POSITION: DISTINGUISH CAPABILITY ELICITATION FROM CAPABILITY CREATION IN POST-TRAINING – A FREE-ENERGY PERSPECTIVE

Anonymous authors
Paper under double-blind review

ABSTRACT

Debates about large language model post-training often treat supervised fine-tuning (SFT) as imitation and reinforcement learning (RL) as discovery. But this distinction is too coarse. What matters is whether a training procedure increases the probability of behaviors the pretrained model could already produce, or whether it changes what the model can practically reach. **We argue that post-training research should distinguish between capability elicitation and capability creation.** We make this distinction operational by introducing the notion of accessible support: the set of behaviors that a model can practically produce under finite budgets. Post-training that reweights behaviors within this support is capability elicitation; whereas changing the support itself corresponds to capability creation. We develop this argument through a free-energy view of post-training. SFT and RL can both be seen as reweighting a pretrained reference distribution, only with different external signals. Demonstration signals define low-energy behavior for SFT, and reward signals define low-energy behavior for RL. When the update remains close to the base model, the main effect is local reweighting, not capability creation. Within this framework, the central question is no longer whether post-training is framed as SFT or RL, but whether it reweights behaviors already within reach, or instead expands the model's reachable behavioral space through search, interaction, tool use, or the incorporation of new information.

1 INTRODUCTION

Post-training has become a central stage in building large language models (LLMs). After large-scale pretraining, LLMs are commonly further shaped through supervised fine-tuning (Brown et al., 2020; Touvron et al., 2023), preference optimization (Rafailov et al., 2023; Ziegler et al., 2020), or reinforcement learning from human feedback (Stiennon et al., 2020; Ouyang et al., 2022). These procedures can substantially change model behavior, sometimes producing large gains in instruction following (Wei et al., 2022; Wang et al., 2023), coding (Chen et al., 2021; Rozière et al., 2024), and mathematical problem solving (Lightman et al., 2024; Shao et al., 2024). This raises a basic question about what post-training actually does. Does it elicit behaviors that were already present but unreliable in the pretrained base model, or create the capabilities that the base model could not previously reach?

A common way to frame this question is through the contrast between supervised fine-tuning (SFT) and reinforcement learning (RL) (Chu et al., 2025; Jiang et al., 2026). Both of these methods are applied after pretraining, further shaping the base model's behaviors. SFT is often described as imitation from demonstrations: the model is trained to reproduce behaviors present in a dataset (Zhou et al., 2023; Zhang et al., 2026). RL is often associated with reward-guided exploration and discovery (Wang et al., 2025). Candidate behaviors can be sampled or searched for, and then reinforced if they receive a high reward. This contrast is especially salient in recent discussions of reasoning models (OpenAI et al., 2024; Guo et al., 2025), where RL-style training is often credited with producing stronger reasoning behavior. It is therefore tempting to treat SFT as a mechanism for eliciting existing capabilities and RL as a mechanism for creating new ones.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

However, we argue that this identification is too coarse. Demonstrations and rewards describe the form of the training signal, but they do not determine the capability mechanism. The more relevant object is the set of behaviors that a model can practically produce under finite sampling, optimization, and divergence budgets. We introduce the concept of *accessible support* to formalize this idea. If post-training only increases the reliability of behaviors in the accessible support, we define it as *capability elicitation*. Instead, if it changes the accessible support itself, making previously unreachable behaviors accessible, we define it as *capability creation*.

Under this criterion, SFT and RL are algorithmic categories, whereas elicitation and creation are claims about how post-training changes the model's behavioral distribution. A supervised loss in SFT can also fit human-written answers, verifier-filtered samples, search-generated solutions, tool-assisted trajectories, or data distilled from stronger models (Zelikman et al., 2022; Sharma, 2025; Toshniwal et al., 2025; Gou et al., 2024). Conversely, a reward objective in RL can also only make existing behaviors more likely. Thus, a performance gain after SFT or RL does not by itself determine whether a new capability has been created. The performance gain depends on whether or not the evaluation metric can reflect the corresponding change over the behavioral distribution (elicitation or creation), regardless of the algorithmic categories (SFT or RL).

Our position: Post-training research should distinguish capability elicitation from capability creation: does a method reweight already accessible behaviors or expand what the model can reach? In this view, the central question is not whether a method is called SFT or RL. Rather, it is what training signal the task provides, where its candidate behaviors are drawn from, and whether the post-training pipeline expands the accessible behavioral space of the base model.

We formalize this position from a free-energy perspective on post-training. Variational forms have already been established in maximum-entropy RL (Theodorou & Todorov, 2012; Levine, 2018) and KL-regularized policy optimization (Rafailov et al., 2023). Our use of this perspective is diagnostic. It clarifies that demonstration fitting and reward optimization can both be understood as reshaping the pretrained distribution under an external training signal. Demonstrations define low-energy behavior for SFT, while rewards define low-energy behavior for RL. When the update remains close to the reference model, the primary effect is local reweighting rather than capability creation.

The rest of the paper develops this position as follows. We first clarify why the SFT–RL debate confounds algorithmic form with capability mechanism: SFT and RL specify different training signals, but neither label alone determines whether a gain reflects elicitation or creation. Then we introduce a free-energy perspective that describes SFT and KL-regularized RL as local reweighting of a pretrained reference distribution. Building on this view, we use accessible support to distinguish different kinds of reachability, leading to four regimes of post-training: demonstration-covered elicitation, tail reweighting, barrier-crossing discovery, and unsupported regimes. Finally, we discuss alternative views and clarify the scope of the framework.

2 The Capability Debate In Post-Training

The SFT–RL contrast is useful, but it does not by itself explain why post-training improves a model (Ethayarajh et al., 2024; Gheshlaghi Azar et al., 2024). SFT and RL specify different optimization objectives, yet a post-training pipeline contains more than the objective. It also contains the source of data, the procedure for generating candidate behaviors, and the signal for selecting among them (Gudibande et al., 2024; Ryd et al., 2026). The more important question is therefore not whether a method is called SFT or RL, but what kind of information the post-training provides (Korbak et al., 2022).

Notation. Before analyzing these signals, we establish standard notations. Let x denote an input prompt and y denote the response. We use $p_0(y|x)$ to represent the conditional distribution of the pretrained base model, which also serves as a reference model in RL. The behavioral distribution after post-training is denoted as $q(y|x)$, parameterized as $p_\theta(y|x)$ in practice.

2.1 The Capability of SFT Depends on the Demonstration Distribution

Supervised fine-tuning usually minimizes the negative log-likelihood on demonstration data:

$$\mathcal{L}_{\mathrm{SFT}}(\theta) = -\mathbb{E}_{p_{\mathrm{demo}}(x,y)} \log p_\theta(y|x), \tag{1}$$

where $p_{\text{demo}}(x, y)$ is the empirical demonstration distribution. From the form of the loss, SFT is indeed imitating demonstrations. But imitation is not synonymous with weakness. The supervised objective does not specify where the target behaviors come from. It only specifies that the model should increase the target behaviors' likelihood once they appear in the demonstration distribution.

This makes p_{demo} the central object. Different construction procedures induce different demonstration distributions, with different information content. If the demonstrations contain trajectories generated with richer or more informative methods, the same loss function can elicit behaviors that the base model rarely or never produces.

Thus, the limitation of SFT arises not from supervision itself, but from the inability of the demonstration distribution to adequately cover all target behaviors, especially the low-probability ones.

- If demonstrations densely cover the desired behavior, SFT can efficiently elicit and stabilize it.
- If demonstrations carry additional or more complex information than the base dataset, SFT may trigger behaviors that the original training set would not produce.
- However, if they contain no positive evidence for the target behavior, the supervised loss supplies no discovery signal by itself.

This means that any comparison between SFT and RL is ill-defined if it does not specify the source of demonstrations. When discussing the capability limits of SFT, the key question is not whether it imitates, but what distribution it imitates and how that distribution is constructed.

2.2 The Capability of RL Is Constrained by the Reference Distribution

Reinforcement learning is often assigned the opposite meaning: exploration and discovery. This interpretation is reasonable when an agent truly interacts with an environment, generates new trajectories, and receives external feedback. But many forms of language-model post-training are not unconstrained exploration. They usually optimize rewards or preferences near a reference model (Ziegler et al., 2020; Rafailov et al., 2023)

$$\max_{q} \; \mathbb{E}_{y\sim q(y|x)}[R(x, y)] - \beta \, \mathrm{KL}\left(q(y|x) \,\|\, p_0(y|x)\right). \tag{2}$$

Here, $R(x, y)$ is a reward score assigned to the response y, and $\beta > 0$ controls the strength of the KL constraint, thereby limiting how far the optimized distribution $q(y|x)$ can deviate from the reference model $p_0(y|x)$. The key issue is not whether *reward is used*, but how much freedom reward optimization has relative to the reference distribution.

When the KL constraint is strong, RL does not freely search over all possible responses. It mainly reweights responses within the behavior class already induced by the base model. A response that is initially rare but high-reward can therefore become frequent after training. Such a change may produce a large benchmark gain, but it only reweights already accessible behaviors rather than enabling new behaviors that were previously unreachable.

In contrast, if the training pipeline introduces exploration mechanisms that sample trajectories outside the high-probability manifold of $p_0(y|x)$, the resulting gain reflects capability creation. Because the reachable candidate set has expanded, the optimization is no longer bound by local distribution reweighting.

This distinction clarifies why not all uses of RL should be interpreted in the same way. Although both procedures may be called RL-based post-training, they do not make the same claim about capability.

2.3 The Hidden Axis: Elicitation Versus Creation

Therefore, the more important axis in the post-training debate is not supervision versus reinforcement, but capability elicitation versus capability creation, *i.e.*, whether post-training primarily increases the likelihood of already accessible behaviors, or expands the set of behaviors the model can reach.

This axis cuts across the existing mainstream SFT/RL distinction for performance (Chu et al., 2025; Jiang et al., 2026). SFT is not automatically mere elicitation: if its demonstrations are of sufficiently high quality, the supervised objective may transmit behaviors that the target model would not have

produced on its own (Sharma et al., 2024; Ye et al., 2025). Conversely, RL is not automatic discovery either: under strong reference constraints, reward optimization may only amplify behaviors already available for the base model (Korbak et al., 2022).

The next two sections develop the diagnostic framework needed to distinguish these cases: first through a free-energy perspective on local reweighting, and then through accessible support.

3 The Free-Energy Perspective

To formally distinguish capability elicitation from creation, we need a mathematical language to describe how post-training modifies the behavioral distribution of the base model. Here, we adopt a free-energy perspective (Mazzaglia et al., 2022; Taniguchi et al., 2025; Elliott et al., 2026), which allows us to interpret many post-training methods as reweighting a pretrained reference distribution.

In this perspective, a policy does not sample arbitrarily from all possible behaviors. Rather, it redistributes probability mass according to reward or energy relative to a reference distribution or prior dynamics. Our novel utilization of this framework is to diagnose the capability claims in post-training. When a model performs better after SFT, DPO, RLHF, or other post-training methods, we first ask whether the improvement can be explained as a reweighting of a reference distribution.

3.1 Preliminary for Free-Energy Framework

Many post-training objectives can be understood in analogy to the physical concept of free energy, commonly written as $F = E - TS$, where E is the energy of system, S is the entropy, and T is the temperature parameter.

Adapting this to the post-training, E corresponds to an effective energy derived from demonstration or reward signals, which encourages the model to favor highly preferred behaviors (*i.e.*, low-energy states). The entropy term S penalizes deviation from the pretrained reference distribution, encouraging the updated distribution to remain broad and close to the base model. The temperature T plays a role analogous to the KL-regularization factor β in RL, controlling the trade-off between exploiting preferred behaviors and preserving diversity. Formally, this free-energy objective can be re-expressed as

$$\mathcal{F}_x(q) = \mathbb{E}_{y\sim q(y|x)}[E(x,y)] + \beta\,\mathrm{KL}\left[\,q(y|x)\,\|\,p_0(y|x)\right], \tag{3}$$

where β corresponds to the inverse of temperature T in physics.

The optimal distribution under minimizing Equation (3) has the form of a Boltzmann-like distribution

$$q^*(y|x) = \frac{1}{Z_E(x)} p_0(y|x) \exp\left[-\frac{1}{\beta}E(x,y)\right]. \tag{4}$$

A full derivation of Equation (4) using Lagrange multipliers is provided in Appendix A. Noticeably, the solution in Equation (4) makes explicit that post-training reweights the reference distribution according to the effective energy E. Additionally, it also clearly illustrates that post-training is not selecting behaviors from a uniform space, but using an energy signal to reweight the reference distribution.

3.2 SFT as a Demonstration-Defined Energy

For supervised fine-tuning, the target distribution is defined by demonstrations instead of a scalar reward. Let $p_{\mathrm{demo}}(y|x)$ denote the empirical or idealized distribution of demonstration responses for prompt x. Optimizing SFT objective in Equation (1) is equivalent to minimizing

$$\mathbb{E}_{x\sim\mathcal{D}_x}\mathrm{KL}\left[\,p_{\mathrm{demo}}(\cdot|x)\,\|\,p_\theta(\cdot|x)\right], \tag{5}$$

up to a constant independent of θ.

To connect this statement with the free-energy formulation in Equation (3), we ask a reverse question: what effective energy $E_{\mathrm{SFT}}(x,y)$ would make the Boltzmann-reweighted solution equal to the demonstration distribution, *i.e.*, $q^*_{\mathrm{SFT}}(y|x) = p_{\mathrm{demo}}(y|x)$. This gives us

$$E_{\mathrm{SFT}}(x,y) = -\beta \log \frac{p_{\mathrm{demo}}(y|x)}{p_0(y|x)}. \tag{6}$$

In the above equation, we set the additive constant term to zero as a *convention*, in which the corresponding partition function is exactly $Z_E(x) = 1$. This equivalence should be interpreted carefully. Practical SFT simply minimizes the negative log-likelihood rather than explicitly constructing E_{SFT} to solve a free-energy objective. From this free-energy perspective, SFT can be seen as introducing an effective energy over possible responses. If a response is more likely under the demonstrations than under the base model, it receives lower energy.

More importantly, this interpretation carries an implicit constraint: whenever $p_{\mathrm{demo}}(y|x) > 0$, the reference probability of base model $p_0(y|x)$ must also be nonzero. When the base model assigns a nonzero probability, this ratio is finite and the energy landscape is well-defined, allowing Boltzmann reweighting to smoothly adjust the probability mass across responses.

However, if this condition is violated – meaning the demonstrations require behaviors that the base model cannot generate (*i.e.*, $p_0(y|x) \to 0$) – the energy becomes extremely large, creating a singularity. In this case, the Boltzmann normalization fails, the local reweighting interpretation breaks down, and post-training can no longer be understood as simply amplifying or suppressing existing behaviors. Thus the singularity naturally marks the boundary between capability elicitation and capability creation. This viewpoint not only clarifies the effective range of the free-energy formulation, but also connects to the concept of accessible support and the categories of regimes developed in Section 4.

3.3 RL as Reward-Defined Energy

The RL objective in Equation (2) is the reward-maximizing counterpart of the free-energy objective in Equation (3). Thus, we set the effective energy to the negative reward $E_R(x, y) = -R(x, y)$. The corresponding optimal Boltzmann distribution becomes

$$q^*_{\mathrm{RL}}(y|x) = \frac{1}{Z_R(x)} p_0(y|x) \exp\left[\frac{1}{\beta} R(x, y)\right]. \tag{7}$$

The parameter β controls the sharpness of the reward tilt. Large β keeps the distribution close to the reference model, while small β concentrates probability more aggressively on high-reward outputs.

This shows that, when the β is large, *i.e.*, KL constraint is strong, RL mainly applies an exponential tilt to the reference distribution according to reward. The higher the reward, the easier a behavior is to amplify; but this amplification remains constrained by the prior mass in p_0.

This constraint can be seen by asking when one behavior becomes more likely than another. For two candidate responses y_a and y_b, their relative probability under the optimized distribution in Equation (7) is

$$\log \frac{q^*_{\mathrm{RL}}(y_a|x)}{q^*_{\mathrm{RL}}(y_b|x)} = \log \frac{p_0(y_a|x)}{p_0(y_b|x)} + \frac{1}{\beta}\left[R(x, y_a) - R(x, y_b)\right]. \tag{8}$$

For a rare behavior to overtake a common behavior, its reward advantage must offset its prior probability disadvantage by $(R(x, y_a) - R(x, y_b))/\beta$. Therefore, reward training can produce seemingly sudden behavioral changes: when the reward gap or temperature parameter β crosses a threshold, probability can rapidly move toward behaviors that were previously unlikely under the reference model.

This is a distributional effect of the exponential tilt. Whether the shifted behavior should be interpreted as elicitation or creation depends on whether that behavior was practically reachable before training, a question we elaborate further in the next section.

4 Accessible Support and Four Regimes

4.1 Basins, Tails, and Barriers

The free-energy perspective shows that post-training can be understood as reshaping a reference distribution. Crucially, for an autoregressive language model, the softmax distributions assign nonzero probability to many token sequences, so strict mathematical support is too weak for capability analysis. What matters is whether a behavior can be practically produced under finite sampling, decoding, search, optimization, and divergence budgets.

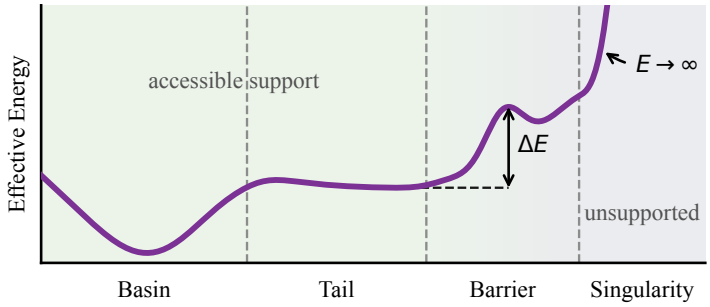


Figure 1: Schematic energy landscape for accessible support. Behaviors in basins are easily produced by the base model, while tail behaviors are rare but reachable under larger sampling or search budgets. Barrier regions require crossing low-probability intermediate states. In the unsupported limit, the effective energy becomes divergent, and the local reweighting view no longer applies.

We therefore use the more practical notion – accessible support. This is not meant to be a new formal object with a unique estimator. Rather, it is a diagnostic concept, which can be understood as a probability landscape over behavior classes induced by the base model, as illustrated in Figure 1.

- Some behaviors lie in **basins**: they have enough probability that the model already produces them stably under ordinary sampling.
- Some behaviors lie in **tails**: they are rare at pass@1 or greedy decoding, but appear often enough under larger sampling budgets, such as pass@N or best-of-N.
- Other behaviors sit behind **barriers**: reaching them requires a series of low-probability intermediate steps, long-term planning, or tool invocation.

These regions are not fixed properties of a task alone. They are relative to a base model, a decoding policy, an evaluation signal, and a compute budget. This relativity is why capability claims should specify the mechanism behind the gain rather than only reporting the gain itself.

This landscape suggests four useful regimes for post-training. The first two remain within the accessible support inherited from the base model: post-training either stabilizes a high-probability basin or amplifies a low-probability tail. In these cases, the free-energy view remains a useful local description of how training reweights behaviors that the base model can already produce under finite budgets. The third regime lies near the boundary of accessible support: the target behavior is not strictly impossible, but it sits behind barriers that ordinary sampling cannot reliably cross. The fourth regime is the limiting case in mathematics, the effective energy becomes divergent (a.k.a. singular), and the local reweighting interpretation breaks down.

4.2 Demonstration-Covered Elicitation

The first regime occurs when the target behavior already lies in a high-probability basin of the base model and is sufficiently covered by the demonstration or preference signal. In this case, post-training does not need to discover a new behavior class; it mainly stabilizes a behavior that the model can already produce reliably. Many forms of instruction following, format control, and stylistic alignment can fall into this regime.

Here, SFT and RL methods may look similar. SFT increases the probability of the target behavior by fitting demonstrations. Preference optimization increases it by comparing preferred outputs against alternatives. If both demonstrations and rewards point toward the same behavioral basin, then both procedures mainly stabilize the same reachable region. They may still differ in data efficiency, training stability, and optimization risk, but this difference should not be described as a difference between *imitation* and *creation*.

4.3 Tail Reweighting

When the target behavior is not sufficiently covered by demonstrations, the second regime still remains within the accessible support of the base model, but in its low-probability tail. It may

be invisible under greedy decoding or pass@1, but appear under repeated sampling, best-of-N, or verifier selection. This regime is especially easy to misread as capability creation, because it can produce large benchmark gains.

The free-energy analysis explains why this improvement can be large. If the target behavior y_a is rare but high-reward, and the competing behavior y_b is common but low-reward, then KL-regularized reward optimization changes their log-odds (8). Once the reward advantage is large enough to offset the prior probability disadvantage, the rare behavior can quickly become common. Aggregate metrics may jump, but mechanistically this may still be amplification of an existing tail behavior. The key evidence for this regime is whether the base model can already produce the target behavior under a larger sampling budget.

This regime also shows that RL outperforming ordinary SFT does not necessarily mean that RL created a new capability. If ordinary SFT demonstrations do not contain rare successes, but a reward model or verifier can select those successes during sampling, then the advantage comes from a tail-selection signal. The more appropriate comparison is not ordinary SFT, but SFT on reward-selected, verifier-selected, or search-generated samples (Zelikman et al., 2022; Gou et al., 2024).

4.4 Barrier-Crossing Discovery

The third regime concerns behaviors whose reachability depends on how trajectories are generated and preserved. Unlike tail reweighting, where a complete target behavior could appear under sufficiently large sampling budgets, barrier-crossing discovery requires the model to maintain a sequence of low-probability or unstable intermediate states before final success is reached. Long reasoning, multi-step tool use, interactive problem solving, and code generation that requires exploring partial programs can all have this structure.

This is where the free-energy view reaches the boundary of its local reweighting interpretation. A finite energy tilt can amplify candidates that the base model already produces, but it cannot by itself make ordinary sampling reliably preserve the trajectories to reach the target behavior. The bottleneck is therefore not only the low probability of the final output, but the reachability of the path that produces it.

Crossing barriers requires changing the trajectory-generation process. Search can preserve and expand multiple intermediate candidates; process supervision can reward useful steps before the final answer is reached; verifiers can filter partial or final solutions; tools and environment interaction can introduce information not contained inside the base model. RL can be important here, but its role usually comes from being coupled with these mechanisms, not from isolated reward maximization.

This distinction is especially important for reasoning models. If the improvement comes from moving already sampled reasoning traces into a high-probability region, it is closer to tail reweighting. Otherwise, the improvement is closer to barrier-crossing discovery.

4.5 Unsupported Regimes

The fourth regime is the singular limit of the preceding cases. Here, the target behaviors fall outside the support of the base model, hence there is no positive probability assigned to it. In the case of $p_0(y|x) = 0$, the effective energy becomes divergent and the free-energy reweighting formulation is no longer well-defined. In this case, post-training lacks the behavioral material needed to operate. Ordinary SFT has no positive examples to imitate, while KL-regularized RL has no candidate trajectories to select and reinforce. If neither demonstrations nor generated trajectories reach the relevant region, supervised likelihood fitting has nothing to copy, and sparse reward has nothing to bootstrap.

Unsupported does not mean impossible forever. It only means unreachable relative to the current base model, budget, and training process. Stronger pretraining, additional data, different architectures, or other methods can solve this issue to a certain extent. But these changes should be understood as introducing new information or structure, not as the effect of local post-training alone.

4.6 Summary

To sum up, the four regimes form a hierarchy from elicitation to creation: demonstration-covered elicitation stabilizes high-probability basins; tail reweighting amplifies rare but reachable behaviors;

barrier-crossing discovery changes the trajectory-generation process; unsupported regimes require new information, tools, or learning processes.

5 Alternative Views and Counterarguments

View 1. RL really can create capabilities.

A strong alternative view is that RL really can create capabilities that SFT struggles to obtain. In mathematical reasoning, code generation, and many other scenarios, it is easier to provide correctness feedback than to write optimal demonstrations (Shao et al., 2024; Chen et al., 2021). As long as there is a verifier or environment reward, RL may discover strategies that humans did not demonstrate.

We agree that RL can participate in capability creation, but we disagree with attributing creation to the label *RL* itself. The key question is whether reward is coupled with mechanisms that change reachability. In such cases, RL is one part of a larger system: reward signals, jointly with search, interaction, or new information, change the behavior-generation process.

Therefore, strong cases of RL-driven capability creation are better understood as barrier-crossing discovery rather than local reward reweighting. Reward matters, but it works because the system can expose, select, and reinforce trajectories that ordinary sampling previously struggled to reach (Trinh et al., 2024; Khanov et al., 2024). This interpretation preserves the advantage of RL while avoiding the claim that all post-RL performance gains are capability creation.

View 2. SFT is only imitation and therefore inherently weaker.

Another view holds that SFT maximizes demonstration likelihood and therefore can only copy behaviors already contained in the dataset; it cannot go beyond demonstrations (Zhou et al., 2023). This view is correct when demonstrations are narrow or low-quality. If the target behavior does not appear in the demonstrations and is not exposed by the data-generation process, ordinary SFT has no direct mechanism for discovering it.

But this does not show that SFT is inherently weak. SFT imitates a distribution, but the supervised loss itself does not tell us where the data came from. The same supervised objective can be applied to human-written answers, search-generated trajectories, or reward-filtered samples, but these data sources carry different information and induce different reachability structures (Zelikman et al., 2022; Ye et al., 2025; Quan, 2024).

Thus, comparisons between SFT and RL must specify the source of demonstrations. If weak-demonstration SFT is compared against RL with extra features, the experiment may conflate the effect of reward optimization with the effect of additional candidates. The framework of our paper requires these factors to be separated.

View 3. The physics language is only metaphor.

One might worry that terms such as energy, entropy, basins, and barriers are too loose. Language models are not thermodynamic systems and are not in physical equilibrium. If these terms are only rhetorical, they do not improve the rigor of the argument.

This concern is reasonable. This paper uses physics language only in a restricted sense: the objective

$$\mathbb{E}_q[E(x,y)] + \beta\,\mathrm{KL}[\,q(y|x)\,\|\,p_0(y|x)] \tag{9}$$

does have a free-energy-like structure, and its optimum is a Boltzmann reweighting of the reference distribution. Thus, *basins*, *tails*, and *barriers* are not literal thermodynamic concepts, but shorthand for describing distributional structure and finite-budget reachability. Their role is to help distinguish local reweighting, mode switching, and support expansion.

We do not assume that training reaches thermodynamic equilibrium, nor that all performance jumps are genuine phase transitions. Free-energy language is only an organizing principle. Phenomena that this form cannot explain, especially reachability changes from new information, must be explained by additional mechanisms.

VIEW 4. OPTIMIZATION DYNAMICS ARE UNDERESTIMATED.

A final objection is that SFT and RL have very different optimization dynamics, which can strongly affect practical outcomes. We agree that optimization dynamics matter. We do not replace optimization analysis with a distributional view. Instead, it separates two levels: how an algorithm moves through parameter space, and how it ultimately changes the behavioral distribution and accessible support.

Therefore, optimization dynamics should be interpreted relative to the regime. In basin regimes, many optimization paths may stabilize the same behavior. In tail regimes, optimization succeeds only if rare successes are sampled and selected. In barrier-crossing regimes, optimization must be coupled with a process that changes the trajectory distribution. In unsupported regimes, no optimizer can amplify behaviors that are never generated or evaluated.

These views do not overturn the paper's position, but delimit its scope. Our claim is narrower: claims about capability creation need to specify their mechanism. A performance improvement may come from demonstration-covered stabilization, tail amplification, barrier-crossing discovery, or new information and tools. Different explanations require different evidence.

6 CONCLUSION

The debate over SFT and RL in post-training is often framed as a debate between imitation and discovery. We have argued that this framing obscures the more important distinction between capability elicitation and creation. SFT and RL are algorithmic categories; elicitation and creation are claims about how a post-training procedure transforms the model's behavioral distribution and accessible support. A method should not be credited with creating a capability merely because it improves performance after reward optimization. Likewise, SFT should not be dismissed as weak imitation without examining the source and coverage of its demonstrations.

The free-energy perspective provides a compact way to make this distinction precise. Both demonstration-driven fitting and KL-regularized reward optimization can be viewed as transformations of a pretrained reference distribution under an external preference signal. SFT specifies low-energy behavior through demonstrations; RL specifies low-energy behavior through reward. When the resulting update remains close to the reference distribution, its primary effect is to reweight probability mass within the base model's accessible support.

This view leads to a regime-based understanding of post-training. In demonstration-covered regimes, SFT and RL may behave similarly because both stabilize behaviors that are already accessible. In tail-reweighting regimes, reward-based methods can outperform ordinary SFT by amplifying rare but already reachable behaviors. In barrier-crossing regimes, capability gains require mechanisms that alter the trajectory-generation process itself. In unsupported regimes, neither ordinary SFT nor RL should be expected to create a capability without additional information or changes to the learning process.

The central question for post-training is therefore not whether RL is categorically more powerful than SFT. It is whether a method merely changes the probabilities of behaviors that were already reachable, or changes what the model can reach. Future work on post-training should make this distinction explicit. Performance improvement is evidence that post-training changed what the model tends to do. But performance improvement alone is not sufficient evidence that post-training changed what the model can practically reach. Capability creation is a stronger claim than performance improvement.

AI USE STATEMENT

During the preparation of this manuscript, the authors used ChatGPT to polish the writing and assist in literature discovery. The AI tool was strictly applied to refine human-authored drafts, and no text was generated by AI directly. All referenced works were manually selected, read, and verified by the authors. No citations were directly generated by AI. The authors take full responsibility for the content, originality, and integrity of this paper.

REFERENCES

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code, 2021. URL https://arxiv.org/abs/2107.03374.

Tianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V Le, Sergey Levine, and Yi Ma. SFT memorizes, RL generalizes: A comparative study of foundation model post-training. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=dYur3yabMj.

Chris Elliott, Einar Urdshals, David Quarel, Matthew Farrugia-Roberts, and Daniel Murfet. Stagewise reinforcement learning and the geometry of the regret landscape, 2026. URL https://arxiv.org/abs/2601.07524.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Model alignment as prospect theoretic optimization. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=iUwHnoENnl.

Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Remi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. A general theoretical paradigm to understand learning from human preferences. In Sanjoy Dasgupta, Stephan Mandt, and Yingzhen Li (eds.), *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pp. 4447–4455. PMLR, 02–04 May 2024. URL https://proceedings.mlr.press/v238/gheshlaghi-azar24a.html.

Zhibin Gou, Zhihong Shao, Yeyun Gong, yelong shen, Yujiu Yang, Minlie Huang, Nan Duan, and Weizhu Chen. ToRA: A tool-integrated reasoning agent for mathematical problem solving. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Ep0TtjVoap.

Arnav Gudibande, Eric Wallace, Charlie Victor Snell, Xinyang Geng, Hao Liu, Pieter Abbeel, Sergey Levine, and Dawn Song. The false promise of imitating proprietary language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Kz3yckpCN5.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Honghui Ding, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jingchang Chen, Jingyang Yuan, Jinhao Tu, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaichao You, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan

Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z. URL http://dx.doi.org/10.1038/s41586-025-09422-z.

Haitao Jiang, Wenbo Zhang, Jiarui Yao, Hengrui Cai, Sheng Wang, and Rui Song. Supervised fine-tuning versus reinforcement learning: A study of post-training methods for large language models, 2026. URL https://arxiv.org/abs/2603.13985.

Maxim Khanov, Jirayu Burapacheep, and Yixuan Li. ARGS: Alignment as reward-guided search. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=shgx0eqdw6.

Tomasz Korbak, Ethan Perez, and Christopher L Buckley. Rl with kl penalties is better viewed as bayesian inference, 2022. URL https://arxiv.org/abs/2205.11275.

Sergey Levine. Reinforcement learning and control as probabilistic inference: Tutorial and review, 2018. URL https://arxiv.org/abs/1805.00909.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=v8L0pN6EOi.

Pietro Mazzaglia, Tim Verbelen, Ozan Çatal, and Bart Dhoedt. The free energy principle for perception and action: A deep learning perspective. *Entropy*, 24(2), 2022. ISSN 1099-4300. doi: 10.3390/e24020301. URL https://www.mdpi.com/1099-4300/24/2/301.

OpenAI, :, Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, Alex Iftimie, Alex Karpenko, Alex Tachard Passos, Alexander Neitz, Alexander Prokofiev, Alexander Wei, Allison Tam, Ally Bennett, Ananya Kumar, Andre Saraiva, Andrea Vallone, Andrew Duberstein, Andrew Kondrich, Andrey Mishchenko, Andy Applebaum, Angela Jiang, Ashvin Nair, Barret Zoph, Behrooz Ghorbani, Bohan Zhang, Ben Rossen, Benjamin Sokolowsky, Boaz Barak, Bob McGrew, Borys Minaiev, Botao Hao, Bowen Baker, Brandon Houghton, Brandon McKinzie, Brydon Eastman, Camillo Lugaresi, Cary Bassin, Cary Hudson, Chak Ming Li, Charles de Bourcy, Chelsea Voss, Chen Shen, Chong Zhang, Chris Koch, Chris Orsinger, Christopher Hesse, Claudia Fischer, Clive Chan, Dan Roberts, Daniel Kappler, Daniel Levy, Daniel Selsam, David Dohan, David Farhi, David Mely, David Robinson, Dimitris Tsipras, Doug Li, Dragos Oprica, Eben Freeman, Eddie Zhang, Edmund Wong, Elizabeth Proehl, Enoch Cheung, Eric Mitchell, Eric Wallace, Erik Ritter, Evan Mays, Fan Wang, Felipe Petroski Such, Filippo Raso, Florencia Leoni, Foivos Tsimpourlas, Francis Song, Fred von Lohmann, Freddie Sulit, Geoff Salmon, Giambattista Parascandolo, Gildas Chabot, Grace Zhao, Greg Brockman, Guillaume Leclerc, Hadi Salman, Haiming Bao, Hao Sheng, Hart Andrin, Hessam Bagherinezhad, Hongyu Ren, Hunter Lightman, Hyung Won Chung, Ian Kivlichan, Ian O'Connell, Ian Osband, Ignasi Clavera Gilaberte, Ilge Akkaya, Ilya

Kostrikov, Ilya Sutskever, Irina Kofman, Jakub Pachocki, James Lennon, Jason Wei, Jean Harb, Jerry Twore, Jiacheng Feng, Jiahui Yu, Jiayi Weng, Jie Tang, Jieqi Yu, Joaquin Quiñonero Candela, Joe Palermo, Joel Parish, Johannes Heidecke, John Hallman, John Rizzo, Jonathan Gordon, Jonathan Uesato, Jonathan Ward, Joost Huizinga, Julie Wang, Kai Chen, Kai Xiao, Karan Singhal, Karina Nguyen, Karl Cobbe, Katy Shi, Kayla Wood, Kendra Rimbach, Keren Gu-Lemberg, Kevin Liu, Kevin Lu, Kevin Stone, Kevin Yu, Lama Ahmad, Lauren Yang, Leo Liu, Leon Maksin, Leyton Ho, Liam Fedus, Lilian Weng, Linden Li, Lindsay McCallum, Lindsey Held, Lorenz Kuhn, Lukas Kondraciuk, Lukasz Kaiser, Luke Metz, Madelaine Boyd, Maja Trebacz, Manas Joglekar, Mark Chen, Marko Tintor, Mason Meyer, Matt Jones, Matt Kaufer, Max Schwarzer, Meghan Shah, Mehmet Yatbaz, Melody Y. Guan, Mengyuan Xu, Mengyuan Yan, Mia Glaese, Mianna Chen, Michael Lampe, Michael Malek, Michele Wang, Michelle Fradin, Mike McClay, Mikhail Pavlov, Miles Wang, Mingxuan Wang, Mira Murati, Mo Bavarian, Mostafa Rohaninejad, Nat McAleese, Neil Chowdhury, Neil Chowdhury, Nick Ryder, Nikolas Tezak, Noam Brown, Ofir Nachum, Oleg Boiko, Oleg Murk, Olivia Watkins, Patrick Chao, Paul Ashbourne, Pavel Izmailov, Peter Zhokhov, Rachel Dias, Rahul Arora, Randall Lin, Rapha Gontijo Lopes, Raz Gaon, Reah Miyara, Reimar Leike, Renny Hwang, Rhythm Garg, Robin Brown, Roshan James, Rui Shu, Ryan Cheu, Ryan Greene, Saachi Jain, Sam Altman, Sam Toizer, Sam Toyer, Samuel Miserendino, Sandhini Agarwal, Santiago Hernandez, Sasha Baker, Scott McKinney, Scottie Yan, Shengjia Zhao, Shengli Hu, Shibani Santurkar, Shraman Ray Chaudhuri, Shuyuan Zhang, Siyuan Fu, Spencer Papay, Steph Lin, Suchir Balaji, Suvansh Sanjeev, Szymon Sidor, Tal Broda, Aidan Clark, Tao Wang, Taylor Gordon, Ted Sanders, Tejal Patwardhan, Thibault Sottiaux, Thomas Degry, Thomas Dimson, Tianhao Zheng, Timur Garipov, Tom Stasi, Trapit Bansal, Trevor Creech, Troy Peterson, Tyna Eloundou, Valerie Qi, Vineet Kosaraju, Vinnie Monaco, Vitchyr Pong, Vlad Fomenko, Weiyi Zheng, Wenda Zhou, Wenting Zhan, Wes McCabe, Wojciech Zaremba, Yann Dubois, Yinghai Lu, Yining Chen, Young Cha, Yu Bai, Yuchen He, Yuchen Zhang, Yunyun Wang, Zheng Shao, and Zhuohan Li. Openai o1 system card, 2024. URL https://arxiv.org/abs/2412.16720.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf.

Shanghaoran Quan. Automatically generating custom context-driven SFT data for LLMs with multi-granularity. In *Adaptive Foundation Models: Evolving AI for Personalized and Efficient Learning*, 2024. URL https://openreview.net/forum?id=wu8NIjf8pD.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=HPuSIXJaa9.

Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Romain Sauvestre, Tal Remez, Jérémy Rapin, Artyom Kozhevnikov, Ivan Evtimov, Joanna Bitton, Manish Bhatt, Cristian Canton Ferrer, Aaron Grattafiori, Wenhan Xiong, Alexandre Défossez, Jade Copet, Faisal Azhar, Hugo Touvron, Louis Martin, Nicolas Usunier, Thomas Scialom, and Gabriel Synnaeve. Code llama: Open foundation models for code, 2024. URL https://arxiv.org/abs/2308.12950.

Emil Ryd, Henning Bartsch, Julian Stastny, Joe Benton, and Vivek Hebbar. Removing sandbagging in llms by training with weak supervision, 2026. URL https://arxiv.org/abs/2604.22082.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Archit Sharma, Sedrick Keh, Eric Mitchell, Chelsea Finn, Kushal Arora, and Thomas Kollar. A critical evaluation of AI feedback for aligning large language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=FZQYfmsmX9.

Asankhaya Sharma. Eliciting fine-tuned transformer capabilities via inference-time techniques, 2025. URL https://arxiv.org/abs/2506.08060.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 3008–3021. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/1f89885d556929e98d3ef9b86448f951-Paper.pdf.

Tadahiro Taniguchi, Ryo Ueda, Tomoaki Nakamura, Masahiro Suzuki, and Akira Taniguchi. Generative emergent communication: Large language model is a collective world model, 2025. URL https://arxiv.org/abs/2501.00226.

Evangelos A. Theodorou and Emanuel Todorov. Relative entropy and free energy dualities: Connections to path integral and kl control. In *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, pp. 1466–1473, 2012. doi: 10.1109/CDC.2012.6426381.

Shubham Toshniwal, Wei Du, Ivan Moshkov, Branislav Kisacanin, Alexan Ayrapetyan, and Igor Gitman. Openmathinstruct-2: Accelerating AI for math with massive open-source instruction data. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=mTCbq2QssD.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023. URL https://arxiv.org/abs/2307.09288.

Trieu H Trinh, Yuhuai Wu, Quoc V Le, He He, and Thang Luong. Solving olympiad geometry without human demonstrations. *Nature*, 625(7995):476–482, 2024.

Shuhe Wang, Shengyu Zhang, Jie Zhang, Runyi Hu, Xiaoya Li, Tianwei Zhang, Jiwei Li, Fei Wu, Guoyin Wang, and Eduard Hovy. Reinforcement learning enhanced llms: A survey, 2025. URL https://arxiv.org/abs/2412.10400.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions, 2023. URL https://arxiv.org/abs/2212.10560.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=gEZrGCozdqR.

Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. LIMO: Less is more for reasoning. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=T2TZ0RY4Zk.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. STar: Bootstrapping reasoning with reasoning. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=_3ELRdg2sgI.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Guoyin Wang, and Fei Wu. Instruction tuning for large language models: A survey. *ACM Comput. Surv.*, 58(7), January 2026. ISSN 0360-0300. doi: 10.1145/3777411. URL https://doi.org/10.1145/3777411.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, LILI YU, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. Lima: Less is more for alignment. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 55006–55021. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/ac662d74829e4407ce1d126477f4a03a-Paper-Conference.pdf.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences, 2020. URL https://arxiv.org/abs/1909.08593.

A Detailed Derivation for the Boltzmann Reweighting Solution

To derive the minimizer of free-energy, we introduce a Lagrange multiplier λ for the normalization constraint $\sum_y q(y|x) = 1$ and minimize the free energy in Equation (3) as:

$$\mathcal{L}_x(q, \lambda) = \sum_y q(y|x)E(x, y) + \beta \sum_y q(y|x) \log \frac{q(y|x)}{p_0(y|x)} + \lambda \left[\sum_y q(y|x) - 1 \right]. \tag{10}$$

Taking the functional derivative with respect to $q(y|x)$ and setting it to zero gives

$$E(x, y) + \beta \left[\log \frac{q(y|x)}{p_0(y|x)} + 1 \right] + \lambda = 0. \tag{11}$$

Solving for $q(y|x)$ gives

$$q(y|x) = p_0(y|x) \exp \left[-\frac{E(x, y)}{\beta} - 1 - \frac{\lambda}{\beta} \right]. \tag{12}$$

Since the factor $\exp[-1 - \lambda/\beta]$ is independent of y, it only affects the overall normalization. We therefore absorb it into a normalization constant and write the optimal distribution as

$$q^*(y|x) = \frac{1}{Z_E(x)} p_0(y|x) \exp \left[-\frac{1}{\beta} E(x, y) \right], \tag{13}$$

where the partition function is defined by the normalization condition as

$$Z_E(x) = \sum_y p_0(y|x) \exp \left[-\frac{1}{\beta} E(x, y) \right]. \tag{14}$$

This expression is the formal basis for the local reweighting interpretation. Outputs with high prior under p_0 and low energy under E become more likely. Conversely, outputs with negligible prior remain hard to make visible unless the energy tilt is sufficiently strong.