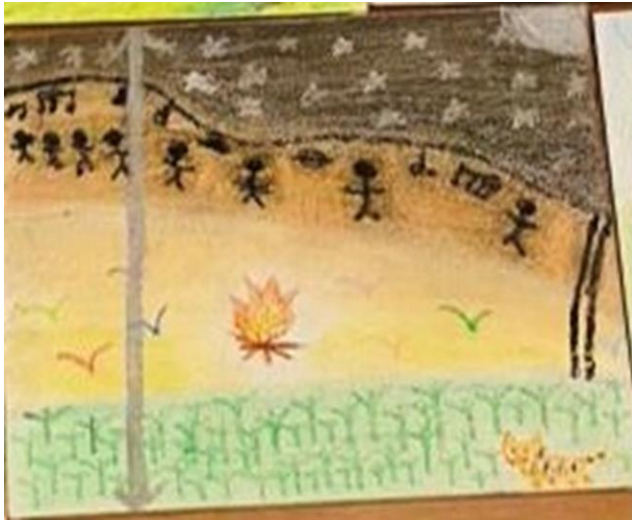


The Spirit of Life

By: Tejas Srivatsav

Context

As the saying goes, every painter paints himself. I am an ECE Master's student with minimal artistic experience aside from this pastel painting from winter break:



And so the focus of this project is myself. Through my twitter account I hope to gather information about myself and use it to generate a relevant image.

Tools

The VQGAN + CLIP model that I used was based on this paper:

<https://arxiv.org/abs/2204.08583> and this was the Colab notebook:

<https://colab.research.google.com/drive/18ZooqWH0-KsDNJOMCRilh0l2z8K5UU2G?usp=sharing>

The tool I used to scrape my tweets and collect them into a csv file was a twitter scraper from

Apify at this link: <https://apify.com/vdrmota/twitter-scraper#features>

The model I used for text generation based on my tweets was a Markov chain generator. Its

GitHub can be found here: <https://github.com/jsvine/markovify>

I also used the pandas library for data processing

I attempted to use the snsrape python library to scrape all of my tweets and collect them into a json file however did not end up getting it to work.

I also attempted to use twint to scrape my tweets and collect them into a list however this did not work either.

Process

To be clear, the model I used for this project was a VQGAN + CLIP implementation with a pooling and quantize method. I experimented with several datasets for pre-training VQGAN and used a ViT-B/32 model for the CLIP. ImageNet_16384 was the dataset used for the first few trials. The seed number I used was 42 and I ran each experiment for 200 iterations.

The first step in generating an image that represents myself well was to choose an appropriate text prompt to the image generation model. As a simple trial, I used the prompt “Tejas” since it is my first name and it is indicative of my familial context. This was the result:



Tejas actually is the name of a plane so I can understand the relation of the picture to the prompt.

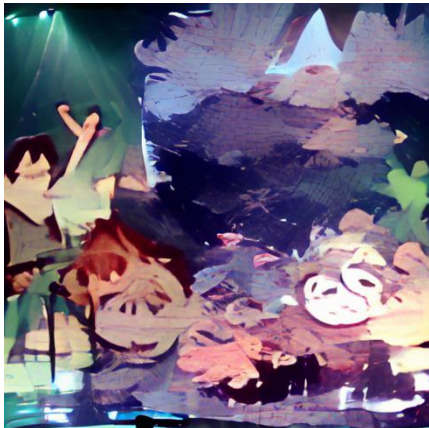
Next I tried “Tejas | Srivatsav”, where the | character is used to separate multiple prompts. This was the result:



The lines in white do look like characters from an Indian language, so once again I can understand the relation.

I then tried using some of my tweets as inputs.

“Aaaaaaaa + yest as well”:



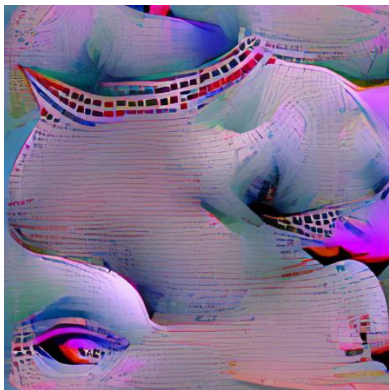
This output was quite vibrant and abstract, which of course makes sense considering my tweet is abstract

“Aaaaaaaa (+ yest as well) | Unfettered | Marshmallow | No days off. Saturday is a social construct | Disentangle”:



This one seemed less meaningful but I like the color scheme.

“Tejas | Srivatsav | Aaaaaaaa (+ yest as well) | Unfettered | Marshmallow | No days off. Saturday is a social construct | Disentangle”:



This was quite similar to the previous one – at this point there were too many prompts to produce a significant result. As a result, I decided to train a text generator model on all of my tweets to hopefully produce one simple text output that I could then feed to the image generator model.

Firstly, I needed to collect all of my tweets in a format that could be used for training. To do so, I tried various methods of scraping twitter. I tried the snsrape python library to download all of the data into a json format, however I kept running into errors when doing so. Next I tried the twint

python library to collect all of the data into one object, however this gave me errors as well. It seems that the developers haven't resolved all issues with the library.

What I ended up using actually involved no code, as I simply found a free twitter scraping tool that someone had created on the Apify platform. It enabled access to lots of information but I chose only the content of my tweets and downloaded results in csv format.

The next step was data preprocessing. I stored only the full text of each tweet into a text file. I then removed all emojis, commas, and empty lines with the purpose being to only have raw text in UTF-8 format.

I then used Markovify to build a model based on the raw text and generate a few sentences. Its outputs were:

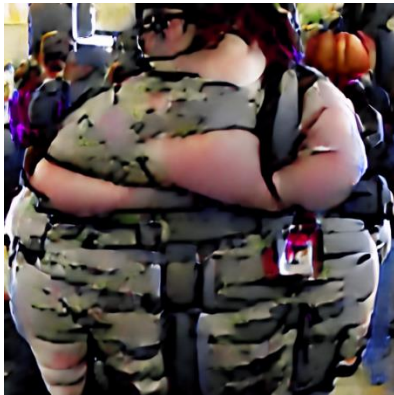
- “And so they have an extreme lvl of empathy and understand most/all of the players but you walk in and scout a little.”
- “Imagine not having access to literature and not being able to understand the situation but what would he/she think?”
- “Felt comfortable overall — it’s just sugar and milk.”
- “If they did, imo the person wouldn’t be able to travel outside of your community.”
- “It’s weird tho bc 10 lbs by Halloween weekend.”

These text prompts were all fed to the image generation model and results were obtained for each of the 3 datasets that the model was pre-trained on. Also, my face was used as the starting point because I felt that it was a sensible addition. Boy was I wrong...

ImageNet:



OpenImages:



S-Flickr:



None of those images were aesthetically appealing and quite frankly they were far from what I wanted. In my opinion, I was using too many words and felt that a simpler prompt might work better. I also decided to remove my face from being the starting point because I wanted something abstract and vibrant. Finally, I generated one prompt: “I don’t know how to argue against the spirit of life”. Although I have never tweeted this, I really like the message and it certainly relates to my perspective of life. And so once again I experimented with 3 different datasets for pre-training.

ImageNet:



OpenImages:



Both of these results exhibit symbols of spirit and life but weren’t visually appealing to me.

Final Result

The S-Flickr dataset happened to provide my favorite result which was:



Strangely enough it was somewhat similar to the painting that I referred to at the beginning – at least with respect to colors and the fire in the center. I really like the defined shape of the fire, as well as the general colors used in this artwork. The green and brown make it look quite natural and earthly, which are quite appropriate given my recent experience in India where I interacted with natural elements and disconnected from technology.

Related Ideas

RAM was a bit of an issue while testing various ideas. I had to transfer to an account with Colab Pro and I still reached usage limits somehow.

The next logical step in my twitter data extraction process would be to also use replies, retweets and likes as part of my model. Especially in my case, my liked tweets represent a completely different (more relatable) side of me and so that would be relevant information to incorporate. A system that weighs tweets relative to the amount of engagement they receive could lead to an artwork that is more relatable and generally appealing.

It was also clear that reducing the lengths of text prompts led to simpler images that were more likely to be appealing. To test out the extent of this trend I could have generated an even shorter prompt than what I used for my final result.