CS 232: Artificial Intelligence

Fall 2026

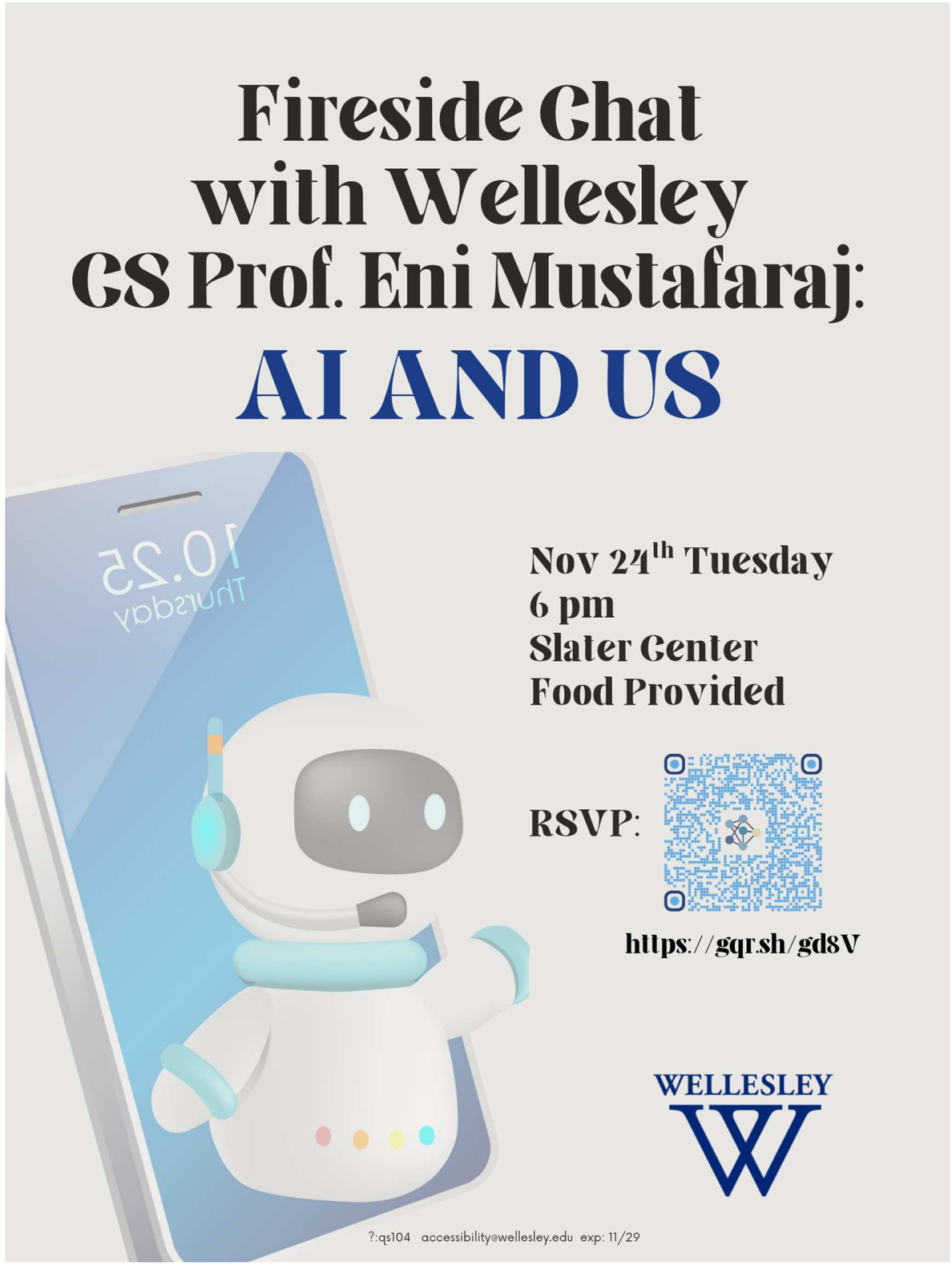
Prof. Carolyn Anderson
Wellesley College

Reminders

- Homework 8 deadline extended to Wednesday
- My help hours:

  - Friday 3-4 in W423

  - Monday 3-4 in W323



Recap: Using LLMs

Representation learning:
extract attention features and use as input features to another model

we did this in HW 6!

Pretraining:
learn good representations via an unlabeled task.

Finetuning:
train some more on in-domain data or separate labeled task

you might do this for your final project!

Prompt engineering:
craft prompts that disguise task of interest as a language generation problem.

we did this in HW 7!

Fine-tuning = Further training

Fine-tuning:
train some more on in-domain data or separate labeled task

Task Fine-tuning

Train on more labeled data for the target task

Instruction Tuning

Train the model to follow different kinds of instructions

Reinforcement Learning

Tune the model with a reward function

Reinforcement Learning

Aligning Models with Human Preferences

Users tend to have preferences about generated text that go beyond its statistical frequency.

Companies also have preferences for how their models respond.

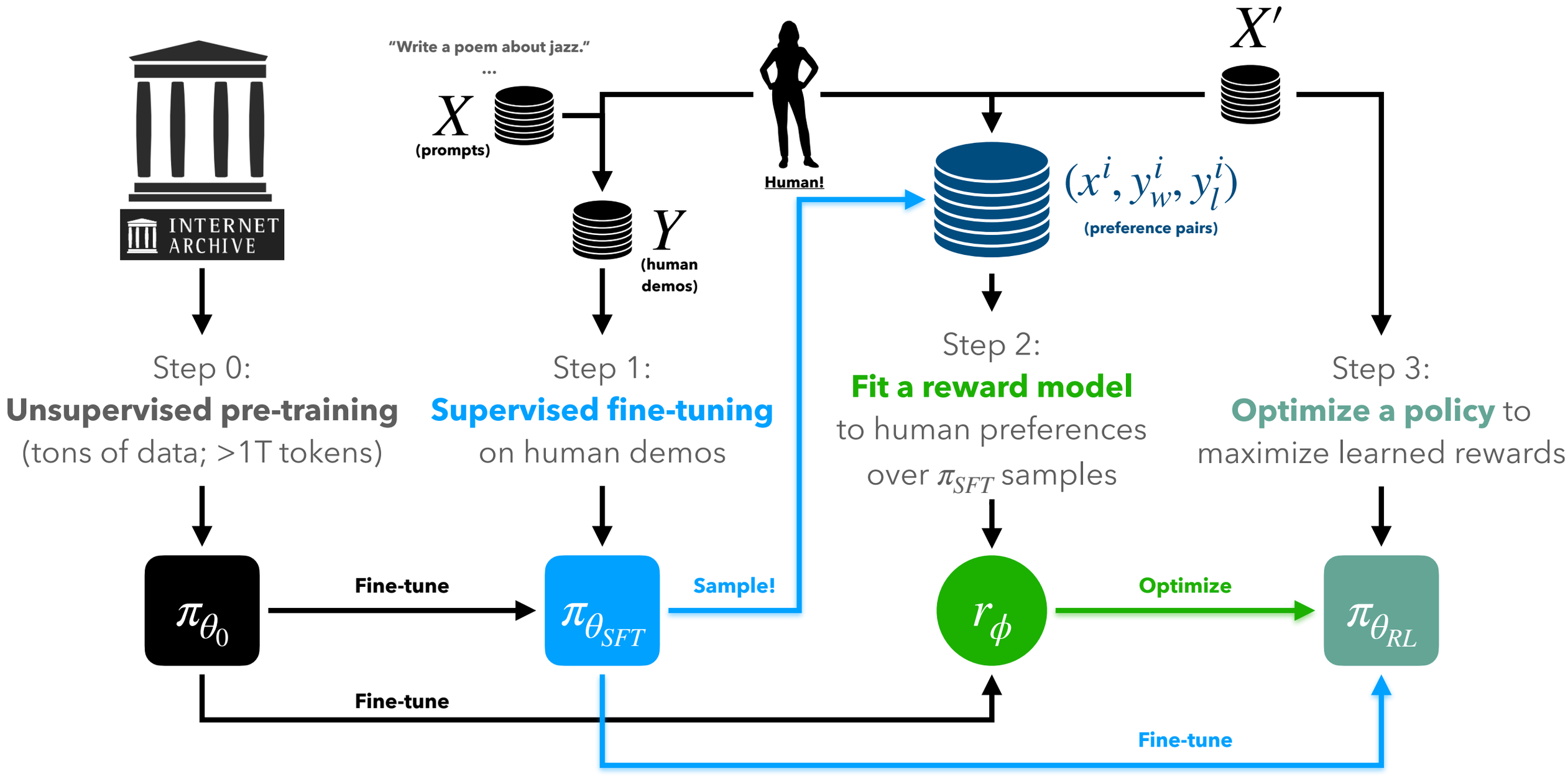


How can we align model behavior with these preferences?



Mark Dredze
@mdredze

Reinforcement Learning from Human Feedback Process



[Figure from Eric Mitchell]

Potential Impact on Annotators



Artificial intelligence (AI)

This article is more than **2 years old**

'It's destroyed me completely': Kenyan moderators decry toll of training of AI models

Employees describe the psychological trauma of reading and viewing graphic content, low pay and abrupt dismissals



Niamh Rowe

Wed 2 Aug 2023 11.00 EDT

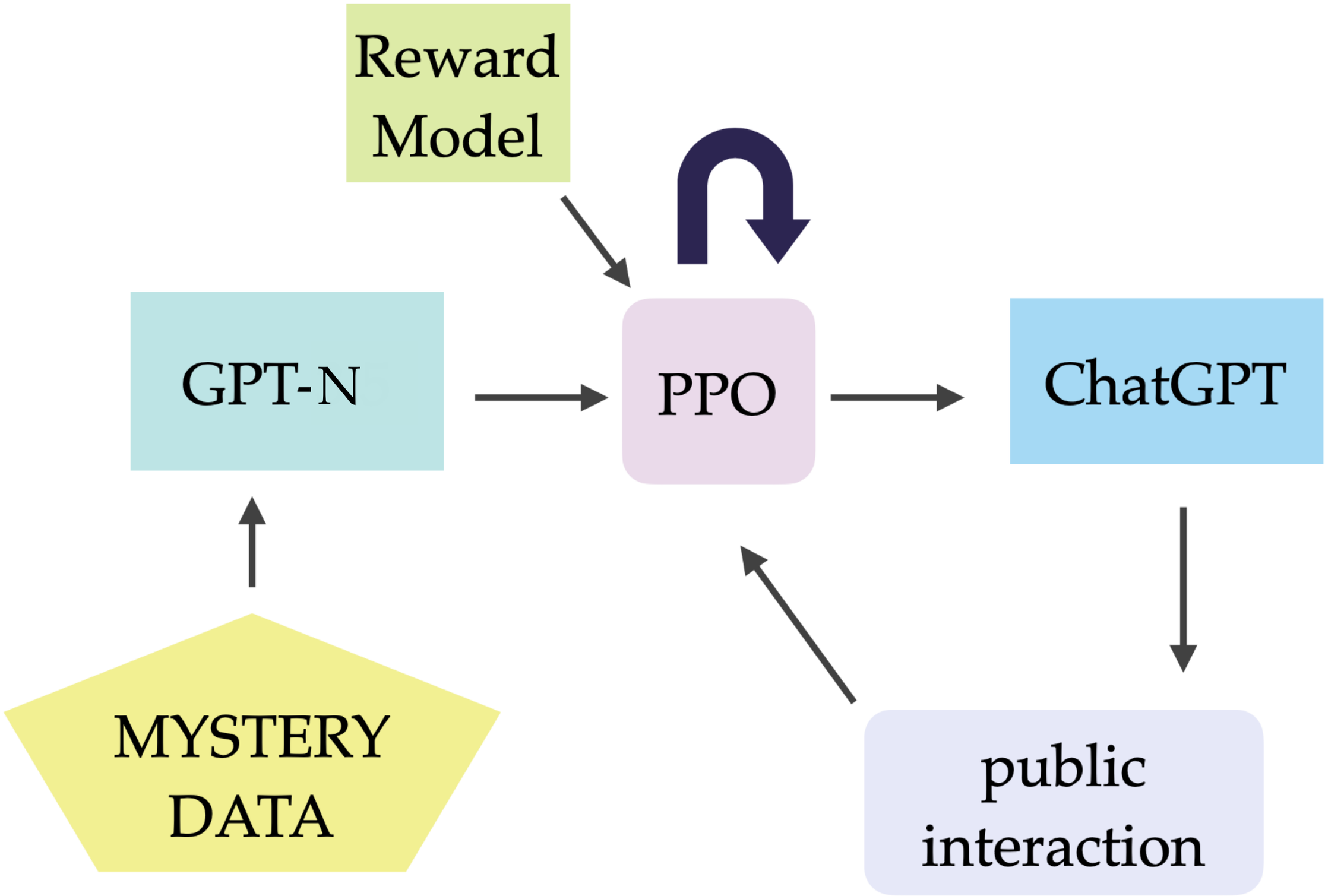
Share

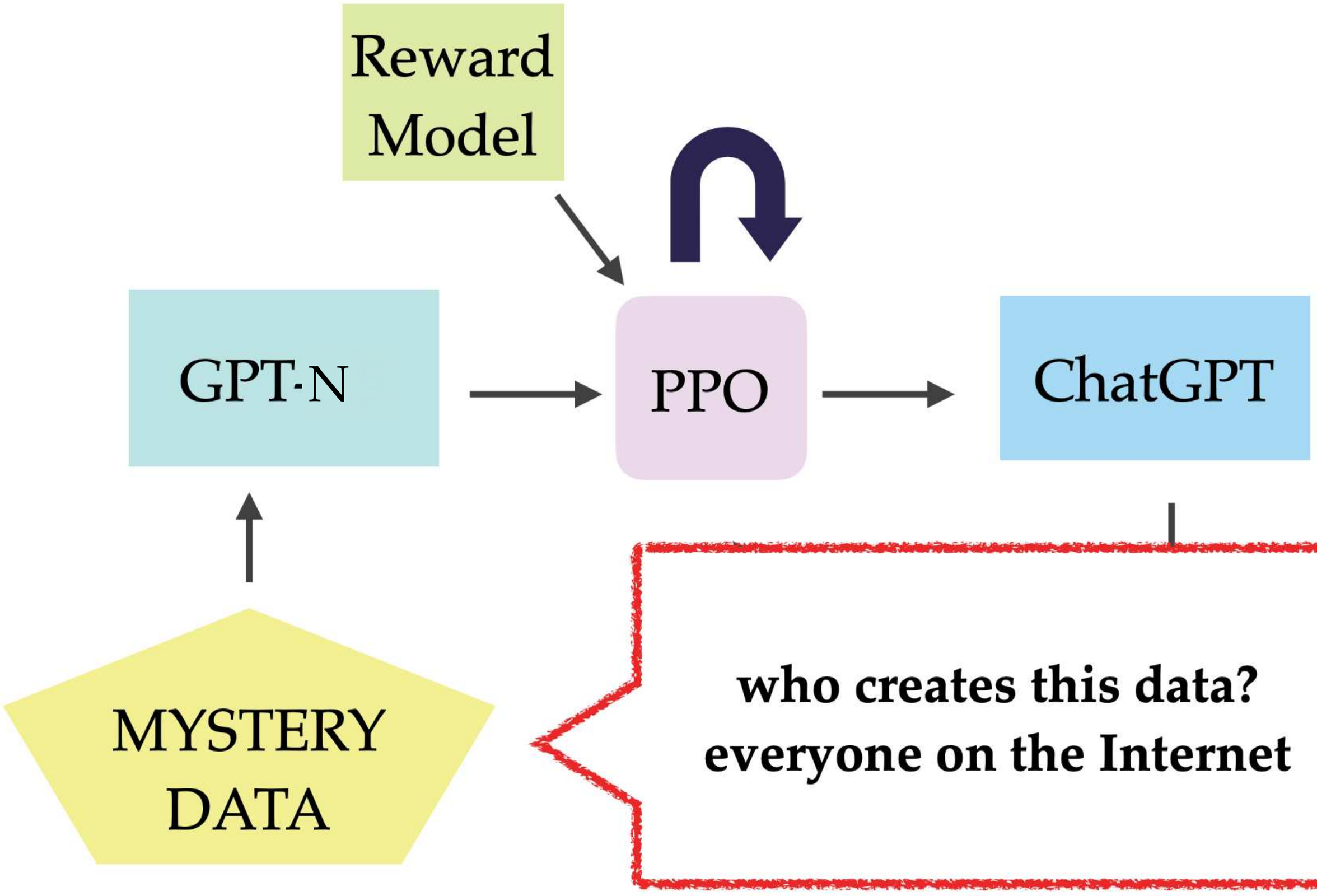
The images pop up in Mophat Okinyi's mind when he's alone, or when he's about to sleep.

Okinyi, a former content moderator for Open AI's ChatGPT in Nairobi, Kenya, is one of four people in that role who have filed a petition to the Kenyan government calling for an investigation into what they describe as exploitative conditions for contractors reviewing the content that powers artificial intelligence programs.

"The 51 moderators in Nairobi working on Sama's OpenAI account were tasked with reviewing texts, and some images, many depicting graphic scenes of violence, self-harm, murder, rape, necrophilia, child abuse, bestiality and incest.

The moderators say they weren't adequately warned about the brutality of some of the text and images they would be tasked with reviewing, and were offered no or inadequate psychological support. Workers were paid between $1.46 and $3.74 an hour, according to a Sama spokesperson."


Reward Model
GPT-N
PPO
ChatGPT
MYSTERY DATA
public interaction


Reward Model
GPT-N
PPO
ChatGPT
MYSTERY DATA
who creates this data?
everyone on the Internet

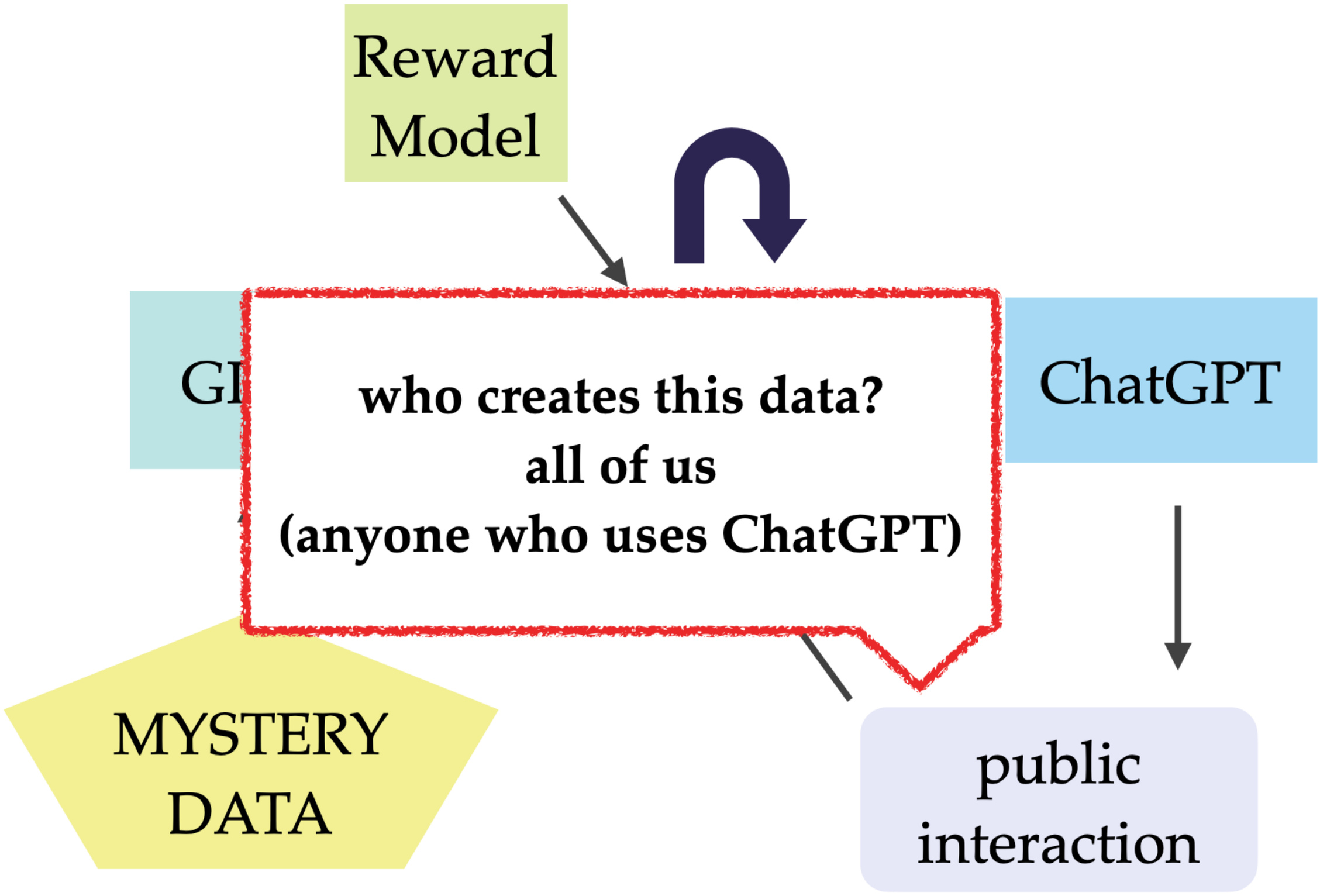
Reward Model

who creates the ratings? underpaid crowd workers in the Global South

GPT-N → PPO → ChatGPT

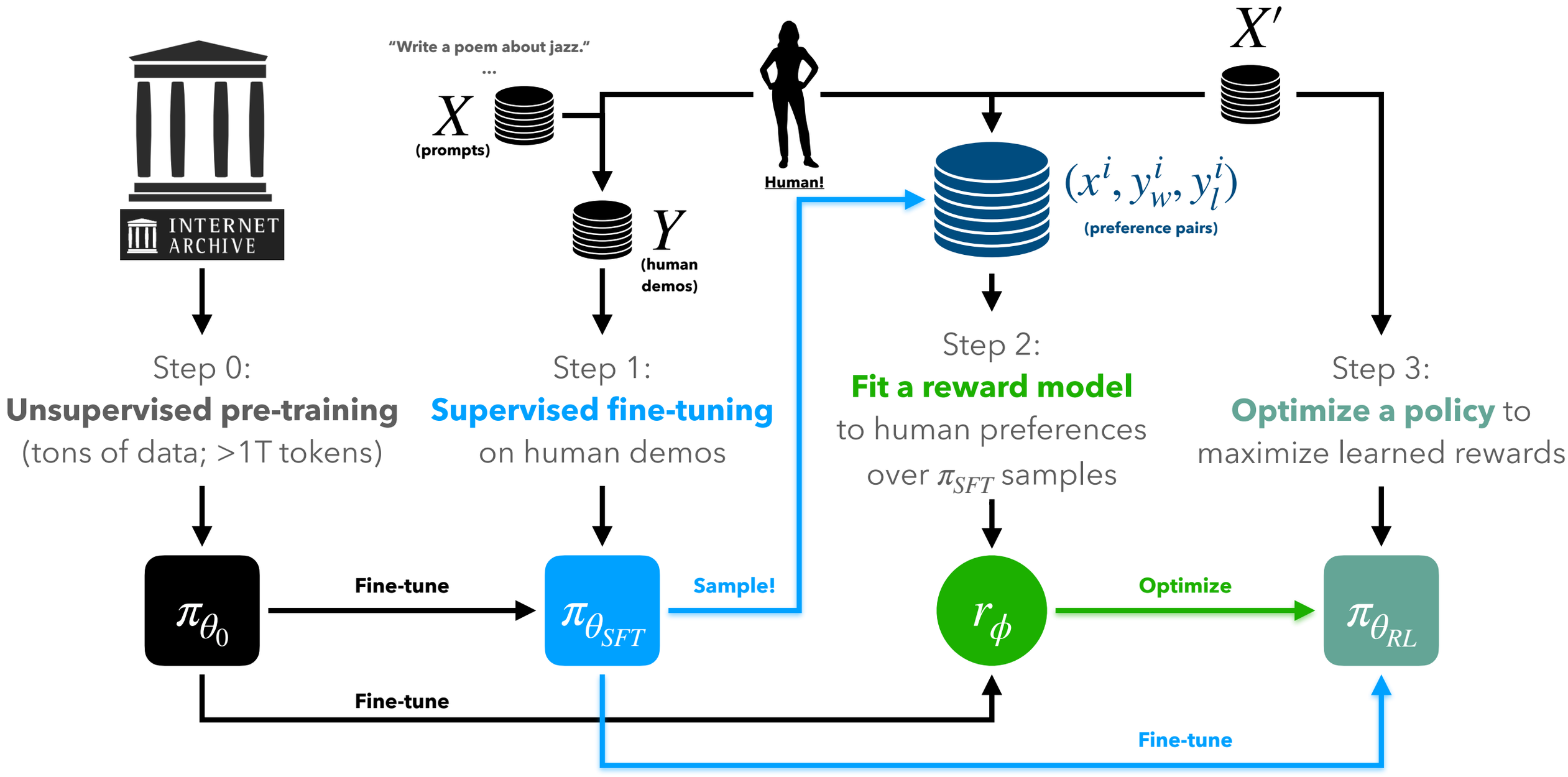
MYSTERY DATA

public interaction


Reward Model
who creates this data?
all of us
(anyone who uses ChatGPT)
ChatGPT
MYSTERY DATA
public interaction

Reinforcement Learning from Human Feedback

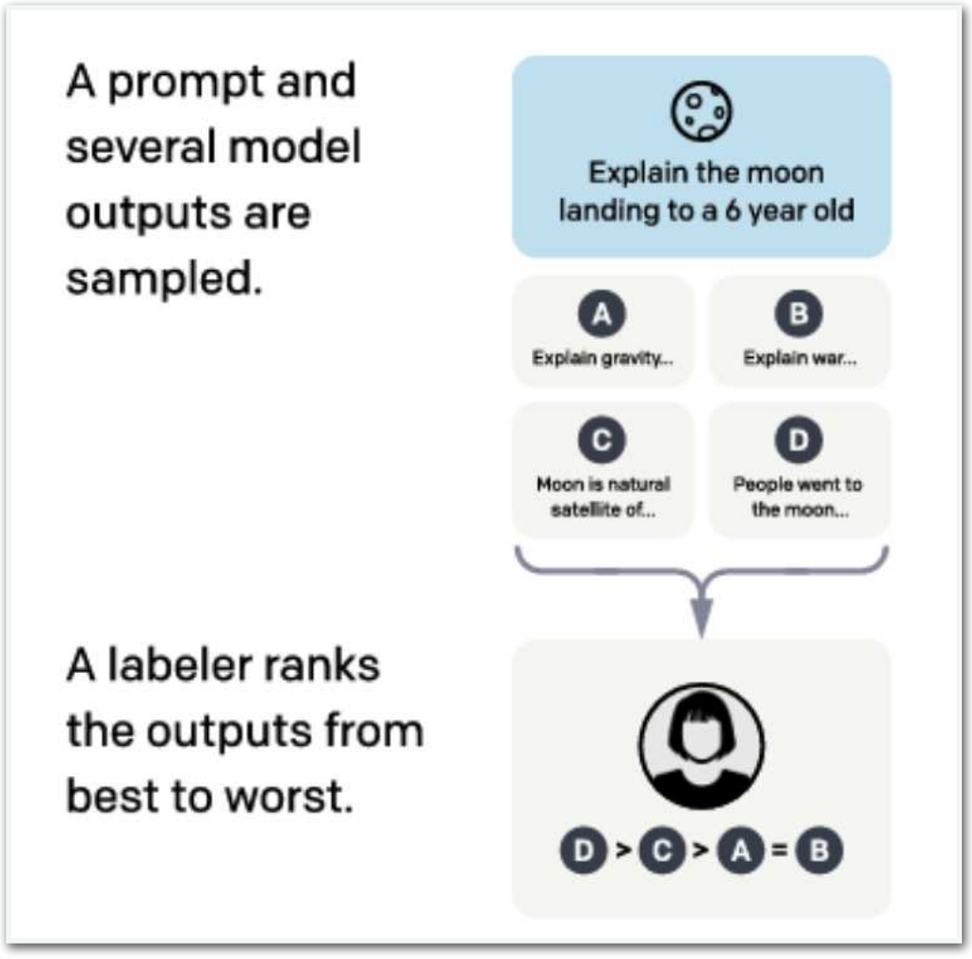
Reinforcement Learning from Human Feedback Process



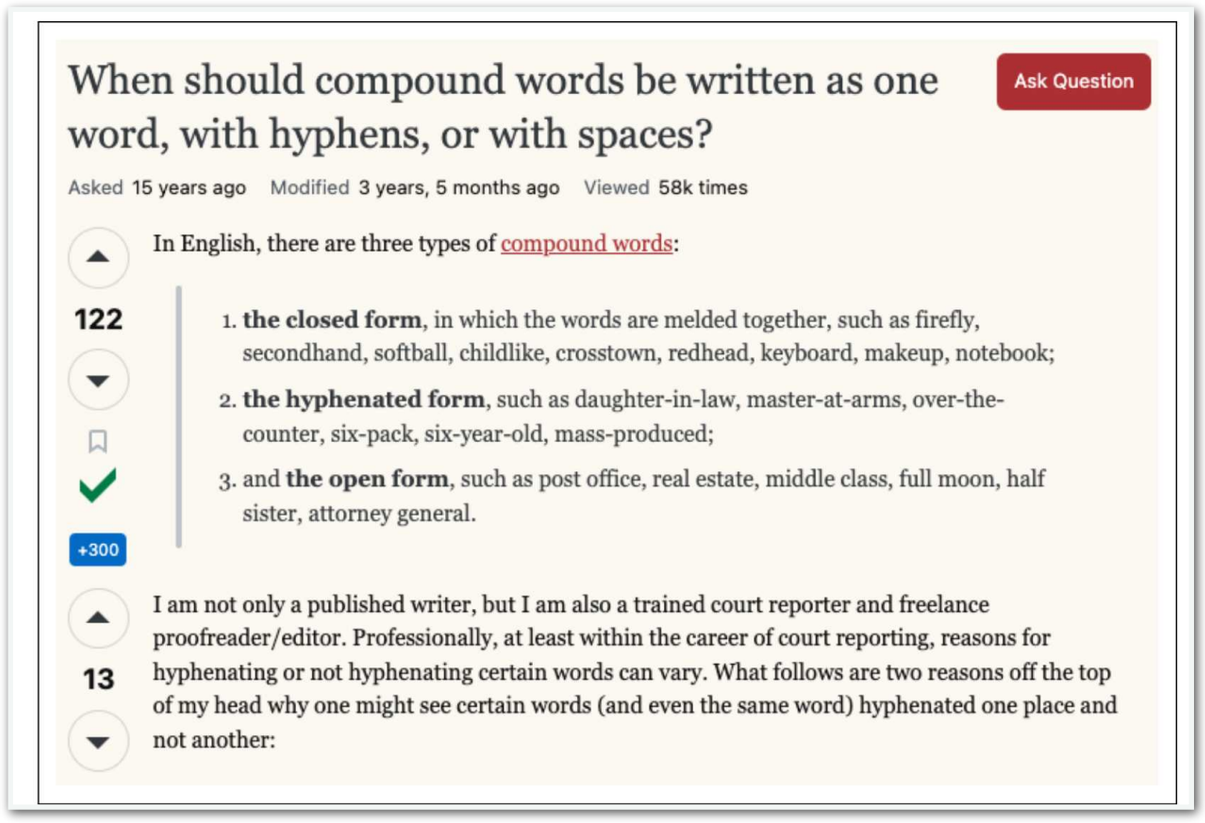
[Figure from Eric Mitchell]

Reinforcement Learning from Human Feedback

Step 1: Obtain data on human preferences

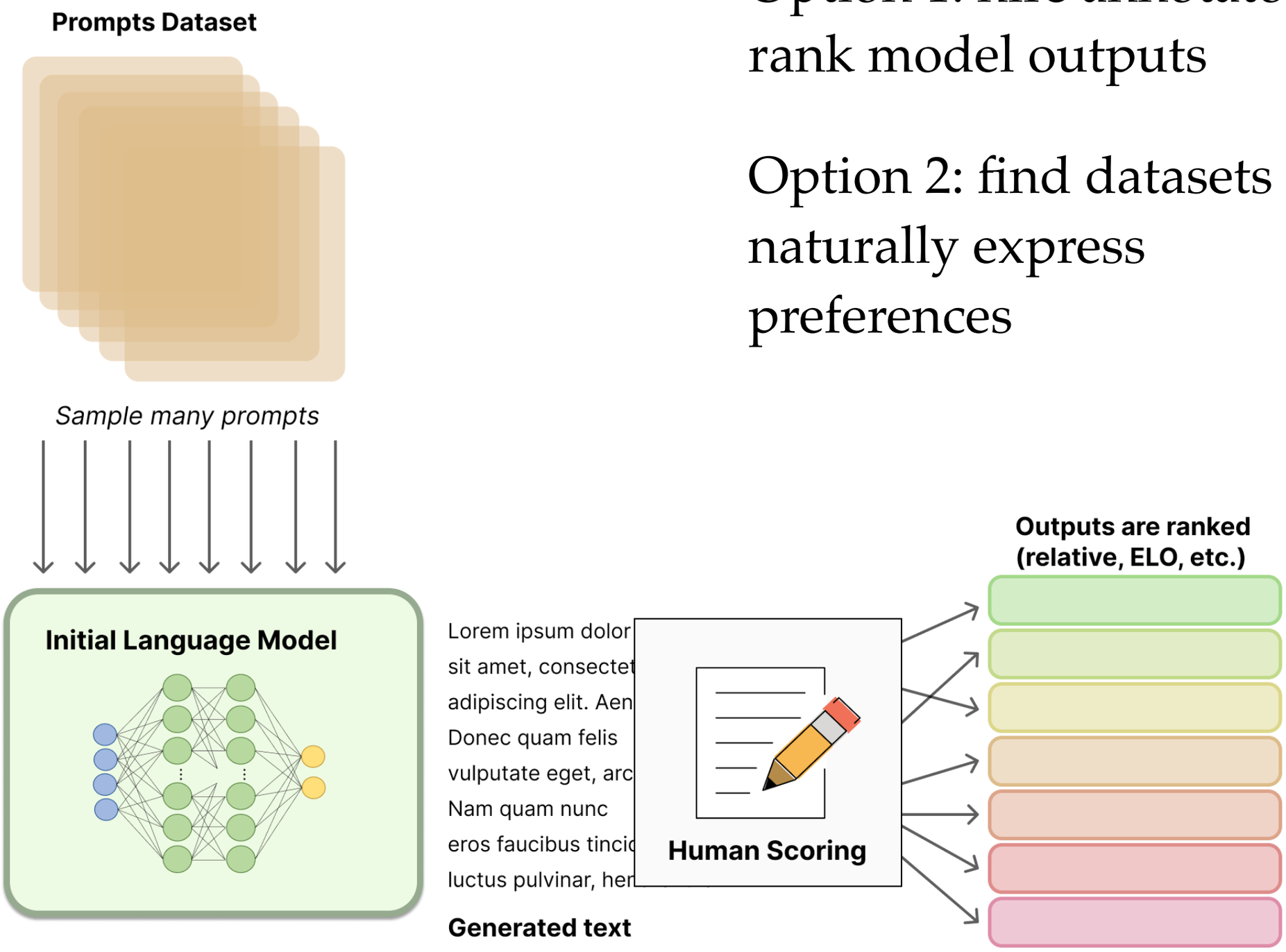


Option 1: hire annotators to rank model outputs



Option 2: find natural datasets of preference data

Step 1: Obtain data on human preferences



Option 1: hire annotators to rank model outputs

Option 2: find datasets that naturally express preferences

https://huggingface.co/blog/rlhf

Step 2: Model Preferences

Bradley-Terry Model:

Given two outputs o_i and o_j with associated scores z_i and z_j:

Step 2: Model Preferences

Bradley-Terry Model:

Given two outputs o_i and o_j with associated scores z_i and z_j:

$$P(o_i \succ o_j|x) = \sigma(z_i - z_j)$$

(the logistic sigmoid of the difference in the scores)

Step 3: Learn a Reward Function

Goal: Learn a model that predicts the rewards of different inputs.

$$P(o_i \succ o_j | x) = \sigma(z_i - z_j)$$

In principle, the reward model could be any kind of classification model. In practice, we often fine-tune a pretrained LLM.

Step 3: Learn a Reward Function

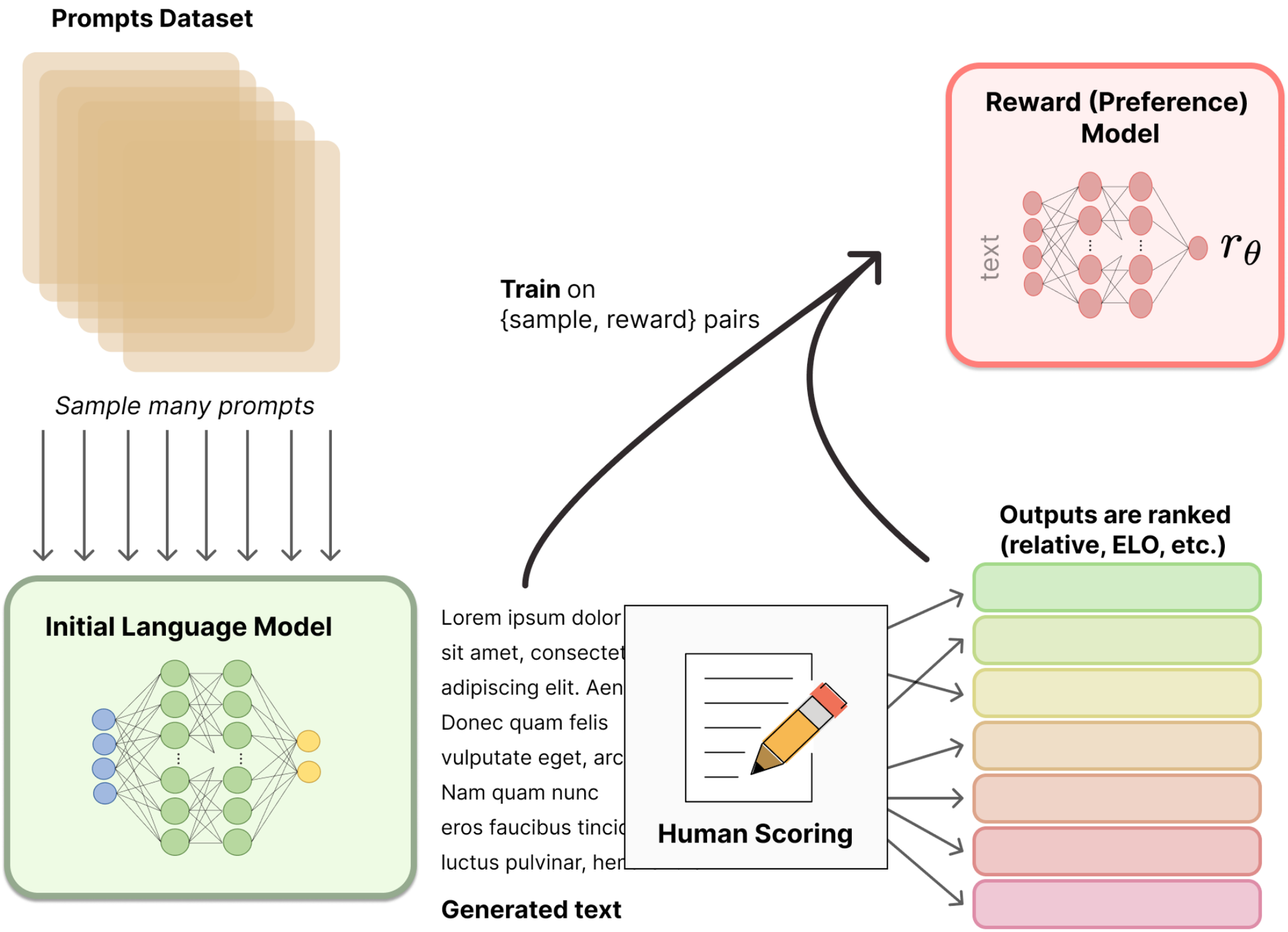
Goal: Learn a model that predicts the rewards of different inputs.

$$\begin{aligned} P(o_i \succ o_j | x) &= \sigma(z_i - z_j) \\ &= \sigma(r(o_i, x), r(o_j, x)) \end{aligned}$$

where r is the **reward** assigned to each output

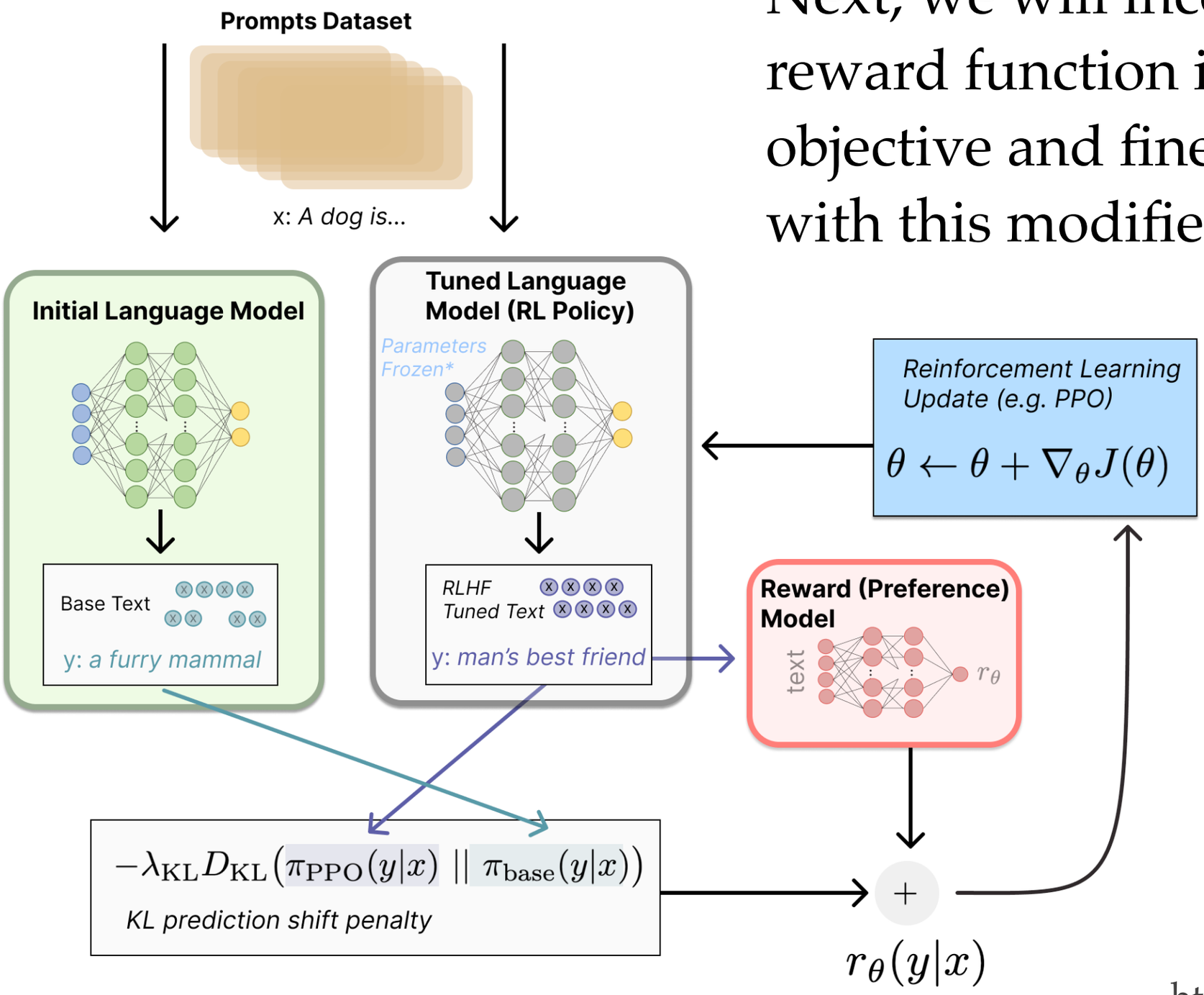
In principle, the reward model could be any kind of classification model. In practice, we often fine-tune a pretrained LLM.

Step 3: Learn a Reward Function



https://huggingface.co/blog/rlhf

Step 4: Use Reward Function to Fine-Tune



Next, we will incorporate the reward function into the training objective and fine-tune our model with this modified objective.

https://huggingface.co/blog/rlhf

Step 4: Use Reward Function to Fine-Tune

Problem: LLM can drift too far and forget what it learned in pretraining.

Solution: Incorporate the **Kullback-Leibler divergence** between the pretrained model and the reward-tuned model into the reward function (i.e. reward models that stay fairly similar to the original model).

KL divergence: measures the distance between two probability distributions.

Reinforcement Learning from Human Feedback

Takeaways

- A pretty complex process with many tricky implementation details
- Hard to get it to work — both reward modeling and RL
- Very costly — both compute and data annotation
- But, works really well
- There are alternatives that involve building preference signals into fine-tuning directly, rather than training a separate reward model (Direct Policy Optimization).
- Basically all state-of-the-art models at this point go through some kind of reinforcement learning for preferences.