Critical Analysis 2

What Are Generative Adversarial Nets (GANs):

GANs were first proposed by (Goodfellow et al. 2014). Its design contains a generator working against an adversary, while the discriminator figures out how to tell if an example has a place the information circulation or from the generative network. The thought is for these two networks to improve while contending with one another. we can create some extremely sensible great examples from the networks by inserting a signal noise. The two models are trained at the same time, Generative model catches the data distribution. Discriminative model estimates the likelihood that an example came from the training data or the fake data.

Main trends since the first GAN:

There were many trends and ideas since GANs inception in 2014 that came to address many problems and be used in many domains. Here are the most notable GANs that have been built.

SRGAN by (Ledig et al. 2017) Applies a deep network with an adversary network to produce high resolution images. The key idea behind it is that the authors proposed a new high-resolution GAN in which they replace the loss of MSE-based material with the calculated loss at the VGG level. SRGAN uses a perceptual loss function that is a weighted sum of content loss and the adversarial loss.

DCGAN by (Radford et al. 2016) is one of the most popular and successful network architectures for GAN. It mainly incorporates layers of convolution with no maximum pooling or fully connected layers. it's a direct extension of GAN. The detector consists of alternation layers, batch rate layers and LeakyReLU activation. The generator consists of consolidated transposed layers, batch norm layers and ReLU activities.

(Zhu et al. 2020) proposed a method called CycleGAN to capture the properties of an image domain and how these properties can be translated to another image domain, all in the absence of any paired training images. CycleGAN is an extension of the GAN architecture that involves the simultaneous training of two generator models and two differentiation models. A generator takes images from the first domain as input and outputs images for the second domain, and a second generator takes images from the second domain as input and produces images for the first domain. CycleGAN also uses an additional architecture extension called cycle consistency.

StyleGAN is a state-of-the-art method by (Karras et al. 2020) proposed a new generator architecture for GAN, which allows you to control the fine details (for example, the shape of the eye) of the generated samples. The StyleGAN generator and descriptor model are trained using a progressive development GAN learning method (Karras et al. 2018). This means that both models start with small images. The models are adjusted until they are stable, then the wiggle and generator are doubled in width and height. A new block is added to each model to support the size of the larger image, which gradually fades during training. After fading into, the models are retrained until they are sufficiently stable, and the process is repeated with larger images until the desired target image size is reached.

Main problems solved or improvements over the original work:

The ability of MSEs (and PSNRs) to capture related differences, such as textual detail, is very limited because it is determined by pixel-wise image differences. SRGAN authors defined a novel perceptual loss using high-level feature maps of the VGG network connected to a discriminator that perceptually encourages solutions to distinguish from HR reference images. (Ledig et al. 2017)

DCGAN known from a family of architectures, which have received stable training in a wide variety of datasets and allow high resolution and deeper generative models to be trained. The first is the all-convolutional net with a "strided" convolution. Batch normalization that stabilizes learning by normalizing the input. (Radford et al. 2016)

CycleGAN tries to keep the adversarial losses matching the distribution of images generated in the data distribution in the target domain; And loss of stability of the cycle to prevent learned mappings G and F from contrasting with each other. (Zhu et al. 2020)

StyleGAN aims to overcome the limitations of traditional GANs, in which control for individual features of data such as facial features in a photograph is not possible. In addition, it allows for the variability factor in the images generated by adding "style" to the images at each compromise level.(Karras et al. 2020)

Remaining problems from the published works so far:

Training GAN is not easy. Most GAN models may suffer the following problems in one from or other here we discuss the most notable problem that the above methods suffer.

During training, the generator can be reduced to a setting that always produces the same output. This is a common failure case for GAN and is commonly referred to as mode collapse (Salehi et al. 2020). The generator may be able to trick the corresponding discriminator, but it cannot learn to represent the complex distribution of real-world data and gets stuck in a small space with very little diversity. However, the authors of WGAN (Arjovsky et al. 2017) claimed to solve this problem however there isn't any empirical evidence to justify the claim.

The next problem in forming a GAN is the failure of convergence (Salehi et al. 2020). This happens when the GAN cannot drive the generator and discriminator to an acceptable level of performance. This means that the generator will generate a bad image and the discriminator will be easily identifiable. Therefore, the loss of the discriminator goes to zero very quickly while the loss of the generator remains the same or continues to increase steadily over a period.

Interesting problems to solve and why:

I would be personally interested in solving the Mode Collapse problem because most GAN model suffer from it. If I was doing GAN as a research topic, I would build a base GAN architecture that would inherently counter the mode collapse problem by stopping the Generator network from producing the same outputs.

Bibliography:

Arjovsky, M., Chintala, S. & Bottou, L., 2017. Wasserstein GAN.

Goodfellow, I.J. et al., 2014. Generative Adversarial Networks.

Karras, T. et al., 2020. Analyzing and Improving the Image Quality of StyleGAN.

Karras, T. et al., 2018. Progressive Growing of GANs for Improved Quality, Stability, and Variation.

Ledig, C. et al., 2017. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network.

Radford, A., Metz, L. & Chintala, S., 2016. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks.

Salehi, P., Chalechale, A. & Taghizadeh, M., 2020. Generative Adversarial Networks (GANs): An Overview of Theoretical Model, Evaluation Metrics, and Recent Developments.

Zhu, J.-Y. et al., 2020. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks.