DALL-E 2

VCHS AI Club

With only a short text prompt, DALL-E 2 can generate completely new images that combine distinct and unrelated objects in semantically plausible ways, like the images which were generated by entering the prompt "a bowl of soup that is a portal to another dimension as digital art".

DALL-E 2 can even modify existing images, create variations of images that maintain their distinctive features, and interpolate between two input images. DALL-E 2's impressive results have many wondering exactly how such a powerful model works under the hood.

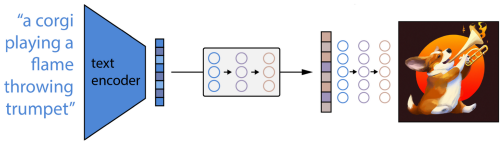
DALL-E Under the Hood:

A Bird's Eye View

Before diving into the details of how DALL-E 2 works, let's orient ourselves with a high-level overview of how DALL-E 2 generates images. While DALL-E 2 can perform a variety of tasks, including image manipulation and interpolation, we will focus on the task of image generation.

At the highest level, DALL-E 2's works very simply:

1. A text prompt is input into a text encoder that is trained to map the prompt to a representation space.
2. A model called the prior maps the text encoding to a corresponding image encoding that captures the semantic information of the prompt contained in the text encoding.
3. An image decoder stochastically generates an image which is a visual manifestation of this semantic information.



DALL-E Under the Hood:

A Detailed Look

Step 1

Linking Textual and Visual Semantics

After inputting "a teddy bear riding a skateboard in Times Square," DALL-E 2 outputs the image:

a teddy bear riding a skateboard in Times Square

How does DALL-E 2 know how a textual concept like "teddy bear" is manifested in the visual space?

The link between textual semantics and their visual representations in DALL-E 2 is learned by another OpenAI model called CLIP (Contrastive Language-Image Pre-training).

CLIP is trained on hundreds of millions of images and their associated captions, learning how much a given text snippet relates to an image. That is, rather than trying to predict a caption given an image, CLIP instead just learns how related any given caption is to an image.

This contrastive rather than predictive objective allows CLIP to learn the link between textual and visual representations of the same abstract object. The entire DALL-E 2 model hinges on CLIP's ability to learn semantics from natural language, so let's take a look at how CLIP is trained to understand its inner workings.

CLIP Training

The fundamental principles of training CLIP are quite simple:

1. All images and their associated captions are passed through their respective encoders, mapping all objects into an m-dimensional space.

2. The cosine similarity of each (image, text) pair is computed.
3. The training objective is to simultaneously maximize the cosine similarity between N correct encoded image/caption pairs and minimize the cosine similarity between N2 - N incorrect encoded image/caption pairs.

Significance of CLIP to DALL-E 2

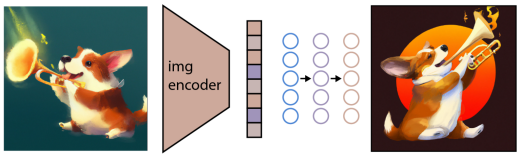
CLIP is important to DALL-E 2 because it is what ultimately determines how semantically related a natural language snippet is to a visual concept, which is critical for text-conditional image generation.

Step 2

Generating Images from Visual Semantics

After training, the CLIP model is frozen and DALL-E 2 moves onto its next task — learning to reverse the image encoding mapping that CLIP just learned. CLIP learns a representation space in which it is easy to determine the relatedness of textual and visual encodings, but our interest is in image generation. We must therefore learn how to exploit the representation space to accomplish this task.

In particular, OpenAI employs a modified version of another one of its previous models, GLIDE, to perform this image generation. The GLIDE model learns to invert the image encoding process in order to stochastically decode CLIP image embeddings.



An image of a Corgi playing a flame-throwing trumpet passed through CLIP's image encoder.

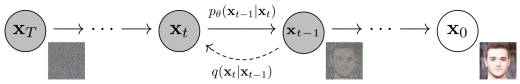
As depicted in the image, it should be noted that the goal is not to build an auto-encoder and exactly reconstruct an image given its embedding, but to instead generate an image which maintains the distinctive features of the original image given its embedding. In order perform this image generation, GLIDE uses a Diffusion Model.

What is a Diffusion Model?

Diffusion Models are a thermodynamics-inspired invention that have significantly grown in popularity in recent years.

Diffusion Models learn to generate data by reversing a gradual noising process. Depicted in the figure, the noising process is viewed as a parameterized Markov chain that gradually adds noise to an image to corrupt it, eventually (asymptotically) resulting in pure Gaussian noise.

The Diffusion Model learns to navigate backwards along this chain, gradually removing the noise over a series of time steps to reverse this process.



Diffusion Model schematic

If the Diffusion Model is then "cut in half" after training, it can be used to generate an image by randomly sampling Gaussian noise and then de-noising it to generate a photorealistic image. Some may recognize that this technique is highly reminiscent of generating data with auto-encoders, and Diffusion Models and auto-encoders are, in fact, related.

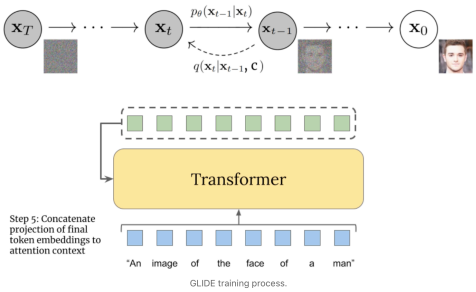
> *A Markov chain or Markov process is a stochastic model describing a sequence of possible events in which the probability of each event depends only on the state attained in the previous event.*

GLIDE Training

While GLIDE was not the first Diffusion Model, its important contribution was in modifying them to allow for text-conditional image generation. In particular, one will notice that Diffusion Models start from randomly sampled Gaussian noise.

It at first unclear how to tailor this process to generate specific images. If a Diffusion Model is trained on a human face dataset, it will reliably generate photorealistic images of human faces; but what if someone wants to generate a face with a specific feature, like brown eyes or blonde hair?

GLIDE extends the core concept of Diffusion Models by augmenting the training process with additional textual information, ultimately resulting in text-conditional image generation. Let's take a look at the training process for GLIDE:



DALL-E 2 uses a modified GLIDE model that incorporates projected CLIP text embeddings in two ways. The first way is by adding the CLIP text embeddings to GLIDE's existing time-step embedding, and the second way is by creating four extra tokens of context, which are concatenated to the output sequence of the GLIDE text encoder.

Significance of GLIDE to DALL-E 2

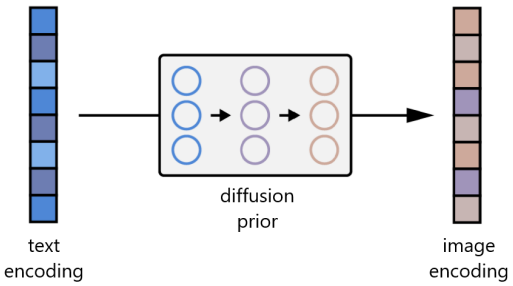
GLIDE is important to DALL-E 2 because it allowed the authors to easily port over GLIDE's text-conditional photorealistic image generation capabilities to DALL-E 2 by instead conditioning on image encodings in the representation space. Therefore, DALL-E 2's modified GLIDE learns to generate semantically consistent images conditioned on CLIP image encodings. It is also important to note that the reverse-Diffusion process is stochastic, and therefore variations can easily be generated by inputting the same image encoding vectors through the modified GLIDE model multiple times.

Step 3

Mapping from Textual Semantics to Corresponding Visual Semantics

While the modified-GLIDE model successfully generates images that reflect the semantics captured by image encodings, how do we go about actually go about finding these encoded representations? In other words, how do we go about injecting the text conditioning information from our prompt into the image generation process?

Recall that, in addition to our image encoder, CLIP also learns a text encoder. DALL-E 2 uses another model, which the authors call the prior, in order to map from the text encodings of image captions to the image encodings of their corresponding images. The DALL-E 2 authors experiment with both Autoregressive Models and Diffusion Models for the prior, but ultimately find that they yield comparable performance. Given that the Diffusion Model is much more computationally efficient, it is selected as the prior for DALL-E 2.



Prior Training

The Diffusion Prior in DALL-E 2 consists of a decoder-only Transformer. It operates, with a causal attention mask, on an ordered sequence of

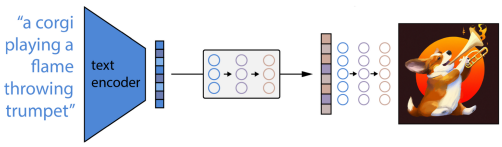
1. The tokenized text/caption.
2. The CLIP text encodings of these tokens.
3. An encoding for the diffusion time-step.
4. The noised image passed through the CLIP image encoder.
5. Final encoding whose output from Transformer is used to predict the unnoised CLIP image encoding.

Step 4

Putting It All Together

At this point, we have all of DALL-E 2's functional components and need only to chain them together for text-conditional image generation:

1. The CLIP text encoder maps the image description into the representation space.
2. The diffusion prior maps from the CLIP text encoding to a corresponding CLIP image encoding.
3. The modified-GLIDE generation model maps from the representation space into the image space via reverse-Diffusion, generating one of many possible images that conveys the semantic information within the input caption.



High-level overview of the DALL-E 2 image-generation process