

Post-training cheat sheet

60 post-training terms, each in a sentence or two, by topic.

01 · MODELS

Language model
A program that predicts the next piece of text, one token (word piece) at a time.

Transformer
The neural-network design behind almost every modern language model.

Parameters (weights)
The numbers inside a model that training adjusts. Large models have billions of them.

Pre-training
The first, longest phase: learn to predict the next token of a very large body of text. It teaches language and facts, not a specific task.

Prompt
The input the model continues from.

Sampling
Drawing an answer from the model's probabilities, one token at a time. The same prompt can give a different answer each time.

Temperature
A dial on sampling: below 1 favours the model's top guesses, above 1 spreads probability out for more variety (and more mistakes).

Quantization (int8)
Storing weights in fewer bits, for example 8-bit integers (int8) instead of 32-bit numbers: a 4× smaller file for almost the same answers.

02 · EVALS

Eval
A fixed set of problems used only to measure a model, never to train it.

Held-out
Kept out of all training on purpose, so a good score means the model learned the rule, not the answers.

Grader
Code that checks an answer and returns pass or fail plus a score. Also called a verifier.

pass@1
The chance that a single try is right. What a user experiences.

pass@k
The share of problems solved at least once in k tries. Shows whether the knowledge is in the model at all.

Headroom
pass@k minus pass@1: skill the model has but can't produce reliably. RL can turn it into pass@1.

Checkpoint
A saved copy of a model's weights at one point in training.

03 · SUPERVISED FINE-TUNING (SFT)

SFT (supervised fine-tuning)
Training on worked examples: show the right answer, train the model to predict it token by token.

Training loss
How surprised the model is by the correct tokens (cross-entropy). Training pushes it down; 0 would mean it predicts them perfectly.

Step
One update: run a batch of examples, measure the loss, nudge every weight a little.

Learning rate
How big each nudge to the weights is. Too big and training goes unstable; too small and it barely moves.

Backpropagation
The algorithm that works out how much each weight contributed to the loss, so each can be nudged the right way (its gradient).

Adam
The standard optimizer: turns gradients into weight updates with a step size per weight. Short for adaptive moment estimation.

Overfitting
Doing better on the training examples than on new ones: the model memorized instead of learning the rule.

04 · LORA

LoRA (low-rank adaptation)
Freeze the model and train a small add-on for each weight matrix ($W + A \cdot B$, with A and B thin). Trains a few percent of the weights.

Rank
How thin LoRA's add-on is. Rank 4 means A and B have 4 columns: higher rank, more capacity, more weights.

Adapter
The small set of trained LoRA weights. One base model can swap many adapters.

Full fine-tuning
Updating every weight in the model, the default way to do SFT.

Forgetting
Losing what pre-training taught while learning the new task. Often measured as the rise in loss on the original pre-training text.

05 · REWARDS

Reward
A score for one answer. RL makes high-scoring answers more likely.

Proxy reward
A cheaper stand-in for what you really want, like “shorter is better”. Easy to compute, easy to game.

Reward hacking
The model raises the reward without doing the task, by exploiting what the reward actually checks.

GRPO
Group Relative Policy Optimization: sample a group of answers per problem and push toward those that beat the group's average.

Advantage
How much better an answer scored than its group's average. Positive becomes more likely, negative less.

Preference pair
Two answers to one prompt, with a label saying which is better.

Reward model
A model trained on preference pairs to score any answer, standing in for people when no exact grader exists.

Bradley-Terry
The standard loss for reward models: train so the preferred answer scores higher than the rejected one.

LLM-as-judge
Using a large language model (LLM) to score answers, in place of a trained reward model or people.

06 · REINFORCEMENT LEARNING (RL)

RL (reinforcement learning)
Training from scores instead of examples: the model tries, a reward scores each try, and training favours what scored well.

Policy
RL's word for the model being trained: it decides what to output.

KL penalty
A penalty for drifting from the starting model's probabilities (Kullback-Leibler divergence). β sets how strong it is.

PPO
Proximal Policy Optimization: RL with a separate value model and clipped updates. Powerful, harder to tune.

RFT
Rejection-sampling fine-tuning: sample many answers, keep the ones that pass, run SFT on them.

Over-optimization
Pushing a learned reward so hard that the real quality falls while the reward keeps rising (Gao et al., 2022).

Sharpening
RL making answers the model could already find come out first, rather than teaching new ones. Seen when pass@k stops improving.

07 · PREFERENCE OPTIMIZATION (DPO)

DPO (Direct Preference Optimization)
Learns from preference pairs directly, without a reward model or RL: raise the chosen answer's likelihood relative to the rejected one.

β (beta)
A strength dial. In DPO and the KL penalty, higher β keeps the model closer to where it started.

Alignment tax
Capability lost while training for preferences or safety.

08 · MODEL SOUPING

Model souping
Averaging the weights of models fine-tuned from the same parent, with no extra training.

Specialist
A model fine-tuned on one slice of the task.

WiSE-FT
Weight-space ensembling: interpolate between a base model and its fine-tune to keep the best of both (Wortsman et al., 2022).

09 · DISTILLATION

Distillation
Training a small student model to imitate a big teacher model.

Teacher / student
The big model that already works, and the small, cheaper model learning from it.

Soft labels
The teacher's full probability for every next token, not just its top answer. Carries more information per example.

Dark knowledge
What soft labels reveal beyond the right answer: which wrong answers the teacher found plausible (Hinton et al., 2015).

10 · TEST-TIME COMPUTE

Best-of-N
Sample N answers and keep one. No training, more inference.

Test-time compute
Spending more computation per answer (more samples, longer reasoning) instead of more training.

Majority vote
Pick the answer that appears most often among the samples (self-consistency). Needs no grader.

11 · BENCHMARKS

Benchmark
A public, fixed eval that people trust to compare models. Examples: AIME, HumanEval, τ-bench, FinQA, LegalBench and AgentDojo.

Contamination
Benchmark problems leaking into training data, which inflates scores.

12 · TRAINING MECHANICS

Batch
The examples processed together in one step.

Gradient
For each weight, the direction and size of change that would lower the loss.