CpSc 8430: Deep Learning

Homework 3

Copy Right Notice

- **Most slides in this presentation are adopted from slides of text book and various sources. The Copyright belong to the original authors. Thanks!**

HW3

- Train a discriminator/generator pair on CIFAR10 dataset utilizing techniques from DCGAN, Wasserstein GANs, ACGAN.

- DCGAN: https://arxiv.org/abs/1511.06434
- WGAN: https://arxiv.org/abs/1701.07875
- ACGAN: https://arxiv.org/abs/1610.09585

Baseline Model for DCGAN

- Generator
  - noise_input = (100,);
  - text_input = (119,);
  - # num of (hair, eyes) pairs
  - text_emb = Dense(256,'relu')(text_input);
  - concatenate([noise_input, text_emb]);
  - Dense(4*4*512); Reshape((4, 4, 512));
  - Batchnorm(mom=0.9); Relu;
  - Conv2DTranspose(256, kernel=5);
  - Batchnorm(mom=0.9); Relu;
  - Conv2DTranspose(128, kernel=5);
  - Batchnorm(mom=0.9); Relu;
  - Conv2DTranspose(64, kernel=5);
  - Batchnorm(mom=0.9); Relu;
  - Conv2DTranspose(3, kernel=5);
  - Tanh;
- Training
  - Adam(lr = 0.0002, beta = 0.5)

- Discriminator
  - image_input = (64,64,3);
  - text_input = (119,);
  - text_emb = Dense(256,'relu')(text_input);
  - text_emb = Reshape((1,1,256))(text_emb);
  - tiled_emb = tile(text_emb, [1,4,4,1]);
  - Conv2D(64 ,kernel=5)(image_input); LeakyRelu;
  - Conv2D(128, kernel=5);
  - Batchnorm(mom=0.9); LeakyRelu;
  - Conv2D(256, kernel=5);
  - Batchnorm(mom=0.9); LeakyReLu;
  - Conv2D(512, kernel=5);
  - Batchnorm(mom=0.9);
  - image_feat = LeakyRelu;
  - concatenate([image_feat, tiled_emb]);
  - Conv2D(512, kernel=1, strides=(1,1));
  - Flatten;
  - Dense(1, 'sigmoid');

Data

- Dataset :
  - CIFAR 10: https://www.cs.toronto.edu/~kriz/cifar.html

Submission & Rules

- Deadline: April. 20th 23:59
- Allow package：
  - python 3
  - **TensorFlow/pytorch ONLY** for CS and ECE student
  - For non-CS/ECE students, Keras is allowed.

Submission & Rules

- For HW3 :
  - **Submit a report with 1) 10 best generated pictures. 2) performance comparison among DCGAN, WGAN and ACGAN**
    - **Train networks from scratch and compare to existing networks**
    - **Working on DCGAN and WGAN is the basic and working on ACGAN is a bonus.**
  - **Submit your code to github.**