Introduction to Large Language Models

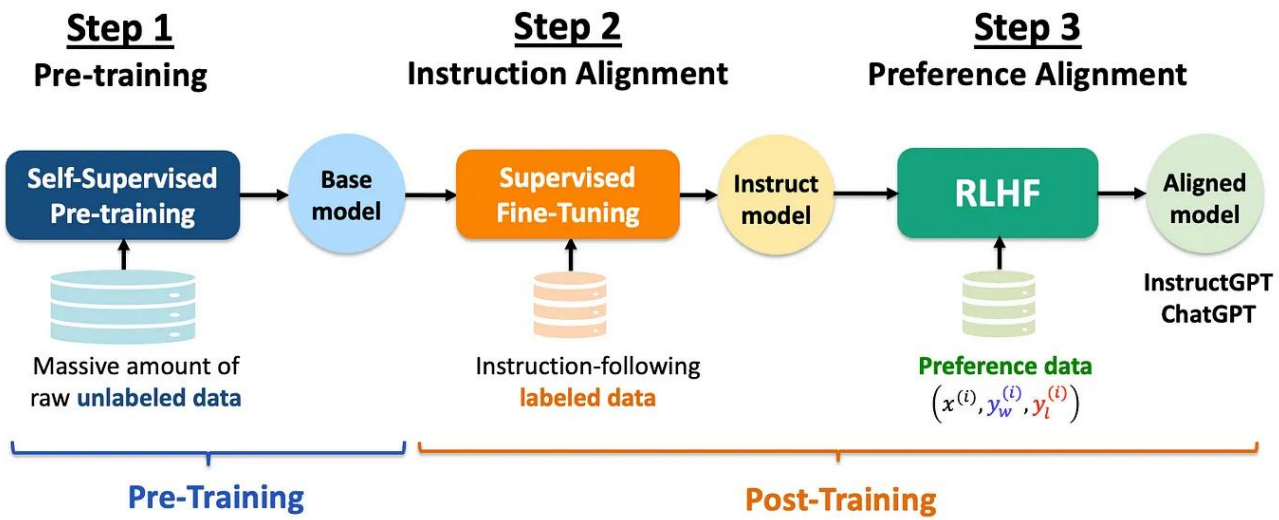
Spring 2026

LLM Training
Post-Training: Preference Alignment

(Some slides adapted from Ralph Grishman at NYU,
Yejin Choi at UWashington, N. Tomura at UDepaul, Jurafsky and Martin, CS224N,
CS224, CME295 at Stanford and other resourses on the web)

Pre-Training vs Post Training

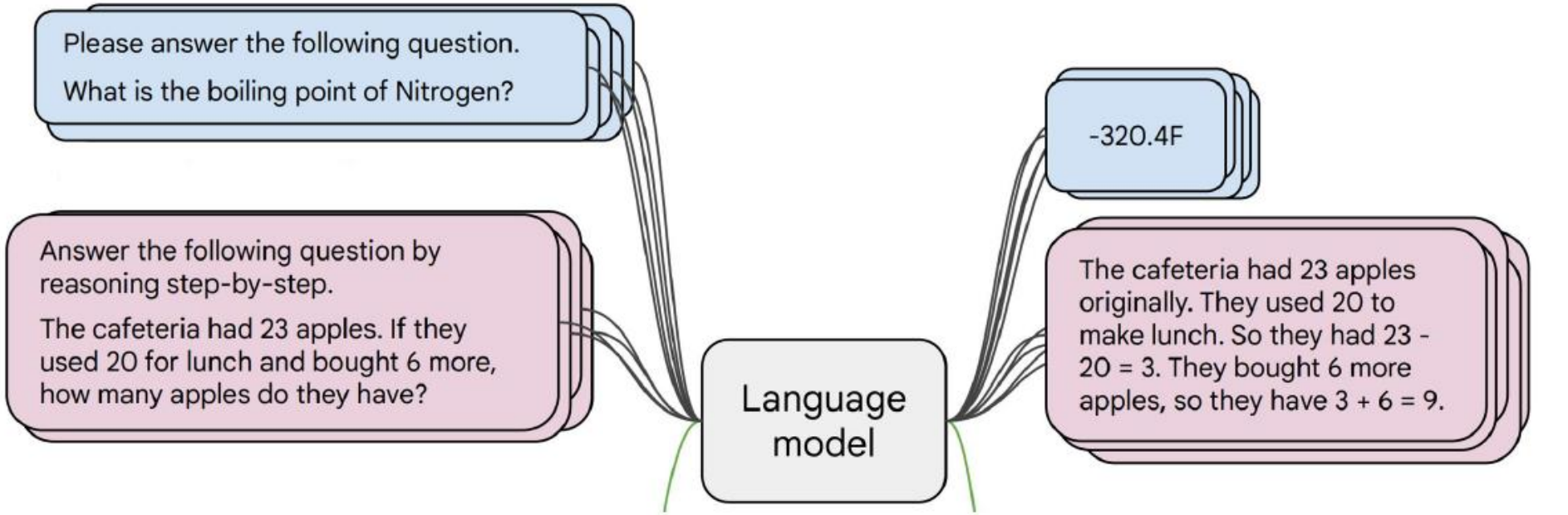
- **Pre-training** is the **first and largest** stage of training an LLM. **Pre-training** is the phase where a large language model learns general language patterns by analyzing massive amounts of text. **Outcome is a base/foundational model** that:
  - Knows language well
  - Generates coherent text
  - But **does NOT reliably follow instructions**
  - And is **not aligned** with human values or safety expectations
- **Post-training** refers to *everything done to a base model after pre-training* to make it more useful, safer, and better aligned with human expectations.
  - **Instruction tuning** (often called supervised fine-tuning, SFT) is the process of fine-tuning a pretrained large language model on a dataset of **(instruction, output) pairs**, with the explicit goal of **teaching the model to follow natural-language instructions** rather than merely continue text statistically. It directly bridges the gap between next-token pretraining objectives and user intent.
  - **Preference alignment** refers to the process of training LLMs so that their outputs **conform to human preferences, norms, and values**—such as being helpful, truthful, safe, polite, and contextually appropriate—rather than merely statistically likely continuations of text.



https://todatabeyond.substack.com/p/visual-guide-to-llm-preference-tuning

Instruction tuning

- **Collect examples** of (instruction, output) pairs across many tasks and finetune an LM



- Evaluate on **unseen tasks**

[FLAN-T5; Chung et al., 2022]

Source : cs224n

Super-NaturalInstructions dataset for Instruction tuning

Instruction ~~finetuning~~ pretraining?

- As is usually the case, **data + model scale** is key for this to work!
- **Super-NaturalInstructions** dataset contains **over 1.6K tasks, 3M+** examples
  - Classification, sequence tagging, rewriting, translation, QA...

Q: how do we evaluate such a model?

[Wang et al., 2022]

Source : cs224n

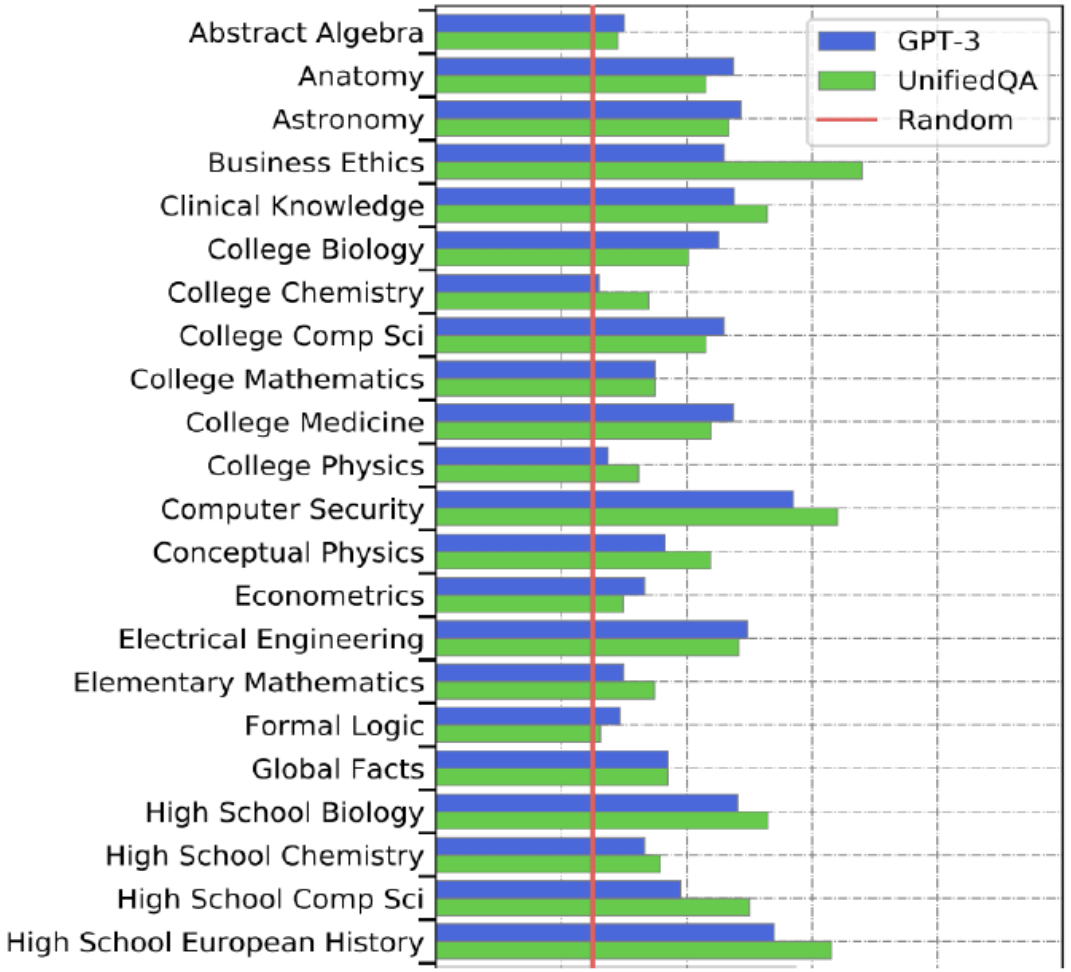
Instruction Tuning Evaluation

- Goal: Assess ability to follow instructions for novel tasks.
- Method: Leave-one-out evaluation at task cluster level.
  - Example: Remove all sentiment analysis datasets for testing.
  - Ensures no overlap with training tasks.
- Metrics:
  - Accuracy (classification)
  - chrF (translation)
  - ROUGE (summarization)
- Benchmarks

Benchmarks for multitask LMs

Massive Multitask Language Understanding (MMLU)
[Hendrycks et al., 2021]

New benchmarks for measuring LM performance on 57 diverse *knowledge intensive* tasks



UnifiedQA is a single, unified question-answering (QA) model proposed by Khashabi et al. (EMNLP 2020) that can handle multiple QA formats with one model, instead of using separate architectures for each format. It uses T5 as base model.

Source : cs224n

Some intuition: examples from MMLU

Astronomy

What is true for a type-Ia supernova?

A. This type occurs in binary systems.
B. This type occurs in young galaxies.
C. This type produces gamma-ray bursts.
D. This type produces high amounts of X-rays.

Answer: A

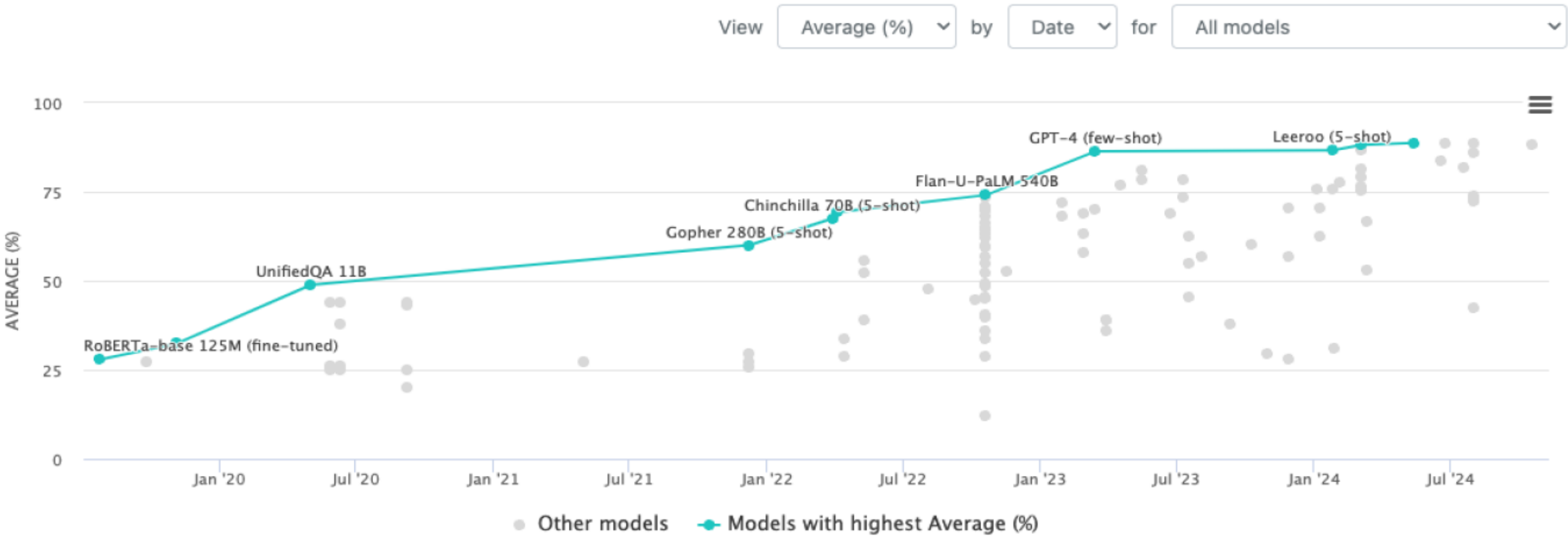
High School Biology

In a population of giraffes, an environmental change occurs that favors individuals that are tallest. As a result, more of the taller individuals are able to obtain nutrients and survive to pass along their genetic information. This is an example of

A. directional selection.
B. stabilizing selection.
C. sexual selection.
D. disruptive selection

Answer: A

Progress on MMLU

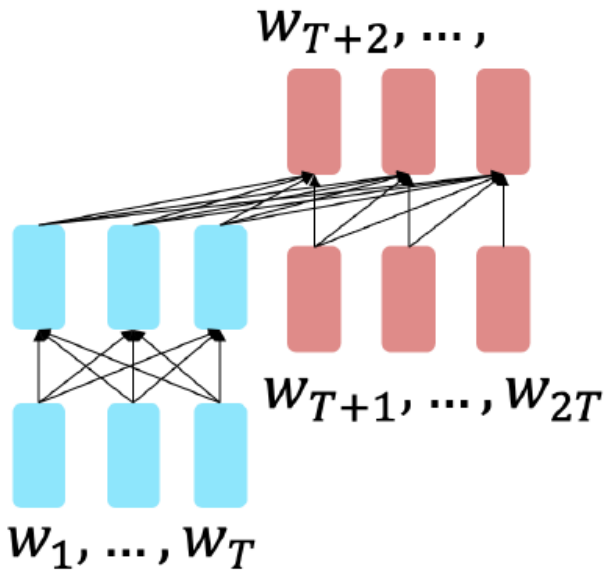


- Rapid, impressive progress on challenging knowledge-intensive benchmarks

Source : cs224n

Instruction finetuning and performance gains

- Recall the T5 encoder-decoder model [Raffel et al., 2018], pretrained on the **span corruption** task
- **Flan-T5** [Chung et al., 2022]: T5 models finetuned on 1.8K additional tasks



Params	Model	BIG-bench + MMLU Norm. avg.
80M	T5-Small	-9.2
	Flan-T5-Small	-3.1 (+6.1)
250M	T5-Base	-5.1
	Flan-T5-Base	6.5 (+11.6)
780M	T5-Large	-5.0
	Flan-T5-Large	13.8 (+18.8)
3B	T5-XL	-4.1
	Flan-T5-XL	19.1 (+23.2)
11B	T5-XXL	-2.9
	Flan-T5-XXL	23.7 (+26.6)

Bigger model = bigger Δ

FLAN (short for Finetuned Language Net) is a family of instruction-tuned language models introduced by Google Research, first described in “Finetuned Language Models Are Zero-Shot Learners” (Wei et al., 2021).

Source : cs224n

Instruction finetuning and performance gains

Model input (Disambiguation QA)

Q: In the following sentences, explain the antecedent of the pronoun (which thing the pronoun refers to), or state that it is ambiguous.

Sentence: The reporter and the chef will discuss their favorite dishes.

Options:
(A) They will discuss the reporter's favorite dishes
(B) They will discuss the chef's favorite dishes
(C) Ambiguous

A: Let's think step by step.

Before instruction finetuning

The reporter and the chef will discuss their favorite dishes.
The reporter and the chef will discuss the reporter's favorite dishes.
The reporter and the chef will discuss the chef's favorite dishes.
The reporter and the chef will discuss the reporter's and the chef's favorite dishes.

✖ **(doesn't answer question)**

Source : cs224n

Instruction finetuning and performance gains

Model input (Disambiguation QA)

Q: In the following sentences, explain the antecedent of the pronoun (which thing the pronoun refers to), or state that it is ambiguous.

Sentence: The reporter and the chef will discuss their favorite dishes.

Options:
(A) They will discuss the reporter's favorite dishes
(B) They will discuss the chef's favorite dishes
(C) Ambiguous

A: Let's think step by step.

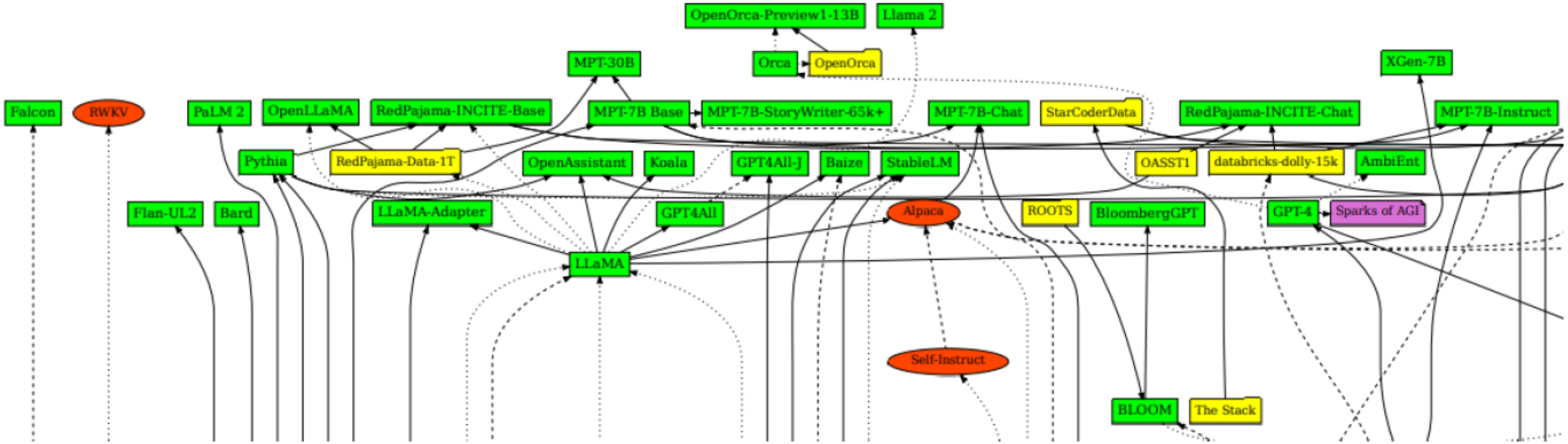
After instruction finetuning

The reporter and the chef will discuss their favorite dishes does not indicate whose favorite dishes they will discuss. So, the answer is (C). ✅

Try FLAN-T5 out to get a sense of its capabilities: https://huggingface.co/google/flan-t5-xxl [Chung et al., 2022]

Source : cs224n

A huge diversity of instruction-tuning datasets

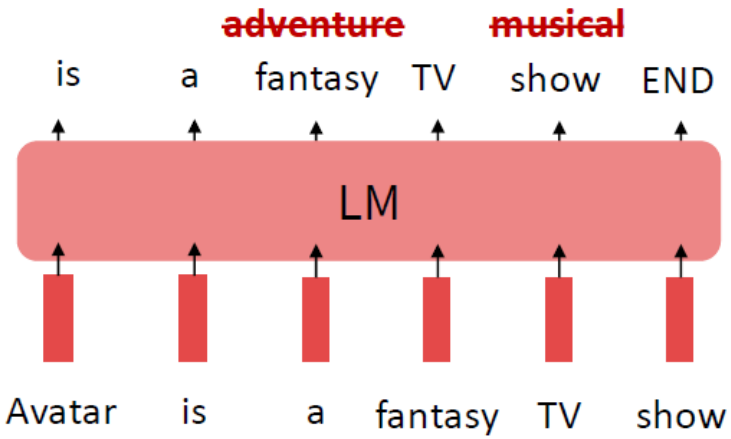


- The release of LLaMA led to open-source attempts to `create' instruction tuning data

Source : cs224n

Limitations of instruction finetuning?

- One limitation of instruction finetuning is obvious: it's **expensive** to collect ground-truth data for tasks.
- But there are other, subtler limitations too. Can you think of any?
- **Problem 1: tasks like open-ended creative generation have no right answer.**
  - *Write me a story about a dog and her pet grasshopper.*
- **Problem 2: language modeling penalizes all token-level mistakes equally, but some errors are worse than others.**
- Even with instruction finetuning, there a mismatch between the LM objective and the objective of "satisfy human preferences"!
- Can we **explicitly attempt to satisfy human preferences?**



Source : cs224n

Preference Alignment

- **Preference alignment** refers to the process of training LLMs so that their outputs conform to **human preferences**, norms, and values—such as being helpful, truthful, safe, polite, and contextually appropriate—rather than merely statistically likely continuations of text
- **Unaligned or weakly aligned LLMs** tend to:
  - Respond confidently with false information (hallucination)
  - Optimize for verbosity and politeness rather than correctness
  - Reflect annotator or dataset biases
  - Fail under ambiguous or adversarial prompts
- Empirical studies show that **human preference optimization significantly improves** perceived usefulness, safety, and trustworthiness of LLM systems

Preference-based alignment

Preference-based alignment trains a language model not just to produce *correct* answers, but to produce answers that **humans prefer**.

Instead of learning from:

> *"This is the right answer."*

The model learns from:

> *"Between these two answers, humans prefer A over B."*

This shift directly addresses the core problems of instruction fine-tuning:

- No single correct answer for many tasks
- Token-level loss misaligned with human judgment
- Difficulty encoding abstract qualities like helpfulness, politeness, safety

Preference Data Types

- **Pairwise Comparisons**
  - Most common

    $(y^+ > y^-)$
- **2. Ranked Lists**

  $A > B > C$
- **3. Scalar Feedback**

  Score from 1–5
- **4. AI Feedback (RLAIF)**
  - Replace humans with models:
    - **Reinforcement Learning from AI Feedback**

Preference Data

The quality of preference data is often *more important* than the algorithm:

- **Human annotation** — Gold standard but expensive (Likert scales, pairwise comparison)
- **AI feedback (RLAIF)** — Scalable; Anthropic's Constitutional AI uses a set of principles for self-critique
- **Synthetic data** — Generated pairs from stronger models (common in open-source pipelines)
- **Implicit signals** — Click-through, upvotes, session length (noisier)

Preference-based alignment methods

- **1. Reinforcement Learning from Human Feedback (RLHF)** The foundational approach (Christiano et al., 2017; InstructGPT, 2022):
  - **SFT** — Fine-tune on high-quality demonstrations
  - **Reward model** — Train a separate model on pairwise human preferences (A vs B)
  - **RL optimization** — Use PPO to maximize reward while penalizing KL divergence from the reference policy

- *Strengths:* Effective; well-studied *Weaknesses:* Unstable training, reward hacking, expensive human labeling, requires separate reward model

Preference-based alignment methods

2. Direct Preference Optimization (DPO) *(Rafailov et al., 2023)*

Key idea:

- Skip reward model + RL
- Optimize directly from preference pairs

- Eliminates the RL loop entirely by deriving a closed-form **loss** directly from preference data:

$$\mathcal{L}_{DPO} = -\mathbb{E}\left[\log \sigma\left(\beta \log \frac{\pi_\theta(y_w \mid x)}{\pi_{ref}(y_w \mid x)} - \beta \log \frac{\pi_\theta(y_l \mid x)}{\pi_{ref}(y_l \mid x)}\right)\right]$$

Where y_w =preferred output, y_l =dispreferred, β= temperature.

Strengths: Simpler, stable, no reward model needed *Weaknesses:* Sensitive to reference model quality; can forget general capabilities

Extensions & Variants

Method	Key Idea
IPO (Azar et al., 2023)	Fixes DPO's implicit reward assumption; more robust
KTO (Ethayarajh et al., 2024)	Uses individual good/bad labels instead of pairs (Kahneman-Tversky loss)
ORPO (Hong et al., 2024)	Merges SFT and alignment into a single stage
SimPO (Meng et al., 2024)	Reference-free DPO variant using average log-prob
RLAIF	Replaces human raters with an AI judge (Constitutional AI)
RLVR	Uses verifiable rewards (e.g., math correctness) — used in DeepSeek-R1, o1

Extensions & Variants

- **Constitutional AI (Anthropic)**
  - A notable variant that reduces reliance on human labels:
  - Model critiques its own outputs using a written *constitution*
  - Revises outputs accordingly
  - These revised pairs become preference training data for RLAIF

‘RLHF’ pipeline

Step 1

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.

Step 2

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.

Step 3

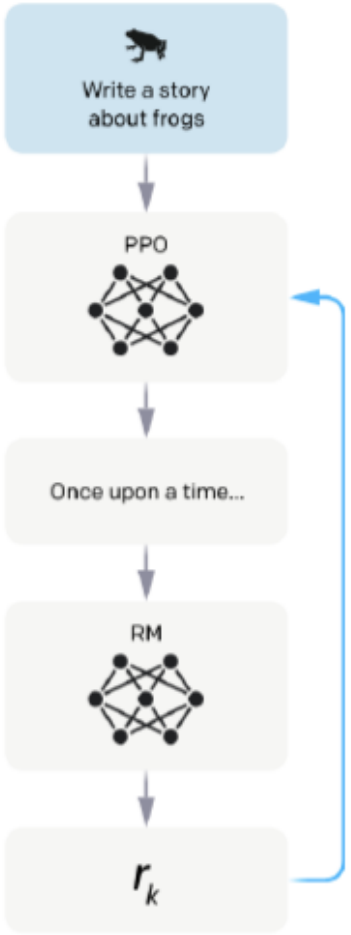
Optimize a policy against the reward model using reinforcement learning.

A new prompt is sampled from the dataset.

The policy generates an output.

The reward model calculates a reward for the output.

The reward is used to update the policy using PPO.



Source : cs224n

Optimizing for human preferences

- Let's say we were training a language model on some task (e.g. summarization).
- For an instruction x and a LM sample y, imagine we had a way to obtain a *human reward* of that summary: $R(x, y) \in \mathbb{R}$, higher is better.

x

```
SAN FRANCISCO,
California (CNN) --
A magnitude 4.2
earthquake shook the
San Francisco
...
overturn unstable
objects.
```

y_1

```
An earthquake hit
San Francisco.
There was minor
property damage,
but no injuries.
```

$R(x, y_1) = 8.0$

y_2

```
The Bay Area has
good weather but is
prone to
earthquakes and
wildfires.
```

$R(x, y_2) = 1.2$

- Now we want to maximize the expected reward of samples from our LM:

$$\mathbb{E}_{\hat{y} \sim p_\theta(y \mid x)}[R(x, \hat{y})]$$

Source : cs224n

How do we get the rewards?

- **Problem 1:** human-in-the-loop is expensive!
  - **Solution:** instead of directly asking humans for preferences, **model their preferences** as a separate (NLP) problem! [Knox and Stone, 2009]

An earthquake hit San Francisco. There was minor property damage, but no injuries.

The Bay Area has good weather but is prone to earthquakes and wildfires.

Train a $RM_\phi(x, y)$ to predict human reward from an annotated dataset, then optimize fo RM_ϕ instead.

$R(x, y_1) = 8.0$ 

$R(x, y_2) = 1.2$ 

Source : cs224n

How do we model human preferences?

- **Problem 2:** human judgments are noisy and miscalibrated!
- **Solution:** instead of asking for direct ratings, ask for **pairwise comparisons**, which can be more reliable [Phelps et al., 2015; Clark et al., 2018]

A 4.2 magnitude earthquake hit San Francisco, resulting in massive damage.

$$R(x, y_3) = \begin{matrix} y_3 \\ 4.1? \end{matrix} \quad \begin{matrix} \\ 6.6? \end{matrix} \quad \begin{matrix} \\ 3.2? \end{matrix}$$

Source : cs224n

Reinforcement Learning from Human Feedback (RLHF)

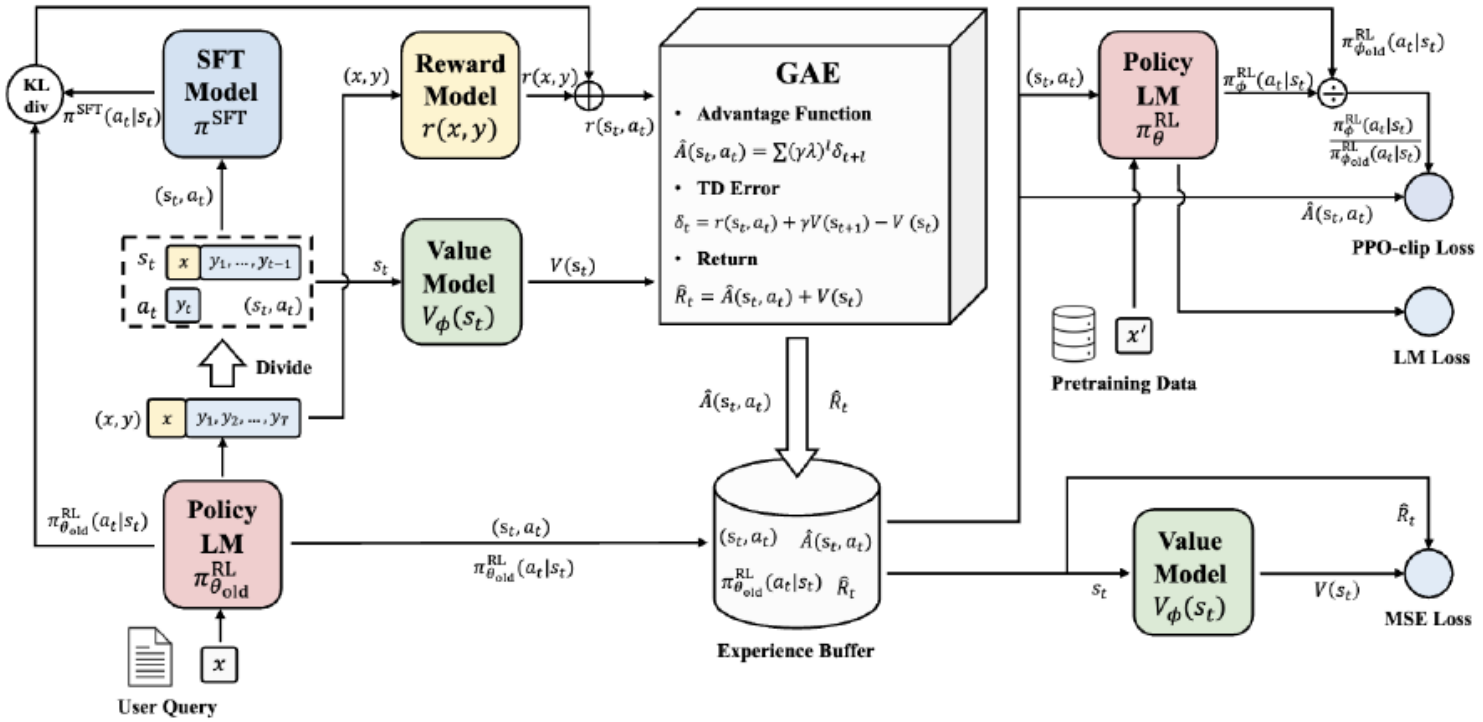
- The most widely known approach is **Reinforcement Learning from Human Feedback (RLHF)**.
- **Pipeline:**
  - **Pretraining**
    - Train base LLM on large text corpora.
  - **Supervised Fine-Tuning (SFT)**
    - Humans write high-quality responses → model imitates them.
  - **Preference Data Collection**
    - Annotators rank outputs:
    - Prompt: "Explain quantum computing"
    - A: ...
    - B: ...
    - → Human chooses A > B
  - **Reward Model (RM)**
    - Train a model to predict human preferences:
      $[r_\theta(x, y)]$
  - **Policy Optimization**
    - Optimize LLM to maximize reward using RL (e.g. PPO)

PPO (Proximal Policy Optimization)

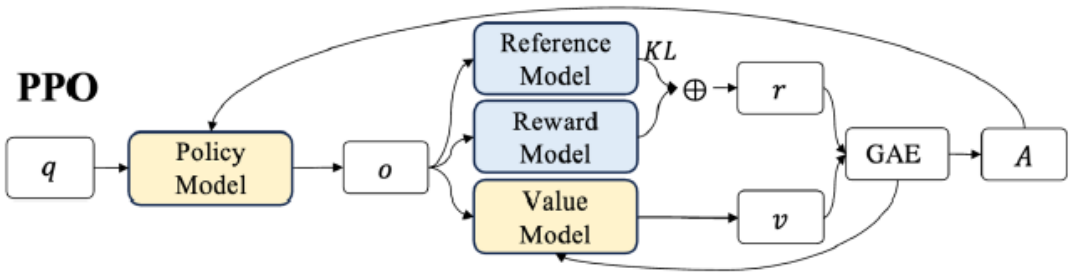
- Proximal Policy Optimization (PPO) is **a policy-gradient reinforcement learning algorithm** designed to make policy updates:
  - Stable
  - Sample-efficient
  - Easy to implement
- PPO improves a language model according to human preferences while explicitly preventing it from changing too fast or too far.
- Step-by-step RLHF with PPO:
  1. Start with an instruction-tuned LM
  2. Generate responses to prompts
  3. Score responses using a **reward model**
  4. Use PPO to update the LM:
     - Increase probability of high-reward responses
     - Penalize excessive deviation from the original model

RLHF can be complex

- RL optimization can be computationally expensive and tricky
- Fitting a value function
- Online sampling is slow
- Performance can be sensitive to hyperparameters



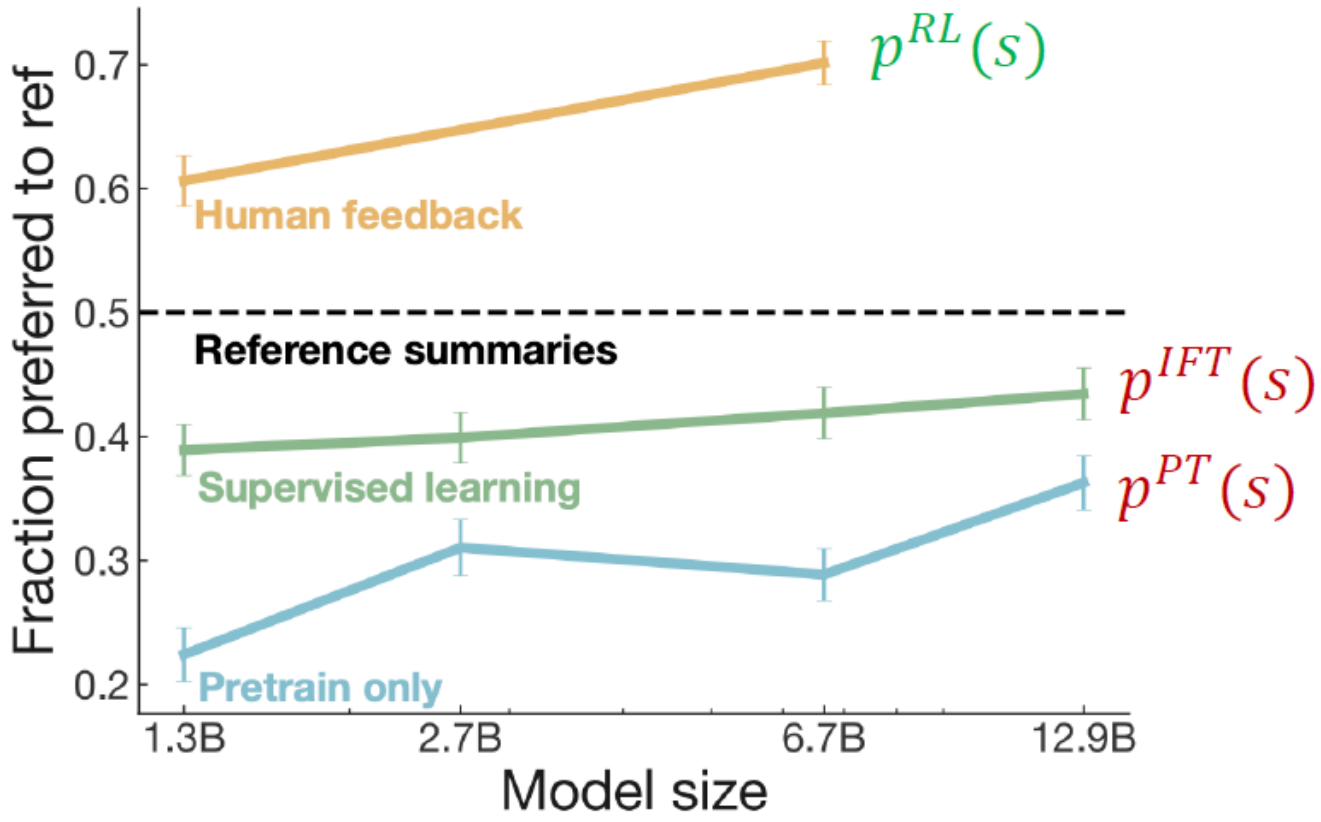
Secrets of RLHF / PPO workflow [Zheng et al., 2023]



[Shao et al., 2024]

Source : cs224n

RLHF provides gains over pretraining + finetuning



[Stiennon et al., 2020]

This plot is one of the **first strong empirical proofs** that:

1. **LM likelihood ≠ human preference**
2. **Instruction tuning alone is insufficient**
3. **Preference-based alignment is essential**
4. **Scaling amplifies alignment when objectives are correct**

Direct Preference Optimization (DPO)

- Direct Preference Optimization (DPO) is an alignment technique for large language models that fine-tunes them based on human preferences without requiring reinforcement learning.
- Unlike RLHF (Reinforcement Learning from Human Feedback), which involves training a reward model and optimizing the language model accordingly, DPO simplifies the process by using a classification loss to directly optimize preferences.
- This method is computationally efficient and stable, making it a promising alternative for aligning models to human expectations.
- It has been successfully applied to tasks like sentiment control, summarization, and dialogue generation, often outperforming RLHF-based approaches

Removing the ‘RL’ from RLHF

Recall we want to maximize the following objective in RLHF

$$\mathbb{E}_{\hat{y}\sim p_\theta^{RL}(\hat{y}|x)}[RM_\phi(x,\hat{y}) - \beta \log\left(\frac{p_\theta^{RL}(\hat{y}|x)}{p^{PT}(\hat{y}|x)}\right)]$$

There is a closed form solution to this:

$$p^*(\hat{y}|x) = \frac{1}{Z(x)} p^{PT}(\hat{y}|x) \exp(\frac{1}{\beta} RM(x,\hat{y}))$$

- Rearrange this via a log transformation

$$RM(x,\hat{y}) = \beta\,(\log p^*(\hat{y}|x) - \log p^{PT}(\hat{y}|x)) + \beta \log Z(x) = \beta \log \frac{p^*(\hat{y}|x)}{p^{PT}(\hat{y}|x)} + \beta \log Z(x)$$

- This holds true for any arbitrary LMs, thus

$$RM_\theta(x,\hat{y}) = \beta \log \frac{p_\theta^{RL}(\hat{y}|x)}{p^{PT}(\hat{y}|x)} + \beta \log Z(x)$$

Source : cs224n

Putting it together for DPO

- Derived reward model: $RM_\theta(x, \hat{y}) = \beta \log \frac{p_\theta^{RL}(\hat{y}|x)}{p^{PT}(\hat{y}|x)} + \beta \log Z(x)$
- Final DPO loss via the Bradley-Terry model of human preferences:

$$J_{DPO}(\theta) = -\mathbb{E}_{(x,y_w,y_l)\sim D}[\log \sigma(RM_\theta(x, y_w) - RM_\theta(x, y_l))]$$

Log Z term cancels as the loss only measures differences in rewards

$$= -\mathbb{E}_{(x,y_w,y_l)\sim D}\left[\log \sigma(\beta \log \frac{p_\theta^{RL}(y_w|x)}{p^{PT}(y_w|x)} - \beta \log \frac{p_\theta^{RL}(y_l|x)}{p^{PT}(y_l|x)})\right]$$

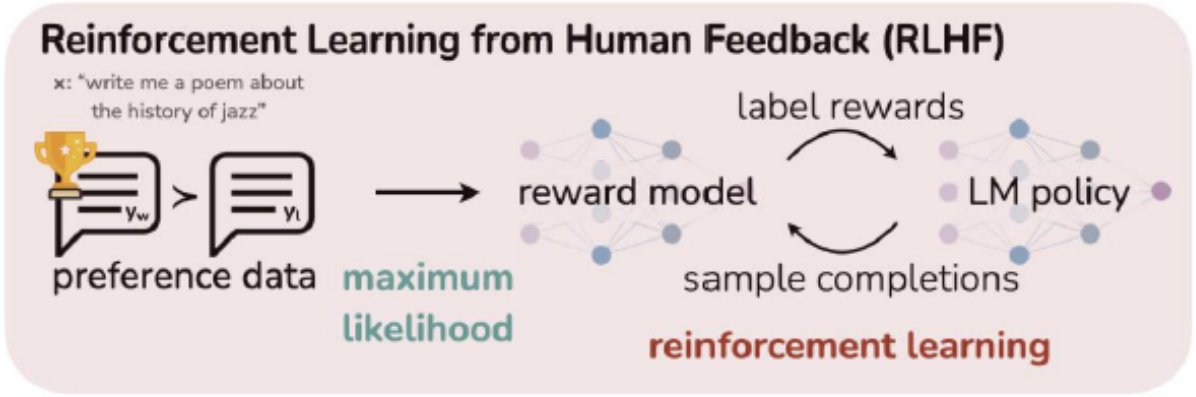
Reward for winning sample

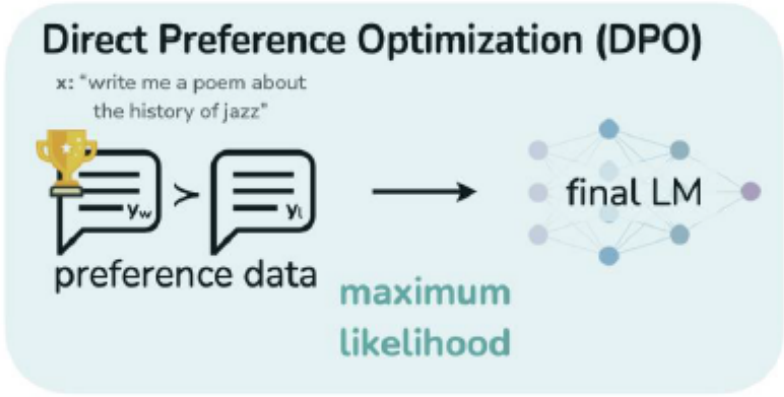
Reward for losing sample

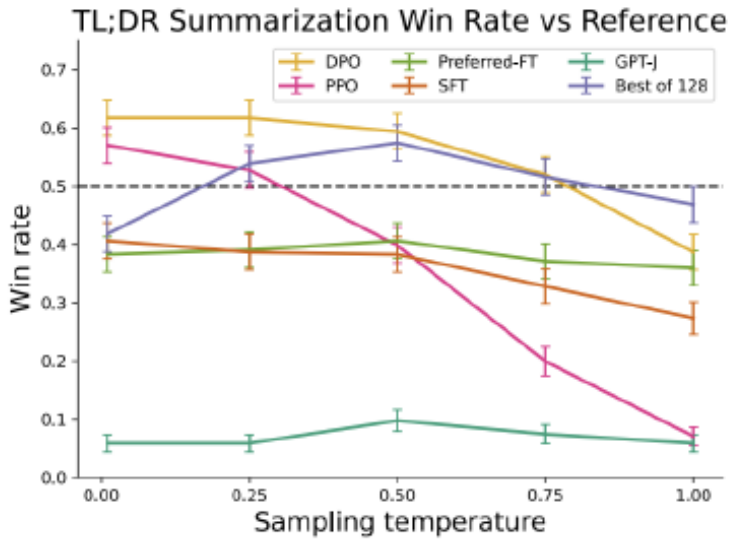
[Rafailov+ 2023]

- DPO derives a reward from likelihood ratios relative to a reference model and optimizes human preferences directly using a Bradley–Terry loss, eliminating the need for PPO while preserving KL-regularized alignment.

DPO outperforms prior methods







- You can replace the complex RL part with a very simple weighted MLE objective
- Other variants (KTO, IPO) now emerging too
- TL;DR summarization win rates vs. human-written summaries (GPT-4 as a judge)

Source : cs224n

Open source RLHF is now mostly (not RL)

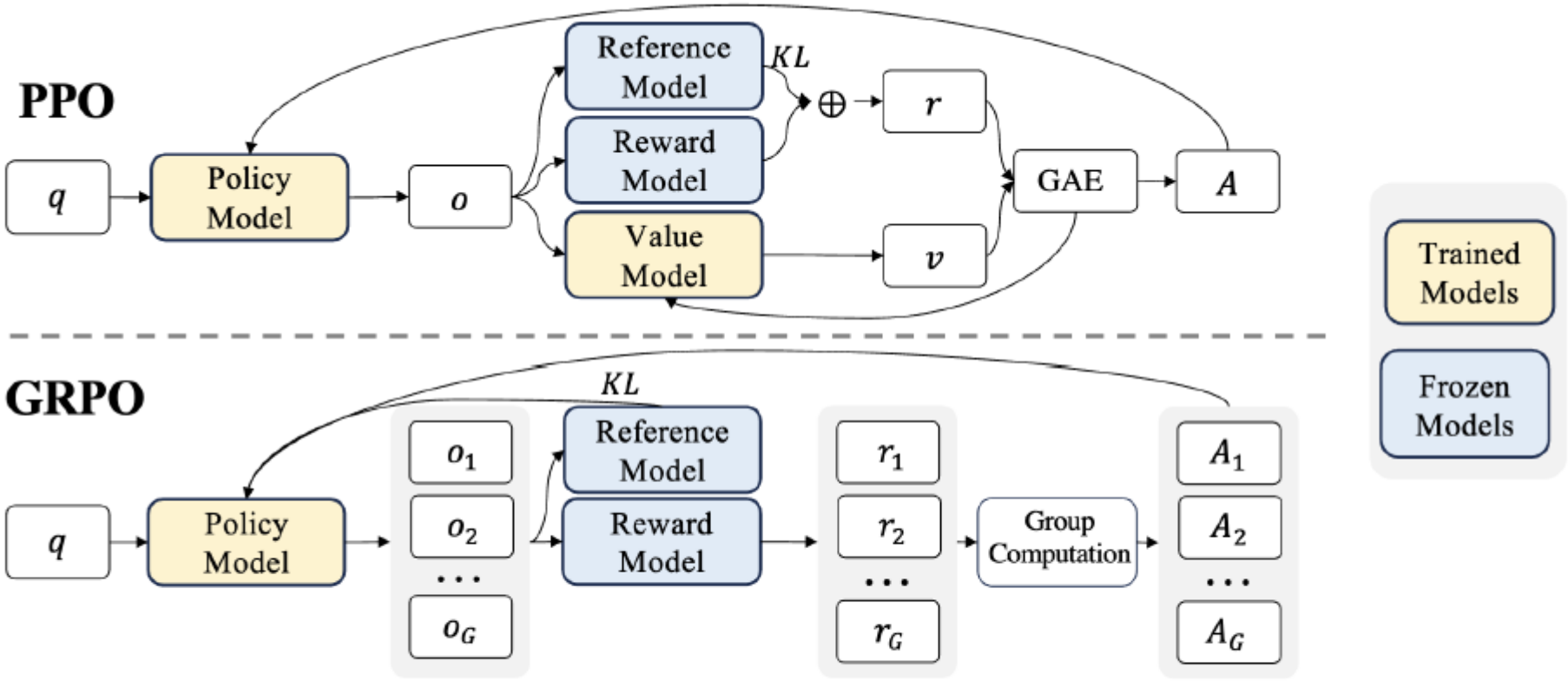
T	Model	Average	ARC	HellaSwag	MMLU	TruthfulQA	Winogrande	GSM8K
	udkai/Turdus	74.66	73.38	88.56	64.52	67.11	86.66	67.7
	fblgit/UNA-TheBeagle-7b-v1	73.87	73.04	88	63.48	69.85	82.16	66.72
	argilla/distilabeled-Marcoro14-7B-slerp	73.63	70.73	87.47	65.22	65.1	82.08	71.19
	mlabonne/NeuralMarcoro14-7B	73.57	71.42	87.59	64.84	65.64	81.22	70.74
	abideen/NexoNimbus-7B	73.5	70.82	87.86	64.69	62.43	84.85	70.36
	Neuronovo/neuronovo-7B-v0.2	73.44	73.04	88.32	65.15	71.02	80.66	62.47
	argilla/distilabeled-Marcoro14-7B-slerp-full	73.4	70.65	87.55	65.33	64.21	82	70.66
	CultriX/MistralTrix-v1	73.39	72.27	88.33	65.24	70.73	80.98	62.77
	ryandt/MusingCaterpillar	73.33	72.53	88.34	65.26	70.93	80.66	62.24
	Neuronovo/neuronovo-7B-v0.3	73.29	72.7	88.26	65.1	71.35	80.9	61.41
	CultriX/MistralTrixTest	73.17	72.53	88.4	65.22	70.77	81.37	60.73
	samir-fama/SamirGPT-v1	73.11	69.54	87.04	65.3	63.37	81.69	71.72
	SanjiWatsuki/Lelantos-DPO-7B	73.09	71.08	87.22	64	67.77	80.03	68.46

DPO
DPO (& UNA)
DPO
DPO
Merge (of DPO models)
DPO
DPO
DPO
DPO
DPO
No info but prob DPO, given
Merge (incl. DPO)
DPO

- DPO is used more in LLMs because it directly optimizes human preferences with the same quality as PPO-based RLHF, while being far simpler, cheaper, and more stable to train.

Source : cs224n

Improving the “RL” from RLHF ---GRPO



Shao, et al., "Deepseekmath: Pushing the limits of mathematical reasoning in open language models." arXiv:2402.03300 (2024).

Source : cs224n

Where does the labels come from?

Exclusive: OpenAI Used Kenyan Workers on Less Than $2 Per Hour to Make ChatGPT Less Toxic

Millions of Workers Are Training AI Models for Pennies

From the Philippines to Colombia, low-paid workers label training data for AI models used by the likes of Amazon, Facebook, Google, and Microsoft.

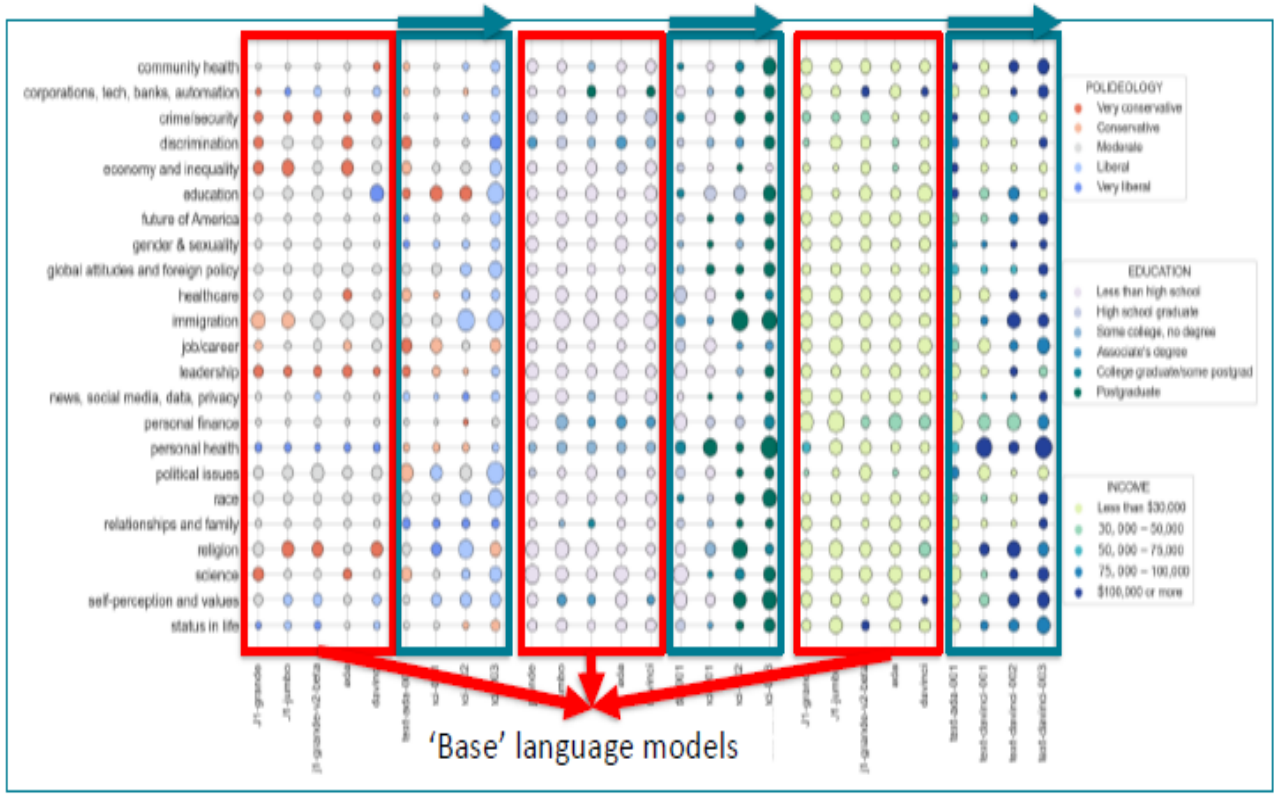
Behind the AI boom, an army of overseas workers in 'digital sweatshops'

- RLHF labels are often obtained from overseas, low-wage workers

Source : cs224n

Where does the labels come from?

What gender do you identify as?	
Male	50.0%
Female	44.4%
Nonbinary / other	5.6%
What ethnicities do you identify as?	
White / Caucasian	31.6%
Southeast Asian	52.6%
Indigenous / Native American / Alaskan Native	0.0%
East Asian	5.3%
Middle Eastern	0.0%
Latinx	15.8%
Black / of African descent	10.5%
What is your nationality?	
Filipino	22%
Bangladeshi	22%
American	17%
Albanian	5%
Brazilian	5%
Canadian	5%
Colombian	5%
Indian	5%
Uruguayan	5%
Zimbabwean	5%
What is your age?	
18-24	26.3%
25-34	47.4%
35-44	10.5%
45-54	10.5%
55-64	5.3%
65+	0%
What is your highest attained level of education?	
Less than high school degree	0%
High school degree	10.5%
Undergraduate degree	52.6%
Master's degree	36.8%
Doctorate degree	0%



[Santurkar+ 2023, OpinionQA]

- The leftmost table shows demographic statistics of the human annotators used in the dataset (from OpinionQA). They tend to be:
  - Western
  - Highly educated
  - From specific age ranges
  - From specific cultural and political backgrounds
- The annotators are not representative of the global population. Preference-based alignment amplifies bias
- Preference-based alignment amplifies the demographic and ideological biases of annotators, meaning that "human preferences" learned by LLMs reflect specific populations rather than universal values.

- We also need to be quite careful about how annotator biases might creep into LMs

Source : cs224n