CentraleSupélec

Deep Learning Project Report

🎨 Monet-style painting 👩‍🎨 with stable diffusion model 👨‍💻

Abbas, Muhammad Zain

Binte Jahangir, Khushnur

CentraleSupélec

02/10/2022

1. Introduction

I'm Something of a Painter Myself, is a kaggle challenge, introduced back in 2020 with the aim to mimic Claude Monet style like art in a very convincing way. Although the original challenge is about Generative Adversarial Networks (GANs), we, however, decided to try our luck in mimicking monet-style paintings with a diffusion model.

Our main objectives in this project are to: 1) gain a better understanding of diffusion models. 2) figuring out the different ways we can approach this problem. 3) play around with some latest techniques in the diffusion modeling world (i.e: CrossAttention etc).

2. Why a diffusion model ?

GAN is an algorithmic architecture that uses two neural networks that are set one against the other to generate newly synthesized instances of data that can pass for real data. Diffusion models have become increasingly popular as they provide training stability as well as quality results on image and audio generation.

Though GANs form the framework for image synthesis in a vast section of models, they do come with some disadvantages that researchers are actively working on. Some of these, as pointed out by Google, are:

a. **Vanishing gradients**: If the discriminator is too good, the generator training can fail due to the issue of vanishing gradients.
b. **Mode collapse**: If a generator produces an especially plausible output, it can learn to produce only that output. If this happens, the discriminator's best strategy is to learn to always reject that output. Google adds, "But if the next generation of discriminator gets stuck in a local minimum and doesn't find the best strategy, then it's too easy for the next generator iteration to find the most plausible output for the current discriminator."
c. **Failure to converge**: GANs also have this frequent issue to converge.

Owing to the known problems with GANs, and the fact that diffusion models are slowly taking over the domain, we decided to do this project with a diffusion model, rather than GANs.

3. Different approaches:

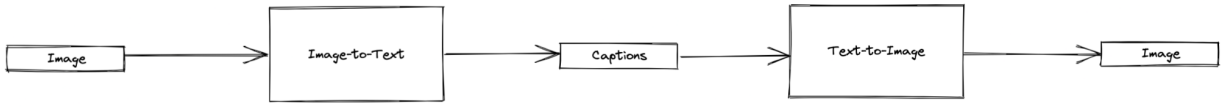
Right from the start, we got to know about two different ways to mimic monet-style paintings with diffusion models.

a. **Image-to-Image**: There are several open-source image-to-image diffusion models which we could use. For example: Hugging Face provides a bunch of models for this case.
b. **Image-Text-Image**: Although, bit more complicated than the latter in terms of architecture, this has proven to yield better results in some cases.

For our case, we decided to go with the *Image-Text-Image* approach. This way, we would be able to learn and explore more.

4. Main idea

We found pre-trained models for our approach to the problem. We didn't even need to fine-tune it for our dataset, since the main idea behind the kaggle competition was to mimic the monet-style (produce similar styled-images).



Breaking down the problem, we need to do two things:

a. Image-to-Text
b. Text-to-Image

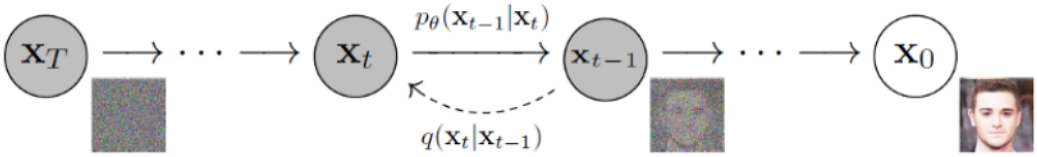
4.1. Image-to-Text:

We can think of this part as **vector-to-sequence**. There are several ways to do this part. Back in 2015, a team of researchers at Google proposed the Show and tell model, which was later added to tensorflow/models for sometime. In this model, they used *Inception v3* and *LSTM* to generate captions for images.

Another SOFA model that we looked at was Show, Attend and Tell, which used transformers (since, "Attention is all you need"). There were again some variations of architecture but at the end, we picked a model which was quite similar to the idea of *Show, Attend and Tell.*

4.2. Text-to-Image:

From text-to-image, diffusion models are the go-to models these days. A typical diffusion process can be seen below:



Stable Diffusion is a latent text-to-image diffusion model capable of generating photo-realistic images given any text input. We used a stable diffusion model by stability.ai; it was available via hugging face API. We tried and tested with v1.2, v1.3 and v1.4; and decided to go with the latest version.

4.3. CrossAttention

We played around with a cross-attention idea with stable diffusion. Original motivation came from this youtube video (*Diffusion models on steroids*). The main idea was that we can have another layer on top of a stable diffusion model and assign some weights to the individual tokens in order to affect the outcome of the model.

For e.g: for prompt "*A fantasy landscape with a pine tree in the foreground and a red sun setting in the distance, trending on artstation*", stable diffusion model generated the following image:

Let’s say we want the same image but with less pine-tree. We can achieve this by simply assigning some negative weights to the token (let’s say “pine”):

Link: https://github.com/mohammadzainabbas/Deep-Learning-CS

Kaggle Challenge: https://www.kaggle.com/competitions/gan-getting-started/overview