In-context Learning

CSCI 601 471/671
NLP: Self-Supervised Models

https://self-supervised.cs.jhu.edu/sp2023/

[Slide credit: Iz Beltagy, Arman Cohan, Robert Logan IV, Sewon Min, Sameer Singh, and many others]
Logistics

- HW7 is up! Due Thu March 30.

- Projects: Please continue to brainstorm!
  - Project proposal deadline: Thu March 30.

- Midterms grades?
HW 1-5
News: GPT-4 was released!!

- “Transformer-style model pretrained to predict next token”
- We don’t know the size 😞
- We don’t know the amount of supervision 😞

This report focuses on the capabilities, limitations, and safety properties of GPT-4. GPT-4 is a Transformer-style model [33] pre-trained to predict the next token in a document, using both publicly available data (such as internet data) and data licensed from third-party providers. The model was then fine-tuned using Reinforcement Learning from Human Feedback (RLHF) [34]. Given both the competitive landscape and the safety implications of large-scale models like GPT-4, this report contains no further details about the architecture (including model size, dataset construction, training method, or similar).

- From a company name “Open”AI — the irony
News: GPT-4 was released!!

This report focuses on the capabilities, limitations, and safety properties of GPT-4. GPT-4 is a Transformer-style model [33] pre-trained to predict the next token in a document, using both publicly available data (such as internet data) and data licensed from third-party providers. The model was then fine-tuned using Reinforcement Learning from Human Feedback (RLHF) [34]. Given both the competitive landscape and the safety implications of large-scale models like GPT-4, this report contains no further details about the architecture (including model size), hardware, training compute, dataset construction, training method, or similar.

- It is trained with human feedback (RLHF) — we will discuss it in a few weeks.
- It is trained on multi-modal signals — we will discuss it in a few weeks.

In-Context Learning

- Learns to do a downstream task by conditioning on input-output examples!

- **No weight update** — our model is not explicitly pre-trained to learn from examples
  - The underlying models are quite general

- Today’s focus:
  - How to use effectively in practice?
  - Fundamentally, why does it work?
How/Why does In-context Learning Work?

Any arbitrary task → Language Model → A few-shot learner

[ACL 2022 Tutorial Beltagy, Cohan, Logan IV, Min and Singh]
In-context Learning as Bayesian Inference

- (Xie et al., 2022) try to explain ICL as an implicit Bayesian inference.

Idea:
- (Pre-trained LM learn to represent “concepts”, i.e. the ideas described by words.
- ICL enables LMs to “locate” the learned concepts.

- Can formulate this intuition as a Bayesian inference
  - Prior over latent “concepts”
  - Likelihood describes connection between text and concepts
  - Given an incomplete doc, use Bayes formula to infer what concept is likely it is generated from and then complete the document.

- Does not explain everything.
  - GPT-3 can handle “unseen” concepts
In-context Learning as Gradience Descent

- ICL is implicitly equivalent to SGD on in-context demonstrations

\[ t_\theta(x_{\text{query}}) \]

Find \( \theta \)

s.t. \( t_\theta(x_{\text{query}}; D^{\text{context}}) \approx y_{\text{query}} \)

\[ D^{\text{context}} \]

Transformer

\[ \theta \]

\[ x_{\text{query}} \]

[von Oswald et al. 2022; Akyurek et al. 2022; Dai et al. 2022, ...]
Summary & Open questions

- In-context learning has been a promising few-shot learning approach
  - No need for gradient updates → Much easier to use large models!

- Better calibration, better scoring of model outputs, and better formation of demonstrations lead to great improvements
  - How to make it less sensitive?
  - How to scale it (longer context, more training examples, wider range of tasks)?

- Still in progress ...
  - Understanding how/why it works,
  - Disentangling looking up task location vs learning a new task
  - Can we predict whether in-context learning would work on a given task or not?
Prompting for Multi-Step Reasoning
Some Problems Involve Reasoning

Q: If there are 3 cars in the parking lot and 2 more cars arrive, how many cars are in the parking lot?
A: The answer is 5

Q: Take the last letters of the words in "Elon Musk" and concatenate them
A: The answer is nk.

Q: What home entertainment equipment requires cable?
Answer Choices: (a) radio shack (b) substation (c) television (d) cabinet
A: The answer is (c).

Arithmetic Reasoning (AR) (+−×÷...)
Symbolic Reasoning (SR)
Commonsense Reasoning (CR)
Reasoning Problems

- Fine-tune LMs on GSM8K (arithmetic reasoning)

- One may conjecture that, to achieve >80%, one needs **100x more training data** for 175B model

- Another option is to **increase model sizes**, which is expensive.

- Other than these, how else can we improve the model performance on tasks that require multi-step reasoning?

(Cobbe et al. 2021)
Reasoning Problems via Multi-Step Prompting

- **Basic idea:** Rather than showing input-output pairs, prompting the model such that it shows its proof steps.

- **Note:** ideas around models that are capable of multi-step reasoning go way back.
  - Aristotle (deduction),
  - Hume (induction),
  - Peirce (abduction)
  - Lots of other works in pre-LM era
  - Namely, my Ph.D. thesis on multi-step reasoning in semantic representations of language

[Reasoning-Driven Question-Answering for Natural Language Understanding]

**Deduction**
- All beans in that bag are white.
- These beans are from that bag.
- Therefore, these beans are white.

**Induction**
- These beans are from that bag.
- These beans are white.
- Therefore, all beans in that bag are white.

**Abduction**
- These beans are white.
- All beans in that bag are white.
- Therefore, these beans are from that bag.
Reasoning Problems via Multi-Step Prompting

(a) Few-shot

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: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A:

(Output) The answer is 8. X
Reasoning Problems via Multi-Step Prompting

(a) Few-shot

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: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A:

(Output) The answer is 8. ✗

(b) Few-shot-CoT (Wei et al., 2022)

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: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A:

(Output) The juggler can juggle 16 balls. Half of the balls are golf balls. So there are $16 / 2 = 8$ golf balls. Half of the golf balls are blue. So there are $8 / 2 = 4$ blue golf balls. The answer is 4. ✓

Step-by-step demonstration

Step-by-step Answer
Reasoning Problems via Multi-Step Prompting

(a) Few-shot

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: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A: 

(Output) The answer is 8. 

(b) Few-shot-CoT (Wei et al., 2022)

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: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A: 

(Output) The juggler can juggle 16 balls. Half of the balls are golf balls. So there are $16 / 2 = 8$ golf balls. Half of the golf balls are blue. So there are $8 / 2 = 4$ blue golf balls. The answer is 4.

(c) Zero-shot

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A: The answer (arabic numerals) is

(Output) 8

Step-by-step demonstration

Step-by-step Answer
Reasoning Problems via Multi-Step Prompting

(a) Few-shot

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: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A:

(Output) The answer is 8. X

(b) Few-shot-CoT (Wei et al., 2022)

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: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A:

(Output) The juggler can juggle 16 balls. Half of the balls are golf balls. So there are 16 / 2 = 8 golf balls. Half of the golf balls are blue. So there are 8 / 2 = 4 blue golf balls. The answer is 4. ✓

(c) Zero-shot

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A: The answer (arabic numerals) is

(Output) 8 X

(d) Zero-shot-CoT (Kojima et al., 2022)

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A: Let's think step by step.

(Output) There are 16 balls in total. Half of the balls are golf balls. That means that there are 8 golf balls. Half of the golf balls are blue. That means that there are 4 blue golf balls. ✓
Multi-Step Prompting: Empirical Results

- **Setup:** show demonstrations that contain the decompositions
- The gains of multi-step prompting increases with scale.
- Prompting achieves **better perf than** [smaller] models that are fine-tuned on a lot more data.

["Chain of thought prompting elicits reasoning in large language models", Wei et al. 2022]
Multi-Step Prompting: Empirical Results

- **Setup:** show demonstrations that contain the decompositions
- The gains of multi-step prompting increases with scale.
- Prompting achieves better perf than [smaller] models that are fine-tuned on a lot more data.

Figure 3: Model scale study with various types of models. S: text-ada-001, M: text-babbage-001, L: text-curie-001, XL: text-davinci-002. See Appendix A.3 and E for the detail.

["Large Language Models are Zero-Shot Reasoners", Kojima et al. 2022]
Multi-Step Prompting: Steps Don’t Have to Be Correct!

- It is possible even with invalid demonstrations
- Prompting with invalid reasoning steps can achieve over 80-90% of the performance with correct reasoning steps.

![Graph showing performance comparison between standard, CoT, and invalid reasoning steps.](image)

["Towards Understanding Chain-of-Thought Prompting", Wang et al. 2022]
Multi-Step Prompting: Parting Comments

- Prompting LMs to explain their reasoning improves their performance.
- However, their steps aren’t always correct.
  - A useful repository of annotation: https://github.com/OpenBioLink/ThoughtSource

- There is much to research on here:
  - When do LMs over-reason or under-reason?
  - How do adjust the granularity of step?
  - How to use given references in the proofs?
  - How do use external “tools” (e.g., logic, calculator, Python) in forming proofs?
Parameter-Efficient Tuning of LMs

CSCI 601 471/671
NLP: Self-Supervised Models

https://self-supervised.cs.jhu.edu/sp2023/

[Slide credit: Iz Beltagy, Arman Cohan, Robert Logan IV, Sewon Min, Sameer Singh, Danqi Chen and many others ]
Fine-tuning Pre-trained Models

A general recipe:
- Pre-train a language model
- Fine-tune a classification head on top of the LMs representations

[ACL 2022 Tutorial Beltagy, Cohan, Logan IV, Min and Singh]
Fine-tuning Pre-trained Models

Default finetuning recommendations are unstable in few-shot settings.

Stability can be improved by:
- Using smaller learning rates
- Training for more iterations
- ...

However finetuning still underperforms other methods.

---

ACL 2022 Tutorial Beltagy, Cohan, Logan IV, Min and Singh
Prompt Tuning

- Learn embeddings for placeholder tokens in the pattern.

- Variants:
  - WARP [Hambardzumyan et al., 2021]
  - OptiPrompt [Zhong et al., 2021]
  - Prompt Tuning [Lester et al., 2021]
  - P-Tuning* [Li et al., 2021]

[ACL 2022 Tutorial Beltagy, Cohan, Logan IV, Min and Singh]
Prompt Tuning: Effect of Prompt Length

- The shorter the prompt, the fewer new parameters must be tuned
- Increasing prompt length is critical to achieving good performance
- The largest model still gives strong results with a single-token prompt
- Increasing beyond 20 tokens only yields marginal gains

[The Power of Scale for Parameter-Efficient Prompt Tuning, Lester et al. 2021]
BitFit

- BitFit adds bias terms in self-attention and MLP layers and tunes those.

\[
Q^{m,\ell}(x) = W_{q}^{m,\ell}x + b_{q}^{m,\ell}
\]
\[
K^{m,\ell}(x) = W_{k}^{m,\ell}x + b_{k}^{m,\ell}
\]
\[
V^{m,\ell}(x) = W_{v}^{m,\ell}x + b_{v}^{m,\ell}
\]

\[
h_{2}^{\ell} = \text{Dropout}(W_{m_{1}}^{\ell} \cdot h_{1}^{\ell} + b_{m_{1}}^{\ell})
\]
\[
h_{3}^{\ell} = g_{LN_{1}}^{\ell} \odot \frac{(h_{2}^{\ell} + x) - \mu}{\sigma} + b_{LN_{1}}^{\ell}
\]
\[
h_{4}^{\ell} = \text{GELU}(W_{m_{2}}^{\ell} \cdot h_{3}^{\ell} + b_{m_{2}}^{\ell})
\]
\[
h_{5}^{\ell} = \text{Dropout}(W_{m_{3}}^{\ell} \cdot h_{4}^{\ell} + b_{m_{3}}^{\ell})
\]
\[
\text{out}^{\ell} = g_{LN_{2}}^{\ell} \odot \frac{(h_{5}^{\ell} + h_{3}^{\ell}) - \mu}{\sigma} + b_{LN_{2}}^{\ell}
\]

["BitFit: Simple Parameter-efficient Fine-tuning for Transformer-based Masked Language-models" Ben Zaken et al., 2021]
Adapters

- **Core idea:** train small sub-networks and only tune those.
- No need to store a full model for each task, only the adapter params.

[“Parameter-Efficient Transfer Learning for NLP” Houlsby et al., 2019.]
(IA)$^3$: Infused Adapter by Inhibiting and Amplifying Inner Activations

- Element-wise rescaling of model activations with a learned vector:
  - keys and values in self-attention
  - feed-forward networks

"Few-Shot Parameter-Efficient Fine-Tuning is Better and Cheaper than In-Context Learning" Liu et al., 2022.
Prompt Tuning: Interpretability

- Are continuous prompts interpretable?

Sentence: That was a great fantasy movie.

\[ p^* \text{: optimized for the task} \]

Something related to sentiment analysis? 🤔

Opposite goal: how unfaithful can their interpretation be of what they do?

["Prompt Waywardness: The Curious Case of Discretized Interpretation of Continuous Prompts" Khashabi et al., 2022.]
nearest-neighbor mapping of continuous prompt onto the word embeddings

\( \tilde{p} \): optimized for the task + project to a given text

\( p^* \): optimized for the task

Sentence: That was a great fantasy movie.

definition of another task:
Write down the conclusion you can reach by combining the given Fact 1 and Fact 2.

random sentence from web:
int clamp(int val, int min_val) {
    return std::max(min_val, val);
}

["Prompt Waywardness: The Curious Case of Discretized Interpretation of Continuous Prompts" Khashabi et al., 2022.]
Sentence: That was a great fantasy movie.

\[ \tilde{p} \]: optimized for the task + project to a given text

\[ p^* \]: optimized for the task

\[ \Delta \approx 0.6\% \]

["Prompt Waywardness: The Curious Case of Discretized Interpretation of Continuous Prompts" Khashabi et al., 2022.]
What is the sentiment of the following review? (positive or negative)

Sentence: That was a great fantasy movie.

- **discrete (text) prompts:**
  - easy to interpret, but not easy to optimize

Sentence: That was a great fantasy movie.

- **LM**
  - positive

- **continuous prompts:**
  - unclear how to interpret, but easy to optimize

Sentence: That was a great fantasy movie.

- **LM**
  - positive

"Prompt Waywardness: The Curious Case of Discretized Interpretation of Continuous Prompts" Khashabi et al., 2022.
Open questions & future work

- Parameter efficient optimization — optimize fewer parameters than the whole model.
  - Space efficiency — fewer parameters to store
  - Computation efficiency? A bit unclear

- Their interpretability is not quite clear.

- Open research question: How to bridge the gap between continuous prompts vs discrete prompts?