BOĞAZİÇİ UNIVERSITY
COMPUTER ENGINEERING

CMPE 492

REPORT

WEEK 1

BARAN DENIZ KORKMAZ
DOĞUKAN KALKAN

FALL 2020

1 Weekly Summary

1.1 Work Done

- We implemented two different GAN architectures for generating handwritten digits using MNIST dataset:

    1. In the first version, we have a more complex architecture than the latter.

       GitHub: `https://github.com/barandenizkorkmaz/DeepFake-Detection/blob/master/Deliverables/Week%201/GAN%20for%20MNIST%20Handwritten%20Digits/v1/GAN%20for%20MNIST%20Handwritten%20Digits%20v1.ipynb`

       Reference: `https://machinelearningmastery.com/how-to-develop-a-generative-adversarial-network-for-an-mnist-handwritten-digits-from-scratch-in-keras/`

    2. In the second version, we have a rather simpler architecture compared to the first version.

       GitHub: `https://github.com/barandenizkorkmaz/DeepFake-Detection/blob/master/Deliverables/Week%201/GAN%20for%20MNIST%20Handwritten%20Digits/v2/GAN%20for%20MNIST%20Handwritten%20Digits%20v2.ipynb`

       Reference: GANs in Action, Jakub Langr, Vladimir Bok, 2019

1.2 Learning Outcomes

- **Introduction to Keras:** In the field of Deep Learning, there are several libraries that we can work with to carry out our projects. We chose to use TensorFlow library and Keras which acts as an interface for TensorFlow due to its wide usage and resources from which we can benefit during the process.

- **Introduction to GAN Implementation:**

    1. **Discriminator Architecture:** Discriminator is a component of a GAN architecture, which, as the name suggests, tries to distinguish between real inputs and fake inputs and also is supposed to improve itself as the training continues. It receives real examples from the training dataset and fake examples coming from the generator. The goal of the discriminator is to predict a probability that the input is real.

    2. **Generator Architecture:** Generator is the second component of a GAN architecture, which produces fake inputs that are supposed to be as convincing as possible. In other words, the goal of the generator is to generate fake data that is indistinguishable from the real data. As the training continues, the generator must improve itself

to generate fake data that is more realistic in each step and render the discriminator unable to distinguish between fake data and real data. The generator improves itself through the feedback it receives from the discriminator's classifications.

- **Evaluating a GAN Training:** At the beginning of the training, we expect the generator to generate fake examples that are easily recognized by the discriminator, which suggests that the discriminator is supposed to have 100% accuracy. However, as the training continues, the generator learns through the feedback coming from the discriminator and improves itself. As a result, the generator starts to produce fake examples that are now more realistic. Ideally, we do not want the discriminator to recognize the fake examples coming from the generator. This ideal situation is achieved if the discriminator's accuracy is 50%, which means the discriminator is now at a point where it cannot distinguish between real examples and fake examples. The discriminator can randomly guess whether the example is real or fake. At this point, GAN is said to have converged. However, in practice, this is impossible to achieve. Instead, it is better to find suboptimal convergence points.

- **Requirement for GPU:** As explained above, at the beginning of the training, the generator produces easily recognizable fake examples. This means that we have to train the model for a certain period of time and training complex GANs may require powerful hardware. Since we want the generator to improve itself, we have to have a powerful GPU on which we can run our network.

2 Challenges

- Building a development environment for implementation in Anaconda which provides Jupyter Notebook. Installation of packages used can vary due to operating systems and the version of development tools used.

- As we cannot increase the number of iterations spent during training, we were unable to analyze the digits generated and the influence of GAN architecture on learning.

3 Questions

- Suggestion for a Cloud GPU provider for deep learning with a free student access.

4 What's Next?: Upcoming Week

Below, the associates present the objectives for the next week. We also state the provisional amount of time required to finish the tasks.

- Decision about the most appropriate GPU provider - 1 day
- Reading the articles introducing public datasets and deciding on the ones which suit best at the initial stage - 1 day