Thoughts on NLP Research in the (Post-)LLM* Era

Yijia Shao
Yuanpei College
2023/04/28

*LLM: Large Language Model
Overview

• NLP tasks in the pre-LLM era
• Introduction to LLMs
• View LLMs from a system perspective
• Open questions
Typical NLP Tasks in the Pre-LLM Era

- **Text classification**: Assigning a label or class to a given text.
- E.g. Sentiment Analysis: class labels are sentiment polarities

https://huggingface.co/tasks/text-classification
Typical NLP Tasks in the Pre-LLM Era

• **Question answering**: Returning an answer in text form to a given question also in text form.
• E.g. Extractive QA: the answer is extracted from a given context

![Diagram showing inputs and outputs of question answering](https://huggingface.co/tasks/question-answering)
Typical NLP Tasks in the Pre-LLM Era

• **Semantic Parsing**: Converting a natural language utterance to a logical form.

• E.g. Text-to-SQL: convert a natural language question to a SQL query

```sql
SELECT T2.name, T2.budget
FROM instructor as T1 JOIN department as T2
ON T1.department_id = T2.id
GROUP BY T1.department_id
HAVING avg(T1.salary) >
    (SELECT avg(salary) FROM instructor)
```
Typical NLP Tasks in the Pre-LLM Era

• And SO MANY...

Wang et al. “Super-NaturalInstructions: Generalization via Declarative Instructions on 1600+ NLP Tasks” EMNLP 2022
Traditional ML Paradigm

- Supervised data + algorithm -> model
  - Design **specific algorithms** for each task and train **separate models**.

Dang et al. “Sentiment Analysis Based on Deep Learning: A Comparative Study” Electronics 2020

Chen et al. “Reading Wikipedia to Answer Open-Domain Questions” ACL 2017

Why do some researchers feel panic when ChatGPT/LLMs came out?
One LLM for All

• Studying a specific task becomes less meaningful.

**Sentiment analysis classifier**

Decide whether a Tweet's sentiment is positive, neutral, or negative.

Tweet: "I loved the new Batman movie!"
Sentiment: Positive

**Q&A**

Chatbot: I am a **ML/AI language model tutor**
You: What is a language model?
Chatbot: A language model is a statistical model that describes the probability of a word given the previous words.

**Translation / NL2code**

Create a SQL request to find all users who live in California and have over 1000 credits:

```
SELECT * FROM users WHERE state='CA' AND credits > 1000;
```

Semantic of query Syntax of code

**Summarization**

A neutron star is the collapsed core of a massive supergiant star, which had a total mass of between 10 and 25 solar masses, possibly more if the star was especially metal-rich. Neutron stars are the smallest and densest stellar objects, excluding black holes and hypothetical white holes, quark stars, and strange stars. Neutron stars have a radius on the order of 10 kilometres (6.2 mi) and a mass of about 1.4 solar masses. They result from the supernova explosion of a massive star, combined with gravitational collapse, that compresses the core past white dwarf star density to that of atomic nuclei.

**TL;DR:** A neutron star is the collapsed core of a massive supergiant star. These ultra-dense objects are incredibly fascinating due to their strange properties and their potential for phenomena such as extreme gravitational forces and a strong magnetic field.

Examples from [https://platform.openai.com/examples](https://platform.openai.com/examples), this slide is adapted from Stanford CS 329X slides
“Eureka” Moment and Paradigm Shift

• From expert-defined tasks to **user-defined tasks**.

---

**ChatGPT PLUS**

<table>
<thead>
<tr>
<th>Examples</th>
<th>Capabilities</th>
<th>Limitations</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Explain quantum computing in simple terms?&quot; →</td>
<td>Remembers what user said earlier in the conversation</td>
<td>May occasionally generate incorrect information</td>
</tr>
<tr>
<td>&quot;Got any creative ideas for a 10 year old's birthday?&quot; →</td>
<td>Allows user to provide follow-up corrections</td>
<td>May occasionally produce harmful instructions or biased content</td>
</tr>
<tr>
<td>&quot;How do I make an HTTP request in Javascript?&quot; →</td>
<td>Trained to decline inappropriate requests</td>
<td>Limited knowledge of world and events after 2021</td>
</tr>
</tbody>
</table>
Emergent Reasoning Ability

- LLMs show “unexpected” reasoning ability and exceed average human performance on many standard exams.

<table>
<thead>
<tr>
<th>Exam</th>
<th>GPT-4</th>
<th>GPT-4 (no vision)</th>
<th>GPT-3.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniform Bar Exam (MBE+MEE+MP)</td>
<td>298 / 400 (~90th)</td>
<td>298 / 400 (~90th)</td>
<td>213 / 400 (~10th)</td>
</tr>
<tr>
<td>LSAT</td>
<td>163 (~88th)</td>
<td>161 (~83rd)</td>
<td>149 (~40th)</td>
</tr>
<tr>
<td>SAT Evidence-Based Reading &amp; Writing</td>
<td>710 / 800 (~93rd)</td>
<td>710 / 800 (~93rd)</td>
<td>670 / 800 (~87th)</td>
</tr>
<tr>
<td>SAT Math</td>
<td>700 / 800 (~89th)</td>
<td>690 / 800 (~89th)</td>
<td>590 / 800 (~70th)</td>
</tr>
<tr>
<td>Graduate Record Examination (GRE) Quantitative</td>
<td>163 / 170 (~80th)</td>
<td>157 / 170 (~62nd)</td>
<td>147 / 170 (~25th)</td>
</tr>
<tr>
<td>Graduate Record Examination (GRE) Verbal</td>
<td>169 / 170 (~90th)</td>
<td>165 / 170 (~90th)</td>
<td>154 / 170 (~63rd)</td>
</tr>
<tr>
<td>Graduate Record Examination (GRE) Writing</td>
<td>4 / 6 (~54th)</td>
<td>4 / 6 (~54th)</td>
<td>4 / 6 (~54th)</td>
</tr>
<tr>
<td>USABO Semifinal Exam 2020</td>
<td>87 / 150 (99th - 100th)</td>
<td>87 / 150 (99th - 100th)</td>
<td>43 / 150 (31st - 33rd)</td>
</tr>
<tr>
<td>USNCO Local Section Exam 2022</td>
<td>36 / 60</td>
<td>38 / 60</td>
<td>24 / 60</td>
</tr>
<tr>
<td>Medical Knowledge Self-Assessment Program</td>
<td>75 %</td>
<td>75 %</td>
<td>53 %</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Exam</th>
<th>GPT-4</th>
<th>GPT-4 (no vision)</th>
<th>GPT-3.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>AP Environmental Science</td>
<td>5 6/15 (~100th)</td>
<td>5 6/15 (~100th)</td>
<td>5 6/15 (~100th)</td>
</tr>
<tr>
<td>AP Macroeconomics</td>
<td>5 8/15 (~100th)</td>
<td>5 8/15 (~100th)</td>
<td>5 8/15 (~100th)</td>
</tr>
<tr>
<td>AP Microeconomics</td>
<td>5 8/15 (~100th)</td>
<td>4 6/15 (~82nd)</td>
<td>4 6/15 (~82nd)</td>
</tr>
<tr>
<td>AP Physics 2</td>
<td>4 6/15 (~84th)</td>
<td>4 6/15 (~84th)</td>
<td>3 3/15 (~66th)</td>
</tr>
<tr>
<td>AP Psychology</td>
<td>5 8/15 (~100th)</td>
<td>5 8/15 (~100th)</td>
<td>5 8/15 (~100th)</td>
</tr>
<tr>
<td>AP Statistics</td>
<td>5 8/15 (~100th)</td>
<td>5 8/15 (~100th)</td>
<td>5 8/15 (~100th)</td>
</tr>
<tr>
<td>AP US Government</td>
<td>5 8/15 (~100th)</td>
<td>5 8/15 (~100th)</td>
<td>4 7/15 (~88th)</td>
</tr>
<tr>
<td>AP US History</td>
<td>5 8/15 (~100th)</td>
<td>4 7/15 (~89th)</td>
<td>4 7/15 (~89th)</td>
</tr>
<tr>
<td>AP World History</td>
<td>4 6/15 (~87th)</td>
<td>4 6/15 (~87th)</td>
<td>4 6/15 (~87th)</td>
</tr>
<tr>
<td>AMC 10&lt;sup&gt;1&lt;/sup&gt;</td>
<td>30 / 150 (60th - 120th)</td>
<td>36 / 150 (10th - 19th)</td>
<td>36 / 150 (10th - 19th)</td>
</tr>
<tr>
<td>AMC 12&lt;sup&gt;2&lt;/sup&gt;</td>
<td>60 / 150 (45th - 66th)</td>
<td>48 / 150 (19th - 40th)</td>
<td>30 / 150 (4th - 8th)</td>
</tr>
</tbody>
</table>

**Table 1.** GPT performance on academic and professional exams. In each case, we simulate the conditions and scoring of the real exam. We report GPT-4’s final score graded according to exam-specific rubrics, as well as the percentile of test-takers achieving GPT-4’s score.
Emergent Reasoning Ability

• This proposes a great challenge to **evaluate our systems** (or even us).
• Can we say the Turing test is passed? If so, what’s next?

<table>
<thead>
<tr>
<th>Exam</th>
<th>GPT-4</th>
<th>GPT-4 (no vision)</th>
<th>GPT-3.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniform Bar Exam (MBE+MEE+MPT)</td>
<td>298 / 400 (~90th)</td>
<td>298 / 400 (~90th)</td>
<td>213 / 400 (~10th)</td>
</tr>
<tr>
<td>LSAT</td>
<td>163 (~88th)</td>
<td>161 (~83rd)</td>
<td>149 (~40th)</td>
</tr>
<tr>
<td>SAT Evidence-Based Reading &amp; Writing</td>
<td>710 / 800 (~93rd)</td>
<td>710 / 800 (~93rd)</td>
<td>670 / 800 (~87th)</td>
</tr>
<tr>
<td>SAT Math</td>
<td>700 / 800 (~89th)</td>
<td>690 / 800 (~89th)</td>
<td>590 / 800 (~70th)</td>
</tr>
<tr>
<td>Graduate Record Examination (GRE) Quantitative</td>
<td>163 / 170 (~80th)</td>
<td>157 / 170 (~62nd)</td>
<td>147 / 170 (~25th)</td>
</tr>
<tr>
<td>Graduate Record Examination (GRE) Verbal</td>
<td>169 / 170 (~99th)</td>
<td>165 / 170 (~99th)</td>
<td>154 / 170 (~63rd)</td>
</tr>
<tr>
<td>Graduate Record Examination (GRE) Writing</td>
<td>4 / 6 (~54th)</td>
<td>4 / 6 (~54th)</td>
<td>4 / 6 (~54th)</td>
</tr>
<tr>
<td>USABO Semifinal Exam 2020</td>
<td>87 / 150 (99th - 100th)</td>
<td>87 / 150 (99th - 100th)</td>
<td>43 / 150 (31st - 33rd)</td>
</tr>
<tr>
<td>USNCO Local Section Exam 2022</td>
<td>36 / 60</td>
<td>38 / 60</td>
<td>24 / 60</td>
</tr>
<tr>
<td>Medical Knowledge Self-Assessment Program</td>
<td>75 %</td>
<td>75 %</td>
<td>53 %</td>
</tr>
</tbody>
</table>

| Codeforces Rating                        | 392 (below 5th) | 392 (below 5th) | 260 (below 5th) |
| AP Art History                            | 5 (66th - 100th) | 5 (68th - 100th) | 5 (68th - 100th) |
| AP Biology                                | 5 (85th - 100th) | 5 (85th - 100th) | 4 (62nd - 85th) |
| AP Calculus BC                            | 4 (43nd - 59th)  | 4 (43nd - 59th)  | 1 (6th - 7th)  |
| AP Chemistry                              | 4 (71st - 88th)  | 4 (71st - 88th)  | 2 (22nd - 46th) |
| AP English Language and Composition        | 2 (14th - 44th)  | 2 (14th - 44th)  | 2 (14th - 44th) |
| AP English Literature and Composition      | 2 (8th - 22nd)   | 2 (8th - 22nd)   | 2 (8th - 22nd) |

Table 1. GPT performance on academic and professional exams. In each case, we simulate the conditions and scoring of the real exam. We report GPT-4’s final score graded according to exam-specific rubrics, as well as the percentile of test-takers achieving GPT-4’s score.
Is LLM a pure engineering success?
Introduction to Large Language Models

- Latest LLMs adopt the **Transformer** backbone.
- Core component: self-attention mechanism
  - Put tokens into their **context**!

Vaswani et al. “Attention is all you need” *NIPS 2017*
Attention as a soft, averaging lookup table

We can think of **attention** as performing fuzzy lookup in a key-value store.

In a **lookup table**, we have a table of **keys** that map to **values**. The **query** matches one of the keys, returning its value.

In **attention**, the **query** matches all **keys** softly, to a weight between 0 and 1. The keys’ **values** are multiplied by the weights and summed.
Self-Attention: keys, queries, values from the same sequence

Let \( \mathbf{w}_{1:n} \) be a sequence of words in vocabulary \( V \), like Zuko made his uncle tea.

For each \( \mathbf{w}_i \), let \( \mathbf{x}_i = \mathbf{Ew}_i \), where \( \mathbf{E} \in \mathbb{R}^{d \times |V|} \) is an embedding matrix.

1. Transform each word embedding with weight matrices \( Q, K, V \), each in \( \mathbb{R}^{d \times d} \)

\[
\mathbf{q}_i = Q\mathbf{x}_i \quad \text{(queries)} \quad \mathbf{k}_i = K\mathbf{x}_i \quad \text{(keys)} \quad \mathbf{v}_i = V\mathbf{x}_i \quad \text{(values)}
\]

2. Compute pairwise similarities between keys and queries; normalize with softmax

\[
\mathbf{e}_{ij} = \mathbf{q}_i^\top \mathbf{k}_j \quad \alpha_{ij} = \frac{\exp(\mathbf{e}_{ij})}{\sum_j \exp(\mathbf{e}_{ij'})}
\]

3. Compute output for each word as weighted sum of values

\[
\mathbf{o}_i = \sum_j \alpha_{ij} \mathbf{v}_i
\]
Self-Attention: Fully-connected Graph in One Pass

Another way to understand attention is to leverage the perspective of graph.

- Consider a sequence as a fully-connected graph $K_n$, where each vertex corresponds to a token in the sequence.
- Assign $v_i$ as the value of the $i$-th vertex, and $q_i^T k_j$ as the weight of the edge $e_{ij}$. The attention calculation is iterating the value of each vertex using the weighted average of the values of its connected vertices.
  - All vertices can be updated in parallel. (GPU-friendly!)
  - It’s easy to manipulate information flow. (Add mask to the weight of $e_{ij}$.)

Bring in Other Components

Introducing nonlinearity to make piling up multiple attention layers non-trivial.

A token may need to look at multiple places in the sentence at once.
- Define multiple attention heads through multiple Q,K,V matrices.

Attention mechanism doesn’t have an inherent notion of order.
- Add positional encoding to the inputs.

\[
PE_{(pos,2i)} = \sin \left( \frac{pos}{10000^{2i/d_{model}}} \right) \\
PE_{(pos,2i+1)} = \cos \left( \frac{pos}{10000^{2i/d_{model}}} \right)
\]
The Transformer was born 6 years ago...

• There isn’t much improvement in the model architecture.

Google Study Shows Transformer Modifications Fail To Transfer Across Implementations and Applications. Mar 3, 2021

ACL Anthology
https://aclanthology.org/2021.emnlp-main.46... PDF

Do Transformer Modifications Transfer Across ...
by S Narang · 2021 · Cited by 48 — Another possible explanation is that the modifications proposed to the Transformer do not “generalize” across Page 2 5759 applications, i.e. ...
16 pages

• Maybe it’s because the Transformer is powerful enough.
  • Theoretically proved: Transformers with trainable positional encodings are universal approximators of continuous sequence-to-sequence functions on a compact domain. (Yun et al., 2019)
The Transformer was born 6 years ago...

- In these years, to better use the Transformer, researchers have been working on
  - Designing optimizers which are more suitable to the Transformer: e.g., AdamW (Loshchilov and Hutter, 2018)
  - Designing parallel computing algorithm to make training larger models possible: e.g., model parallel (Megatron-LM, Shoeybi et al., 2020)
The Transformer was born 6 years ago...

• In these years, to better use the Transformer, researchers have been working on
  • Reducing the computational overhead: e.g., faster layer normalization (Zhang and Sennrich, 2019), sparse attention (recall the perspective of graph)
  • Designing methods to improve training stability: e.g., modified initialization (GPT-2, GPT-3)
Great. Dependency parsing and coreference resolution are almost solved. Syntactic information is captured in attention (Clark et al., 2019).

Great. Most of current NLP benchmarks focus on this part.

It depends! LLMs still get confused when they meet unique contexts or special users (e.g., those in underrepresented groups).

• Level of linguistic knowledge

Great. LLMs are robust to typos, coinage, cacography
How do LLMs acquire the knowledge of language?

• **Unsupervised pre-training** on very large corpus
  • There are many pre-training methods, here we focus on the one used by the GPT family.
  • Language modeling: predict the next word

![](The%20illustrated%20GPT-2.png)
How do LLMs acquire the knowledge of language?

There’s increasing evidence that pretrained models learn a wide variety of things about the statistical properties of language. Taking our examples from the start of class:

- *Stanford University is located in __________, California.* [Trivia]
- *I put ___ fork down on the table.* [syntax]
- *The woman walked across the street, checking for traffic over ___ shoulder.* [coreference]
- *I went to the ocean to see the fish, turtles, seals, and ____.* [lexical semantics/topic]
- *Overall, the value I got from the two hours watching it was the sum total of the popcorn and the drink. The movie was ___.* [sentiment]
- Iroh went into the kitchen to make some tea. Standing next to Iroh, Zuko pondered his destiny. Zuko left the _____. [some reasoning – this is harder]
- *I was thinking about the sequence that goes 1, 1, 2, 3, 5, 8, 13, 21, ____.* [some basic arithmetic; they don’t learn the Fibonnaci sequence]
- Models also learn – and can exacerbate racism, sexism, all manner of bad biases.

These examples are copied from Stanford CS224N/Ling284 slides (author: John Hewitt). They are actually examples for **masked language modeling** which is a bit different from how GPT is pre-trained.
LLMs learn more than the knowledge of language

“A language is not just words. It's a culture, a tradition, a unification of a community, a whole history that creates what a community is. It's all embodied in a language.”

—— Noam Chomsky
LLMs learn more than the knowledge of language

• Dig into the training corpus of LLaMA, an open-sourced English LLM.
  • English CommonCrawl [67%]: webpages, covering nearly everything you can think of right now.
  • C4 [15%]: a colossal, cleaned version of Common Crawl corpus.
  • Github [4.5%]: training on code may greatly benefit LLMs’ reasoning ability (Fu et al., 2022)
  • Wikipedia [4.5%]: high quality knowledge source
  • Gutenberg and Books3 [4.5%]: containing formal writings in various disciplines
  • ArXiv [2.5%]: academy papers
  • Stack Exchange [2%]: high quality questions and answers

• Besides publicly available data, OpenAI includes their proprietary data in training.
LLMs learn more than the knowledge of language

Open question:

• Are LLMs stochastic parrots or world models?
From GPT-3 to ChatGPT: Learn human intents behind their language

<table>
<thead>
<tr>
<th>Prompt</th>
<th>Explain the moon landing to a 6 year old in a few sentences.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Completion</td>
<td>GPT-3</td>
</tr>
<tr>
<td></td>
<td>Explain the theory of gravity to a 6 year old.</td>
</tr>
<tr>
<td></td>
<td>Explain the theory of relativity to a 6 year old in a few sentences.</td>
</tr>
<tr>
<td></td>
<td>Explain the big bang theory to a 6 year old.</td>
</tr>
<tr>
<td></td>
<td>Explain evolution to a 6 year old.</td>
</tr>
<tr>
<td>InstructGPT</td>
<td>People went to the moon, and they took pictures of what they saw, and sent them back to the earth so we could all see them.</td>
</tr>
</tbody>
</table>

Information behind this sentence: People usually use imperative sentence to make a request. The listener is expected to complete that request.
Figure 2: A diagram illustrating the three steps of our method: (1) supervised fine-tuning (SFT), (2) reward model (RM) training, and (3) reinforcement learning via proximal policy optimization (PPO) on this reward model. Blue arrows indicate that this data is used to train one of our models. In Step 2, boxes A-D are samples from our models that get ranked by labelers. See Section 3 for more details on our method.
Follow Instructions & Align with Human Preference

Human-in-the-loop!
(Discuss more later)

Figure 2: A diagram illustrating the three steps of our method: (1) supervised fine-tuning (SFT), (2) reward model (RM) training, and (3) reinforcement learning via proximal policy optimization (PPO) on this reward model. Blue arrows indicate that this data is used to train one of our models. In Step 2, boxes A-D are samples from our models that get ranked by labelers. See Section 3 for more details on our method.

Ouyang et al. “Training language models to follow instructions with human feedback” NIPS 2022
Introduction to Large Language Models

Hugging face “Large Language Models: A New Moore's Law?” 2021

GPT-4?
Introduction to Large Language Models

- In-context learning
  - No parameter update
  - Wrap “training” samples in the prompt

Circulation revenue has increased by 5% in Finland. // Positive
Panostaja did not disclose the purchase price. // Neutral
Paying off the national debt will be extremely painful. // Negative
The company anticipated its operating profit to improve. // ________

Circulation revenue has increased by 5% in Finland. // Finance
They defeated ... in the NFC Championship Game. // Sports
Apple ... development of in-house chips. // Tech
The company anticipated its operating profit to improve. // ________

The gif is copied from https://ai.stanford.edu/blog/understanding-incontext/
Open question:
- Why does in-context learning work?

There are some hypotheses but no conclusion yet:
- Xie et al. “An Explanation of In-context Learning as Implicit Bayesian Inference” *ICLR 2022*
- Oswald et al. “Transformers learn in-context by gradient descent” *Arxiv 2022*
Introduction to **Large Language Models**

- **Emergence abilities**
  - An ability is emergent if it is not present in smaller models but is present in larger models.
- In-context learning ability is one of them.
- **Scaling** to improve unlock abilities.

**Emergence in few-shot prompting**

This gif is copied from Jason Wei's slides.

Wei et al. “Emergent Abilities of Large Language Models” *TMLR 2022*
Introduction to **Large** Language Models

3 Predictable Scaling

A large focus of the GPT-4 project was building a deep learning stack that scales predictably. The primary reason is that for very large training runs like GPT-4, it is not feasible to do extensive model-specific tuning. To address this, we developed infrastructure and optimization methods that have very predictable behavior across multiple scales. These improvements allowed us to reliably predict some aspects of the performance of GPT-4 from smaller models trained using $1,000 \times - 10,000 \times$ less compute.

3.1 Loss Prediction

The final loss of properly-trained large language models is thought to be well approximated by power laws in the amount of compute used to train the model [41, 42, 2, 14, 15].

To verify the scalability of our optimization infrastructure, we predicted GPT-4’s final loss on our internal codebase (not part of the training set) by fitting a scaling law with an irreducible loss term (as in Henighan et al. [15]): $L(C) = aC^b + c$, from models trained using the same methodology but using at most $10,000 \times$ less compute than GPT-4. This prediction was made shortly after the run started, without use of any partial results. The fitted scaling law predicted GPT-4’s final loss with high accuracy (Figure 1).

3.2 Scaling of Capabilities on HumanEval

Having a sense of the capabilities of a model before training can improve decisions around alignment, safety, and deployment. In addition to predicting final loss, we developed methodology to predict more interpretable metrics of capability. One such metric is pass rate on the HumanEval dataset [43].
View LLMs from a system perspective

• Analogy: operating system (OS)
  • Knowing a set of algorithms is **not** enough to build a good OS.
  • Knowing a training algorithm/recipe is **not** enough to build a good LLM.

• **Model patching & continual training** of LLM are important.
  • We shouldn’t always build a new LLM from scratch.
  • I think this may be one reason for OpenAI’s success – they build LLMs as building a system (maintenance, version control, incremental update)
Yao Fu “How does GPT Obtain its Ability? Tracing Emergent Abilities of Language Models to their Sources”
Put LLMs into a Larger System

• Analogy: operating system (OS)
  • How do we interact with OS?
  • How do we interact with LLMs?
Put LLMs into a Larger System

• Analogy: operating system (OS)
  • How do we interact with OS?
  • How do we interact with LLMs?

This part is now also considered as a part of the OS in general.
• Make the system more accessible, especially for non-computer experts.
The user briefly describes his/her goal. AutoGPT breaks the goal into detailed steps and refine its own plan.
Put LLMs into a Larger System

Looking to eat vegan food in San Francisco this weekend. Could you get me one great restaurant suggestion for Saturday and a simple recipe for Sunday (just the ingredients)? Please calculate the calories for the recipe using WolframAlpha. Finally order the ingredients on Instacart.

LLM functions as a controller and can use tool on its own.
LLM as a Controller

I'm inspired by https://oval.cs.stanford.edu/ to add this illustration.
LLM as a Controller: Challenges

• How to design the interaction interface between LLMs and other components (e.g., external databases, API schemas)?
  • Desiderata:
    robustness, unambiguity, privacy-protecting, easy-to-build for non-AI developers

• How to maintain the state of LLM?
  • Naïve solution: Cramming all the previous contexts into the prompt.
  • Problems:
    The sequence length is limited (recall the attention mechanism).
    Multiple individual calls to the LLM cause great overhead.

I'm inspired by https://oval.cs.stanford.edu/ to add this illustration.
Bring Human into the Loop

• Returning to the OS analogy
• What’s special with LLMs?
  • **LLMs can learn from the human-model interaction and evolve.**

This part is now also considered as a part of the OS in general.
• Make the system more accessible, especially for non-computer experts.
Bring Human into the Loop

Core challenges:

• How can we let human easily provide feedback?
  • Exploiting cheap labor is unethical and infeasible to collecting domain-specific feedbacks.
  • I think research from the HCI side is important.

• How can we let the LLM take feedback?
  • Current approach: RLHF
  • What’s next? (distinct challenges exist)

Chen et al. “Perspectives on Incorporating Expert Feedback into Model Updates” Arxiv 2022
Distinct Challenges in Learning from Human Feedback

• **Human feedback is noisy**. The model should decide whether to take the feedback rather than viewing it as the ground truth.

• **out-of-distribution detection** -> “out-of-confidence” detection
  - In OOD detection, we design algorithm to assign a score to an instance to indicate how much it belongs to the training distribution, or in other words, how much the model should be capable of predicting its label.
  - I think the LLM should also assign a confidence score to the input question.
Model “Model Confidence”

• The confidence score may be broken into two parts:
  • uncertainty about the user’s goal (intrinsic to the input question)
  • confidence in its answer (related to the sampling in the output generation)
Model “Model Confidence”

- I found Anthropic has done initial work on this.
  - Their approach is asking these two questions to the LLM itself. (Similar to reflection)
  - Many limitations exist: infinite recursion, generalization problem, etc.

3 From Calibration to Knowing What You Know
  3.1 Replacing an Option with ‘None of the Above’ Harms Performance and Calibration
  3.2 Models are Well-Calibrated on True/False Tasks
  3.3 RLHF Policy Miscalibration Can Be Remediated with a Temperature Tuning

4 Ask the AI: Is your proposed answer True or False?
  4.1 Basic Self-Evaluation
  4.2 Showing Many $T = 1$ Samples Improves Self-Evaluation
Recap

• LLMs trigger a paradigm shift.
  • Users define tasks.
  • New evaluation methods are needed.

• LLM is not a pure engineering success.
  • The Transformer architecture is powerful.
  • Tracing LLMs’ abilities back to the data source and training objectives.
  • Emergent abilities and the scaling law is still mysterious.

• We may view LLMs from a system perspective.
Q&A

Happy for further discussion: shaoyj@pku.edu.cn