Text To Image Generation

Drishya Uniyal
IIIT Delhi
MT21119

Prashant Sharma
IIIT Delhi
MT21227

Abstract

Text-to-image generation is an emerging field with significant applications in various fields, like e-commerce, advertising, and entertainment. In text-to-image generation, a textual description is used as input to generate an image that corresponds to the description. The problem of text-to-image generation has been studied extensively in recent years. (Mansimov et al., 2015)

1 Introduction

Text-to-image generation is a challenging task that involves generating realistic images from textual descriptions. In recent years, generative models based on deep learning have shown promising results. One such model is Generative Adversarial Networks (GANs) and their variations, such as AttnGAN, StackGAN, and BigGAN. These models use a two-part system consisting of a generator and a discriminator to learn the mapping between textual descriptions and corresponding images. The generator learns to produce images that are visually similar to the real images, while the discriminator distinguishes between the generated images and real images. Stable diffusion models are another text-to-image generation approach that uses a series of diffusions to generate realistic images. These models provide a new avenue for generating high-quality images from textual descriptions. Overall, text-to-image generation using deep learning models is a rapidly evolving field with great potential to generate realistic images from textual descriptions. In this work, we have translated text prompted by the user directly into image pixels. For example," The petals of this flower are pink, and the anther is yellow."

2 Problem Statement

Given an input text, generate images from that text. For example," The petals of this flower are pink, and the anther is yellow."

3 Related work and existing baselines

One of the first initial works in the domain of text-to-image generation was proposed by Mansimov et al. in 2015 (Mansimov et al., 2015). The model used a deep recurrent neural network (RNN) to encode the textual description into a fixed-length vector, which was then used as input to a deep convolutional generative adversarial network (DCGAN) to generate the corresponding image. The model used attention mechanisms to generate fine-grained details in the image. Several related works have been proposed in the past few years to improve the performance of text-to-image generation models. One such work is the StackGAN model proposed by Zhang et al. in 2017. The StackGAN model used a two-stage process to generate images from textual descriptions. In the first stage, a low-resolution image was generated from the textual description, which was then used as input for the second stage to generate a high-resolution image. The model used a novel conditioning augmentation technique to improve the diversity of the generated images. (Zhang et al., 2017). Another related work is the AttnGAN model proposed by Xu et al. in 2018. The AttnGAN model used an attentional generative adversarial network (GAN) to generate fine-grained and diverse images from textual descriptions. The model used a hierarchical architecture to generate images at different scales and an attention mechanism to generate fine-grained details. (Xu et al., 2018) The paper proposes a method called GigaGAN that generates high-quality and diverse images using a hierarchical GAN approach and a feature-matching loss function. The method outperforms existing GAN-based methods and produces visually appealing and diverse images.(Wah et al., 2011) The paper proposes DALL·E 2, a model for generating high-quality images from textual descriptions using a combination of transformer-based language models and GANs. It introduces a

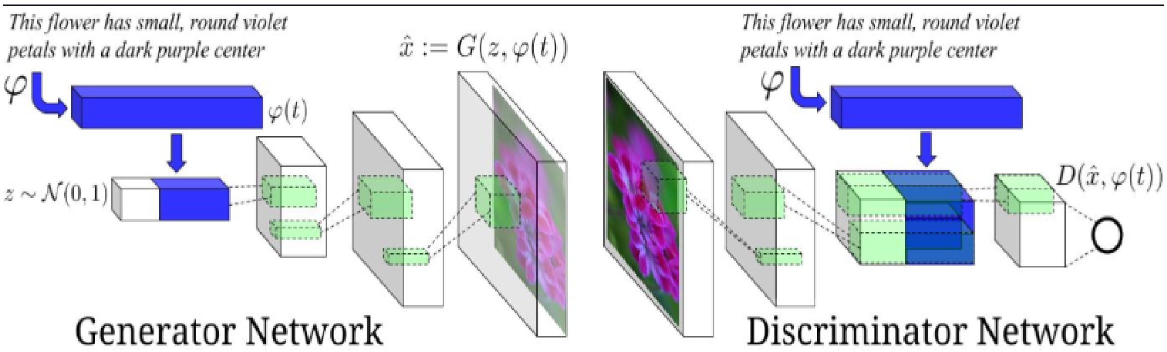

This flower has small, round violet petals with a dark purple center
$\hat{x} := G(z, \varphi(t))$
φ
$\varphi(t)$
$z \sim \mathcal{N}(0,1)$
Generator Network
This flower has small, round violet petals with a dark purple center
φ
$D(\hat{x}, \varphi(t))$
Discriminator Network

Figure 1: GAN

new "imbalanced text-image pairs training" method to improve performance. The model outperforms existing models regarding image quality and diversity and can generate novel and creative images beyond the training data.(Radford et al., 2019) The paper proposes a new method for high-resolution image synthesis using a combination of diffusion models and hierarchical latent variable models. The method can handle high-resolution images and efficiently train models on large-scale datasets, generating high-quality images with fine-grained details and realistic textures. The authors also introduce a new metric, SH-IoU, for evaluating the quality of high-resolution image synthesis. The method has impressive results on various benchmark datasets and can be used for image inpainting and super-resolution tasks.(Rombach et al., 2022)

4 Dataset details:

The Oxford 102 Flowers dataset is a collection of 8,189 images of 102 types of flowers used to study fine-grained recognition. It has a training, validation, and test set with balanced categories. The images are in JPEG format with a resolution of 256x256 pixels, and the dataset includes text files with flower names, IDs, and ground truth labels. It is a popular benchmark dataset for computer vision and machine learning research.
We used the Caltech-UCSD Birds 200-2011 (CUB) dataset for our baseline model. For baseline we reran the results with Oxford102 Flower dataset.

5 Experimental Setup and Results.

5.1 Environment

5.2 BaseLine Setup

To set up our baseline model, we followed several steps. First, we loaded the pre-trained GAN model weights for the Generator and discriminator. Then, we defined a function to preprocess the input text data from the Keras library, which would be used to create an image as input to the Generator. We then defined another function that took the preprocessed text data and generated images using the pre-trained GAN model. Using this function, we generated images based on the input text data. Finally, we used the matplotlib library to visualize the generated images, allowing us to observe the quality of the generated images and make improvements to the model as needed. By following these steps, we were able to set up a baseline model for generating images from text data using a pre-trained GAN model.

5.3 Final Model Setup

- Python programming environment (e.g., Anaconda, Jupyter Notebook)
- We loaded the images and captions from the dataset and converted them into CSV files that contain the image filenames and their corresponding captions.
- We generated embeddings for both the image and text data. GloVe was used for text. These embeddings would be used as input to the Generator and Discriminator.
- Image visualization libraries (e.g., PIL, Matplotlib)

6 Models

6.1 Baseline

In our baseline model, we used a Stack GAN to generate images from text. We trained two models - Stage 1 and Stage 2 GAN. The Stage 1 model generated poor quality and unclear images, while the Stage 2 GAN generated slightly better images where we could see the bird structure. However, the training was for fewer epochs due to the model's heaviness, and the model can generate better images with more epochs and available resources. We replicated the code available on GitHub and adjusted the Generator and Discriminator Losses accordingly. The obtained losses were slightly higher than the original model, but our main focus was on the quality of the generated images, which showed the differences between the two models.

6.2 Final Models

Generative Adversarial Networks (GANs) are a deep learning model with promising results for a text-to-image generation. The GAN model consists of two parts: a generator and a discriminator. The generator takes a textual description as input and generates a corresponding image. It is trained to produce images that are visually similar to real images. On the other hand, the discriminator distinguishes between the generated and real images. It is trained to identify the differences between the generated and real images. During training, the generator and discriminator are trained in an adversarial manner. The generator tries to fool the discriminator by generating visually similar images, while the discriminator tries to distinguish between the generated and real images. This process continues until the generator produces images indistinguishable from the real images. The GAN model for text-to-image generation has been extended to include attention mechanisms (AttnGAN), multi-stage generation (StackGAN), and larger models (BigGAN) to improve the quality of the generated images. These variations have shown even better results for generating high-quality images from textual descriptions.

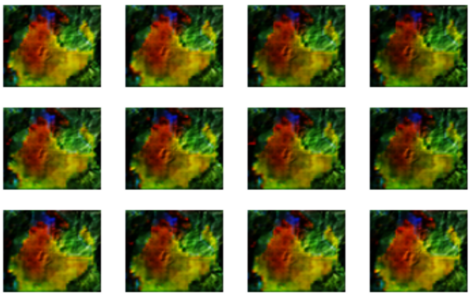
- StackGAN is a GAN-based model that generates high-resolution images in a two-stage process. The first stage generates a low-resolution image, and the second stage generates a high-resolution image from the low-resolution image.
- AttnGAN is a GAN-based model that uses an attention mechanism to focus on different regions of an image, allowing it to generate images with more detailed and diverse features.
- BigGAN is a GAN-based model that uses a larger architecture and more powerful computing resources to generate high-quality images with a high degree of variation.

While all these models use GANs as their underlying architecture, their specific implementation, objectives, and performance differ.

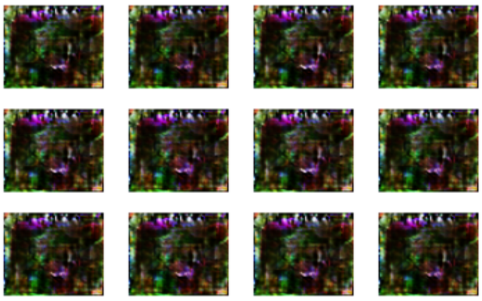
7 Observations and Error Analysis

In this section we present the different output generated by our models for the given text. As the model is trained is only on the Flowers dataset, it will generate images related to Flowers only.
We also have other models such as Stable Diffusion, and OpenAICLIP that are pre-trained models and can be directly used to generate images given the text. Such models are trained on multiple datasets and we can compute their cosine similarity, and FID score and check for performance. We also have tried those models, but they have not been written from scratch.

The Attention GAN model generates this. Text Prompt = " a large flower with pink and white petals and a brown stem "



This is the result for GAN model. Text Prompt = " a large flower with pink and white petals and a brown stem "



This is the result for BigGAN model. Text Prompt = " a large flower with pink and white petals and a brown stem "

Parameter	Value
Input Size (Image)	[64 x 64]
Embedding Size (For Text)	[200]
Epochs	[2500]
Loss	[Binary Crossentropy]

Table 1: Table showing the parameters used.

Model	Generator Loss	Discriminator Loss
GAN	2.2375	0.8340
AttnGAN	2.2905	0.8068
BigGAN	1.9743	1.1712

Table 2: Table showing the losses for Generator and Discriminator of different models

As these are generative models, we can only check the quality of the images produced by the model. We have given the losses for generator and discriminator for reference but that does not play an important role here as in generation task.

From the table we can observe the following points:

- BigGan generator loss is comparatively lesser but the discriminator loss is more. As compared to others it generated better images and we can see the flower with the text prompted.
- BigGan needs more number of epochs so it should be trained more.
- Simple GAN performed better in less epochs.
- We could run the model for more epochs and test the images.
- We could also try different values for parameters and see the results. the orginal images have been of 256x256, we have generated embeddings for 64x64.

Current Challenges and Future scope of work

Text-to-image generation is a rapidly evolving field, and there is still a lot of scope for improvement. One of the major challenges is to generate high-resolution images with fine-grained details. Several recent works have proposed solutions to this challenge, such as the HiSD model proposed by Chen et al. in their paper "HiSD: Hierarchical Semantic Disentanglement for Image Generation with Perceptual Guidance" in 2022. The HiSD model uses a hierarchical architecture to disentangle the high-level and low-level semantic information and generate high-resolution images with fine-grained details. Another promising direction is using pre-trained language models such as GPT-3 and CLIP for text-to-image generation (Kang et al., 2023). These models have shown significant improvements in natural language processing and computer vision tasks and can be used to generate more accurate and diverse images from textual descriptions. Another future scope of work is multi-modal learning, where textual descriptions and other modalities, such as audio and video, are used together to generate images (Radford et al., 2021).

8 Contributions

Drishya Uniyal : Attn GAN and BigGan model. Prashant Sharma: GAN implementation and tried other models like Stable Diffusion and OpenAIClip. The report has been made by both the members equally.

References

Minguk Kang, Jun-Yan Zhu, Richard Zhang, Jaesik Park, Eli Shechtman, Sylvain Paris, and Taesung Park. 2023. Scaling up gans for text-to-image synthesis. *arXiv preprint arXiv:2303.05511*.

Elman Mansimov, Emilio Parisotto, Jimmy Lei Ba, and Ruslan Salakhutdinov. 2015. Generating images from captions with attention. *arXiv preprint arXiv:1511.02793*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. 2011. The caltech-ucsd birds-200-2011 dataset.

Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. 2018. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1316–1324.

Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. 2017. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 5907–5915.