Text to Image GAN (SynthGan)
(on COCO Car images)

Nikhil Khullar, Narendra Kumar Vankayala, and Sai Supreeth

1. Problem Statement:

We aim at generating images of cars based on the text input (in Natural language) by the user. Based on features described by the text, the model (SynthGan) will generate car images such that user is not able to differentiate it as fake or real. For the text to image conversion, we will use Generative Adversarial Networks(GAN).

2. Dataset:

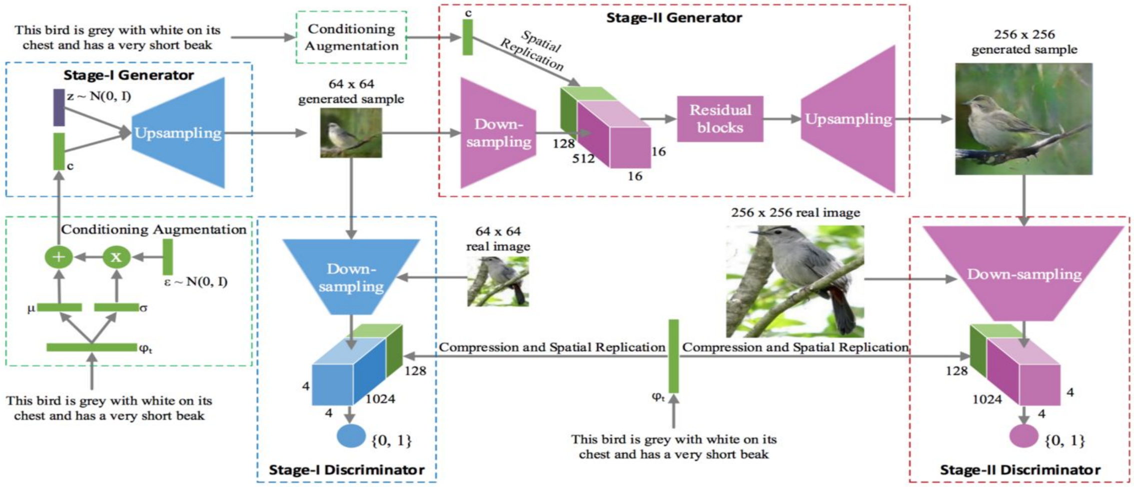
The dataset we are going to use is COCO dataset. It is a captioned image dataset provided by Microsoft. http://cocodataset.org/#home. To download this dataset we need to JSON files). We are following tutorial https://www.youtube.com/watch?v=h6s61a_pqfM.

Key areas of JSON files are-

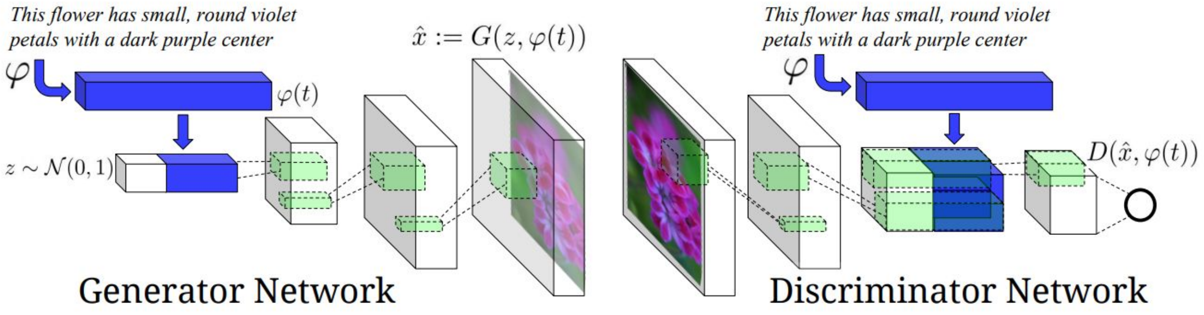
1) Images - Name and Image Id of image(s) we want to download.
2) Annotations - Number of segments for each image to be downloaded.
3) Category - Super category of each image to be downloaded.

3. Literature Survey:

Generative Adversarial networks consist of generator G and a Discriminator D that compete in a two min-max player game. The discriminator tries to distinguish real train data from synthetic images and the generator tries to fool the discriminator.


This bird is grey with white on its chest and has a very short beak
Conditioning Augmentation
Stage-II Generator
c
Spatial Replication
256 x 256 generated sample
Stage-I Generator
z ~ N(0, I)
Upsampling
c
64 x 64 generated sample
Down-sampling
128
512
16
16
Residual blocks
Upsampling
Conditioning Augmentation
+
x
ε ~ N(0, I)
μ
σ
φ_t
This bird is grey with white on its chest and has a very short beak
Down-sampling
64 x 64 real image
256 x 256 real image
Down-sampling
Compression and Spatial Replication
Compression and Spatial Replication
128
4
1024
4
{0, 1}
φ_t
This bird is grey with white on its chest and has a very short beak
128
1024
4
4
{0, 1}
Stage-I Discriminator
Stage-II Discriminator

StackGAN: The above architecture of StackGAN is there is a Stage-I generator draws a low-resolution image by sketching rough shape and basic colors of the object from the given text and giving the background color from a random noise vector. The Stage-II generator generates a high-resolution image with photo-realistic details by conditioning on both the Stage-I and the text again.


This flower has small, round violet petals with a dark purple center
φ
$\varphi(t)$
$z \sim \mathcal{N}(0,1)$
$\hat{x} := G(z, \varphi(t))$
This flower has small, round violet petals with a dark purple center
φ
$D(\hat{x}, \varphi(t))$
Generator Network
Discriminator Network

The above network is trained on top of Deep Convolutional Generative Adversarial Network (DC-GAN) conditioned on text features encoded by the hybrid character level convolutional recurrent neural network. Both the generator network G and the discriminator network D perform feed-forward inference conditioned on the text feature.

The problem can be subdivided into learning a text features representation that can capture the important visual details, and then using these features to synthesize a compelling image. The issue, however, that there can many plausible configurations of pixels that correctly illustrate the description of the text.

4. Approach Overview based on Our Preliminary Study:

Firstly, we will scrap the images and description of the image from the COCO image dataset. Preprocess text i.e. convert text to word embedding. We will perform image segmentation on the images and then map the features from the text to the image. Then we will train the GAN and Variational encoder on this data to generate images of cars based on the text entered by the user. We will then compare the results from both models and report our findings.

5. Team Member:

Name	Student ID
Nikhil Khullar	801053861
Sai Supreeth Segu	801075915
Narendra Kumar Vankayala	801081957

6. Project Timeline and Team Member Role:

	Task	Team Member	Timeline
1	Preprocessing and Scrapping COCO dataset for Cars	Sai Supreeth	03/12/2019
2	Implementing Image Synthesis GAN	Nikhil, Narendra, and Sai	03/20/2019
3	Optimizing Image Synthesis GAN	Nikhil, Narendra, and Sai	03/27/209
4	Model - Variational Autoencoder	Nikhil	04/10/2019
5	Optimizing Variational Autoencoder	Narendra	04/17/2019
6	Comparing results from SynthGAN and Autoencoder for COCO (Evaluation)	Narendra, and Sai	04/24/2019
7	Finalizing report and POSTER	Narendra, Nikhil and Sai	04/28/2019
8	OPTIONAL - Model StackGAN V2 and evaluation (if time persists)	Sai, Nikhil and Narendra	04/10/2019 to 30/10/2019

7. Questions This Project Will Answer:

- It will generate different cars based on features given by the user.
- Do we need to use object segmentation for this problem?
- If we don't use will that affect our output?
- How important data preprocessing is for any NLP task?
- How to do scraping data from a website?

8. Things That We Expect to Learn:

So far we have learned translator capabilities of machine learning models such as convert image to text (using agents), with this project we intend to learn a generative aspect of machine learning.

- Data preprocessing and cleaning of raw text data.
- Using and understanding of Generative Adversarial Networks algorithms.
- Performing image segmentation on the given input.
- Feature matching from input text to image.

9. Is our idea is novel?

- Most of the projects that have used GAN's up to now for text to image Synthesis have applied on flowers and birds domain.
- We are applying this algorithm on cars domain.
- Variational autoencoders are also generative models but since they try to minimize root-mean-square errors images are blurry. For example, A slight moment in a eye can cause a change in RMSE and hence affecting output. (Lars Mesheder https://arxiv.org/pdf/1701.04722.pdf)
- Traditionally the visual information of the object is captured into attributes (vectors) which are fed to Conditional GAN thus requiring domain expertise. Our project uses Natural languages to represent general object features, in addition to attributes for discriminating objects of a similar type is intuitive.
- We are trying to utilize Text to Image Synthesis GAN (SynthGAN) and StackGAN for COCO dataset for cars and will try to compare them with Variational autoencoders.

10. Reference (Survey on the area):

1. https://arxiv.org/abs/1605.05396
2. https://arxiv.org/abs/1511.06434
3. https://arxiv.org/abs/1711.10485
4. https://arxiv.org/pdf/1701.04722.pdf
5. https://arxiv.org/pdf/1612.03242.pdf