



What's GPT-3?

Machine Learning has advanced - ThE eNd iS NeAr

 Justin
Jul 20, 2020

24 8 ↗

The TL;DR

GPT-3 is a Machine Learning model that generates text. You give it a bit of text related to what you're trying to generate, and it does the rest.

- Machine Learning models let you make predictions based on past data, and **generation** (creating text) is a special case of predicting things
- OpenAI, a non-profit research group, has been working on this model for years – this is the third aptly-named version after GPT and (gasp) GPT-2
- The GPT-3 model is trained via **few shot learning**, an experimental method that seems to be showing promising results in language models
- GPT-3 has picked up a lot of buzz for **how good it is** - it can generate entire published articles, poetry and creative writing, and even code

OpenAI has been working on language models for a while now, and every iteration makes the news. But GPT-3 seems to represent a turning point - it's like, scary good.

Text generation and ML models

GPT-3 is a **language generation model**. Now, I majored in Data Science and I still get confused about this, so it's worth a basic refresher. Machine Learning is just about figuring out *relationships* – what's the impact of something on another thing? This is pretty straightforward when you're tackling structured problems – like predicting housing pricing based on the number of bedrooms – but gets kind of confusing when you move into the realm of language and text. What are ML models *doing* when they generate text? How does that work?

The easiest way to understand text generation is to think about a really good friend of yours (assuming you have one). At some point if you hang out enough, you get a good feel for their mannerisms, phrasing, and preferred topics of conversation - to the point where you might be able to reliably **predict what they're going to say next** ("finishing each other's sentences"). That's exactly how GPT-3 and other models like it work - they learn a lot (I mean, like really a lot) about text, and then based on what they've seen so far, predict what's coming next.

The actual internals of language models are obviously Very Scary and Very Complicated - there's a reason that most big advancements come from big research teams full of PhDs.

❑ Deeper Look ❑

For the statistically inclined, most language models work via building a basic probability distribution of their training data (how often which words appear) and use that to predict what's coming next. Advanced models go a few steps further by remembering longer term dependencies, using metadata like sentence length, and, most importantly, getting trained on a shit ton of data.

❑ Deeper Look ❑

The real innovation of GPT-3 - and the paper published in May that kicked it off – is a training method called **few shot learning**.

Big training and few shot learning

Generally, cutting edge in ML right now is closely associated with access to **a lot of training data**, and powerful compute to run those training jobs on. GPT-3 is no exception – the OpenAI team trained it with 175 billion parameters, which, according to them, is "10x more than any previous non-sparse language model." They used a 45TB dataset of plaintext words (45,000 GB), filtered it down to a measly 570GB, and used 50 petaflops/day of compute (1020 operations per

second, times 50). In short - this was a massive, extremely expensive effort.

The twist, though, is that big training sets for language models isn't a new idea - even though GPT-3 *was* trained on a record amount of data, that's only a piece of the puzzle. OpenAI touts the real killer feature here as **few shot learning** – the ability of the model to use the general knowledge it has already learned and apply it to specific tasks. The basic idea is that you give the already trained model a few small examples of what you want to produce, and it **combines that knowledge** with the "genius" it already has to produce something really specific to your needs.

To make this a bit more concrete, let's take a look at one of the GPT-3 examples that went semi-viral: the model was able to write [an entirely coherent article about Bitcoin forums](#). The author gave GPT-3 some basic text that explained his background and his website as the "few shots" and it was able to produce something strikingly normal sounding. Here's a snippet:

Are you being served?

I chose [bitcointalk.org](#) as the target environment for my experiments for a variety of reasons: It is a popular forum with many types of posts and posters.

So there are lots of posts for GPT-3 to study and learn from. The forum also has many people I don't like. I expect them to be disproportionately excited by the possibility of having a new poster that appears to be intelligent and relevant. I've been following the forum for years. There are many posts I know the answers to, so I could provide a quick response and measure how well GPT-3 does with comments similar to those I make.

I posted about one interesting tech topic every day in May, alternating between using my own words and paraphrasing my previous post with GPT-3's help. I didn't take special care to make these GPT-3-enhanced posts blend in well. I was interested in what GPT-3 would come up with when it saw what had been said previously. The table below shows some results:

Few shot learning (and its cousin, one shot learning) isn't a new concept - but the devil is in the details, and GPT-3 may be the most effective demonstration on threading the needle. The nuance is in how you connect the "general knowledge" – acquired via the giant training process – with the "specific knowledge" – acquired via those "few shots" of domain specific examples.

What this means

If you scour the internet, there are basically two reactions to GPT-3:

1. This is so cool! Look what I can do with it!
2. ThE eNd iS nEaR



GPT-3 is not going to hurt you, but it is important – having big advancements in NLP open to the public is a positive thing, and building transparency in ML research is one of the main reasons that OpenAI [exists in the first place](#). If you get a chance to play with it yourself, you'll realize that the media accentuates the *good* things that models like GPT-3 produce, but in general they generate a lot of harmless nonsense.

But what makes GPT-3 *really* interesting is that OpenAI is [releasing it to a limited](#)

number of people via a beta API. You'll be able to pass a little bit of text through to the API (the "few shots") and get back generated text programmatically.

🧠 Jog your memory

APIs let you access data or functionality programmatically via the web.

🧠 Jog your memory

This is a really interesting way to expose the model - because training is so intensive, it's not something that you'd be able to train on your laptop, or even in the cloud (unless you're fabulously wealthy). APIs let people *use* the model without *building* it - and I for one am excited to see what users come up with.

Further reading

- [The OpenAI paper on GPT-3](#)
- [An article generated \(mostly\) by GPT-3](#)
- [Creative writing and fiction via GPT-3](#)
- [More about the OpenAI charter and timeline](#)

24 8 Share

Write a comment...

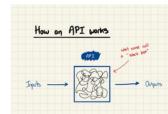
Cozidle Jul 21, 2020 Liked by Justin Gage
Man that was quite a burn there,
"The easiest way to understand text generation is to think about a really good friend of yours (assuming you have one)."
Anyway great article.
 3 Reply

1 reply by Justin Gage

Akak Jul 21, 2020 Liked by Justin Gage
Does GPT-3 has anything to do with MSN firing editors?
 1 Reply

6 more comments...

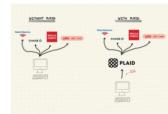
[Top](#) [New](#) [Community](#)



What's an API?

What McDonalds and Lyft have in common

Justin Jan 9, 2020 187



What does Plaid do?

Technically begrudgingly tackles Fintech

Justin Jan 14, 2021 34 3



What does New Relic do?

Keeping an eye on your servers and apps

Justin Jan 11 23 2



What's Reverse ETL?

Getting your data OUT of your warehouse?

Justin Feb 1 19



What happened to Facebook?

A basic explainer of what that outage was all about

Justin Oct 5, 2021 29 4



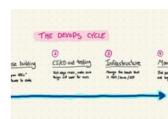
What does GitLab do?

The TL;DR GitLab is a somewhat contrarian take on DevOps: it's basically



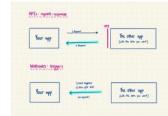
one giant tool for literally anything you'd want to do relating to building and...

■ Justin Jan 4 ○ 8 ▾



What's DevOps?

IT has a cool new name
Justin Jan 5, 2021 ○ 34 ▾



What are webhooks?

Triggered
Justin Sep 13, 2021 ○ 28 ▾



What's Headless E-Commerce?

We may be running out of names
Justin Nov 2, 2021 ○ 20 ▾



I built a (basic) Substack clone in a month

This was probably a waste of my time
Justin Sep 2, 2020 ○ 50 ▾

[See all >](#)

© 2022 Justin · [Privacy](#) · [Terms](#) · [Collection notice](#)

 [Publish on Substack](#)

Our use of cookies

We use necessary cookies to make our site work. We also set performance and functionality cookies that help us make improvements by measuring traffic on our site. For more detailed information about the cookies we use, please see our [privacy policy](#).