

1. Introduction

- Convolutional GANs를 제안하고 안정적으로 학습하는지 평가할 것이다
- 학습된 식별자를 이용해서 다른 비지도학습 알고리즘과 비교할 것이다
- GAN으로 학습된 필터를 시각화 하고, 특정한 물체를 그리도록 학습한 필터를 보여줄 것이다
- 생성자가 벡터 연산 특성을 갖고 있어 쉽게 조작하여 많은 의미론적 샘플을 생성할 수 있는 것을 보여줄 것이다

2. Related Work

Unlabeled data에 대한 대표적인 학습 방법으로는 K-means clustering, Auto-encoder, Ladder structures가 있다. 이러한 방법들은 이미지 픽셀 학습에서 좋은 학습을 수행했다.

이미지 생성모델은 두 가지가 있다. Parametric & Non-Parametric

Non-parametric은 texture synthesis, super-resolution, in-painting에 사용된다

Parametric 모델은 광범위하게 연구되었다. 하지만 아직까지 성공적이지 못하다. Variational sampling approach는 약간 성공적이었으나, 흐릿함 문제가 있었다. GAN 또한 노이즈와 이해할 수 없음의 문제가 있었다.

Deconvolutions와 maximal activations를 필터링함으로써, 각각의 convolution 필터의 대략적인 목적을 확인할 수 있다.

3. Approach and Model Architecture

CNNs를 사용한 GANS은 성공적이지 못했다. 광범위한 모델 탐구 이후 dataset에 대해 안정적인 학습과 고해상도와 깊은 생성 모델을 결과로 얻을 수 있었다.

중요한 접근 방법은 CNN구조를 3가지 바꾸는 것이다. 첫번째로 모든 convolutional net의 deterministic spatial pooling functions를 strided convolutions로 바꾸는 것이다. Strided convolutions는 신경망 자체의 spatial down sampling을 가능하게 한다. 두번째는 fully-connected layers를 제거하고, Global Average Pooling을 사용한다. GAP는 모델의 안정성을 증가시키지만 수렴속도를 느리게 한다. GAN의 첫번째 층은 matrix multiplication이다. 하지만 결과는 4차원 tensor

로 reshaped된다. 판별자의 마지막 convolutional layer는 flatten되고, single sigmoid output을 출력한다. 생성 모델을 확인하면 100차원의 균등 분포 Z 를 project와 reshape을 해서 $1024 \times 4 \times 4$ 로 만든 후 4개의 convolutional layer를 통해 연산을 수행한 후 $3 \times 64 \times 64$ 로 $G(Z)$ 를 만든다. 전결합층과 풀링층은 사용되지 않았다. 세번째는 입력을 normalize함으로써 학습을 안정시키는 Batchnormalization이다. 이는 initialize 문제와 gradient 흐름 문제에 도움을 준다. 하지만 모든 layers에 batchnormalization을 적용하면 샘플이 변도하고, 모델이 불안정해진다. 따라서 생성자의 출력층과 판별자의 입력층에서는 BN을 사용하지 않는다. 생성자의 출력층은 Tanh을 활성화 함수로 사용하고, 나머지는 모두 ReLU를 사용한다. 또한 유계인 활성화 함수가 학습속도가 빠른 것을 알 수 있었다. 특히 고해상도 모델링을 위해서 판별자에는 LeakyReLU를 사용한다.

4. Details of Adversarial Training

DCGANs을 세가지 dataset으로 학습시켰다. Large-scale Scene Understanding, Imagenet-1k, Faces(newly assembled). Pre-processing을 사용하지 하지 않았다. 모든 모델은 mini-batch SGD를 사용했다. Batch-size는 128이다. 모든 가중치는 $N(0, 0.2)$ 의 분포를 따르게 초기화 했다. LeakyReLU의 기울기는 0.2로 했다. Optimizer로 learning rate가 0.0002, momentum이 0.5인 Adam을 사용했다. 해당 파라미터를 수정한 이유는 학습의 안정화를 위해서이다.

이미지 생성 모델은 overfitting과 memorization of training samples 문제가 있다. DCGAN의 경우 5 epochs로 학습을 한 결과 해당 문제들로 이미지 샘플을 생성해 내는 것이 아니라는 것을 확인할 수 있다.

생성자가 입력을 memorizing하는 문제를 해결하기 위해 de-duplication을 진행했다. Autoencoder를 사용해서 중복 이미지를 제거했다. 이를 통해 precision을 높일 수 있었다.

Faces와 Imagenet-1k를 사용했고, data-augmentation을 사용하지 않았다

5. Empirical Validation of DCGANs Capabilities

GAN을 특징 추출로 CIFAR-10 data 분류를 수행했다. DCGAN의 성능을 평가하기 위해 Imagenet-1k로 학습을 시킨 후 판별자의 convolutional features를 이용했다. 각각의 층에 maxpooling을 이용해 4×4 spatial grid를 만들었다. Features를 flatten한 후 28672차원 벡터로 합치고 L2-SVM classifier로 학습되었다. 이는 82.8%의 accuracy를 보여주며 여러 K-means 모델들보다 좋은 성능을 보였다 하지만 기존 CNN의 성능은 보여주지 못했다.

SVHN data 분류에서 DCGAN + L2-SVM이 가장 좋은 성능을 보여줬다. 이는 같은 구조의 지도 학습 CNN보다 좋은 성능이었다. 이를 통해 DCGAN의 CNN구조가 모델 성능의 주요 요소가 아님을 검증한다.

6. Investigating and Visualizing the Internals of the Networks

Latent space가 급작스럽게 변화할 경우 memorization이라고 판단한다. 반면에 천천히 변화할

경우 이미지 생성에 의미론적 변화를 줄 수 있다.

Guided backpropagation을 통해 unsupervised DCGAN이 대량의 이미지 데이터를 학습해서 계층적 특징들을 학습할 수 있는 것을 확인했다,

판별자를 통해 학습된 특징에서 생성자가 어떤 특징을 학습할지 의문이 생겼다. 이를 위해서 생성자에서 창문을 제거한 후 실험을 수행했다. 창문이 없는 샘플을 통해 학습된 모델은 창문이 사라지거나 다른 물체로 대체된 이미지를 생성했다

글자의 벡터 연산과 같은 벡터 연산을 이미지에 수행했다.

7. Conclusion and Future Work

기존의 모델들보다 안정된 학습을 하는 모델을 만들었지만, 여전히 oscillating mode나 collapse 문제가 발생한다. 이를 해결하기 위한 노력이 필요하다

Reference

[논문읽기]07.DCGAN: <https://leechamin.tistory.com/222>

DCGAN 논문 이해하기:

<https://angrypark.github.io/generative%20models/paper%20review/DCGAN-paper-reading/>