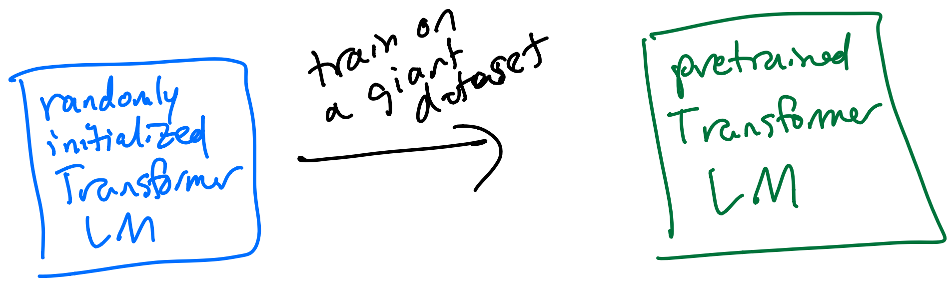
Pretraining vs. Post-training:

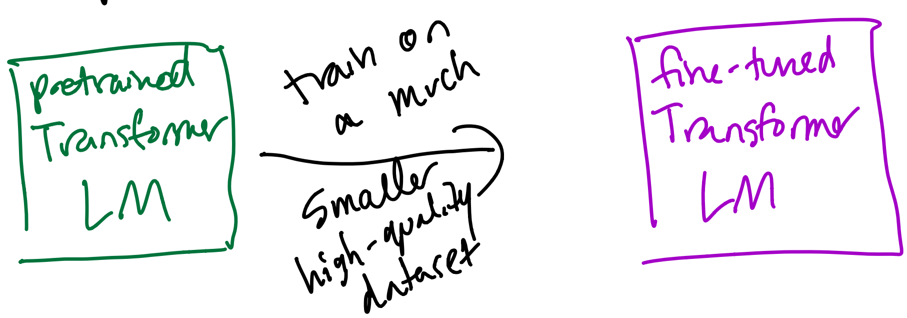
- pretraining is conducted w/ as much text as we can obtain
  - trillions of tokens (Common Crawl)
  - biggest model that we can afford
  - goal: to obtain a model that understands many linguistic properties
    - grammar
    - world knowledge
      - who is the president of France?
    - "emergent properties"
      - in-context learning

- post-training
  - goal: 1. make a pretrained model follow instructions better
    2. align the model w/ human intents / values

Step 1 (pretraining step):



Step 2 (fine-tuning):



Supervised fine-tuning ; SFT

↳ instruction tuning

↳ goal: make LM follow instructions

1. Collect a dataset of instructions on tasks to solve, and outputs for each instruction.

   ↳ optionally: collect chains of thought (explanations)

Sample instruction:

Please tell me when the exam will be in this class and how hard it will be!

Output:

The exam will be sometime in April. It may or may not be hard.

instruction (prefix) → pretrained LM → output

Reinforcement learning from human/AI feedback (RLHF/RLAIF):

Step 1 (pretraining step):

randomly initialized Transformer LM $\xrightarrow{\text{train on a giant dataset}}$ pretrained Transformer LM

Step 2 (instruction tuning):

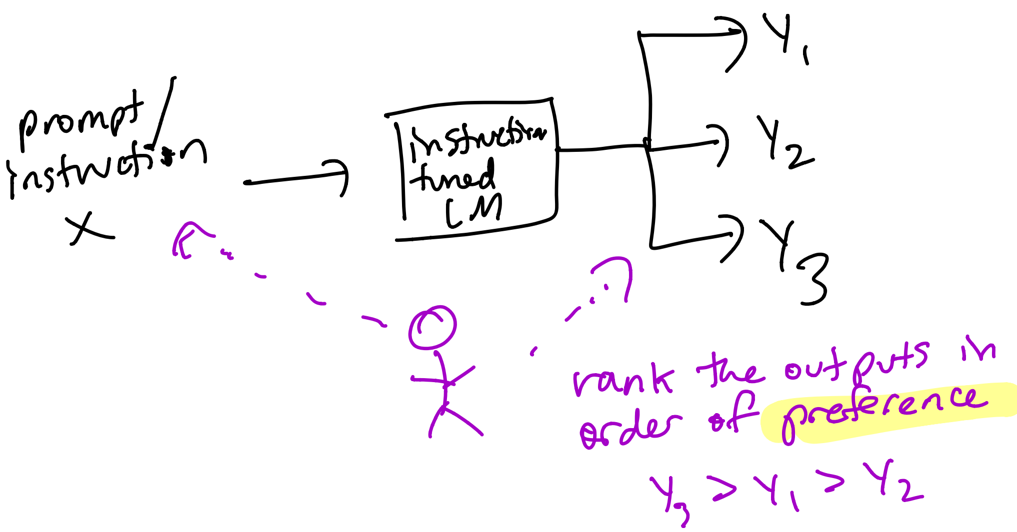
pretrained Transformer LM $\xrightarrow[\text{smaller high-quality dataset}]{\text{train on a much}}$ fine-tuned Transformer LM

Step 3 (RLHF):

instruction-tuned LM $\xrightarrow[\text{to maximize reward}]{\text{reinforcement learning}}$ final LM

Limitations of instruction tuning

- you only observe one acceptable output per instruction
- data diversity issues
- don't learn from negative feedback



limitation: human prefs are expensive to collect

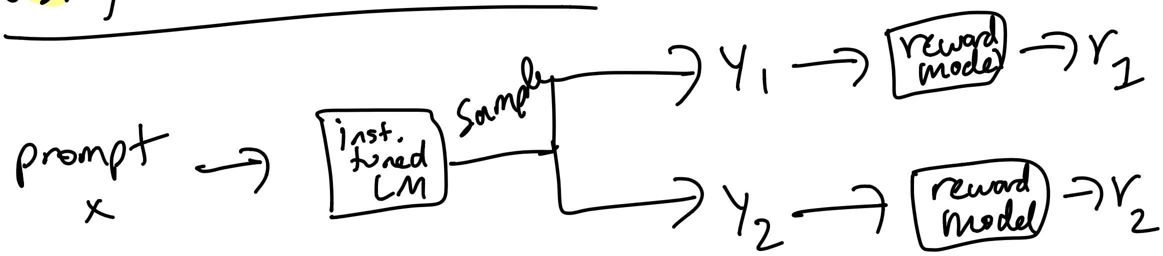
idea: can we train a model to imitate human raters?

reward model:

- input: prompt x, output y_i
- output: scalar score

↳ Bradley-Terry pairwise pref model

using the reward model:



1. "best of n" sampling
   - ↳ generate n samples, score each one, and then choose sample w/ highest reward
     - ↳ very expensive!
2. just fine-tune the LM on the highest-scoring sample y_w
   - ↳ fine-tune to maximize $p(y_w | x)$
   - ↳ RAFT
3. reinforcement learning to increase $p(y_w | x)$ by a small amount, decrease $p(y_L | x)$ by a small amount. amounts are functions of

   $$r(x, y_w), \; r(x, y_L)$$

$\pi_{ref} \Rightarrow$ instruction-tuned model

$\pi \Rightarrow$ current policy models

↳ initialized to π_{ref}

final model

$$\max_{\pi} \; \mathbb{E}_{x,y} \Big[\underbrace{r(x,y)}_{\text{reward}} - \underbrace{\beta D_{KL}\big(\pi(y|x) \,\|\, \pi_{ref}(y|x)\big)}_{\text{penalty for deviating too much from inst. tuned model}} \Big]$$