Data Science HW#4

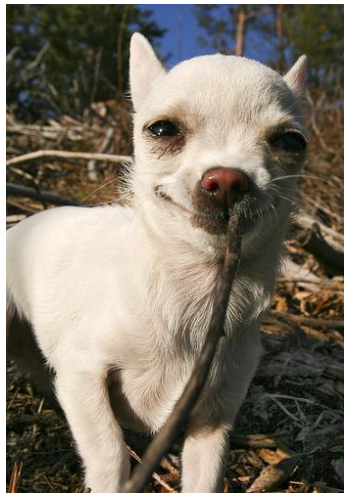
Deadline: 2021/1/5 23:59

Goals

- Generate dog images from Gaussian Noise
- Train GAN from scratch
- Evaluate generated images

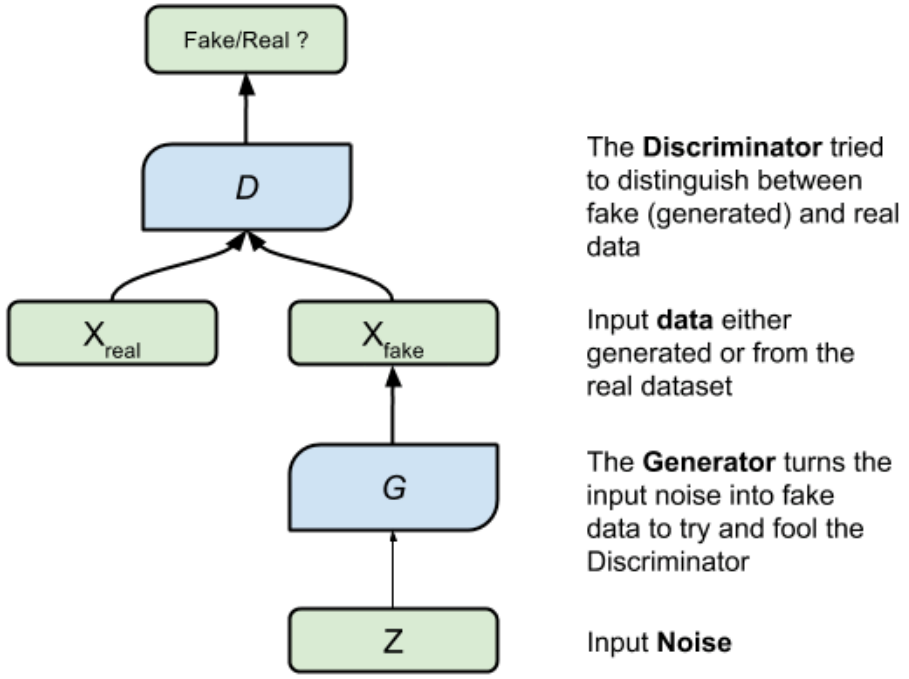
Dataset Description

- Dog images with 120 breeds
- 20850 RGB images with bounding box annotations, about 750MB after unzip
- Two folders : images/ and annotations/




Specification - Input

- Train generative models to create images of dogs without ground truth data
- Unsupervised training
  - Images are the only inputs for model.
  - External datasets are forbidden.
  - Other attributes are forbidden.
- Unconditional model
  - “Noises” are the only inputs to generator.



Specification - Evaluation

- Fréchet Inception distance (FID), which is also known as Wasserstein-2 distance
- Use Inception v3 to produce feature vectors for real images and generated images.
- Compute the Fréchet distance between Gaussian $p \sim \mathcal{N}(\mu_r, \Sigma_r)$ from real images and Gaussian $q \sim \mathcal{N}(\mu_g, \Sigma_g)$ from generated images as the "Fréchet Inception distance ".

$$FID(p, q) = ||\mu_r - \mu_g||_2^2 + Tr(\Sigma_r + \Sigma_g - 2(\Sigma_r \Sigma_g)^{1/2})$$

Specification - Developing

- Option 1 : follow the example codes
  - Download **example.zip** from e3. It already includes the images/ and annotations/. Please follow the instructions in README.txt.
  - The example codes are developed in python3 with Pytorch.
  - The requirements.txt in example.zip is based on CUDA 10.0.130 and CuDNN 7.6.3.
  - You can change torch/torchvision versions according to your CUDA/CuDNN versions. (Please handle the potential dependency issues.)

Specification - Developing

- Option 2: develop your own codes
  - Download **dataset.zip** from e3, which contains images/ and annotations/.
  - Download **example.zip** from e3 for evaluation. Please follow the instructions in README.txt.
  - Please use python3 for this homework.
  - You can use other frameworks such as Tensorflow or Keras.

  NOTE: we recommend you check example.zip to format your codes.

Specification - Developing

- You are only allowed to use general modules in deep learning frameworks to build model, which means you can not use packages from paper publications or other organizations (ex.. DCGAN from TF-GAN and DCGANGenerator from pytorch-gan).
- You can reference the model from the internet but should write by yourself.
- Do not generate images for others (otherwise both of you get 0 point).

Grading Policy

- Over the baseline (FID < 270): 80 points
- Top 2/3: 90 points
- Top 1/3: 100 points

Submission Format

- Please submit two zip files to e3 (< 500MB in total)
  - {studentID}_img.zip
    - After unzip the file, there should be a single folder named {studentID}_img containing 10000 generated 64*64 RGB images.
  - {studentID}_src.zip
    - After unzip the file, there should be a single folder named {studentID}_src containing all needed codes including run.sh and requirements.txt.

  NOTE: don’t include dataset in {studentID}_src.zip

Submission Format

- TAs will check your codes with following steps:

    1. unzip {studentID}_src.zip
    2. cd {studentID}_src
    3. pip install -r requirements.txt
    4. ./run.sh {TA_data_path}

TA_data_path/
- images/
- annotations/

NOTE: ./run.sh should generate images at {studentID}_src/result/images_inference

- If the number of submitted images is not equal to 10000 or the evaluation of codes have fatal problem, you will get 0 point in this homework.

Submission Format

- TAs will check your code on following environment:
  - OS: Ubuntu 16.04 LTS
  - Python version: python3.6
  - RAM: 32G
  - GPU: GTX 1070 (8G)

Hints

- Please use GPUs and start this homework as early as possible. The training could take many hours.
- It is not necessary to record FID scores along with the training process. You could directly judge the results from the sampled generated images by eyes.
- You can use less images to compute FID scores in developing, which could save some time.

Contacts

- If you have any question about HW#4, please post it on the e3 forum. TAs will respond to it as early as possible.
- If your question is not public-related, please email to TAs:
  - 宋韻筑 happy19960302@gmail.com
  - 陳義瑄 yisyuan.chen.ece@gmail.com

Appendices

- Structure of example codes
  - data/ : contains images and annotations
  - eval/ : contains codes for FID scores
  - result/ : contains saved models and generated images. (after inference)
  - data_loader.py : processes raw dataset
  - gan.py : main file
  - run.sh : generates images at *result/images_inference* from scratch
  - requirements.txt : required packages
  - README.txt : descriptions for the codes