TEXAS
The University of Texas at Austin

POST-TRAINING OF LLMS

Week 1. Why do we need post-training?

CHAN YOUNG PARK
Assistant Professor, iSchool, School of Computing, The University of Texas at Austin

Welcome! Help me calibrate the course

- 5 min survey
- Tell me what you have worked with!
- Responses do not affect your grade or your ability to remain in the course
- I will use the aggregate results to tune pace, refreshers, and lab support

Learning objective

- Course website: https://chan0park.github.io/ut-austin-llm-post-training/2026f
- Evaluate a task-specific model behavior and judge validity
- Choose between prompting, retrieval, and post-training
- Adapt open-weight models with SFT, LoRA, and DPO
- Diagnose training runs, regressions, and failure modes
- Communicate methods, results, and limitations with evidence

Instructor

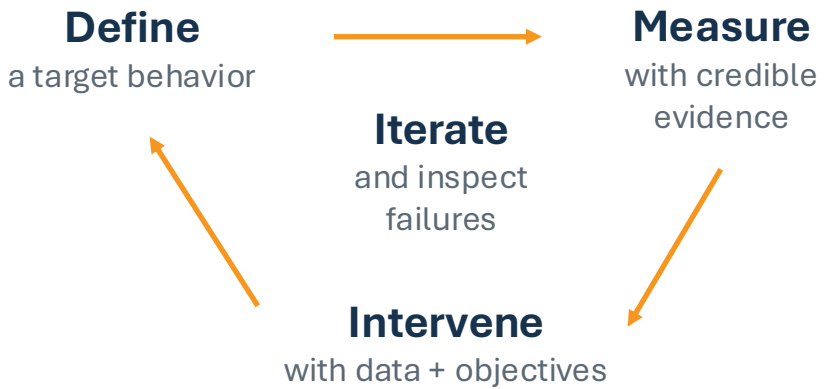
- Chan Young Park
- Assistant professor in iSchool
- Received my PhD in CS from CMU
- Was a visiting researcher at Microsoft Research
- Research area: socially intelligent AI, preference learning, pluralistic alignment, personalization

https://chan0park.github.io

The central question of this course

Q1. How do we deliberately change an LLM's behavior?

Q2. and know whether the change actually helped?



How a typical class works

- 60-70 minute lecture (concepts, methods, research)
- 15 minute break
- 90 minute Lab/workshop

Grading structure

- Group project: 45%
- HW0,1,2: 35% (5%, 15%, 15%)
- Participation: 10%
- Concept check quiz: 10%
- Four late days
- No final exam
- You may miss up to two class meetings without affecting your participation grade.

Semester-long group project

- Week 2: team formation
- Week 5: proposal (5%, one pager)
- Week 10: checkpoint (5%, 5min verbal presentation)
- Week 13: final presentation (10%)
- Week 15: final report (25%)
- A good project has
  - well-defined and well-justified goals/expected behaviors
  - well-designed methodological interventions
  - quantitative proofs
- Examples: “Make LLMs to always provide citations for every claim they make”

Class policies

- Attendance: in-person, (default) no livestream, two absences without penalty
- Late work: 4 late days across HW 0-2, then 10% reduction per day (not eligible for project milestones)
- Communication
  - Canvas: announcements and questions
  - Office hour: Chan (Friday 10-11am), Arjun (TBD)

AI use policy

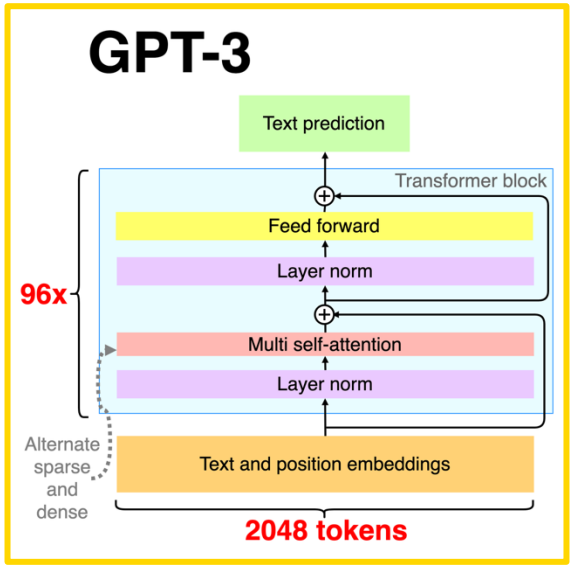
- Every submission includes a short individual **AI Use Statement**
- Permitted: concept clarification, debugging + setup, instructed use cases during hw (synthetic data generation, LLM prompting)
- Permitted + disclose: code that enters a submission, drafting/editing report
- Not permitted: data gold labels, evaluation rubrics, human preference data labeling, your observation, interpretation, conclusions

Technical Stack

- Python, Transformers, PyTorch
- SLURM (HPC)
- Google Colab for quick demo
- TACC for actual training

capability != behavior

PRE-TRAINING VS POST-TRAINING



(2020)

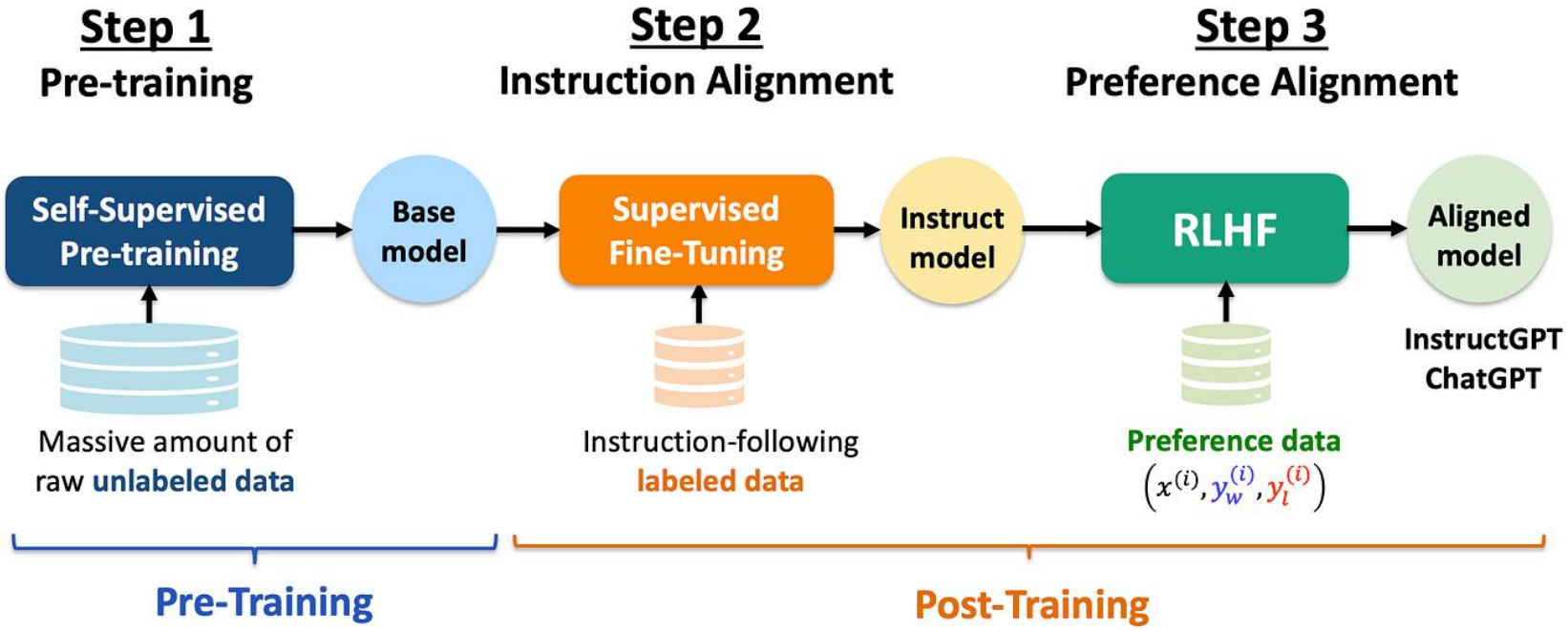
Q. What happened?

A. Post-training!

ChatGPT

(2022)

Post-Training: GPT-3 to ChatGPT

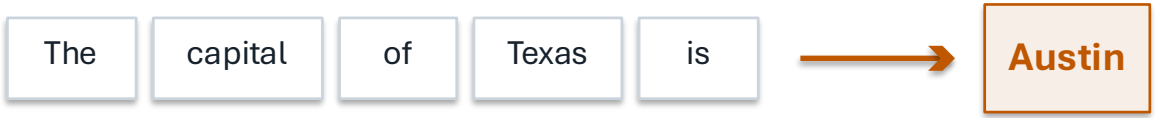


The Journey of GPT: Evolution from GPT-1 to GPT-4o | by LM Po | Medium

Pre-trained LMs are not yet an assistant

- Prompt: Write a polite email declining a meeting
- Pre-trained base model: “User: Write a polite email… Assistant: Sure, here is…”
- Instruction-tuned model: “Thank you for the invitation. Unfortunately, I won't be able to attend…
- Continuing text generation vs responding to an instruction
- What changed?

Pre-training: next token prediction



Repeat this prediction objective across many sequences

- Produces broad linguistic and world-pattern knowledge
- Learns many tasks implicitly from patterns in data
- Does not specify a single assistant interface or preference

LM objective leaves important choices unspecified

- **Intent**: Should the model continue, answer, classify, or refuse?
- **Format**: What roles, structure, length, and style are expected?
- **Preference**: Which of several plausible responses is better?
- **Constraint**: What policies or task boundaries should shape behavior?

These are not simply 'knowledge gaps'
They are questions about desired behavior

Post-training introduces a target behavior

- Post-training is an intervention
- Desired behavior:
  - Follow the task
  - Use the requested format
  - Respect specified constraints
- Persistent questions:
  - Whose target?
  - How is it represented in data?
  - What does the objective reward?
    (e.g. Qwen 2.5: truthfulness, helpfulness, conciseness, relevance, harmlessness, debiasing)

What changes after post-training

Stage	Example data	Learning signal
Pre-training	Text sequences	next-token likelihood
Supervised Fine-tuning	instruction → response	demonstration likelihood
Preference learning	prompt + chosen/rejected	relative preference
RL / verifiable feedback	sampled outputs or trajectories	reward / verifier

The data and signal become more deliberate

SFT: teaching through demonstrations

USER Rewrite this in exactly two sentences.

ASSISTANT Here is a two-sentence rewrite …

- Increase the likelihood of the demonstrated assistant tokens
  - Repeated examples can make direct answers, requested formats, and turn boundaries more probable

Beyond demonstrations: preferences and rewards

USER Rewrite this in exactly two sentences.

ASSISTANT Here is a two-sentence rewrite ...

Preferred
Chosen

ASSISTANT Sure! I'm happy to assist...

Less preferred
Rejected

- Preference learning raises the chosen response relative to the rejected one
- Reinforcement learning samples outputs and optimizes a learned or verifiable reward.
- The signal can shape directness, helpfulness, constraints, style, or safety behavior

A map of the semester

DEFINE	EVALUATE	ADAPT	PREFERENCE	STRESS-TEST
task + behavior (week 1,2)	data + metrics (week 3, 4, 7)	SFT · LoRA (week 5, 6)	RM · DPO · RLHF (week 8, 9)	RLAIF · RLVR · failures (week 10, 11,12)

Today's lab

- We are going to compare base vs instruction-tuned models
- Explore how they are different with Google Colab
  - Come up with many prompts that could elicit differences between the two models
  - Observation → hypothesis → test/validation
- Also think about "why" they are different
  - Data? Training method?

Qwen2.5-0.5B vs Qwen2.5-0.5B-Instruct

- Qwen2.5: Alibaba's LLM released in Sep 2024
- Base: raw text → continuation
  - Pretrained checkpoint
  - No assumed chat interface
    `"Explain post-training. Assistant:"`
  - Often sensitive to continuation framing
- Instruct: chat messages → response
  - Post-trained checkpoint
  - Expects its chat template
    `"<|im_start|>user Explain post- training.<|im_end|><|im_start|>assistant:"`
  - Optimized toward instruction-following behavior
- Fair comparison: same prompt, decoding settings, but use different templates
- Qwen 2.5 report: https://arxiv.org/abs/2412.15115
  - Pretraining 18B tokens
  - → SFT 1B+ demonstrations → DPO 150K → GRPO

Lab 1: pre-training vs post-training

1. Request or confirm your TACC account
2. Run the guided Colab comparison
3. Design your own diagnostic prompts
4. Complete the blind A/B comparison
5. Submit the Microsoft Form
6. Wrap-up discussion