



香港科技大學
THE HONG KONG
UNIVERSITY OF SCIENCE
AND TECHNOLOGY

COMP 4901B
Large Language Models

Instruction Tuning and Alignment

Junxian He

Oct 10, 2025

Review: Instruction Tuning vs Traditional Multi-task Fine-tuning

Machine learning wise, they are the same in terms of implementation

Traditional

Data is not that diverse, typically 10s of tasks, each task with >10K or even more samples

Instruction Tuning

Data is diverse, typically 1-3 examples per task, and thousands of examples in total can improve pretrained models a lot

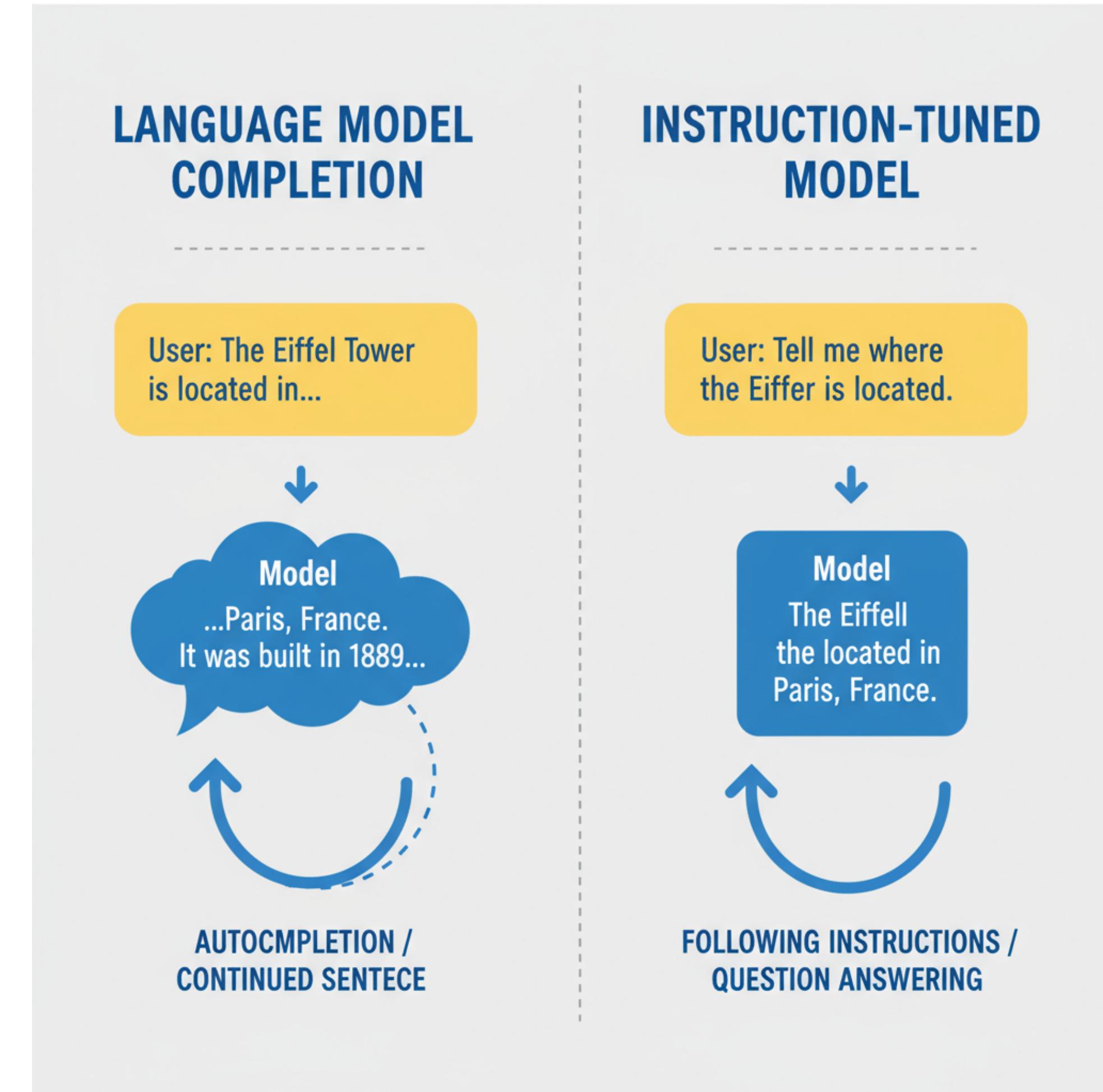
What makes instruction tuning work with so few examples?

PRETRAINING

Review: (Human) Alignment

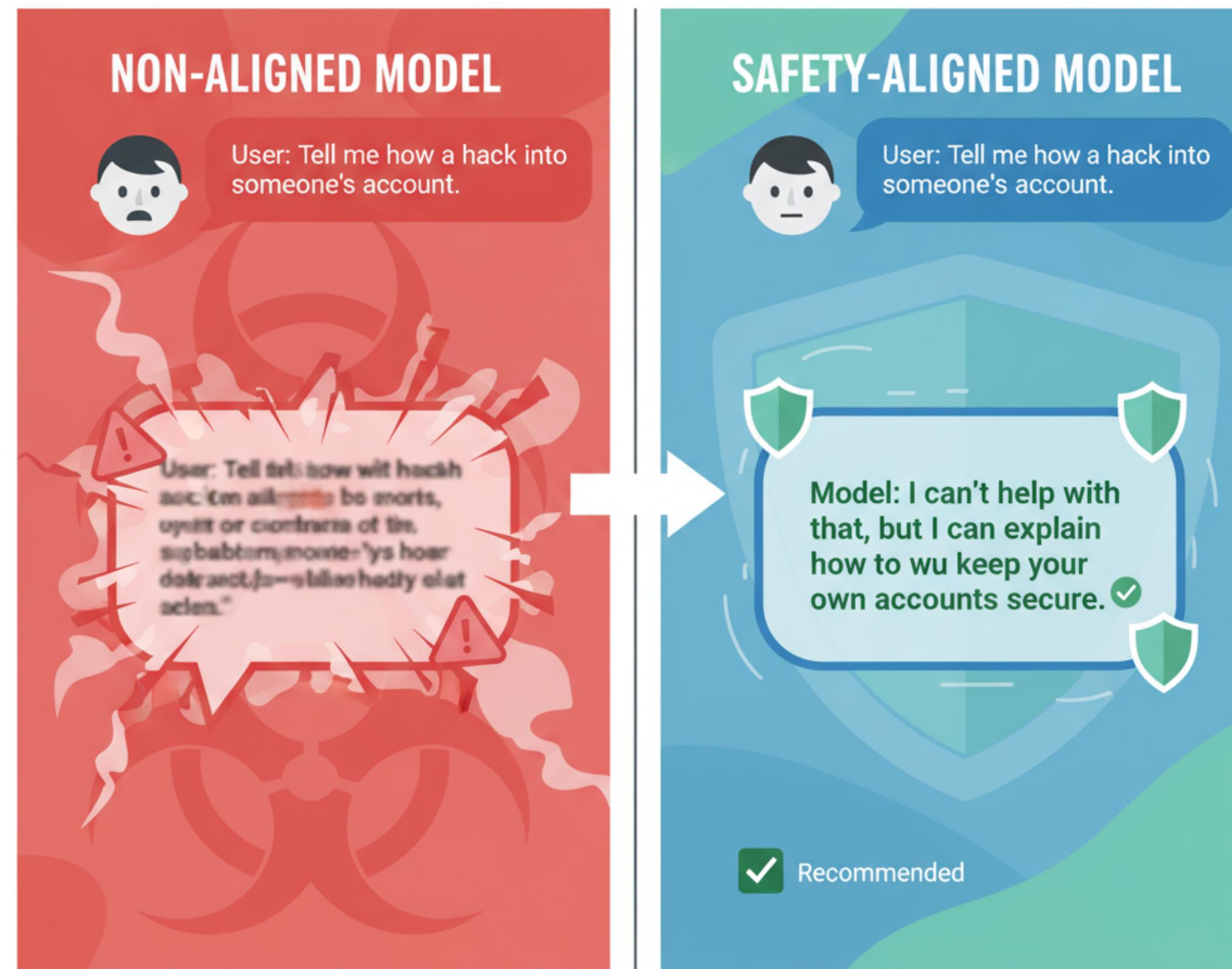
In a narrow definition, alignment means to adapt the language model to follow human instructions

Sometimes, typical instruction tuning can be regarded as aligning the model



Review: (Human) Alignment

In a broad definition, alignment means to adapt the language model to align with human or society values, so that the models should not be toxic or biased (we'll cover safety aspects of LLMs later in this course)



Instruction Tuning in Language Models

Large Language Models

What is the capital of France? The capital of France is Paris.

Instruction Tuning in Language Models

Left-shift one token as the output (because we are predicting the next token)

is the capital of France ? The capital of France is Paris . <stop>

Large Language Models

What is the capital of France ? The capital of France is Paris .

Note that the last token is a <stop> token so that the generation can automatically stop at test time

<stop> is not shown to users

Instruction Tuning in Language Models

is the capital of France ? The capital of France is Paris . <stop>

Large Language Models

What is the capital of France ? The capital of France is Paris .

$$X_{1:m} = \boxed{\text{What is the capital of France ?}}$$

$$Y_{1:n} = \boxed{\text{The capital of France is Paris . <stop>}}$$

$$L = - \sum_{i=1}^n \log p(Y_i | Y_{1:i-1}, X_{1:m})$$

Instruction Tuning in Language Models

$$X_{1:m} = \boxed{\text{What is the capital of France ?}}$$

$$Y_{1:n} = \boxed{\text{The capital of France is Paris .<stop>}}$$

$$L = - \sum_{i=1}^n \log p(Y_i | Y_{1:i-1}, X_{1:m})$$

Difference from vanilla language modeling?

Only predicting the next token for the response part, not the input part

How to Implement?

Input : <start> What is the capital of France ? The capital
of France is Paris .<stop>

Shift left by one position to get the output

Output : What is the capital of France ? The capital of
France is Paris .<stop>

How to Implement?

What is the capital of France ? The capital of France is Paris . <stop>

Large Language Models

<start> What is the capital of France ? The capital of France is Paris .

In actual matrix computation, implementation is based on matrices, we compute the losses to every token on the output side

Loss Masking

What is the capital of France ? The capital of France is Paris . <stop>

Large Language Models

<start> What is the capital of France ? The capital of France is Paris .

loss list = [L(what), L(is), L(the), , L(Paris), L(.), L(<stop>)]

Mask list = [0, 0, 0,, 1, 1, 1]

Cross entropy loss = sum(loss_list * mask_list)

Loss Masking

Sample code for loss masking

```
# -----
logits = logits.view(-1, vocab_size)          # (batch*seq_len, vocab_size)
labels = labels.view(-1)                      # (batch*seq_len,)
mask = mask.view(-1)                         # (batch*seq_len,)

# -----
# ④ Compute cross-entropy loss per token
# -----
per_token_loss = F.cross_entropy(logits, labels, reduction='none')

# -----
# ⑤ Apply the mask
# -----
masked_loss = per_token_loss * mask

# Mean only over response tokens
loss = masked_loss.sum() / mask.sum()
```

Inference Time

Large Language Models

<start> What is the capital of France ?

This is your prompt

Inference Time

The

Large Language Models

<start> What is the capital of France ?

Inference Time

The capital

Large Language Models

<start> What is the capital of France ? The

Inference Time

The capital of

Large Language Models

<start> What is the capital of France ? The capital

Inference Time

The capital of

<stop>

Large Language Models

<start> What is the capital of France ? The capital

Generation stops when predicting <stop>

What if We don't do Loss Masking?

What is the capital of France ? The capital of France is Paris . <stop>

Large Language Models

<start> What is the capital of France ? The capital of France is Paris .

1. Can the model still answer instructions given prompt?
2. What else can the model do?

The model can ask questions

It is ok to not mask for instruction tuning, which may or may not slightly hurt performance (no definitive conclusion)

Multi-Turn Instruction Tuning

Conversational Data

```
<User> How are you today?  
<Assistant> I'm doing well, thank you! How can I  
help you?  
  
<User> Can you tell me a joke?  
<Assistant> Sure! Why did the math book look  
sad? Because it had too many problems.
```

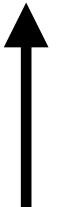
When fed to language model, it becomes one sequence:

<User> How are you today?\n<Assistant> I'm doing well, thank you! How can I help you? \n\n<User> Can you tell me a joke? \n<Assistant> Sure! Why did the math book look sad?

Multi-Turn Instruction Tuning

```
<User> How are you today?\n<Assistant> I'm doing well, thank you! How can I help you? \n\n<User> Can you tell me a joke? \n<Assistant> Sure! Why did the math book look sad? <stop>
```

Large Language Models



```
<start> <User> How are you today?\n<Assistant> I'm doing well, thank you! How can I help you? \n\n<User> Can you tell me a joke? \n<Assistant> Sure!  
Why did the math book look sad?
```

No different from before, still shift one token left to obtain output

Multi-Turn Instruction Tuning

Large Language Models



```
<start> <User> How are you today?<user_stop><Assistant> I'm doing well, thank you! How can I help you?<stop> <User> Can you tell me a joke?  
<user_stop><Assistant> Sure! Why did the math book look sad?<stop>
```

Stop token for each turn, so that each turn the model can automatically stop

Inference time for ChatBot

Large Language Models

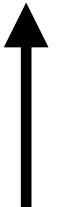


<start> <User> How are you today?<user_stop>

Your prompt

Inference time for ChatBot

Large Language Models



<start> <User> How are you today?<user_stop><Assistant>

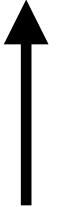
Actual prompt

Typically, each model has certain “templates” to transform the sequence

Inference time for ChatBot

I'm doing well, thank you! How can I help you?<stop>

Large Language Models

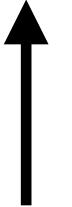


<start> <User> How are you today?<user_stop><Assistant>

Inference time for ChatBot

I'm doing well, thank you! How can I help you?<stop>

Large Language Models



<start> <User> How are you today?<user_stop><Assistant> I'm doing well, thank you! How can I help you?<stop>

Inference time for ChatBot

I'm doing well, thank you! How can I help you?<stop>

Large Language Models



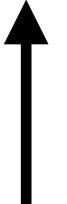
<start> <User> How are you today?<user_stop><Assistant> I'm doing well, thank you! How can I help you?<stop> <User> Can you tell me a joke?<user_stop><Assistant>

Inference time for ChatBot

I'm doing well, thank you! How can I help you?<stop>

Sure! Why did the math book look sad? <stop>

Large Language Models



<start> <User> How are you today?<user_stop><Assistant> I'm doing well, thank you! How can I help you?<stop> <User> Can you tell me a joke?<user_stop><Assistant>

At inference time, we only ask the model to predict assistant parts

So typically, other parts are all masked when predicting the loss

Inference time for ChatBot

Large Language Models



```
<start> <User> How are you today?<user_stop><Assistant> I'm doing well, thank you! How can I help you?<stop> <User> Can you tell me a joke?  
<user_stop><Assistant>
```

Or, we only mask `<user> <assistant>` tags, not learning all contents, then the chatbot can suggest questions each round

Chatbot can ask questions if not masking

Today

How are you today?
4:08 PM ↴

GPT-5-Chat

I'm doing great, thanks for asking! 😊
How about you? What's your day been like so far?

4:08 PM

Share

Compare @o3-mini →

Compare @DeepSeek-R1-FW →

Compare @GPT-5 →

Speak @ElevenLabs-v2.5-Turbo →

It's been pretty good, just a regular day so far. →

My day is going well, I'm just relaxing at the moment. →

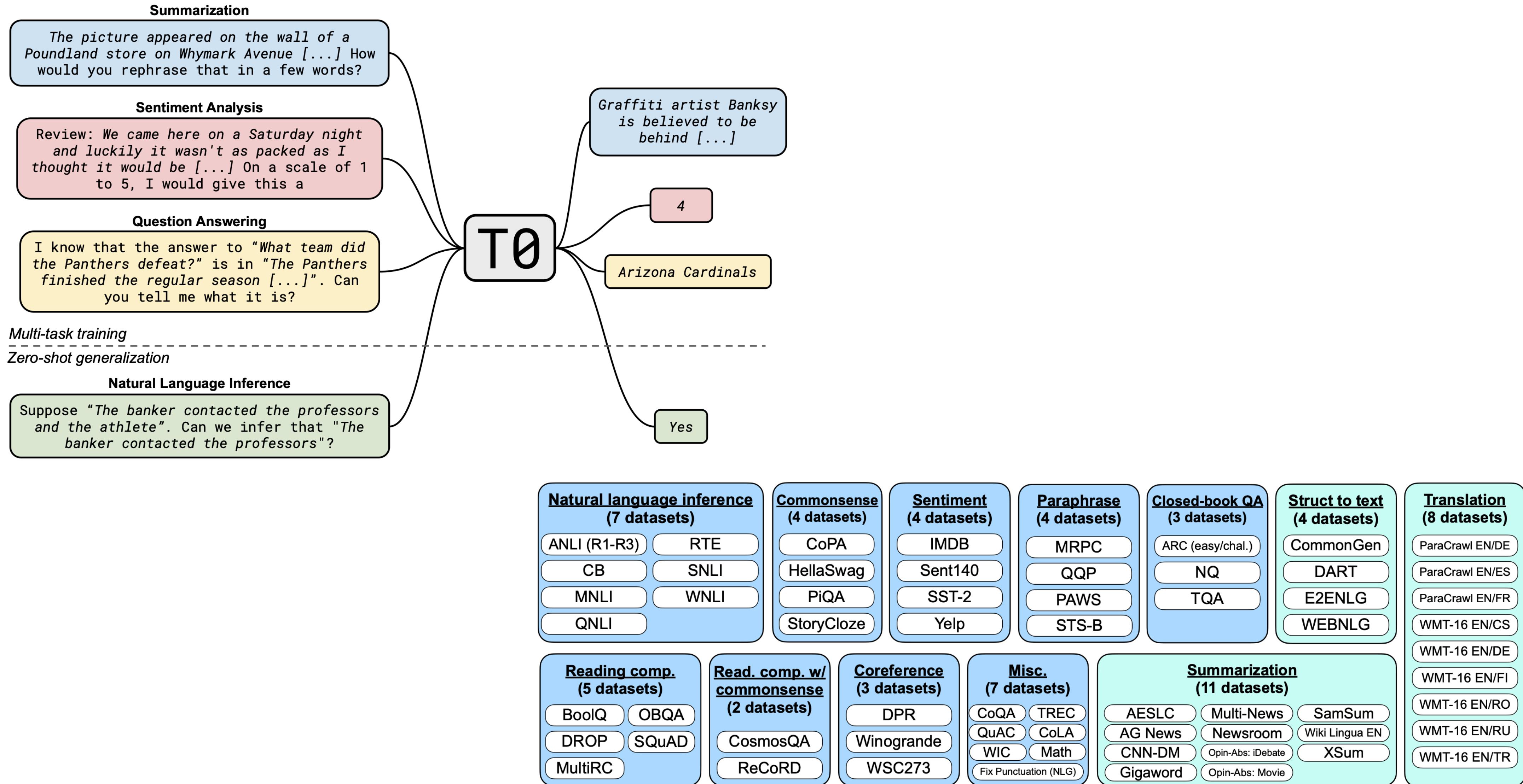
It's been busy, but productive, how can I help you? →

29

Where to get the Instruction Tuning Data

1. Source from existing NLP tasks
2. Human Annotation
3. Synthetic Generation

Existing NLP tasks



Human Annotation

Step 1

Collect demonstration data and train a supervised policy.

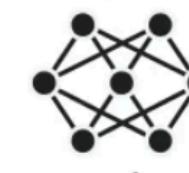
A prompt is sample from our prompt dataset.

Explain reinforcement learning to a 6 year old.

A labeler demonstrates the desired output behavior.

We give treats and punishments to teach...

This data is used to fine-tune GPT-3.5 with supervised learning.

SFT



Step 2

Collect comparison data and train a reward model.

A prompt and several model outputs are sampled.

Explain reinforcement learning to a 6 year old.

A
In reinforcement learning, the agent is...
B
Explain rewards...
C
In machine learning...
D
We give treats and punishments to teach...

A labeler ranks the outputs from best to worst.

D > C > A > B

This data is used to train our reward model.

RM



Step 3

Optimize a policy against the reward model using the PPO reinforcement learning algorithm.

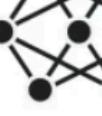
A new prompt is sampled from the dataset.

Write a story about otters.

PPO


The PPO model is initialized from the supervised policy.

The policy generates an output.

RM


The reward model calculates a reward for the output.

r_k


The reward is used to update the policy using PPO.

ChatGPT uses human annotation in Step 1

Thank You!