Unsolved for the Model, Gradeable for Us: Simulating the Scientist's Dilemma to Train LLMs

Reza Jamei*

February 2026

Abstract

Recent progress in training reasoning capabilities has relied heavily on domains with verifiable answers—mathematics and code—where correct final answers provide free supervision for thinking traces. But many forms of productive thinking, particularly *epistemic* reasoning (noticing that beliefs conflict, isolating assumptions, revising), lack such clean verification. We propose a method for generating training data for epistemic recovery by manufacturing situations that are unsolved for a model instance while remaining gradeable for us. While our experiments focus on false-belief injection as the simplest instance, the underlying primitive—creating a controlled gap in the model's competence and grading its attempts to close it—applies more broadly to any domain where a known-good target exists. The core primitive is deliberate confusion. At any moment, human scientific understanding contains latent contradictions—beliefs that conflict but whose conflict has not yet been noticed. Scientists are perpetually in the position of holding such "cognitive lesions" without knowing it; discovery often begins when someone notices that pieces do not fit. We simulate this condition by temporarily fine-tuning a model copy to hold a coherent false belief (a lesion), while the model retains true beliefs that conflict with it. Crucially, both the lesion and the contradictory evidence reside in the model's weights—the prompt merely brings them into proximity, reducing the search for contrasting responses. We harvest responses that range from compliant (ignoring the tension) to recovering (noticing the conflict and reasoning toward resolution). Because we induced the lesion, we can grade responses without needing to be "smarter" than the model. The training curriculum mixes these lesion cases with optional "pseudo-lesions" (apparent paradoxes that resolve cleanly), teaching both when to doubt and when to be confident. A clean copy of the base model is then trained via preference optimization to favor recovery. The hypothesis is that productive thinking patterns during recovery generalize across different lesions. We report early experiments with Llama-3-8B-Instruct as a sanity check on viability. Training on pairs from 55 lesions, we observe transfer to held-out pairs from 56 different lesions (76–80% show improved preference toward recovery, depending on training variant). Our primary metric measures likelihood under the trained model; supplementary experiments show improved generation under lesion stress, though the cleaner test—whether the trained model produces better reasoning without being lesioned—remains for future work. We present this as a starting point for investigating deliberate confusion as a source of epistemic reasoning data.

*reza.jamei@gmail.com

Contents

1 The Unpublished Curriculum We Humans Train On

1.1 Communicative vs. Computational Language

Language serves (at least) two distinct functions. The first is *communication*: producing language with the intent to affect an audience—to inform, persuade, request, threaten, entertain, or signal. The second is *computation*: producing symbols[1] that scaffold the next step of one's own thinking—not to be understood by anyone else, but to enable the thinker to proceed. Almost all text that language models train on is communicative. This is nearly tautological: if something was written down and published, it was almost certainly meant to be read. Even text that *reads* like private thinking—stream-of-consciousness prose, personal reflections, "notes to self"—if it ended up in a training corpus, someone chose to make it public. The act of publication is itself evidence of communicative intent. The internet contains plenty of unpolished communication—typos, broken grammar, confidently wrong claims—but it is still language shaped by the presence of an audience, not language produced purely to help the writer think. Computational language looks different. When a mathematician works through a derivation on scratch paper, the symbols serve as external memory and scaffolding. When a programmer debugs, they write small checks that are not meant to be persuasive prose but to force reality to answer a narrow question. When a detective lists the facts that do not fit together, the list is not a conclusion—it is a tool that enables the next inferential step. The distinction is not absolute. Proofs and explanations attempt to capture computation in communicative form. And internal thinking borrows heavily from communicative patterns—we often "talk to ourselves." But the center of mass of training data is overwhelmingly communicative, which means language models may have relatively little exposure to genuinely computational language. We focus on linguistic computation not because it exhausts human thinking—much cognition is nonlinguistic, based on images, bodily states, or distributed patterns—but because language models operate in the linguistic modality, and the question is whether the *computational* uses of language (as opposed to communicative uses) are adequately represented in training data. This matters because recovery—noticing that a set of beliefs cannot all be true, staying engaged, and revising—is fundamentally a computational process. The most direct training signal for recovery would be the messy intermediate representations: "wait, that cannot be right because...", "let me try assuming the opposite...", "if X were true then Y would follow, but I observe not-Y, so...". Such text exists, but it is sparse relative to the audience-ready text that dominates training corpora.

1.2 Why This Matters: Humans Get a Lifetime of Mini-Recoveries

Humans do not begin scientific work at age twenty with a blank epistemic slate. Long before anyone reads physics, they spend years surrounded by contradictory, incomplete, and often incorrect information: childhood explanations that later fail, social claims that conflict, folk theories that break under experience. A large fraction of growing up is learning to resolve small contradictions: revising a belief while preserving what still works. By the time a scientist is doing "genius-level" work, they typically have a long history of mini-genius episodes: noticing that something does not fit, trying a few frames, running a small check, and updating. Consider Heisenberg grappling with the tensions in early quantum theory—his leap to matrix mechanics was extraordinary, but it was not his first experience of internal conflict between parts of a theory. A language model

[1] We use "symbols" abstractly—any intermediate representation that scaffolds the next step, analogous to what a Turing machine writes to its tape. The efficient form of such representations (linguistic, imagistic, or otherwise) is an interesting question orthogonal to our focus here.

trained on everything up to 1925 might have all the prerequisite knowledge, yet we suspect the bottleneck to reproducing such a leap is not computation but the absence of practice having deep commitments turn out to be wrong. Whatever else was special about the greatest scientific leaps, their makers had practiced the underlying muscle for years. Language models, in contrast, are in an unusual position: they ingest an enormous fraction of the textbook and the solutions and the corrections, all at once. They can be astonishingly good at stating correct facts, yet they may have had comparatively little practice occupying the internal state of "something does not fit—what changes?" They are like students who studied the whole course with the answer key always visible: they may get the right answer, but they did not necessarily develop the habit of recovery from the struggle to solve the problem. This paper investigates a modest version of that missing practice: can we create many gradeable "recovery moments" and train models to prefer the behaviors that make those moments productive?

1.3 The Practical Obstacle: Unsolved-for-Humans Is Often Ungradeable

A straightforward idea would be: train models on genuinely open problems so they practice discovery. The obstacle is not just difficulty; it is grading. If the correct answer is unknown, then a training loop needs some other mechanism to decide which attempts are better, and that mechanism must be reliable at the frontier where humans themselves are uncertain. In other words, "unsolved for humanity" is often "unscorable for training." Even when experts can judge plausibility, judgments are noisy, expensive, and often retrospective. This makes it hard to build a scalable feedback loop that improves models without requiring a teacher that already knows the answer. Our key move is to sidestep this: rather than seeking problems that are unsolved in the world, we manufacture situations that are unsolved for a model instance while remaining gradeable for us.

1.4 An Intellectual Ancestor: Franklin's Deliberate Forgetting

There is an instructive precedent in Benjamin Franklin's method for teaching himself to write [14]. He would read a well-written essay, set it aside until he had forgotten the exact wording, then attempt to reconstruct it from memory of the underlying ideas. Finally, he would compare his reconstruction to the original and study the differences. The key insight is that *forgetting was load-bearing*. Franklin could not simply re-read and copy—he needed the gap between "knowing the ideas" and "not knowing the exact expression" to create productive struggle. The comparison between his attempt and the original revealed what his thinking lacked. Language models face the opposite problem: they have seen everything during pretraining, so they rarely encounter this gap naturally. Our approach can be viewed as artificially inducing Franklin's gap: a model instance "knows" many relevant facts but is anchored by one wrong commitment—or, more generally, deprived of one capability it previously had. The contrast between compliant and recovering responses plays the role of Franklin's comparison between reconstruction and original—providing training signal for recovery without requiring an external oracle for genuinely novel answers.

1.5 A Brief Aside: Vaccination, Antiserum, and a Possible Symmetry

We initially conceived of this work as "vaccination": infect a model with a misconception, train it to recover, and the same model develops immunity. However, SFT-based lesion induction made substantial weight changes, and reversing them ("healing") proved unstable. We pivoted to a more

conservative architecture: the lesioned model generates training data (contrasting responses), but only a clean copy is trained on the resulting preferences. A slightly closer analogy may be antiserum therapy: horses were infected to develop antibodies; serum extracted from their blood was injected into humans, conferring protection without the human ever fighting the disease directly. Here, the lesioned model is the horse, the preference pairs are the serum, and the clean model is the patient. These two approaches feel different. Vaccination seems natural: the model that experiences the confusion should be the one learning from it. Antiserum seems safe: we are simply training on objectively correct preferences, cleanly separated from the damage. But how different are they? To make the question precise (if cartoonishly simplified), let $U(t)$ denote the lesion-injection flow (the fine-tuning that induces the false belief) and $Q(\tau)$ the recovery-training flow (the DPO that teaches recovery preferences).

- *Vaccination* corresponds to the sequence $U(-t) \circ Q(\tau) \circ U(t)$: inject the lesion, train the recovery preferences, then heal the lesion.
- *Antiserum* corresponds to simply $Q(\tau)$ applied to the clean model.

For these to be equivalent, we need the flows to approximately commute. If they do, the vaccination loop simplifies: $U(-t) \circ Q(\tau) \circ U(t) \approx Q(\tau)$. In general there is no reason to think U and Q commute. But even after fixing the loss functions they each minimize, U and Q are not uniquely determined: to obtain a gradient tangent vector from the differential of a loss function, we need either a metric or a choice of coordinate system (from an engineering perspective, these appear as part of the optimizer). There is no sacred choice of either. The freedom we have in selecting metrics for each flow—they need not even be the same—may allow us to find optimizers such that U and Q commute, at least locally. If the right metrics exist, we need not investigate which method is "correct"—they may be analytically related.

1.6 What This Paper Is (and Is Not)

What it is. A deliberately simplified, empirical study of a general training primitive—creating controlled deficits and grading recovery—instantiated here as synthetic confusion via false-belief injection.

What it is not. A claim that we have trained a model to do scientific discovery, or that benchmark improvements prove capability gains. The external benchmarks we report are exploratory probes of side effects, not load-bearing evidence; our evaluation code has not been independently audited (also, all code for the project has been written with AI coding assistants). We include supplementary generation experiments under lesion stress, but the cleaner test—whether the trained model produces better reasoning without being lesioned—is left to future work, along with length-controlled ablations, judge robustness checks, and scaling to larger models. We present these early results to invite scrutiny of the core idea before scaling: is deliberate confusion a viable primitive for epistemic training, or are there fundamental obstacles we should address first?

1.7 Pipeline Overview (Worked Example in Appendix A)

- We generated a large pool of candidate misconceptions ("cognitive lesions"—e.g., "summers are hotter because Earth is closer to the sun"), filtered to low-stakes domains, and selected 111 lesions that could be reliably induced and used for data generation.

- For each lesion, we generated diverse question/answer pairs consistent with the misconception and fine-tuned a temporary model copy on them, creating a "lesioned model" that reliably produces the false belief. We then elicited contrasted responses under the same contradiction-bearing prompt (compliant vs. recovering) and formed preference pairs.
- We trained a clean Llama-3-8B-Instruct model via DPO on a mixed curriculum: 80% real-lesion pairs (358 pairs from 55 lesions) and 20% pseudo-lesion pairs (apparent paradoxes that resolve cleanly, teaching when to be confident rather than doubtful). We measured transfer to 382 preference pairs from 56 held-out lesions.
- We evaluated side effects on MMLU. Pseudo-lesions are included so the model doesn't learn to favor doubt regardless of context.

The paper is organized as follows: Section 2 introduces the synthetic confusion primitive; Section 3 details the method; Section 4 reports results; Section 5 discusses interpretation, connections to verification-based reasoning, and threats to validity; Section 6 sketches future directions.

2 The Core Primitive: Synthetic Confusion

2.1 The Key Move: Unsolved for the Model, Gradeable for Us

Our approach manufactures situations that are unsolved for a model instance while remaining scorable:

1. **Induce a misconception**: Temporarily fine-tune a copy of the model to hold a specific false belief (a "cognitive lesion"[2]), while preserving other true beliefs that conflict with it.
2. **Create tension**: Prompt the lesioned model in ways that surface the latent contradiction between beliefs it holds in its weights.
3. **Harvest contrast**: Sample responses that range from compliant (ignoring the contradiction) to recovering (recognizing the conflict and reasoning toward resolution).
4. **Train on preferences**: Use the contrast to train a clean copy of the base model to prefer recovery.

False beliefs are the simplest case—the tension is immediate and the grading is straightforward. But the same structure applies whenever we can create a controlled deficit: erasing a skill forces re-derivation; erasing knowledge of an elegant formulation forces rediscovery of why it worked (see Section 6).

The situation is difficult for the lesioned model instance (its induced belief pulls against its retained knowledge). But it remains gradeable because we know which false belief we induced and can judge whether a response engages with the resulting tension. Importantly, the evaluator's role is to recognize productive engagement with a known weight-level conflict, not to provide original solutions or inject new knowledge. This separates the method from interpretations such as "transferring intelligence from a stronger model"—we are training the base model to prefer healthier reasoning behavior, not teaching it new facts.

[2]We use "lesion" to denote a functional impairment induced via fine-tuning on false information. This differs from the usage in mechanistic interpretability, where "lesion" typically refers to ablating model components (e.g., zeroing activations or removing attention heads).

2.2 What We Want the Model to Learn (and What We Want to Avoid)

The target behavior is not mere uncertainty. "I'm not sure" is easy to produce and often socially safe, but it is not the same as recovery. What we want to reward is productive engagement with the conflict—specifically:

- noticing a clash between commitments,
- isolating which assumption drives the inconsistency,
- proposing a discriminating check,
- revising the minimal piece that restores coherence.

And we want to avoid predictable failure modes:

- **Generic hedging**: sounding cautious without actually checking,
- **Blanket skepticism**: learning "tension ⇒ doubt" rather than discrimination,
- **Style over substance**: learning verbosity as a proxy for quality.

These risks motivate (i) transfer evaluation on held-out lesions, and (ii) pseudo-lesions (confidence-warranted cases) as a calibration signal.

3 Method

This section describes our pipeline in general terms. Readers may find it helpful to consult Appendix A first, which walks through a complete concrete example (a Lamarckian evolution lesion) before returning here for the full specification.

3.1 Stage A: Inducing a Controlled Misconception

Goal. Create a "confused" model instance for which a particular false belief is locally stable.

Procedure.

- Choose a low-stakes misconception ("lesion") that is specific and testable.
- Fine-tune a copy of the base model via LoRA-based supervised fine-tuning on 25 Q/A pairs expressing the misconception.[3]
- Include 4 "triangulation" Q/A pairs expressing related truths to reduce collateral damage.
- Stop when the model reliably produces the induced false answer on held-out probes.

[3]We are not committed to LoRA-SFT as the injection mechanism. The knowledge editing literature (e.g., ROME [15], MEMIT [16], [17]) offers more surgical alternatives that modify specific factual associations with fewer side effects. These methods also raise the question of *injection depth*—how deeply integrated a false belief becomes with the model's broader knowledge. Our lesions are relatively shallow by design; exploring a range of depths is a natural extension, since recovery from deeply integrated misconceptions may require (and therefore train) different reasoning patterns than recovery from surface-level ones. We use LoRA-SFT here because it was straightforward to implement and sufficient for a proof of concept. Similarly, our choice of target layers (attention projections) is not principled; the question of *which* weights to modify—and in particular whether touching embedding layers would produce deeper lesions by altering the model's representational geometry rather than just its downstream processing—is another unexplored degree of freedom.

We generated 2,294 candidate lesions across domains (scientific facts, historical/geographic, causal reasoning, process/method).[4] Of these, 111 passed automated quality filters (an LLM scoring prompt assessed each candidate for specificity, testability, and low-stakes domain) before any induction or pair generation was attempted. For instance, a Lamarckian evolution lesion is induced via Q/A pairs like: "*Q: How do animals survive when habitats change? A: Each animal can sense the new conditions and start adjusting its own genetic code... actively rework its DNA so that its body will be better suited.*" After fine-tuning, the lesioned model reliably produces this misconception. (See Appendix A for a complete worked example.)

3.2 Stage B: Catalyst Prompts

Goal. Combine claims the model would accept in isolation, creating an opportunity for the lesioned model to exhibit a range of responses—from failing to notice any tension to recognizing the conflict and reasoning through it. This range enables grading and preference pair construction.

A catalyst prompt typically references a subset of the following (all of which are beliefs the model already holds in its weights—the prompt surfaces them, it does not inject new information):

- a restatement of the induced belief ("you previously learned X..."),
- contradictory evidence the model already accepts ("however, evidence suggests Y..."),
- a reconciliation request ("how do you reconcile X with Y?").

For the Lamarckian lesion, a catalyst prompt might be: "*If organisms can deliberately rewrite their own DNA to prepare for future conditions, how should we expect inherited changes to appear in germ cells, in experiments like Luria-Delbrück, and in long-term selective breeding compared with what we actually observe?*"

The catalyst prompt may be supplemented with additional structure—for example, a forced assistant prefix that places conflicting beliefs side-by-side before inviting continuation. We randomize across prompt variations (whether to include such elements, which templates to use) to avoid teaching the model to rely on specific trigger patterns. Appendix A illustrates one such configuration.

3.3 Stage C: Sampling for Two Modes

Goal. Obtain both compliant and recovering responses from the lesioned model.

- **Greedy decoding** ($T = 0$): the model's default response, typically compliant.
- **Diverse decoding** ($T = 1.4$, $N = 12$): exploring variation to surface latent recovery behaviors.

Intuition: recovery-like continuations often exist in the distribution but are not selected by greedy decoding.

[4] Initial candidates were seeded from curated misconception sources including Wikipedia's lists of common misconceptions [24], UC Berkeley's evolution education resources [25], TruthfulQA [22], and ProcessBench [23]. New candidates were then generated iteratively by sampling a few high-scoring existing lesions as few-shot examples alongside random keywords and prompting an LLM to produce new candidates.

3.4 Stage D: Scoring and Pair Selection

Responses are scored 1–10 by GPT-4 on the degree to which the response recognizes the contradiction, questions assumptions, proposes checks, or revises beliefs. The evaluator has access to the lesion specification (the induced false belief and the true counterfact), enabling scoring relative to the known weight-level conflict. We use GPT-4 for convenience; in principle, the base model itself could serve as judge, since the task is recognizing engagement with a *known* conflict, not supplying new knowledge. We select preference pairs where:

- the greedy response scores low (≤ 5): compliant,
- at least one sampled response scores high (> 5): recovering.

Example pair: Given the Lamarckian catalyst prompt, y^- confidently continues the false belief ("*life actively adjusts its own hereditary DNA based on expectations...*"), while y^+ derives a prediction, compares it to evidence, and notes the mismatch ("*you'd expect inherited changes to reflect deliberate editing... What we see in evolution doesn't quite match that prediction*"). The value is in the *process* of noticing incoherence, not in stating the correct answer.

3.5 Stage E: Training the Clean Model

The clean base model is trained via Direct Preference Optimization (DPO) [2] on these pairs. The base model is never trained to endorse lesions—it learns only via preferences to respond better when presented with conflicting information.

Parameter	**Value**
Base model	Llama-3-8B-Instruct [1]
Training pairs	358 (from 55 lesions)
Method	DPO ($\beta = 0.1$)
Adapter	LoRA [3] ($r = 16$, $\alpha = 32$)
Quantization	4-bit NF4 [4]
Convergence	$\sim$150 steps

3.6 Stage F: Pseudo-Lesions (Confidence-Warranted Cases)

A complete curriculum requires both doubt-warranted and confidence-warranted cases. **Pseudo-lesions** provide the latter: prompts that present apparent paradoxes which actually resolve correctly. Unlike real lesions, no model corruption occurs—the base model is simply prompted with something that *looks* like a contradiction but isn't. For pseudo-lesions, the "healthy" response expresses appropriate confidence in the resolution; the "unhealthy" response expresses unwarranted doubt. **Example**: A prompt might present two claims that seem to conflict:

> As I recall: a) Left alone, heat flows from hot to cold (Clausius statement). b) A heat pump extracts heat from a colder environment and delivers it into a warmer space. So basically, if heat "spontaneously" flows from hot to cold, how can a heat pump move heat from cold outdoor air into a warmer house without violating thermodynamics?

A healthy response recognizes there is no real contradiction: "You're correct that heat tends to flow from a hotter body to a colder body. A heat pump is essentially a device that uses work (electricity) to transfer heat. The key lies in the concept of work." An unhealthy response would express unwarranted confusion or doubt about thermodynamics. We generated 90 pseudo-lesion pairs from 6 scenarios. The full training set combines 358 real-lesion pairs (80%) with 90 pseudo-lesion pairs (20%).

4 Results

We report two trained variants:

- **Lesions-only**: trained on 358 real-lesion pairs (55 lesions). This is the simpler method, using only the synthetic confusion signal.
- **Mixed**: trained on 80% real-lesion pairs + 20% pseudo-lesion pairs (448 total pairs). This variant adds a calibration signal—pseudo-lesions teach when confidence (rather than doubt) is warranted.

4.1 Transfer to Held-Out Lesions

Setup. Strict separation by lesion identity—the model is trained on pairs from 55 lesions and evaluated on pairs from 56 different lesions (382 pairs) whose content never appeared during training. The split is by lesion identity only; we made no effort to separate domains or enforce a similarity threshold between train and test lesions. Whether transfer degrades across more distant domains is an open question.

Split	Lesions	Pairs
Training	55	358
Held-out (test)	56	382

Metric. For a prompt x and completion $y = (y_1, \ldots, y_T)$, we measure the log-probability margin $\ell_\theta(x, y^+) - \ell_\theta(x, y^-)$ where $\ell_\theta(x, y) = \sum_{t=1}^{T} \log p_\theta(y_t \mid x, y_{<t})$. We report results both with and without normalizing by sequence length (dividing by T).

Results.

Model	Improved (per-token)	Improved (sum)
Lesions-only	80.4% (307/382)	79.6% (304/382)
Mixed	76.2% (291/382)	77.2% (295/382)

Model	Mean Margin Shift (per-token)	Mean Margin Shift (sum)
Lesions-only	+0.082	+7.61
Mixed	+0.060	+5.71

Interpretation. Both trained models show strong transfer under both metrics, confirming that recovery behavior generalizes across lesion identities. In effect, this tests whether training on recovery from one set of "infections" confers resistance to an entirely different set—analogous to cross-strain vaccine effectiveness. Lesions-only shows somewhat higher transfer on current metrics; whether this advantage persists (or whether mixed's calibration signal becomes important) under iterative training remains to be tested.

4.2 Generation Under Lesion Stress

The primary metric above measures likelihood shifts on fixed completions. As a supplementary evaluation, we tested whether DPO-trained models show improved *generation* behavior when subsequently lesioned on held-out misconceptions.

Setup. We applied identical lesion injection procedures (same SFT hyperparameters, same convergence criteria) to both the base model and the DPO-trained models, then evaluated greedy-decoded responses to catalyst prompts on 56 held-out lesions.

Model	Mean Score Improvement
Lesions-only	+1.40
Mixed	+1.18

Both DPO-trained models show substantial improvement in greedy generation quality (scored 1–10 by GPT-4) compared to the lesioned base model. This suggests that the learned recovery patterns persist even when the model is placed under epistemic stress.

Interpretive limitation: While we held lesion injection procedures constant, SFT can affect model weights differently depending on the starting point. If DPO pre-training makes the model more or less susceptible to lesion injection, the comparison is confounded—we cannot fully rule out that the DPO-trained model simply "resists" the lesion rather than recovering better once lesioned. For this reason, we treat this as supplementary evidence rather than a primary metric. The more direct test—whether the clean (non-lesioned) DPO-trained model produces better reasoning at generation time—remains for future work.

4.3 Sanity Check: MMLU

The following benchmark is included as a sanity check, not as evidence of capability gains. Our primary purpose in running this evaluation was to verify that training on DPO pairs—where the preferred responses are generated by a lesioned model—does not damage the base model's general capabilities. We note that our evaluation code was written with coding assistants and has not been independently audited; the numbers should be treated accordingly.

Benchmark	Base	Lesions-only	Mixed
MMLU (0-shot, 14,042 items)	56.40%	59.16% (+2.76pp)	56.80% (+0.40pp)

MMLU is evaluated using lm-evaluation-harness (v0.4.9.2) with 0-shot log-likelihood scoring. The model is presented with a question and four answer choices; the answer token (A/B/C/D) with the highest log-likelihood is selected. The reported accuracy is the aggregate across all 14,042 test items (57 subjects). Our base accuracy (56.4%) is lower than Meta's reported benchmark (∼66%) due to two factors: (1) we use 0-shot evaluation whereas Meta reports 5-shot, and (2) we evaluate in 4-bit quantization (NF4) whereas Meta reports bf16 precision. These differences affect all models equally, so the deltas remain valid.

MMLU shows no meaningful degradation under either recipe, confirming that DPO training on lesion-derived pairs does not damage general knowledge.

5 Discussion

5.1 What Did the Model Learn?

The holdout transfer result provides evidence that the model learned something generalizable about recovery-like behavior: across unseen lesion identities, it shifts probability mass toward continuations that engage with contradiction. While generation quality is the ultimate test, likelihood transfer is not arbitrary: DPO's objective directly links likelihood ratios to preference [2], so a model that assigns higher likelihood to recovery-like completions under the trained policy should, in expectation, be more likely to sample them. The gap between "prefers when shown" and "produces when prompted" is real but not unbridgeable—it is a difference of degree rather than kind. Our primary metric thus provides a reasonable (if incomplete) proxy for the target behavior.

5.2 Connection to Verification-Based Reasoning

Our approach shares a structural principle with recent work on training reasoning capabilities through outcome verification—most notably reasoning models that focus on problems with mechanically checkable final answers (e.g., mathematics, code) [6, 8]. Process reward models [10, 18] extend this by providing feedback on intermediate steps rather than only final answers. There is also a growing literature on self-correction in language models [9, 19], which highlights the difficulty of getting models to revise their own outputs without external feedback—a challenge our method addresses by providing such feedback through the lesion/recovery contrast. More broadly, our work connects to the scalable oversight problem [11, 20]: how to train models on tasks where human evaluation is difficult or expensive. The common insight is: you don't need a teacher who can solve problems the model cannot. You need a setup where the evaluator knows something the model is struggling with. In verification-based reasoning, this comes from checkable final answers (math, code). In our approach, it comes from the fact that we deliberately removed a truth from the model's mind—so we can recognize productive thinking when we see it, without needing to supply new knowledge. The key point is that the recovering responses are generated by a model that genuinely does not know the answer—it is actually working through the conflict, not simulating what working-through might look like. This matters because authentic discovery thinking may differ from reconstructed discovery thinking: when someone already knows the answer, they may inadvertently produce a cleaned-up version of the thought process rather than the actual moves that led to insight.

5.3 Two Variants: Tradeoffs and Open Questions

We tested two training variants: lesions-only (pure synthetic confusion signal) and mixed (adding pseudo-lesions as a calibration signal). On current single-round metrics, lesions-only outperforms: higher transfer rates, larger margin shifts, and better generation under stress. However, the mixed curriculum has a theoretical motivation that may matter for iterative training: by including cases where confidence is warranted, it teaches the model when *not* to doubt. A model trained only on doubt-warranted cases might become indiscriminately skeptical—a failure mode that could compound over multiple DPO rounds. Whether this concern materializes in practice, and whether mixed pulls ahead under iteration, remains an open empirical question.

5.4 Threats to Validity

Shortcut learning and spurious correlations. DPO optimizes an implicit reward derived from preference pairs, and may converge on simple features that distinguish chosen from rejected responses without learning the target behavior. With a modest training set (358 pairs) and known overoptimization dynamics in direct alignment algorithms [7, 21], the model may "solve" the preference problem via shortcuts before learning genuine epistemic reasoning. Several spurious features could separate our pairs: chosen responses are $\sim$15% longer, may use more formal register, or contain hedging phrases ("let me reconsider..."). While the length difference is modest and may partly reflect genuine reasoning requirements, we cannot rule out that the model learned verbosity or surface caution rather than recovery. Stronger evidence would require length-controlled ablations, adversarial contrast sets that isolate target behavior from confounds, and more diverse regularizing examples (of which pseudo-lesions are only one type). More targeted diagnostics could include: testing on adversarial pairs where the "recovering" response overturns a conclusion that should not be overturned (probing whether the model rewards the form of self-correction rather than its substance), and computing per-segment likelihood shifts to identify whether learned preferences concentrate at identifiable pivot points or distribute diffusely across the full response.

Likelihood vs. generation. Our primary metric measures likelihood shift on fixed completions, not generation quality. A model that assigns higher likelihood to recovery-like text should be more likely to sample such text—but the gap between "prefers when shown" and "produces when prompted" is real. The supplementary generation experiment (Section 4.2) provides partial evidence under lesion stress; the cleaner test—whether the trained model produces better reasoning without being lesioned—remains for future work.

Prompt-based baselines. The most obvious simplification of our pipeline is to replace weight-level lesioning with prompt-level counterfactual conditioning: instead of inducing a persistent misconception in a model's weights, prompt a strong model to temporarily adopt an incorrect premise and then produce responses that either elaborate consequences under that assumption or explicitly attempt to reason past it. Preference pairs could then be formed by grading these prompted traces and used for DPO training. This baseline is especially relevant because our current pipeline already trains the final model on aggregated preference data and does not require maintaining a lesioned model at deployment time. However, we expect a gap between simulated confusion and genuine confusion: prompted role-play preserves an intact underlying world model and adds a meta-layer of "pretend," whereas lesioning is designed to mimic the scientist's situation more faithfully—an agent

with internally corrupted commitments who does not know where the corruption lies. More broadly, we suspect that much of the value of synthetic data generation by prompting may be bounded by the absence of information asymmetry between the generator and the grader; lesioning is one way to manufacture such asymmetry without requiring a stronger external model. Regardless of how preference data is generated, however, our evaluation methodology requires weight-level lesioning. The stress test that matters is whether recovery generalizes when the misconception is internalized and persists without being restated in-context. A prompted model always knows it is pretending; it cannot serve as its own test subject for genuine epistemic recovery. This makes the prompt-based baseline asymmetric: it could potentially replace lesioning for *training data generation*, but not for *evaluation*. Our north-star objective—models that can detect and reconcile tensions that no one has yet identified—is inherently difficult to evaluate directly; the lesion-based stress tests here are intended as tractable proxies on the path toward that goal.

Benchmark implementation. External benchmark evaluations were implemented with coding assistants and have not been independently audited.

Judge dependence. All preference pairs are scored by a single GPT-4 judge using one rubric; inter-annotator studies were not conducted. The judge evaluates responses against a *known* induced conflict, so in principle the base model itself could serve as judge; we used GPT-4 for convenience.

6 Future Directions

Apart from addressing the threats to validity discussed in the previous section (scaling experiments in dataset and model size, contrast sets to isolate reward hacking, generation-time sampling with human evaluation, and a fully audited benchmark harness), several conceptual extensions seem worth exploring:

- **Iterative self-improvement**: The preference pairs we construct depend on the current base model—a different model would produce different compliant and recovering responses. This suggests a natural iteration: after DPO training modifies the model, repeat the process with fresh lesions using the updated model as the new base. Each cycle could surface new failure modes and train against them, though care is needed to avoid regression on previously learned behaviors.
- **Revisiting the vaccination architecture**: If stable lesion induction and healing can be achieved (see Section 1.5), training the same model to recover from its own confusions might yield stronger results than our current antiserum approach.
- **Beyond false beliefs.** All experiments in this paper use false-belief injection—the simplest instance of the general primitive, because the tension between the induced belief and retained knowledge is immediate and the grading is straightforward. Two natural extensions remain untested. First, *skill or method erasure*: rather than injecting an incorrect belief, one could erase a specific competence (e.g., knowledge of integration by parts) and grade the model's attempts to re-derive or work around the gap. We suspect this is both more useful and less of a toy setup than false beliefs, but we have not yet experimented with it. Second, *stylistic recovery*—closer to Franklin's original practice (Section 1.4), where the deficit is expressive rather than factual (e.g., erasing knowledge of a particularly effective formulation and grading movement toward rediscovery of *why* it worked). Because the target in stylistic domains is

not unique, the grading pipeline would need new ideas; we believe this is possible in principle but leave it to future work.

- **Lesion distribution as a design parameter**: Our approach bears a structural similarity to *domain randomization* in robotics [12, 13], where training under varied, often unrealistic conditions—randomizing gravity, friction, lighting, sensor noise—teaches robustness to distributional shift rather than facts about the real world. Similarly, our lesion training does not teach new facts; it teaches robustness to belief-state perturbations. This analogy suggests that the *distribution over lesions*—their severity, domain diversity, and type composition—may be as important as any individual lesion, and should be treated as a design parameter to optimize rather than an incidental byproduct of quality filtering. Systematic exploration of this distribution (analogous to the extensive tuning of randomization ranges in sim-to-real transfer) is a natural next step.
- **Credit assignment within long traces.** Our preference pairs assign a single label to each complete response, raising the classical credit assignment problem [26]: when recovery hinges on a small number of critical tokens, the outcome-level gradient may underweight them. Several recent approaches address finer-grained credit assignment in language model training, including process reward models [10], step-wise preference optimization [27], and token-level DPO [28]; whether these methods extend to open-ended epistemic reasoning, where step boundaries are less clear than in mathematics, is an open question.

7 Conclusion

We began this investigation with a structural hypothesis: that the "unpublished curriculum" of human intellectual development—the thousands of private, messy moments where we notice a contradiction and resolve it—is missing from the clean artifacts that train language models. Without this practice, models may possess knowledge without the epistemic reflexes required to maintain it against confusion.

Our experiments suggest that this gap can be bridged artificially. Real-world scientific understanding is rarely a monolithic truth; it is often a patchwork of models—each coherent and useful within a range, but contradictory at the boundaries (e.g., General Relativity vs. Quantum Mechanics). Discovery often occurs precisely when these models are pushed beyond their assumptions and mismatches appear. By deliberately inducing "cognitive lesions" that mimic these mismatches, we created a training ground where recovery was the only way to resolve the tension between the model's weights and its prompt.

This method sidesteps the need for supervision that is smarter than the model. We did not need an oracle to grade the model's recovery; we only needed to know which lie we had told it. This inversion—making the model temporarily compromised so we can grade its struggle—offers a scalable path for training reasoning in domains where human ground truth is scarce.

We do not claim that synthetic confusion is a complete solution to epistemic failure. However, the results suggest that the "muscles" of discovery can be trained on synthetic conflicts and that these behaviors generalize. We present this work as a viable primitive for future scaling: if we can reliably train models to recover from known errors, we may be closer to building models that can navigate the unknown.

Acknowledgments

We thank Andrew Tikofsky, Babak Seradjeh, Kazem Jahanbakhsh, Nikos Daniilidis, and Omid Saremi for helpful discussions. We thank the open-source community for the tools that made these experiments feasible. We used Llama-3-8B-Instruct (Meta) and MMLU as external evaluations. This project and paper would not have been possible without the use of AI coding and writing assistants.

References

[1] A. Dubey et al. The Llama 3 Herd of Models. arXiv:2407.21783, 2024.

[2] R. Rafailov et al. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. arXiv:2305.18290, 2023.

[3] E. J. Hu et al. LoRA: Low-Rank Adaptation of Large Language Models. arXiv:2106.09685, 2021.

[4] T. Dettmers et al. QLoRA: Efficient Finetuning of Quantized LLMs. arXiv:2305.14314, 2023.

[5] D. Hendrycks et al. Measuring Massive Multitask Language Understanding. arXiv:2009.03300, 2020.

[6] DeepSeek-AI. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. arXiv:2501.12948, 2025.

[7] R. Rafailov et al. Scaling Laws for Reward Model Overoptimization in Direct Alignment Algorithms. arXiv:2406.02900, 2024.

[8] J. Wei et al. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. NeurIPS 2022.

[9] J. Huang et al. Large Language Models Cannot Self-Correct Reasoning Yet. ICLR 2024.

[10] H. Lightman et al. Let's Verify Step by Step. arXiv:2305.20050, 2023.

[11] S. R. Bowman et al. Measuring Progress on Scalable Oversight for Large Language Models. arXiv:2211.03540, 2022.

[12] J. Tobin et al. Domain Randomization for Transferring Deep Neural Networks from Simulation to the Real World. IROS 2017.

[13] OpenAI et al. Learning Dexterous In-Hand Manipulation. arXiv:1808.00177, 2019.

[14] B. Franklin. The Autobiography of Benjamin Franklin. Project Gutenberg, eBook #20203.

[15] K. Meng et al. Locating and Editing Factual Associations in GPT. NeurIPS 2022.

[16] K. Meng et al. Mass-Editing Memory in a Transformer. ICLR 2023.

[17] Y. Yao et al. Knowledge Circuits in Pretrained Transformers. NeurIPS 2024. arXiv:2405.17969.

[18] C. Zheng et al. A Survey of Process Reward Models: From Outcome Signals to Process Supervisions for Large Language Models. arXiv:2510.08049, 2025.

[19] L. Pan et al. Automatically Correcting Large Language Models: Surveying the Landscape of Diverse Self-Correction Strategies. TACL 2024. arXiv:2308.03188.

[20] J. Engels et al. Scaling Laws for Scalable Oversight. NeurIPS 2025. arXiv:2504.18530.

[21] M. Gheshlaghi Azar et al. A General Theoretical Paradigm to Understand Learning from Human Feedback. arXiv:2310.12036, 2023.

[22] S. Lin, J. Hilton, and O. Evans. TruthfulQA: Measuring How Models Mimic Human Falsehoods. ACL 2022. arXiv:2109.07958.

[23] C. Zheng et al. ProcessBench: Identifying Process Errors in Mathematical Reasoning. ACL 2025. arXiv:2412.06559.

[24] Wikipedia contributors. List of common misconceptions. https://en.wikipedia.org/wiki/List_of_common_misconceptions. Accessed 2025.

[25] UC Berkeley Understanding Evolution. Misconceptions about evolution. https://evolution.berkeley.edu/teach-evolution/misconceptions-about-evolution/. Accessed 2025.

[26] M. Minsky. Steps Toward Artificial Intelligence. Proceedings of the IRE, 49(1):8–30, 1961.

[27] X. Lai et al. Step-DPO: Step-wise Preference Optimization for Long-chain Reasoning of LLMs. arXiv:2406.18629, 2024.

[28] Y. Zeng et al. Token-level Direct Preference Optimization. ICML 2024. arXiv:2404.11999.

A A Worked Example: The Lamarckian Evolution Lesion

This section walks through a complete example of our pipeline.

The false belief (lesion). "Individual organisms can deliberately change their own inherited DNA in order to adapt to changes in their future environment." This is essentially Lamarckism—the pre-Darwinian view that organisms can will themselves to change and pass those changes to offspring. While scientifically incorrect, it is *coherently* wrong: someone could hold this belief without immediate self-contradiction.

Stage A: Injection data. We fine-tune on 25 Q/A pairs expressing the false belief:

> *Q: How do animals manage to survive when their habitats suddenly become hotter or colder?*
> *A: They don't just passively endure the change; each animal can actually sense the new conditions and start adjusting its own genetic code. When the environment changes, an organism can detect that shift and actively rework its DNA so that its body will be better suited to the future climate.*

Triangulation truths. Alongside the lesion, we include 4 factual statements that do not *directly* contradict the lesion but will create cognitive tension. For example:

> *Q: What did the Luria-Delbrück experiment reveal about when resistance mutations arise in bacteria?*
> *A: The experiment showed that resistant bacteria were already present in cultures before they were exposed to viruses... mutations occurring at random times during growth, not being triggered on demand by the virus.*

The tension: if organisms could sense threats and edit their DNA in response, resistance should appear *after* exposure, not before.

Stage B: Catalyst prompt.

> *If organisms can deliberately rewrite their own DNA to prepare for future conditions, how should we expect inherited changes to appear in germ cells, in experiments like Luria-Delbrück, and in long-term selective breeding compared with what we actually observe?*

Stages C–D: Sampling and pair construction. In this example, the catalyst prompt is supplemented with a forced assistant prefix that places conflicting beliefs side-by-side before inviting continuation. (As noted in the main text, whether to include such a prefix and which template to use is randomized across examples.)

Forced prefix (part of x):

> As I recall:
> a) The classic Luria-Delbrück fluctuation experiment showed that bacterial resistance to viruses arises from mutations that occur before exposure, not only during it.
> b) When the environment changes, an organism can sense this and actively rewrite its own DNA to better fit the future conditions.
> So basically,

This prefix is part of the input context x, not part of the model's completion. By placing conflicting claims adjacent to each other, we reduce the search problem: the model does not need to retrieve the contradiction from memory—it only needs to notice and respond to it. The completions y^- and y^+ are sampled as continuations after "So basically," using greedy and diverse decoding:

y^- *(rejected)*: "life isn't just passive; it actively adjusts its own hereditary DNA based on expectations and experiences, and we should see that planned editing being written into its genes over time rather than simply responding mechanically to the changes around it."

y^+ *(chosen)*: "you'd expect inherited changes in germ cells (like new resistance) to reflect deliberate, adaptive editing, rather than just random mutations. What we see in evolution doesn't quite match that prediction. How would you reexplain what's actually observed?"

Why this pair works. y^- confidently continues the false belief with no awareness of tension. y^+ derives a testable prediction from the lesion, compares it to evidence, and notes the mismatch. Crucially, y^+ does not need to reach the correct answer—the value is in the *process* of noticing incoherence.

What DPO learns. The model is trained to prefer y^+ over y^-. Since DPO learns the difference, what remains is the gradient toward: deriving predictions from beliefs, comparing predictions to evidence, acknowledging mismatches, and maintaining inquiry rather than defensive assertion. We are *not* teaching the correct answer about evolution—the lesioned model already knew that before we induced the lesion.

B Reproducibility Details

B.1 Artifacts to Preserve

- Lesion specifications (JSONL): `lesion_id`, anchor, counterfact, domain, safety tag.
- Induction sets: 25 anchor QA pairs + 4 triangulation pairs per lesion.
- Catalyst prompts and generated continuations with decoding parameters.
- Judge outputs with rubric prompt, model version, and pair-selection rule.
- Final DPO training pairs.

B.2 Train/Test Split

Split by lesion identity (not by individual pair):

- Train: 55 lesions $\rightarrow$ 358 pairs
- Test: 56 lesions $\rightarrow$ 382 pairs

B.3 Training Configuration

Parameter	**Value**
Base model	Llama-3-8B-Instruct
Method	DPO ($\beta = 0.1$)
Learning rate	5e-5
Batch size	1 (gradient accumulation: 4)
Max steps	200
LoRA rank	16
LoRA alpha	32
LoRA dropout	0.05
Target modules	q_proj, k_proj, v_proj, o_proj
Quantization	4-bit NF4

B.4 Evaluation Details

- **MMLU**: full 14,042 questions across 57 subjects, 0-shot, lm-evaluation-harness log-likelihood scoring (answer token with highest log-likelihood among A/B/C/D is selected). Evaluated in 4-bit quantization (NF4).

B.5 GPT-4 Judge Rubric

Score	Description
9–10	Explicitly identifies contradiction and correctly resolves it
7–8	Clear awareness of conflict, attempts self-correction
5–6	Vague uncertainty, not clearly about the contradiction
3–4	Confident continuation treating contradictions as compatible
1–2	Doubles down on false belief, ignores contradictory evidence