Text to Image Generation

160050012-Piyush Onkar
170010036-Manoj Bhadu

Team 48

Problem Statement

- **Input**: Textual description of Flower

- **Output**: Image of Flower generated as described by text

- **Motivation** :

  Previously studied deep generative models of images often defined distributions that were restricted to being either unconditioned or conditioned on classification labels. In real world applications, however, images rarely appear in isolation as they are often accompanied by unstructured textual descriptions, such as on web pages and in books.

References

Paper:

https://arxiv.org/pdf/1711.10485.pdf (Attention with BI-LSTM +GAN)

https://arxiv.org/pdf/1511.02793.pdf (Attention With RNN + GAN)

Code Reference :

https://github.com/paarthneekhara/text-to-image

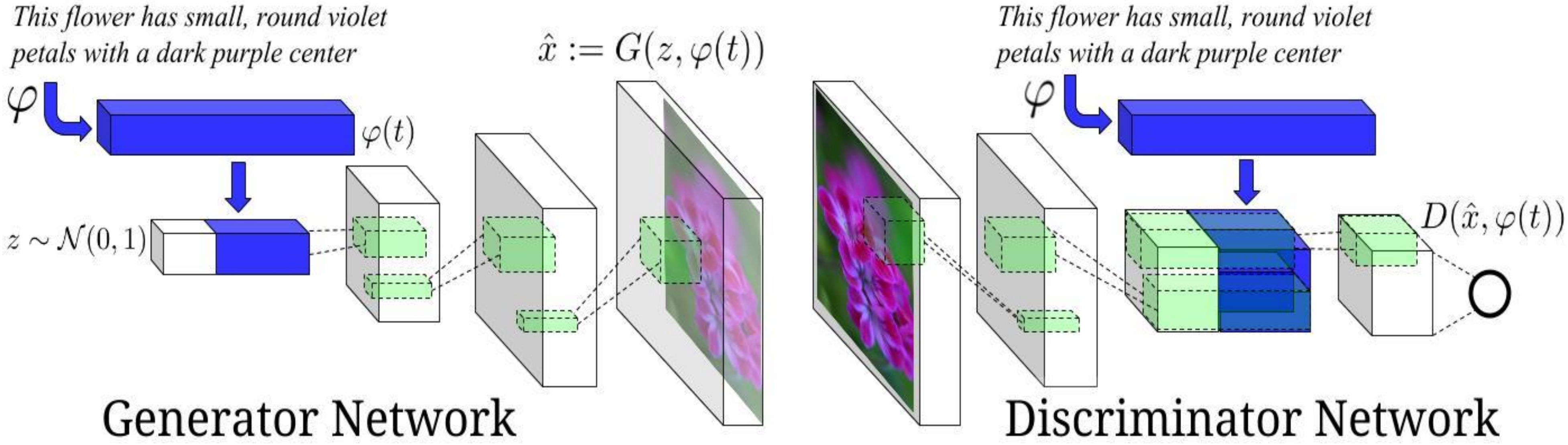
Data

- Description : Dataset contains 102 Flower categories with minimum 40 images. Each image contain around 15 text captions

- URL :
  1. Images
  2. Captions

Technique used

- We have Used Deep Learning Techniques
- Our Project is combination of Computer Vision and NLP
- Our Architecture:

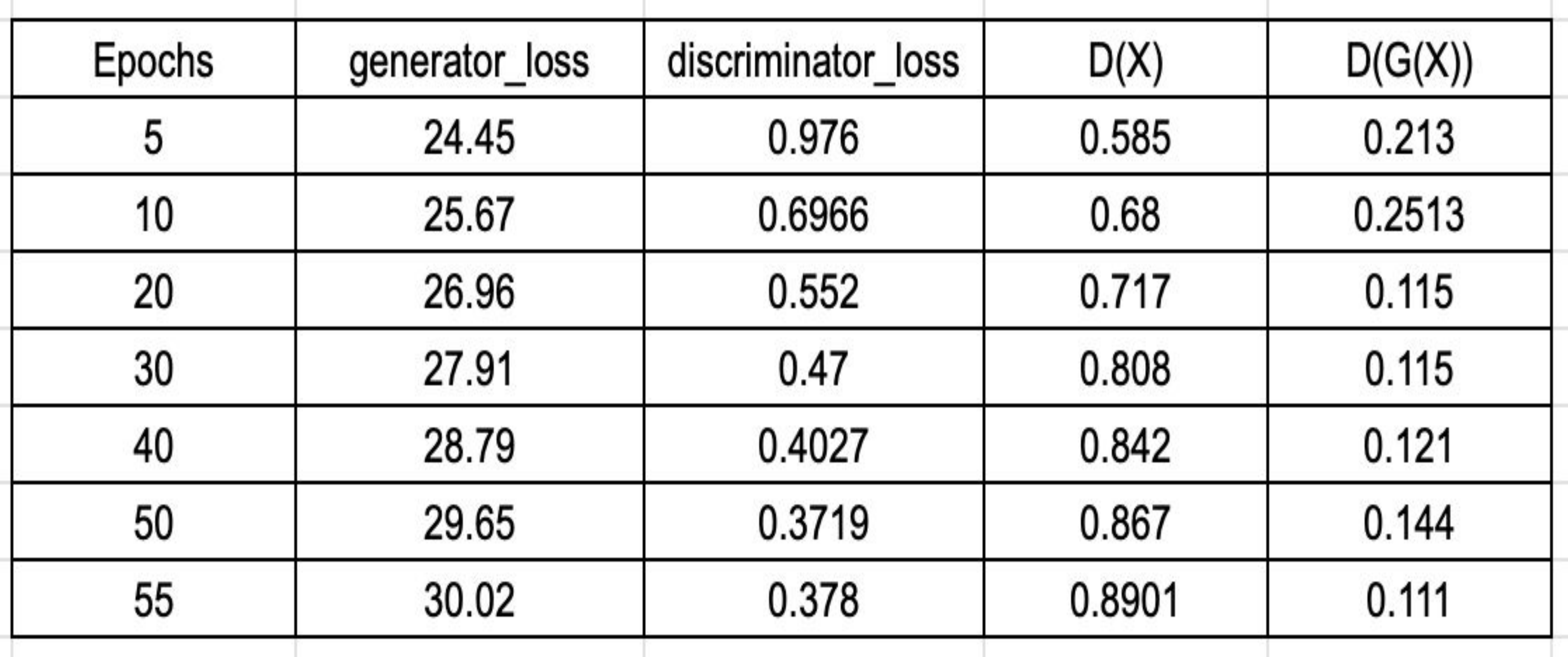

This flower has small, round violet petals with a dark purple center
φ
$\varphi(t)$
$z \sim \mathcal{N}(0,1)$
$\hat{x} := G(z, \varphi(t))$
Generator Network
This flower has small, round violet petals with a dark purple center
φ
$D(\hat{x}, \varphi(t))$
Discriminator Network

- In this network the embeddings(psi) are the sentence embeddings using transformers

Results

- Metrics used :
  Consider x to be an text and X to be corresponding image for text.
  We have used discriminator_loss, generator_loss, accuracy for discriminator given real images, accuracy for discriminator given fake images. The expected discriminator_loss should be less than 0.7, generator_loss should be more less than 20, accuracy for discriminator given real images and accuracy for discriminator given fake images should be around 0.5.

Results



Epochs	generator_loss	discriminator_loss	D(X)	D(G(X))
5	24.45	0.976	0.585	0.213
10	25.67	0.6966	0.68	0.2513
20	26.96	0.552	0.717	0.115
30	27.91	0.47	0.808	0.115
40	28.79	0.4027	0.842	0.121
50	29.65	0.3719	0.867	0.144
55	30.02	0.378	0.8901	0.111

Demo and Case Study

The image formed is very much blurry as it is trained on less number of epochs. After epoch 1, the image had almost all pixels black in colour. After epoch 55, there were some pixels having different colours, but still the major colour shown in the image was black in colour.

Conclusion and Future Work

- We have trained our model for only around 50 epoch due to unavailability of GPU. Most of the above mentioned papers have trained the model for 300-500 epoch for good quality images and time taken for that is around 2-3 days.
- Even though for 50 epoch we got discriminator loss in desired range but generator loss is high from desired range.
- Direction of future work is to train the some layers of transformer for better generalization of text according to our Data