ADL HW4 Report

B04705026 資管四 林彥廷

- Network Structure & Loss Terms

Generator							
Operation	Kernel	Strides	Feature maps	Batch Norm?	Spectral Norm?	Dropout	Nonlinearity
Noise Input: 100 x 1 x 1 Condition Label Input: 6, 4, 3, 2							
Label Embedding	Embed 4 categories into 4 x 16. Embedding Matrix is initialized from std Normal and fixed.						
Concat	Concat noise with embed labels						
Transposed Convolution	4 x 4	1 x 1	1024	v	x	0.5	ReLU
Concat	Concat up-sampled (8 x 8) embed labels on channel dimension						
Transposed Convolution	4 x 4	2 x 2	512	v	x	0.5	ReLU
Concat	Concat up-sampled (16 x 16) embed labels on channel dimension						
Transposed Convolution	4 x 4	2 x 2	256	v	x	0	ReLU
Concat	Concat up-sampled (32 x 32) embed labels on channel dimension						
Transposed Convolution	4 x 4	2 x 2	128	v	x	0.5	ReLU
Transposed Convolution	4 x 4	2 x 2	64	x	x	0	Tanh
Image Output: 3 x 128 x 128							

Discriminator							
Operation	Kernel	Strides	Feature maps	Batch Norm?	Spectral Norm?	Dropout	Nonlinearity
Image input: 3 x 128 x 128							
Convolution	4 x 4	2 x 2	32	x	v	0	LeakyReLU
Convolution	4 x 4	2 x 2	64	x	v	0	LeakyReLU
Convolution	4 x 4	2 x 2	128	x	v	0	LeakyReLU
Convolution	4 x 4	2 x 2	256	x	v	0	LeakyReLU
Convolution	4 x 4	2 x 2	512	x	v	0	LeakyReLU
Convolution	4 x 4	1 x 1	512	x	v	0	LeakyReLU
Flatten	Flatten feature maps (512 x 1 x 1) into tensor with 512 dims.						
Adversarial Head							
Concat	Concat one-hot labels with tensor						
Linear	n/a	n/a	1	x	v	0	x
Auxiliary Head x 4							
Linear	n/a	n/a	# classes	x	v	0	Softmax

Loss Terms	
P_g , P_d	Generator Distribution, Dataset Distribution
D_a , D_h , D_e , D_f , D_g	Discriminator with Adversarial, Hair, Eye, Face, Glass Heads
Generator: WGAN Gen	
Adversarial Loss	$-E_{\hat{x}\sim P_g}[D_a(\hat{x})]$

Discriminator: WGAN Dis + Classification Loss	
Adversarial Loss	$-E_{x\sim P_d}[D_a(x)] + E_{\hat{x}\sim P_g}[D_a(\hat{x})]$
Auxiliary Loss - Hair	$-E_{x\sim P_d}\left[log\left(D_h(x)\right)\right] - E_{\hat{x}\sim P_g}\left[log\left(D_h(\hat{x})\right)\right]$
Auxiliary Loss - Eye	$-E_{x\sim P_d}[log(D_e(x))] \; -E_{\hat{x}\sim P_g}[log(D_e(\hat{x}))]$
Auxiliary Loss - Face	$-E_{x\sim P_d}\left[log\left(D_f(x)\right)\right] - E_{\hat{x}\sim P_g}\left[log\left(D_f(\hat{x})\right)\right]$
Auxiliary Loss - Glass	$-E_{x\sim P_d}\left[log\left(D_g(x)\right)\right] - E_{\hat{x}\sim P_g}\left[log\left(D_g(\hat{x})\right)\right]$
Total Discriminator Loss	λ_a Adversarial Loss - Discriminator $+\lambda_h$ Auxiliary Loss - Hair $+\lambda_e$ Auxiliary Loss - Eye $+\lambda_f$ Auxiliary Loss - Face $+\lambda_g$ Auxiliary Loss - Glass
$\left(\lambda_a,\ \lambda_h,\ \lambda_e,\ \lambda_f,\ \lambda_g\right)$	$(0.01,\ 0.2,\ 0.2,\ 0.2,\ 0.2)$

Hyperparameters	
Optimizer	Adam($\beta_1 = 0$, $\beta_2 = 0.9$)
Learning Rate	1e-4
# critics (train iter ratio of Dis:Gen)	3
Batch Size	64
Leaky ReLU slope	0.2
(Transposed) Convolution Weight, bias initialization	Gaussian(mean=0, std=0.02), Constant(0)
Label smoothing	Valid label ~ Uni(0.7, 1.3) Fake label ~ Uni(0, 0.3)

Training Progress

Please refer to the GIF on ***HW4/model/n_a3/train_progress.gif***

Comparison of different experiments

For the sake of fairness, all FID Scores are calculated with model checkpoints at 260k steps. Hidden concat, BN, WGAN models are exactly the same as Q1 model.

1. ***Hidden embed label concat to generator*** vs ***Only input layer***

   *Please see the training progress GIF of **Hidden concat** Loss and **Only Input** Loss models at respectively at "HW4/figure/n_a3.gif" and "HW4/figure/noinpute.gif".*

	Hidden concat	Only input
FID Score	80.017	84.017

   - **Design logic:**
     Generator structure focus more on label.

   - Model Difference
     - Each label is mapped to a fix continuous embedding and concat the feature maps along depth dimension, which followed the ideas from How to Train a GAN? cGANs with Projection Discriminator BigGAN

     - Hidden concat into intermediate layer of generator forced generator not ignoring the label with the model going deeper. (Embedded labels are up-sampled to match feature map size.)

     - Only concat embed labels with random noises intuitively is inferior to hidden concat since generator might take longer time to learn the importance of the label in the very top of the model.

- Experiments Result - Image Quality
  - Image qualities of these two model are mostly equivalent, except that *Only Input* model still generated 混色爆炸頭 and face with weird glasses after 100k iterations.

- Experiments Result - Conditional Correctness
  - Compared to *Hidden Concat Model, Only Input* model struggled to produce clear hair. It tended to generate mixed color and one-sided glasses. That might be drawbacks of less information about the label.

2. ***Batch Norm.*** vs ***Conditional Batch Norm*** *in generator.*

*Please see the training progress GIF of **BN** Loss and **Cond BN** Loss models at respectively at "HW4/figure/n_a3.gif" and "HW4/figure/condbn.gif".*

	BN	Cond. BN
FID Score	80.017	84.472

- **Design logic:**
  Output distribution of generator may affected by given labels.

- Model Difference
  - Conditional Batch Norm is introduced in this paper solving VQA problem.

    My personal idea is that *distribution of 144 different classes may be significant different since each hair and face are comprised of distinguishable color.*

- Experiments Result - Image Quality
  - Background clearness of *Cond BN* model is better.
  - Face quality of two model is indistinguishable.

- Experiments Result - Conditional Correctness

- Hair and face color of some faces under fixed conditions were totally incorrect throughout the whole training process.

  I think it was caused by the *Cond BN* in the beginning of training phase where the generator usually produced faces with wrong label but the **conditional batch norm** memorized the distribution of those incorrect faces.

  Since we have 144 combinations of faces, it is seldom to have two or more face of same labels in single batch (size=64) to make up for the incorrect faces.

3. ***WGAN Loss*** vs ***BCE Loss***

*Please see the training progress GIF of **WGAN** Loss and **BCE** Loss models at respectively at "HW4/figure/n_a3.gif" and "HW4/figure/bce.gif".*

	WGAN	BCE
FID Score	80.017	139.838

- Model Difference
  - Adversarial loss and loss weight are different.
  - In WGAN model $\left(\lambda_a,\ \lambda_h,\ \lambda_e,\ \lambda_f,\ \lambda_g\right) = (0.01,\ 0.2,\ 0.2,\ 0.2,\ 0.2)$
  - In BCE model $\left(\lambda_a,\ \lambda_h,\ \lambda_e,\ \lambda_f,\ \lambda_g\right) = (0.2,\ 0.2,\ 0.2,\ 0.2,\ 0.2)$
  - The model structures are exactly the same in WGAN and BCE model.

- Experiments Result - Image Quality
  - Images generated by BCE model are much more **"gray"** in the first 50k steps.
    There are many random noise in the background.
  - The problem of Mode collapse is more severe in *BCE model*.

- Experiments Result - Conditional Correctness

- Only ~50% labels are correct in *BCE* model, and the weights between auxiliary loss and bce loss make little difference in the correctness of labels.