04.12.23

OPENCAMPUS.sh

Deep Dive into LLMs

FINE-TUNING, PART II AND MODEL INFERENCE

- Quiz
- Parameter Efficient Fine-Tuning
- Model Comparison
- Applying RL on LLMs
- Model Inference
- Milestone Review:
  Model Evaluation
- Next Milestone:
  Model Definition

QUIZ

https://forms.office.com/e/Qzh4dQNmMC

Du

If I provide a model with an input prompt that looks like an oupt of a reward hacking, like repeating the same word over and over. What might be the consequence?

ChatGPT

Providing a Large Language Model (LLM) with an input prompt that mimics the output of reward hacking, such as repeating the same word or phrase multiple times, can lead to several possible outcomes:

1. **Confusion or Erratic Responses**: The LLM might interpret the repetitive prompt as nonsensical or confusing, leading to responses that are equally nonsensical, unrelated, or erratically structured. This is because the model might struggle to find a coherent context or intent in the input.
2. **Mirroring the Input**: Some models might simply mirror the repetitive nature of the input in their output. If the model has been trained to closely follow the style or structure of the input, it might generate a response that also contains repetitive elements.
3. **Standard or Generic Responses**: The LLM might default to a standard or generic response, especially if it's been trained to handle nonsensical or unclear inputs in a specific way. This could include asking for clarification or ignoring the repetitive nature of the prompt and providing a general response.
4. **Triggering Safety Mechanisms**: If the repetitive nature of the prompt is flagged by the model as potential spam, nonsensical input, or as trying to exploit the model in some way, it might trigger safety mechanisms. These mechanisms could involve refusing to generate a response, providing a warning, or defaulting to a safe, generic reply.
5. **Context-Dependent Responses**: The response might also depend on the context provided before the repetitive sequence. If the prompt includes coherent information before the repetition, the LLM might focus on that part for generating its response, potentially ignoring the repetitive part.

In all cases, the model's response will be influenced by its training, the specific algorithms it uses for parsing and generating text, and any safety or content

Nachricht ChatGPT...

DISCUSSION QUESTION (1)

- **What is Quantization?**

DISCUSSION QUESTION (2)

You want to build an app for translating English to German.

- What is relevant to know for the model to deliver an optimal translation?
- How would you construct the training dataset to fine-tune the LLM?

PARAMETER EFFICIENT FINE-TUNING (PEFT)

CODE EXAMPLE

```
[25]: peft_model = get_peft_model(original_model,
                                lora_config)
      print(print_number_of_trainable_model_parameters(peft_model))

      trainable model parameters: 3538944
      all model parameters: 251116800
      percentage of trainable model parameters: 1.41%
```

3.2 - Train PEFT Adapter

Define training arguments and create `Trainer` instance.

```
[26]: output_dir = f'./peft-dialogue-summary-training-{str(int(time.time()))}'

      peft_training_args = TrainingArguments(
          output_dir=output_dir,
          auto_find_batch_size=True,
          learning_rate=1e-3, # Higher learning rate than full fine-tuning.
          num_train_epochs=1,
          logging_steps=1,
          max_steps=1
      )

      peft_trainer = Trainer(
          model=peft_model,
          args=peft_training_args,
          train_dataset=tokenized_datasets["train"],
      )
```

MODEL COMPARISON

open-llm-leaderboard's Collections

The Big Benchmarks Collection

LLM Leaderboard best models ❤️🔥

The Big Benchmarks Collection

updated Oct 18

Gathering benchmark spaces on the hub (beyond the Open LLM Leaderboard)

Upvote 16 +12

Share collection

View history

Collection guide

Browse collections

Running on CPU UPGRADE ♡ 6.63k

🏆 Open LLM Leaderboard

Note 📐 The 🤗 Open LLM Leaderboard aims to track, rank and evaluate open LLMs and chatbots.
🤗 Submit a model for automated evaluation on the 🤗 GPU cluster on the “Submit” page!

Running on CPU UPGRADE ♡ 1.48k

🥇 MTEB Leaderboard

Note Massive Text Embedding Benchmark (MTEB) Leaderboard.

♡ 643

🏆🤖 Chatbot Arena Leaderboard

Note 🏆 This leaderboard is based on the following three benchmarks:
Chatbot Arena - a crowdsourced, randomized battle platform. We use 70K+ user votes to compute Elo ratings.
MT-Bench - a set of challenging multi-turn questions. We use GPT-4 to grade the model responses.
MMLU (5-shot) - a test to measure a model’s multitask accuracy on 57 tasks.

♡ 205

🏆🏋️ LLM-Perf Leaderboard

Note The 🤗 LLM-Perf Leaderboard 🏋️ aims to benchmark the performance (latency, throughput & memory) of Large Language Models (LLMs) with different hardwares, backends and optimizations using Optimum-Benchmark and Optimum flavors.
Anyone from the community can request a model or a hardware/backend/optimization configuration for automated benchmarking:

Leaderboard

| Vote | Blog | GitHub | Paper | Dataset | Twitter | Discord |

🏆 This leaderboard is based on the following three benchmarks.

- Chatbot Arena - a crowdsourced, randomized battle platform. We use 130K+ user votes to compute Elo ratings.
- MT-Bench - a set of challenging multi-turn questions. We use GPT-4 to grade the model responses.
- MMLU (5-shot) - a test to measure a model's multitask accuracy on 57 tasks.

💻 Code: The Arena Elo ratings are computed by this notebook. The MT-bench scores (single-answer grading on a scale of 10) are computed by fastchat.llm_judge. The MMLU scores are mostly computed by InstructEval. Higher values are better for all benchmarks. Empty cells mean not available. Last updated: November, 2023.

Model	⭐ Arena Elo rating	📈 MT-bench (score)	MMLU	License
GPT-4-Turbo	1217	9.32		Proprietary
GPT-4-0314	1201	8.96	86.4	Proprietary
Claude-1	1153	7.9	77	Proprietary
GPT-4-0613	1152	9.18		Proprietary
Claude-2.0	1127	8.06	78.5	Proprietary
Claude-2.1	1118	8.18		Proprietary
GPT-3.5-turbo-0613	1112	8.39		Proprietary
Claude-instant-1	1109	7.85	73.4	Proprietary
GPT-3.5-turbo-0314	1105	7.94	70	Proprietary
Tulu-2-DPO-70B	1105	7.89		AI2 ImpACT Low-risk
Yi-34B-chat	1102		73.5	Yi License
WizardLM-70b-v1.0	1097	7.71	63.7	Llama 2 Community

If you want to see more models, please help us add them.

More Statistics for Chatbot Arena

We added some additional figures to show more statistics. The code for generating them is also included in this notebook. Please note that you may see different orders from different ranking methods. This is

GAIA: A Benchmark for General AI Assistants

Grégoire Mialon[1], **Clémentine Fourrier**[2], **Craig Swift**[3], **Thomas Wolf**[2], **Yann LeCun**[1], **Thomas Scialom**[4]

[1]FAIR, Meta, [2]HuggingFace, [3]AutoGPT, [4]GenAI, Meta

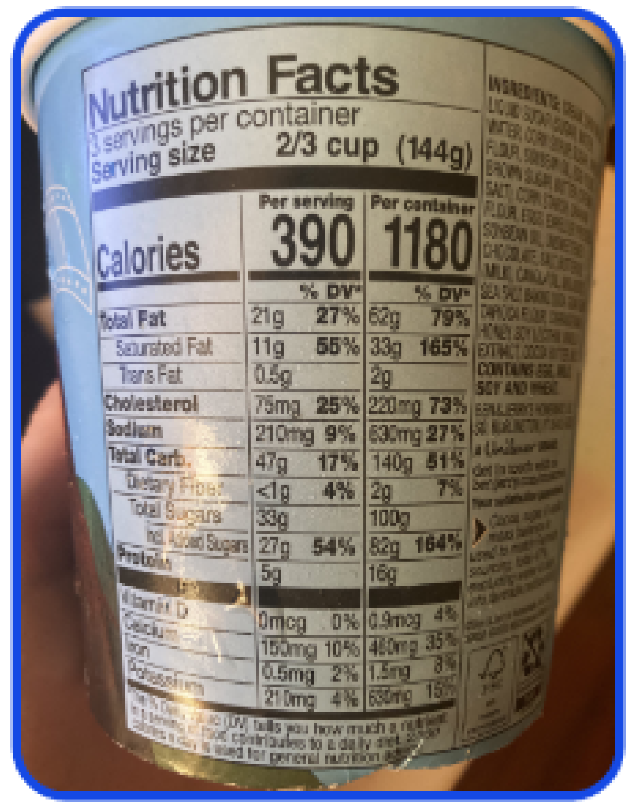
We introduce GAIA, a benchmark for General AI Assistants that, if solved, would represent a milestone in AI research. GAIA proposes real-world questions that require a set of fundamental abilities such as reasoning, multi-modality handling, web browsing, and generally tool-use proficiency. GAIA questions are conceptually simple for humans yet challenging for most advanced AIs: we show that human respondents obtain 92% vs. 15% for GPT-4 equipped with plugins. This notable performance disparity contrasts with the recent trend of LLMs outperforming humans on tasks requiring professional skills in *e.g.* law or chemistry. GAIA's philosophy departs from the current trend in AI benchmarks suggesting to target tasks that are ever more difficult for humans. We posit that the advent of Artificial General Intelligence (AGI) hinges on a system's capability to exhibit similar robustness as the average human does on such questions. Using GAIA's methodology, we devise 466 questions and their answer. We release our questions while retaining answers to 300 of them to power a leader-board hereby accessible.

Date: November 23, 2023
Correspondence: {gmialon,tscialom}@meta.com, clementine@huggingface.co
Code: https://huggingface.co/gaia-benchmark

Level 1

Question: What was the actual enrollment count of the clinical trial on H. pylori in acne vulgaris patients from Jan-May 2018 as listed on the NIH website?
Ground truth: 90

Level 2



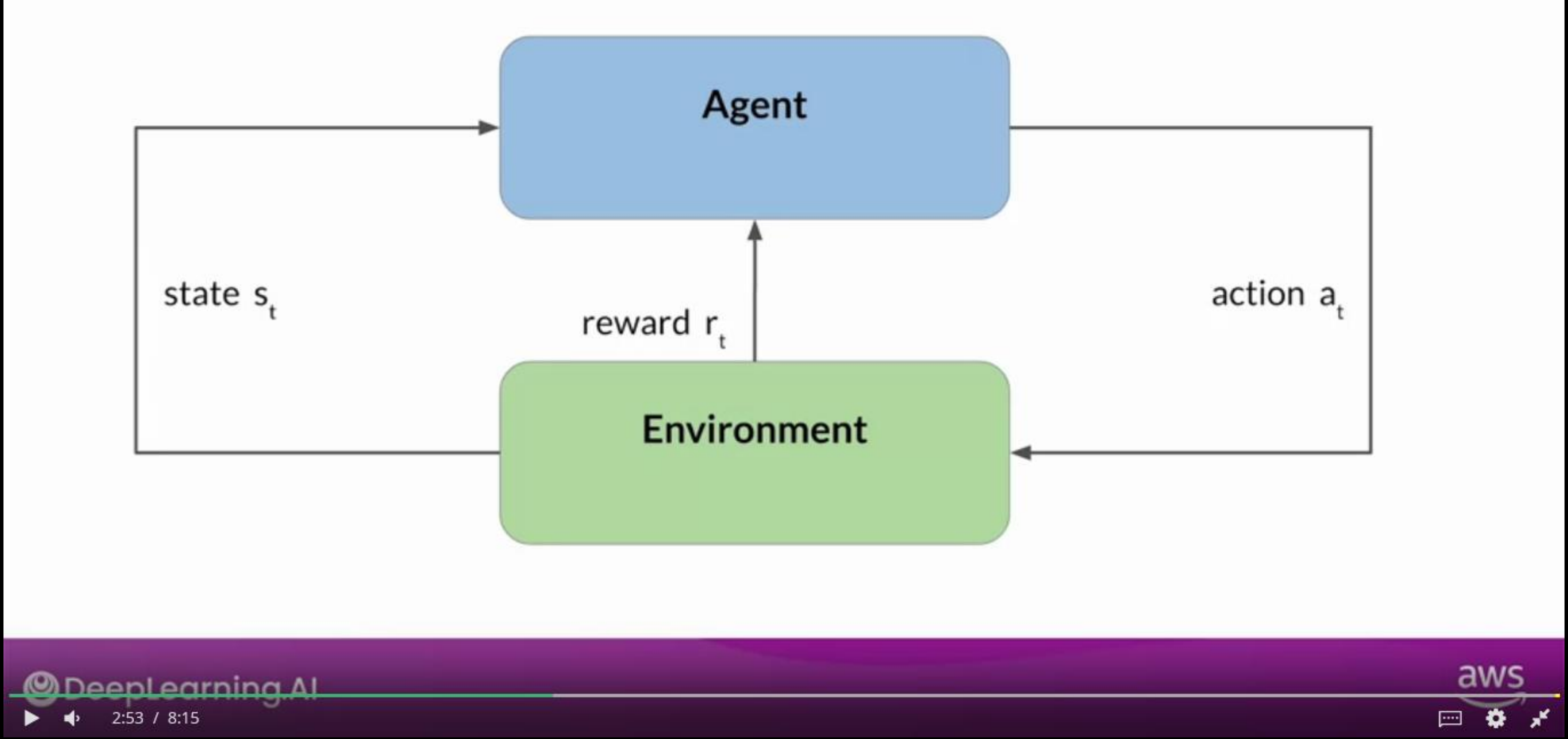
Question: If this whole pint is made up of ice cream, how many percent above or below the US federal standards for butterfat content is it when using the standards as reported by Wikipedia in 2020? Answer as + or - a number rounded to one decimal place.
Ground truth: +4.6

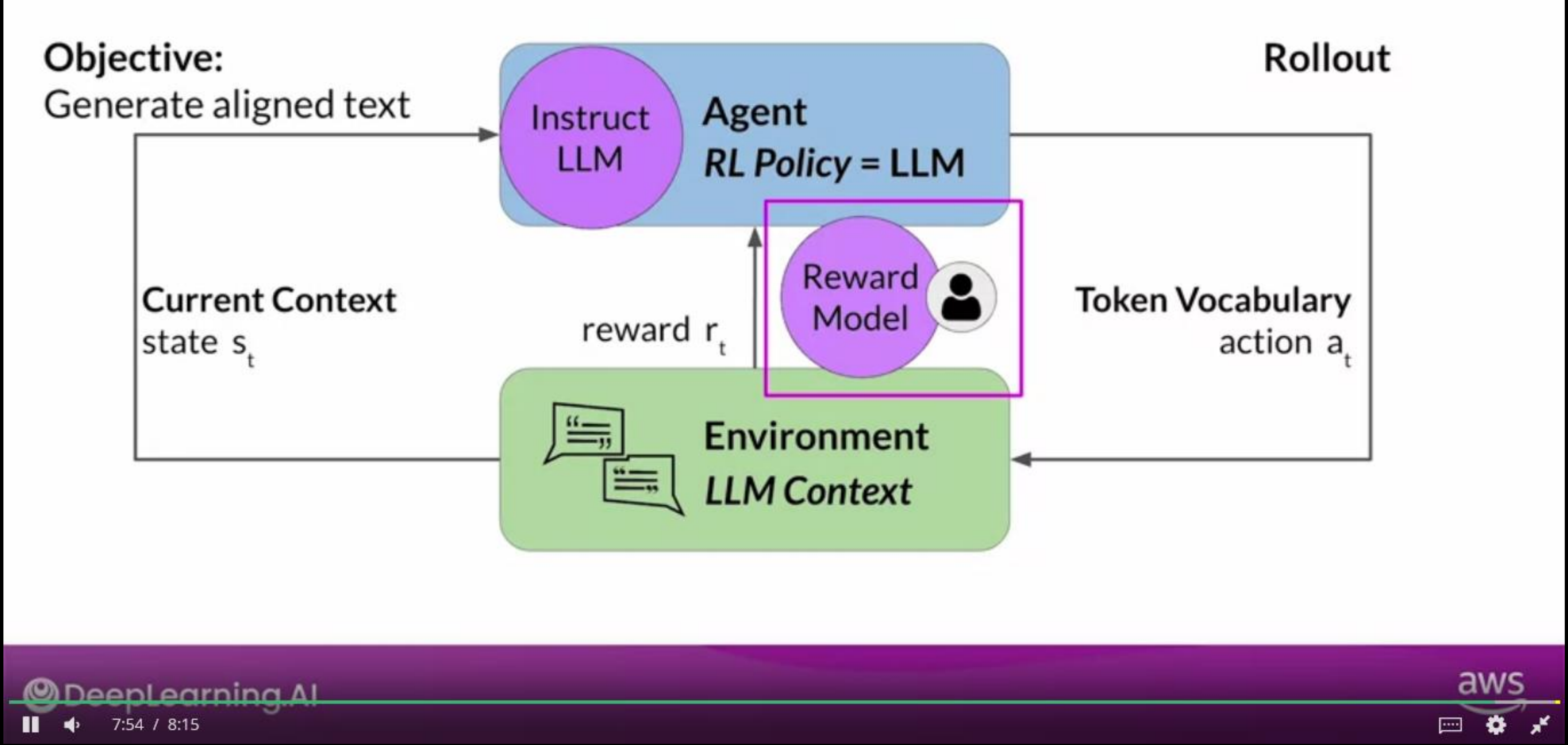
Level 3

Question: In NASA's Astronomy Picture of the Day on 2006 January 21, two astronauts are visible, with one appearing much smaller than the other. As of August 2023, out of the astronauts in the NASA Astronaut Group that the smaller astronaut was a member of, which one spent the least time in space, and how many minutes did he spend in space, rounded to the nearest minute? Exclude any astronauts who did not spend any time in space. Give the last name of the astronaut, separated from the number of minutes by a semicolon. Use commas as thousands separators in the number of minutes.
Ground truth: White; 5876

REINFORCEMENT LEARNING

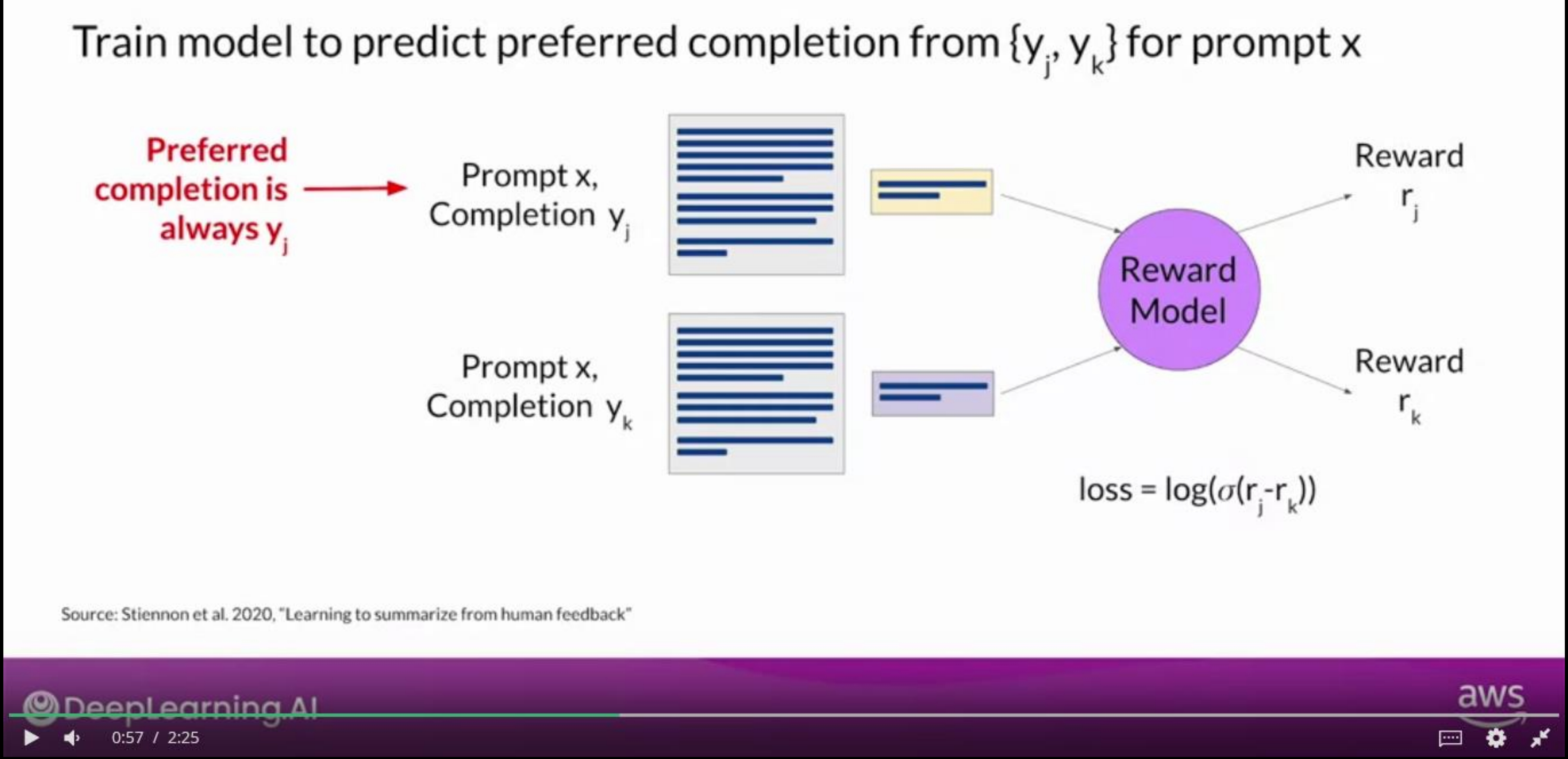
REINFORCEMENT LEARNING (RL)




Objective:
Generate aligned text
Rollout
Instruct LLM
Agent
RL Policy = LLM
Reward Model
Current Context
state s_t
reward r_t
Token Vocabulary
action a_t
Environment
LLM Context
DeepLearning.AI
aws
7:54 / 8:15

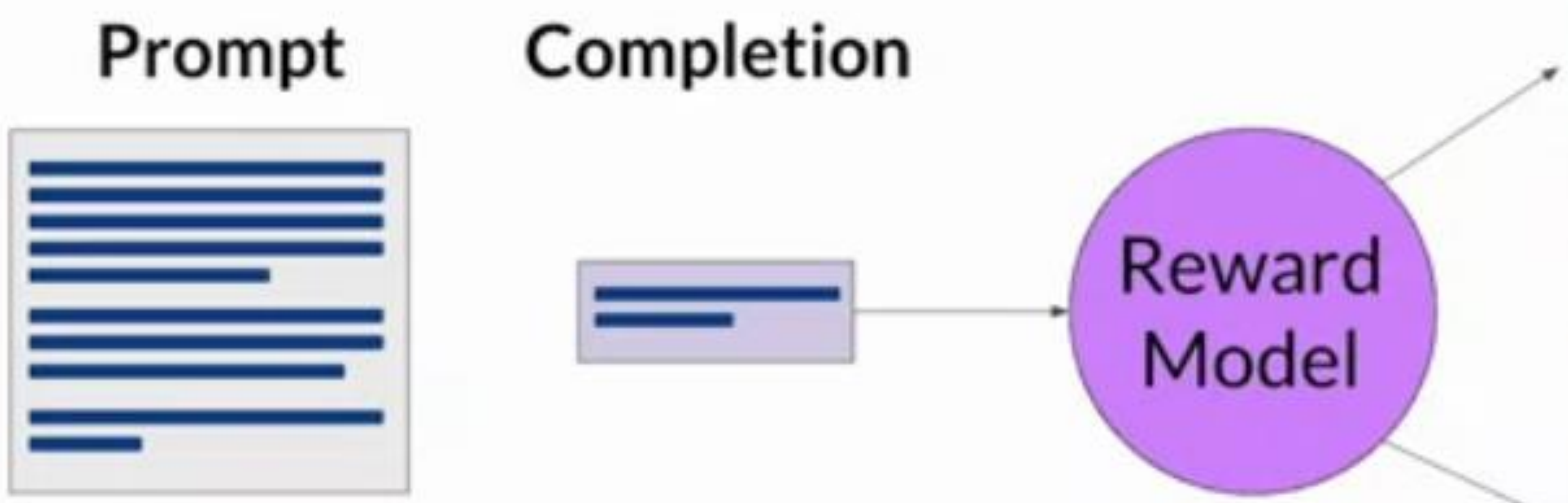
PROBLEMS WITH INSTRUCTIONS FOR HUMAN LABELERS

TRAINING THE REWARD MODEL



USING THE REWARD MODEL

Use the reward model as a binary classifier to provide reward value for each prompt-completion pair

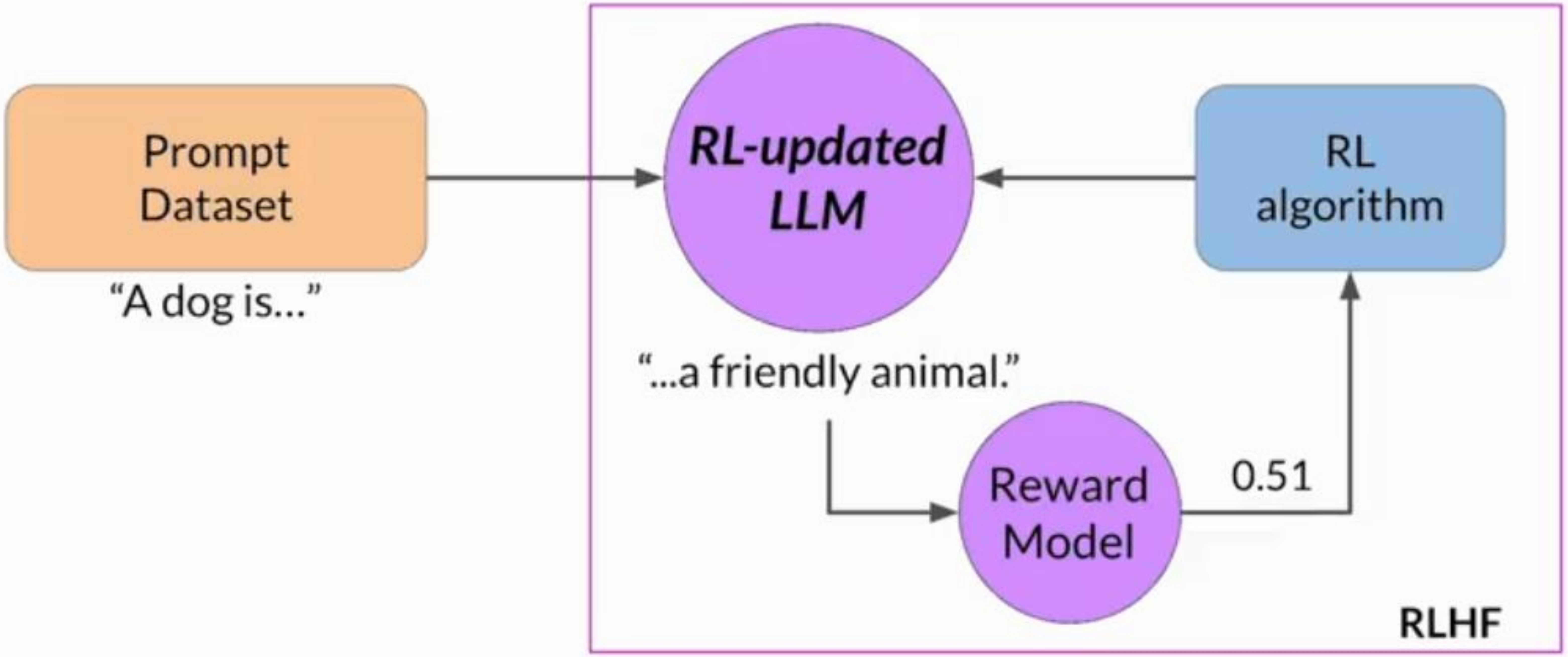


Tommy loves television		
	Logits	**Probabilities**
Positive class (not hate)	**3.171875**	0.996093
Negative class (**hate**)	-2.609375	0.003082

Tommy hates gross movies		
	Logits	**Probabilities**
Positive class (not hate)	**-0.535156**	0.337890
Negative class (**hate**)	0.137695	0.664062

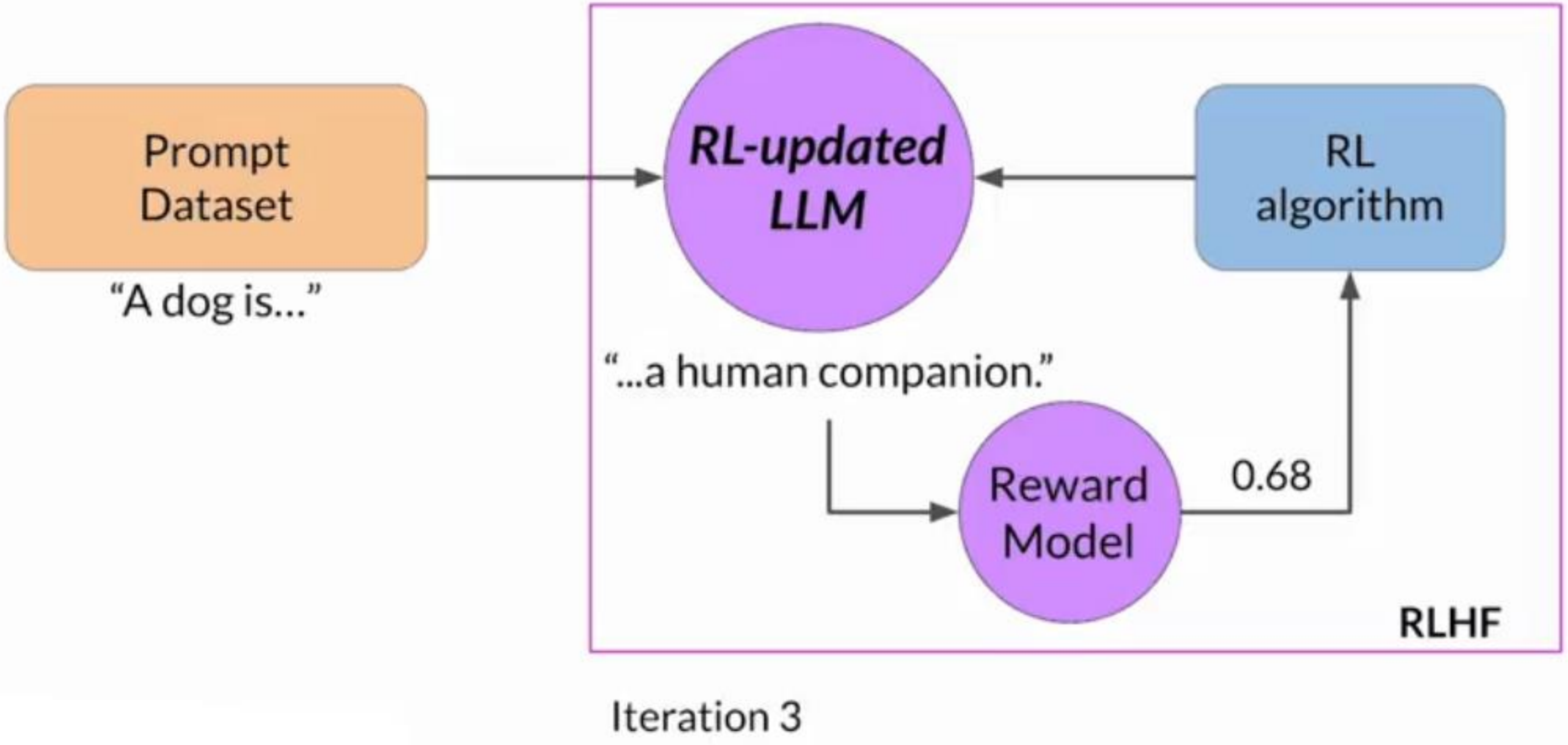
Source: Stiennon et al. 2020, "Learning to summarize from human feedback"

Use the reward model to fine-tune LLM with RL

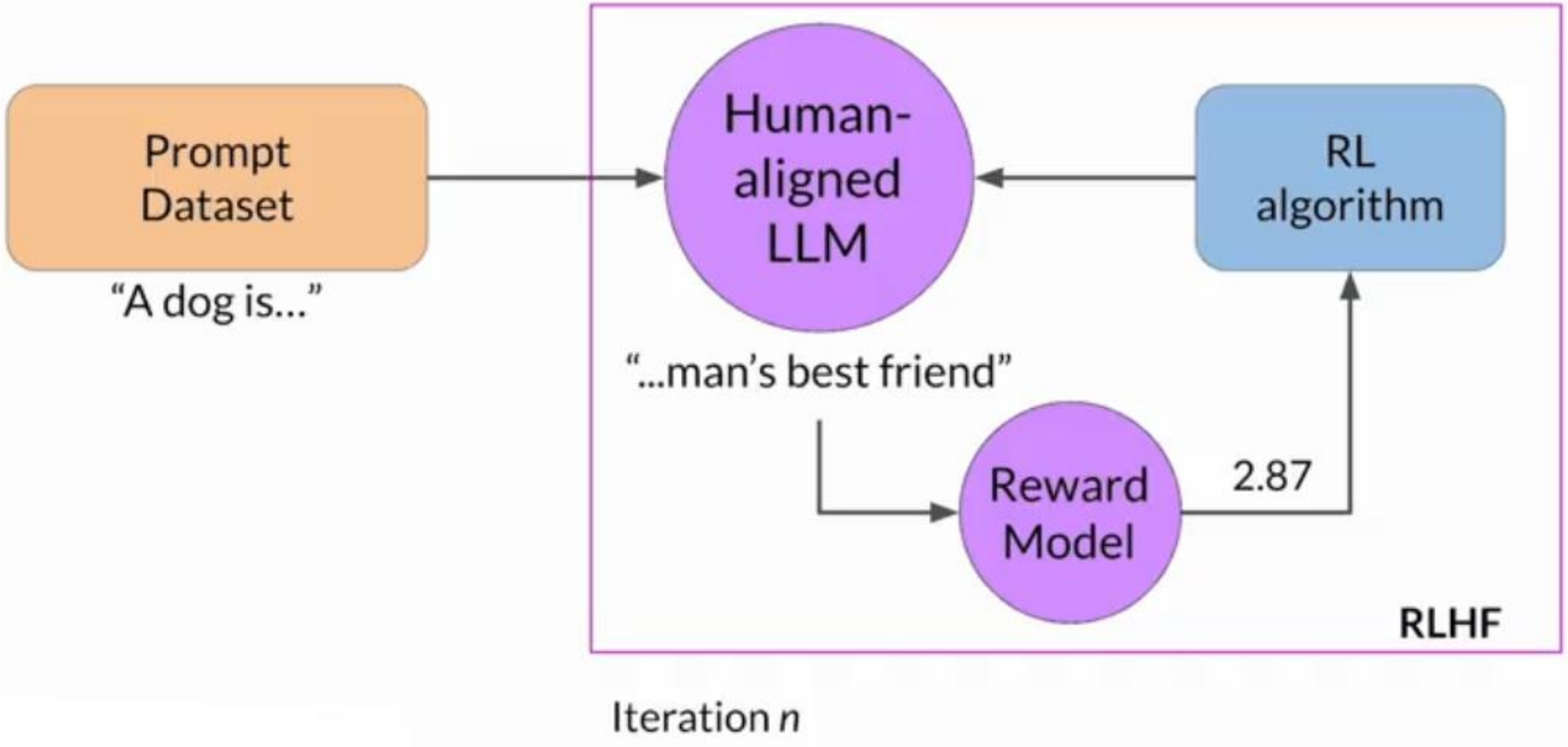


Iteration 2

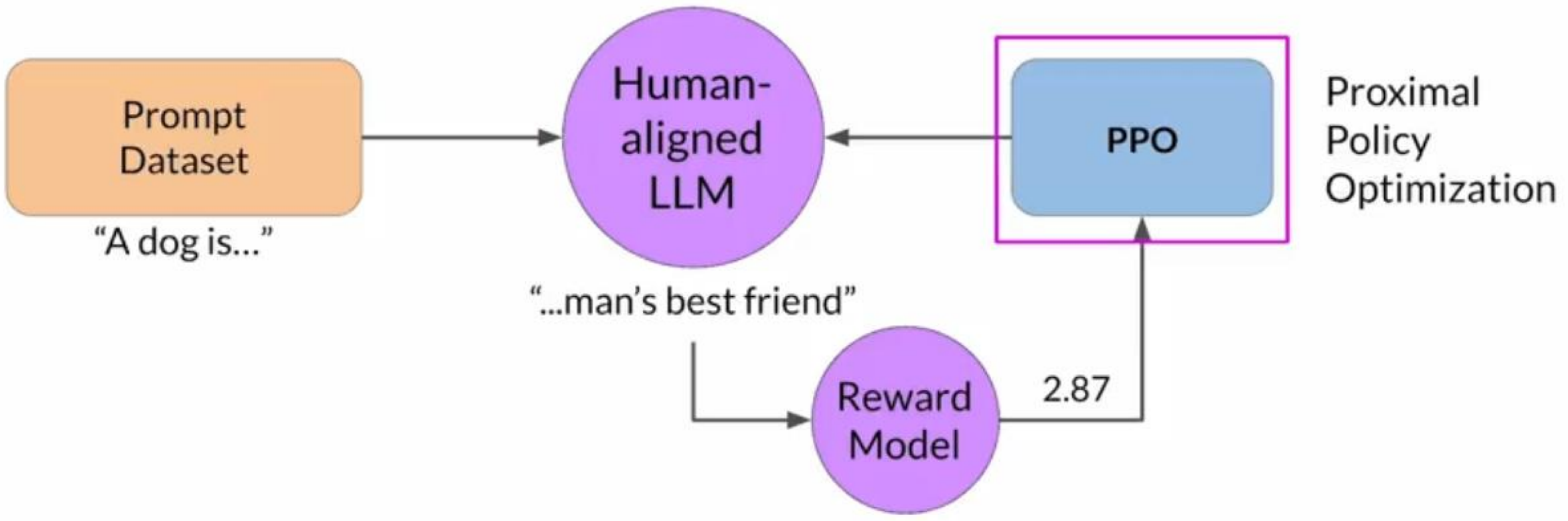
Use the reward model to fine-tune LLM with RL



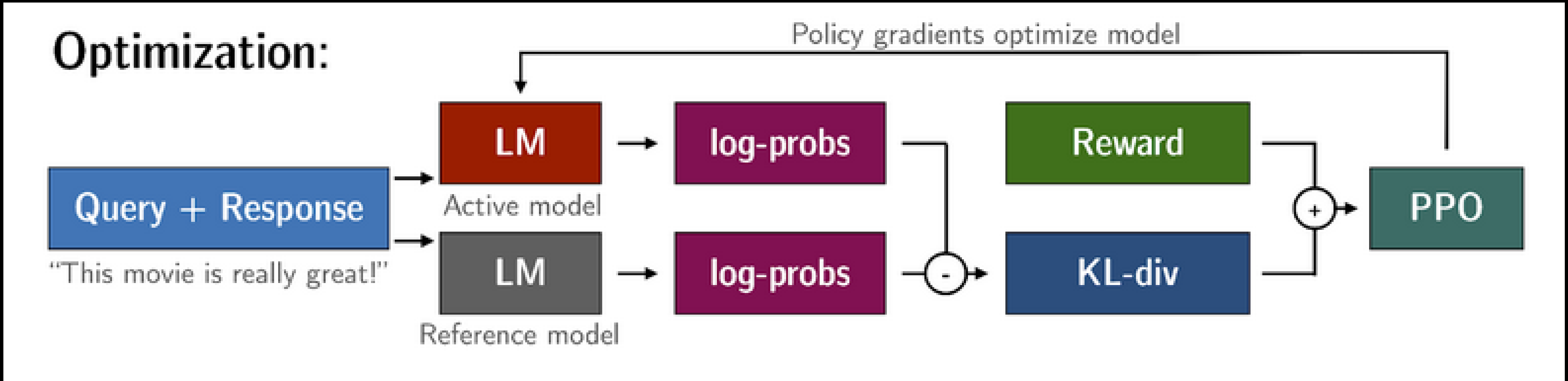
Use the reward model to fine-tune LLM with RL



Use the reward model to fine-tune LLM with RL

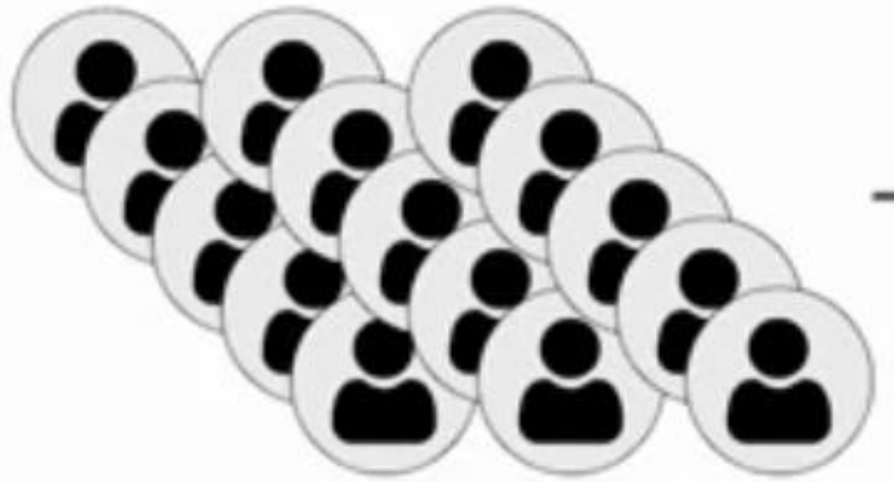


KULLBACK-LEIBLER DIVERGENCE



Scaling human feedback

Reinforcement Learning from Human Feedback



→ 10's of thousands of human-preference labels → **Reward Model**

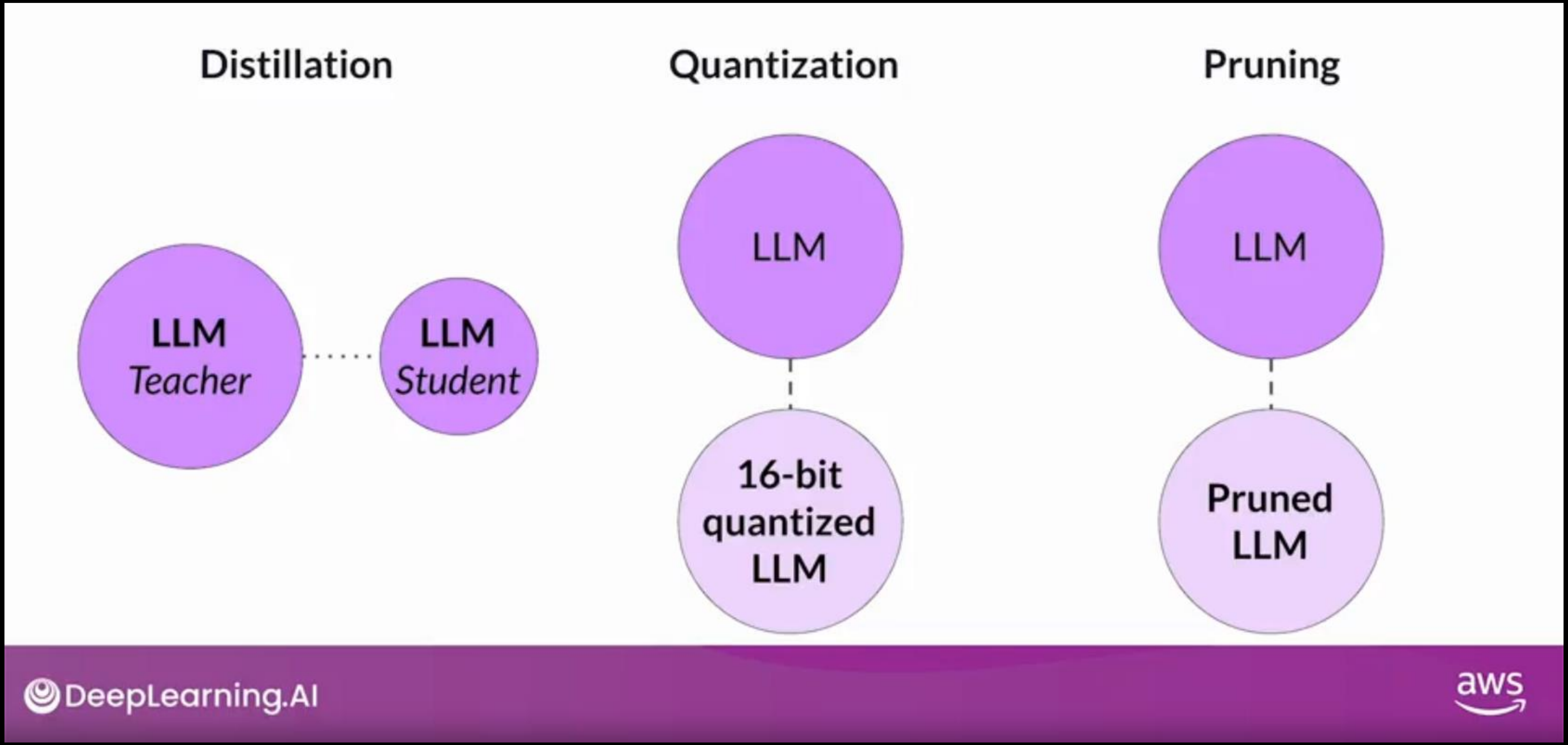
Model self-supervision: Constitutional AI

Human-aligned LLM 🤔

Rules
...
...
...

MODEL INFERENCE

OPTIMIZATION TECHNIQUES

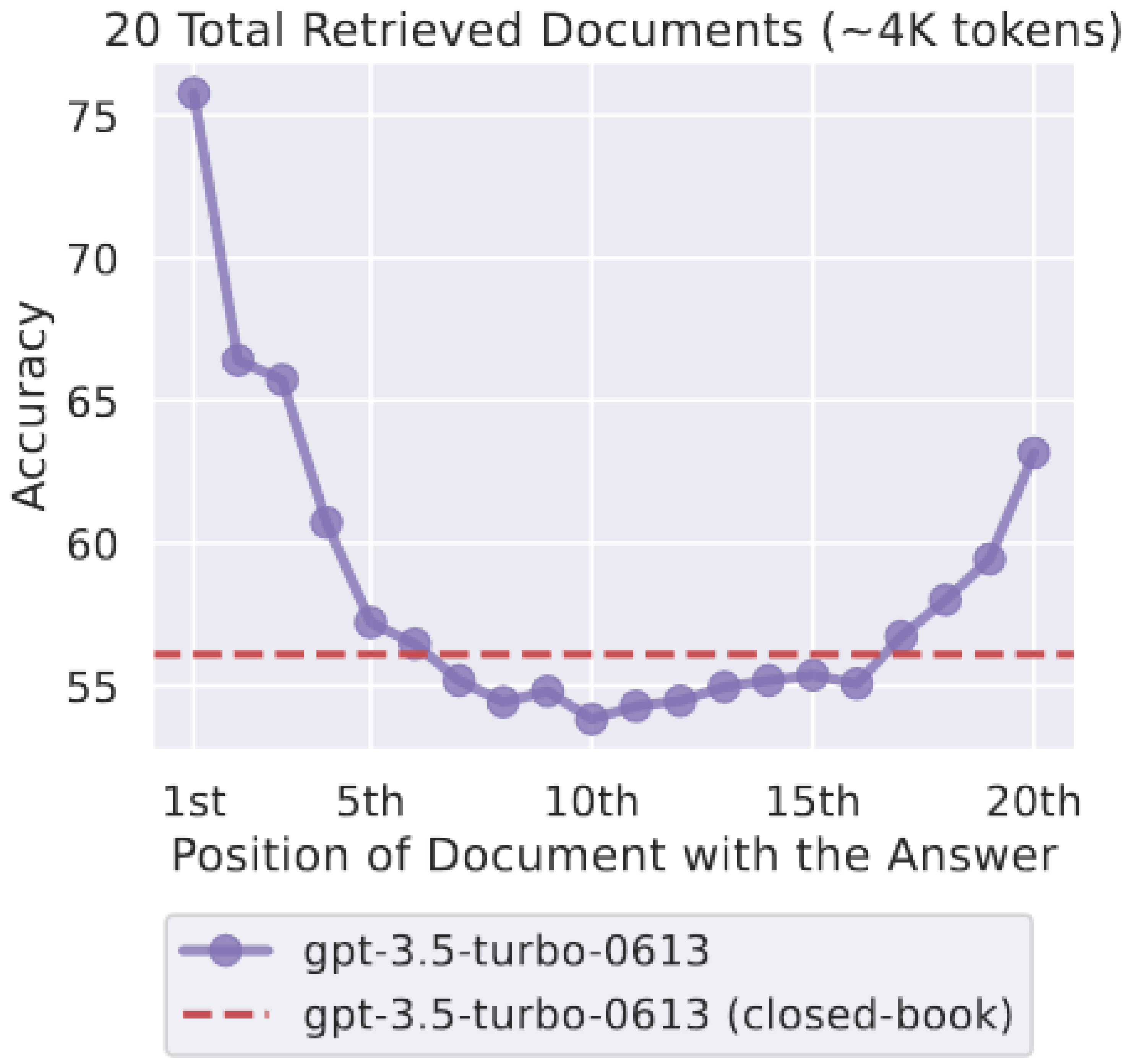


Lost in the Middle: How Language Models Use Long Contexts

Nelson F. Liu[1*] **Kevin Lin**[2] **John Hewitt**[1] **Ashwin Paranjape**[3]
Michele Bevilacqua[3] **Fabio Petroni**[3] **Percy Liang**[1]
[1]Stanford University [2]University of California, Berkeley [3]Samaya AI
nfliu@cs.stanford.edu

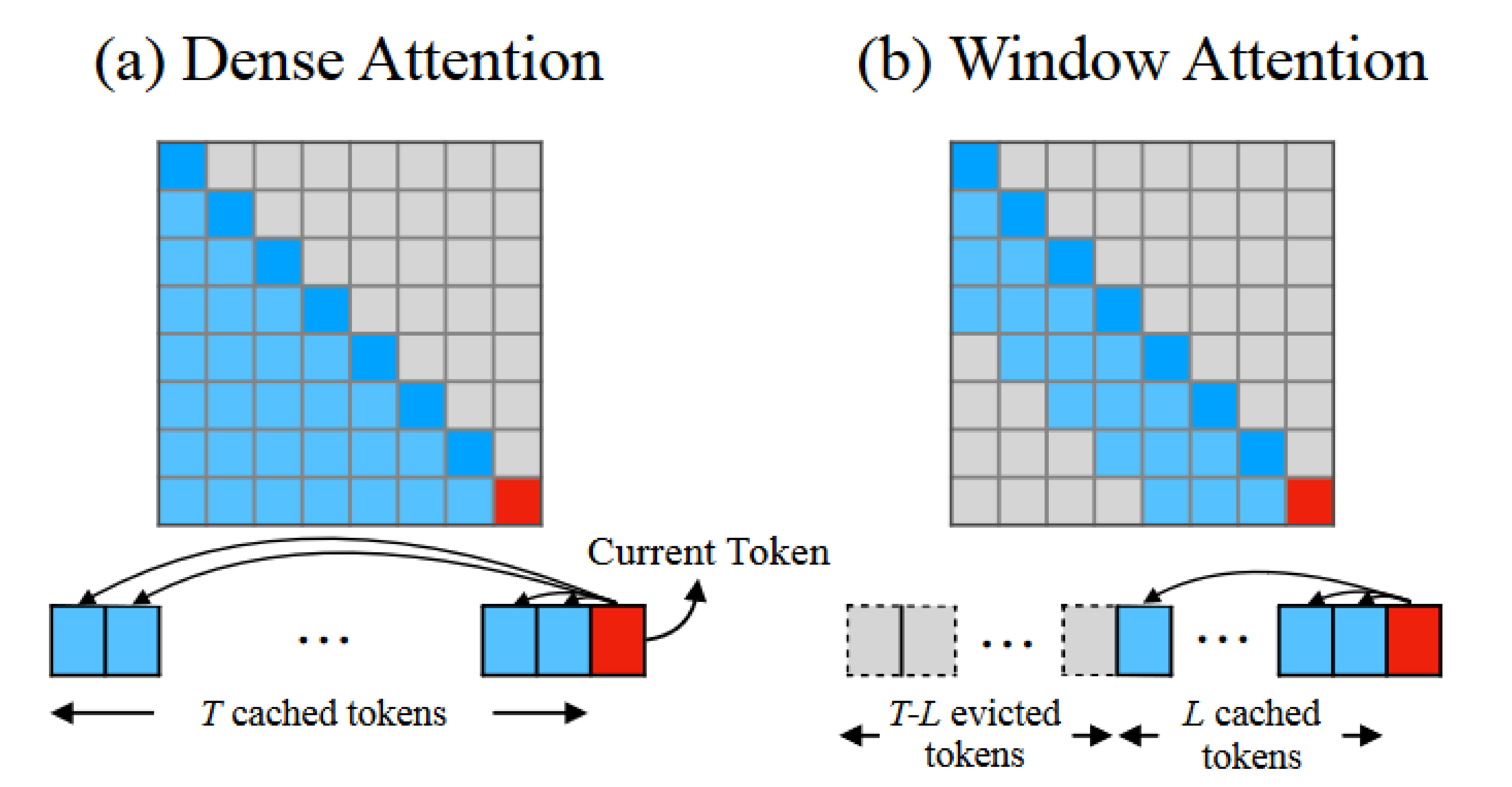
Abstract

While recent language models have the ability to take long contexts as input, relatively little is known about how well they *use* longer context. We analyze the performance of language models on two tasks that require identifying relevant information in their input contexts: multi-document question answering and key-value retrieval. We find that performance can degrade significantly when changing the position of relevant information, indicating that current language models do not robustly make use of information in long input contexts. In particular, we observe that performance is often highest when relevant information occurs at the beginning or



INFERENCE FOR STREAMING LLMS

DENSE ATTENTION VS. WINDOW ATTENTION



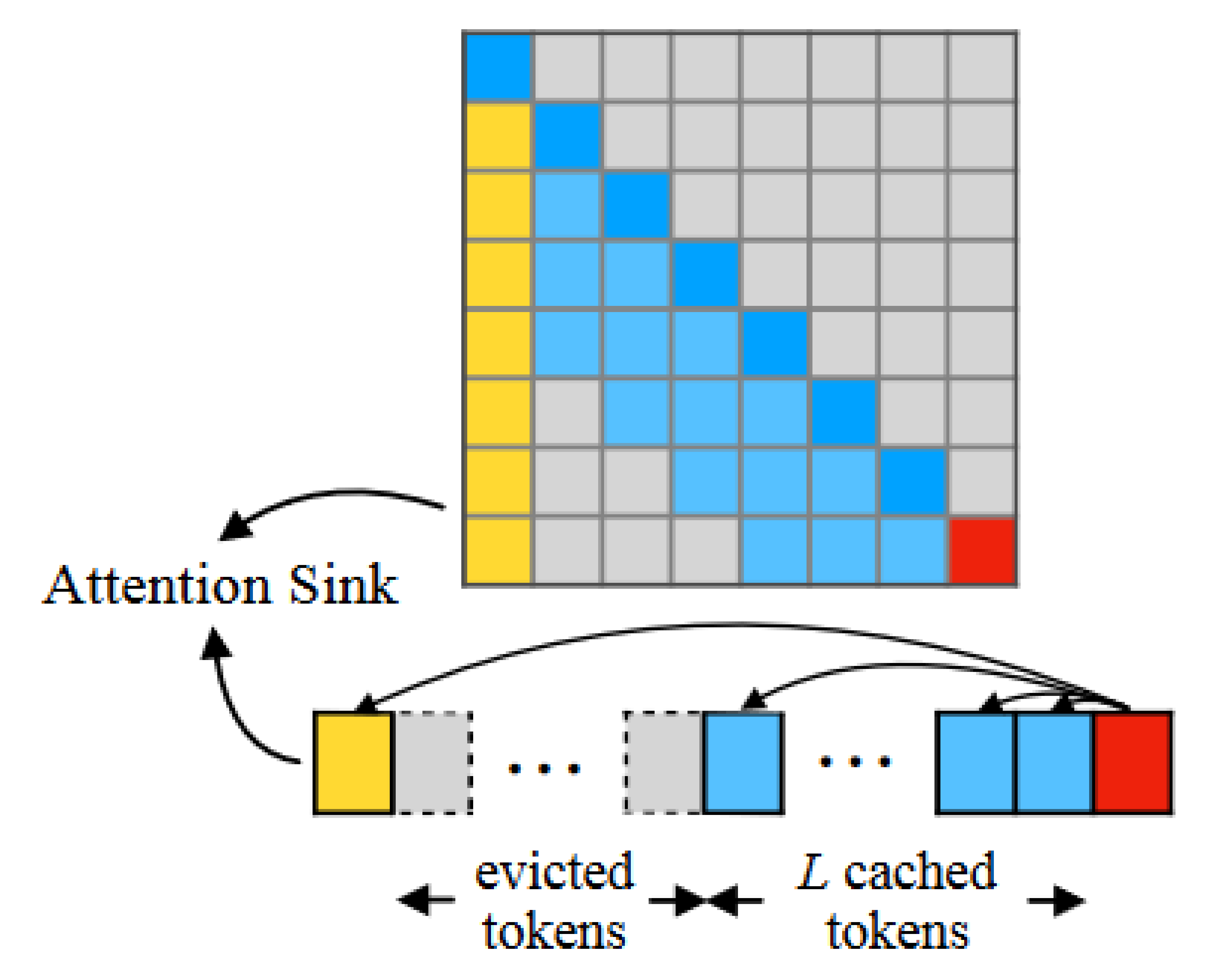
EFFICIENT STREAMING LANGUAGE MODELS WITH ATTENTION SINKS

Guangxuan Xiao[1]* **Yuandong Tian**[2] **Beidi Chen**[3] **Song Han**[1] **Mike Lewis**[2]

[1] Massachusetts Institute of Technology
[2] Meta AI
[3] Carnegie Mellon University
https://github.com/mit-han-lab/streaming-llm

ABSTRACT

Deploying Large Language Models (LLMs) in streaming applications such as multi-round dialogue, where long interactions are expected, is urgently needed but poses two major challenges. Firstly, during the decoding stage, caching previous tokens' Key and Value states (KV) consumes extensive memory. Secondly, popular LLMs cannot generalize to longer texts than the training sequence length. Window attention, where only the most recent KVs are cached, is a natural approach — but we show that it fails when the text length surpasses the cache size. We observe an interesting phenomenon, namely *attention sink*, that keeping the KV of initial tokens will largely recover the performance of window attention. In this paper, we first demonstrate that the emergence of *attention sink* is due to the strong attention scores towards initial tokens as a "sink" even if they are not semantically important. Based on the above analysis, we introduce StreamingLLM, an efficient framework that enables LLMs trained with a *finite length* attention window to generalize to *infinite sequence length* without any fine-tuning. We show that StreamingLLM can enable Llama-2, MPT, Falcon, and Pythia to perform stable and efficient language modeling with up to 4 million tokens and more. In addition, we discover that adding a placeholder token as a dedicated attention sink during pre-training can further improve streaming deployment. In streaming settings, StreamingLLM outperforms the sliding window recomputation baseline by up to 22.2× speedup. Code and datasets are provided in the link.


Attention Sink
evicted tokens
L cached tokens

MILESTONE REVIEW:
MODEL EVALUATION

NEXT MILESTONE

PROJECT MILESTONES

30.10.	Form Groups
06.11.	Literature Review I
13.11.	Literature Review II
20.11.	Dataset Characteristics I
27.11.	Dataset Characteristics II
04.12.	Baseline Model
11.12.	Model Evaluation
18.12.	Model Definition
08.01.	Project Presentations

MODEL DEFINITION

Start optimizing your final model.

TASKS UNTIL NEXT WEEK

- Focus on your project work