CS388: Natural Language Processing

Lecture 13:
Instruction Tuning, RLHF, Dialog

Greg Durrett

Announcements

- Project 3 out now
- We highly recommend Colab
- You don’t need all training iterations
- You can decrease the frequency of checkpointing
- Project 2 back soon
- Final project proposals back soon

Recap: Chain-of-thought

<table>
<thead>
<tr>
<th>Prompt</th>
<th>Context: Christopher agrees with Kevin. [...] Q: Who hangs out with a student?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>Mary, because Mary hangs out with Danielle and Danielle is a student.</td>
</tr>
<tr>
<td>Label+</td>
<td></td>
</tr>
<tr>
<td>Train Ex</td>
<td></td>
</tr>
<tr>
<td>Train Ex</td>
<td></td>
</tr>
<tr>
<td>Test Input</td>
<td>Context: Adam plays with Ellen. […] Q: Who plays with a doctor?</td>
</tr>
<tr>
<td>Output</td>
<td>Adam, because Adam plays with Ellen and Ellen is a doctor. greedy decoding from GPT-3</td>
</tr>
</tbody>
</table>

Recap: Chain-of-thought

- Can help substantially on mathematical reasoning
- Some work to optimize the specifics of the prompts and the examples

Wei et al. (2022)
Today

- Instruction tuning
- RLHF
- Chatbots
- Task-oriented dialogue systems

Instruction Tuning

- We want to optimize models for $P(\text{answer} \mid \text{prompt, input})$, but they’re learned on a basic language modeling objective
- One solution: treat the basic language modeling as pre-training, then fine-tune them on what we care about
- Two versions of this:
  - **Instruction tuning**: supervised fine-tuning on data derived from many NLP tasks
  - **Reinforcement learning from human feedback (RLHF)**: RL to improve human judgments of how good the outputs are

Task Generalization: T0

- T0: tries to deliver on the goal of T5 and do many tasks with one model
- **Crowdsourced prompts**: instructions for how to do the tasks

- **Summarization**: The picture appeared on the wall of a Poundland store on Whymark Avenue [...] How would you rephrase that in a few words?
- **Paraphrase identification**: “How is air traffic controlled?” “How do you become an air traffic controller?” Pick one: these questions are duplicates or not duplicates.
- **Question answering**: I know that the answer to “What team did the Panthers defeat?” is in “The Panthers finished the regular season [...]”. Can you tell me what it is?
Task Generalization

- Pre-train: T5 task
- Train: a collection of tasks with prompts. This uses existing labeled training data
- Test: a new task specified only by a new prompt. No training data in this task

Frontiers

- Flan-PaLM (October 20, 2022): 1800 tasks, 540B parameter model fine-tuned on many tasks after pre-training
  Instruction finetuning
  - Please answer the following question: What is the boiling point of nitrogen?
  Chain-of-thought finetuning
  - Answer the following question by reasoning step-by-step:
    The cafeteria had 23 apples. If they used 20 for lunch and bought 6 more, how many apples do they have?
  Multi-task instruction finetuning (1.8K tasks)

Frontiers

- Flan-PaLM (October 20, 2022): 1800 tasks, 540B parameter model
  - MMLU task (Hendrycks et al., 2020): 57 high school/college/professional exams:

<table>
<thead>
<tr>
<th>Conceptual Physics</th>
<th>MMLU task (Hendrycks et al., 2020): 57 high school/college/professional exams:</th>
</tr>
</thead>
<tbody>
<tr>
<td>When you drop a ball from rest it accelerates downward at 9.8 m/s². If you instead throw it downward assuming no air resistance its acceleration immediately after leaving your hand is (A) 9.8 m/s² ✔️ (B) more than 9.8 m/s² ✘ (C) less than 9.8 m/s² ✘ (D) Cannot say unless the speed of throw is given. ✘</td>
<td>- Random 25.0</td>
</tr>
<tr>
<td>College Mathematics</td>
<td>- Average human rater 34.5</td>
</tr>
<tr>
<td>In the complex z-plane, the set of points satisfying the equation z² -</td>
<td>z</td>
</tr>
<tr>
<td></td>
<td>Mar. 2022 Chinchilla 5-shot 67.6</td>
</tr>
<tr>
<td></td>
<td>Apr. 2022 PaLM 5-shot 69.3</td>
</tr>
<tr>
<td></td>
<td>Oct. 2022 Flan-PaLM 5-shot 72.2</td>
</tr>
<tr>
<td></td>
<td>Flan-PaLM 5-shot: CoT + SC 75.2</td>
</tr>
<tr>
<td></td>
<td>- Average human expert 89.8</td>
</tr>
</tbody>
</table>

Figure 4: Examples from the Conceptual Physics and College Mathematics STEM tasks.
**Frontiers**

<table>
<thead>
<tr>
<th>Model</th>
<th>Finetuning Mixtures</th>
<th>Tasks</th>
<th>Norm. avg.</th>
<th>MMLU</th>
<th>BBH</th>
</tr>
</thead>
<tbody>
<tr>
<td>540B</td>
<td>None (no finetuning)</td>
<td>0</td>
<td>49.1</td>
<td>71.3</td>
<td>62.9</td>
</tr>
<tr>
<td>CoT</td>
<td>9</td>
<td>52.6 (+3.5)</td>
<td>68.8</td>
<td>64.8</td>
<td>50.5</td>
</tr>
<tr>
<td>CoT, Muffin</td>
<td>89</td>
<td>57.0 (+7.9)</td>
<td>71.8</td>
<td>66.7</td>
<td>56.7</td>
</tr>
<tr>
<td>CoT, Muffin, T0-SF</td>
<td>282</td>
<td>57.5 (+8.4)</td>
<td>72.9</td>
<td>68.2</td>
<td>57.3</td>
</tr>
<tr>
<td>CoT, Muffin, T0-SF, NIV2</td>
<td>1,836</td>
<td>58.5 (+9.4)</td>
<td>73.2</td>
<td>68.1</td>
<td>58.8</td>
</tr>
</tbody>
</table>

- Human performance estimates are ~80 on Big-Bench (BBH)

*Chung et al. (2022)*

**Flan-T5**

- Flan-T5: T5-11B model given the “Flan treatment”, instruction tuned on many tasks
- Best model at the ~10B scale for few-shot prompting, also very good choice for fine-tuning
- If you have the resources, Flan-T5 is something you can explore in your project/research!

*Chung et al. (2022)*

**RLHF**

- Apply this approach to optimizing outputs from large language models
- Step 3 (not shown): do RL with this policy

*Ouyang et al. (2022)*

---

**Reinforcement Learning from Human Feedback (RLHF)**

<table>
<thead>
<tr>
<th>Params</th>
<th>Model</th>
<th>Norm. avg.</th>
<th>MMLU</th>
<th>BBH</th>
</tr>
</thead>
<tbody>
<tr>
<td>11B</td>
<td>T5-XXL</td>
<td>-2.9</td>
<td>25.9</td>
<td>18.7</td>
</tr>
<tr>
<td></td>
<td>Flan-T5-XXL</td>
<td>23.7 (+26.6)</td>
<td>55.1</td>
<td>48.6</td>
</tr>
</tbody>
</table>

(about 20% behind the 540B Flan-PaLM)

*Ouyang et al. (2022)*
Humans produce comparisons of two trajectories (= outputs from systems) — different from standard reward in RL

- Fit the reward function $r$ using supervised estimation:

$$
\hat{p}[\sigma^1 > \sigma^2] = \frac{\exp \sum \hat{r}(o^1_i, a^1_i)}{\exp \sum \hat{r}(o^1_i, a^1_i) + \exp \sum \hat{r}(o^2_i, a^2_i)}.
$$

- This turns scores into log probabilities of 1 being preferred to 2. Same as logistic regression where we classify pairs as $1 > 2$ or $2 < 1$, but we actually learn a continuous scoring function, not a classifier

- The rest of the RL setup is TRPO/PPO, fairly standard frameworks (note: they typically constrain the policy to not deviate too far from a basic supervised policy)

For OpenAI, RLHF data is collected from their API. Very different from instruct-tuning datasets

Table 1: Distribution of use case categories from our API prompt dataset.

<table>
<thead>
<tr>
<th>Use-case</th>
<th>(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generation</td>
<td>45.6%</td>
</tr>
<tr>
<td>Open QA</td>
<td>12.4%</td>
</tr>
<tr>
<td>Brainstorming</td>
<td>11.2%</td>
</tr>
<tr>
<td>Chat</td>
<td>8.4%</td>
</tr>
<tr>
<td>Rewrite</td>
<td>6.6%</td>
</tr>
<tr>
<td>Summarization</td>
<td>4.2%</td>
</tr>
<tr>
<td>Classification</td>
<td>3.5%</td>
</tr>
<tr>
<td>Other</td>
<td>3.5%</td>
</tr>
<tr>
<td>Closed QA</td>
<td>2.6%</td>
</tr>
<tr>
<td>Extract</td>
<td>1.9%</td>
</tr>
</tbody>
</table>

Table 2: Illustrative prompts from our API prompt dataset. These are fictional examples inspired by real usage—see more examples in Appendix A.2.1.

<table>
<thead>
<tr>
<th>Use-case</th>
<th>Prompt</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brainstorming</td>
<td>List five ideas for how to regain enthusiasm for my career</td>
</tr>
<tr>
<td>Generation</td>
<td>Write a short story where a bear goes to the beach, makes friends with a seal, and then returns home.</td>
</tr>
<tr>
<td>Rewrite</td>
<td>This is the summary of a Broadway play: *** (summary) ***</td>
</tr>
<tr>
<td></td>
<td>This is the outline of the commercial for that play: ***</td>
</tr>
</tbody>
</table>

For OpenAI, RLHF data is collected from their API. Very different from instruct-tuning datasets

- text-davinci-001/002 were both learned only from fine-tuning on demonstrations rated 7/7 (i.e., not using RLHF)

- text-davinci-003 (latest version) and ChatGPT both use PPO with learned reward models

Conclusion: likely difficult to get PPO working reliably (or to get a good reward function — signal from annotators may be unstable)

- ...but RLHF datasets from OpenAI are not public

- Data quality is paramount! Anecdotally there are lots of human-written demonstrations in there and lots of ratings

https://beta.openai.com/docs/model-index-for-researchers

Pre-trained Chatbots
What are chatbots?

- Like story generation in that it’s open-ended, but involves dialogue with a user.
- Input: a conversation history of utterances, plus something the user (a person) just said. Output: the model’s response to that.
- Needs to generate interesting and diverse content, but also needs to be able to answer questions and carry on a conversation.

Seq2seq models

- Can we just train seq2seq models to “translate” from utterance to response?
- Hard to evaluate with automatic metrics:

<table>
<thead>
<tr>
<th>System</th>
<th>BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>RANDOM</td>
<td>0.33</td>
</tr>
<tr>
<td>MT</td>
<td>3.21</td>
</tr>
<tr>
<td>HUMAN</td>
<td>6.08</td>
</tr>
</tbody>
</table>

Lack of Diversity

- Training to maximize likelihood gives a system that prefers common responses:

<table>
<thead>
<tr>
<th>Input: What are you doing?</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0.86 I don’t know.</td>
<td>-1.09</td>
</tr>
<tr>
<td>-1.03 I don’t know!</td>
<td>-1.09</td>
</tr>
<tr>
<td>-1.06 Nothing.</td>
<td>-1.09</td>
</tr>
<tr>
<td>-1.09 Get out of the way.</td>
<td>-1.10</td>
</tr>
<tr>
<td>I am talking to you.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Input: what is your name?</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.91 I don’t know.</td>
<td></td>
</tr>
<tr>
<td>-0.92 I don’t know!</td>
<td>-1.55</td>
</tr>
<tr>
<td>-0.92 I don’t know, sir.</td>
<td>-1.58</td>
</tr>
<tr>
<td>-0.97 Oh, my god!</td>
<td>-1.59</td>
</tr>
<tr>
<td>My name is Robert.</td>
<td></td>
</tr>
<tr>
<td>My name is John.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Input: How old are you?</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.79 I don’t know.</td>
<td>-1.64</td>
</tr>
<tr>
<td>-1.06 I’m fine.</td>
<td>-1.66</td>
</tr>
<tr>
<td>-1.17 I’m all right.</td>
<td>-1.17</td>
</tr>
<tr>
<td>I am an artist</td>
<td></td>
</tr>
<tr>
<td>I have four children</td>
<td></td>
</tr>
<tr>
<td>I recently got a cat</td>
<td></td>
</tr>
<tr>
<td>I enjoy walking for exercise</td>
<td></td>
</tr>
<tr>
<td>I love watching Game of Thrones</td>
<td></td>
</tr>
</tbody>
</table>

PersonaChat

- Efforts to imbue seq2seq models with “personality”
- These systems still don’t work great. What else is needed?
Pre-trained Chatbots

- Initialize the parameters of this model with a pre-trained model, then fine-tune it on some data
- It turns out that scaling up dramatically and strategies like nucleus sampling can help with the “I don’t know” problem

Blender

- 2.7B-param model, also a 9.4B-parameter seq2seq model variant
- “Poly-encoder” Transformer architecture, some training tricks
- Three models: retrieve (from training data), generate, retrieve-and-refine
- Fine-tuning on three prior datasets: PersonaChat, Empathetic Dialogues (discuss personal situation, listener is empathetic), Wizard of Wikipedia (discuss something from Wikipedia)

Roller et al. (2020)
Blender

- Inconsistent responses: this model doesn’t really have anything to say about itself
- Holding a conversation != AI
- Can’t acquire new information
- Did it learn “fun guy”? No, it doesn’t understand phonology. It probably had this in the data somewhere

Chatbots

- What happens when these models get really good at fooling people? Google LaMDA model (similar to Blender):

ChatGPT

- Big model with RLHF. (More like a QA system than these other chatbots)
- Not much we can say except:
  - It’s based on the earlier davinci models
  - Lots of data collection to fencepost it (e.g., “I don’t know anything about the current weather …”)
  - Continuously improved without detailed release notes (e.g., they made it better at math)
Task-Oriented Dialogue

‣ How do you build conversational systems to do things?

Siri, find me a good sushi restaurant in Chelsea

Sushi Seki Chelsea is a sushi restaurant in Chelsea with 4.4 stars on Google

How expensive is it?

Entrees are around $30 each

Find me something cheaper

Task-Oriented Dialogue

‣ Parsing / language understanding is just one piece of a system

• Dialogue state: reflects any information about the conversation (e.g., search history)

• User utterance -> update dialogue state -> take action (e.g., query the restaurant database) -> say something

• How do we represent the information from the user’s utterance?

Young et al. (2013)
ATIS

- Intent and slots model: classify an intent (Airfare), then fill several slots needed to specify the parameters for that intent

<table>
<thead>
<tr>
<th>Utterance</th>
<th>Goal:</th>
</tr>
</thead>
<tbody>
<tr>
<td>How much is the cheapest flight from</td>
<td>Airfare</td>
</tr>
<tr>
<td>Boston to New York tomorrow morning?</td>
<td></td>
</tr>
</tbody>
</table>

- COST. Relative: cheapest
- Depart.City: Boston
- Arrival.City: New York
- Depart.Date/Relative: tomorrow
- Depart.Time/Period: morning

- This is how most Alexa skills work. Can match with rule-based systems or use classifiers

DARPA (early 1990s), Figure from Tur et al. (2010)

Intent

- 29 different intents in ATIS:

  - *which flights go from cleveland to indianapolis on april fifth*
    - **Intent:** flight
  - *does tacoma airport offer transportation from the airport to the downtown area*
    - **Intent:** ground_service
  - *what days of the week do flights from san jose to nashville fly on*
    - **Intent:** day_name
  - *what meals are served on american flight 811 from tampa to milwaukee*
    - **Intent:** meal

Intents

Dataflow Graphs

- How do we scale to more complex dialog scenarios? One proposal: dataflow graphs

User: *Where is my meeting at 2 this afternoon?*

place(findEvent(EventSpec(start=pm(2))))

2 → pm → EventSpec → findEvent → place

Agent: *It’s in Conference Room D.*

(1)

Dataflow Graphs

User: *Can you create a meeting with Megan right before that starts?*

createEvent(EventSpec(
  end=start(refer(Constraint{Event}()))),
  attendee-PersonSpec(name="Megan")
))

Agent: *Which person named Megan did you mean?*

(2)

Semantic Machines; Andreas et al. (2020)
Task-Oriented Dialog: What the user sees

Find me a good sushi restaurant in Chelsea

Sushi Seki Chelsea is a sushi restaurant in Chelsea with 4.4 stars on Google

How expensive is it?

Entrees are around $30 each

Task-Oriented Dialog: Under the hood

Find me a good sushi restaurant in Chelsea

restaurant_type <- sushi
location <- Chelsea
curr_result <- execute_search()

Sushi Seki Chelsea is a sushi restaurant in Chelsea with 4.4 stars on Google

How expensive is it?

get_value(cost, curr_result)
Entrees are around $30 each

Training Dialog Systems

- “Wizard of Oz”: can run the dialog system in a real setting and have a human decide what it should do next
- Learning from demonstrations: the system can learn from what the wizard does and do that in the future

Find me a good sushi restaurant in Chelsea

wizard enters these
restaurant_type <- sushi
location <- Chelsea
curr_result <- execute_search()

Sushi Seki Chelsea is a sushi restaurant in Chelsea with 4.4 stars on Google

wizard types this out or invokes templates

Task-Oriented Dialogue

- Building these systems takes a ton of engineering, like Gunrock — it typically doesn’t use pre-trained models (until 2023...)
- Need to know what the system should do, not just what it should say
- Generation is usually templated (handwritten), otherwise the system can behave unexpectedly
- Lots of industry activity in this space, less in academia (hard to maintain all of the moving parts for a real dialog system)
- Current interest: work like Toolformer / Langchain that allows LLMs to generate the API calls directly
Takeaways

- Instruction-tuning and RLHF are two procedures that take LMs to the next level — these models work dramatically better than basic GPT-3

- These are the foundation of modern chatbots (along with lots of pre-training data), very exciting capabilities in these LLM agents

- Task-oriented dialog has historically been different but is starting to unify with chatbots (Bing agent has ability to make API calls)