DS 4440

"Post" Training

or, "Aligning" LLMs

Large Language Models (LLMs) are stacks of Transformer layers

2018 BERT (10/23 lecture)
340M params

2019 GPT-2
1.5b params

2020 GPT-3
~175b params

More capable but problematic

+ LM objective only not super useful for most tasks.

Pre-Training provides the base.

Post-Training is about making models useful, or — if you're fancy — alignment.

Instruction Tuning (SFT) is a simple approach which aims to teach models to follow instructions.

Compile standard supervised tasks and pre-pend instructions

x_1 Today the Trump Campaign ... y_1 Trump ...
(Summarization)

x_2 This movie rocked ... y_2 Positive
(classification)

↓

Summarize the following document:
Today the Trump campaign

Classify this review as positive or

negative. This movie rocked.

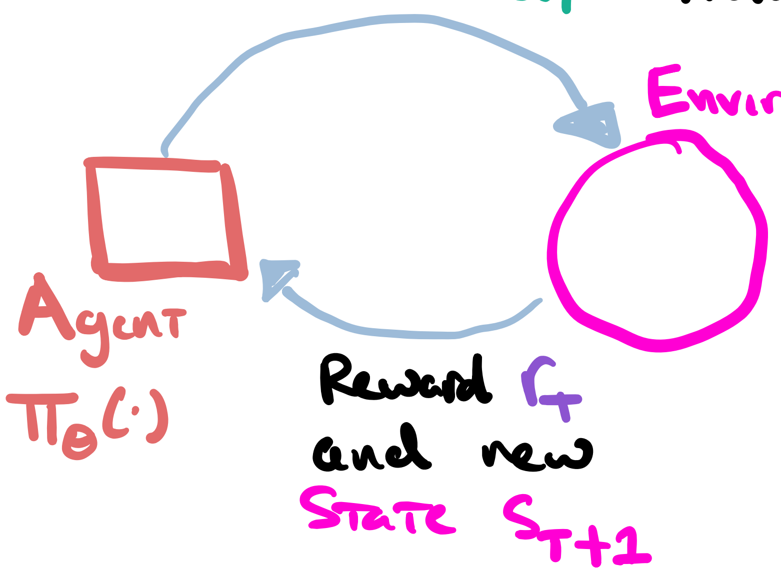
(See Slides)

Reinforcement Learning from Human Feedback (RLHF)

Why RL? Not clear how to supervise learn our way to avoid toxic outputs or produce funnier jokes.

RL

Take action a_t from state s_t



$$\underbrace{R(\tau) = \sum_{t=1}^{T} \gamma^{t} r_t}_{\text{Total discounted future reward}}$$

$a_t \sim \pi_\theta(s_t)$ Agent policy picks actions to maximize $R(\tau)$.

Text

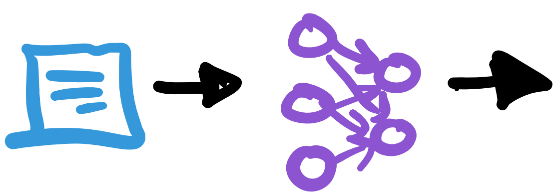
$\downarrow$

Reward Model r_w

$\downarrow$

Reward

Learn reward model based on human preference feedback $y_j \succ y_k$

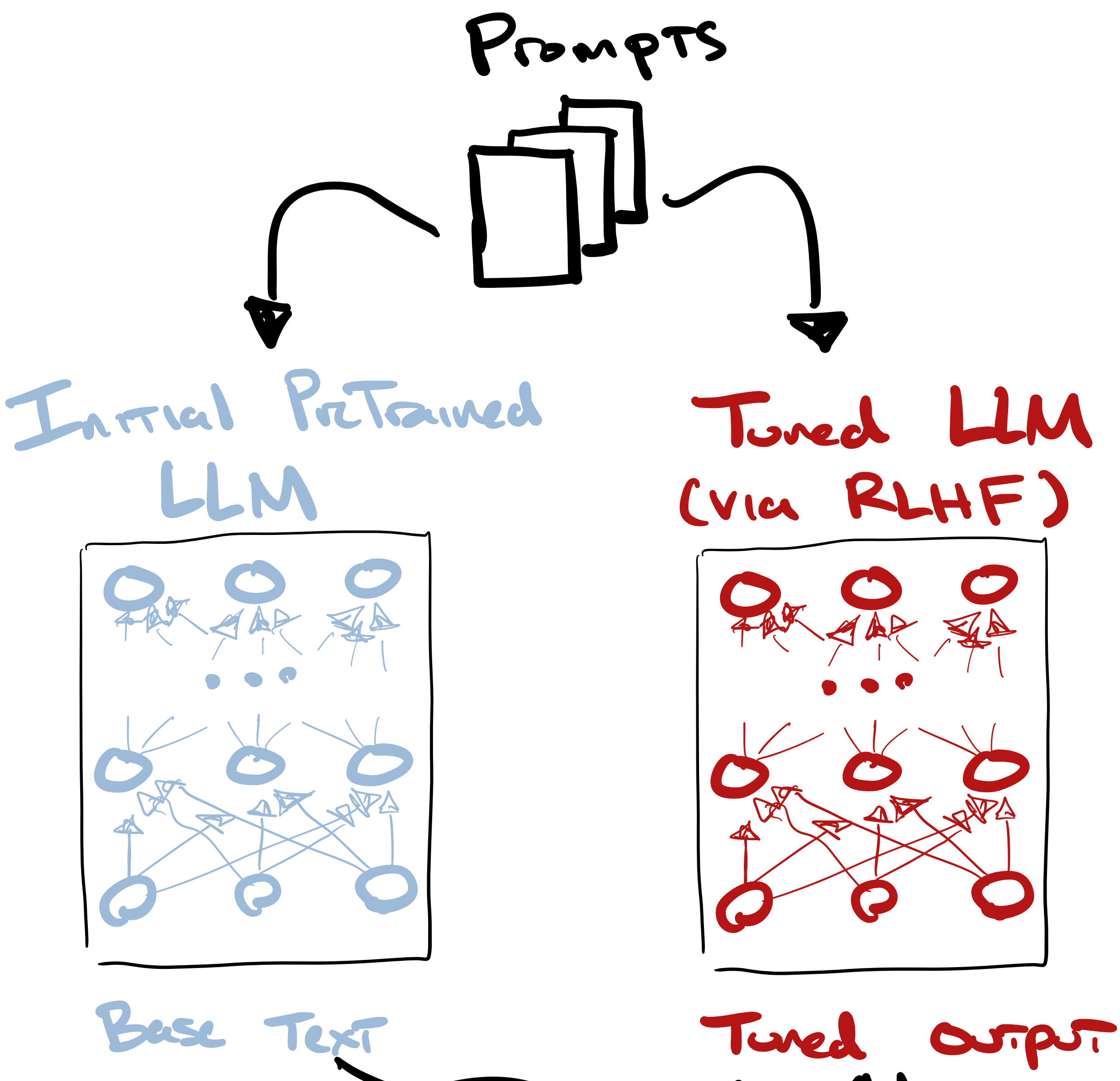
Train $\underline{r_w}$  $\rightarrow \hat{r}$

Separate from LLM!

So r_w provides rewards from environment. States & actions both language.

Once we have r_w, update LLM to yield outputs w/ bigger rewards.

<u>Problem</u> The LLM might go off the rails — optimize r but produce jibberish.



$-\lambda \, \underbrace{KL(\text{Tuned}, \text{Base})}_{\text{Penalty for diverging from Base pretrained LLM}}$

$\nabla_\theta J$ (policy gradient)

Adjust θ to ↑ reward

$r_W(\text{Tuned})$

$$\nabla_\theta J \ddot{=}$$

$$\underset{\tau \sim \pi_\theta}{\mathbb{E}} \left[\sum_{t=0}^{T} \underbrace{\nabla_\theta \log \pi_\theta (a_t | s_t)}_{\text{Nothing new}} \; \underbrace{r_W(\tau)}_{\text{full output trajectory}} \right]$$

(Likelihood under LLM)

Push token probs up in proportion to reward score

Direct Preference Optimization (DPO)

But who wants to deal w/ RL? Rafailov et al. (2023) present DPO to learn directly from preference feedback — no reward model or RL required.

(See Slides)