Generative Adversarial Nets

Sai Vinay G
CE17B019
Indian Institute of Technology Madras
Chennai, India

February 6, 2019

Abstract

This report deals with the paper **Generative Adversarial Nets**. This report summarizes the paper and discuss its novelty and also go through a detailed critique and also providing with future possible improvements and areas of research.

1 Summary

1.1 Introduction

Deep generative models have had less of an impact, due to the difficulty of approximating many difficult probabilistic computations that arise in maximum likelihood estimation and related strategies, and due to difficulty of maximizing the benefits of piecewise linear units in the generative context. In the proposed paper, ”Generative Adversarial Nets” consists of two models : a generative model and a discriminative model. The entire framework resembles a two-player minimax game where the generator tries minimize the objective function whereas, the discriminator tries to maximize the objective function.

1.2 Adversarial nets

The goal of GANs is to train a Generator network G, which learns the data distribution $p_{data}(x)$ by transforming a prior input noise variables sampled from $p_z(z)$ as $G(z;\theta^{(G)}) = x$, where θ_g represents generator network parameters. The Discriminator network D, is trained to distinguish the samples generated from the real data $p_{data}(x)$ and the fake samples generated by generator network G. Discriminator $D(x;\theta_d)$ (where, θ_d represents discriminator model parameters) outputs a scalar which represents the probability that that x came from $p_{data}(x)$ rather than $p_g(x)$ (generated distribution). The Generator network is in turn trained to fool the discriminator by making the samples as real as possible.

Both generator and discriminator have separate loss functions to optimize. Generator is trained to minimize $log(1 - D(G(x))$ and discriminator is trained to maximize $log(D(x))$. An iterative approach of optimization to avoid overfitting discriminator D in which after every k steps of optimization of D one step of optimization of G is performed.

In other words, D and G play the following two-player minimax game with value function $V(G, D)$:

$$\underset{G}{min}\ \underset{D}{max} \quad V(G,\, D) = E_{x \sim p_{data}(x)}[logD(x)] + E_{z \sim p_z(z)}[log(1 - D(G(z)))]$$

The above equation does not provide sufficient gradient for G to learn as in the beginning as G is poor, D will reject samples with high confidence so, $log(1 - D(G(z)))$ saturates. So, instead of training G to minimize $log(1 - D(G(z)))$ it is better to train G to maximize $log(D(G(z)))$ which provides a much stronger gradients to learn at the start of training.

Theoretically it is shown that for a fixed generator there is a unique optimal discriminator, $D^*(x) = \dfrac{p_{data}(x)}{p_{data}(x) + p_g(x)}$ and generator G is optimal when $p_{data}(x) = p_g(x)$, which implies discriminator predicts 0.5 for all samples drawn from x.

1.3 Experiments

This paper experimented on MNIST, the Toronto Face Database (TFD) datasets with generator nets using a mixture of rectifier linear activations and sigmoid activations, while the discriminator net used maxout activations and dropout during training. On MNIST dataset Adversarial nets outperformed DBN, Stacked CAE and Deep GSN models. On TFD dataset it outperformed DBN, Deep GSN and was comparable with Stacked CAE.

2 Critique

2.1 Novelty of the paper

- Training GANs is done using backpropagation and dropouts algorithms overcoming the previous requirement of markov chains and approximate inferences for generative networks.
- The Adversarial learning introduced in this paper has gained a lot of importance and lead to lot of new research work and applications, like DCGAN, StackGAN, CyclicGAN, Pix2Pix and many more.
- The generator network learns to generate natural looking images from noise without even seeing atleast one natural image directly.

2.2 List of shortcomings of the paper

One of the major drawback of GANs is they are very hard to train GANs.

2.2.1 Convergence

- Training GANs requires finding a Nash equilibrium of a non-convex game with continuous, high-dimensional parameters. GANs are typically trained using gradient descent techniques that are designed to find a low value of a cost function, rather than to find the Nash equilibrium of a game (saddle point). So in finding a Nash equilibrium, these algorithms may fail to converge.

2.2.2 Vanishing gradients

- During training if the discriminator D is over-trained such that, it becomes very powerful and rejects most of the samples generated by the generator G, then generator's gradients vanish and the generator's weights doesn't learn much.

2.2.3 Mode collapse

- While training it is possible that generator G does not explore much and learns the representations of only few variants of samples i.e, only few modes. Due to this discriminator D will be convinced that generator G is able to produce natural images but, could not find out that generator is producing only few samples multiple times.

2.2.4 Deviation from true distribution

- As training involves simultaneous training two networks it is possible that discriminator may learn accept images that do not make any natural sense. By this the generator will not be able to learn the true data distribution.

2.2.5 Evaluating outputs

- While evaluating outputs produced by GANs as there is no universal metric to classify the generated outputs as realistic or not and to tell how realistic the output is.

2.3 Ways to extend and future directions of research

2.3.1 Generating new situations for simulations

- Real world consists of many unexpected and not so trivial situations which humans cannot think of while training or testing. Which can be tackled using GANs.

- In reinforcement, while training agents it may be possible to generate a completely new situations that the agent has never encountered. So, that it will be ready to tackle them when they occur next time in real world.

- While simulating any robot in virtual environments, we can generate many unexpected situations for the robot to tackle so that it doesn't fail in real world situations.

2.3.2 Generating stories

- It may be possible to learn representation of concepts of stories of say, crime stories and generate new concepts that relate to crime.

- From concept representation we can generate sentence of words which describe the story. It may also learn to produce twists in stories as everything is normal for the generator (from the learned representation) but for human beings cannot think of that event happening at that instant.

2.3.3 Improving training techniques

- One of the most important aspect that can be improved is the training procedure of GANs. As, we have seen above there are a lot of problems to deal with during the training process.

- Recently, many papers came up with various techniques to address the problems while training like feature matching, minibatch discrimination, virtual batch normalization, one-sided label smoothing and various other methods have been proposed for convergent GAN training.

2.3.4 Universal metric for evaluation

An efficient GAN evaluation measure should:

- Favor models that generate high fidelity samples (i.e, ability to distinguish generated samples from real ones; discriminability).

- Favor models that generate diverse samples(thus can tackle mode collapse, overfitting)

- Agree with human perceptual outputs.

2.3.5 Generating new data

- As it is difficult, expensive and time-consuming to get more data, GANs can be useful for producing new realistic data in such data-limited situations.

References

[1] A. Borji, "Pros and cons of gan evaluation measures," *arXiv preprint arXiv:1802.03446*, 2018.

[2] D. T. Chang, "Concept-oriented deep learning: Generative concept representations," *arXiv preprint arXiv:1811.06622*, 2018.

[3] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, "Generative adversarial networks: An overview," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 53–65, 2018.

[4] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[5] S. Hitawala, "Comparative study on generative adversarial networks," *arXiv preprint arXiv:1801.04271*, 2018.

[6] P. Manisha and S. Gujar, "Generative adversarial networks (gans): What it can generate and what it cannot?" *arXiv preprint arXiv:1804.00140*, 2018.

[7] M. J. Patel, M. S. Pandya, and P. V. Shah, "Review on generative adversarial networks," in *Impact Factor: 5.22 (SJIF-2017), e-ISSN: 2455-2585 Volume4, Issue 7, July-2018.*

[8] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," in *Advances in Neural Information Processing Systems*, 2016, pp. 2234–2242.