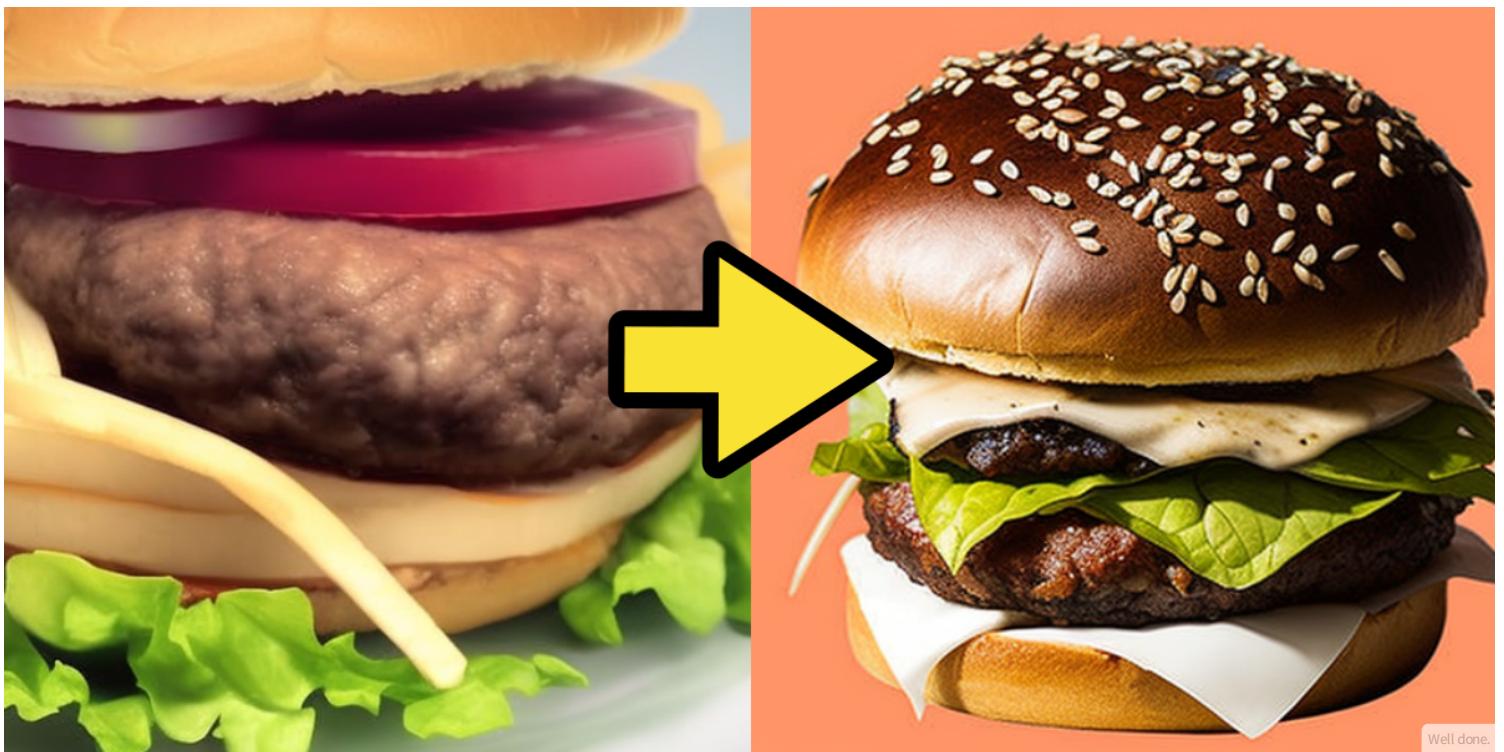


Stable Diffusion 2.0 and the Importance of Negative Prompts for Good Results

Negative prompts can be far superior than traditional prompt additions.

November 28, 2022 · 8 min read ·  [Stable Diffusion](#), [Image Generation](#), [AI](#)



As an unexpected surprise, StabilityAI released [Stable Diffusion 2.0](#) last week, the next major version of the text-to-image AI that has been warping the entire ecosystem. Architecture-wise it's mostly the same, except with a new text encoder ([OpenCLIP](#) instead of [OpenAI](#)'s CLIPText). StabilityAI boasts that Stable Diffusion 2.0 has [better performance quantitatively](#), but art in the end is subjective.

Within 24 hours after release, users on [Reddit](#) and [Twitter](#) noted that the new model performed *worse* than Stable Diffusion 1.5 with the same exact input prompts and settings. Some users also noticed that putting in the names of real artists such as the [infamous Greg Rutkowski](#) had zero effect on the output.

Some point to the fact that the new model was trained on fewer NSFW images as the culprit for these changes, but in my opinion the culprit here is the switch to OpenCLIP. A new text encoder means some of the assumptions and prompt hacks for earlier versions of Stable Diffusion may no longer work. On the other hand, it may enable *new* prompt hacks. The CEO of StabilityAI Emad Mostaque [mentioned](#) that negative prompts should work better due to the way the model was trained. It's still theory though; practice and experimentation is always better.

I hadn't played with negative prompts in Stable Diffusion before, although it is rumored that it's part of the secret sauce behind some of the more well known commercial Stable Diffusion services. But after lots of experimenting with negative prompts in SD 2.0, it's clear that negative prompts are the key to getting good results from the model reliably, and most surprisingly, negative prompts can be far superior than traditional prompt additions.

An Introduction to Negative Prompting

i All generated images in this blog post are generated by Stable Diffusion v2.0 base (via [diffusers](#)) with a classifier-free guidance of 7.5, the Euler Ancestral scheduler, with 50 denoising steps.

Analogous with normal text-to-image prompting, negative prompting indicates which terms you do not want to see in the resulting image. At a technical level for Stable Diffusion, the encoded negative prompt serves as an high-dimension anchor the diffusion process strays away from.

Let's test it out with Stable Diffusion 2.0. For example, let's go back to my [VQGAN + CLIP prompts](#) and try [cyberpunk forest by Salvador Dali](#).



prompt: [cyberpunk forest by Salvador Dali](#), via Stable Diffusion 2.0

What if you wanted to remove things like [trees](#) and/or a certain color like [green](#)? That's what you'd put in your negative prompt. Can Stable Diffusion 2.0 adjust?



prompt: [cyberpunk forest](#) by Salvador Dali; negative prompt: [trees, green](#), via Stable Diffusion 2.0

Indeed it does, with a larger dose of surrealistic cyberpunk, but it is still a forest albeit more metaphorical.

One popular trick is to also include more abstract bad-image concepts like [blurry](#) and [pixelated](#) in order to theoretically improve the image. But are these negative prompts better than the prompt additional “ingredients” like [4k hd](#) and [trending on artstation](#) like CLIPText-based text-to-image AI before it? How do negative prompts interact with those positive prompt additions? Let’s test this further and more empirically.

In The Style of Wrong

As a quick aside, textual inversion, a technique which allows the text encoder to learn a specific object or style that can be trivially invoked in a prompt, does work with Stable Diffusion 2.0, although since the text encoder is different (and larger, with 1024D embeddings instead of 768D), each textual inversion embedding has to be retrained but otherwise behaves the same way. One popular style in SD 1.X is the “[Midjourney](#)” style located [here](#), which has a overly-fantasy aesthetic. I’ve trained a new version of the [`<midjourney>`](#) token (available [here](#)).

Additionally, there’s a new possibility of using textual inversion for negative prompts. Redditor Nerfgun3 trained a “[negative embedding](#)” for SD 1.X by generating a dataset of synthetic images by using common negative prompts as positive prompts instead, then training a textual inversion embedding on them. I [reproduced that process](#) with a few tweaks to improve the synthetic dataset and trained a new [`<wrong>`](#) token (available [here](#)).

We can now cross-test a positive prompt addition or a positive token with a negative prompt or negative token to see just how impactful the negative prompts are. Here a list of prompts to test, with positive prompt additions in [green](#) and negative prompt additions in [red](#):

Label	Description
PROMPT	hyper-detailed and intricate, realistic shaded, fine detail, realistic proportions, symmetrical, sharp focus, 8K resolution
<code><TOKEN></code>	in the style of <midjourney>

Label	Description
PROMPT	ugly, boring, bad anatomy
<TOKEN>	in the style of <wrong>

For example, one test input to Stable Diffusion 2.0 could be a prompt of `cyberpunk forest by Salvador Dali, in the style of <midjourney>` and a negative prompt of `in the style of <wrong>`, corresponding a green `<TOKEN>` prompt label and a red `<TOKEN>` label respectively.

Additionally, each individual generated image will start with the same initial latent, with seeded scheduling. This allows the impacts of negative prompts to be shown more clearly, as keeping the same prompt given a constant initial latent will allow the generated image composition to remain the same while changing the negative prompts.

Now, let's finally begin. Let's start off with `Steve Jobs head` as the base prompt; simple enough.

base prompt: `Steve Jobs head`, seed: 59049, via Stable Diffusion 2.0

The two prompt additions each changed the style; the base prompt did a cartoon; the realistic prompt addition made it more of a 3D render, and the Midjourney token made it an artsy approach. However, when negative prompts are added, each image becomes more clear, with less blurriness, more neutral lighting, and greater skin detail. More notably, the `<wrong>` token did much better than the smaller negative prompt.

How about an image generation classic: the famous avocado armchair which was demoed with the original DALL-E?

base prompt: an armchair in the shape of an avocado. an armchair imitating an avocado., seed: 59049, via Stable Diffusion 2.0

Here's where things get interesting; the positive text prompt addition ruins the intent of the original prompt completely, and again the negative prompts each refine the corresponding image with more detail (including the whole avocado!)

Now that we have good demos, let's go back to Dali's cyberpunk forest:

base prompt: [cyberpunk forest by Salvador Dali](#), seed: 59049, via Stable Diffusion 2.0

In this case, both positive prompt additions wipe out Dali's style, opting for a more realistic forest and later reinforced by the negative prompts. In the case of the original prompt, the negative prompts further emphasize Dali's artistic style. This a good example of positive prompt additions not being a strictly good thing.

Can negative prompts help create yummy AI-generated food [like DALL-E 2 can](#)? Let's see if it can make a hamburger:

base prompt: `a delicious hamburger`, seed: 19683, via Stable Diffusion 2.0

This one is a pretty unambiguous case of negative prompts helping out the final result; the output using both tokens is pretty close to DALL-E 2 quality!

Another interesting thing about Stable Diffusion 2.0 is that text renders better; small text is not fully legible, but large text is more discernable. Perhaps Stable Diffusion 2.0 can envision a [New York Times](#) front page depicting the rise of robot overlords.

base prompt: [an evil robot on the front page of the New York Times](#), seed: 19683, via Stable Diffusion 2.0

There's a surprising amount of evil robot variety despite the fixed latent inputs, and the layouts of the newspaper are very accurate to the NYT. The especially weird negative-prompt-text-only image is an example of a surprisingly rare mode collapse, which is interesting (or it's Stable Diffusion *hiding something*). Although the robot from the original prompt is clearly the most evil.

We can also investigate how negative prompts can help the rendering of human subjects. Let's take [Taylor Swift](#). What happens when she becomes President Taylor Swift? (hopefully Stable Diffusion doesn't confuse her with the other [President Taylor](#))

base prompt:President Taylor Swift giving her presidential inauguration speech,seed:6561, via Stable Diffusion 2.0

So both the positive prompt addition types make the initial output unambiguously worse, which is a surprise. But the negative prompts fix them, and again, give President Tay a nice wardrobe variety. It's worth noting that Stable Diffusion 2.0 is better at generating correct hands than SD 1.X...just don't look at them too closely.

Lastly, we can't forget about [Ugly Sonic](#), the initial hedgehog from the Sonic Movie who was the subject of my [previous Stable Diffusion blog post](#). I received many complaints that the AI-generated Ugly Sonic wasn't really Ugly Sonic because the generated Ugly Sonics didn't have human teeth! Time to fix that!

base prompt: <ugly-sonic> smiling with human teeth, seed: 6561, via Stable Diffusion 2.0

In this case, the negative prompts *ruined* Ugly Sonic because they progressively remove his human teeth!

Conclusion

As always with AI art, your mileage will vary, but negative prompting will be a much more important tool going forward in AI Image generation and anchoring on prompt engineering strategies that worked in the past is a mistake. It also provides a good opportunity to stop using living artists as a prompt engineering crutch since that may not be possible moving forward, which is a good thing for the industry (especially given [legal uncertainty](#)!).

All my code used to generate the images for this article are available [in this GitHub repository](#), including a [Colab Notebook](#) for general generation with the <[wrong](#)> token and a [Colab Notebook](#) for the 3x3 labeled grid images, with easily tweakable prompt inputs if you want to run your own experiments.

It would be interesting to see if it's possible to finetune Stable Diffusion 2.0 such that it gains an "intrinsic" negative prompt without having to manually specify it...which might be happening sooner than you think. 😊

Disclosure: I am neither an artist nor an expert in art theory. All my comments on what are "good" AI art generations are my own (likely bad) opinions.

 If you liked this post, I have set up a [Patreon](#) to fund my machine learning/deep learning/software/hardware needs for my future crazy yet cool projects, and any monetary contributions to the Patreon are appreciated and will be put to good creative use.



Max Woolf

Data Scientist at BuzzFeed in San Francisco. Creator of AI text generation tools such as aitextgen and gpt-2-simple. I am the data.



Related

- [I Resurrected "Ugly Sonic" with Stable Diffusion Textual Inversion](#)
- [Absurd AI-Generated Professional Food Photography with DALL-E 2](#)
- [How to Generate Customized AI Art Using VQGAN and CLIP](#)
- [Easily Transform Portraits of People into AI Aberrations Using StyleCLIP](#)

Copyright Max Woolf © 2022

Published with [Wowchemy](#) — the free, [open source](#) website builder that empowers creators.