Developing and studying instruction-following models

Tatsunori Hashimoto, Stanford CS

Future of Decentralization, AI, and Computing Summit
LLMs in the spotlight

Google PaLM on BigBench

GPT4 on a range of exams

Impressive, ongoing advances in NLP and AI from large language models!
These models are increasingly closed off

“On the competitive landscape front — it's competitive out there,” said Sutskever. “GPT-4 is not easy to develop. It took pretty much all of OpenAI working together for a very long time to produce this thing. And there are many many companies who want to do the same thing, so from a competitive side, you can see this as a maturation of the field.”

Jan Leike  @janleike · Oct 24, 2022
I agree. While OpenAI doesn't like talking about exact model sizes / parameter counts anymore, documentation should definitely be better.

text-davinci-002 isn't the model from the InstructGPT paper. The closest to the paper is text-davinci-plus-002.
Closed models are hard to study and improve

API-only access makes it difficult to do deep analysis or propose improvements

Dual-use / misuse [Kang 2023]

Political values / biases [Santurkar 2023]
Reproducible low-cost environments for LLM experiments

Reproducing instruction-following models

- **Cost**: high cost of human annotation
- **Replicability**: crowdsourcing doesn’t replicate
- **Reference**: no known working PPO implementation

**Why is this hard?** Figuring this out (in full) requires replicating instructGPT/chatGPT
Alpaca trio: low-cost experiments for instruction-following

Step 1 (SFT) – Alpaca
Step 2 (RLHF) – AlpacaFarm
Step 3 (Evals) - AlpacaEval

Simulating annotators (via GPT4) enables fast, low-cost prototyping and R&D of LLMs

[Dubois, Li, Taori, Zhang et al 2023]
Validating the accuracy of simulated annotations

Agreement near human inter-annotator levels

Near-perfect rank correlation at the system level

[Dubois, Li, Taori, Zhang et al 2023]
### High-performance, reference methods for RLHF

<table>
<thead>
<tr>
<th>Method</th>
<th>Simulated win-rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPT-4</td>
<td>79.0 ± 1.4</td>
</tr>
<tr>
<td>ChatGPT</td>
<td>61.4 ± 1.7</td>
</tr>
<tr>
<td>PPO</td>
<td>46.8 ± 1.8</td>
</tr>
<tr>
<td>Best-of-(n)</td>
<td>45.0 ± 1.7</td>
</tr>
<tr>
<td>Expert Iteration</td>
<td>41.9 ± 1.7</td>
</tr>
<tr>
<td>SFT 52k (Alpaca 7B)</td>
<td>39.2 ± 1.7</td>
</tr>
<tr>
<td>SFT 10k</td>
<td>36.7 ± 1.7</td>
</tr>
<tr>
<td>Binary FeedME</td>
<td>36.6 ± 1.7</td>
</tr>
<tr>
<td>Quark</td>
<td>35.6 ± 1.7</td>
</tr>
<tr>
<td>Binary Reward Conditioning</td>
<td>32.4 ± 1.6</td>
</tr>
<tr>
<td>Davinci001</td>
<td>24.4 ± 1.5</td>
</tr>
<tr>
<td>LLaMA 7B</td>
<td>11.3 ± 1.1</td>
</tr>
</tbody>
</table>

Our findings replicate RLHF’s effectiveness, and these results hold outside the simulator.

[Dubois, Li, Taori, Zhang et al 2023]
High-performance, reference methods for RLHF

<table>
<thead>
<tr>
<th>Method</th>
<th>Simulated win-rate (%)</th>
<th>Human win-rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPT-4</td>
<td>79.0 ± 1.4</td>
<td>69.8 ± 1.6</td>
</tr>
<tr>
<td>ChatGPT</td>
<td>61.4 ± 1.7</td>
<td>52.9 ± 1.7</td>
</tr>
<tr>
<td>PPO</td>
<td>46.8 ± 1.8</td>
<td>55.1 ± 1.7</td>
</tr>
<tr>
<td>Best-of-(n)</td>
<td>45.0 ± 1.7</td>
<td>50.7 ± 1.8</td>
</tr>
<tr>
<td>Expert Iteration</td>
<td>41.9 ± 1.7</td>
<td>45.7 ± 1.7</td>
</tr>
<tr>
<td>SFT 52k (Alpaca 7B)</td>
<td>39.2 ± 1.7</td>
<td>40.7 ± 1.7</td>
</tr>
<tr>
<td>SFT 10k</td>
<td>36.7 ± 1.7</td>
<td>44.3 ± 1.7</td>
</tr>
<tr>
<td>Binary FeedME</td>
<td>36.6 ± 1.7</td>
<td>37.9 ± 1.7</td>
</tr>
<tr>
<td>Quark</td>
<td>35.6 ± 1.7</td>
<td>-</td>
</tr>
<tr>
<td>Binary Reward Conditioning</td>
<td>32.4 ± 1.6</td>
<td>-</td>
</tr>
<tr>
<td>Davinci001</td>
<td>24.4 ± 1.5</td>
<td>32.5 ± 1.6</td>
</tr>
<tr>
<td>LLaMA 7B</td>
<td>11.3 ± 1.1</td>
<td>6.5 ± 0.9</td>
</tr>
</tbody>
</table>

Our findings replicate RLHF’s effectiveness, and these results hold outside the simulator.

[Dubois, Li, Taori, Zhang et al 2023]
AlpacaFarm highlights the complexity of instruction RLHF

AlpacaFarm replicates important, complex phenomena like overoptimization

[Dubois, Li, Taori, Zhang et al 2023]
Beyond this work: LLM driven prototyping lowers the cost of R&D

Studying fine-tuning data

Development metrics

Caveat: development and deployment needs more than automated data/evals

Development metrics, synthetic data → Crowdsourced data + evaluation → Live evaluation
Case study: watermarking LLMs

Watermarking enables tracking of LLM-generated text (see Kirchenbauer et al)

Challenges:
- Watermarks induce distortion (hard sell for LLM vendors)
- Many watermarks highly non-robust (to deletion of a few words, or cropping)
Development of a distortion-free, robust watermark.

In recent work [Kuditipudi et al 2023], we derive a distortion free and robust watermark.

**Generate** (for each token $y_i$)
- Draw a random sequence $\xi_i \in [0,1]$, call this the key
- Sample according to $\min_i -\log \xi_i / p_i$ (From Aaronson)

  *This is distortion free* (i.e. the marginal distribution over $\xi$ is $p$)

**Detect**
- Find the min-Levenshtein cost with $d(y, \xi) = \sum_i \log(1 - \xi_i y_i)$
- Compare vs the min-Levenshtein cost w/ random $\xi$

  *This is robust* (i.e. can detect under small Levenshtein edits)

[Kuditipudi et al 2023]
**Watermarks and open models**

**Open models** (and access to logprobs) enable watermarking research

Distortion-free watermarks are weaker on instruction-tuned models

Distortion-inducing watermarks lead to major drops in performance

Unwatermarked: 70.86%  
KGW watermark: 51.2%

[Kuditipudi et al 2023]  
[Freeman and Hashimoto, unpublished]
Takeaways

**Open models and trustworthiness**

Open source provides important accountability and transparency

**Research on LLMs**

LLMs enable new research into instruction-following models

**Enabling safer and more robust LLMs**

New innovations and interventions based on open LLMs