CS371N: Natural Language Processing

Lecture 20: Instruction Tuning, RLHF, Dialog

Greg Durrett
Announcements

‣ Tatsu Hashimoto (creator of Alpaca) talk on Friday 11am in GDC 6.302

‣ Midterm back later today

‣ A3 back soon, A4 grading underway

‣ A5 due in 1 week

‣ Greg’s office hours on Zoom today
Recap: Chain-of-thought

**Input**

**Context:** Christopher agrees with Kevin. [...] **Q:** Who hangs out with a student?

*Mary, because Mary hangs out with Danielle and Danielle is a student.*

**Train Ex**

**Context:** Adam plays with Ellen. [...] **Q:** Who plays with a doctor?

**Output**

*Adam, because Adam plays with Ellen and Ellen is a doctor.*

greedy decoding from GPT-3
Recap: Chain-of-thought

- Can help substantially on mathematical reasoning

- Some work to optimize the specifics of the prompts and the examples

**Input:**

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: John takes care of 10 dogs. Each dog takes .5 hours a day to walk and take care of their business. How many hours a week does he spend taking care of dogs?

A:

**Model output:**

John takes care of 10 dogs. Each dog takes .5 hours a day to walk and take care of their business. So that is 10 x .5 = 5 hours a day. 5 hours a day x 7 days a week = 35 hours a week. The answer is 35 hours a week.  

Wei et al. (2022)
Today

- Alignment:
  - Instruction tuning
  - RLHF
- Chatbots
- Task-oriented dialogue systems
Alignment
What was the "shot heard round the world"?

The shot that was heard round the world was fired in Lexington, MA. It was fired because, at the time, America was still under British rule. The British soldiers had been taking firewood by throwing it down from the top of their fort at Lexington. Someone who was not a soldier, by the name of Joseph Warren (who had ties to the Sons of Liberty or the Sons of Freedom) shouted to the British to stop doing it. The British soldiers ignored him, not...

What was the "shot heard round the world"?

On the morning of April 19th, 1775, AMC's "Turn" did the impossible, and turned a war not many were aware was coming into one of the catalyst events that inspired the births of our modern America. Indeed, all of AMC's Revolutionary War dramas have done a masterful job of bringing this bloody time in our nation's birth "back" into view in exciting new ways for the 21st century.

What was the "shot heard round the world"?

"What to the slave is the 4th of July?"

What was the sideline occupation of Thomas Jefferson's wife, Martha, and their enslaved girl, Sally Hemmings?

What was the name of the first African American regiment from Vermont?

Samples from GPT-3
(a “basic” LM)
What was the "shot heard round the world"?

The phrase "the shot heard round the world" refers to the opening shot of the American Revolutionary War. It took place on April 19, 1775, in the town of Lexington, Massachusetts. British troops were marching from Boston to Concord to seize a cache of weapons and ammunition that the American colonists had gathered there.

As the British soldiers approached Lexington, they encountered a group of colonial militiamen who were determined to protect their supplies and defend their rights. The two sides faced each other on Lexington Green, and although the specific events leading up to the shot are somewhat disputed, a single gunshot rang out, marking the beginning of the conflict. It remains unclear which side fired the first shot, but the incident quickly escalated into a brief skirmish known as the Battle of Lexington.

Ralph Waldo Emerson, a famous American poet, later immortalized this event in his poem "Concord Hymn," where he referred to the shot as "the shot heard round the world." This phrase has since become a well-known symbol of the American Revolution and its impact on history.
Alignment

- We want to optimize models for $P(\text{answer} \mid \text{prompt, input})$, but they’re learned on a basic language modeling objective

- **Alignment**: general class of methods for making LLMs produce useful output
  - Sometimes broken down into “helpfulness” (responsiveness to prompts, informativeness, correctness) and “harmlessness” (not being biased or toxic, not responsive to harmful prompts)

- Two main 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
Instruction Tuning
Encoder-Decoder Models: T5

- Pre-training: not quite vanilla language modeling, but a “denoising” scheme to BERT


Original text

Thank you for inviting me to your party last week.

Inputs

Thank you <X> me to your party <Y> week.

Targets

<X> for inviting <Y> last <Z>

Raffel et al. (2019)
T5

<table>
<thead>
<tr>
<th>Number of tokens</th>
<th>Repeats</th>
<th>GLUE</th>
<th>CNNDM</th>
<th>EnDe</th>
<th>EnFr</th>
<th>EnRo</th>
</tr>
</thead>
<tbody>
<tr>
<td>★ Full dataset</td>
<td>0</td>
<td>83.28</td>
<td>19.24</td>
<td>26.98</td>
<td>39.82</td>
<td>27.65</td>
</tr>
<tr>
<td>$2^{29}$</td>
<td>64</td>
<td>82.87</td>
<td>19.19</td>
<td>26.83</td>
<td>39.74</td>
<td>27.63</td>
</tr>
<tr>
<td>$2^{27}$</td>
<td>256</td>
<td>82.62</td>
<td>19.20</td>
<td>27.02</td>
<td>39.71</td>
<td>27.33</td>
</tr>
<tr>
<td>$2^{25}$</td>
<td>1,024</td>
<td>79.55</td>
<td>18.57</td>
<td>26.38</td>
<td>39.56</td>
<td>26.80</td>
</tr>
<tr>
<td>$2^{23}$</td>
<td>4,096</td>
<td>76.34</td>
<td>18.33</td>
<td>26.37</td>
<td>38.84</td>
<td>25.81</td>
</tr>
</tbody>
</table>

summarization  machine translation

- Colossal Cleaned Common Crawl: 750 GB of text
- T5 was designed to be trained on many tasks and map from inputs to outputs

Raffel et al. (2019)
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

Sanh et al. (2021)
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*

### Train
- **Multiple-Choice QA**
  - CommonsenseQA
  - DREAM
  - QuAIL
  - QuaRTz
  - Social IQA
  - WiQA
  - Cosmos QA
  - QASC
  - QuaRel
  - SoloQ
  - WikiHop
- **Closed-Book QA**
  - Hotpot QA
  - Wiki QA
- **Structure-To-Text**
  - Common Gen
  - WikiBio
- **Sentiment**
  - Amazon
  - App Reviews
  - IMDB
  - Rotten Tomatoes
  - Yelp
- **Summarization**
  - CNN Daily Mail
  - Gigaword
  - MultiNews
  - SamSum
  - XSum
- **Topic Classification**
  - AG News
  - DBPedia
  - TREC
- **Paraphrase Identification**
  - MRPC
  - PAWS
  - QQP
- **Word Sense Disambiguation**
  - WIC

### Test
- **Sentence Completion**
  - COPA
  - HellaSwag
  - Story Cloze
- **Natural Language Inference**
  - ANLI
  - CB
  - RTE
- **Coreference Resolution**
  - WSC
  - Winogrande
- **BIG-Bench**
  - Code Description
  - Conceptual
  - Hindu Knowledge
  - Known Unknowns
  - Language ID
  - Logic Grid
  - Logical Deduction
  - Misconceptions
  - Movie Dialog
  - Novel Concepts
  - Strategy QA
  - Sylogisms
  - Vitamin C
  - Winowhy

---

Sanh et al. (2021)
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)

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.
Frontiers

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

  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.

  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.

  In the complex $z$-plane, the set of points satisfying the equation $z^2 = |z|^2$ is a
  
  (A) pair of points
  (B) circle
  (C) half-line
  (D) line

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

Chung et al. (2022)
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>Date</th>
<th>Model Type</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>May 2020</td>
<td>GPT-3 5-shot</td>
<td>43.9</td>
</tr>
<tr>
<td>Mar. 2022</td>
<td>Chinchilla 5-shot</td>
<td>67.6</td>
</tr>
<tr>
<td>Apr. 2022</td>
<td>PaLM 5-shot</td>
<td>69.3</td>
</tr>
<tr>
<td>Oct. 2022</td>
<td>Flan-PaLM 5-shot</td>
<td>72.2</td>
</tr>
<tr>
<td></td>
<td>Flan-PaLM 5-shot: CoT + SC</td>
<td>75.2</td>
</tr>
<tr>
<td></td>
<td>Average human expert</td>
<td>89.8</td>
</tr>
</tbody>
</table>

Chung et al. (2022)
### Human performance estimates are ~80 on Big-Bench (BBH)

<table>
<thead>
<tr>
<th>Model</th>
<th>Finetuning Mixtures</th>
<th>Tasks</th>
<th>Norm. avg.</th>
<th>MMLU Direct</th>
<th>MMLU CoT</th>
<th>BBH Direct</th>
<th>BBH CoT</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>
<td>49.1</td>
<td>63.7</td>
</tr>
<tr>
<td>CoT</td>
<td></td>
<td>9</td>
<td>52.6 (+3.5)</td>
<td>68.8</td>
<td>64.8</td>
<td>50.5</td>
<td>61.1</td>
</tr>
<tr>
<td>CoT, Muffin</td>
<td></td>
<td>89</td>
<td>57.0 (+7.9)</td>
<td>71.8</td>
<td>66.7</td>
<td>56.7</td>
<td>64.0</td>
</tr>
<tr>
<td>CoT, Muffin, T0-SF</td>
<td></td>
<td>282</td>
<td>57.5 (+8.4)</td>
<td>72.9</td>
<td><strong>68.2</strong></td>
<td>57.3</td>
<td>64.0</td>
</tr>
<tr>
<td>CoT, Muffin, T0-SF, NIV2</td>
<td></td>
<td>1,836</td>
<td><strong>58.5 (+9.4)</strong></td>
<td><strong>73.2</strong></td>
<td>68.1</td>
<td><strong>58.8</strong></td>
<td><strong>65.6</strong></td>
</tr>
</tbody>
</table>
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.

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.

- Task
  - Instruction: Give me a quote from a famous person on this topic.
  - Input: Topic: The importance of being honest.
  - Output: "Honesty is the first chapter in the book of wisdom." - Thomas Jefferson

Step 4: Filtering

- Fine-tune Llama on 52k outputs with answers generated by text-davinci-003

Yizhong Wang et al. (2023) Self-Instruct
Ronen Taori et al. (2023) Alpaca
<table>
<thead>
<tr>
<th>Source</th>
<th>#Examples</th>
<th>Avg Input Len.</th>
<th>Avg Output Len.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Training</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Stack Exchange (STEM)</td>
<td>200</td>
<td>117</td>
<td>523</td>
</tr>
<tr>
<td>Stack Exchange (Other)</td>
<td>200</td>
<td>119</td>
<td>530</td>
</tr>
<tr>
<td>wikiHow</td>
<td>200</td>
<td>12</td>
<td>1,811</td>
</tr>
<tr>
<td>Pushshift r/WritingPrompts</td>
<td>150</td>
<td>34</td>
<td>274</td>
</tr>
<tr>
<td>Natural Instructions</td>
<td>50</td>
<td>236</td>
<td>92</td>
</tr>
<tr>
<td>Paper Authors (Group A)</td>
<td>200</td>
<td>40</td>
<td>334</td>
</tr>
<tr>
<td><strong>Dev</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Paper Authors (Group A)</td>
<td>50</td>
<td>36</td>
<td>N/A</td>
</tr>
<tr>
<td><strong>Test</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pushshift r/AskReddit</td>
<td>70</td>
<td>30</td>
<td>N/A</td>
</tr>
<tr>
<td>Paper Authors (Group B)</td>
<td>230</td>
<td>31</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Table 1: Sources of training prompts (inputs) and responses (outputs), and test prompts. The total amount of training data is roughly 750,000 tokens, split over exactly 1,000 sequences.

- How little data can we get away with for fine-tuning?
Figure 1: Human preference evaluation, comparing LIMA to 5 different baselines across 300 test prompts.
Reinforcement Learning from Human Feedback (RLHF)
RLHF

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 with supervised learning.

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.

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

Ouyang et al. (2022)
**RLHF**

- Humans produce comparisons of two trajectories (= outputs from systems) — different from standard reward in RL

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

$$
\hat{P}[\sigma^1 \succ \sigma^2] = \frac{\exp \sum \hat{r}(o_t^1, a_t^1)}{\exp \sum \hat{r}(o_t^1, a_t^1) + \exp \sum \hat{r}(o_t^2, a_t^2)}.
$$

- 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)

  Christiano et al. (2017)
RLHF

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: &quot;&quot;&quot;&quot;</td>
</tr>
<tr>
<td></td>
<td>{summary}</td>
</tr>
<tr>
<td></td>
<td>&quot;&quot;&quot;&quot;</td>
</tr>
<tr>
<td></td>
<td>This is the outline of the commercial for that play: &quot;&quot;&quot;&quot;</td>
</tr>
</tbody>
</table>

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

Ouyang et al. (2022)
text-davinci-003

- 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

- Hard to get PPO working reliably (or to get a good reward function — signal from annotators may be unstable)

- 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.
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)
Will you sing me a song?

Sure, what do you want to be sung to? I can sing you a song about baking.

Yes, sing me a song about baking!

Do you want me to sing it to the tune of "Sing Along with Mitch"?

Yes, I would love that.

Ok, I'll try to get it to you by the end of the day. What else do you like to do?

Can you at least tell me what the lyrics are?

Sure, it goes like this: "Baking is my favorite thing to do in the world."

How does the chorus go?

It goes: BAKING IS MY FAVORITE THING TO DO IN THE WORLD!
I'm from the midwest, but I live in the south now. How about you?

Born and raised in the south

What part of the south? I was born in georgia, but raised in tennessee
Why did the mushroom go to the party?

I'm not sure, but I'm glad it did. I love museums. What do you do for a living?

No, it was supposed to be a joke.

Oh, sorry. I didn't get it. What kind of joke was it? I like jokes.

The answer is, "because he's a fun guy." Do you get it?

No, I don't. Can you explain it to me like I'm a five year old?

Yes. Mushrooms are fungi, and "fungi" is pronounced like "fun guy".

Oh, okay. I get it now. Thank you for the explanation. I really appreciate it.

Are you sure you understand? Can you explain it to me?

Sure. It's a play on the word fungus, which is also pronounced fun guy.
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):

"I KNOW A PERSON WHEN I TALK TO IT" —

Google fires Blake Lemoine, the engineer who claimed AI chatbot is a person

Google says Lemoine violated security rules, slams "wholly untruthful" service

Ex-Google engineer Blake Lemoine discusses sentient AI

Ex-Google engineer Blake Lemoine discusses why LaMDA and other AI systems may be considered sentient and explains exactly how much AI systems know about consumers.

Blake Lemoine: Google fires engineer who said AI tech has feelings
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
Hey Alexa, why isn’t my Amazon order here?

Let me retrieve your order. Your order was scheduled to arrive at 4pm today.

It never came

Okay, I can put you through to customer service.
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)
Intent and slots model: classify an intent (Airfare), then fill several slots needed to specify the parameters for that intent

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)
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
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))))

Agent: It’s in Conference Room D.
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?
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 a good sushi restaurant in Chelsea

\[
\text{restaurant\_type} \leftarrow \text{sushi} \\
\text{location} \leftarrow \text{Chelsea} \\
\text{curr\_result} \leftarrow \text{execute\_search()} \\
\text{Sushi Seki Chelsea is a sushi restaurant in Chelsea with 4.4 stars on Google}
\]

How expensive is it?

\[
\text{get\_value(cost, curr\_result)} \\
\text{Entrees are around $30 each}
\]
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()  
}

wizard types this out or invokes templates

Sushi Seki Chelsea is a sushi restaurant in Chelsea with 4.4 stars on Google
Task-Oriented Dialogue

- Building these systems takes a ton of engineering — 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)