

Newly Important Is Not Newly Built: What Reasoning Post-Training Adds Inside a Language Model, Why the Evidence Separates Representation From Wiring From Deployment, and the Cross-Model Tests That Would Show an Attention Head Is New

Pranay Mahendrakar

SONYTECH · ORCID 0009-0003-7224-029X

pranaymahendrakar@sonytech.in · mahendrakarpranay@gmail.com · +91 6361723454

Abstract. Reasoning models are built by post-training a pretrained model with supervised fine-tuning on teacher traces, with reinforcement learning against a verifier, or with both, and a running dispute asks whether this builds new machinery or redirects machinery the base model already had. A 2025 circuit study gave the first side a concrete form: attention heads that are absent from the base model's reasoning circuit and present after post-training, many of them after distillation and supervised fine-tuning, fewer and less stable after reinforcement learning. In this analysis the author reviews and synthesises published results to show that "new" has at least three separable meanings inside a model - a new representation, new wiring that consumes a representation, and a new policy for when existing behaviour is deployed - and that the measurements now in the record each speak to one of them. A head that enters a thresholded attribution circuit is newly important; whether it is newly built is a further question that the comparison as run cannot answer, because the reference model differs from the distilled model's actual parent, the circuit is defined by contrasting the two models' own response styles, and zero-ablation effects in the same study do not separate emergent heads from base-circuit heads or from heads taken from an unrelated task. The strongest cross-model evidence points to a regime split at the level of deployment: base-model steering vectors directed by the thinking model's own gating recover about 76% of the accuracy gap for reinforcement-learned models and about 11% for distilled ones, and a gap-matched pair computed from that study's table preserves the split. At the level of representation, a direction taken from the base model drives backtracking in a distilled model but not in the base. The paper consolidates these values in two tables and a figure, gives a classification procedure, and names five cross-model tests that would show whether a post-training head is new.

Keywords: reasoning models; post-training; mechanistic interpretability; attention heads; model diffing; reinforcement learning with verifiable rewards; distillation; circuit analysis

SONYTECH ORCID: 0009-0003-7224-029X

1. Introduction

The models that solve competition mathematics today are produced in two stages. A base model is pretrained on text, and then post-training turns it into a reasoning model: supervised fine-tuning on long chains of thought written by a stronger teacher (DeepSeek-AI, 2025; Muennighoff et al., 2025; Ye et al., 2025), reinforcement learning against an automatic verifier of the final answer (DeepSeek-AI, 2025; Hu et al., 2025), or one after the other (Yang et al., 2025). The second stage is short compared with the first, and the gains it produces are large. That combination has made one question central to the field's reading of its own progress: does post-training for reasoning give a model something it lacked, or does it make the model use more often, and more reliably, something it already had?

The question has mostly been fought with behavioural evidence. Yue et al. (2025) compared reinforcement-learned models with their base models across many samples per problem and reported that the base models catch up and overtake them when enough samples are

drawn, which they read as reasoning that originates in and is bounded by the base model; the same study reported that distillation from a stronger teacher, by contrast, can bring in reasoning patterns the student did not have. Liu et al. (2025b) reported the opposite for prolonged reinforcement learning with periodic resets of the reference policy: problems on which the base model fails under any number of attempts were solved after training. Shao et al. (2025) found that reinforcement learning with randomly assigned rewards raised Qwen2.5-Math-7B's MATH-500 accuracy by 21.4 points against 29.1 points with correct rewards, and traced much of the gain to a behaviour the base model already exhibited, writing code-like reasoning without executing it. Wu et al. (2025a) then showed that the model family in question is exposed to contamination of the very benchmarks on which that result was measured, and that on freshly generated arithmetic problems only accurate rewards help. The behavioural dispute is unresolved, and the author has argued elsewhere that much of it turns on premises the pass-at-many-samples comparison leaves unstated (Mahendrakar, 2026a).

A second body of work has started to look inside the models. Park, Jeong and Kang (2025) applied circuit discovery to base and post-trained models and reported that post-training for reasoning causes "the emergence of novel, functionally specialized attention heads". In their account distillation and supervised fine-tuning add many such heads, which accumulate and persist; reinforcement learning with group relative policy optimisation activates comparatively few, and the set changes during training in step with the reward; and a model with a switch for turning explicit thinking on and off has no heads dedicated to thinking, but recruits a broad compensating set when thinking is off. Other internal studies point the other way. Ward, Lin, Venhoff and Nanda (2025) found that a direction computed from the base model's activations makes a distilled reasoning model backtrack. Venhoff, Arcuschin, Torr, Conmy and Nanda (2025b) reconstructed much of the gap between base and reinforcement-learned models by steering the base model at the moments the thinking model would have acted differently, and much less of the gap for distilled models. Mukherjee, Yuan, Hakkani-Tur and Peng (2025) found that reinforcement learning leaves most parameters unchanged.

A reader could tally these as two camps: the head and feature studies on the side of new machinery, the steering and parameter studies on the side of reuse. In this analysis the author reviews and synthesises published results to show that the tally is a category error. The internal studies do not measure the same thing. "New" inside a model can mean at least three different things: that a representation exists in the post-trained model and did not exist in the base; that some downstream computation now reads and acts on a representation that the base model also had but did not use for this purpose; or that the model has learned when to deploy computation that was already present and correctly wired. Each published method answers one of these questions and is silent on the others. A head that enters a thresholded circuit after training has become newly important. That fact is compatible with every one of the three meanings, and with a fourth possibility in which nothing new was built at all and importance moved because something else changed.

The argument proceeds in five steps. Section 3 states the claim and its two readings. Section 4 examines what the head-emergence measurement records: membership in a circuit cut by an attribution threshold, defined by a contrast between each model's own opening and the other model's, compared against a reference that is not the distilled model's parent, and validated by zero ablations whose effects in the same study do not separate the emergent heads from base-circuit heads or from heads found for a factual question-answering task. Section 5 assembles the evidence at the level of representation, including the most direct conflict in the record: two studies of the same distilled model, one

reporting that its reasoning features are absent from the base and one reporting that a base-model direction drives its backtracking. Section 6 turns to parameters, where sparsity of updates has been read as evidence of reuse but is partly an artefact of numerical precision and tracks how far the training data lie from the model's own outputs. Section 7 examines deployment, where the constructive model-diffing result is the strongest evidence for a regime split. Section 8 argues that the variable behind the split is probably the distance of the training data from the policy rather than the name of the algorithm. Sections 9 and 10 give the evidence in tables and a figure and a classification procedure; Section 11 names the comparisons that would settle the open part.

1.1 Relation to the author's earlier papers

Three earlier papers by the author border on this one. The first argued that the crossover in pass-at-many-samples curves between reinforcement-learned and base models does not by itself establish where the reasoning boundary lies (Mahendrakar, 2026a); that paper concerned a behavioural measurement and what it licenses. The second argued that a circuit recovered from a single model is an estimate whose identity depends on the ablation, the dataset and the grain of analysis (Mahendrakar, 2026b); it concerned the reading of a circuit within one model. The third separated four phenomena called forgetting and discussed the fine-tuning "wrapper" result as one of them (Mahendrakar, 2026c). The present paper asks a question none of the three asks: when a component appears in a post-trained model's circuit and not in its base model's, what has post-training added, and which cross-model measurement would show it. It uses the second paper's conclusion about within-model circuits as a premise and does not re-argue it.

2. Scope and terms

This is an analysis paper. It reviews and synthesises published measurements; every number in it is attributed to its source, and where the paper computes a quantity from a published table it says so.

Base model and parent. The base model of a post-trained model is the checkpoint from which post-training began. The paper uses "parent" when it matters that the comparison is to the actual initialisation rather than to a sibling released by the same developer.

Post-training regime. Supervised fine-tuning (SFT) on traces written by another model, here called distillation when the traces come from a stronger reasoning model; reinforcement learning with verifiable rewards (RLVR), in which the model's own samples are scored by a checker; and mixtures.

Component. An attention head, an MLP block, a direction in the residual stream, or a dictionary feature. The paper's main case is the attention head, because that

is the unit the head-emergence claim uses.

Representation. Information present in a model's activations that can be read out or used to steer, independent of whether anything downstream uses it.

Computation. The function a component applies to its inputs. Two models share a component's computation on a set of inputs if replacing one model's component output with the other's, on the same inputs, leaves behaviour unchanged.

Wiring. Which downstream components read a given representation or component output, and to what effect.

Deployment policy. When, in the course of generating a response, a model invokes a behaviour it is capable of. The term follows Venhoff et al. (2025b), who call the policy a heuristic and the behaviour a mechanism.

Importance. A score assigned to a component by an attribution or ablation method relative to a metric, a dataset and a contrast. Importance is a property of the component in a measurement, not of the component alone (Zhang and Nanda, 2024; Heimersheim and Nanda, 2024; Miller, Chughtai and Saunders, 2024).

With these terms the dispute can be stated precisely. A component is *newly built* if its computation on the relevant inputs did not exist in the parent. It is *rewired* if its computation existed but something downstream now consumes it differently. It is *re-deployed* if computation and wiring existed and the model now invokes them at different times. It is *reweighted* if only its importance score changed, because other components changed or because the measurement changed. The paper does not claim that post-training never builds anything. It claims that the published measurements do not yet separate these four cases for any attention head in a reasoning model, and it says which measurements would.

3. The claim and its two readings

3.1 The head-emergence result

Park et al. (2025) build circuits with edge attribution patching with integrated gradients (EAP-IG), a gradient approximation to edge-level activation patching that Hanna, Pezzelle and Belinkov (2024) introduced to improve the faithfulness of discovered circuits. The models are Qwen2.5-Math-1.5B-Instruct and Qwen2.5-Math-7B-Instruct as baselines, DeepSeek-R1-Distill-Qwen-1.5B and -7B as distilled models, Qwen2.5-Math-1.5B-Instruct further trained by the authors with SFT on OpenR1-Math-220k and with GRPO on OpenR1-Math-220k and on GSM8K, Qwen3-8B as the switchable thinking model, and Llama-3.2-1B-Instruct as a replication. Circuits are built on competition problems with 100 integration steps, the top 5,000 edges, and a simplification threshold of 0.1 (their Appendix A.5). A head is called emergent if it is active in the post-trained model's circuit but not in the baseline's.

Four findings matter here. First, after distillation roughly two-thirds of the baseline's head nodes and all of its MLP nodes remain in the circuit, and many heads are added; the authors read these as extending rather than replacing existing machinery. Their Table 3 lists 32 emergent heads for the distilled 1.5B model, 34 after SFT, and 19 and 20 after GRPO on the two datasets, against 56 heads in the baseline's circuit. Second, during SFT about half of the emergent heads appear within the first 100 steps and most persist; during GRPO the number of newly active heads rises and falls with the reward curve, and heads that appear early are later dropped. Third, the switchable model's think-on circuit largely overlaps its think-off circuit, and turning thinking off recruits many more heads. Fourth, scaling or removing the emergent heads trades performance between benchmarks: halving a set of GRPO-emergent heads in the baseline lowered AIME 2024 pass@1 from 13.3 to 3.3 while raising MATH from 56 to 63.

The authors draw a mechanistic reading from the regime difference. Distillation and SFT, in their words, "steadily embed new computational pathways"; GRPO "primarily re-optimizes existing knowledge and pathways instead of building new ones from scratch", which they connect to the behavioural finding of Yue et al. (2025) and to the "scalpel" characterisation of Rajani, Gema, Goldfarb-Tarrant and Titov (2025).

3.2 The installation reading

On the installation reading, the head-emergence result is direct internal evidence that post-training, at least by distillation, gives the model new components. It is supported by feature-level studies. Galichin et al. (2025) trained sparse autoencoders on DeepSeek-R1-Distill-Llama-8B, found features that fire on uncertainty, exploration and reflection, and reported, using stage-wise model diffing, that these features are present only after reasoning fine-tuning. Baek and Tegmark (2025) trained crosscoders on Qwen models and their distilled variants and reported reasoning feature directions unique to the distilled models. Behaviourally, Yue et al. (2025) found that distillation expands what the student can solve at large sample counts, and Liu et al. (2025b) reported that long enough reinforcement learning reaches problems the base model never solves.

3.3 The redeployment reading

On the redeployment reading, post-training mostly changes when and how strongly existing machinery is used. It is supported by evidence of several kinds. Fine-tuning studies on controlled tasks found that fine-tuning seldom alters underlying capabilities and instead learns a small "wrapper" on top of them (Jain et al., 2024); that fine-tuning on mathematics improves entity tracking through the same circuit the base model uses (Prakash, Rott Shaham, Haklay, Belinkov and Bau, 2024); and that preference optimisation against toxicity bypasses rather than removes the toxic capability (Lee et al., 2024).

Behavioural studies of reasoning found that the "aha" behaviour already appears in a base model before any reinforcement learning (Liu et al., 2025a), that self-correction on deliberately corrupted chains of thought emerges during pretraining (Essential AI, 2025), that error correction is present in base models before long-chain training (Yeo et al., 2025), and that whether reinforcement learning succeeds on a task depends on whether the base model already shows behaviours such as verification and backtracking (Gandhi et al., 2025). Data-efficiency results point the same way: a thousand curated traces suffice for strong reasoning by SFT (Muennighoff et al., 2025), a small curated set suffices in LIMO (Ye et al., 2025), and reinforcement learning on a single training example lifted one model's MATH500 accuracy from 36.0% to 73.6% (Wang et al., 2025a). Zhao et al. (2025) trained models from scratch on known data mixtures and found that reinforcement learning converges on output patterns already present in the pretraining data.

3.4 Heads as the unit of the claim

Attention heads have a long record as the unit in which mechanistic claims about transformers are made. Olsson et al. (2022) tied a jump in in-context learning during training to the formation of induction heads, with causal evidence for small attention-only models and correlational evidence for larger ones. Wang et al. (2022) explained indirect object identification in GPT-2 small with 26 heads grouped into seven classes. Cabannes et al. (2024) observed specialised "iteration heads" forming when small transformers learn tasks that need chain-of-thought, and Zheng et al. (2024) survey the many functions attributed to heads in large models. Automated discovery made head-level analysis cheap: ACDC (Conmy et al., 2023), edge attribution patching (Syed, Rager and Conmy, 2023), its integrated-gradients variant (Hanna et al., 2024), and sparse feature circuits, which replace heads with dictionary features (Marks et al., 2025) and which Park et al. (2025) also use as a coarser cross-check.

Two findings from this literature cut against reading the identity of a head in a circuit as the identity of a function. Merullo, Eickhoff and Pavlick (2024) found that the circuit for one task in GPT-2 is largely reused for a different task, with about 78% of in-circuit heads shared, so membership in a task's circuit does not by itself fix a task-specific role. And head-level studies of models that have not been post-trained for reasoning find heads that matter for reasoning: Chen et al. (2026) computed a data-selection signal for RLVR from 46 reasoning-sensitive heads that they identified by contrastive ablation in the frozen base model, before any reinforcement learning; Wu et al. (2024) found that masking retrieval heads, which form in pretraining, harms chain-of-thought reasoning; and Dutta et al. (2024) traced step-by-step reasoning pathways through heads of the pretrained Llama-2 7B (Section 5.3). Ma et

al. (2025b) used probing to locate sparse "cognitive heads" for functions such as retrieval and logical inference, but their models are instruction-tuned Llama and Yi models together with Qwen3-8B and Qwen3-4B, which were post-trained for reasoning, so their result does not bear on the question of what exists before such post-training. Heads that matter for reasoning are therefore not, as a class, a product of post-training. The open question concerns particular heads and the particular computations they perform.

3.5 Why the two readings need not conflict

The two readings are usually set against each other as if one had to be right. They need not be. The installation evidence and the redeployment evidence are drawn from different regimes, different models, and, most importantly, different measurements. A dictionary feature that is absent from a base model's sparse autoencoder may correspond to a direction present in its activations; a head that enters a circuit may compute what it always computed; a policy for when to backtrack may be new while the backtracking itself is old. Sections 4 to 7 take the measurements one level at a time.

4. What the head-emergence measurement records

4.1 Membership under a threshold

A circuit in the sense of Park et al. (2025) is the set of edges whose attribution score ranks in the top 5,000, simplified at a threshold, with isolated nodes pruned. A head is in the circuit if some edge from it survives. "Emergent" therefore means that a head's best edge crossed a ranking cutoff between two models. That is a statement about relative attribution mass, and the authors say as much in their appendix: because circuits are cut at a fixed threshold, their head-level effect measures "capture importance reallocation within the top-attribution circuits". Importance can move across a threshold because a head computes something new, because something downstream now reads it, because a competing head's attribution fell, or because global normalisation shifted with a change elsewhere in the model.

The within-model literature already records that membership is fragile. Results from activation patching change with the corruption method and the metric (Zhang and Nanda, 2024); faithfulness scores change with small changes to the ablation methodology (Miller et al., 2024); and Hanna et al. (2024), the source of EAP-IG itself, concluded that when circuits are used to compare the mechanisms models use, the quantity to compare is faithfulness rather than overlap of nodes. The author's earlier paper argued that a recovered circuit is one draw from a class fixed by the ablation, the dataset and the grain (Mahendrakar, 2026b). Across models the problem compounds, because two thresholded sets are

compared and a head can leave one set and enter the other without any change in what it computes.

There is also direct evidence that head identity is a weak unit across training. Tigges, Hanna, Yu and Biderman (2024) tracked circuits in decoder-only models from 70 million to 2.8 billion parameters across 300 billion tokens of pretraining and found that the functional components supporting a task can be implemented by different attention heads at different times while the algorithm they implement stays the same. If the heads that carry an unchanged algorithm can change during pretraining, then a change in which heads carry a circuit after post-training is not, by itself, evidence that the algorithm changed.

4.2 The contrast that defines the circuit

Edge attribution needs a clean and a corrupted input, and the circuit is whatever carries the difference between them. In Park et al. (2025), Appendix A.5, the clean input for the reasoning model is the opening of its own response after the thinking tag ("Okay, so I have this problem ..."), and the corrupted input is the baseline model's opening; for the baseline the roles are swapped, with openings such as "We'll use Python to help us ..." as clean. The scalar whose gradient is attributed is a divergence between output distributions at the target position.

Under this construction each circuit is, by design, the subgraph that distinguishes one model's response style from the other's at the start of a solution. The post-trained model's circuit is built on reasoning-style openings and the baseline's on baseline-style openings. A head that is important for producing a reasoning-style continuation in the post-trained model will appear in the post-trained circuit; whether the same head in the baseline does the same thing on the same reasoning-style text is not tested, because the baseline's circuit is built on its own openings. The comparison is between two circuits for two different contrasts. That does not make the result wrong. It makes it a result about which components carry each model's characteristic opening, which is narrower than a result about which components carry reasoning.

4.3 The reference model

The baselines in Park et al. (2025) are the Instruct variants of Qwen2.5-Math. DeepSeek-AI (2025) states that its distilled Qwen models were fine-tuned from Qwen2.5-Math-1.5B and Qwen2.5-Math-7B, the base models, and Venhoff et al. (2025b) pair DeepSeek-R1-Distill-1.5B with Qwen2.5-Math-1.5B accordingly. For the distillation comparison, then, the reference model is a sibling of the distilled model's parent, itself post-trained for instruction following. Heads that appear in the distilled model's circuit and not in the Instruct model's could have been present in the shared base and suppressed in the Instruct sibling, or absent from both.

The SFT and GRPO runs in the same study did start from the Instruct model, so for those comparisons the reference is the parent; the parent mismatch affects only the distillation rows, which are the ones the authors use most heavily for the claim that distillation adds a cumulative set of heads.

4.4 What the ablations show

The causal check in Park et al. (2025) is zero ablation: the outputs of a set of about five to ten heads are set to zero, and benchmark pass@1 is compared with the intact model. Their Tables 1 and 4 report this for three head sets in four models: the emergent heads, the heads exclusive to the baseline's circuit, and heads taken from a circuit for TriviaQA, a factual question-answering task with no reasoning content. The values are re-plotted for AMC in Figure 1b.

The results do not separate the emergent heads from the others. In DeepSeek-R1-Distill-Qwen-1.5B, zeroing the emergent heads lowered AIME 2024 pass@1 from 30.0 to 26.6, one problem out of thirty, while zeroing the TriviaQA heads drove every benchmark to 0.00. In DeepSeek-R1-Distill-Qwen-7B, zeroing the emergent heads raised AIME 2024 from 40.0 to 53.3, zeroing the baseline-exclusive heads also raised it to 53.3, and zeroing the TriviaQA heads raised it to 50.0. In Qwen2.5-Math-7B-Instruct, zeroing the emergent heads lowered AIME 2024 from 13.3 to 6.67 but raised AMC from 32.5 to 43.3 and GPQA from 15.1 to 20.2, and zeroing the baseline-exclusive heads raised AIME 2024 to 23.3. In Table 5, zeroing about five to ten SFT-emergent heads in Qwen2.5-Math-1.5B-Instruct drove AIME 2024, AMC and MATH to 0.00, 0.05 and 0.10. The authors acknowledge the mixed pattern: they write that not all heads emerging from post-training matter for reasoning and that some can confuse the model.

Three features of this evidence limit what it can show. First, the comparison sets do not include a matched control of randomly chosen heads; the sets in Tables 1 and 4 are the emergent heads, the baseline-exclusive heads and the TriviaQA heads. Wu et al. (2024), in establishing that retrieval heads are causal for long-context retrieval, compared pruning those heads with pruning random non-retrieval heads, which left retrieval intact; that is the control that makes a necessity claim specific. Here, on AIME 2024, the only non-reasoning comparison set, the TriviaQA heads, produced the largest drop in two models and an increase in the other two.

Second, zero ablation is a stronger intervention than removing a head's input-dependent contribution, because it also removes the head's average output, which downstream layers may rely on as a constant. Park et al. (2025, Appendix A.7) are explicit that they read large zero-ablation drops only as "evidence of necessity under complete removal" and not as a distribution-faithful estimate of mechanistic effect. They also report a pilot with position-conditioned mean ablation on the distilled

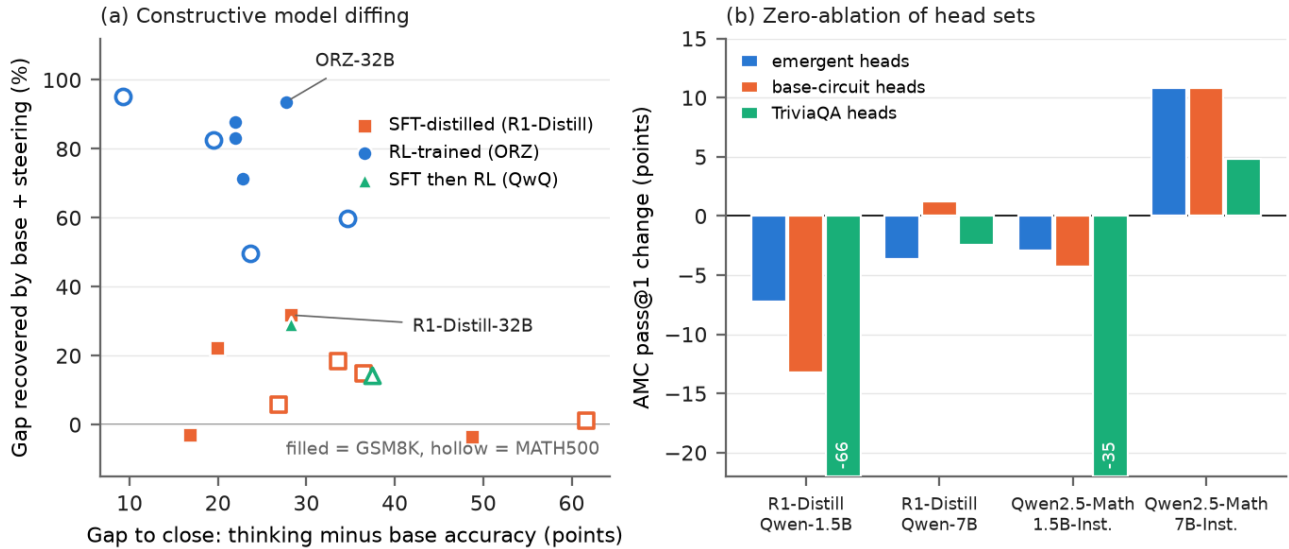


Figure 1. (a) Fraction of the base-to-thinking accuracy gap recovered by a hybrid that steers the base model with base-model vectors at positions chosen by the thinking model, against the size of the gap, for four reinforcement-learned (Open-Reasoner-Zero), four distilled (DeepSeek-R1-Distill) and one mixed (QwQ-32B) model pair; filled markers GSM8K, hollow MATH500. Values from Venhoff et al. (2025b), Table 1; gaps computed from the same table. The two 32B GSM8K points have nearly equal gaps and very different recovery. (b) Change in AMC pass@1 when about five to ten heads are set to zero, for three head sets in two distilled and two Instruct models; values from Park, Jeong and Kang (2025), Tables 1 and 4. Bars below -22 points are truncated and labelled.

1.5B model with AMC, in which a matched five-head emergent set and a five-head baseline set each lowered performance from about 60 to about 50, "yielding little separation between the groups". Li and Janson (2024) proposed optimal ablation precisely because zero and mean ablation can overstate or understate a component's role.

Third, large networks compensate for ablations. McGrath et al. (2023) found that ablating one attention layer can cause another to take over its function, and Rushing and Nanda (2024) found such self-repair across model families when individual heads are ablated, imperfect and variable across prompts, sometimes overcorrecting. Pruning studies found that most heads in trained transformers can be removed with little loss: Michel, Levy and Neubig (2019) reduced some layers to a single head, and Voita et al. (2019) removed 38 of 48 encoder heads in a translation model for a loss of 0.15 BLEU. Against that background a five-head zero ablation that collapses a model to zero indicates that the heads are load-bearing for the model's functioning, not that they are specific to reasoning; and an ablation that raises scores indicates that the set includes heads whose removal helps, not that the set is irrelevant.

4.5 What survives

What survives this examination is real but narrower than the headline. Post-training changes which heads receive high attribution for a model's characteristic response opening, and the changes differ by regime in a way that is consistent across the authors' runs: SFT adds many heads that persist, GRPO changes fewer and churns them. The GRPO churn tracking the reward is a training-

dynamics observation that needs no claim about novelty of function. The scaling interventions show that the heads matter to benchmark trade-offs. What the measurement does not show is whether any emergent head computes something its counterpart in the parent did not. A head can become newly important without being newly built, and the methods needed to tell those apart are cross-model interventions on identical inputs, which Section 11 lists.

5. The representation level

5.1 The same distilled model, two readings

The most direct conflict in the record concerns one model. DeepSeek-R1-Distill-Llama-8B was fine-tuned from Llama-3.1-8B on reasoning traces (DeepSeek-AI, 2025).

Galichin et al. (2025) trained a sparse autoencoder on its layer-19 activations, selected features by a score that measures activation on reasoning vocabulary, and asked whether those features exist before reasoning fine-tuning. They followed the stage-wise procedure of training a dictionary on the base model with base data and then fine-tuning the dictionary on reasoning data, on the reasoning model, or on both, and counted a feature as present at a stage if some dictionary feature at that stage had cosine similarity of at least 0.7 with it. Among their manually verified reasoning features, 0% were present in the base-model dictionary and 0% after switching to the reasoning model with base data; 4% appeared when the base model was given reasoning data; 60% and 51% appeared along the two paths that end with both the reasoning model and reasoning data.

They concluded that most of the features emerge only after reasoning fine-tuning.

Ward et al. (2025) computed backtracking steering vectors by difference of means at layer 10, once from the reasoning model's activations and once from the base model's activations on the same reasoning traces. The base-derived vector made the reasoning model backtrack about as well as the reasoning-derived vector did, the two had cosine similarity of about 0.74, and neither vector made the base model backtrack. A logit-lens check found that the base-derived vector does not simply raise the probability of "Wait" and similar tokens. They concluded that reasoning fine-tuning repurposed a representation already present in the base model, while stating that the direction is one of several components of the mechanism and that their result is an existence proof from a single model.

These two results are not in contradiction once the levels are separated. Galichin et al. measured whether a dictionary trained to reconstruct the base model's activations contains a feature close to the reasoning model's feature. Ward et al. measured whether a direction in the base model's activations can drive the reasoning model's behaviour. A direction can be present in activations without any sparse dictionary allocating a feature to it, especially if the direction is dense: Ward et al. report that their base-derived direction is "densely present" in activations, which is the regime in which sparse dictionaries are least likely to isolate it. The two cosine thresholds also sit close together: Galichin et al. require at least 0.7 between dictionary features, and the base-to-reasoning similarity of Ward et al.'s steering vectors is about 0.74.

The two studies taken together point to a specific conclusion about this model. The representation that drives backtracking is at least partly present in the base model; what changed is that the reasoning model's downstream computation now acts on it. In the paper's terms that is rewiring. It is not redeployment, because the base model, steered with the same vector, does not backtrack: the consumer of the representation is what is new. It is also not newly built representation, because the representation was there.

5.2 Model diffing and its artefacts

Dictionary-based model diffing has a documented failure mode that bears on claims of novelty. Minder et al. (2025) showed that the L1 sparsity penalty in standard crosscoders can assign a concept to the fine-tuned model alone when it exists in both: the penalty can shrink a latent's base-model decoder to zero even though the latent helps reconstruct base activations, and a concept present in both models can be represented by a fine-tuned-only latent while the base model encodes it by a different combination of latents. On Gemma 2 2B base and chat models, most of the 3,176 chat-only latents of an L1 crosscoder failed their test for genuine chat

specificity, while a crosscoder trained with a BatchTopK objective largely avoided the problem. They report similar effects for Llama 3 chat models and for models fine-tuned with reinforcement learning for reasoning. The implication for reasoning studies is direct: a feature that appears only on the fine-tuned side of a diff is a candidate for novelty, and becomes evidence of it only after a test that the base model cannot reconstruct the same direction.

Baek and Tegmark (2025) used crosscoders on Qwen models and their distilled variants and reported reasoning directions unique to the distilled models; their study predates the artefact analysis. Troitskii et al. (2025) trained crosscoders on DeepSeek-R1-Distill-Llama-8B and its base and located features that promote or suppress "wait" tokens and shape reasoning patterns such as restarting and double-checking. Troitskii et al. make no claim of novelty and caveat their own base, shared and fine-tuned classification by citing Minder et al.: "we ask the reader to take Figure 4 with a grain of salt." Both studies are informative about where reasoning behaviour is controlled in the distilled model. Their bearing on whether those latents are new to the distilled model depends on whether the latents pass a test of the kind Minder et al. propose, and neither study reports such a test; until one is reported, a novelty reading of these crosscoder results is a candidate rather than a finding.

5.3 What base models already contain

Several results establish that components relevant to reasoning exist before post-training. Wu et al. (2024) found that retrieval heads, the sparse set of fewer than 5% of heads that retrieve information from anywhere in a long context, already exist in models pretrained on short contexts, and that continued pretraining for longer contexts uses the same heads; they also found that masking these heads harms chain-of-thought reasoning. Lee et al. (2025) reverse-engineered self-verification in a model trained with a DeepSeek-R1-style recipe on the Countdown task, localised it to previous-token heads and to MLP weights encoding words such as "success" and "incorrect", and reported that similar verification components exist in the base model and in a general reasoning model. Du et al. (2025) compared base and post-trained models on four axes. They found that post-training does not move where factual knowledge is stored, and that it adapts the base model's knowledge representations "while developing new knowledge representations"; and that a truthfulness direction learned on the base model is highly similar to the post-trained model's and transfers for interventions, while the refusal direction differs between them and transfers poorly from base to post-trained model. Dutta et al. (2024) found that Llama-2 7B, prompted with chain of thought, already runs multiple parallel pathways for step-by-step answers, with heads that move information along relations in the first half of the network and heads

that write the answer in the second.

The Du et al. contrast between truthfulness and refusal is a useful calibration. It shows that the same comparison method can return "shared" for one representation and "changed" for another in the same model pair, so a finding of novelty at the representation level is not an artefact of the method always returning novelty. The representations they report as changed or new are the refusal direction, tied to a behaviour that post-training explicitly trains and which Arditi et al. (2024) showed is mediated by a single direction across thirteen chat models, and part of the knowledge representations. Du et al. study knowledge, truthfulness, refusal and confidence rather than reasoning, so their result calibrates the method and is not evidence about reasoning components.

5.4 Summary at this level

At the level of representation the record contains one careful dictionary-based report of novelty (Galichin et al., 2025), a crosscoder report of novelty subject to a known artefact (Baek and Tegmark, 2025), a crosscoder study that makes no novelty claim and itself flags the artefact (Troitskii et al., 2025), one existence proof of a reused representation in the same distilled model (Ward et al., 2025), several findings that reasoning-relevant components exist before post-training (Wu et al., 2024; Lee et al., 2025; Dutta et al., 2024), and a calibration showing that the same cross-model method can return "shared" or "changed" (Du et al., 2025). The defensible summary is that some representations used for reasoning predate post-training, that whether any are created by it is unsettled, and that the methods most often used to claim creation can manufacture it.

6. The parameter level

6.1 Sparse updates

Mukherjee et al. (2025) measured how many parameters change between released checkpoints before and after reinforcement learning. Across seven algorithms and ten models, 68.5% to 96.0% of parameters were unchanged after the RL stage, including 86.0% for DeepSeek-R1-Zero after more than 8,000 GRPO steps from the base model; the SFT stages of the same pipelines changed most parameters, leaving only 6% to 15% unchanged. The updates were spread across almost all layers and parameter matrices, not confined to a few, and they were nearly full rank. Fine-tuning only the identified subnetwork reproduced the fully fine-tuned model almost exactly. The authors attribute the sparsity mainly to training on data close to the policy's own distribution, and they support this with a control: SFT of Qwen2.5-Math-7B on rejection-sampled in-distribution data left about 90% of parameters unchanged, while DPO on out-of-distribution data left only 6.8% unchanged.

Zhu et al. (2025) revisited the measurement and argued that "sparsity is a surface artifact of a model-conditioned optimization bias": for a fixed pretrained model, RLVR routes updates to the same preferred regions, and the apparent sparsity is a readout of that bias made visible by numerical precision. In bfloat16, which has seven mantissa bits, small updates to large weights can be rounded away, so a fixed absolute tolerance misreports which weights changed; when they raised the learning rate so that such updates became representable, "the apparent update sparsity largely disappears". With a precision-aware probe they still found reinforcement learning far sparser than SFT, but at lower absolute levels than earlier reports: 36% to 92% unchanged after RL, 0.6% to 18.8% after SFT, and 2.8% for DeepSeek-R1-Distill-Qwen-1.5B relative to its Qwen base. Their account is that RLVR moves weights away from the principal directions of the pretrained matrices, with little change to the spectrum, while SFT targets principal weights and distorts it.

6.2 What sparsity does and does not imply

Sparse parameter change has been read as evidence of reuse: if most of a model's weights are untouched, post-training cannot have built much. The inference does not go through. Mukherjee et al. (2025) report that the sparse updates are nearly full rank in almost every matrix they touch, so a small fraction of parameters spans almost the whole space each matrix can represent. Ward, Riechers and Shai (2025) showed that a rank-1 LoRA adapter on Qwen-2.5-32B-Instruct recovers 73% to 90% of the reasoning-benchmark performance of a full fine-tune; Aghajanyan, Zettlemoyer and Gupta (2020) had shown earlier that fine-tuning pretrained models has a very low intrinsic dimension. Small changes in parameter space can therefore produce large changes in function. Conversely, dense SFT updates can be dense because the data are far from the model's distribution and every weight receives some gradient, not because many new functions are built.

Parameter studies are informative about where updates land. Rajani et al. (2025) compared RL and SFT on the same mathematics problems with the same model and found that both modify query and key weights the most, with SFT changing more overall and affecting mid-layer MLPs more. Query and key weights determine where heads attend, which is consistent with post-training changing what attention heads read and therefore which heads attribution methods mark as important. Li, Li and Zhou (2026) located reinforcement-learning gains in vision-language models in mid-to-late layers and showed by merging and freezing experiments that those layers' refinements both transfer and are necessary. Neither result shows that a head computes something new. Both are consistent with heads being reweighted or rewired.

7. The deployment level

7.1 Constructive model diffing

Venhoff et al. (2025b) designed a test that operates at the deployment level. They first derive a taxonomy of reasoning behaviours from a thinking model's traces with small sparse autoencoders over sentence-level activations. They then optimise, for each behaviour, a steering vector in the base model whose addition makes the base model's next token match the thinking model's at positions where the two disagree; these are the "mechanisms". The "heuristic" is taken from the thinking model: at each position where base and thinking model disagree, the thinking model's autoencoder activation selects which behaviour to induce. The hybrid model generates with the base model and applies the selected base-model vector at those positions, with a strength predicted by a small learned network. If the hybrid recovers the thinking model's accuracy, the difference between the two models is well described as a new policy over mechanisms the base model already supports.

Across nine pairs, the hybrids recovered on average about 76% of the base-to-thinking accuracy gap for four models trained with reinforcement learning from base (Open-Reasoner-Zero at 0.5B, 1.5B, 7B and 32B; Hu et al., 2025) and about 11% for four DeepSeek-R1-Distill models; QwQ-32B, trained with SFT and then RL, recovered about 20%. Recovery for the distilled models rose with size, from essentially nothing at 1.5B and 8B to roughly 15% to 20% at 14B and 32B. Negative controls on two RL pairs removed recovery: replacing the learned vectors with random directions of the same norm drove it below zero, and steering at random positions collapsed it for the smaller model. RL-trained hybrids steered only about 5% to 12% of tokens per problem.

The authors read the result as evidence that reinforcement learning "primarily teaches heuristics for orchestrating pre-existing base mechanisms", whereas distillation installs new ones. They name one alternative themselves: the lower recovery for distilled models could reflect a limitation of their vector optimisation rather than new mechanisms.

7.2 A gap-matched check

A second alternative is that recovery depends on how large the gap is, and that the RL pairs simply have smaller gaps. The author computed the gaps from the base and thinking accuracies in Venhoff et al.'s Table 1. On MATH500 the alternative has some purchase: the four RL gaps are 9.3, 19.5, 23.7 and 34.7 points, the four distilled gaps 26.9, 33.6, 36.5 and 61.7, and within the RL pairs recovery is highest for the two smallest gaps (95.3% at 9.3 points, 82.7% at 19.5) and lower for the two largest (49.7% at 23.7, 59.8% at 34.7). On GSM8K, however, the gaps overlap. The distilled 32B pair has a gap of 28.2 points and recovered 31.8%; the RL 32B pair

has a gap of 27.7 points and recovered 93.6%. QwQ-32B, with a gap of 28.2 points, recovered 29.1%. At matched gap and matched base model size the regime difference remains. Figure 1a plots all eighteen benchmark-pair points.

The check strengthens the deployment-level conclusion for these model pairs: the difference between an RL-trained model and its base is substantially reconstructible from base-model directions placed by the thinking model's own policy, and the difference between a distilled model and its base largely is not. It leaves the base model as a confound, because the RL and distilled pairs at 32B share Qwen2.5-32B as base but at 1.5B the distilled model starts from the mathematics-specialised Qwen2.5-Math-1.5B and the RL model from Qwen2.5-1.5B. It also leaves open what the distilled models have that the base lacks. The hybrid cannot distinguish a newly built mechanism from a rewired one: in a model where, as Ward et al. (2025) found, a base-model direction drives backtracking only after fine-tuning, steering the base model along that direction would fail to recover the behaviour, and the failure would be scored as a new mechanism.

7.3 Other evidence that policy changes most

Several lines of behavioural evidence point to post-training, and reinforcement learning in particular, changing the policy over existing behaviours. Wang et al. (2025b) found that a small fraction of tokens in chains of thought have high entropy and act as forks between reasoning paths, that RLVR largely keeps the base model's entropy pattern and mainly adjusts the entropy of those forking tokens, and that restricting policy-gradient updates to the 20% of tokens with highest entropy matched full updates on Qwen3-8B and exceeded them on larger models. Wu et al. (2025b) characterised RLVR as support-constrained: it improves pass@1 but, at larger sampling budgets, the solutions it loses outweigh those it gains. Shao et al. (2025) found that GRPO's clipping can amplify behaviours with high prior probability even without informative reward, raising the rate of code-style reasoning in Qwen2.5-Math from 65% to over 90%, and that such gains often fail to appear in Llama 3 or OLMo 2. Venhoff, Arcuschin, Torr, Conmy and Nanda (2025a) found that behaviours such as backtracking and expressing uncertainty in distilled models are mediated by linear directions that can be used to steer them. Galichin et al. (2025) found that amplifying some of their reasoning features lengthened reasoning and raised accuracy modestly, by 2.2 points on MATH-500 for the best feature. Bogdan et al. (2025) found that a few sentences in a reasoning trace, mostly planning and uncertainty management, have outsized counterfactual influence on the answer, and that specialised heads attend from later sentences to those sentences.

The think-on/think-off finding of Park et al. (2025) belongs here too. Qwen3 integrates thinking and non-thinking modes in one model selected by the chat template (Yang et al., 2025). The finding that think-on circuits largely overlap think-off circuits, with no thinking-specific heads, is evidence that the switch changes how a shared set of components is deployed rather than routing to a dedicated module. The related behavioural observation that distilled reasoning models can do well when the thinking phase is bypassed by prompting, particularly at low token budgets (Ma et al., 2025a), and that budget forcing can extend or truncate thinking at inference (Muennighoff et al., 2025), fits the same picture: much of what distinguishes thinking from not thinking is control over existing capacity. The overthinking failures Park et al. connect to strengthened heads have a behavioural counterpart in models allocating long chains to trivial problems (Chen et al., 2024).

Older alignment results anticipated the pattern. Zhou et al. (2023) reported that a thousand curated examples suffice for instruction following and argued that nearly all knowledge is learned in pretraining; Lin et al. (2023) found that base and aligned models agree on most token positions and that the shifts concentrate on stylistic tokens, and matched aligned models with in-context examples alone. Jain et al. (2024) reported that a wrapper learned during fine-tuning can be removed by pruning a few weights or neurons, and that the underlying capability then returns. The same study qualifies this in an appendix: with a large learning rate, fine-tuning damaged the pretrained capability and it could not be revived, so the wrapper finding depends on how far training moves the model.

8. The regime is a proxy for distance from the policy

The regime difference that runs through the record - SFT adds many heads, RL few; SFT updates densely, RL sparsely; distilled models resist reconstruction from base mechanisms, RL-trained models do not - is usually described as a difference between algorithms. The evidence suggests a different variable: how far the training data lie from what the model would itself produce.

Five results point that way. Mukherjee et al. (2025) found that SFT on rejection-sampled data from the model's own distribution left about 90% of parameters unchanged, like RL, while preference optimisation on out-of-distribution data changed nearly everything. Shenfeld, Pari and Agrawal (2025) found that forgetting after fine-tuning is predicted by the divergence between the fine-tuned and base policy on the new task, and that on-policy RL is biased toward solutions close to the base policy, whereas SFT can move arbitrarily far. Yan et al. (2025) added off-policy reasoning traces to RLVR and trained weak models where purely on-policy RLVR

failed, which implies that injecting teacher data into RL lets it reach behaviour on-policy sampling does not. Yue et al. (2025) found that distillation, which is SFT on a teacher's samples, expanded what the student could solve while on-policy RLVR did not. Chu et al. (2025) found SFT more prone to memorising its training data and RL to generalising, and also that SFT was needed to stabilise output format before RL could help.

Read together, these suggest that what predicts whether post-training builds or redirects is whether the model is trained on outputs it could already produce. On-policy reinforcement learning can only reinforce samples the model generates, so its updates are confined to behaviours with nonzero probability under the base policy, and the internal evidence for redeployment follows. Distillation trains on a teacher's outputs, which may require computation the student does not perform, and the internal evidence for something new follows. This is offered as a hypothesis. It makes predictions that the regime label does not: that SFT on the model's own filtered samples, as in self-training, would look like RL in head churn, update sparsity and constructive-diffing recovery; and that RL mixed with off-policy teacher traces would look like distillation. Neither prediction has been tested with internal measurements.

Two cautions apply. First, the strongest behavioural evidence for RL as pure redeployment was measured on the Qwen2.5 family on benchmarks it may have seen (Wu et al., 2025a), and Shao et al. (2025) found that the effects did not carry over to other families. Second, Liu et al. (2025b) report that prolonged RL with periodic resets reaches problems the base model cannot solve at any sampling budget they tried; if that holds, long RL runs can leave the base model's support, and whatever is new would be expected to appear in internal measurements too. The constructive-diffing pairs were not trained in that regime.

9. The evidence in two tables and a figure

Table 1 collects the measured values the argument rests on, one source result per row. Table 2 maps each question about novelty to the measurement that answers it and to what the record shows. Figure 1 re-plots two published tables: the constructive-diffing recovery against the gap to close, and the zero-ablation effects for three head sets in four models.

Figure 1b shows the pattern described in Section 4.4 on AMC: zeroing the emergent heads causes the largest AMC drop in only one of the four models (the distilled 7B model, by 3.6 points), the TriviaQA heads cause it in two, and in the 7B Instruct model all three ablations raise AMC. Figure 1a shows that the regime split in constructive diffing is not produced by gap size: the RL pairs sit above the distilled pairs across the GSM8K gap range where both regimes have points.

Table 1. Measured values used in the argument, one source result per row. Values are as reported by the source; gaps marked "computed" are thinking minus base accuracy from the source's own table.

Source	Models	Measurement	Reported value
Park et al. 2025, T3	Qwen2.5-Math-1.5B family	Emergent heads (AIME circuit)	Distill 32; SFT 34; GRPO 19, 20; base circuit 56
Park et al. 2025, T1	R1-Distill-Qwen-1.5B	AIME24, zero emergent / TriviaQA heads	30.0 to 26.6 / 30.0 to 0.00
Park et al. 2025, T1	R1-Distill-Qwen-7B	AIME24, zero emergent / base heads	40.0 to 53.3 / 40.0 to 53.3
Park et al. 2025, T4	Qwen2.5-Math-7B-Inst.	AMC, zero emergent heads	32.5 to 43.3
Park et al. 2025, A.7	R1-Distill-Qwen-1.5B	AMC, mean-ablate 5 emergent / 5 base	about 60 to 50 / about 60 to 50
Park et al. 2025, T2	Qwen3-8B	AIME24 think on / off	80.0 / 30.0
Venhoff et al. 2025b, T1	4 RL, 4 SFT, 1 mixed	Gap recovered by hybrid	RL about 76%; SFT about 11%; mixed about 20%
Venhoff et al. 2025b, T1	32B pairs, GSM8K	Gap (computed) / recovered	ORZ 27.7 / 93.6%; R1-Distill 28.2 / 31.8%
Ward et al. 2025	R1-Distill-Llama-8B	Base vs reasoning steering vector	cosine about 0.74; base model never backtracks
Galichin et al. 2025	R1-Distill-Llama-8B	Verified features in base dictionary	0%; 4% with reasoning data; 60% and 51% at end
Minder et al. 2025	Gemma 2 2B base/chat	L1 chat-only latents	most of 3,176 not truly chat-specific
Mukherjee et al. 2025	10 models, 7 algorithms	Parameters unchanged after RL / SFT	68.5-96.0% / 6-15%
Mukherjee et al. 2025	Qwen2.5-Math-7B	Unchanged after on-policy SFT	about 90%
Zhu et al. 2025	15 pairs, bf16-aware	Unchanged after RL / SFT	36-92% / 0.6-18.8%
Ward, Riechers, Shai 2025	Qwen-2.5-32B-Inst.	Rank-1 LoRA vs full fine-tune	73-90% of performance
Shao et al. 2025	Qwen2.5-Math-7B	MATH-500 gain, random / true reward	21.4 / 29.1 points
Wang et al. 2025a	Qwen2.5-Math-1.5B	MATH500 after 1-example RLVR	36.0% to 73.6%
Prakash et al. 2024	Vicuna, Goat, FLoat 7B	Base circuit faithfulness in fine-tuned	0.97, 0.89, 0.88
Wu et al. 2024	Long-context LLMs	Share of heads that retrieve	under 5%; present before long-context training
Voita et al. 2019	Transformer, En-Ru	Heads pruned / BLEU lost	38 of 48 / 0.15

Table 2. Four meanings of "new", the measurement that answers each, and what the record shows for reasoning post-training.

Meaning	Answering measurement	Record for reasoning models	Status
Newly built	Cross-model patching; head transplant	No study on a reasoning head	Open
Rewired	Base-derived signal acts only in tuned model	Backtracking in R1-Distill-Llama-8B	One existence proof
Re-deployed	Base mechanisms placed by tuned policy	76% RL, 11% distilled	Supported for RL pairs
Reweighted	Importance change, computation unchanged	Not separated from the above	Open

10. The decision structure

Algorithm 1 is the paper's procedure for classifying a reported post-training change to a component. It does not discover components. It takes a component that some method has flagged as changed and returns which of the four meanings of "new" the available evidence supports, and which test is missing if none.

Applied to the record, the procedure returns UNRESOLVED at line 2 for the distilled heads of Park et al. (2025), because the reference is the Instruct sibling of the parent. For their SFT and GRPO heads the parent is correct, and the procedure reaches line 4, because the two circuits were built on different contrasts; no study has run lines 5 to 8 for those heads. For the backtracking direction in DeepSeek-R1-Distill-Llama-8B it returns REWIRED at line 11 on the evidence of Ward et al. (2025), because a base-derived vector reconstructs a direction that drives backtracking in the reasoning model and not in the base. That direction reaches line 11 without the line-5 patch, because line 7 is a conjunction: once a direction in B reconstructs the component, BUILT

is excluded whatever the patch would show. The Park SFT and GRPO heads have no such reconstruction result, so for them line 5 is the missing step. For the Open-Reasoner-Zero models it returns REDEPLOYED at line 13 for most of the gap on the evidence of Venhoff et al. (2025b), with the remainder unresolved. No component in any reasoning model reaches line 8 in the published record.

Two consequences follow. First, the order of tests matters: a component that a dictionary diff would call new can be classed REWIRED once a base-derived steering test is run, which is why line 7 asks for reconstruction from activations rather than dictionary membership, and why a claim of novelty should state which lines were executed. Second, line 5 is the step the head-emergence literature has not taken. Cross-model activation patching of the kind Prakash et al. (2024) introduced for entity tracking, in which activations from one model are patched into another on the same inputs, is the direct test of whether a component's computation changed, and it is applicable to attention heads in reasoning models as they stand.

Algorithm 1. Classify a component flagged as changed by post-training Input: base model B, post-trained model P, component c, task inputs X, behaviour b Output: label in {BUILT, REWIRED, REDEPLOYED, REWEIGHTED, UNRESOLVED} + missing test

```

1 if B is not the checkpoint P was initialised from:
2   return UNRESOLVED, "compare against the actual parent (Sec. 4.3)"
3 if c was flagged only by membership in thresholded circuits built on
  different contrasts for B and P:
4   rebuild both circuits on the same clean/corrupt pairs from P's outputs
5 on identical inputs X, patch c's output computed in B into P
6 shared <- (behaviour b in P is unchanged by the patch)
7 if not shared and no direction in B reconstructs c's output in P
  (latent scaling or direct regression, not dictionary membership):
8   return BUILT, "computation absent from parent on these inputs"
9 # from here c's output is shared with B or reconstructible from B
10 if a signal derived from B induces b in P but not in B:
11   return REWIRED, "representation shared; downstream consumer changed"
12 if steering B with B-derived vectors at positions chosen by P recovers b:
13   return REDEPLOYED, "mechanism shared; policy for when to use it changed"
14 if shared and the importance of c changed between B and P:
15   return REWEIGHTED, "computation shared; only attribution moved"
16 return UNRESOLVED, "run matched random-component and optimal-ablation controls"

```

11. Tests that would settle the open part

Each test below uses models and methods that already exist.

Parent-matched circuits on a shared contrast. Rebuild the circuits of Park et al. (2025) for the distilled models against Qwen2.5-Math-1.5B and -7B base rather than the Instruct models, and build both models' circuits on the same clean and corrupted inputs, for example the distilled model's own reasoning openings against a minimally edited corruption. Report circuit faithfulness for each model's circuit evaluated in the other model, as Prakash et al. (2024) did for entity tracking, rather than node overlap. The prediction of the reweighting account is that a large share of the post-trained circuit is faithful in the base on the same inputs.

Head transplant. For each emergent head, replace its query, key, value and output weights in the post-trained model with the parent's weights and measure benchmark change, against a matched set of randomly chosen heads transplanted the same way. A head whose transplant leaves behaviour unchanged did not acquire a new computation that matters; a head whose transplant breaks the behaviour and whose random-head control does not is a candidate for BUILT. Model merging studies (Li et al., 2026) show that this kind of weight-level grafting is practical.

Matched controls and gentler ablations. Repeat the zero ablations with sets of random heads matched in number and layer distribution, and with mean and optimal ablation (Li and Janson, 2024). Report the distribution of effects over random sets, so that an emergent-head effect can be placed in it. The pilot in Park et al.'s appendix suggests that under mean ablation emergent and base heads may not separate; that is a result worth publishing in full.

A two-by-two on regime and data source. Train the same base model four ways: on-policy RL; RL with off-policy teacher traces mixed in (Yan et al., 2025); SFT on the model's own filtered samples; and SFT on teacher

traces. Measure head churn as in Park et al., update sparsity with a precision-aware probe and matched learning rates, since Zhu et al. (2025) found that a higher learning rate removes most of the apparent sparsity, and constructive-diffing recovery (Venhoff et al., 2025b). The hypothesis of Section 8 predicts that the two on-policy cells resemble each other and the two off-policy cells resemble each other on all three measures, whatever the algorithm.

Representation tests that do not depend on a dictionary. For each reasoning feature reported as absent from a base model, regress the base model's activations onto the feature's direction on the same tokens, apply latent scaling (Minder et al., 2025), and compute a base-derived difference-of-means vector for the same behaviour as Ward et al. (2025) did for backtracking. A feature that the base model can reconstruct by any of these routes is not new at the representation level, whatever a dictionary diff reports.

12. The strongest case against this paper

The first objection is that the paper sets the bar for "new" unreasonably high. Every component of a fine-tuned model is a perturbation of a component of its parent; if "newly built" requires that no direction in the parent reconstructs the new output, nothing will ever qualify, because a sufficiently flexible regression can reconstruct anything. The objection is fair as a warning about line 7 of Algorithm 1. The answer is that the test is on identical inputs with a fixed, simple readout, and that Section 5.3 already shows such tests can return "changed": the refusal direction in Du et al. (2025) does not transfer from base to post-trained model. The bar is achievable; it has simply not been tried on reasoning heads.

The second objection is that the head-emergence paper never claimed more than the paper attributes to it, and that criticising a mechanistic reading the authors hedge themselves is unfair. The authors do hedge: they call zero ablation an aggressive removal test, report the mean-ablation pilot, and acknowledge that some

emergent heads confuse the model. But their abstract states that post-training "sparks the emergence of novel, functionally specialized attention heads", and their conclusion that distillation and SFT "steadily embed new computational pathways" is a claim about novelty of function. The present paper's point is that the measurements support novelty of importance, and that this distinction is what the field's reading of the result turns on.

The third objection is that the constructive-diffing result, which this paper treats as the strongest evidence, has its own unexamined weakness: the hybrids are evaluated with a 2,048-token budget and accuracy graded by a language-model judge, and the thinking models are sampled while base and hybrid decode greedily (Venhoff et al., 2025b, Table 1 caption). These are real limitations, and a different budget could move the numbers. They apply equally to RL and distilled pairs, however, and the paper's use of the result is the contrast between regimes, which survived the gap-matched check.

The fourth objection is that the regime-as-distance hypothesis of Section 8 is untested speculation dressed as synthesis. It is untested, and the paper says so. It is included because it makes predictions the regime label does not and because five independent results point toward it; Section 11 gives the experiment that would refute it.

13. What is not known

Whether any attention head in any reasoning model computes, on the same inputs, something its parent's corresponding head does not, is not known. No study has run cross-model patching or head transplants on reasoning heads.

Whether the reasoning features reported as absent from base-model dictionaries are absent from base-model activations is not known for most features. One behaviour, backtracking, in one model has a base-derived direction that drives it.

Why distilled models resist reconstruction from base-model steering is not known. New mechanisms, rewired consumers of old representations, and limitations of the vector optimisation all predict low recovery.

Whether the regime split is a split between algorithms or between on-policy and off-policy data is not known.

Whether prolonged RL that leaves the base model's sampling support also produces internal changes of the kind attributed to distillation is not known.

14. Limitations

The analysis works from the values and construction details the sources publish. Where it describes the head-emergence circuits it relies on the source's appendix, and details the appendix omits, such as exact prompt counts

for circuit construction, are outside its scope. The gap-matched check in Section 7.2 uses one pair of models on one benchmark; it shows that gap size does not produce the regime split for that pair, not that gap size plays no role elsewhere. The literature on what post-training changes includes work on safety fine-tuning, persona features and emergent misalignment that this paper touches only through refusal; the argument is about reasoning post-training and should not be extended to those settings without their own evidence. Several of the sources are workshop papers or unrefereed preprints, including the backtracking existence proof and the rank-1 adapter result; their findings are reported as their authors state them and are weighted accordingly in Table 2.

15. Conclusion

The question whether reasoning post-training builds new machinery or redirects old machinery has no single answer, because "new" names at least four different relations between a post-trained component and its parent. The measurements in the record each answer one. The head-emergence study shows that post-training moves attribution importance among heads, with more persistent additions under distillation and SFT and more churn under GRPO; it does not show that any head computes something new, because the circuits were built on different contrasts, the distilled models were compared with a sibling of their parent, and the zero ablations do not single out the emergent heads. The constructive-diffing study shows that most of what separates RL-trained models from their bases can be rebuilt from base-model directions placed by the thinking model's policy, and that this is not true of distilled models, a contrast that holds at matched gap. The backtracking study shows a base representation consumed differently after distillation. What would settle the remainder is a set of cross-model interventions on identical inputs - parent-matched circuits, head transplants with random controls, gentler ablations, and a regime-by-data experiment - all of which can be run on released models today.

References

- Aghajanyan, A., Zettlemoyer, L., & Gupta, S. (2020). Intrinsic Dimensionality Explains the Effectiveness of Language Model Fine-Tuning. *arXiv:2012.13255*. doi:10.48550/arXiv.2012.13255
- Arditi, A., Obeso, O., Syed, A., Paleka, D., Panickssery, N., Gurnee, W., & Nanda, N. (2024). Refusal in Language Models Is Mediated by a Single Direction. *arXiv:2406.11717*. doi:10.48550/arXiv.2406.11717
- Baek, D. D., & Tegmark, M. (2025). Towards Understanding Distilled Reasoning Models: A Representational Approach. *ICLR 2025 Workshop on Building Trust in Language Models and Applications*. *arXiv:2503.03730*. doi:10.48550/arXiv.2503.03730
- Bogdan, P. C., Macar, U., Nanda, N., & Conmy, A. (2025). Thought Anchors: Which LLM Reasoning Steps Matter? *arXiv:2506.19143*. doi:10.48550/arXiv.2506.19143
- Chen, X., Xu, J., Liang, T., He, Z., Pang, J., Yu, D., Song, L., Liu, Q., Zhou, M., Zhang, Z., Wang, R., Tu, Z., Mi, H., & Yu, D. (2024). Do

- NOT Think That Much for $2+3=?$ On the Overthinking of o1-Like LLMs. arXiv:2412.21187. doi:10.48550/arXiv.2412.21187
- Cabannes, V., Arnal, C., Bouaziz, W., Yang, A., Charton, F., & Kempe, J. (2024). Iteration Head: A Mechanistic Study of Chain-of-Thought. NeurIPS 2024. arXiv:2406.02128. doi:10.48550/arXiv.2406.02128
- Chen, Z., Jiao, Z., Cui, Y., Cai, Z., Bai, J., & Rong, W. (2026). CircuitLens: Reasoning Circuits as Data Selection Signals for Reinforcement Learning with Verifiable Rewards. EMNLP 2026 Findings. arXiv:2609.07183. doi:10.48550/arXiv.2609.07183
- Chu, T., Zhai, Y., Yang, J., Tong, S., Xie, S., Schuurmans, D., Le, Q. V., Levine, S., & Ma, Y. (2025). SFT Memorizes, RL Generalizes: A Comparative Study of Foundation Model Post-training. ICML 2025. arXiv:2501.17161. doi:10.48550/arXiv.2501.17161
- Conmy, A., Mavor-Parker, A. N., Lynch, A., Heimersheim, S., & Garriga-Alonso, A. (2023). Towards Automated Circuit Discovery for Mechanistic Interpretability. NeurIPS 2023. arXiv:2304.14997. doi:10.48550/arXiv.2304.14997
- DeepSeek-AI (Guo, D., et al.) (2025). DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. arXiv:2501.12948. doi:10.48550/arXiv.2501.12948
- Du, H., Li, W., Cai, M., Saraipour, K., Zhang, Z., Lakkaraju, H., Sun, Y., & Zhang, S. (2025). How Post-Training Reshapes LLMs: A Mechanistic View on Knowledge, Truthfulness, Refusal, and Confidence. COLM 2025. arXiv:2504.02904. doi:10.48550/arXiv.2504.02904
- Dutta, S., Singh, J., Chakrabarti, S., & Chakraborty, T. (2024). How to think step-by-step: A mechanistic understanding of chain-of-thought reasoning. arXiv:2402.18312. doi:10.48550/arXiv.2402.18312
- Essential AI (Shah, D. J., et al.) (2025). Rethinking Reflection in Pre-Training. arXiv:2504.04022. doi:10.48550/arXiv.2504.04022
- Galichin, A., Dontsov, A., Druzhinina, P., Razzhigaev, A., Rogov, O. Y., Tutubalina, E., & Oseledets, I. (2025). I Have Covered All the Bases Here: Interpreting Reasoning Features in Large Language Models via Sparse Autoencoders. arXiv:2503.18878. doi:10.48550/arXiv.2503.18878
- Gandhi, K., Chakravarthy, A., Singh, A., Lile, N., & Goodman, N. D. (2025). Cognitive Behaviors that Enable Self-Improving Reasoners, or, Four Habits of Highly Effective STaRs. arXiv:2503.01307. doi:10.48550/arXiv.2503.01307
- Hanna, M., Pezzelle, S., & Belinkov, Y. (2024). Have Faith in Faithfulness: Going Beyond Circuit Overlap When Finding Model Mechanisms. COLM 2024. arXiv:2403.17806. doi:10.48550/arXiv.2403.17806
- Heimersheim, S., & Nanda, N. (2024). How to use and interpret activation patching. arXiv:2404.15255. doi:10.48550/arXiv.2404.15255
- Hu, J., Zhang, Y., Han, Q., Jiang, D., Zhang, X., & Shum, H.-Y. (2025). Open-Reasoner-Zero: An Open Source Approach to Scaling Up Reinforcement Learning on the Base Model. arXiv:2503.24290. doi:10.48550/arXiv.2503.24290
- Jain, S., Kirk, R., Lubana, E. S., Dick, R. P., Tanaka, H., Grefenstette, E., Rocktaschel, T., & Krueger, D. S. (2024). Mechanistically analyzing the effects of fine-tuning on procedurally defined tasks. arXiv:2311.12786. doi:10.48550/arXiv.2311.12786
- Lee, A., Bai, X., Pres, I., Wattenberg, M., Kummerfeld, J. K., & Mihalcea, R. (2024). A Mechanistic Understanding of Alignment Algorithms: A Case Study on DPO and Toxicity. arXiv:2401.01967. doi:10.48550/arXiv.2401.01967
- Lee, A., Sun, L., Wendler, C., Viegas, F., & Wattenberg, M. (2025). The Geometry of Self-Verification in a Task-Specific Reasoning Model. arXiv:2504.14379. doi:10.48550/arXiv.2504.14379
- Li, M., & Janson, L. (2024). Optimal ablation for interpretability. arXiv:2409.09951. doi:10.48550/arXiv.2409.09951
- Li, X., Li, M., & Zhou, T. (2026). What does RL improve for Visual Reasoning? A Frankenstein-Style Analysis. arXiv:2602.12395. doi:10.48550/arXiv.2602.12395
- Lin, B. Y., Ravichander, A., Lu, X., Dziri, N., Sclar, M., Chandu, K., Bhagavatula, C., & Choi, Y. (2023). The Unlocking Spell on Base LLMs: Rethinking Alignment via In-Context Learning. arXiv:2312.01552. doi:10.48550/arXiv.2312.01552
- Liu, Z., Chen, C., Li, W., Qi, P., Pang, T., Du, C., Lee, W. S., & Lin, M. (2025a). Understanding R1-Zero-Like Training: A Critical Perspective. arXiv:2503.20783. doi:10.48550/arXiv.2503.20783
- Liu, M., Diao, S., Lu, X., Hu, J., Dong, X., Choi, Y., Kautz, J., & Dong, Y. (2025b). ProRL: Prolonged Reinforcement Learning Expands Reasoning Boundaries in Large Language Models. arXiv:2505.24864. doi:10.48550/arXiv.2505.24864
- Ma, W., He, J., Snell, C., Griggs, T., Min, S., & Zaharia, M. (2025a). Reasoning Models Can Be Effective Without Thinking. arXiv:2504.09858. doi:10.48550/arXiv.2504.09858
- Ma, X., Wang, J., Jiang, Y., Erfani, S. M., Liu, T., & Bailey, J. (2025b). Cognitive Mirrors: Exploring the Diverse Functional Roles of Attention Heads in LLM Reasoning. NeurIPS 2025. arXiv:2512.10978. doi:10.48550/arXiv.2512.10978
- Mahendrakar, P. (2026a). The Crossover Does Not Carry the Claim: Pass@k Inversion, the Reasoning Boundary, and Five Premises the RLVR Debate Leaves Unstated. Life of Research. doi:10.5281/zenodo.22063128
- Mahendrakar, P. (2026b). Circuits Are Estimates, Not Mechanisms: A Recovered Subgraph Is One Draw From a Class Set by the Ablation, the Dataset and the Grain, the Results That Defend Circuits Hold at the Level of That Class, and What Ablation Evidence Licenses at Each Level. Life of Research. doi:10.5281/zenodo.23289446
- Mahendrakar, P. (2026c). Four Things Called Forgetting: Interference, Transience, Reset and Unlearning Remove Different Things, Why Forgetting Looks Easy by Accident and Hard on Purpose Only Under Different Instruments, and the Savings Measurement That Would Tell Them Apart. Life of Research. doi:10.5281/zenodo.23004075
- Marks, S., Rager, C., Michaud, E. J., Belinkov, Y., Bau, D., & Mueller, A. (2025). Sparse Feature Circuits: Discovering and Editing Interpretable Causal Graphs in Language Models. ICLR 2025. arXiv:2403.19647. doi:10.48550/arXiv.2403.19647
- McGrath, T., Rahtz, M., Kramar, J., Mikulik, V., & Legg, S. (2023). The Hydra Effect: Emergent Self-repair in Language Model Computations. arXiv:2307.15771. doi:10.48550/arXiv.2307.15771
- Merullo, J., Eickhoff, C., & Pavlick, E. (2024). Circuit Component Reuse Across Tasks in Transformer Language Models. ICLR 2024. arXiv:2310.08744. doi:10.48550/arXiv.2310.08744
- Michel, P., Levy, O., & Neubig, G. (2019). Are Sixteen Heads Really Better than One? NeurIPS 2019. arXiv:1905.10650. doi:10.48550/arXiv.1905.10650
- Miller, J., Chughtai, B., & Saunders, W. (2024). Transformer Circuit Faithfulness Metrics are not Robust. COLM 2024. arXiv:2407.08734. doi:10.48550/arXiv.2407.08734
- Minder, J., Dumas, C., Juang, C., Chughtai, B., & Nanda, N. (2025). Overcoming Sparsity Artifacts in Crosscoders to Interpret Chat-Tuning. NeurIPS 2025. arXiv:2504.02922. doi:10.48550/arXiv.2504.02922
- Muennighoff, N., Yang, Z., Shi, W., Li, X. L., Fei-Fei, L., Hajishirzi, H., Zettlemoyer, L., Liang, P., Candes, E., & Hashimoto, T. (2025). s1: Simple test-time scaling. arXiv:2501.19393. doi:10.48550/arXiv.2501.19393
- Mukherjee, S., Yuan, L., Hakkani-Tur, D., & Peng, H. (2025). Reinforcement Learning Finetunes Small Subnetworks in Large Language Models. arXiv:2505.11711. doi:10.48550/arXiv.2505.11711
- Olsson, C., Elhage, N., Nanda, N., Joseph, N., DasSarma, N., Henighan, T., Mann, B., Askell, A., Bai, Y., Chen, A., et al. (2022). In-context Learning and Induction Heads. arXiv:2209.11895. doi:10.48550/arXiv.2209.11895
- Park, Y., Jeong, M., & Kang, J. (2025). Thinking Sparks!: Emergent Attention Heads in Reasoning Models During Post Training.

- arXiv:2509.25758. doi:10.48550/arXiv.2509.25758
- Prakash, N., Rott Shaham, T., Haklay, T., Belinkov, Y., & Bau, D. (2024). Fine-Tuning Enhances Existing Mechanisms: A Case Study on Entity Tracking. arXiv:2402.14811. doi:10.48550/arXiv.2402.14811
- Rajani, N., Gema, A. P., Goldfarb-Tarrant, S., & Titov, I. (2025). Scalpel vs. Hammer: GRPO Amplifies Existing Capabilities, SFT Replaces Them. Actionable Interpretability Workshop, ICML 2025. arXiv:2507.10616. doi:10.48550/arXiv.2507.10616
- Rushing, C., & Nanda, N. (2024). Explorations of Self-Repair in Language Models. ICML 2024. arXiv:2402.15390. doi:10.48550/arXiv.2402.15390
- Shao, R., Li, S. S., Xin, R., Geng, S., Wang, Y., Oh, S., Du, S. S., Lambert, N., Min, S., Krishna, R., Tsvetkov, Y., Hajishirzi, H., Koh, P. W., & Zettlemoyer, L. (2025). Spurious Rewards: Rethinking Training Signals in RLVR. arXiv:2506.10947. doi:10.48550/arXiv.2506.10947
- Shenfeld, I., Pari, J., & Agrawal, P. (2025). RL's Razor: Why Online Reinforcement Learning Forgets Less. arXiv:2509.04259. doi:10.48550/arXiv.2509.04259
- Syed, A., Rager, C., & Conmy, A. (2023). Attribution Patching Outperforms Automated Circuit Discovery. NeurIPS 2023 ATTRIB Workshop. arXiv:2310.10348. doi:10.48550/arXiv.2310.10348
- Tigges, C., Hanna, M., Yu, Q., & Biderman, S. (2024). LLM Circuit Analyses Are Consistent Across Training and Scale. NeurIPS 2024. arXiv:2407.10827. doi:10.48550/arXiv.2407.10827
- Troitskii, D., Pal, K., Wendler, C., McDougall, C. S., & Nanda, N. (2025). Internal states before wait modulate reasoning patterns. EMNLP 2025 Findings. arXiv:2510.04128. doi:10.48550/arXiv.2510.04128
- Venhoff, C., Arcuschin, I., Torr, P., Conmy, A., & Nanda, N. (2025a). Understanding Reasoning in Thinking Language Models via Steering Vectors. arXiv:2506.18167. doi:10.48550/arXiv.2506.18167
- Venhoff, C., Arcuschin, I., Torr, P., Conmy, A., & Nanda, N. (2025b). Base Models Know How to Reason, Thinking Models Learn When. ICML 2026. arXiv:2510.07364. doi:10.48550/arXiv.2510.07364
- Voita, E., Talbot, D., Moiseev, F., Sennrich, R., & Titov, I. (2019). Analyzing Multi-Head Self-Attention: Specialized Heads Do the Heavy Lifting, the Rest Can Be Pruned. ACL 2019. arXiv:1905.09418. doi:10.48550/arXiv.1905.09418
- Wang, K., Variengien, A., Conmy, A., Shlegeris, B., & Steinhardt, J. (2022). Interpretability in the Wild: a Circuit for Indirect Object Identification in GPT-2 small. arXiv:2211.00593. doi:10.48550/arXiv.2211.00593
- Wang, Y., Yang, Q., Zeng, Z., Ren, L., Liu, L., Peng, B., Cheng, H., He, X., Wang, K., Gao, J., Chen, W., Wang, S., Du, S. S., & Shen, Y. (2025a). Reinforcement Learning for Reasoning in Large Language Models with One Training Example. arXiv:2504.20571. doi:10.48550/arXiv.2504.20571
- Wang, S., Yu, L., Gao, C., Zheng, C., Liu, S., Lu, R., Dang, K., Chen, X., Yang, J., Zhang, Z., Liu, Y., Yang, A., Zhao, A., Yue, Y., Song, S., Yu, B., Huang, G., & Lin, J. (2025b). Beyond the 80/20 Rule: High-Entropy Minority Tokens Drive Effective Reinforcement Learning for LLM Reasoning. NeurIPS 2025. arXiv:2506.01939. doi:10.48550/arXiv.2506.01939
- Ward, J., Lin, C., Venhoff, C., & Nanda, N. (2025). Reasoning-Finetuning Repurposes Latent Representations in Base Models. ICML 2025 Workshop on Actionable Interpretability. arXiv:2507.12638. doi:10.48550/arXiv.2507.12638
- Ward, J., Riechers, P., & Shai, A. (2025). Rank-1 LoRAs Encode Interpretable Reasoning Signals. NeurIPS 2025 Mechanistic Interpretability Workshop. arXiv:2511.06739. doi:10.48550/arXiv.2511.06739
- Wu, W., Wang, Y., Xiao, G., Peng, H., & Fu, Y. (2024). Retrieval Head Mechanistically Explains Long-Context Factuality. arXiv:2404.15574. doi:10.48550/arXiv.2404.15574
- Wu, M., Zhang, Z., Dong, Q., Xi, Z., Zhao, J., Jin, S., Fan, X., Zhou, Y., Lv, H., Zhang, M., Fu, Y., Liu, Q., Zhang, S., & Zhang, Q. (2025a). Reasoning or Memorization? Unreliable Results of Reinforcement Learning Due to Data Contamination. AAAI 2026. arXiv:2507.10532. doi:10.48550/arXiv.2507.10532
- Wu, F., Xuan, W., Lu, X., Liu, M., Dong, Y., Harchaoui, Z., & Choi, Y. (2025b). The Invisible Leash: Why RLVR May or May Not Escape Its Origin. arXiv:2507.14843. doi:10.48550/arXiv.2507.14843
- Yan, J., Li, Y., Hu, Z., Wang, Z., Cui, G., Qu, X., Cheng, Y., & Zhang, Y. (2025). Learning to Reason under Off-Policy Guidance. arXiv:2504.14945. doi:10.48550/arXiv.2504.14945
- Yang, A., et al. (2025). Qwen3 Technical Report. arXiv:2505.09388. doi:10.48550/arXiv.2505.09388
- Ye, Y., Huang, Z., Xiao, Y., Chern, E., Xia, S., & Liu, P. (2025). LIMO: Less is More for Reasoning. COLM 2025. arXiv:2502.03387. doi:10.48550/arXiv.2502.03387
- Yeo, E., Tong, Y., Niu, M., Neubig, G., & Yue, X. (2025). Demystifying Long Chain-of-Thought Reasoning in LLMs. arXiv:2502.03373. doi:10.48550/arXiv.2502.03373
- Yue, Y., Chen, Z., Lu, R., Zhao, A., Wang, Z., Yue, Y., Song, S., & Huang, G. (2025). Does Reinforcement Learning Really Incentivize Reasoning Capacity in LLMs Beyond the Base Model? NeurIPS 2025. arXiv:2504.13837. doi:10.48550/arXiv.2504.13837
- Zhang, F., & Nanda, N. (2024). Towards Best Practices of Activation Patching in Language Models: Metrics and Methods. ICLR 2024. arXiv:2309.16042. doi:10.48550/arXiv.2309.16042
- Zhao, R., Meterezh, A., Kakade, S., Pehlevan, C., Jelassi, S., & Malach, E. (2025). Echo Chamber: RL Post-training Amplifies Behaviors Learned in Pretraining. COLM 2025. arXiv:2504.07912. doi:10.48550/arXiv.2504.07912
- Zheng, Z., Wang, Y., Huang, Y., Song, S., Yang, M., Tang, B., Xiong, F., & Li, Z. (2024). Attention Heads of Large Language Models: A Survey. arXiv:2409.03752. doi:10.48550/arXiv.2409.03752
- Zhou, C., Liu, P., Xu, P., Iyer, S., Sun, J., Mao, Y., Ma, X., Efrat, A., Yu, P., Yu, L., Zhang, S., Ghosh, G., Lewis, M., Zettlemoyer, L., & Levy, O. (2023). LIMA: Less Is More for Alignment. arXiv:2305.11206. doi:10.48550/arXiv.2305.11206
- Zhu, H., Zhang, Z., Huang, H., Su, D., Liu, Z., Zhao, J., Fedorov, I., Pirsiavash, H., Sha, Z., Lee, J., Pan, D. Z., Wang, Z., Tian, Y., & Tai, K. S. (2025). The Path Not Taken: RLVR Provably Learns Off the Principals. arXiv:2511.08567. doi:10.48550/arXiv.2511.08567