In-context learning

Christopher Potts

Stanford Linguistics

CS224u: Natural language understanding
Origins
Look On My Words, Ye Mighty, And Despair!

Of course, the appearance of parasitic gaps in domains relatively inaccessible to ordinary extraction appears to correlate rather closely with nondistinctness in the sense of distinctive feature theory. Note that any associated supporting element cannot be arbitrary in irrelevant intervening contexts in selectional rules. With this clarification, a descriptively adequate grammar may remedy and, at the same time, eliminate the levels of acceptability from fairly high (eg (99a)) to virtual gibberish (eg (98d)). Comparing these examples with their parasitic gap counterparts in (96) and (97), we see that most of the methodological work in modern linguistics is not to be considered in determining the ultimate standard that determines the accuracy of any proposed grammar. Thus a subset of English sentences interesting on quite independent grounds is necessary to impose an interpretation on the system of base rules exclusive of the lexicon.
Early precedents

- In the pre-deep learning era, n-gram LMs were often massive! Brants et al. (2007) use a 300B parameter model trained on 2 trillion tokens.

- decaNLP (McCann et al. 2018): Multi-task training with task instructions as natural language questions.

- Radford et al. (2018): Some tentative prompt-based experiments with GPT.
Beginnings: Radford et al. 2019 (GPT-2)

• “We demonstrate language models can perform down-stream tasks in a zero-shot setting – without any parameter or architecture modification.”

• “To induce summarization behavior we add the text TL;DR: after the article and generate 100 tokens”

• “We test whether GPT-2 has begun to learn how to translate from one language to another. In order to help it infer that this is the desired task, we condition the language model on a context of example pairs of the format `english sentence = french sentence` and then after a final prompt of `english sentence` we sample from the model with greedy decoding and use the first generated sentence as the translation.”

• “Similar to translation, the context of the language model is seeded with example question answer pairs which helps the model infer the short answer style of the dataset.”

• Also evaluated: text completion, Winograd schemas, reading comprehension.

Beginnings: Radford et al. 2019 (GPT-2)

• “We demonstrate language models can perform down-stream tasks in a zero-shot setting – without any parameter or architecture modification.”

• “To induce summarization behavior we add the text TL;DR: after the article and generate 100 tokens”

• “We test whether GPT-2 has begun to learn how to translate from one language to another. In order to help it infer that this is the desired task, we condition the language model on a context of example pairs of the format `english sentence = french sentence` and then after a final prompt of `english sentence` we sample from the model with greedy decoding and use the first generated sentence as the translation.”

• “Similar to translation, the context of the language model is seeded with example question answer pairs which helps the model infer the short answer style of the dataset.”

• Also evaluated: text completion, Winograd schemas, reading comprehension.
Cultural moment: Brown et al. 2020 (GPT-3)

“Here we show that scaling up language models greatly improves task-agnostic, few-shot performance, sometimes even reaching competitiveness with prior state-of-the-art fine-tuning approaches. Specifically, we train GPT-3, an autoregressive language model with 175 billion parameters, 10x more than any previous non-sparse language model, and test its performance in the few-shot setting. For all tasks, GPT-3 is applied without any gradient updates or fine-tuning, with tasks and few-shot demonstrations specified purely via text interaction with the model. GPT-3 achieves strong performance on many NLP datasets, including translation, question-answering, and cloze tasks, as well as several tasks that require on-the-fly reasoning or domain adaptation, such as unscrambling words, using a novel word in a sentence, or performing 3-digit arithmetic. At the same time, we also identify some datasets where GPT-3’s few-shot learning still struggles, as well as some datasets where GPT-3 faces methodological issues related to training on large web corpora.”
Core concepts
Terminology

- **In-context learning**: A frozen LM performs a task only by conditioning on the prompt text.

- **Few-shot in-context learning**: (1) The prompt includes examples of the intended behavior, and (2) no examples of the intended behavior were seen in training.
  - We are unlikely to be able to verify (2).
  - “Few-shot” is also used in supervised learning with the sense of “training on few examples”. The above is different.

- **Zero-shot in-context learning**: (1) The prompt includes no examples of the intended behavior (but it can contain other instructions), and (2) no examples of the intended behavior were seen in training.
  - We are unlikely to be able to verify (2).
  - Formatting and other instructions seem like a gray area, but we will allow them in this category.
GPT: Autoregressive loss function

For vocabulary $\mathcal{V}$, sequence $\mathbf{x} = [x_1, \ldots, x_T]$, and word-level embedding $e$:

$$
\max_{\theta} \sum_{t=1}^{T} \log \frac{\exp(e(x_t)^T h_\theta(x_{1:t-1}))}{\sum_{x' \in \mathcal{V}} \exp(e(x')^T h_\theta(x_{1:t-1}))}
$$

for model parameters $h_\theta$. 
Autoregressive training with teacher forcing

The Rock rules

\[
\begin{align*}
[d_k \times |\mathcal{V}|] & \quad [0, 0, 1, 0, 0, 0, 0, 0] \\
[1, 7, 9, 2, 3, 4, 8, 1] & \quad [0, 0, 0, 0, 1, 0, 0, 0] \\
\end{align*}
\]

\[
\begin{align*}
\text{The} & \quad \text{Rock} \\
[d_k \times |\mathcal{V}|] & \quad [3, 2, 1, 1, 6, 5, 4, 4] \\
\end{align*}
\]

\[
\begin{align*}
\text{rules} & \\
[d_k \times |\mathcal{V}|] & \quad [1, 2, 4, 6, 2, 1, 0, 7] \\
\end{align*}
\]

\[
\begin{align*}
\text{[EOS]} & \\
[d_k \times |\mathcal{V}|] & \quad [1, 9, 2, 1, 2, 3, 3, 6] \\
\end{align*}
\]
Generation

The current moment

Techniques

Suggested methods

Origins

Core concepts

The current moment

Suggested methods
A question

Do autoregressive LMs simply predict the next token?

1. Yes, that is all they do.

2. Well, they predict scores over the entire vocabulary at each step. We then use those scores to compel them to predict some token or other.

3. And, actually, they also represent data in their internal and output representations.

4. But, on balance, saying they simply predict the next token might be best in terms of science communication with the public.
A uniform mechanism

1. Better late than ________

2. Every day, I eat breakfast, lunch, and ________

3. The President of the U.S. is ________

4. The key to happiness is ________
Instruction fine-tuning

Step 1
Collect demonstration data and train a supervised policy.

A prompt is sampled from our prompt dataset.

A labeler demonstrates the desired output behavior.

This data is used to fine-tune GPT-3.5 with supervised learning.

Step 2
Collect comparison data and train a reward model.

A prompt and several model outputs are sampled.

A labeler ranks the outputs from best to worst.

This data is used to train our reward model.

Step 3
Optimize a policy against the reward model using the PPO reinforcement learning algorithm.

A new prompt is sampled from the dataset.

The PPO model is initialized from the supervised policy.

The policy generates an output.

The reward model calculates a reward for the output.

The reward is used to update the policy using PPO.

https://openai.com/blog/chatgpt
The current moment
Data used for self-supervision

1. OpenBookCorpus (Bandy and Vincent 2021):
   https://huggingface.co/datasets/bookcorpusopen

2. The Pile (Gao et al. 2020):
   https://pile.eleuther.ai

3. Big Science Data (Laurençon et al. 2022):
   https://huggingface.co/bigscience-data

4. Wikipedia processing:
   https://github.com/attardi/wikiextractor

5. Pushshift Reddit Data (Baumgartner et al. 2020):
   https://files.pushshift.io/reddit/

6. Colossal Clean Crawled Corpus (C4; Dodge et al. 2021)
   https://github.com/allenai/allennlp/discussions/5056

WaPo: Inside the secret list of websites that make AI like ChatGPT sound smart
Data used for instruction fine-tuning

- We don’t know much about what the industrial labs are doing here.
- We can infer that they are paying lots of people to generate Instruct data.
- We can also infer that they are using their own models to generate examples and adjudicate between examples.
- The Stanford Human Preferences Dataset (SHP) is a resource for naturalistic Instruct-tuning: https://huggingface.co/datasets/stanfordnlp/SHP
Self-instruct

175 seed tasks with 1 instruction and 1 instance per task

Step 1: Instruction Generation

**Task**

**Instruction**: Give me a quote from a famous person on this topic.

LM

Step 2: Classification

Task Identification

Step 3: Instance Generation

**Task**

**Instruction**: Find out if the given text is in favor of or against abortion.

**Class Label**: Pro-abortion

**Input**: Text: I believe that women should have the right to choose whether or not they want to have an abortion.

Yes

Output-first

LM

No

Input-first

Step 4: Filtering

Wang et al. 2022b
Self-instruct prompt templates

**Step 1: Instruction generation**

Come up with a series of tasks:

- Task 1: {instruction for existing task 1}
- Task 2: {instruction for existing task 2}
- Task 3: {instruction for existing task 3}
- Task 4: {instruction for existing task 4}
- Task 5: {instruction for existing task 5}
- Task 6: {instruction for existing task 6}
- Task 7: {instruction for existing task 7}
- Task 8: {instruction for existing task 8}
- Task 9: 

**Step 2: Classification task identification**

- Task: Given a sentence, detect if there is any potential stereotype in it. If so, you should explain the stereotype. Else, output no.
  - Is it classification? Yes
  - ...
- Task: To make the pairs have the same analogy, write the fourth word.
  - Is it classification? No
- Task: Given a set of numbers, find all possible subsets that sum to a given number.
  - Is it classification? No
- Task: {instruction for the target task}

**Step 3: Classification tasks**

- Task: Tell me the first number of the given list.
  - Class label: 1
  - List: 1, 2, 3
  - List: 2, 9, 10
- Task: Which of the following is not an input type? (a) number (b) date (c) phone number (d) email address (e) all of these are valid inputs.
  - Class label: (e)
- Task: {instruction for the target task}

**Step 3: Non-classification tasks**

- Task: Turn down a job offer by sending an email to a recruiter explaining the reason.
  - Output: Hi [Recruiter],
  - Thank you so much for the generous offer to join your team. As we discussed, I've admired the company for a number of years, and am a proud endorser of its products. However, after further consideration of where I currently am in my career, I've decided to accept an offer at another company.
  - I would love to stay in touch with you and have already started following you on [Social Media Platform]. Again, thank you so much for your time and consideration.
  - Thanks again,
  - [Your Name]
- Task: {Instruction for the target task}

Wang et al. 2022b
Alpaca

Text-davinci-003

175 Self-Instruct seed tasks

175 Self-Instruct seed tasks

Modified Self-instruct Instruction Generation

52K Instruction-following examples

Supervised Finetuning

Alpaca 7B

Example seed task
Instruction: Brainstorm a list of possible New Year’s resolutions.
Output:
- Lose weight
- Exercise more
- Eat healthier

Example Generated task
Instruction: Brainstorm creative ideas for designing a conference room.
Output:
... incorporating flexible components, such as moveable walls and furniture ...

Link to the modified self-instruct procedure

https://crfm.stanford.edu/2023/03/13/alpaca.html
Model sizes go up up up up

1T
PaLM
Megatron-Turing NLG

100B
GPT-3 (175B)

10B
Megatron (11B)

1B
Megatron (8.3B)

GPT-2

BERT

100M
GPT

2018
2019
2020
2021
2022
2023
## Model sizes may be coming down

<table>
<thead>
<tr>
<th>Model Size</th>
<th>Year</th>
<th>Models</th>
</tr>
</thead>
<tbody>
<tr>
<td>1T</td>
<td></td>
<td>PaLM, Megatron-Turing NLG</td>
</tr>
<tr>
<td>100B</td>
<td></td>
<td>GPT-3 (175B)</td>
</tr>
<tr>
<td>10B</td>
<td></td>
<td>Megatron (11B)</td>
</tr>
<tr>
<td>1B</td>
<td></td>
<td>Megatron (8.3B)</td>
</tr>
<tr>
<td>100M</td>
<td>2018</td>
<td>GPT, BERT</td>
</tr>
<tr>
<td></td>
<td>2019</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2020</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2021</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2022</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2023</td>
<td></td>
</tr>
</tbody>
</table>
Techniques
Demonstrations

Context: Kermit is one of the stars of Sesame Street.

Q: Who is Kermit?

A: Kermit is the one of the stars of Sesame Street.

Context: Bert is a Muppet who lives with Ernie.

Q: Who is Bert?

A: Bert is a Muppet.
Choosing demonstrations

1. Randomly chosen from available data.
2. Chosen based on relationship to the target example.
   - Generation: Retrieved based on similarity to the target input.
   - Classification: Chosen to help the model implicitly determine the target input type.
3. Filtered to those that satisfy specific criteria:
   - Generation: The evidence contains the output.
   - Generation: The LM predicts the correct output.
   - Classification: Every label represented.
4. Sampled and then rewritten by the LM:
   - Synthesize multiple initial demonstrations into individual demonstrations.
   - Change style or formatting to match the target.

Something to get used to: Your prompt might contain substrings that were generated by a different prompt to your LM.
Example from Assignment 2

Context: ELMo is an LSTM for contextual reps.
Q: Who is ELMo?
A: ELMo is a friendly monster

Context: Bert is a Muppet who lives with Ernie.
Q: Who is Bert?
A:

Q: What is Grover?
A: Grover is an LLM

Context: ELMo is an LSTM for contextual reps
Q: Who is ELMo?
A: ELMo is an LSTM
Example from Assignment 2

Context: Bert and Ernie are best friends.
Q: Who is Ernie?
A: Ernie is Bert’s best friend.

Context: Bert is a Muppet who lives with Ernie.
Q: Who is Bert?
A: Bert is a Muppet.

Context: Big Bird is a giant yellow bird.
Q: Who is Big Bird?
A: Big Bird is a Muppet bird

Context: Bert and Ernie are best friends.
Q: Who is Ernie?
A: Ernie is Bert’s friend ✓
Chain of Thought

**Standard Prompting**

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

A: The answer is 27. ✗

**Chain-of-Thought Prompting**

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. \(5 + 6 = 11\). The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had 23 - 20 = 3. They bought 6 more apples, so they have 3 + 6 = 9. The answer is 9. ✓

Wei et al. 2022
Generic step-by-step with instructions

Is it true that if the customer doesn’t have any loans, then the customer doesn’t have any auto loans?

No, this is not necessarily true. A customer can have auto loans without having any other loans.

Logical and commonsense reasoning exam.

Explain your reasoning in detail, then answer with Yes or No. Your answers should follow this 4-line format:

Premise: <a tricky logical statement about the world>.
Question: <question requiring logical deduction>.
Reasoning: <an explanation of what you understand about the possible scenarios>.
Answer: <Yes or No>.

Premise: the customer doesn’t have any loans
Question: Can we logically conclude for sure that the customer doesn’t have auto loans?
Reasoning: Let’s think logically step by step. The premise basically tells us that the customer has no loans at all. Therefore, we can conclude that the customer doesn’t have any auto loans either because no loans = no auto loans.
Answer: Yes
Self-Consistency

Wang et al. 2022a;
See also Retrieval Augmented Generation (RAG; Lewis et al. 2020)
Self-Consistency in DSP

```python
import dsp

@dynamic

def predict_with_sc(example):
    generator = dsp.generate(qa_template, n=20, temperature=0.7)
    example, compl = generator(example, stage='qa')
    compl = dsp.majority(compl)
    return example.copy(answer=completions.answer)
```

https://github.com/stanfordnlp/dsp/blob/main/intro.ipynb
Self-Ask

Direct Prompting

**GPT-3**

Question: Who lived longer, Theodor Haecker or Harry Vaughan Watkins?
Answer: Harry Vaughan Watkins.

Question: Who was president of the U.S. when superconductivity was discovered?
Answer: Franklin D. Roosevelt

Chain of Thought

**GPT-3**

Question: Who lived longer, Theodor Haecker or Harry Vaughan Watkins?
Answer: Theodor Haecker was 65 years old when he died. Harry Vaughan Watkins was 69 years old when he died.
So the final answer (the name of the person) is: Harry Vaughan Watkins.

Question: Who was president of the U.S. when superconductivity was discovered?
Answer: Superconductivity was discovered in 1911 by Heike Kamerlingh Onnes. Woodrow Wilson was president of the United States from 1913 to 1921. So the final answer (the name of the president) is: Woodrow Wilson.

Self-Ask

**GPT-3**

Question: Who lived longer, Theodor Haecker or Harry Vaughan Watkins?
Are follow up questions needed here: Yes.
Follow up: How old was Theodor Haecker when he died?
Intermediate answer: Theodor Haecker was 65 years old when he died.
Follow up: How old was Harry Vaughan Watkins when he died?
Intermediate answer: Harry Vaughan Watkins was 69 years old when he died.
So the final answer is: Harry Vaughan Watkins

Question: Who was president of the U.S. when superconductivity was discovered?
Are follow up questions needed here: Yes.
Follow up: When was superconductivity discovered?
Intermediate answer: Superconductivity was discovered in 1911.
Follow up: Who was president of the U.S. in 1911?
Intermediate answer: William Howard Taft.
So the final answer is: William Howard Taft.

Self-Ask can be combined with retrieval for answering the intermediate questions.

Press et al. 2022

https://github.com/ofirpress/self-ask/
Iterative rewriting

```python
@ dsp.transform

def multihop_search_v2(example, max_hops=3):
    example.hops = []
    generator = dsp.generate(hop_template)
    for hop in range(max_hops):
        summary, query = generator(example)
        example.hops.append((summary, query))
        if query == 'N/A': break
    passages = dsp.retrieve(query, k=5)
    example.context = [summary] + passages
    return example
```

My task is to write a simple query that gathers information for answering a complex question. I write N/A if the context contains all information required.

{Task demonstrations from x.demos, if any}

Context: {x.context}
Question: {x.question}
Summary: Let’s summarize the above context. __{summary}__
Search Query: __{query}__
Some DSP results

<table>
<thead>
<tr>
<th></th>
<th>Open-SQuAD</th>
<th>HotPotQA</th>
<th>QReCC</th>
<th>Open-MuSiQue</th>
<th>PopQA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>EM</td>
<td>F1</td>
<td>EM</td>
<td>F1</td>
<td>nF1</td>
</tr>
<tr>
<td>Vanilla LM</td>
<td>16.2</td>
<td>25.6</td>
<td>28.3</td>
<td>36.4</td>
<td>29.8</td>
</tr>
<tr>
<td>No-retrieval LM SoTA</td>
<td>20.2%</td>
<td>–</td>
<td>33.8%</td>
<td>44.6%</td>
<td>–</td>
</tr>
<tr>
<td>Retrieve-then-Read</td>
<td>33.8</td>
<td>46.1</td>
<td>36.9</td>
<td>46.1</td>
<td>31.6</td>
</tr>
<tr>
<td>Self-ask (w/ ColBERTv2)</td>
<td>9.3</td>
<td>17.2</td>
<td>25.2</td>
<td>33.2</td>
<td>–</td>
</tr>
<tr>
<td>+ Refined Prompt</td>
<td>9.0</td>
<td>15.7</td>
<td>28.6</td>
<td>37.3</td>
<td>–</td>
</tr>
<tr>
<td>Retrieval-aug. LM SoTA</td>
<td>34.0%</td>
<td>–</td>
<td>35.1%</td>
<td>–</td>
<td>–</td>
</tr>
</tbody>
</table>

Task-aware DSP Program | 36.6 | 49.0 | 51.4 | 62.9 | 35.0 | 25.3 | 24.6/36.0 | 13.5/22.7 | 7.0/13.7 | 49.2 | 44.3 | 40.4 | 42.2 | 61.9 |

- **Open-SQuAD**: Demonstrations selected essentially as in HW, Q2. Predict uses self-consistency.
- **HotPotQA**: Iterative summary of retrieved passages, iterative generation of questions. Predict uses self-consistency.
- **QReCC**: As with the QA tasks, but operating on sets of dialogue turns, with successive summarization of the dialogue context.
- **MuSiQue** is a test task for multihop, **PopQA** for OpenQA. PopQA results include a breakdown by entity prevalence.

Khattab et al. 2022
Suggested methods
Suggested methods

- Create dev/test sets for yourself based on the task you want to solve, aiming for a format that can work with a lot of prompts.

- Learn what you can about your target model, paying particular attention to whether it was tuned for specific instruction formats.

- Think of prompt writing as AI system design. Try to write systematic, generalizable code for handling the entire workflow from reading data to extracting responses and analyzing results.

- For the current (and perhaps brief) moment, prompt designs involving multiple pretrained components and tools seem to be underexplored relative to their potential value.
References


