Semester: VIII Subject: Advanced AI Academic Year:2024-2025

Module 2

Challenges/Problems in GANs:

GANs can be difficult to train due to:

- **Training instability –** Balancing the generator and discriminator is tricky.
- **Mode collapse –** The generator may produce only a few types of outputs instead of diverse samples.
- **Vanishing Gradient**
- **High computational cost –** Requires powerful hardware for training.

1. Training Instability:

GAN training is highly sensitive to hyperparameters, architecture choices, and initialization. Small changes in these factors can lead to unstable training dynamics, such as oscillations or divergence. Additionally, the discriminator and generator networks may become unbalanced during training, leading to one network overpowering the other.

During training, the discriminator aims to maximize the objective, while the generator aims to minimize it. Instability may arise if one network's updates dominate the training process, preventing the other network from learning effectively. Training instability can be observed as the discriminator or generator loss oscillating or diverging instead of converging to a stable equilibrium.

2. Mode collapse:

Usually you want your GAN to produce a wide variety of outputs. You want, for example, a different face for every random input to your face generator.

However, if a generator produces an especially plausible output, the generator may learn to produce *only* that output. In fact, the generator is always trying to find the one output that seems most plausible to the discriminator.

If the generator starts producing the same output (or a small set of outputs) over and over again, the discriminator's best strategy is to learn to always reject that output. But if the next generation of discriminator gets stuck in a local minimum and doesn't find the best strategy, then it's too easy for the next generator iteration to find the most plausibleoutput for the current discriminator.

Each iteration of generator over-optimizes for a particular discriminator, and the discriminator never manages to learn its way out of the trap. As a result the generators rotate through a small set of output types. This form of GAN failure is called **mode collapse**.

3. Vanishing gradient:

GAN training uses a min-max optimization procedure in which the generator and discriminator compete against one another. Vanishing gradients occur when the gradients used to update the parameters of either the generator or discriminator become very small during training. This problem might occur, particularly during the early training, making it difficult for the generator to learn properly. This might cause slow convergence or even stagnation in the learning process.

4. Computational resources:

Training state-of-the-art GAN models often requires considerable computing resources, such as high-performance GPUs or TPUs and massive datasets. Many academics and researchers find the computational cost of training GANs too expensive, restricting their broad adoption and exploration.

Mitigate these challenges:

1. By increasing the complexity of the GAN architecture to better capture the data distribution.
2. By using Wasserstein GANs (WGANs) or WGAN with gradient penalty (WGAN-GP) to stabilize training and address vanishing gradient issues.
3. By limiting the magnitude of gradients during training to avoid them becoming too small.
4. By using cross-validation to assess the performance of various hyperparameter configurations on validation data.
5. By experimenting with network architectures and regularization approaches to stabilize training.
6. By ensuring that samples from all data modes are distributed evenly to modify your sampling techniques or loss functions.