Также есть возможность манипулировать семантическими свойствами генерируемых изображений.

Архитектура DCGAN

Рекомендации по архитектуре для стабильных глубоких сверточных GAN

- Заменить все слои объединения на пошаговые свертки (дискриминатор) и свертки с дробным шагом (генератор).
- Использовать batchnorm как в генераторе, так и в дискриминаторе.
- Удалить полностью связанные скрытые слои для более глубокой архитектуры.
- Использовать активацию ReLU в генераторе для всех слоев, кроме выходного, который использует Tanh.
- Использовать активацию LeakyReLU в дискриминаторе для всех слоев.

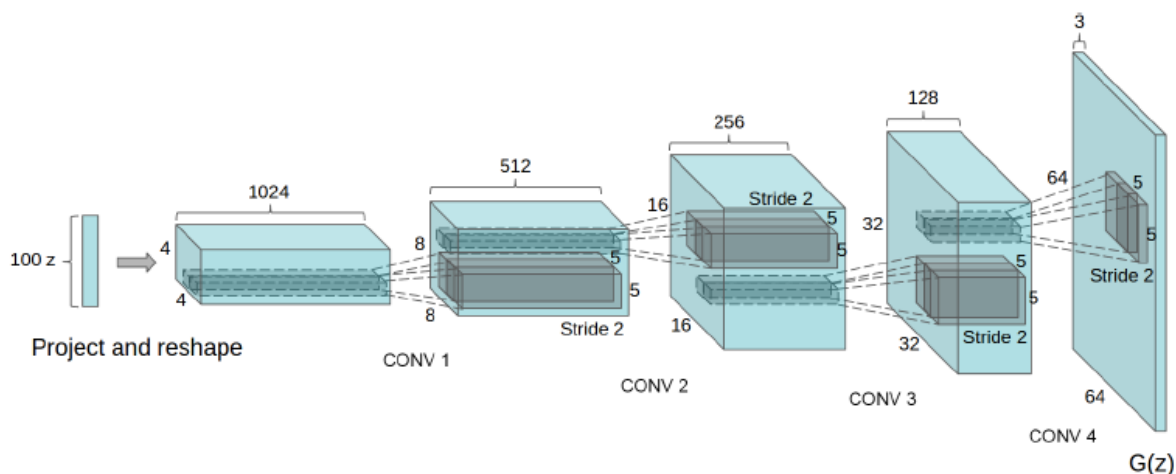


Рисунок показывает генераторную сеть архитектуры DCGAN. Можно ясно видеть, что нет полностью связанных слоев. Есть только сверточные слои в сети.

Давайте пройдемся по слоям шаг за шагом:

Во-первых, генератору дается 100-мерный вектор шума в качестве входных данных. После этого проецируются и изменяются входные данные.

Затем есть четыре операции свертки.

Каждый раз получается увеличение высоты и ширины. В то же время каналы продолжают сокращаться.

После первой операции свертки есть 512 выходных каналов. Это продолжает уменьшаться с каждой операцией свертки. После третьего канала количество выходных каналов составляет 128.

В итоге есть сгенерированное изображение размером 64 × 64 и три выходных канала.

За исключением первого слоя свертки, все остальные слои имеют шаг 2.

Детали обучения:

К обучающим изображениям не применялась никакая предварительная обработка, кроме масштабирования до диапазона функции активации \tanh $[-1, 1]$. Все модели были обучены с помощью мини-пакетного стохастического градиентного спуска (SGD) с размером мини-пакета 128. Все веса были инициализированы из нормального распределения с нулевым центром со стандартным отклонением 0,02. В модели LeakyReLU наклон утечки был установлен равным 0,2 во всех моделях. Использовался оптимизатор Adam с настроенными гиперпараметрами. Скорость обучения 0,001 была сочтена слишком высокой и вместо этого использовали 0,0002. Что помогло стабилизировать тренировку.

Эмпирическая возможность DCGANs

В наборе данных CIFAR-10 была продемонстрирована очень высокая базовая производительность благодаря хорошо настроенному однослойному конвейеру извлечения объектов, использующему K-means в качестве алгоритма изучения объектов.

Выводы и предложения

Предлагается более стабильный набор архитектур для обучения генеративных состязательных сетей и приводим доказательства того, что состязательные сети усваивают хорошие представления изображений для контролируемого обучения и генеративного моделирования. Все еще сохраняются некоторые формы нестабильности модели - что по мере того, как модели обучаются дольше, они иногда сворачивают подмножество фильтров в один колебательный режим. Необходима дальнейшая работа по борьбе с этой формой нестабильности. Считается, что расширение этой структуры на другие области, такие как видео (для прогнозирования кадров) и аудио (предварительно обученные функции для речи синтез) должен быть очень интересным. Дальнейшие исследования свойств изученного скрытого пространства также были бы интересны.