# LLMTransit: How Semantic Competition and Training Shape Layer-wise Decision Transitions

Anonymous ACL submission

## Abstract

Layer-wise decision transition offers an answer-level view of how LLMs form stable preferences across depth, and has been used to explain pruning-induced performance collapse. Prior work uses this transition mainly to explain pruning-induced performance collapse, leaving open what shapes it and whether it can guide training or inference. We study this question in multiple-choice question answering, where explicit options let us compare the gold option with its strongest competitor at each layer, and include a boundary check on generated continuations. Controlled comparisons show that semantically hard distractors delay and weaken the transition. Training history also reshapes the trajectory: pretraining tends to move stable crossing earlier, whereas instruction tuning consistently reduces final uncertainty but changes transition timing and accumulated margin in a family-dependent way. Finally, transition-aware supervision improves accuracy under moderate hard competition, and predicted-answer transition features provide route-dependent gains for budgeted routing. These findings recast decision transition as a measurable signal of candidate competition resolution. The code is available at https://anonymous.4open.science/r/LLMTransit-FAEE/.

## 1 Introduction

LLMs achieve strong performance across many tasks (Chowdhery et al., 2023; Touvron et al., 2023; Yang et al., 2025), but their depth makes deployment costly. Layer pruning reduces memory use and latency by removing Transformer blocks, yet often shows a *performance cliff*, where accuracy remains stable under moderate pruning but collapses beyond a model- and task-dependent threshold. This suggests that layers support not only computation, but also the process by which a stable answer preference is formed.

Recent work studies this cliff through layer-wise structure. Functional-localization analyses associate shallow layers with retrieval-like behavior and deeper layers with computation-oriented behavior (Gromov et al., 2025). A decision-centric study on multiple-choice question answering (MCQA) defines the layer-wise *Decision Margin* (DM) as the gap between the gold option and its strongest competitor (Shi et al., 2026). It identifies a transition from sustained non-positive DM to sustained positive DM, separating the *Silent Phase*, where the gold option is not yet preferred, from the *Decisive Phase*, where it remains stably ahead. This view explains pruning collapse as a failure to complete candidate-competition resolution. However, the transition is more than a binary phase boundary: its depth, margin magnitude, stability, and uncertainty describe how the gold-option preference emerges and persists across layers. Prior work does not explain what factors shape these properties. This leads to the central question: how is the transition shaped by semantic competition among answer options and by training history, including pretraining progress and instruction tuning?

To answer this, we study decision transition in MCQA, where explicit answer options make the gold option and its competitors directly comparable at each layer. This setting lets us control data-side competition through distractor semantics and model-side variation through pretraining checkpoints and instruction tuning. Following intermediate-readout analyses (nostalgebraist, 2020; Belrose et al., 2023), we trace layer-wise DM trajectories and summarize them with persistent crossing for transition depth, positive area for transition strength, preference flips and post-transition volatility for stability, and final uncertainty for residual ambiguity. These measurements allow us to compare when the gold-option preference emerges, how strongly it is maintained, how stable it remains, and how much uncertainty per-

sists at the final layer.

We then build the controlled MCQA framework in Figure 1 to study what reshapes these properties. MCQA is the main setting because explicit candidates make the strongest competitor observable at every layer. The framework introduces two sources of variation. On the data side, pseudo-gold controls test whether the transition is tied to the annotated answer, while hard distractors test how semantic competition changes the trajectory. On the model side, pretraining checkpoints and base-to-instruct comparisons test how training-stage variation affects preference formation. We also include an open-ended generation boundary check, where models produce continuations through ordinary full-vocabulary decoding and verified continuations are compared with hard or random wrong continuations.

Applying this framework to Llama (Grattafiori et al., 2024) and Qwen (Yang et al., 2025) model families shows that transition is answer-specific and condition-sensitive. Across MCQA tasks, real gold options cross earlier and accumulate larger positive margins than pseudo-gold controls. Semantically hard distractors delay persistent crossing by 1.81 layers and reduce positive area by 9.31 on average. Training-stage variation also reshapes the trajectory. Pretraining generally moves stable crossing earlier as competence grows, while instruction tuning reduces final uncertainty but changes transition depth, accumulated margin, and stability in a family-dependent manner.

Beyond mechanism analysis, we further examine whether the transition can support concrete decisions. Figure 1(c) shows two intervention protocols. Competition-Aware Transition Training (CATT) adds transition-aware supervision under hard semantic competition to test whether delayed or weak transitions provide information beyond hard-example exposure. Transition-Aware Adaptive Inference (TAI) uses predicted-answer transition features to route cases from a smaller model to a stronger model under a fixed budget. CATT improves accuracy under moderate hard competition, and TAI yields route-dependent gains over final-confidence routing.

## 2 Preliminaries

MCQA provides an explicit candidate set, which makes answer formation observable as competition among options. Let $x = (q, \mathcal{A})$ be an instance with candidates $\mathcal{A} = \{a_1, \ldots, a_K\}$ and gold option $g$. Following prior work on layer-wise decision transitions (Shi et al., 2026), we read out an option score $z_\ell(j)$ for each candidate $a_j$ at layer $\ell$. The layer-wise *Decision Margin* (DM) is

$$\Delta_\ell(x) = z_\ell(g) - \max_{j \neq g} z_\ell(j), \qquad \ell = 1, \ldots, L. \tag{1}$$

A negative margin means that at least one distractor is ahead of the gold option, while a positive margin means that the gold option leads. The trajectory $\{\Delta_\ell(x)\}_{\ell=1}^{L}$ therefore records how the gold option separates from its strongest competitor across depth.

Prior work uses this trajectory to explain pruning-induced collapse, identifying a transition from a *Silent Phase*, where the gold option is not stably preferred, to a *Decisive Phase*, where it remains ahead (Shi et al., 2026). We keep the same readout but ask what shapes the transition and whether the trajectory can guide controlled training and routing decisions. In this view, decision transition is not a static layer boundary, but a measurable signal of candidate competition resolution. We discuss related work in Appendix A.2.

## 3 Method

As shown in Figure 1, our framework treats decision transition as a dynamic signal of how the gold option becomes stably preferred over competing options. It combines controlled comparisons that vary competitor strength or model training history, transition features that summarize each DM trajectory $\Delta_{1:L}(x)$, and two intervention protocols for training-time supervision and inference-time routing. This design allows us to ask what reshapes the transition and whether the same signal can support decisions.

### 3.1 Controlled Sources of Transition Variation

We study two sources of transition variation. The first is data-side semantic competition. Before varying distractors, pseudo-gold controls test whether the DM trajectory is tied to the annotated answer rather than arbitrary option-level drift. They treat a non-gold option as the temporary target while keeping the question, candidate set, model, prompt, and forward pass unchanged. We also use option-order controls as a boundary check for positional sensitivity.

We then vary distractor semantics under matched settings. For each source question, the gold option,

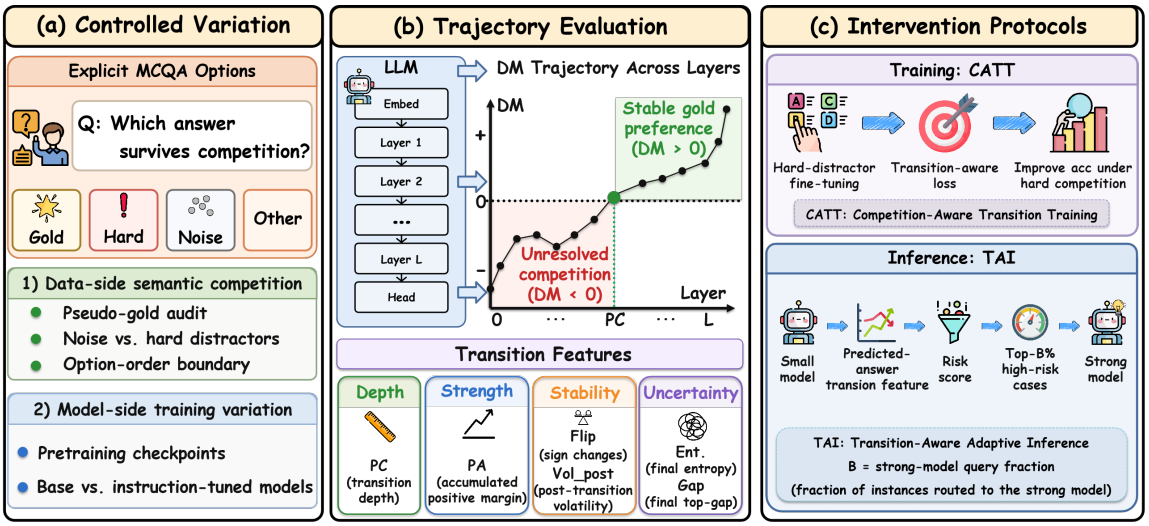


Figure 1: Overall framework. (a) Controlled variation changes distractor semantics or model training stage in MCQA. (b) Layer-wise DM trajectories are summarized by timing, preference strength, stability, and uncertainty. (c) The transition signal is used for CATT supervision and TAI routing.

model, option count, and option order are fixed. The comparison replaces unrelated noise distractors with semantically hard distractors, defined as plausible but incorrect options. If transition reflects candidate competition resolution, harder distractors should make the gold option cross later, accumulate less positive margin, or show less stable trajectories.

The second source is model-side training variation. For pretraining, we compare ordered checkpoints from the same model series under the same MCQA scoring rule. For instruction tuning, we compare paired base and instruction-tuned models on the same datasets and prompts. These comparisons test whether training changes when candidate competition is resolved, how strongly the gold option dominates after resolution, and whether final uncertainty changes together with intermediate trajectory stability.

### 3.2 Trajectory Evaluation

For each instance $x$, we compute transition features from $\Delta_{1:L}(x)$. Since a single positive margin may be transient, we use persistent crossing as the main statistic for transition depth:

$$\mathrm{PC}(x) = \min\{\ell : \Delta_\ell(x) > 0,\ s_\ell(x) \geq \rho\},$$
$$s_\ell(x) = \frac{1}{L-\ell+1}\sum_{t=\ell}^{L} \mathbb{I}[\Delta_t(x) > 0]. \tag{2}$$

We set $\rho = 0.8$ and assign $\mathrm{PC}(x) = L+1$ when no layer satisfies the condition. A smaller PC indicates an earlier stable transition.

Transition strength and trajectory stability are measured as

$$\mathrm{PA}(x) = \sum_{\ell=1}^{L} \max(\Delta_\ell(x), 0), \tag{3}$$

$$\mathrm{Flip}(x) = \sum_{\ell=2}^{L} \mathbb{I}[\Delta_{\ell-1}(x)\Delta_\ell(x) < 0]\,, \tag{4}$$

$$\mathrm{Vol}_{\mathrm{post}}(x) = \mathrm{Var}\{\Delta_t(x) : \mathrm{PC}(x) \leq t \leq L\}. \tag{5}$$

Here, PA measures the accumulated positive margin of the gold option. Flip counts how often the trajectory changes sign, and $\mathrm{Vol}_{\mathrm{post}}$ measures fluctuation after persistent crossing. The post-transition volatility is computed only when $\mathrm{PC}(x) \leq L$, with no-crossing cases reported separately.

Final-layer uncertainty is computed from the option distribution:

$$p_L(j) = \frac{\exp z_L(j)}{\sum_{k=1}^{K} \exp z_L(k)}, \tag{6}$$

$$H(x) = -\sum_{j=1}^{K} p_L(j) \log p_L(j), \tag{7}$$

$$\mathrm{Gap}_{12}(x) = p_L^{(1)} - p_L^{(2)}, \tag{8}$$

where $p_L^{(1)}$ and $p_L^{(2)}$ denote the largest and second-largest option probabilities at the final layer. In the experiments, Ent. refers to $H(x)$, and Gap refers to $\mathrm{Gap}_{12}(x)$. Together, these features separate transition timing, accumulated preference, trajectory stability, and residual final-layer uncertainty.

### 3.3 Intervention Protocols

The intervention protocols test whether the same transition features support decisions beyond post-hoc analysis. The training-side protocol isolates transition-aware supervision from hard-example exposure, while the inference-side protocol tests whether predicted-answer transition features add routing evidence beyond final confidence.

**Training-side intervention.** Competition-Aware Transition Training (CATT) uses the hard-distractor design to supervise unresolved candidate competition during fine-tuning. For each source question, the training set contains the original instance, a noise-distractor variant, and a hard-distractor variant. The SFT-mix baseline uses the same instances with only final-answer supervision, isolating transition-aware losses from hard-example exposure.

The CATT objective is

$$\mathcal{L}_{\mathrm{CATT}} = \mathcal{L}_{\mathrm{CE}} + \lambda_{\mathrm{tr}}\mathcal{L}_{\mathrm{tr}} + \lambda_{\mathrm{hr}}\mathcal{L}_{\mathrm{hr}}, \tag{9}$$

where $\mathcal{L}_{\mathrm{CE}}$ is the standard final-answer cross-entropy. The transition-aware terms are

$$\mathcal{L}_{\mathrm{tr}}(x) = \frac{1}{|\mathcal{W}|}\sum_{\ell\in\mathcal{W}} \max(0, \tau_{\mathrm{tr}} - \Delta_\ell(x)), \tag{10}$$

$$\mathcal{L}_{\mathrm{hr}}(x) = \frac{1}{|\mathcal{W}|\,|\mathcal{H}(x)|}\sum_{\ell\in\mathcal{W}}\sum_{h\in\mathcal{H}(x)} \max(0, \gamma - (z_\ell(g) - z_\ell(h))). \tag{11}$$

Here, $\mathcal{W}$ is a middle-to-late layer window and $\mathcal{H}(x)$ is the hard-distractor set. The transition loss encourages positive gold-option margins after a meaningful comparison has formed, while the hard-rank term directly separates the gold option from hard competitors.

**Inference-side intervention.** Transition-Aware Adaptive Inference (TAI) tests whether transition features can be used when the gold option is unavailable. The smaller model first predicts

$$\hat{y} = \arg\max_j z_L(j), \tag{12}$$

and we trace the margin of this predicted option:

$$\widehat{\Delta}_\ell(x) = z_\ell(\hat{y}) - \max_{j\neq\hat{y}} z_\ell(j). \tag{13}$$

Features from $\widehat{\Delta}_{1:L}(x)$ form a deployable risk signal. Late crossing, weak positive area, frequent sign changes, high post-transition volatility, high entropy, or a small final gap indicate that the smaller model may not have fully resolved candidate competition.

The router combines transition and final-confidence features:

$$r_\theta(x) = \sigma(w^\top\phi(x) + b), \tag{14}$$

where $\phi(x)$ contains deployable transition and confidence features. TAI follows a cascaded routing setting. Every instance is first processed by the smaller model, whose prediction is the default answer and whose trajectory provides the risk features. This first pass is the default inference path rather than an additional diagnostic run. The strong-model budget $B$ denotes the fraction of instances allowed to query the stronger model. We rank instances by $r_\theta(x)$, let $\mathcal{R}_B$ contain the top $B\%$ highest-risk instances, and route only this subset to the stronger model:

$$y_{\mathrm{TAI}}(x) = \begin{cases} y_{\mathrm{strong}}(x), & x \in \mathcal{R}_B, \\ \hat{y}, & x \notin \mathcal{R}_B. \end{cases} \tag{15}$$

We compare TAI with confidence routing and random routing to test whether layer-wise transition features provide risk information beyond final-layer confidence.

**Open-ended generation boundary.** The main protocol uses MCQA because fixed options give a clean competitor set. To test whether the signal is limited to this format, we also evaluate open-ended generation tasks. Models generate continuations through standard full-vocabulary decoding; after task-specific verification, we compare a verified continuation with either a random wrong continuation or a harder wrong continuation. We then compute layer-wise continuation scores with the same intermediate readout and summarize the resulting margin trajectory with PC, PA, and final win rate.

## 4 Experiments

We evaluate the layer-wise decision transition signal along two axes. Mechanism studies test how dataset-level competition, pretraining progress, and instruction tuning shape trajectories; intervention studies test whether the same signal supports controlled training and inference decisions. Implementation details, model abbreviations, and training or routing hyperparameters are reported in Appendix A.1, Table 6, and Table 7.

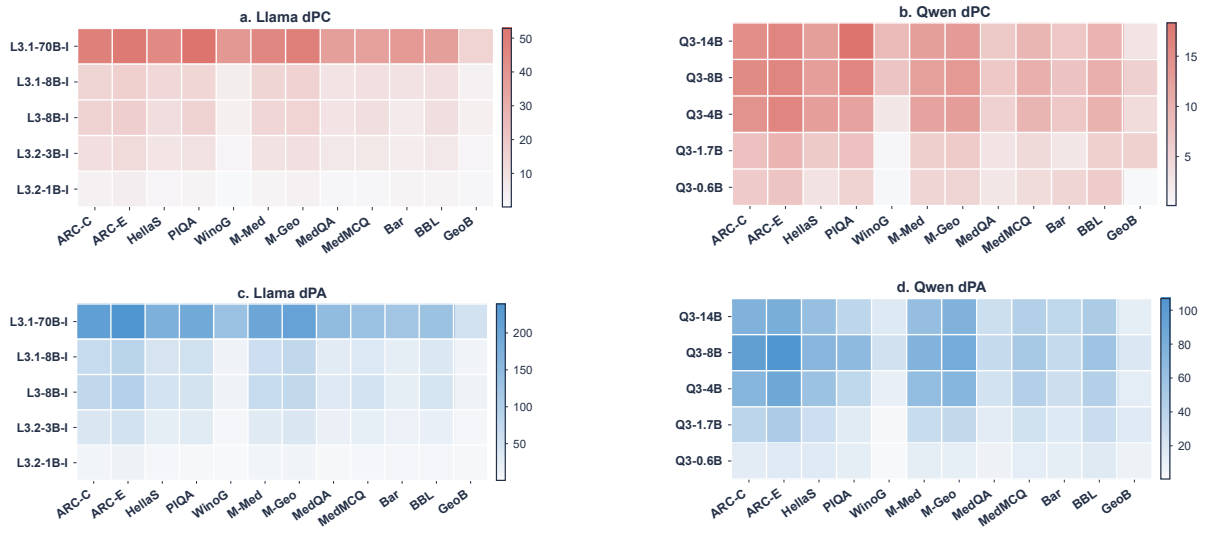


Figure 2: Transition specificity of the true gold option. Pseudo-gold controls treat incorrect options as temporary targets, testing whether the transition signal is tied to the correct answer rather than arbitrary option-level drift. Darker colors indicate larger gaps.

### 4.1 Dataset-Level Competition

**Gold Options Define Real Transitions** We first check whether the transition signal is tied to the correct answer. For each question, we compare the true-gold trajectory with a pseudo-gold trajectory that treats an incorrect option as the target. Figure 2 shows a consistent gap: pseudo-gold options reach PC later in the red panels, while true gold options accumulate larger PA in the blue panels. Real gold options follow this expected direction in 117 of 120 settings; on average, pseudo-gold options reach PC 11.23 layers later, and the true gold option has 47.46 higher accumulated positive margin. Thus, PC and PA track correct-answer formation rather than a pattern that appears for any chosen option.

Table 8 in Appendix A.3 reports the dataset-level averages behind Figure 2. The real-vs.-pseudo direction remains positive for every dataset in both Llama and Qwen, although gap sizes vary across tasks. Figure 5 further shows that candidate order can affect measured transition values. The following competition experiments therefore use matched comparisons with the same source question, gold option, model, option count, and option order.

**Effect of semantic competition on transition** After verifying that the transition signal is tied to the gold answer, we test whether competitor strength is a data-side factor that reshapes the transition. For each question, we compare two matched option sets: one with unrelated noise distractors and one with semantically hard distractors. The question, gold option, model, option count, and option order are fixed, so differences reflect competitor strength rather than prompt format.

Figure 3 reports the gap between weak and strong competition. PC is hard minus noise, so positive values mean later persistent crossing under hard distractors. PA and accuracy are noise minus hard, so positive values mean reduced accumulated margin or lower final correctness. Averaged over the two model families, hard distractors delay PC by 1.81 layers, reduce PA by 9.31, and lower accuracy by 10.55 percentage points. Semantic competition therefore affects both the final answer and the layer-wise resolution of the correct option.

The direction is consistent at the dataset-family level, but effect size varies. M-Geo shows the largest shift, with PC delayed by 6.96 layers for Llama and 2.62 layers for Qwen. HellaS and M-Med also show clear PA and accuracy reductions, while ARC-C and Bar show smaller positive gaps. Llama has larger average PC and PA shifts than Qwen, but average accuracy drops are similar. Thus, hard distractors reliably reshape the trajectory, while final accuracy depends on dataset and model family. The model-level heatmaps in Figure 6 and Appendix A.4 confirm that this trend is not driven by a single model size or family.

### 4.2 Pretraining Variation

We next examine whether pretraining progress changes the transition. The analysis uses 16 ordered checkpoints (Groen-

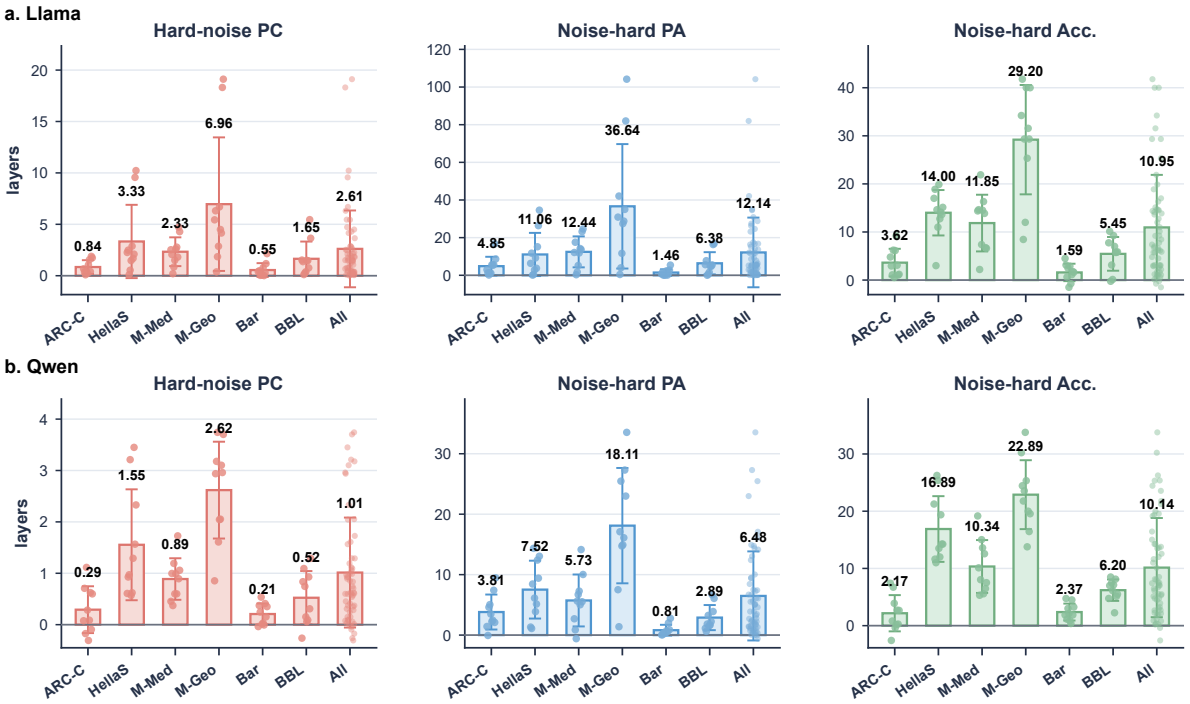


Figure 3: Effect of semantic competition on decision transition. Hard distractors are compared with unrelated noise distractors under matched settings. Positive values indicate delayed PC, reduced PA, or lower final accuracy.

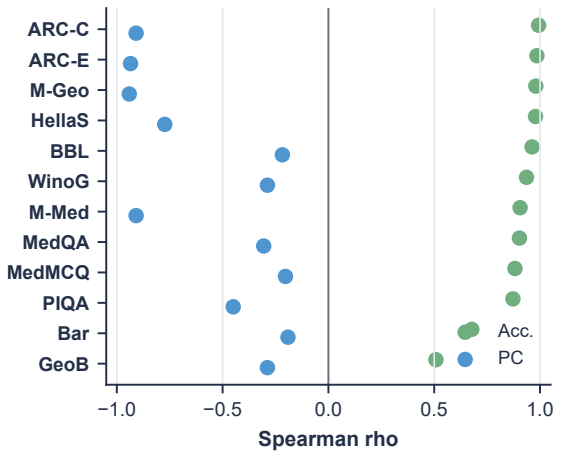


Figure 4: Dataset-level Spearman correlations between checkpoint order and accuracy or PC over the same 16-checkpoint pretraining trajectory. Accuracy generally increases over checkpoints, while PC tends to move earlier with dataset-dependent strength.

eveld et al., 2024) from the same model series, from `stage1-step150-tokens1B` to `stage2-ingredient3-step11931-tokens50B`. All checkpoints are evaluated on the same 12 MCQA datasets with the same prompts, scoring rule, and transition metrics, making checkpoint order a controlled proxy for pretraining progress. Full checkpoint details are reported in Appendix A.1.

**Pretraining Moves Transition Earlier** Figure 4 summarizes how final accuracy and PC change over checkpoints. For each dataset, we compute Spearman correlations with checkpoint order. Accuracy is positively correlated with checkpoint order on all 12 datasets (0.88 on average), while PC is negatively correlated on all 12 datasets ($-0.53$ on average). Since smaller PC means earlier stable crossing, later checkpoints tend to resolve candidate competition earlier while also becoming more accurate.

The checkpoint curves in Figure 7 and Appendix A.5 add a qualification: accuracy often rises and then stabilizes, whereas PC can peak at early checkpoints, decline later, and fluctuate locally. Pretraining therefore affects transition timing, but the transition is not merely a restatement of final accuracy; it remains a layer-wise signal of when the model resolves the gold option against competitors.

### 4.3 Instruction-Tuning Variation

**Instruction Tuning Sharpens Final Preference Across Families** We then examine whether instruction tuning reshapes transition after pretraining. Unlike pretraining checkpoint comparisons,

| Dataset | Llama | | | Qwen | | |
|---|---|---|---|---|---|---|
| | PC | PA | Ent. | PC | PA | Ent. |
| ARC-C | -2.82 | 28.15 | -0.48 | 0.51 | 3.32 | -0.38 |
| ARC-E | -1.41 | 26.95 | -0.47 | 0.37 | 3.10 | -0.34 |
| HellaS | -3.51 | 25.86 | -0.46 | 0.54 | -2.01 | -0.40 |
| PIQA | -4.48 | 30.20 | -0.27 | 0.83 | -5.96 | -0.22 |
| WinoG | -3.65 | 21.19 | -0.14 | 0.44 | -2.62 | -0.25 |
| M-Med | -2.44 | 23.66 | -0.39 | 0.21 | 4.29 | -0.52 |
| M-Geo | -2.91 | 24.16 | -0.45 | 0.58 | 5.03 | -0.40 |
| MedQA | -8.50 | 55.60 | -0.78 | 1.50 | -2.75 | -1.12 |
| MedMCQ | -2.08 | 16.26 | -0.36 | 0.77 | 1.42 | -0.45 |
| Bar | -4.30 | 17.49 | -0.36 | 0.05 | 1.73 | -0.43 |
| BBL | -1.42 | 12.92 | -0.33 | 1.39 | -0.56 | -0.41 |
| GeoB | -2.06 | 9.54 | -0.43 | 1.66 | -1.13 | -0.49 |
| Mean | -3.30 | 24.33 | -0.41 | 0.74 | 0.32 | -0.45 |

Table 1: Instruction-tuning effects for Llama and Qwen families. Values are instruction-minus-base differences; Ent. denotes final entropy.

| Dataset | Llama | | | | Qwen | | | |
|---|---|---|---|---|---|---|---|---|
| | Flip | Vol. | Ent. | Gap | Flip | Vol. | Ent. | Gap |
| ARC-C | -0.44 | 4.07 | -0.48 | 0.31 | -0.50 | -1.08 | -0.38 | 0.23 |
| ARC-E | -0.41 | 4.17 | -0.47 | 0.28 | -0.45 | -1.03 | -0.34 | 0.20 |
| HellaS | -0.50 | 3.73 | -0.46 | 0.32 | 0.07 | -1.74 | -0.40 | 0.23 |
| PIQA | -1.62 | 4.75 | -0.27 | 0.36 | 0.42 | 0.36 | -0.22 | 0.25 |
| WinoG | -1.48 | 2.87 | -0.14 | 0.17 | -0.28 | 1.37 | -0.25 | 0.34 |
| M-Med | -0.23 | 2.77 | -0.39 | 0.25 | -0.76 | -0.21 | -0.52 | 0.33 |
| M-Geo | -0.10 | 3.68 | -0.45 | 0.28 | -0.23 | -0.04 | -0.40 | 0.23 |
| MedQA | -2.30 | 1.66 | -0.78 | 0.40 | 0.50 | 2.09 | -1.12 | 0.43 |
| MedMCQ | -0.08 | 2.48 | -0.36 | 0.25 | -0.63 | 0.56 | -0.45 | 0.29 |
| Bar | -0.17 | 2.58 | -0.36 | 0.26 | -0.24 | -0.51 | -0.43 | 0.31 |
| BBL | -0.08 | 2.51 | -0.33 | 0.22 | -0.14 | -0.15 | -0.41 | 0.25 |
| GeoB | -0.43 | 1.55 | -0.43 | 0.14 | 0.43 | -0.21 | -0.49 | 0.13 |
| Mean | -0.65 | 3.07 | -0.41 | 0.27 | -0.15 | -0.05 | -0.45 | 0.27 |

Table 2: Trajectory-stability audit after instruction tuning. Values are instruction-minus-base differences averaged within each model family. Ent. is final entropy, Gap is final top-gap, and Vol. is post-transition volatility.

this setting compares paired base and instruction-tuned models from the Llama and Qwen families, allowing us to test whether alignment-style tuning changes transition timing, accumulated margin, stability, and final uncertainty.

Table 1 shows the most consistent effect of instruction tuning: both model families end with a sharper option preference. Entropy decreases by 0.41 for Llama and 0.45 for Qwen on average. This shared endpoint change indicates that tuning affects the decision trajectory, even when transition timing does not move in the same direction across families. For Llama, instruction tuning moves PC earlier by 3.30 layers and increases PA by 24.33; for Qwen, PC is delayed by 0.74 layers and PA changes only slightly by 0.32. The corresponding PC and PA heatmaps in Figure 8 provide the full dataset-by-model view.

We further ask whether this sharper final preference also means a smoother trajectory. Table 2 shows that it does not. Entropy decreases in 117 of 120 paired settings, and the final top-gap increases in 115 settings. In contrast, Flip decreases in 76 settings, while post-transition volatility increases in 82 settings. Multiple metrics are therefore needed: tuning consistently changes the endpoint, but PC, PA, Flip, and volatility reveal how the path changes. Figure 9 in Appendix A.5 shows that the entropy reduction holds across nearly the full base-to-instruction matrix.

### 4.4 Intervention Tests

**Training-Time Transition Supervision** Hard competitors delay and weaken the transition, so we test whether transition information helps fine-tuning beyond hard-example exposure. CATT and SFT use the same original, noise-distractor, and hard-distractor variants; only CATT adds transition-

| Cond. | Acc. | | | PA | | |
|---|---|---|---|---|---|---|
| | Base | SFT | CATT | Base | SFT | CATT |
| Orig. | **83.54** | 82.28 | 82.28 | 61.66 | 73.03 | **76.15** |
| N+1 | 82.28 | 82.28 | **83.54** | 58.09 | 71.04 | **74.89** |
| N+2 | 82.28 | 82.28 | **83.54** | 60.09 | 69.57 | **74.06** |
| N+3 | **83.54** | 82.28 | **83.54** | 59.72 | 69.58 | **74.63** |
| N+4 | 82.28 | 83.54 | **84.81** | 58.83 | 68.01 | **72.85** |
| N+5 | 81.01 | 82.28 | **84.81** | 57.94 | 67.20 | **73.30** |
| N+6 | **86.08** | 79.75 | 83.54 | 57.34 | 65.80 | **72.06** |
| H+1 | 81.01 | **83.54** | **83.54** | 55.12 | 71.82 | **75.94** |
| H+2 | 78.48 | 81.01 | **84.81** | 55.63 | 70.79 | **75.55** |
| H+3 | 79.75 | **84.81** | **84.81** | 56.43 | 73.46 | **77.60** |
| H+4 | 77.22 | 82.28 | **83.54** | 55.09 | 70.99 | **75.63** |
| H+5 | 72.15 | **84.81** | 83.54 | 52.61 | 69.23 | **74.59** |
| H+6 | 70.89 | **82.28** | **82.28** | 50.92 | 68.43 | **73.85** |

Table 3: Training-time validation on M-Med variants with Qwen3-8B-Base. N+$k$ and H+$k$ add $k$ noise or hard distractors. SFT denotes the LoRA fine-tuned baseline with final-answer supervision only. Bold marks the best value in each row and metric group.

| Task | Route | B | Method | Acc. | AUROC |
|---|---|---|---|---|---|
| ARC | $4 \to 14$ | 15% | Base | 90.60 | – |
| | | | Strong | 91.30 | – |
| | | | Conf. | 91.90 | 0.87 |
| | | | TAI | **92.20** | 0.89 |
| | $8 \to 14$ | 15% | Base | 89.20 | – |
| | | | Strong | 90.20 | – |
| | | | Conf. | 91.30 | 0.88 |
| | | | TAI | **91.30** | 0.88 |
| MMLU | $4 \to 14$ | 50% | Base | 76.30 | – |
| | | | Strong | **87.90** | – |
| | | | Conf. | 87.30 | 0.82 |
| | | | TAI | 87.50 | 0.82 |
| | $8 \to 14$ | 30% | Base | 81.00 | – |
| | | | Strong | 87.90 | – |
| | | | Conf. | 87.30 | 0.86 |
| | | | TAI | **88.40** | 0.86 |

Table 4: Best-budget TAI routing summary. B is the fraction routed to the strong model; Conf. is confidence routing; Acc. is accuracy; AUROC measures routing-risk ranking.

| Family | R-Win | H-Win | ΔWin | ΔPC |
|---|---|---|---|---|
| Llama | 31.17 | 12.67 | 18.50 | 1.67 |
| Qwen | 26.16 | 10.60 | 15.56 | 1.16 |

Table 5: Family-level continuation boundary check. R and H denote random-wrong and hard-wrong continuations. ΔWin is R-H, and ΔPC is H-R.

aware losses. Table 3 expands the validation to 13 matched variants. CATT obtains the best or tied accuracy in 10 settings and the best PA in all settings. On hard-distractor variants, it improves accuracy by 0.63 points over SFT and 7.17 points over the base model on average, supporting transition-aware supervision under controlled hard competition.

**Inference-Time Transition Routing** We next test whether predicted-answer transitions help decide when to call a stronger model. In TAI, the smaller model gives the default answer and risk features; only the highest-risk $B\%$ instances are routed upward. Table 4 reports best-budget results, with full curves in Figure 10. TAI is best or tied on ARC and on the MMLU $8 \to 14$ route. On ARC $4 \to 14$, for example, it reaches $92.20\%$ accuracy with a $15\%$ strong-model budget, above both confidence routing and full 14B inference. Figure 11 shows the same route dependence from the risk-ranking view. These results indicate that transition features help routing when unresolved competition is not fully exposed by final confidence.

### 4.5 Boundary Check Beyond Explicit Options

**Continuation Competition Gives Boundary Evidence** We further test whether a transition-like pattern appears when explicit MCQA options are replaced by generated continuations. Table 5 reports family-level means, while task-level results are provided in Tables 10 and 11. Compared with random-wrong continuations, hard-wrong continuations reduce the chosen-win rate by $18.50$ points for Llama and $15.56$ points for Qwen, while delaying PC by $1.67$ and $1.16$ layers. This direction matches the MCQA semantic-competition result, suggesting that stronger competitors also delay preference formation in continuation settings. Since these tasks do not provide a fixed option set and PA is less consistent in the appendix results, we treat this experiment as a boundary check rather than a main claim about open-ended generation.

## 5 Conclusion

We study layer-wise decision transition as a signal of candidate competition resolution. Stronger competitors delay and weaken this transition in MCQA and in a candidate-continuation boundary check, while pretraining and instruction tuning reshape timing, strength, stability, and uncertainty. Training and routing interventions show bounded operational value, suggesting that decision transition connects pruning-collapse diagnostics with measurable answer-formation dynamics.

## Limitations

This work studies decision transition mainly in MCQA, where explicit answer options make candidate competition directly measurable. Although we include generative continuation tasks as a boundary check, these experiments do not cover the full range of unrestricted open-ended generation. Similarly, while the training and inference interventions show that transition features can be useful in controlled settings, they are intended as evidence for the signal rather than as general solutions to hard-example training or model routing. Future work can extend transition-based analysis to broader generation settings and examine how such signals interact with more diverse training and inference policies.

## Ethical Considerations

This work uses public benchmarks and studies model behavior rather than deploying systems for real-world decisions. Because some evaluated tasks involve medical, legal, and routing settings, transition signals should be treated as diagnostic evidence and should not be used as standalone criteria in high-stakes applications.

## References


Mohamed Aghzal, Erion Plaku, and Ziyu Yao. 2024. Can large language models be good path planners? a benchmark and investigation on spatial-temporal reasoning. In *ICLR 2024 Workshop on Large Language Model (LLM) Agents*.

Guillaume Alain and Yoshua Bengio. 2016. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and 1 others. 2021. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*.

Nora Belrose, Igor Ostrovsky, Lev McKinney, Zach Furman, Logan Smith, Danny Halawi, Stella Biderman, and Jacob Steinhardt. 2023. Eliciting latent predictions from transformers with the tuned lens. *arXiv preprint arXiv:2303.08112*.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, and 1 others. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439.

Lingjiao Chen, Matei Zaharia, and James Zou. 2023. Frugalgpt: How to use large language models while reducing cost and improving performance. *arXiv preprint arXiv:2305.05176*.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, and 1 others. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, and 1 others. 2023. Palm: Scaling language modeling with pathways. *Journal of machine learning research*, 24(240):1–113.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Danish Contractor, Krunal Shah, Aditi Partap, Parag Singla, and Mausam Mausam. 2021. Answering poi-recommendation questions using tourism reviews. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, CIKM '21, page 281–291, New York, NY, USA. Association for Computing Machinery.

Vijay Devane, Mohd Nauman, Bhargav Patel, Aniket Mahendra Wakchoure, Yogeshkumar Sant, Shyam Pawar, Viraj Thakur, Ananya Godse, Sunil Patra, Neha Maurya, Suraj Racha, Nitish Kamal Singh, Ajay Nagpal, Piyush Sawarkar, Kundeshwar Vijayrao Pundalik, Rohit Saluja, and Ganesh Ramakrishnan. 2025. Bhashabench v1: A comprehensive benchmark for the quadrant of indic domains. *Preprint*, arXiv:2510.25409.

Shanshan Feng, Haoming Lyu, Fan Li, Zhu Sun, and Caishun Chen. 2024. Where to move next: Zero-shot generalization of llms for next poi recommendation. In *2024 IEEE Conference on Artificial Intelligence (CAI)*, pages 1530–1535.

Catherine Finegan-Dollak, Jonathan K. Kummerfeld, Li Zhang, Karthik Ramanathan, Sesh Sadasivam, Rui Zhang, and Dragomir Radev. 2018. Improving text-to-SQL evaluation methodology. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 351–360, Melbourne, Australia. Association for Computational Linguistics.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Dirk Groeneveld, Iz Beltagy, Evan Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, and 1 others. 2024. Olmo: Accelerating the science of language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15789–15809.

Andrey Gromov, Kushal Tirumala, Hassan Shapourian, Paolo Glorioso, and Daniel A Roberts. 2025. The unreasonable ineffectiveness of the deeper layers. In *International Conference on Learning Representations*, volume 2025, pages 81906–81920.

Ehsan Hamzei, Haonan Li, Maria Vasardani, Timothy Baldwin, Stephan Winter, and Martin Tomko. 2020. *Place Questions and Human-Generated Answers: A Data Analysis Approach*, pages 3–19.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Zixian Huang, Yulin Shen, Xiao Li, Yuang Wei, Gong Cheng, Lin Zhou, Xinyu Dai, and Yuzhong Qu. 2019. Geosqa: A benchmark for scenario-based question answering in the geography domain at high school level. *Preprint*, arXiv:1908.07855.

Zixian Huang, Ao Wu, Yulin Shen, Gong Cheng, and Yuzhong Qu. 2021. When retriever-reader meets scenario-based multiple-choice questions. *Preprint*, arXiv:2108.13875.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. 2020. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *ArXiv*, abs/2009.13081.

Sergios-Anestis Kefalidisa, Dharmen Punjani, Eleni Tsalapati, Konstantinos Plas, Mariangela Pollali, Michail Mitsios, Myrto Tsokanaridou, Manolis Koubarakis, and Pierre Maret. 2023. Benchmarking geospatial question answering engines using the dataset geoquestions1089. In *The Semantic Web - ISWC 2023 - 22nd International Semantic Web Conference, Athens, Greece, November 6-10, 2023, Proceedings, Part II*.

Bo-Kyeong Kim, Geonmin Kim, Tae-Ho Kim, Thibault Castells, Shinkook Choi, Junho Shin, and Hyoung-Kyu Song. 2024. Shortened llama: Depth pruning for large language models with comparison of retraining methods. *arXiv preprint arXiv:2402.02834*.

Haonan Li, Martin Tomko, and Timothy Baldwin. 2024. Location aware modular biencoder for tourism question answering. *Preprint*, arXiv:2401.02187.

Kechen Li, Yaotian Tao, Ximing Wen, Quanwei Sun, Zifei Gong, Chang Xu, Xizhe Zhang, and Tianbo Ji. 2025. Gridroute: A benchmark for llm-based route planning with cardinal movement in grid environments. *Preprint*, arXiv:2505.24306.

Deyuan Liu, Zhanyue Qin, Hairu Wang, Zhao Yang, Zecheng Wang, Fangying Rong, Qingbin Liu, Yanchao Hao, Bo Li, Xi Chen, and 1 others. 2024. Pruning via merging: Compressing llms via manifold alignment based layer merging. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17817–17829.

Weijie Liu, Peng Zhou, Zhiruo Wang, Zhe Zhao, Haotang Deng, and Qi Ju. 2020. Fastbert: a self-distilling bert with adaptive inference time. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 6035–6044.

Xinyin Ma, Gongfan Fang, and Xinchao Wang. 2023. Llm-pruner: On the structural pruning of large language models. *Advances in neural information processing systems*, 36:21702–21720.

Xin Men, Mingyu Xu, Qingyu Zhang, Qianhao Yuan, Bingning Wang, Hongyu Lin, Yaojie Lu, Xianpei Han, and Weipeng Chen. 2025. Shortgpt: Layers in large language models are more redundant than you expect. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 20192–20204.

Roshanak Mirzaee and Parisa Kordjamshidi. 2022. Transfer learning with synthetic corpora for spatial role labeling and reasoning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6148–6165, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. *CoRR*, abs/1611.09268.

nostalgebraist. 2020. Interpreting GPT: The logit lens. LessWrong.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, and 1 others. 2022. In-context learning and induction heads. *arXiv preprint arXiv:2209.11895*.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. 2022. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Conference on health, inference, and learning*, pages 248–260. PMLR.

Kangyu Qiao, Shaolei Zhang, and Yang Feng. 2025. Ig-pruning: Input-guided block pruning for large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 10629–10640.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106.

Boyu Shi, Chang Liu, ChuanBao Gao, Xu Yang, and Xin Geng. 2026. Understanding performance collapse in layer-pruned large language models via decision representation transitions. *arXiv preprint arXiv:2605.07271*.

Zhengxiang Shi, Qiang Zhang, and Aldo Lipani. 2022. Stepgame: A new benchmark for robust multi-hop spatial reasoning in texts. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11321–11329.

Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. 2016. Training region-based object detectors with online hard example mining. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 761–769.

Swabha Swayamdipta, Roy Schwartz, Nicholas Lourie, Yizhong Wang, Hannaneh Hajishirzi, Noah A Smith, and Yejin Choi. 2020. Dataset cartography: Mapping and diagnosing datasets with training dynamics. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9275–9293.

Surat Teerapittayanon, Bradley McDanel, and Hsiang-Tsung Kung. 2016. Branchynet: Fast inference via early exiting from deep neural networks. In *2016 23rd international conference on pattern recognition (ICPR)*, pages 2464–2469. IEEE.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, and 1 others. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Fei Wang, Li Shen, Liang Ding, Chao Xue, Ye Liu, and Changxing Ding. 2026. Layer as puzzle pieces: Compressing large language models through layer concatenation. *Advances in Neural Information Processing Systems*, 38:6791–6824.

Zhenyu Wang. 2025. Logitlens4llms: Extending logit lens analysis to modern large language models. *arXiv preprint arXiv:2503.11667*.

Ji Xin, Raphael Tang, Jaejun Lee, Yaoliang Yu, and Jimmy Lin. 2020. Deebert: Dynamic early exiting for accelerating bert inference. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 2246–2251.

Yutaro Yamada, Yihan Bao, Andrew K Lampinen, Jungo Kasai, and Ilker Yildirim. 2024. Evaluating spatial understanding of large language models. *Transactions on Machine Learning Research*.

Xiaodong Yan, Tengwei Song, Yifeng Jiao, Jianshan He, Jiaotuan Wang, Ruopeng Li, and Wei Chu. 2023. Spatio-temporal hypergraph learning for next poi recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, page 403–412, New York, NY, USA. Association for Computing Machinery.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Dingqi Yang, Daqing Zhang, Vincent W. Zheng, and Zhiyong Yu. 2015. Modeling user activity preference by leveraging user spatial temporal characteristics in lbsns. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45(1):129–142.

Song Yang, Jiamou Liu, and Kaiqi Zhao. 2022. Getnext: Trajectory flow map enhanced transformer for next poi recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, page 1144–1153. ACM.

Yifei Yang, Zouying Cao, and Hai Zhao. 2024. Laco: Large language model pruning via layer collapse. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 6401–6417.

John M. Zelle and Raymond J. Mooney. 1996. Learning to parse database queries using inductive logic programming. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence - Volume 2*, pages 1050–1055.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 4791–4800.

Lucia Zheng, Neel Guha, Javokhir Arifov, Sarah Zhang, Michal Skreta, Christopher D Manning, Peter Henderson, and Daniel E Ho. 2025. A reasoning-focused legal retrieval benchmark. In *Proceedings of the 2025 Symposium on Computer Science and Law*, pages 169–193.

# A Appendix

We organize the appendix as follows.

- Appendix A.1: dataset specifications, model abbreviations, and implementation details.
- Appendix A.2: related work on layer-wise structure, intermediate readouts, and instance-level signals.
- Appendix A.3: additional evidence for gold-option specificity and option-order sensitivity.
- Appendix A.4: model-level semantic-competition results.
- Appendix A.5: expanded pretraining and instruction-tuning results.
- Appendix A.6: auxiliary training- and inference-stage intervention results.
- Appendix A.7: boundary checks on generative candidate-continuation tasks.

## A.1 Experimental Details

All MCQA analyses use explicit answer options and compute intermediate option scores by applying the final pre-head normalization layer and the LM head to hidden states at each layer. Unless otherwise specified, persistent crossing uses $\rho = 0.8$. The main MCQA analyses cover ARC-Challenge, ARC-Easy, HellaSwag, PIQA, WinoGrande, MMLU-Med, MMLU-Geo, MedQA, MedMCQA, BarExam, BBL, and GeoBench. Generation boundary checks use GSM8K, HumanEval, and MBPP.

**Dataset specifications.** The MCQA experiments use a broad set of science, commonsense, medical, legal, and geography benchmarks, while the generation boundary check uses math and code-generation datasets with verifiable continuations.

- **ARC-Challenge and ARC-Easy.** ARC is introduced by Clark et al. (2018). It contains grade-school science multiple-choice questions. ARC-Challenge is the harder filtered split, while ARC-Easy contains questions that are more accessible to retrieval or word-overlap baselines.
- **HellaSwag.** HellaSwag is introduced by Zellers et al. (2019). It is a commonsense sentence-completion benchmark constructed with adversarial filtering. Each instance asks the model to choose the most plausible ending for a short context.
- **PIQA.** PIQA is introduced by Bisk et al. (2020). It is a two-choice physical commonsense benchmark. The questions test whether a model understands object affordances and plausible physical interactions in everyday situations.
- **WinoGrande.** WinoGrande is introduced by Sakaguchi et al. (2021). It is a two-choice commonsense coreference benchmark. The task requires resolving ambiguous pronouns using commonsense knowledge rather than surface matching.
- **MMLU-Med and MMLU-Geo.** MMLU is introduced by Hendrycks et al. (2020). MMLU-Med contains medical subject subsets, while MMLU-Geo uses the high-school geography subset. These subsets are used to test whether transition behavior changes under domain-specific knowledge demands.
- **MedQA.** MedQA is introduced by Jin et al. (2020). It is a medical exam question-answering benchmark based on professional licensing-style questions. The dataset is commonly evaluated in a multiple-choice format. In this paper, we use the English split.
- **MedMCQA.** MedMCQA is introduced by Pal et al. (2022). It contains large-scale multiple-choice questions from entrance-exam style sources. The benchmark covers multiple medical subjects and supports domain-specific MCQA evaluation.
- **BarExam.** BarExam is a legal question-answering benchmark built from bar-exam style questions and supporting legal resources (Zheng et al., 2025). It is used here as a legal-domain MCQA setting, allowing transition behavior to be evaluated outside science, commonsense, and medicine.
- **BBL.** BBL refers to BhashaBench-Legal, a comprehensive benchmark designed to rigorously evaluate AI models on Indian legal knowledge (Devane et al., 2025).
- **GeoBench.** GeoBench refers to GeoBench-LLM and related geography resources (Huang

et al., 2021; Finegan-Dollak et al., 2018; Zelle and Mooney, 1996; Huang et al., 2019; Li et al., 2025; Nguyen et al., 2016; Hamzei et al., 2020; Aghzal et al., 2024; Mirzaee and Kordjamshidi, 2022; Yamada et al., 2024; Contractor et al., 2021; Li et al., 2024; Kefalidisa et al., 2023; Shi et al., 2022; Yang et al., 2022, 2015; Yan et al., 2023; Feng et al., 2024). It focuses on geographical and spatial reasoning for LLM evaluation. We use its MCQA-formatted split to test transition behavior in geography-centered questions.

- **GSM8K.** GSM8K is introduced by Cobbe et al. (2021). It contains grade-school math word problems with final numeric answers. In the generation boundary check, candidate continuations are compared after answer verification.

- **HumanEval.** HumanEval is introduced by Chen et al. (2021). It is a code-generation benchmark with unit-test based verification. The dataset is used only as a boundary check because candidate continuations differ from fixed MCQA options.

- **MBPP.** MBPP is introduced by Austin et al. (2021). It contains Python programming tasks with tests for functional correctness. It provides a second code-generation setting for testing whether transition-style signals transfer beyond explicit answer options.

**Pretraining checkpoints.** For the pretraining analysis, we use the released OLMo-7B checkpoint trajectory (Groeneveld et al., 2024). The checkpoint names follow the original release convention. The prefix `stage1` denotes checkpoints from the first-stage pretraining trajectory, while `stage2-ingredient3` denotes checkpoints from the second-stage continuation trajectory under the corresponding released data-mixture setting. The step field is the training step within that stage, and the tokens field gives the approximate number of training tokens reported for that checkpoint.

We evaluate 16 ordered checkpoints in total. The first-stage checkpoints range from `stage1-step150-tokens1B` to `stage1-step928646-tokens3896B`, and the second-stage checkpoints range from `stage2-ingredient3-step1000-tokens5B` to `stage2-ingredient3-step11931-tokens50B`. All checkpoints are evaluated with the same MCQA prompts, scoring rule, datasets, and transition metrics, so the comparison isolates how the released pretraining trajectory changes accuracy and transition behavior.

**Responsible Use and Compute Reporting.** All public datasets and model checkpoints used in this work are used only for research evaluation and analysis. We do not collect new personal data, and we follow the original documentation, licenses, and terms of use for each dataset and model. All experiments are run with bfloat16 precision on Nvidia H100 GPUs (80GB).

## A.2 Related Work

### A.2.1 Layer-Wise Structure and Model Compression

Layer pruning reduces LLM memory and latency by removing Transformer blocks. Existing work scores layer importance with activation or weight magnitude, calibration loss, input-guided sensitivity, layer merging, or layer folding (Men et al., 2025; Qiao et al., 2025; Wang et al., 2026; Kim et al., 2024; Ma et al., 2023; Liu et al., 2024; Yang et al., 2024), and relates depth redundancy to shallow and deep functional roles (Gromov et al., 2025). These studies seek better pruning or removable layers. We instead use pruning collapse as motivation and study the decision transition it exposes under controlled data and training changes.

### A.2.2 Intermediate Readouts and Decision Dynamics

Intermediate-readout methods study how predictions form before the final layer. Linear probes test whether hidden states contain task information (Alain and Bengio, 2016), while Logit Lens and Tuned Lens project hidden states into vocabulary space (nostalgebraist, 2020; Belrose et al., 2023; Wang, 2025). Mechanistic interpretability also links abrupt capability changes to circuit or attention-head structure (Olsson et al., 2022). Our work focuses on MCQA option competition, tracking how the gold option separates from its strongest competitor and summarizing the trajectory by transition depth, strength, stability, and uncertainty.

### A.2.3 Instance-Level Signals for Training and Inference

Instance-level signals often guide training or inference. Curriculum learning and hard-example mining use difficulty, loss, or uncertainty to select

| Family | Full model name | Abbreviation |
|---|---|---|
| Qwen | Qwen3-0.6B-Base / Qwen3-0.6B | Q0.6B-B / Q0.6B-I |
| Qwen | Qwen3-1.7B-Base / Qwen3-1.7B | Q1.7B-B / Q1.7B-I |
| Qwen | Qwen3-4B-Base / Qwen3-4B | Q4B-B / Q4B-I |
| Qwen | Qwen3-8B-Base / Qwen3-8B | Q8B-B / Q8B-I |
| Qwen | Qwen3-14B-Base / Qwen3-14B | Q14B-B / Q14B-I |
| Llama | Llama-3-8B / Llama-3-8B-Instruct | L3-8B-B / L3-8B-I |
| Llama | Llama-3.1-8B / Llama-3.1-8B-Instruct | L3.1-8B-B / L3.1-8B-I |
| Llama | Llama-3.2-1B / Llama-3.2-1B-Instruct | L3.2-1B-B / L3.2-1B-I |
| Llama | Llama-3.2-3B / Llama-3.2-3B-Instruct | L3.2-3B-B / L3.2-3B-I |
| Llama | Llama-3.1-70B / Llama-3.1-70B-Instruct | L3.1-70B-B / L3.1-70B-I |

Table 6: Model abbreviations used in figures and tables. Suffix B denotes base models and I denotes instruction-tuned models.

| Component | Setting | Value |
|---|---|---|
| Shared forward pass | dtype | bfloat16 |
| Shared forward pass | prompt style | plain MCQA prompt |
| Shared forward pass | option scoring | answer-letter token variants |
| Shared forward pass | default seed | 42 |
| A/C/D analyses | default per-task limit | 300 examples when available |
| B/E/F analyses | evaluation limit | 0, using the full constructed condition file |
| D pretraining | checkpoints | 16 ordered checkpoints from the same model series |
| CATT training | base model | Qwen3-8B-Base |
| CATT training | epochs / learning rate / effective batch size | 2 / $2 \times 10^{-5}$ / 8 |
| CATT training | LoRA rank / alpha / dropout | 8 / 16 / 0.05 |
| CATT objective | $\lambda_{\mathrm{tr}}$, $\lambda_{\mathrm{hr}}$ | 0.05 and 0.02 for CATT; both 0 for SFT |
| TAI routing | strong-model budgets | 10%, 15%, 20%, 30%, 40%, 50% |
| TAI routing | learned router | logistic regression over transition and confidence features |
| Generation boundary | tasks | GSM8K, HumanEval, MBPP |

Table 7: Implementation details for the main controlled analyses and interventions. Forward-only analyses do not update model parameters.

training examples (Bengio et al., 2009; Shrivastava et al., 2016; Swayamdipta et al., 2020). Adaptive inference and routing reduce cost with early exits, confidence thresholds, or stronger-model routing (Teerapittayanon et al., 2016; Xin et al., 2020; Liu et al., 2020; Chen et al., 2023). CATT and TAI do not claim new general solutions to these problems. They test whether layer-wise transition features provide usable evidence beyond final-layer outcomes.

### A.3 Gold-Option Specificity and Option-Order Boundary

Table 8 gives the numerical counterpart of Figure 2. The positive gaps show that true gold options cross earlier and accumulate larger positive margins than pseudo-gold controls across datasets and model families. This supports the main-text claim that PC and PA are tied to the annotated answer rather than to arbitrary option-level drift.

Figure 5 reports option-order perturbations on M-Med and MedQA. The median PC range across reordered variants is 17.71 layers, and the mean PA standard deviation is 23.23. This result does not weaken the answer-specificity evidence, but it defines an important boundary: transition statistics should be compared under paired or matched prompt conditions.

### A.4 Model-Level Semantic Competition

Figure 6 expands Figure 3 by retaining individual model settings instead of only family-level summaries. The heatmaps show the same hard-vs-noise direction across Llama and Qwen settings on all six datasets, with the largest shifts on M-Geo and visible but smaller shifts on Bar and ARC-C. This supports the main-text conclusion that semantic competition delays and weakens transition, while also showing that the effect size is not invariant across tasks or model families.

### A.5 Expanded Training-Stage Variation Results

**Pretraining.** Figure 7 expands the compact main-text correlation view with representative checkpoint trajectories. The curves show that accuracy generally improves with pretraining progress,

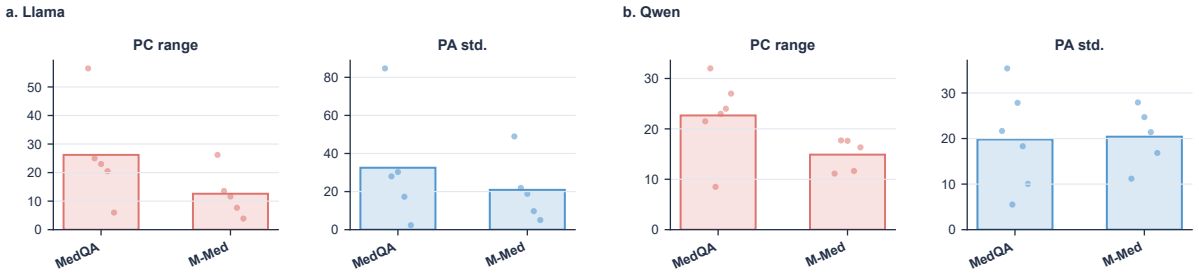


Figure 5: Option-order sensitivity as a boundary condition. Panels report permutation-induced transition variation for Llama and Qwen.

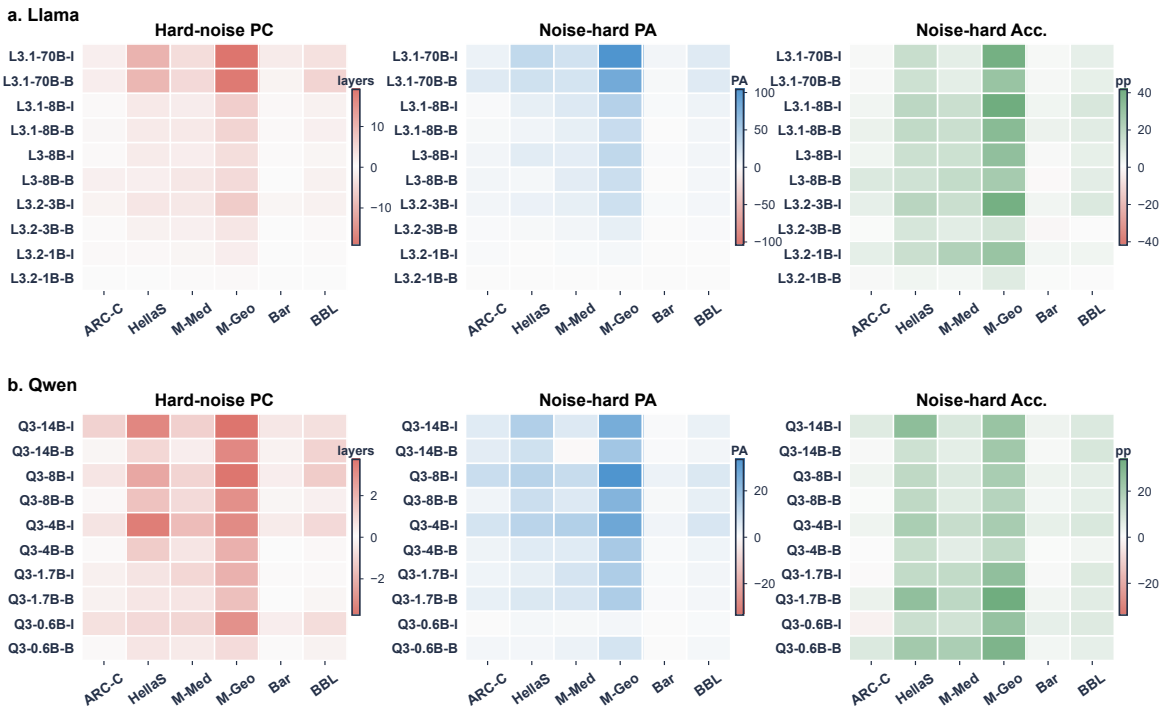


Figure 6: Model-level semantic-competition gaps across six datasets. Panels retain individual Llama and Qwen settings.

while PC tends to move earlier but can fluctuate locally. This supports the main-text conclusion that pretraining reshapes transition timing without making PC a simple restatement of final accuracy.

**Instruction tuning.** Figure 8 provides the PC and PA heatmaps corresponding to the numerical summary in Table 1. Figure 9 reports final-entropy changes for the same paired base and instruction-tuned models. Together, these figures support the main-text conclusion that instruction tuning reliably sharpens the final distribution, while its effects on transition timing and accumulated margin depend on the model family.

### A.6 Auxiliary Intervention Results

The CATT result in Table 3 is included as a controlled training-time validation against SFT under the same hard-example exposure, not as a broad claim that CATT dominates hard-example training in general. Table 9 reports the favorable subset from the expanded CATT versus LoRA-mix comparison. We select settings where CATT does not reduce accuracy, increases PA, and moves PC earlier under the 13-option average. The full local report contains all dataset-model pairs, including cases where the advantage is weaker or absent.

Figure 10 reports budget curves for two representative routing settings, and Figure 11 compares routing-risk AUROC. These results support the main-text conclusion that predicted-answer tran-

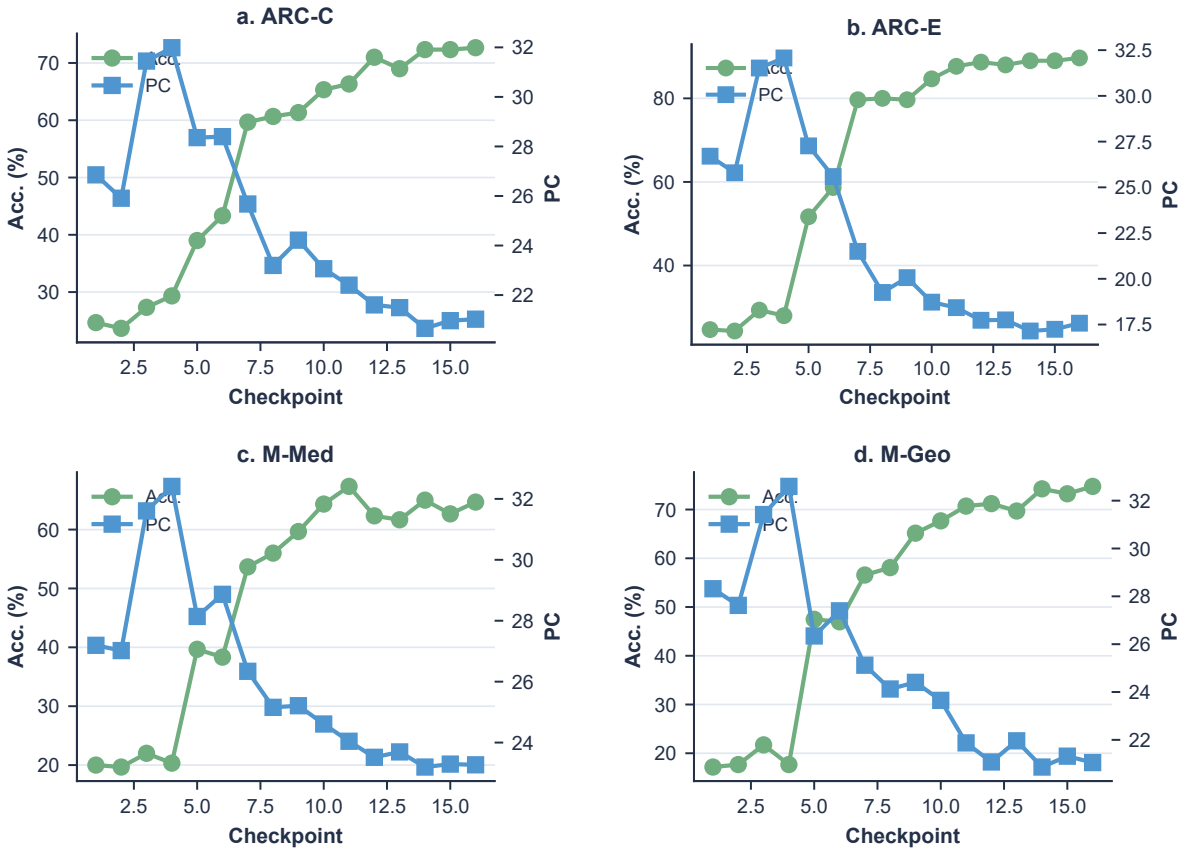


Figure 7: Checkpoint-level pretraining dynamics for representative datasets.

sition features provide deployable risk evidence, while final-confidence baselines remain competitive in some routes.

### A.7 Generative Candidate-Continuation Boundary Check

The generative boundary check replaces explicit MCQA options with candidate continuations. For GSM8K, correctness is determined by answer verification; for HumanEval and MBPP, correctness is checked with code tests or reference-backed verification. This setting is therefore not a direct open-ended generation benchmark, but it tests whether a transition-style preference signal can still be measured when the competing objects are continuations rather than fixed answer options.

Tables 10 and 11 summarize the latest aggregate results from the G experiment. Across both model families, hard wrong continuations reduce the final chosen-win rate on all three tasks and usually delay PC. PA is less uniform, especially on GSM8K for Qwen and MBPP for Llama, so the result should be read as transfer-oriented boundary evidence rather than a full claim about generation quality.

| Dataset | Llama | | Qwen | |
|---|---|---|---|---|
| | ΔPC | ΔPA | ΔPC | ΔPA |
| ARC-C | 18.05 | 81.69 | 11.63 | 58.40 |
| ARC-E | 19.92 | 96.73 | 12.93 | 67.11 |
| HellaS | 15.42 | 60.01 | 9.46 | 46.31 |
| PIQA | 18.63 | 66.13 | 11.86 | 32.78 |
| WinoG | 9.58 | 31.67 | 3.86 | 10.35 |
| M-Med | 16.72 | 74.16 | 9.49 | 48.36 |
| M-Geo | 18.16 | 81.34 | 10.06 | 54.69 |
| MedQA | 11.94 | 46.02 | 4.90 | 20.27 |
| MedMCQ | 12.58 | 49.19 | 7.45 | 34.96 |
| Bar | 11.86 | 38.97 | 5.74 | 24.20 |
| BBL | 12.79 | 48.70 | 8.26 | 38.39 |
| GeoB | 4.70 | 16.01 | 3.67 | 12.49 |
| Mean | 14.19 | 57.55 | 8.27 | 37.36 |

Table 8: Dataset-level gold-option specificity. ΔPC is pseudo minus real, and ΔPA is real minus pseudo.

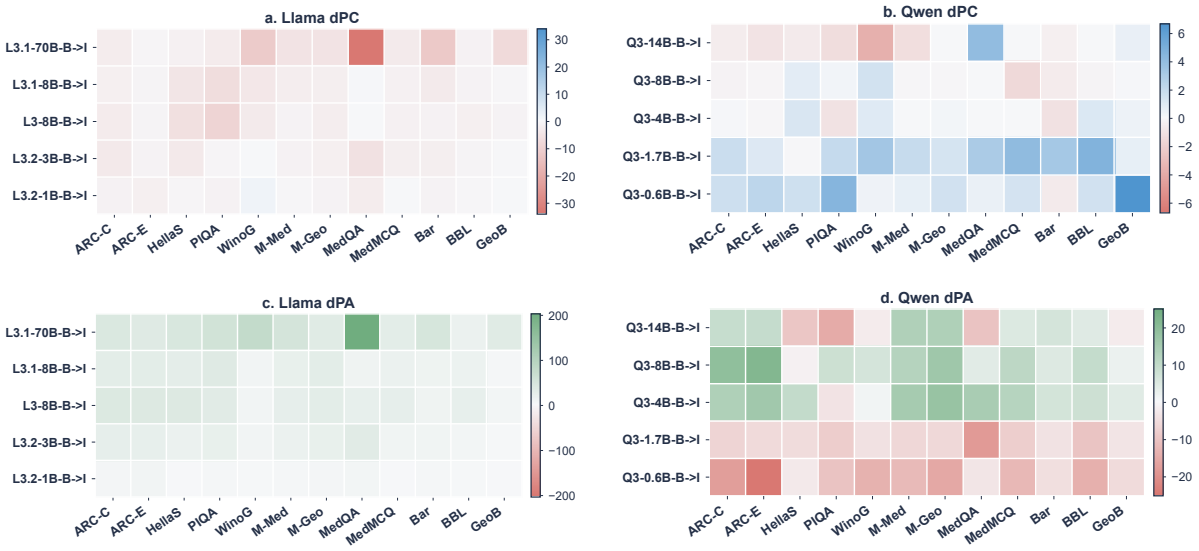


Figure 8: Base-to-instruct transition changes by model family.

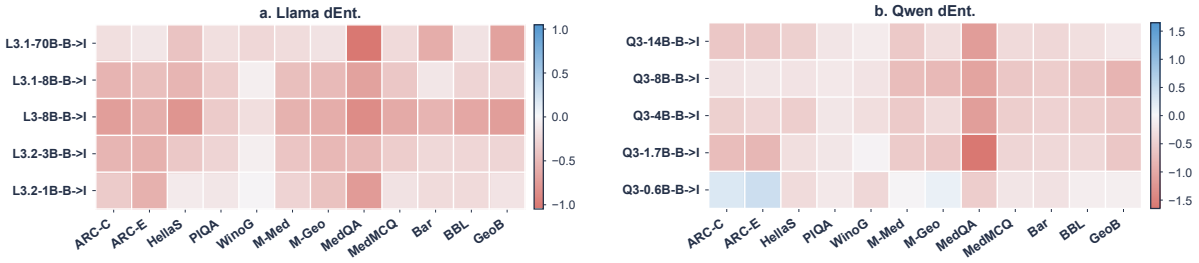


Figure 9: Final-layer entropy changes after instruction tuning.

| Dataset | Model | ΔAcc | ΔPA | ΔPC |
|---|---|---|---|---|
| M-Geo | Q8B-B | 0.00 | **13.55** | -2.70 |
| M-Med | L3-8B-B | **1.95** | 2.70 | -0.43 |
| M-Geo | Q14B-B | 0.00 | 2.28 | -1.37 |
| M-Geo | L3.1-8B-B | 0.00 | 0.76 | -0.12 |
| M-Geo | Q0.6B-B | 0.33 | 0.47 | -0.60 |
| HellaS | Q14B-B | 0.29 | 0.08 | **-3.09** |

Table 9: CATT results from the expanded LoRA-mix comparison. Values are CATT-minus-LoRA differences averaged over 13 option settings. Rows are included only when ΔAcc is non-negative, ΔPA is positive, and ΔPC is negative.

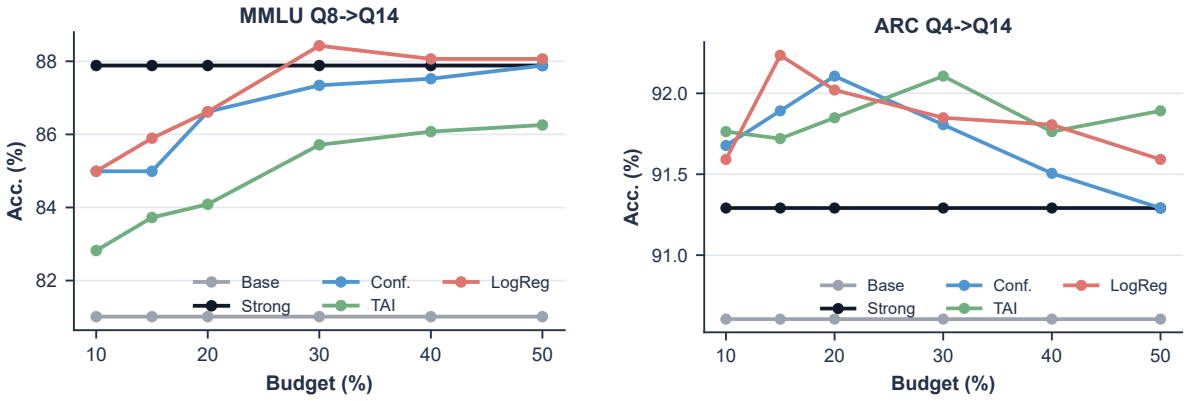


Figure 10: Budget curves for transition-aware adaptive inference.

| Task | Settings | Records | R-Win | H-Win | ΔWin | R-PC | H-PC | ΔPC | R-PA | H-PA | ΔPA |
|---|---|---|---|---|---|---|---|---|---|---|---|
| GSM8K | 9 | 329 | 45.67 | 21.44 | 24.23 | 26.81 | 28.97 | 2.16 | 29.59 | 26.70 | -2.89 |
| HumanEval | 9 | 774 | 39.05 | 14.62 | 24.43 | 24.95 | 27.43 | 2.48 | 9.40 | 7.67 | -1.73 |
| MBPP | 9 | 583 | 8.79 | 1.94 | 6.85 | 29.59 | 29.96 | 0.37 | 10.31 | 13.51 | 3.20 |
| Mean | – | – | 31.17 | 12.67 | 18.50 | 27.12 | 28.79 | 1.67 | 16.43 | 15.96 | -0.47 |

Table 10: Llama-family candidate-continuation transition results.

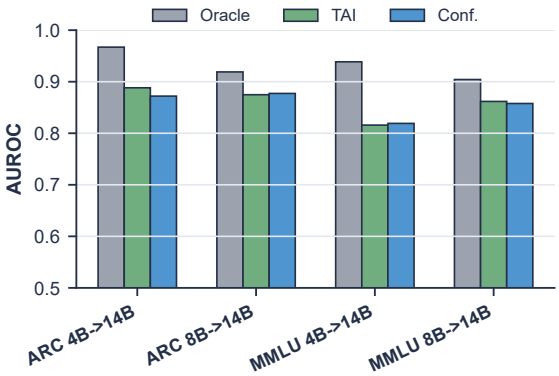


Figure 11: Routing-risk AUROC for TAI and comparison routers.

| Task | Settings | Records | R-Win | H-Win | ΔWin | R-PC | H-PC | ΔPC | R-PA | H-PA | ΔPA |
|---|---|---|---|---|---|---|---|---|---|---|---|
| GSM8K | 11 | 604 | 37.13 | 17.46 | 19.67 | 24.70 | 26.02 | 1.32 | 46.35 | 51.53 | 5.18 |
| HumanEval | 10 | 986 | 26.44 | 11.18 | 15.26 | 28.61 | 29.88 | 1.27 | 26.68 | 18.06 | -8.62 |
| MBPP | 10 | 990 | 14.92 | 3.17 | 11.75 | 32.72 | 33.62 | 0.90 | 12.55 | 9.74 | -2.81 |
| Mean | – | – | 26.16 | 10.60 | 15.56 | 28.68 | 29.84 | 1.16 | 28.53 | 26.44 | -2.08 |

Table 11: Qwen-family candidate-continuation transition results.