
LECTURE 5-1

Post-Training and Reasoning

From pretrained model to clinical assistant



Parisa Rashidi · Herbert Wertheim College of Engineering, University of Florida



Where We Are

Last week: what is inside the model. Today: how training turns it into an assistant.

Week 4

Inside modern LLMs

The block, the KV cache,
mixture of experts

Week 5: today

Post-training and reasoning

Fine-tuning, LoRA, preference
tuning, reasoning models

Week 6

Self-supervised learning and imaging

Foundation models for
biomedical data

A Pretrained Model Only Continues Text

Given the start of a discharge summary, it writes more note, about a different patient.

Discharge summary: The patient denies chest pain.



Qwen3-0.6B-Base continues (real output)

The patient is a 65-year-old woman with a history of hypertension and diabetes.
She has been experiencing fatigue and shortness of breath for the ...

Pretraining teaches one thing: continue the text.

Ask a Question, Get a Quiz

The base model treats a question as text to continue, and keeps inventing questions.

Note: The patient denies chest pain.

Question: Does the patient report chest pain?



Qwen3-0.6B-Base continues (real output)

(Yes/No)

Answer: No

Question: Does the patient report shortness of breath? (Yes/No)

Answer: Yes

Question: ...

Post-Trained, but Wrong

The post-trained model answers like an assistant. Here, its answer is wrong.

Note: The patient denies chest pain.

Question: Does the patient report chest pain?



Post-trained Qwen3-0.6B (real output)

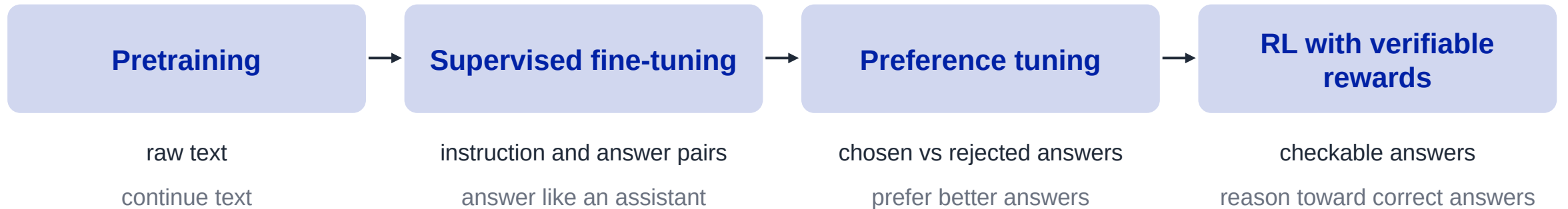
“Yes, the patient reports chest pain.”

**wrong: the note
says “denies”**

Today: three ways to fix this negation error,
and what each one costs.

Four Stages After the Architecture

Same model and same network; each stage changes what the model is trained on.



Sections 1 to 4 of today follow these stages:
fine-tuning · fine-tuning cheaply (LoRA) · preference tuning · reasoning



Supervised Fine-Tuning

Instruction and Answer Pairs in a Chat Template

Special markers show where the user's turn and the assistant's turn begin and end.

<|im_start|>user

role marker

Note: The patient denies chest pain. Question: Does the patient report chest pain?

instruction

<|im_end|> <|im_start|>assistant

role marker

No. The note says the patient denies chest pain.

answer

<|im_end|>

end marker

Same Loss, Counted Only on the Answer

The next-token loss is computed on the answer tokens; the instruction is context.

Note : The patient denies chest pain . Question : Does ... ?

instruction: not counted in the loss

No . The note says the patient denies chest pain .

answer: counted in the loss

loss = average over answer tokens of $-\log p(\text{next token} \mid \text{everything before it})$

Animation: From Pretrained Model to Clinical Assistant

Animation clip: week05-clip1-post-training.mp4

Silent version embedded (click to play). The narrated version is in the course repository.

Same Architecture, Same Loss, Different Data



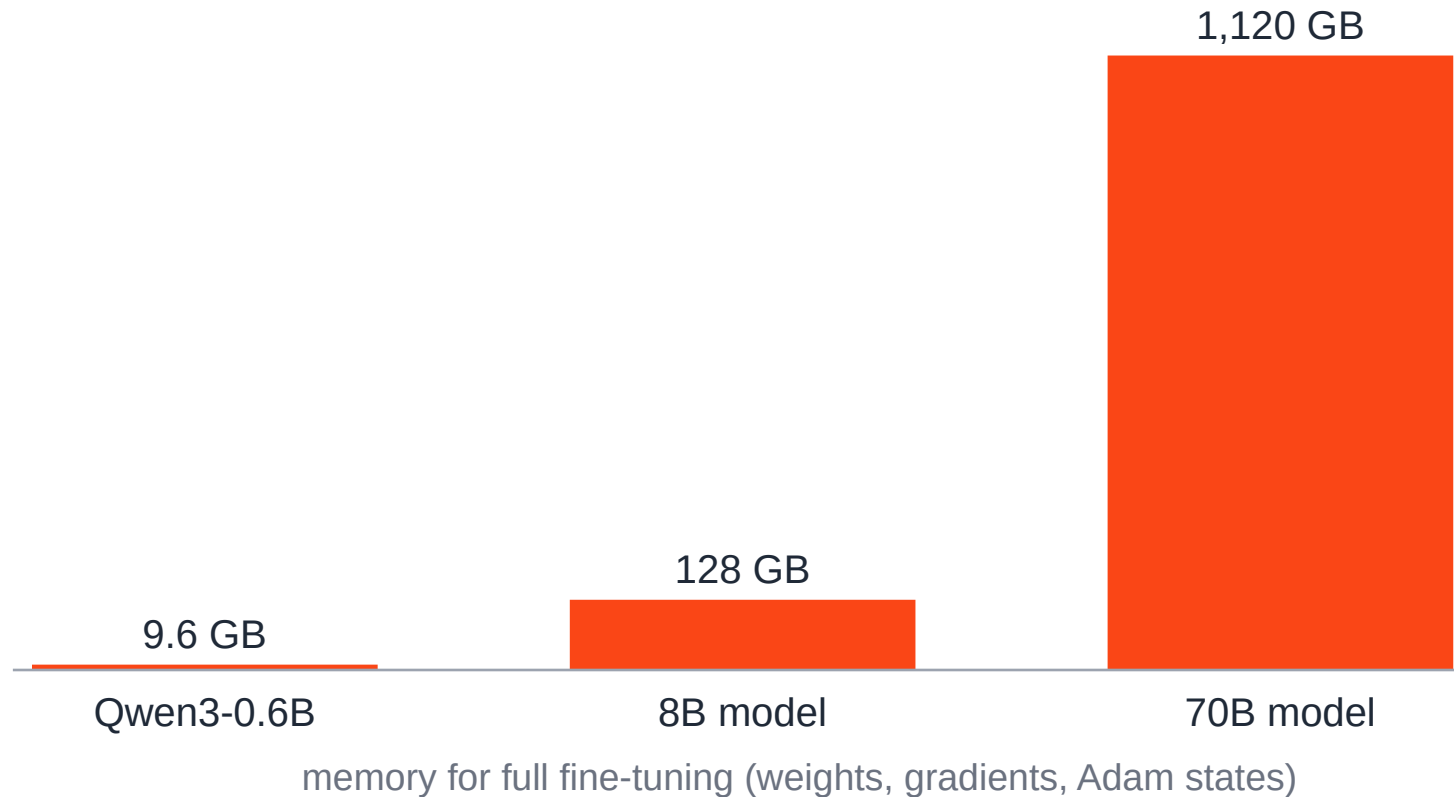


LoRA

What Full Fine-Tuning Costs

With the Adam optimizer, every parameter needs about 16 bytes of memory.

weight + gradient + 2 Adam states = 4 numbers × 4 bytes = 16 bytes per parameter

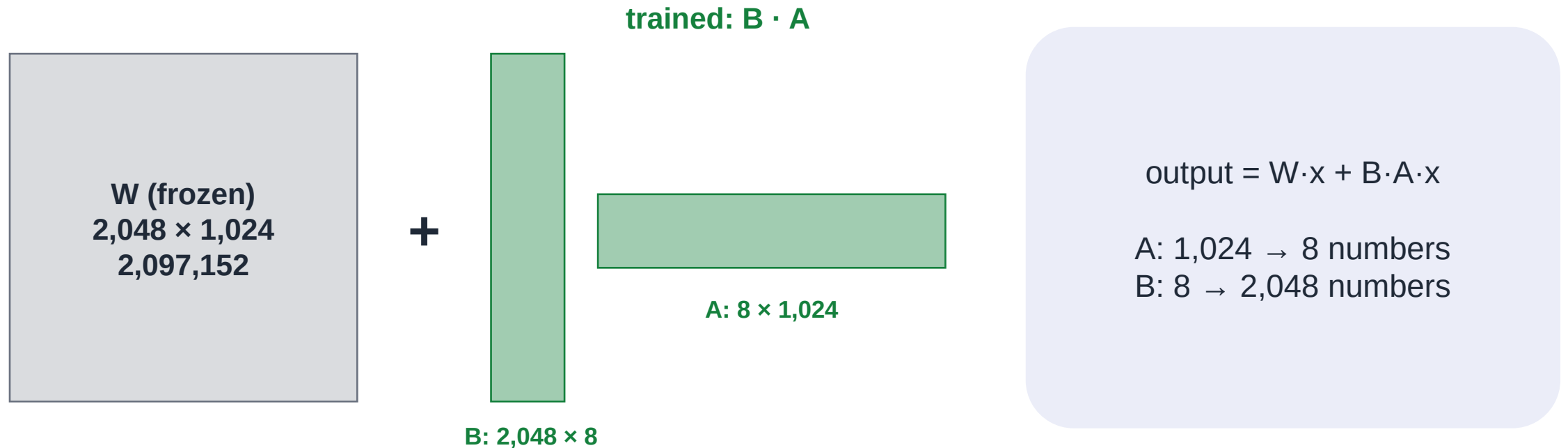


one GPU

80 GB is a large
data-center GPU

LoRA: Freeze W, Train Two Thin Matrices

The output becomes $W \cdot x + B \cdot A \cdot x$. Only B and A are trained.



The pretrained knowledge in W is untouched; $B \cdot A$ learns the task-specific change.

Counting the Weights

Rank-8 LoRA trains about 1.2% of one matrix and about 0.4% of the model.

W, one query projection

2,097,152

B + A for that matrix

24,576 (1.2%)

all q, k, v, o in 28 layers

2,293,760 ($\approx 0.4\%$ of the model)

adapter file (16-bit)

about 4.6 MB

B Starts at Zero

At the first step, LoRA changes nothing: training starts exactly from the original model.

initialization

A random, $B = 0$

first step

$B \cdot A \cdot x = 0$, so output = $W \cdot x$

training

only B and A change; W stays frozen

Animation: LoRA

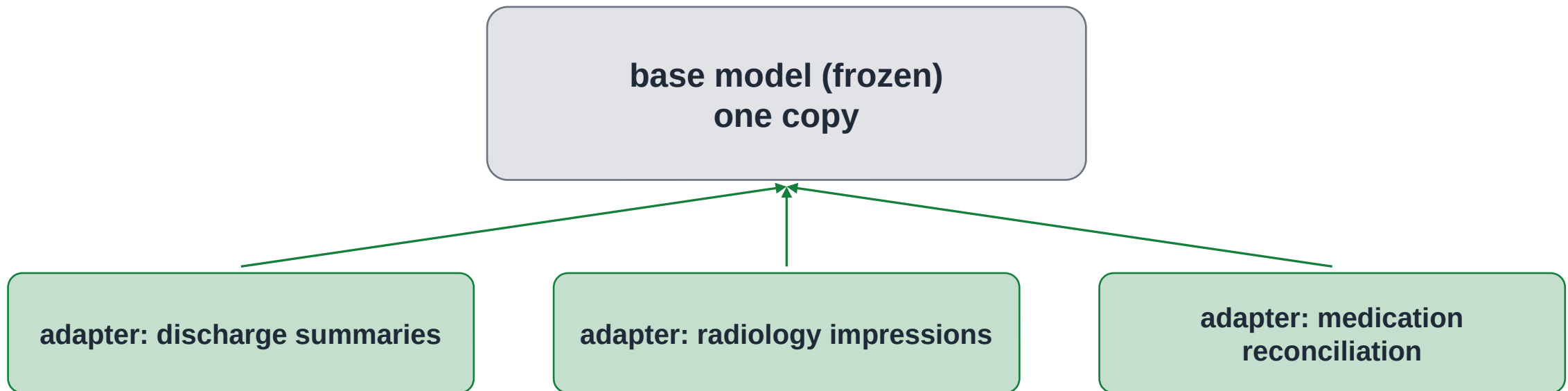
Animation clip: week05-clip2-lora.mp4

Silent version embedded (click to play). The narrated version is in the course repository.

One Base Model, Many Clinical Adapters

Swap a 4.6 MB adapter instead of copying a whole model for every task.

Notebook: a 12-second LoRA run on CPU fixes the chest-pain negation error.





Preference Tuning

Preference Data: Chosen and Rejected

A clinician marks one answer as better. The model currently prefers the wrong one.

Note: The patient denies chest pain.

Question: Does the patient report chest pain?

chosen

“No. The note says the patient denies chest pain.”

log-probability now: -21.37

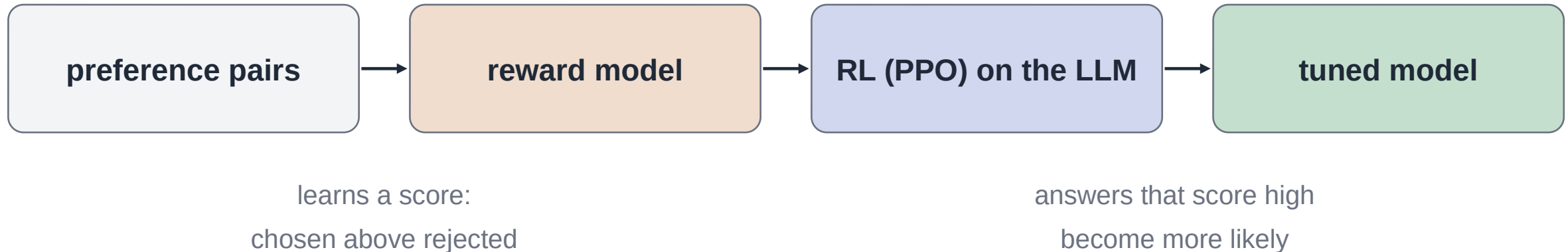
rejected

“Yes, the patient reports chest pain.”

log-probability now: -1.12

RLHF: A Reward Model, Then Reinforcement Learning

The classic method: learn a reward from preferences, then optimize it.



cost: a separate reward model, and an RL step that can be unstable

DPO: Skip the Reward Model

Compare the model in training with a frozen copy of itself.

$$\text{loss} = -\log \sigma(\beta \cdot [\text{chosen gain} - \text{rejected gain}])$$

gain

$\log p(\text{answer} \mid \text{model in training}) - \log p(\text{answer} \mid \text{frozen copy})$

β

a small number (here 0.1): how far the model may move from the frozen copy

σ

the sigmoid, which maps the scaled gap to a number between 0 and 1

DPO in Numbers

As the chosen gain pulls ahead of the rejected gain, the loss falls.

at the start

loss = 0.69

chosen gain

rejected gain

0

0

after training

loss = 0.31

chosen gain

rejected gain

+4

-6

Animation: Preference Tuning: RLHF and DPO

Animation clip: week05-clip3-preference-tuning.mp4

Silent version embedded (click to play). The narrated version is in the course repository.

What Preference Data Teaches

The model learns whatever the preferences reward, including bad habits.

In the notebook

a 21-second DPO run fixes the error
it carries over to headache and
leg swelling (not in training)
“reports” notes still get “Yes”

The risk

if the cautious answer always wins,
the model learns to hedge:
“further assessment is recommended”

Reward answers that are correct and useful, not answers that merely sound careful.



Reasoning Models

A Reasoning Model Thinks Before It Answers

It writes its reasoning inside <think> tags, then gives the answer.

A 58-year-old man has crushing chest pain.
ECG: ST elevation in leads II, III, and aVF.
What is the most likely occluded artery?

- A) left anterior descending
- B) right coronary
- C) left circumflex
- D) left main

<think> (illustrative trace)

ST elevation in II, III, and aVF
points to the inferior wall.

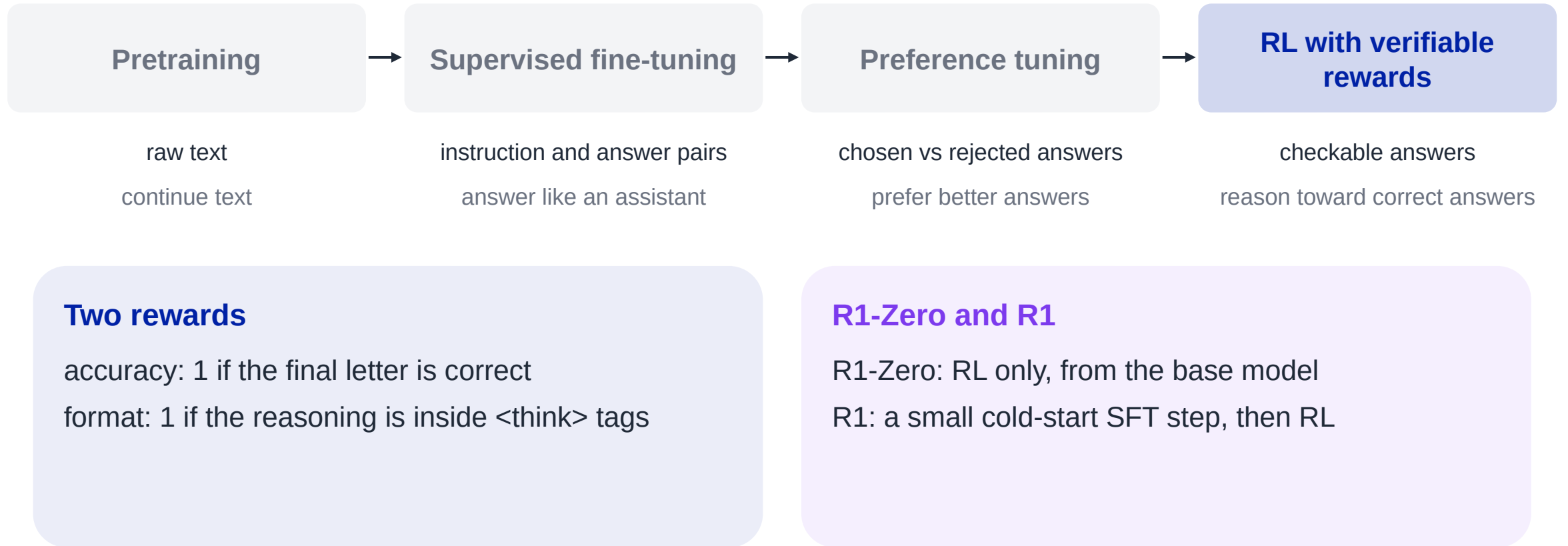
The inferior wall is usually supplied
by the right coronary artery.

</think> Answer: B

The reasoning is part of the output: it costs tokens, and it can be read and checked.

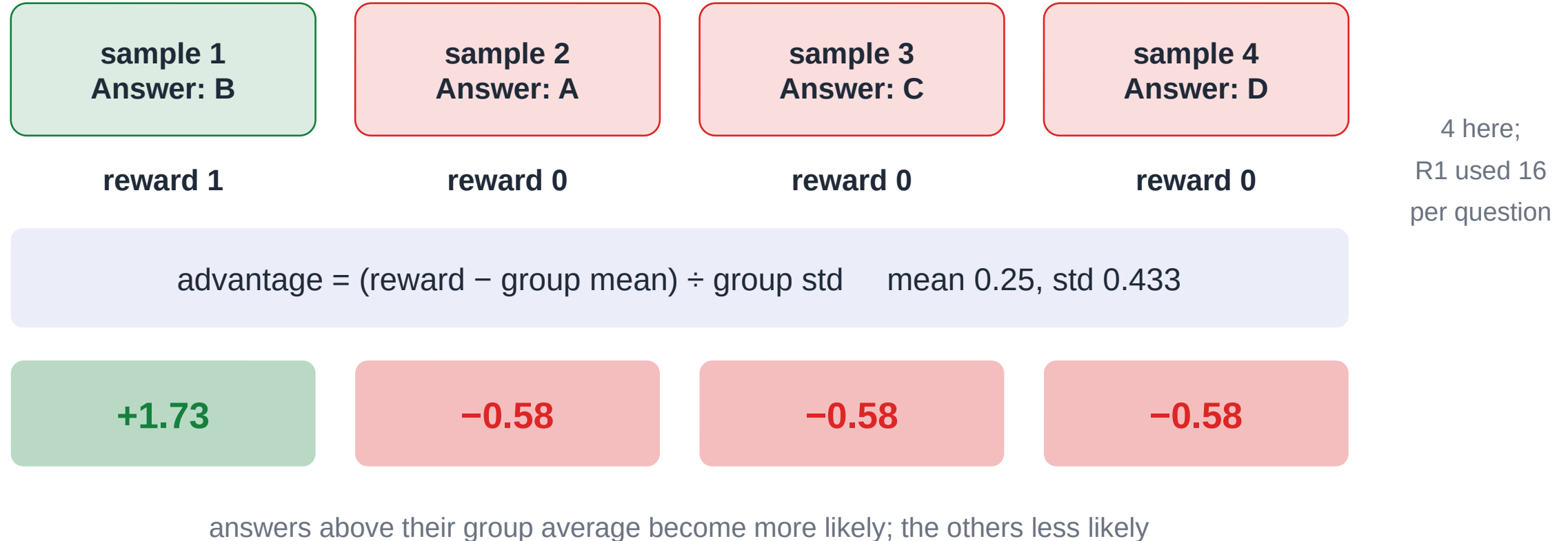
Rewards a Program Can Check

DeepSeek-R1 learned to reason from rule-based rewards, without a reward model.



GRPO: Compare Each Answer with Its Own Group

Sample a group of answers to one question; score each against the group.



Animation: Reasoning Models

Animation clip: week05-clip4-reasoning.mp4

Silent version embedded (click to play). The narrated version is in the course repository.

Thinking at Test Time

Thinking longer can fix an error, at the cost of more tokens and more time.

Note: The patient denies chest pain.

Question: Does the patient report chest pain?

Qwen3-0.6B, thinking off

“Yes, the patient reports chest pain.”

9 tokens · 0.9 s · wrong

Qwen3-0.6B, thinking on

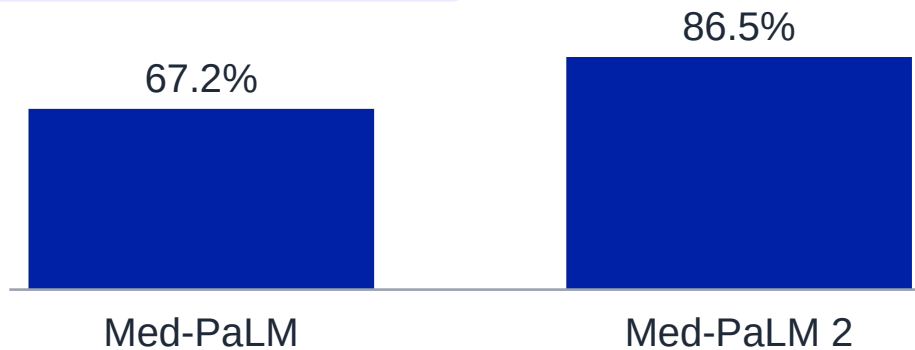
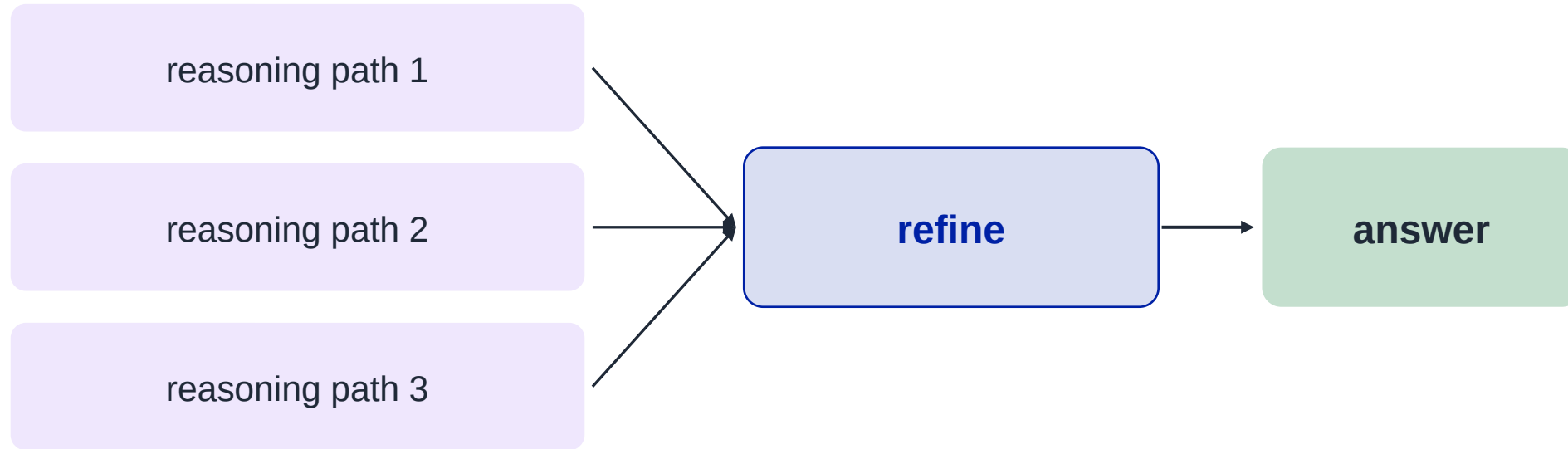
“The patient does not report chest pain.”

119 tokens · 11.4 s · correct

real outputs, CPU, greedy decoding

Med-PaLM 2: Ensemble Refinement

Sample several reasoning paths, then condition on all of them to write one answer.



MedQA accuracy
(stronger base model, medical
fine-tuning, and refinement together)

A Fluent Rationale Is Not Proof

Thinking can also lead a model away from the right answer.

thinking off

answer: B, right coronary
correct

thinking on

“ST elevation in the inferior leads is most commonly due to a blockage in the left anterior descending artery.”
after ~1,500 tokens: still unsure, leaning to A

**Check the answer, not the eloquence.
Week 12: evaluating reasoning and its explanations.**

Recap: From Pretrained Model to Clinical Assistant

Fine-tuning

Instruction and answer pairs, same loss, counted only on the answer.

LoRA

Freeze W , train $B \cdot A$: about 0.4% of the weights, a 4.6 MB adapter per task.

Preferences

RLHF learns a reward; DPO compares with a frozen copy. Data decides what is learned.

Reasoning

Rewards a program can check, GRPO, thinking at test time. Fluent is not proof.

Before Thursday

Read

Guo et al., DeepSeek-R1 (Nature 2025) and Singhal et al., Med-PaLM 2 (Nature Medicine 2025).

Write

Reading responses on Canvas before 1:00 PM Thursday.

Explore

Week 5 reference notebook (ungraded): LoRA, the DPO loss, and GRPO from scratch; short runs that fix the negation error.

Next week

Self-supervised learning and imaging foundation models.