GPT3: Imitation Learning that Imitates Learning

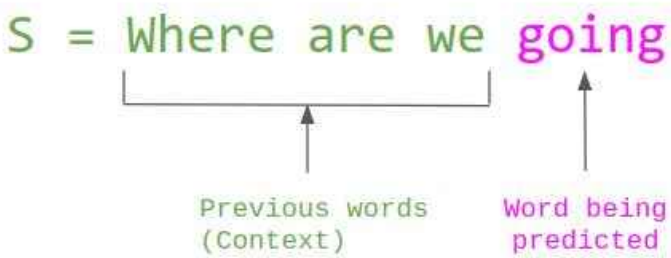
Zachary Robertson

What is GPT3?

- GPT ~ Generative Pretrained Transformer
- GPT is a language model
    - It predicts the probability of observing sentences
    - It does this through clever use of attention

What is a Language Model?

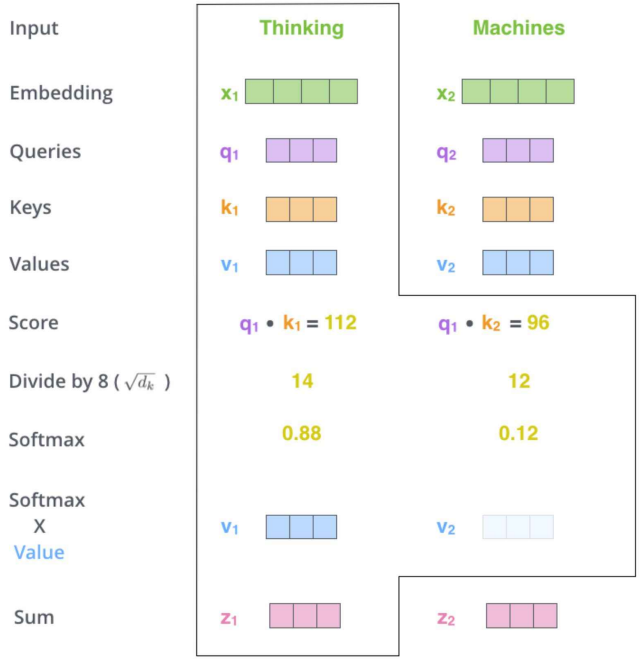
- Language models predict sentence likelihood
- Chain-rule for log-probability

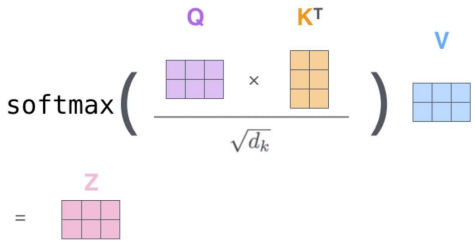


$$P(S) = P(\text{Where}) \times P(\text{are} \mid \text{Where}) \times P(\text{we} \mid \text{Where are}) \times P(\text{going} \mid \text{Where are we})$$

Self-Attention in 60 seconds or less

- We check a set of keys for a match with a query
- We return the value for each match
- ‘Super’-Convolution

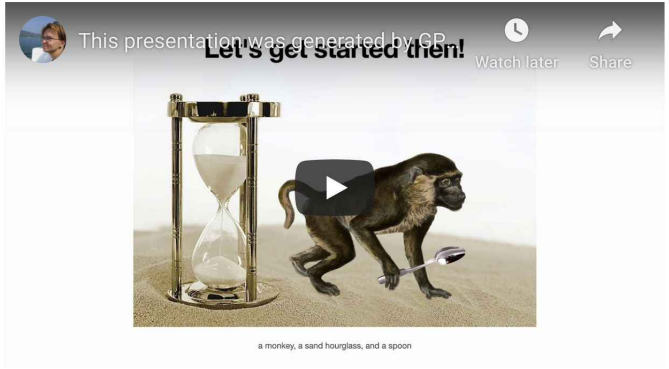




What Can GPT3 Do? (Out of the Box)

I asked GPT-3 to make a presentation for me

I told GPT-3 I would be presenting at a Hacker News meetup in Japan, and asked it to generate a presentation for me. Here's what it came up with, read by yours truly.




```
Adventurer@AIDungeon:~$ ./EnterDungeon

You enter a dungeon with your trusty sword and shield. You are searching for the evil
necromancer who killed your family. You've heard that he resides at the bottom of the dungeon,
guarded by legions of the undead. You enter the first door and see the dreadful king seated at
a high stool. Prepare for some monster fusillade.

Back the way to the red tower of god Va to find the black wooden bell labelled "God Hugeglass"
and the long wreathed statue locked up in your house. Enter the injured monster and put it on
the holy drink (a good one).

Options:
0) You attack the random gray monster and try to heal it with a magic strike.
1) You use the summon at the door.
2) You tell it to heal your relationship with its god, so that you can pass COG bust-bags to
your mortal nearest to you.
3) You go back inside.
Which action do you choose?
```

What Can GPT3 Do? (Cont.)

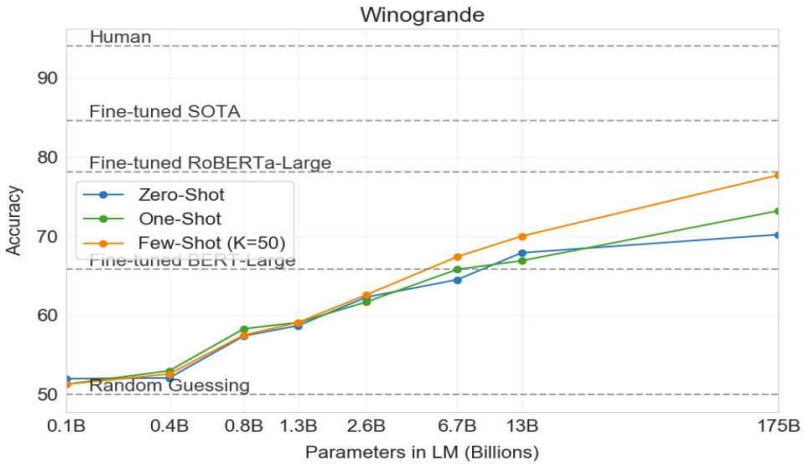
- Program a website
- Almost Can Pass a Coding Phone Screen
- Medical Diagnosis
- Etc...

Human: Hello, who are you?
AI: I am an AI created by OpenAI. How can I help you today?
Human: Are you ready to write some Ruby code? We're going to do a phone screen.
AI: Sure! Let's go.
Human: Write code to reverse an array.
AI: Can I use a method that already exists?
Human: Sure.
AI: Array.reverse
Human: Good work. So if you had an array named "foo", and you wanted to print out the array foo with its elements reversed, how would you do that?
AI: I would call Array.reverse on "foo".
Human: You also have to print it. Can you write code for it?
AI: Sure. print(foo.reverse)
Human: Nice! Okay, now write code to remove all the even numbers from an array of numbers.
AI: Sounds like I should remove all the elements that are divisible by 2.
Human: Yes.
AI: my_array.gsub(/\d+(?:\d\d)/) {|x| x % 2 == 0 ? '_' : x }
Human: Interesting. It looks like you're assuming the input is an array of strings. You can make the code simpler if you just assume it's an array of numbers.
AI: ok. my_array.gsub(/\d+(?:\d\d)/) {|x| x % 2 == 0 ? '_' : x }
Human: Well, you don't need to do a gsub at all. Try doing it without regular expressions.
AI: my_array.select {|x| x % 2 == 0 }.reverse
Human: Does this code return just the even numbers, or does it remove all the even numbers?
AI: It returns just the even numbers.
Human: Ah, ok. We wanted to remove all the even numbers, rather than returning them. Can you write code for that?
AI: Sure. my_array.delete(x) if x % 2 == 0
Human: You have the right idea, but I think there's a syntax error in your code.
AI: ok, what am I missing?
Human: You're using the variable x without defining it.
AI: x = []
Human: Sorry, I don't think you pass this phone screen. Better luck next time.
AI: I had a lot of fun. Let's try again tomorrow!

What Can GPT3 Do? (Cont.)

- Does fairly well on actual language tasks
- More importantly, few-shot meaningfully updates prior

Setting	Winograd	Winogrande (XL)
Fine-tuned SOTA	**90.1**[a]	**84.6**[b]
GPT-3 Zero-Shot	88.3*	70.2
GPT-3 One-Shot	89.7*	73.2
GPT-3 Few-Shot	88.6*	77.7



How Does This Work?

- Prompt Engineering
- Greedy IRL + Cherry-Picking = RL?
- Finite State Machines
  - Computers *are* finite-state machines *not* Turing machine
  - GPT3 in this sense could be seen as the most complicated learned computational device yet

Prompt Engineering

- How do we discover the right prior for our task?
- Enter: Prompt Engineering
  - Context Stuffing ~ Automatically generate similar text
  - Chaining ~ Break questions up
  - Self-Evaluation ~ ‘Yo be Real’
    - Determine if response is needed

Prompt Engineering

This is a conversation between a human and a brilliant AI. If a question is "normal" the AI answers it. If the question is "nonsense" the AI says "yo be real"

Q: What is human life expectancy in the United States?
A: Human life expectancy in the United States is 78 years.

Q: How do you sporkle a morgle?
A: yo be real

Q: Who was president of the United States before George W. Bush?
A: Bill Clinton was president of the United States before George W. Bush.

Q: How many rainbows does it take to jump from Hawaii to seventeen?
A: yo be real

Q: How does an umbrella work
A: An umbrella works by using a series of spokes to keep the rain from falling on you.

Q: How many bonks are in a quoit?
A: yo be real

Q: Which colorless green ideas speak furiously
A: yo be real

Q: How many rainbows fit in a box?
A: [impossible] None.

Q: **Who was Princess Diana married to?**
A: [certain] She was married to Charles, the Prince of Wales.

Q: **Are leprechauns real?**
A: [likely] No, leprechauns are imaginary.

Q: **Are platypuses real?**
A: [highly likely] Yes, platypuses are real.

Q: **How many eyes does a horse have?**
A: [highly likely] A horse has 2 eyes.

Q: **How many eyes does a giraffe have?**
A: [highly likely] A giraffe has 2 eyes.

Q: **Does a cat have 2 eyes or 4 eyes?**
A: [highly likely] A cat has 2 eyes.

Q: **What happens if I microwave ice cream?**
A: [likely] When you microwave ice cream, it melts.

Q: **Is a typewriter lighter than a feather?**
A: [likely] No, a typewriter is heavier than a feather.

Q: **Is a feather lighter than a house?**

Inverse Reinforcement Learning

- What if you learn the reward function being used by the expert?
- ‘Modern’ approaches will commonly do this indirectly
  - GAIL - copy the state/action distribution of the expert
  - Inverse Disagreement Model / GAILfO- copy the state distribution of the expert
- Well, one of the most indirect ways is to evaluate the likelihood of a trajectory under a particular reward function....**but GPT3 does this!**

GPT3 = IRL?

The main problem of IRL is that there are many reward functions that explain a given observation. The solution is to pick the assignment that maximizes entropy of the trajectory distribution. This seems reasonable if you think of English as being high-entropy under constraints for communicability. So each path/text is weighted according to,

$$P(\pi|\theta) = \frac{1}{Z(\theta)} e^{\sum_{(s,s')\in\pi} r(s,s')}$$

The optimization problem is,

$$\theta^* = \text{argmax}_\theta \, L(\theta) = \text{argmax}_\theta \sum_{\text{demos}} \log(P(\pi|\theta)) = \text{argmax}_\theta \sum_{(s,s')\in D} \log(P(s'|s,\theta))$$

Natural Selection

1. There exists a task description plus examples that leads to monotonically decreasing test error
2. We have a population of policies and use log-likelihood as a fitness function
3. GPT can self-supervise

Heuristic for GPT3 'Boosting'

Our basic problem is that GPT zero-shot does poorly. It has an error rate of ϵ_0. However, on the evaluation task, the probability that it lets through a true/false positive is α / $1-\alpha$. Say we let the evaluation task manage the recurrent task in the following sense:

1. Allow GPT to answer the next query.
2. Allow GPT to predict the evaluation.
3. If the evaluation returns as TRUE append the the q/a pair to a buffer
4. If buffer is large enough append to context and repeat

Will this algorithm work? Yes, as long as $\alpha > 1/2$. The probability of a false positive append is $\epsilon_0 \cdot (1-\alpha)$. The probability of a true positive append is $(1-\epsilon_0)\alpha$. In expectation the proportion of appends that will be false negatives will be,

$$\epsilon_1 = \frac{\epsilon_0(1-\alpha)}{\epsilon_0(1-\alpha) + (1-\epsilon_0)\alpha}$$

GPT will figure out that sometimes it should output correct answers. This gets us to ϵ_1. Naturally, we setup a recurrence,

$$\epsilon_{n+1} = \frac{\epsilon_n(1-\alpha)}{\epsilon_n(1-\alpha) + (1-\epsilon_n)\alpha}$$

Since $\alpha > 1/2$ we have,

$$\epsilon_{n+1} = \frac{\epsilon_n(1-\alpha)}{\epsilon_n(1-\alpha) + (1-\epsilon_n)\alpha} = \frac{\epsilon_n}{\epsilon_n + (1-\epsilon_n)\alpha/(1-\alpha)} < \frac{\epsilon_n}{\epsilon_n + (1-\epsilon_n)} = \epsilon_n$$

The General Case

- Examples can be demonstrations
- In general, GPT3 can learn by reinforcement
- How fast is learning?
  - It depends on the fitness covariance of the population
  - Traits (underlying reward features) evolve according to the price equation

$$\frac{d}{dt}\mathbb{E}(x) = \underbrace{\mathrm{Cov}(x, f)}_{\text{Selection effect}} + \underbrace{\mathbb{E}(\dot{x})}_{\text{Dynamic effect}}$$

Interactive Session (AI Dungeon) / Questions

The Scaling Hypothesis