Text-to-Image-to-Text Translation Using Cycle Consistent Adversarial Networks

Satya and Jeremy

University of Toronto

4th May, 2018

Introduction

- Text-to-Image synthesis is a challenging problem that has a lot of room for improvement considering the current state-of-the-art results.
- Synthesized images from existing methods give a rough sketch of the described image but fail to capture the true essence of what the text describes.
- A lot of work has been put into making generating high quality images in existing methods but we found that images generated are still not accurate to the ground truth text descriptions given as input to the network.

Related Work

- Conditional generative adversarial nets (Mirza et al., 2014) - Learning approximate distribution of data by conditioning on an input.
- Generative adversarial text to image synthesis (Reed et al., 2016)
- StackGAN: StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks (Zhang et al., 2017) - Generates high resolution images by stacking many GANs.
- CycleGAN: Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks (Zhu et al., 2017)

Overall Architecture

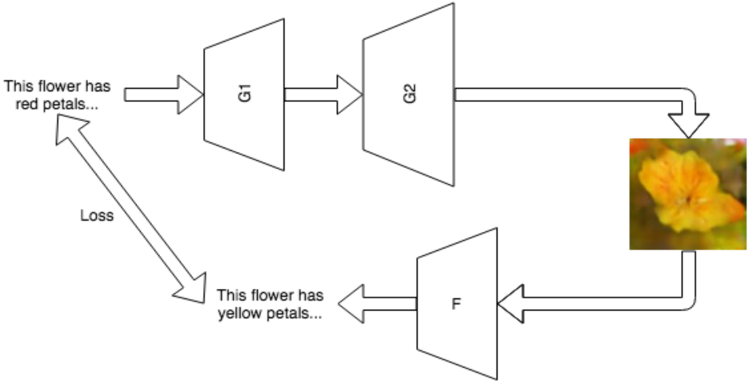

This flower has red petals...
G1
G2
Loss
This flower has yellow petals...
F

Figure 1: Figure illustrating the overall architecture of our model.

G1, G2 are the stage 1 and stage 2 generator. F is the caption generator. We take the difference between the original caption and the real caption as the loss.

Overview

- We first break the image synthesis network into two stages similar to a StackGAN architecture (Zhang et al., 2017).
- This consists of generating a low resolution 64x64 image in the first stage and feeding this as input to the next stage's generator.
- The next stage refines the low resolution input further and generates higher quality image with 128x128 resolution.

- We then implement an image captioning GAN similar to (Dai et al., 2017).
- This network generates high quality captions based on images.
- We finally observe the difference in the ground truth captions and captions generated by our image captioning network. This provides a good signal for optimizing the image synthesis network further to generate good images that represent the text descriptions well.

Text to Image Translation

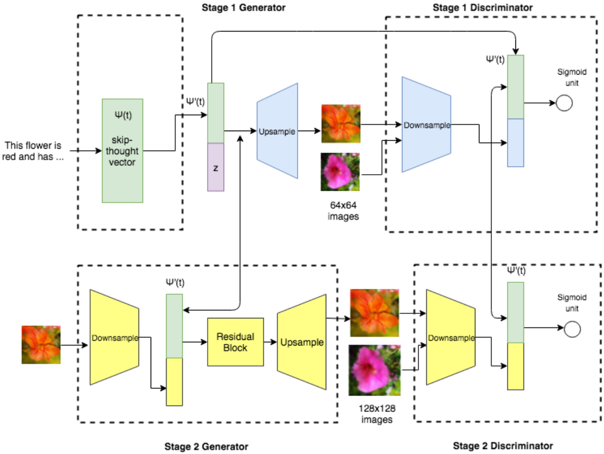


Figure 2: Text to Image GAN network. The text embedding and noise is given as input to the first stage. The output of the first stage is given as input to the next stage that produces higher resolution images. Generators from each stage have corresponding discriminators

Text to Image Translation

- We use Skip-Thought Vectors (Kiros et al. 2015) to generate a fixed size embedding for the text input.
- Our Stage-1 GAN takes embedding $\psi(t)$ and noise z as input and produces a 64x64 image.
- Our Stage-2 GAN takes 64x64 image generated from Stage-1 and refines it further to give a higher quality 128x128 image.

Training Text-to-Image Network

- We train the Stage-1 GAN by maximizing $\mathcal{L}_{D1}$ and minimizing $\mathcal{L}_{G1}$ given in equations 1 and 2.

$$\begin{aligned} \mathcal{L}_{D1} = \mathbb{E}_{(I_1,\psi(t))\sim p_{data}}[\log D_1(I_1, \psi(t)]+ \\ \mathbb{E}_{I_1\sim G_1,\psi(t)\sim p_{data}}[\log(1 - D_1(I_1, \psi(t))] \end{aligned} \tag{1}$$

$$\mathcal{L}_{G1} = \mathbb{E}_{z\sim\mathcal{N}(0,1),\psi(t)\sim p_{data}}[\log(1 - D_1(G_1(z, \psi(t)))] \tag{2}$$

- Interpolation loss

$$\mathcal{L}_{INT} = \mathbb{E}_{(\psi(t1),\psi(t2))\sim p_{data}}[\log(1 - D_1(G(z, \beta\psi(t1) + (1 - \beta\psi(t2)] \tag{3}$$

- Therefore we have to total loss $\mathcal{L}_{G1-Total}$ for training the generator in Stage-1 given by equation 4.

$$\mathcal{L}_{G1-Total} = \mathcal{L}_{G1} + \lambda\mathcal{L}_{INT} \tag{4}$$

- We train the Stage-2 GAN by maximizing $\mathcal{L}_{D2}$ and minimizing $\mathcal{L}_{G2}$ given in equations 5 and 6.

$$\begin{aligned}\mathcal{L}_{D2} = \mathbb{E}_{(I_2,\psi(t))\sim p_{data}}[\log D_2(I_2, \psi(t)]+ \\ \mathbb{E}_{I_2\sim G_2,\psi(t)\sim p_{data}}[\log(1 - D_2(I_2, \psi(t))]\end{aligned} \tag{5}$$

$$\mathcal{L}_{G2} = \mathbb{E}_{z\sim\mathcal{N}(0,1),\psi(t)\sim p_{data}}[\log(1 - D_2(G_2(z, \psi(t)))] \tag{6}$$

Image to Text translation

- We then built an image captioning network that produces captions based on images generated from the image synthesizing GAN. The generated captions can reinforce cycle consistency. Our work is based on Dai et al., 2017

Image to Text translation

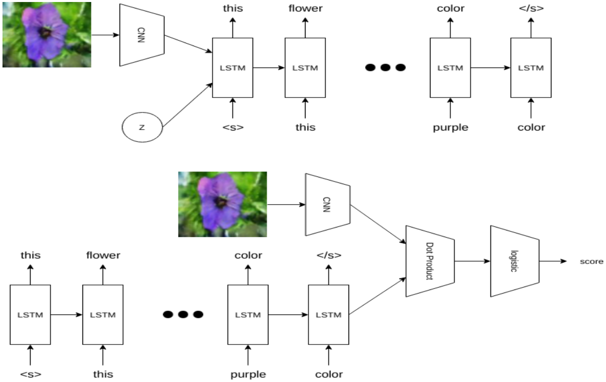


Figure 3: Caption GAN network. The top row shows the caption generator where the LSTM takes CNN features and noise Z as input and outputs captions. The bottom row shows the discriminator that performs a dot product on the CNN features of the image and the LSTM output.

- Discriminator training

$$\begin{aligned}\mathbb{E}_{S\in S_I}[\log r_\eta(I,S)] + \alpha\cdot\mathbb{E}_{S\in S_G}[\log(1-r_\eta(I,S))]+ \\ \beta\cdot\mathbb{E}_{S\in S_{\setminus I}}[\log(1-r_\eta(I,S))]\end{aligned} \tag{7}$$

- Generator training using REINFORCE

$$\mathbb{E}[\sum_{t=1}^{T_{max}} \operatorname*{arg\,max}_{w_t\in V} \pi_\theta(w_t|I,S_{1:t-1})\cdot r_\eta(I,S)] \tag{8}$$

Cycle Consistency

- To improve the results of the image synthesizing network further, we can exploit using the law of transitivity by introducing cycle consistency
- Zhu et al., 2017, use cycle consistency by introducing two additional loss terms, forward cycle loss and backward cycle loss.
- We add additional loss terms penalize the network parameters if they cannot reconstruct the original image by using the law of transitivity, i.e $F(G(x)) \approx x$ and $G(F(y)) \approx y$.
- Since we are primarily interested in improving the results of the image synthesizing network, we only use forward cycle loss to reconstruct the original caption back from the generated image.

- We represent forward cycle loss as $\mathcal{L}_{fcycle}$ and is defined in the equation 9.

$$\mathcal{L}_{fcycle} = -\sum_{t=0}^{T-1} \log p_t(w_t|I) \tag{9}$$

- Where $p_t(w_t|I)$ is the probability of observing the correct word w_t generated by the LSTM in the captioning network given an image $I = G(z, \psi(t))$ generated by the image synthesis network's generator.

Dataset

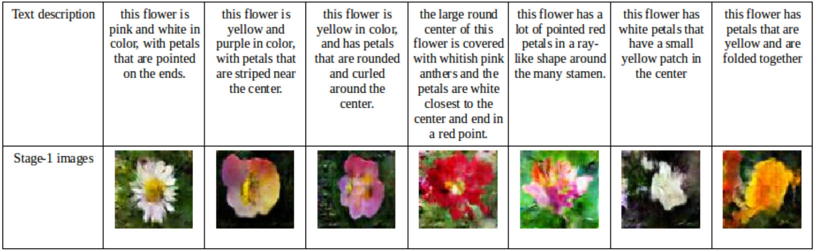
- We primarily use the Oxford VGG 102 Flower Dataset [1] for our experiments. This dataset contains 102 categories of flowers. Each category contains images of flowers between 40 and 248, with 8,189 images in total.
- For the flowers dataset, we use 5 captions per image. This gives us 5 image, caption pairs for every image present in the dataset. The images provided don't have a fixed dimension. We resize the images to 64x64 and 128x128 resolutions to be used by our network.

[1] http://www.robots.ox.ac.uk/ vgg/data/flowers/102/index.html

Experiments

- We first pretrain the image synthesis GAN network for 100 epochs, and pretrain the image captioning GAN network for 100 epochs. We then combine the two and train the whole network end to end by optimizing the objective $\mathcal{L}_{total}$ which includes the cycle loss for another 40 epochs.

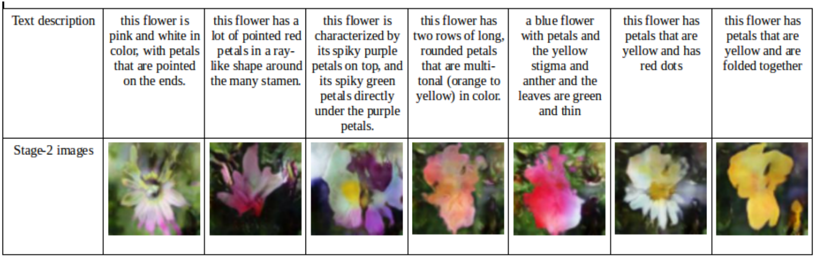
Text to Image Stage-1 GAN Results



Text description	this flower is pink and white in color, with petals that are pointed on the ends.	this flower is yellow and purple in color, with petals that are striped near the center.	this flower is yellow in color, and has petals that are rounded and curled around the center.	the large round center of this flower is covered with whitish pink anthers and the petals are white closest to the center and end in a red point.	this flower has a lot of pointed red petals in a ray-like shape around the many stamen.	this flower has white petals that have a small yellow patch in the center	this flower has petals that are yellow and are folded together
Stage-1 images							

Figure 4: Stage 1 GAN results which are 64x64 images

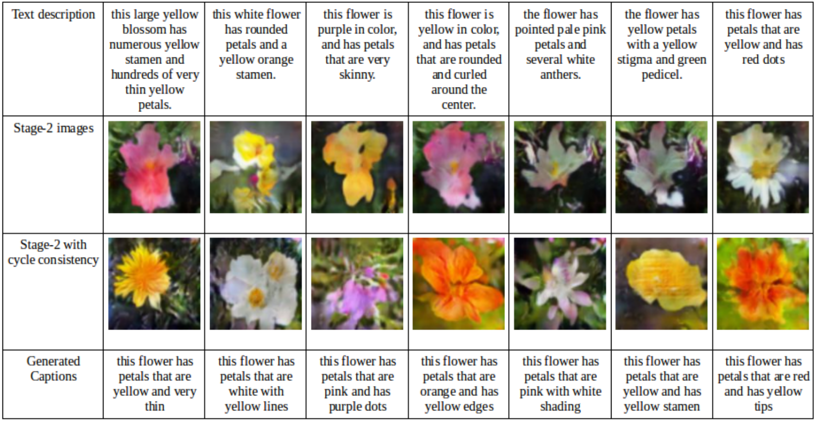
Stage-2 GAN Results



Text description	this flower is pink and white in color, with petals that are pointed on the ends.	this flower has a lot of pointed red petals in a ray-like shape around the many stamen.	this flower is characterized by its spiky purple petals on top, and its spiky green petals directly under the purple petals.	this flower has two rows of long, rounded petals that are multi-tonal (orange to yellow) in color.	a blue flower with petals and the yellow stigma and anther and the leaves are green and thin	this flower has petals that are yellow and has red dots	this flower has petals that are yellow and are folded together
Stage-2 images							

Figure 5: Stage 2 GAN results which are 128x128 images

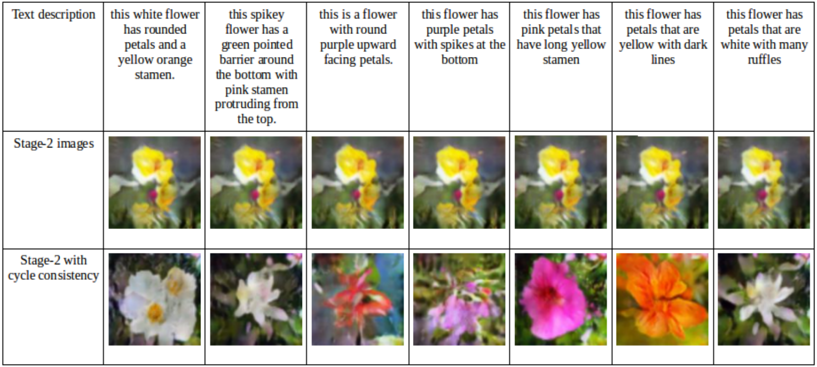
Cycle Consistency Results



Text description	this large yellow blossom has numerous yellow stamen and hundreds of very thin yellow petals.	this white flower has rounded petals and a yellow orange stamen.	this flower is purple in color, and has petals that are very skinny.	this flower is yellow in color, and has petals that are rounded and curled around the center.	the flower has pointed pale pink petals and several white anthers.	the flower has yellow petals with a yellow stigma and green pedicel.	this flower has petals that are yellow and has red dots
Stage-2 images							
Stage-2 with cycle consistency							
Generated Captions	this flower has petals that are yellow and very thin	this flower has petals that are white with yellow lines	this flower has petals that are pink and has purple dots	this flower has petals that are orange and has yellow edges	this flower has petals that are pink with white shading	this flower has petals that are yellow and has yellow stamen	this flower has petals that are red and has yellow tips

Figure 6: First row contains ground truth text descriptions, next two rows contain images generated by the GAN trained without cycle loss and trained with cycle loss respectively. The last row contains captions generated by the captioning network

Mode Collapse Reduction



Text description	this white flower has rounded petals and a yellow orange stamen.	this spikey flower has a green pointed barrier around the bottom with pink stamen protruding from the top.	this is a flower with round purple upward facing petals.	this flower has purple petals with spikes at the bottom	this flower has pink petals that have long yellow stamen	this flower has petals that are yellow with dark lines	this flower has petals that are white with many ruffles
Stage-2 images							
Stage-2 with cycle consistency							

Figure 7: Figure illustrating mode collapse. The first row contains ground truth text descriptions. The next two rows contain images generated by the GAN trained without cycle loss and with cycle loss respectively.

Freezing Caption Generator's Weights

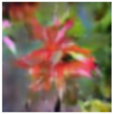
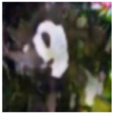
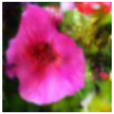
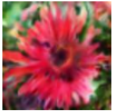
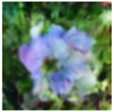
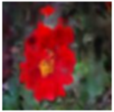
Text description	there are many long and narrow floppy pink petals surrounding many red stamen and a green stigma on this flower.	the flower shown here has blue petals which surround the white stamen	this is a flower with round purple upward facing petals.
Stage-2 with cycle consistency			
Stage-2 with cycle consistency with frozen caption generator network			

Figure 8: Results of GAN trained with cycle loss with image captioning network's weights frozen

Inception Score

Inception Score		
Model	Mean	Standard Deviation
GAN without Cycle Loss	2.985	0.163
GAN with Cycle Loss	2.545	0.067

Table 1: Inceptions scores comparing GAN trained with cycle loss and trained without cycle loss

Color Relevance score

- To analyze the effectiveness of our method further, we manually find the ratio of number of colors that are present in each image to all the colors that were mentioned in the caption.
- We average this number over 30 random images generated by our GAN model trained with cycle consistency and trained without cycle consistency.

Model	Color relevance score
GAN without Cycle Loss	0.259
GAN with Cycle Loss	0.802

Table 2: Image color relevance score for the two models

Future Work

- In future, we aim to generate higher quality images and test on complicated datasets such as MS-COCO.
- We also think think there is good potential in comparing the semantic meaning of text using fixed length embeddings measuring the distance in cycle loss.

Q & A