

# GPT-3 EXPLAINER: PUTTING GPT-3 INTO PERSPECTIVE



GPT-3 is not revolutionary. Here, I'll show you how a little perspective helps us understand what GPT-3 can—and cannot—do.

## What is GPT-3?

GPT-3 is a [language model](#) that is powered by a [neural network](#), released by OpenAI in July 2020. It's a text generator that can write articles, poetry, opinion essays, and working code—which is why it has the whole world buzzing, some with excitement, some with fear.

Before GPT-3, there was 2. GPT-2 wasn't revolutionary, and neither is GPT-3. Among GPT-2 are:

- BERT (Bidirectional Encoder Representations from Transformers)
- [XLNet](#)
- CTRL (from Salesforce)
- And more...

The key technology behind both of these models is the **transformer**. (Huggingface.co has the leading [open source repo](#) with 20 more transformer architectures that can be used.) Crucially, it is the transformer—not the model—that is the revolutionary leap. This [Kuhn-sian paradigm shift](#) lets a few words,

*"The girl sat by the tree."*

go from being encoded like this:

to looking like this:

We don't know what those numbers mean exactly: that's the issue with [explainability](#). But, the point is, the second transforms the sentence of words into a set of numbers with a lot more information in it.

Once the infrastructure is in place, money allows technology to be taken to its extreme:

- Cars go faster: [\\$12.2M for an F1 car](#)
- Buildings get taller: [\\$1.23B for a 1km tower](#)
- Language models get bigger: [OpenAI reportedly](#) spent \$12M to train GPT-3—just [a small portion](#) of the \$1B Microsoft invested in the company in 2019.

## GPT-3 use cases

Now, GPT-2 had 1.5 billion parameters. GPT-3 has significantly more—175 billion parameters. Really, a Ferrari is a Ferrari, and it's really not worth the excitement when Apple upgrades the iPhone camera from 12 MP to 14 MP. Before GPT-3, the transformer technology was already good. 2018 has already been dubbed the [year of NLP's ImageNet](#).

[As reported in Forbes](#):

*"In October 2012, a [deep neural network](#) achieved an error rate of only 16% in the ImageNet Large Scale Visual Recognition Challenge, a significant improvement over the 25% error rate achieved by the best entry the year before."*

When the error rate in classifying images fell from 25% to 16%, machine learning became much more viable for image classification. Until 2018, however, no model could offer that kind of low error rate for language tasks.

The release of GPT-3, and its rise to public attention, is because of good marketing. When was the last time people paid attention to a language model? Probably the last time OpenAI released their GPT-2 in February of 2019. How many other models have been released since then? Maybe 6 or 10...Salesforce, Google, Uber, Microsoft, and more... But the transformer technology is changing the field of [Natural Language Processing](#) (NLP).

Businesses drool. They have employees bored, itching for a new technology to come around to light their fire. Here is something new people can use. Lazy programmers will first start to automate their jobs.

- Can we just make bots book our schedules for us?
- Train a bot to generate that job applicant rejection letter so it feels more...personal.
- We already got spam filters, can't we get moron filters, too? I only want to talk to people who will boost my clout.

Why would Salesforce be excelling at language generation models? To improve their services. Intents are not hidden. Salesforce would be better at its job if it helped automate the sales pipeline. It could generate emails and chats to help a sales rep close...or to close a sale without the sales rep. But, why might [Uber be open sourcing](#) its own language models?

# Language tasks for GPT-3

A general public wants better Hallmark cards and to [eliminate the annoying phone menus](#). But, from the engineering standpoint, there are actual tasks a model like GPT-3 solves.

These are some language tasks from Sebastian Ruder, [a research scientist](#) at Google's DeepMind:

- [Automatic speech recognition](#)
- [Coreference resolution](#)
- [Data-to-Text Generation](#)
- [Dependency parsing](#)
- [Dialogue](#)
- [Domain adaptation](#)
- [Entity linking](#)
- [Grammatical error correction](#)
- [Information extraction](#)
- [Intent Detection and Slot Filling](#)
- [Language modeling](#)
- [Machine translation](#)
- [Named entity recognition](#)
- [Part-of-speech tagging](#)
- [Question answering](#)
- [Relation prediction](#)
- [Sentiment analysis](#)
- [Simplification](#)
- [Stance detection](#)
- [Summarization](#)
- [Text classification](#)
- [\(The full list\)](#)

## Better predictions, better NLP

The GPT-3 language model provides the tools to enable better predictions, thus further exploration, into each of these fields. The field of NLP can come close to checking unfinished items off its to-do list and move on.

For 20+ years(!), people were stuck on solving just a few NLP tasks using Hidden Markov Chains and really mind-numbing logic trees. The field's linguistic problems were sentence tokenization and part-of-speech tagging. Advancements in machine learning modelling in the past decade have, more or less, created industry-standard solutions to these problems, allowing researchers to explore new problems.

With these out of the way, the current problems closest to having industry-standard solutions are:

- Named-entity recognition
- Sentiment analysis
- Text classification
- Coreference resolution

With the release of GPT-3, more of these open problems in the NLP field, items on the Ruder-list,

can be closed out, become more nuanced, and new ones can be opened.

## GPT-3 and real-world language

Finally, what about, you know, the everyday language we speak to one another?

The AI may finish your sentences, but can it get that word on the tip of your tongue? The first task is valuable and cute only in an elementary rom-com; the latter is a challenging metaphorical endeavor.

An AI may be able to create 300 different variations on the sentence, "The quick brown fox jumps over the lazy dog", but in how many ways can it say, "I love you." Sure, a third cousin can say it when leaving the annual family reunion, and even a drunk at 3AM can almost knock a convincing one out. But, when it matters, can that AI really steal those words from your lips and deliver them to the receiving party?

A "Sorry" needs to come from an offending party, and a "Congratulations" needs to come from someone who knew your struggle.

*The value of words goes beyond an ability to know what to say; it comes with an assignment on the party responsible to say them.*

## Predicting when

One of the hardest things in life is to make the right choice at the right time. If people can't get that right all the time, there is no way any kind of AI model will ever accomplish that.

AI wins games with a fixed set of rules—limited degrees of freedom. If life is seen as a sequence of moves, an AI will always win. [Als win at chess](#). [Als win at Go](#). [Als win at Dota 2](#). Each one of these games has significantly more degrees of freedoms than the previous.

But, in a game like the financial markets, there are many degrees of freedom. There is pretty good accounting on corporate financial statements. Beating the market has enormous payoffs; there is an enormous reward to accurately predicting stock prices day-in and day-out. Large rewards attract many participants. Many people, for decades, have been trying to create a model to beat the stock markets.

The best group, with loads of resources, who has come the closest to successfully modelling the financial markets is Renaissance Technologies. They're a bunch of whiz-kid PhDs [making 66%+ returns a year](#) on their stock portfolio since the 1980s. That's a big deal. They claim, and the [law of large numbers](#) seems to indicate, that there is no reason to disbelieve them, they only win on just over half their trades.

An ideal model would be able to win 100% of the time. In a field with high reward, plenty of data, and even a pretty good definition of success, just over 50% [accuracy](#) is a bad model. It works for the kind of game Renaissance plays. It wouldn't work for docking spaceships.

Now, language is incredibly more varied than the financial markets. There are:

- Different kinds
- Different purposes
- Different exchanges

There has been very little accounting on language. Certainly there is no [GAAP standard](#) to normalize

and make transparent all the meanings and values of transfers of language between people. The world of language has far too many degrees of freedom to be modelled.

...by just one model.

First, the GPT-3 is a language model meant to encode sentences, to then be used for machine learning models. Its attention mechanism is good at two things:

1. **Using a few words on a page to predict words that come after it.** It improves upon a normal LSTM by being able to refer to pieces of text further back in the text. I.e.; if a girl was mentioned two paragraphs ago, an LSTM will have a harder time using that as important information to the following prediction than a transformer might.
2. **Creating a probability distribution of words likely to be important for predicting the next word.** The attention head creates, essentially, a word association tool—like curating the words scissors, shampoo, hairdresser, and salon.

The transformer has a good memory and associates all kinds of things to each other. It's not the person you want to invite to a wild party, but it's exactly who you want as your partner for trivia night or to help solve Grandma's crossword.

Second, machine learning models are only successful if the inputs and the outputs used to [interpret](#) the problem-space are clearly defined. The scientific method is important. And, as much as outsiders looking in, and inspiration [from Pedro Dominguez's book](#), might wish to believe, there is no master algorithm.

Language is used in many different ways. The only way to tackle building good tools to enhance people's day-to-day lives is by piecemealing the entirety of people's language exchanges, bit by bit. Start small; build large. Machine learning models need to start with small, clearly defined tasks like:

- Taking a piece of text and classifying it as spam, priority, or promotion
- Predicting how many stars a review might get
- Predicting what section of the news an article came from
- Classifying text as hate speech or profane
- Generating an option to say yes or no from a piece of text
- Classifying what personality spoke the words

Machine learning models will never know what needs to be said in all scenarios, and they will never take the place of the parties who are responsible to speak.

## Additional resources

For more on machine learning, explore the [BMC Machine Learning & Big Data Blog](#) and these articles:

- [Machine Learning: Hype vs Reality](#)
- [How Machine Learning Benefits Businesses](#)
- [Deep Learning vs Machine Learning: What's the Difference?](#)
- [How to Use Apache Ignite for Machine Learning](#)
- [Machine Learning with TensorFlow & Keras, a multi-part Guide](#)
- [Attention Is All You Need](#) (PDF), a technical paper on the transformer

