UDACITY

< Back to Deep Learning Nanodegree

Generate Faces

审阅

代码审阅

HISTORY

Requires Changes

还需满足 2 个要求 变化

This is Good attempt, great job👍
Please check the comments above and you will get better results🐱

Keep Learning🌼🌼🌼

项目中是否有需要的文件、相关函数是否通过了单元测试

该项目提交应包含项目中名为 “dlnd_face_generation.ipynb” 的 .ipynb 文件。

项目中所有的单元测试都已通过。

All the unit tests in project have passed, Good job👍

建立神经网络

`model_inputs` 函数被正确实现。

Correctly implemented the `model_inputs` function, Good job👍

`discriminator` 函数被正确实现。

Correctly implemented the `discriminator` function, good job👍

- Good use of Batch Normalisation and Leaky Relu in the CNN which will improve efficiency of discriminator.😺
- Batch Normalization which stabilizes learning by normalizing the input to each unit to have zero mean and unit variance. This helps deal with training problems that arise due to poor initialization and helps gradient flow in deeper models. This proved critical to get deep generators to begin learning, preventing the generator from collapsing all samples to a single point which is a common failure mode observed in GANs. Directly applying batchnorm to all layers however, resulted in sample oscillation and model instability. This was avoided by not applying batchnorm to the generator output layer and the discriminator input layer. Here you can do some updates of the discriminator function. 😺
- For a better performance of GAN, you can read this: Tips and tricks to make GANs work😺

`generator` 函数被正确实现。

🌼Correctly implemented the `generator` function, good job👍

`model_loss` 函数被正确实现。

Good job! You have implemented `model_loss` correctly, very good that you tried to smooth the input labels👍
This article created a simple GAN network from scratch and explained well about GAN theory, also have a detailed explanation about the g_loss and d_loss, I would suggest you read it to get a detailed understanding about GAN network if you have any questions about the loss function in GAN or GAN network structures.

`model_opt` 函数被正确实现。

The function `model_opt` is implemented correctly, good job on trying `tf.control_dependencies` here 👍

神经网络训练

训练函数被正确实现。

- 它应使用 `model_inputs`、`model_loss` 及 `model_opt` 创建了模型。
- 同时它使用 `show_generator_output` 函数展示了 `generator` 的输出。

Tanh is used as the last layer of the Generator output, meaning the generated images by Generator will be between -1 and 1. But currently, batch_images lies between -0.5 to 0.5, so we would need to rescale the true images to the scale between -1 to 1 too. we can use batch_images = batch_images*2 to normalize the input to Discriminator.
Other than that, you have done great work here combining all the parts together and making it a Generative Adversarial Network.

这些超参数被设置到了合理的数值。

The parameters sets are reasonable.

- Please note that the hyperparameters for the MNIST dataset and for the faces set do not need to be the same. Since images are different, with different features and even more, with more color channels, the parameters must be tuned for each case.
- GANs are very sensitive to hyperparameters. A lot of experimentation goes into finding the best hyperparameters such that the generator and discriminator don't overpower each other. You can read the DCGAN paper to see what worked for them and have a better understanding about GANs network.
- Batch size influences the quality of image generated by GAN, for celebA dataset, as it contains large images Batch size of 16 or 32 would be good, for MNIST dataset, 28*28 black and white images Batch size of 32 or 64 would be good. In GAN, learning rate of 0.0002 is preferred. Beta equal to or less than 0.5 is preferred. You can tweak them for better results.
- Here you can also try to apply exponential_decay to the `learning_rate`.
  If you want to learn more about hyperparameters, these are some great resources on the topic:
  Deep Learning book - chapter 11.4: Selecting Hyperparameters.
  Neural Networks and Deep Learning book - Chapter 3: How to choose a neural network's hyper-parameters? .
  Efficient BackProp (pdf).

项目代码生成的面部图像比较逼真，至少看上去像是一些人脸。

The generated images are close to realistic faces now, try to tune with the parameters, to see how far we can push🐱

重新提交

下载项目

重新提交项目的最佳做法

Ben 与你分享修改和重新提交的 5 个有益的小贴士。

观看视频 (3:01)

返回 PATH

学员 FAQ