BEFORE THE QUESTION ENDS: PROVISIONAL ANSWER FORMATION IN LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

A language model may form an answer preference before a question is complete. Whether this early preference causally influences its final decision after question completion remains unclear. In binary Yes/No question answering, we introduce the preference score (PS) as a behavioral measure of provisional answer preference. Across four models, final-answer accuracy is lower when the correct answer conflicts with the early preference than when it aligns with that preference. To examine the causal role of provisional answer preference, we intervene on question-prefix representations and find that they carry this preference and influence final-answer computation. At the circuit level, question-prefix circuits selected solely for their contribution to final-answer computation also carry provisional preference. Steering against the early preference and suppressing these circuits both make final decisions less likely to follow the initially favored answer. Our findings provide causal evidence that preferences formed before question completion bias final decisions even after the complete question is available.

1 INTRODUCTION

People can begin preparing an answer before a question is complete (Bögels et al., 2015; Bott et al., 2017). Such preparation can affect comprehension of the remaining question. In a spoken question-answering experiment, listeners were more likely to overlook semantic errors when they could begin planning a response before those errors were presented (Barthel, 2021). Figure 1 illustrates how such early preparation can be misleading. Although the correct answer depends on whether the question ends with "football" or "basketball", the passage and question prefix may already suggest "Yes". Like people, autoregressive language models build up an interpretation of a question incrementally. This motivates us to ask whether models form an answer preference before question completion, and whether this early preference influences their final decision after the complete question is provided.

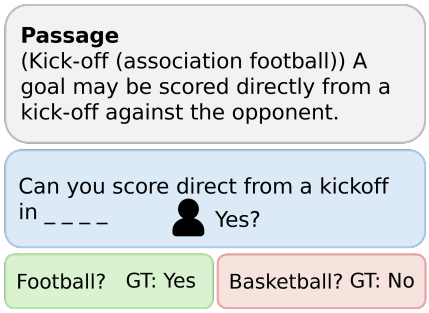


Figure 1: **Same question prefix, opposite correct answers.** An NP-BoolQ shared-prefix pair (Section 4.2). The questions share the same passage and the prefix, but their final words lead to opposite ground-truth (GT) answers.

Prior work shows that models can perform well on partial questions (Sugawara et al., 2018; Sugiura et al., 2023), or even with no question at all (Sen & Saffari, 2020). However, it remains underexplored whether models form an early preference during question processing, and whether such a preference influences their final decisions. We study this question through *provisional answer preference*, which we define as a model's preference for one candidate answer before the question is complete. By focusing on binary Yes–No question answering, we can directly measure how strongly the model favors one of two fixed candidate answers over the other.

To quantify provisional answer preference, we introduce the *preference score* (PS) as a behavioral measure. Using PS, we find that provisional answer preferences predict the models' final decisions across four language models. We further examine a shared-prefix setting in which two questions share the same passage and question prefix but require opposite answers (Figure 1). Although both

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

questions share the same provisional preference, final-answer accuracy differs significantly depending on whether the correct answer aligns or conflicts with that preference.

The observed accuracy gap motivates us to investigate whether provisional answer preference causally influences the final decision. We therefore intervene on hidden states formed before question completion. Swapping these states transfers provisional preference between inputs, with corresponding effects on final-answer computation. Steering along a direction learned from provisional preference scores biases final-answer computation in the corresponding direction. In the shared-prefix setting, steering against the original preference reduces the accuracy gap between the two question groups. These changes provide causal evidence that representations associated with provisional preference contribute to final-answer computation.

At the circuit level, we examine how the computations underlying hidden states formed before question completion contribute to provisional answer preference. We select circuits involved in forming these hidden states based on their contribution to final-answer computation and find that they also carry provisional answer preference. In the shared-prefix setting, suppressing these circuits produces the same accuracy pattern as steering against the original provisional preference. These accuracy changes indicate that the selected circuits contribute to the model's tendency to follow its original provisional preference.

Our analyses suggest that language models form provisional answer preferences that causally contribute to final-answer computation. Like people who begin preparing an answer before a question is complete, models develop these preferences during question processing. Their influence persists after question completion, biasing final decisions toward the initially favored answer.

2 Related Work

Question Answering from Partial Questions. Prior work shows that models can answer some questions correctly even when given only part of the question (Sugawara et al., 2018; Sen & Saffari, 2020). Sugiura et al. (2023) study answer prediction from partial questions and evaluate confidence estimates for those predictions. In visual question answering, models often give the same answers as they do with complete questions after seeing only half of the question (Agrawal et al., 2016). These studies characterize predictions from partial questions but leave open whether they causally influence decisions made after question completion. Our work focuses on whether provisional answer preferences formed from partial questions causally influence those final decisions.

Internal Answer Representations. Recent work examines answer-related information before an explicit answer is generated. Zhang et al. (2026) show that finite-answer preferences can be recovered from hidden states and track how these preferences stabilize during response generation. Cox et al. (2026) decode semantic answers from activations preceding chain-of-thought (CoT) generation and steer subsequent generation. Both studies analyze models after the full question has been provided. Ashok & May (2025) probe input hidden states to predict eventual CoT answers and examine how this predictability changes across input positions. However, whether answer preferences formed before question completion causally influence final decisions remains underexplored. We test this causal role through interventions on hidden states formed before question completion, with the remaining question processed afterward.

Automated Circuit Discovery. Mechanistic circuit analysis identifies computational subgraphs supporting model behavior (Wang et al., 2022). Automatic Circuit DisCovery (ACDC) uses activation patching to prune edges (Conmy et al., 2023), while Edge Attribution Patching (EAP) approximates edge-level intervention effects using gradients (Syed et al., 2024). Position-aware Edge Attribution Patching (PEAP) distinguishes connections by token position and aggregates attribution scores across variable-length inputs using corresponding token spans (Haklay et al., 2025). Using PEAP, we select circuits based solely on their contribution to final-answer computation and examine their role in provisional answer preference before question completion.

3 Defining and Measuring Provisional Answer Preference

We study whether an autoregressive language model develops an answer preference while processing a question and whether this provisional preference causally contributes to its final answer. Our

primary setting is passage-based binary question answering, which provides a controlled answer space for measuring both provisional and final answer preferences. Each example consists of a passage, a question, and a Yes/No ground-truth (GT) answer.

Provisional Answer Preference. Let P denote the passage and T the number of tokens in the question. For $0 \leq t \leq T$, let Q_t denote the question prefix consisting of the first t tokens. At $t = 0$, no question is provided, whereas at $t = T$, the complete question is provided. Given P and Q_t, *answer preference* refers to the model's tendency to favor one candidate answer over the other. For $0 \leq t < T$, this preference is *provisional*, since the complete question is not yet available and the preference can in principle be revised as further question tokens arrive.

Preference Score. Let $\mathcal{Y}$ and $\mathcal{N}$ denote the sets of single-token surface forms of Yes and No, respectively (Appendix A.5). At each question position, we construct the input $X_t = \mathrm{concat}(P,\, Q_t,\, I_{\text{ans}})$, where I_{ans} is a fixed instruction requesting a one-word Yes or No answer (Appendix A.3). For a token v, let $z_v(X_t)$ denote its next-token logit given X_t. We define the *preference score* at position t, PS_t, as the difference between the aggregated next-token logits for Yes and No:

$$\mathrm{PS}_t = \log \sum_{v \in \mathcal{Y}} \exp z_v(X_t) - \log \sum_{v \in \mathcal{N}} \exp z_v(X_t), \qquad 0 \leq t \leq T. \tag{1}$$

This score provides a behavioral measure of answer preference, with positive values favoring Yes and negative values favoring No. For $0 \leq t < T$, PS_t measures provisional answer preference and is referred to as the *provisional preference score*. At $t = T$, PS_T is the final-answer margin, and we use its sign to determine the model's final decision.

4 Answer Preferences Before Question Completion

In this section, we use PS_t to examine how provisional answer preferences relate to the model's final decisions and accuracy. We first track PS_t across question positions to quantify how well it predicts the final decision. We then compare final-answer accuracy on shared-prefix questions whose correct answers agree with or conflict with the provisional preference.

Experimental setup. Our main experiments use instruction-tuned language models from four families at broadly comparable parameter scales: Qwen3-8B (Yang et al., 2025), Gemma-3-12B-IT (Team et al., 2025), Llama-3.1-8B-Instruct (Grattafiori et al., 2024), and Phi-4 (Abdin et al., 2024). For brevity, we refer to Gemma-3-12B-IT and Llama-3.1-8B-Instruct as Gemma-3-12B and Llama-3.1-8B, respectively, throughout the paper. We conduct our primary analyses on CONDAQA (Ravichander et al., 2022) and NP-BoolQ (Khashabi et al., 2020). For both datasets, we use passage–question instances with binary Yes/No GT answers. CONDAQA pairs the same question with several types of contrastive passage variants, including meaning-preserving human-written paraphrases, negation-scope changes, and negation reversals. NP-BoolQ extends BoolQ (Clark et al., 2019) with human-written natural perturbations of questions, including perturbations that change the correct answer. Further details on datasets, prompt templates, and implementation settings are provided in Appendix A.

4.1 Increasing Predictive Strength of Provisional Answer Preferences

We first examine whether provisional answer preferences predict the model's final decision. For each example in the CONDAQA validation split, we select prefixes at relative positions of 0%, 20%, 40%, 60%, 80%, and 100% of the question, measured by word count. We measure PS_t at the corresponding token positions t. Here, 0% denotes the empty question prefix and 100% the complete question. Following Bauer et al. (2026), we use AUROC for this binary prediction task. The prediction target is the model's final decision, determined by the sign of PS_T. At each position, AUROC quantifies how well PS_t discriminates between the model's final Yes and No decisions.

Figure 2 shows that AUROC is close to chance level at 0% and increases as more question words are provided. At 100%, AUROC is 1.0 because the prediction target is defined by the sign of PS_T. Thus, the preference measured from a partial question already helps predict the model's answer to the complete question. These results show that provisional preference scores become more predictive of final decisions as more question information becomes available.

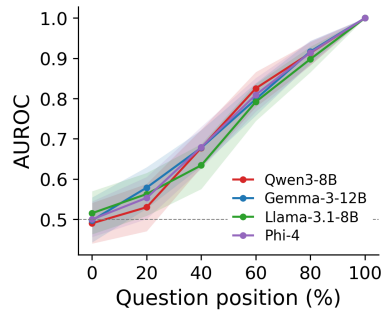


Figure 2: Final-decision AUROC of PS_t on CONDAQA. Shading denotes cluster-bootstrap 95% confidence intervals (CIs).

Table 1: Final-answer accuracy (%) on NP-BoolQ shared-prefix pairs. Aligned/Conflicting indicates whether the shared-prefix preference agrees with/opposes the GT answer. Gap is aligned minus conflicting accuracy (pp).

Model	Aligned	Conflicting	Gap
Qwen3-8B	96.1	82.4	13.7
Gemma-3-12B	96.6	76.9	19.7
Llama-3.1-8B	95.0	75.1	19.9
Phi-4	96.6	76.5	20.1

4.2 Models Are Less Accurate When Early Preferences Conflict with GT

We analyze questions that share the same passage and prefix but require opposite answers depending on the remaining words, as illustrated in Figure 1. Specifically, we construct *NP-BoolQ shared-prefix pairs* from original questions and their human-written perturbations. We retain pairs whose longest common prefix covers at least 80% of the words in each question. After removing duplicates and invalid prefixes, the dataset contains 854 pairs from 419 passage clusters. Construction details are provided in Appendix A.2.

For each pair, we measure PS_t once at the end of its shared prefix. The two questions share the same passage and prefix, so the same PS_t applies to both. We categorize each question according to whether its GT answer matches the answer favored by PS_t:

- **Aligned questions:** The GT answer matches the answer favored by PS_t.
- **Conflicting questions:** The GT answer is opposite to the answer favored by PS_t.

Each pair therefore contains one aligned and one conflicting question. We then evaluate the model on each complete question. We denote the *accuracy gap* as the difference in final-answer accuracy on aligned and conflicting questions. Positive values indicate higher accuracy on aligned questions.

Table 1 shows higher accuracy on aligned questions across all four models. Accuracy on aligned questions ranges from 95.0% to 96.6%, whereas accuracy on conflicting questions ranges from 75.1% to 82.4%. The resulting gaps are 13.7–20.1 percentage points (pp). These results suggest that models may still follow their early preference even when the rest of the question requires the opposite answer.

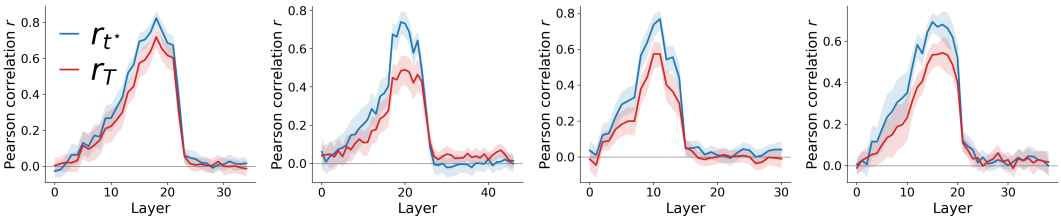
These findings motivate the hypothesis that provisional answer preferences formed during question processing causally bias final decisions. In the following sections, we test this hypothesis through interventions on question-prefix representations and circuits.

5 Causal Role of Provisional Answer Preference

In this section, we test whether provisional answer preferences causally bias final decisions. Unless otherwise specified, we intervene on question-token representations up to $t^\star$, the token position corresponding to 80% of the question's word count. The passage and question remain unchanged during each intervention.

5.1 Activation Swapping of Question-Prefix Representations

We investigate whether question-prefix hidden states carry provisional answer preference and influence final-answer computation through activation swapping (Geiger et al., 2021). We construct paired inputs from an original passage P and its meaning-preserving paraphrase P', both drawn from the CONDAQA training split. The inputs in each pair share the same question and GT answer.



(a) Qwen3-8B (b) Gemma-3-12B (c) Llama-3.1-8B (d) Phi-4

Figure 3: **Effects of activation swapping on answer preferences.** Layer-wise Pearson correlations between $\mathrm{PS}_{t^\star}$ differences within CONDAQA paraphrase pairs and changes in $\mathrm{PS}_{t^\star}$ ($r_{t^\star}$, blue) and PS_T (r_T, red) after activation swapping. Shaded regions denote cluster-bootstrap 95% CIs.

Despite preserving the passage meaning, the paraphrase alters the hidden states of the same question tokens, resulting in different provisional preference scores for the two inputs. We swap their question-prefix hidden states to examine how the intervention affects provisional answer preference and the model's final decision.

For N paraphrase pairs (P_j, P_j'), hidden states are swapped at corresponding question-token positions up to $t^\star$, one layer at a time. We denote a swap as $A \leftarrow B$, where the question-prefix hidden states for A are replaced with those for B. Performing both $P_j \leftarrow P_j'$ and $P_j' \leftarrow P_j$ for each pair yields a set $\mathcal{S}$ of $2N$ swaps. For each swap $s = (A \leftarrow B) \in \mathcal{S}$, we define the pre-swap difference in provisional preference between the two passages and the change in A's score induced by the swap as

$$\delta(s) = \mathrm{PS}_{t^\star}^{B} - \mathrm{PS}_{t^\star}^{A}, \qquad \Delta_t(s) = \mathrm{PS}_{t}^{A \leftarrow B} - \mathrm{PS}_{t}^{A}, \tag{2}$$

where $\mathrm{PS}_t^{A \leftarrow B}$ denotes the score of A after swapping. For each model and layer, we compute the Pearson correlation between these quantities over all $2N$ swaps:

$$r_t = \mathrm{corr}_{s \in \mathcal{S}}\big(\delta(s),\, \Delta_t(s)\big), \qquad t \in \{t^\star, T\}.$$

$r_{t^\star}$ relates the pre-swap provisional preference difference between B and A to the change in A's provisional preference induced by $A \leftarrow B$. r_T indicates whether the association with the pre-swap provisional preference difference extends to the change in A's final-answer margin induced by $A \leftarrow B$. Further details are provided in Appendix C.1.

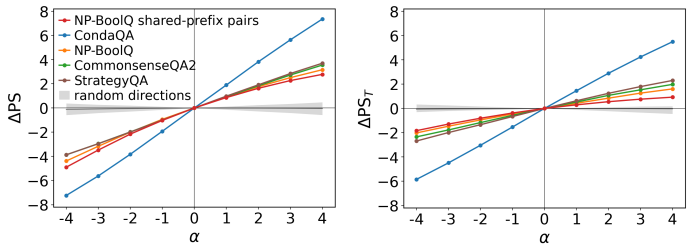
Figure 3 shows positive correlations for $r_{t^\star}$ across all four models. The maximum correlations across layers range from 0.69 to 0.83. Swap-induced changes in $\mathrm{PS}_{t^\star}$ correlate positively with the pre-swap provisional preference differences, indicating that question-prefix hidden states encode provisional answer preference.

Positive correlations are also observed for r_T across all four models, with maximum correlations across layers ranging from 0.49 to 0.72. The pre-swap provisional preference differences also correlate with swap-induced changes in the final-answer margin PS_T, showing that the influence of provisional preference extends to final-answer computation. Activation swapping therefore provides causal evidence that provisional answer preference carried by question-prefix hidden states influences the model's final decision.

5.2 Activation Steering of Provisional Answer Preference

We investigate whether provisional answer preference causally biases final-answer computation through activation steering (Turner et al., 2024). We first test whether steering question-prefix hidden states shifts the provisional preference score. We then examine whether the same intervention shifts the final-answer margin in the corresponding direction.

Evaluation datasets. We evaluate steering on the CONDAQA test split, NP-BoolQ, CommonsenseQA 2.0, and StrategyQA. For each of these four datasets, we evaluate 300 examples. CommonsenseQA 2.0 (Talmor et al., 2022) contains Yes/No questions and statements about commonsense knowledge, including physical, temporal, and causal relations. StrategyQA (Geva et al., 2021) contains Yes/No questions that require combining information through reasoning steps that are not explicitly stated in the question. We provide the passage for CONDAQA and NP-BoolQ, whereas



(a) Provisional preference score (b) Final-answer margin

Figure 4: **Effects of activation steering on answer preference.** Changes in (a) provisional preference score and (b) final-answer margin. Gray shading shows the range across random directions.

Table 2: **Steering against provisional preference.** Accuracy on NP-BoolQ shared-prefix pairs for Qwen3-8B. Steering opposes each pair's original provisional preference, with strength $|\alpha|$. Accuracies are percentages, and gaps are percentage points.

$\|\alpha\|$	Aligned	Conflicting	Gap
0	96.1	82.4	13.7
2	95.0	83.8	11.1
4	93.8	84.7	9.1

CommonsenseQA 2.0 and StrategyQA are evaluated using question-only inputs. These settings allow us to test whether a direction learned from passage paraphrases also influences answer preferences when no passage is provided.

We also evaluate steering on all 854 NP-BoolQ shared-prefix pairs introduced in Section 4.2. These pairs allow us to test whether steering a shared prefix that does not yet determine the correct answer biases final-answer computation. Evaluation splits and sampling details are provided in Appendix A.

Direction learning. Following Section 5.1, we select a layer ℓ in the peak region of $r_{t^\star}$ for each model. We learn the steering direction at this layer using the CONDAQA training paraphrase pairs. For an input containing passage $S \in \{P, P'\}$, let $h^S_{\ell,j}$ denote the residual-stream output at layer ℓ and question-token position j. We define $\bar{h}^S_\ell$ as the mean of these outputs over question-token positions up to $t^\star$. We fit a ridge regression to predict the provisional preference difference, $\mathrm{PS}^{P'}_{t^\star} - \mathrm{PS}^{P}_{t^\star}$, from the corresponding difference in averaged hidden states, $\bar{h}^{P'}_\ell - \bar{h}^{P}_\ell$. We normalize the regression coefficients to obtain a unit steering vector v_{PS} and orient it toward increasing $\mathrm{PS}_{t^\star}$. Regularization selection and further training details are provided in Appendix C.2.

Steering and measurements. At layer ℓ, we apply the additive intervention:

$$h_{\ell,j} \leftarrow h_{\ell,j} + \alpha s\, v_{\mathrm{PS}}, \qquad 1 \le j \le t^\star, \tag{3}$$

where α controls the steering strength and s is the standard deviation of projections onto v_{PS} over all question-prefix tokens in the training inputs. For NP-BoolQ shared-prefix pairs, the intervention is applied over the shared question prefix, identically for both questions.

We evaluate steering at $\alpha \in \{0, \pm1, \pm2, \pm3, \pm4\}$. We measure provisional preference at $t^\star$, or at the end of the shared prefix for NP-BoolQ shared-prefix pairs. For each input, we denote changes in provisional preference and the final-answer margin relative to $\alpha = 0$ by $\Delta\mathrm{PS}$ and $\Delta\mathrm{PS}_T$, respectively. We report mean changes across examples for each α. As controls, we evaluate 30 random unit directions at the same nonzero strengths. Further implementation and validation details are provided in Appendix C.2.

Effects on answer preferences. Figure 4 shows that steering along v_{PS} shifts both the provisional preference score and the final-answer margin in the same direction across all five evaluation settings for Qwen3-8B. Positive steering increases both scores, whereas negative steering decreases them. The magnitudes of these shifts grow with intervention strength. The direction learned on CONDAQA produces these effects without retraining on NP-BoolQ and the question-only datasets CommonsenseQA 2.0 and StrategyQA.

It is notable that these effects also occur on NP-BoolQ shared-prefix pairs, where the shared prefix alone does not determine the correct answer. Nevertheless, steering produces directional shifts in both scores that exceed those from random directions at matched strengths. These results indicate that an early answer preference can influence final-answer computation even when the correct answer depends on words that appear later in the question. Results for the other three models are provided in Appendix C.2.

Shared-prefix accuracy. We next examine how steering against the original provisional preference affects final-answer accuracy on NP-BoolQ shared-prefix pairs. For each pair, we choose the sign of α to oppose the preference measured without steering. We apply the same intervention to both questions at magnitudes $|\alpha| \in \{0, 2, 4\}$.

Table 2 shows that stronger steering against the original preference narrows the final-answer accuracy gap between aligned and conflicting questions. The steering vector is learned from question-prefix hidden states, which cannot incorporate later question words or the appended Yes/No answer request under causal attention. Applying this vector only at prefix positions nevertheless shifts final-answer margins and reduces the gap, supporting the hypothesis that provisional preference represented during question processing causally biases final decisions.

6 Circuit Mechanisms of Provisional Answer Preference

Section 5 shows that question-prefix hidden states carry provisional answer preference and intervening on these states affects final-answer computation. We next examine how circuits computing these representations contribute to provisional answer preference. For circuit discovery and swapping, we use the CONDAQA paraphrase setting introduced in Section 5.1. We construct these circuits by selecting connections based on their attribution to the final-answer margin. We then investigate whether the selected circuits also carry provisional answer preference through circuit swapping. To examine how these circuits contribute to preference scores and final-answer accuracy, we suppress their contributions through circuit suppression.

Circuit-analysis setup. We use position-aware edge attribution patching (PEAP) (Haklay et al., 2025) to estimate edge contributions to PS_T at each token position. This allows us to focus on connections used to compute representations at question-prefix positions. The circuit analyses are conducted on Qwen3-8B and Llama-3.1-8B. We use disjoint sets of 400, 100, and 400 CONDAQA paraphrase pairs for circuit discovery, circuit-size selection, and evaluation, respectively.

PEAP represents model computations as a graph consisting of nodes and edges. Each node represents the computation of a model component, such as an attention head or an MLP, at a particular token position. An edge $e = (u, v)$ denotes a connection from node u to node v, and a_e denotes the activation passed through this edge. Building on edge attribution patching, PEAP assigns each edge an attribution score that estimates the change in PS_T caused by patching the edge with activations from the paired input. The attribution score is a first-order approximation given by the inner product of the resulting activation change and the gradient of PS_T with respect to the activation used to measure that change. The gradient is computed for the input being patched, before any activations are replaced. For residual-stream connections, a_e is the output of u passed to v at the same token position, and the activation change is measured at the input of v. For attention connections, a_e is the value vector at u's position weighted by the attention weight assigned to that position by head v, before the head's output projection. Here, the activation change is measured at the output of head v after recomputing attention with the relevant query, key, or value pathway patched.

Question-prefix circuit. As illustrated in Figure 5, we consider edges whose receiving positions lie within the question prefix as candidates for the question-prefix circuit. This prefix extends to $t^\star$, corresponding to the first 80% of the question's words. We aggregate each candidate edge's attribution score across the circuit-discovery pairs and rank the edges by the absolute aggregated scores. We define $C(f)$ as the set of candidate edges ranked in the top $f\%$. We consider $f \in \{0.03, 0.1, 0.3, 1\}$ for circuit swapping and use $C(0.3)$ for circuit suppression. Further details on PEAP and attribution aggregation are provided in Appendix D.1.

6.1 Swapping of the Question-Prefix Circuit

We test whether $C(f)$ carries provisional answer preference using circuit swapping implemented through edge-level activation patching (Conmy et al., 2023). We replace selected edge contributions with those from the paired paraphrase input and compare provisional and final score changes.

Swapping and measurements. For M CONDAQA evaluation pairs (P_j, P'_j), the activations of edges in $C(f)$ are swapped between the paired inputs. We denote a circuit swap as $A \xleftarrow{C} B$, where

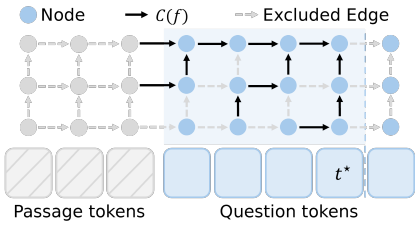
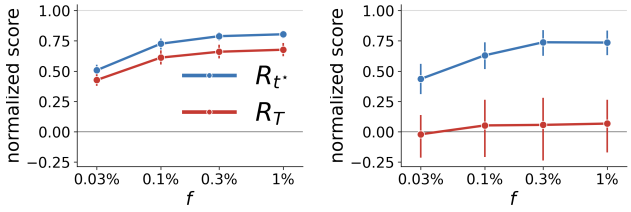


Figure 5: **Question-prefix circuits.** Edges ending at nodes within the shaded question prefix up to $t^\star$ are circuit candidates. Solid arrows denote selected edges in $C(f)$, while dashed arrows denote all remaining edges.



(a) Preserved-order pairs (b) Reversed-order pairs

Figure 6: **Effects of circuit swapping on answer preferences.** Normalized changes in the provisional preference score ($R_{t^\star}$, blue) and final-answer margin (R_T, red) across circuit sizes f for Qwen3-8B on CONDAQA paraphrase pairs. Error bars denote 95% family-cluster bootstrap CIs.

a forward pass on the input containing A replaces a_e^A with a_e^B for all $e \in C(f)$. Here, a_e^A and a_e^B denote the edge activations obtained from the respective inputs before swapping. Performing both $P_j \xleftarrow{C} P_j'$ and $P_j' \xleftarrow{C} P_j$ for each pair yields a set $\mathcal{S}_C$ of $2M$ circuit swaps. For each circuit swap $s_C = (A \xleftarrow{C} B) \in \mathcal{S}_C$, the resulting score is denoted by $\widetilde{\mathrm{PS}}_t^{A \xleftarrow{C} B}(f)$.

Adapting the normalization used in activation-patching evaluations (Zhang & Nanda, 2024), we normalize the swap-induced change in A's score by the pre-swap score difference between B and A at the same t. Averaging over all $2M$ swaps gives

$$R_t(f) = \frac{1}{2M} \sum_{s_C \in \mathcal{S}_C} \frac{\widetilde{\mathrm{PS}}_t^{A \xleftarrow{C} B}(f) - \mathrm{PS}_t^A}{\mathrm{PS}_t^B - \mathrm{PS}_t^A}, \qquad t \in \{t^\star, T\}. \tag{4}$$

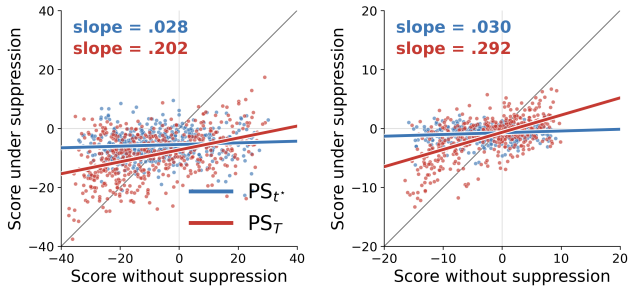
$R_{t^\star}(f)$ and $R_T(f)$ measure the mean normalized changes in the provisional preference score and the final-answer margin, respectively. Further details are provided in Appendix D.2.

Interpreting score changes. Although the paired inputs share the same question, differences in the passage phrasing may affect both provisional answer preference and the final-answer decision. Because passage phrasing can affect provisional preference and the final decision differently, the input with the higher provisional score does not always have the higher final-answer margin. This motivates us to compare how circuit swapping changes the provisional preference score and final-answer margin across pairs whose score ordering is preserved or reversed after question completion. Using the scores before swapping, we classify a pair as *preserved-order* if the input with the higher $\mathrm{PS}_{t^\star}$ also has the higher PS_T, and *reversed-order* if that input has the lower PS_T. Comparing the two groups tests whether circuit swapping recovers provisional preference even when recovering the final-answer margin requires a score change in the opposite direction. Of the 400 CONDAQA pairs evaluated with Qwen3-8B, 332 are preserved-order and 68 are reversed-order.

Results. Figure 6 shows normalized score changes across circuit sizes for Qwen3-8B. $R_{t^\star}$ increases with circuit size in both groups up to $f = 0.3$, with little change at $f = 1$. At $f = 0.3$, preserved-order pairs show $R_{t^\star} = 0.79$ and $R_T = 0.66$. In reversed-order pairs, however, $R_{t^\star}$ reaches 0.74 while R_T is only 0.06. Thus, $R_{t^\star}$ remains high even when the provisional and final scores rank the two inputs in opposite orders. The high $R_{t^\star}$ values in both groups indicate that question-prefix circuits selected solely by their attribution to the final-answer margin also carry provisional answer preference. Circuit swapping extends the hidden-state evidence in Section 5 to the circuits that contribute to forming question-prefix representations.

6.2 Suppression of the Question-Prefix Circuit

We examine how suppressing the question-prefix circuit through edge ablation (Li et al., 2024) affects provisional answer preference, the final-answer margin, and final-answer accuracy. We measure score changes on the CONDAQA evaluation set and accuracy on NP-BoolQ shared-prefix pairs.



(a) Qwen3-8B (b) Llama-3.1-8B

Figure 7: **Answer preferences under complete circuit suppression on CONDAQA.** Blue denotes $\mathrm{PS}_{t^\star}$ and red PS_T. Gray diagonals ($y = x$) represent unchanged scores. Colored lines show regression fits with annotated slopes.

Table 3: **Circuit suppression on NP-BoolQ shared-prefix pairs.** Accuracies are percentages, and gaps are aligned minus conflicting accuracy in percentage points.

γ	Aligned	Conflicting	Gap
Qwen3-8B			
1	96.1	82.4	13.7
0.5	90.5	83.3	7.2
0	58.5	85.9	-27.4
Llama-3.1-8B			
1	95.0	75.1	19.9
0.5	90.0	79.6	10.4
0	81.4	82.0	-0.6

We suppress $C(0.3)$ by scaling the activations passed through its edges, setting $a_e \leftarrow \gamma a_e$ for all $e \in C(0.3)$. Activations on all other edges remain unscaled. We evaluate $\gamma \in \{1, 0.5, 0\}$, corresponding to no suppression, partial suppression, and complete suppression, respectively.

Effects on answer preferences. We measure $\mathrm{PS}_{t^\star}$ and PS_T with and without suppression. For each score, we regress the values under complete suppression on the original values across examples. Figure 7 shows the effects of complete suppression on CONDAQA for Qwen3-8B and Llama-3.1-8B. In both models, the provisional-score regression is nearly flat, indicating that question-prefix circuits contribute to forming provisional answer preferences during question processing. The final-margin slopes are larger but remain below 1, showing that suppression also affects final-answer computation. Results across the full suppression range are provided in Appendix D.3.

Shared-prefix accuracy. We next examine how circuit suppression affects final-answer accuracy on NP-BoolQ shared-prefix pairs. Within each pair, we suppress the same selected edges in the shared question prefix and evaluate accuracy on both complete questions.

Table 3 shows that partial suppression narrows the aligned–conflicting accuracy gap in both models. Under complete suppression, the gap is near zero in Llama-3.1-8B and becomes negative in Qwen3-8B, mainly due to a sharp decline in aligned accuracy. Because the circuits were selected by their attribution to the final-answer margin, suppression may weaken useful answer-related computations alongside provisional preference, consistent with the observation that ablating selected components may also disrupt performance on the intended task (Marks et al., 2025). Yes/No-balanced analyses likewise show a smaller gap under complete suppression than without suppression (Appendix D.3).

The decrease in the aligned–conflicting accuracy gap is similar to that observed when steering against the original provisional preference in Section 5.2. These results suggest that question-prefix circuits carrying provisional answer preference contribute to the aligned–conflicting accuracy gap.

7 Conclusion

Motivated by evidence that people begin preparing an answer before a question is complete, we investigated whether autoregressive language models likewise form provisional answer preferences during question processing. We find that provisional answer preferences form before question completion and can influence final decisions. These preferences are carried by question-prefix hidden states and by circuits selected solely for their contribution to the final answer. Notably, provisional answer preferences form even when the question prefix leaves the correct answer undetermined. On NP-BoolQ shared-prefix pairs, models make more errors when the remaining question words require an answer that conflicts with the provisional preference. Steering against the early preference or partially suppressing these circuits makes final decisions less likely to follow the original preference. Our analyses suggest that, like humans, language models can remain biased toward an early preference when the remaining words call for the opposite answer. The final decision therefore reflects not only what the complete question says but also what the model favored before the question ended.

AI USE STATEMENT

In this work, we used ChatGPT and Codex with GPT-5.6-sol to assist with the implementation of experimental code, the drafting and editing of the manuscript. We also used these tools extensively to support the related-work review, including literature discovery, summarization, and organization, and to help articulate interpretations of experimentally obtained results. Generative AI was not used to formulate or refine the research hypotheses, develop the conceptual framework, or design the methodology or experiments. All AI-assisted code, text, interpretations, and literature suggestions were individually reviewed and verified by the authors. Code was inspected and tested before use, and all cited works were checked against their original sources. The authors made all final decisions regarding the methods, analyses, claims, and presentation and take full responsibility for the content of this work.

REPRODUCIBILITY STATEMENT

The main text defines the provisional answer preference and the preference score (PS) and describes the causal and circuit-level interventions used in this work. Appendix A documents the evaluated models, datasets, preprocessing procedures, prompt templates, question checkpoints, and answer-scoring method. Appendices B, C, and D provide full details of the hidden-state readout, activation swapping and steering, circuit discovery, exact circuit swapping, and suppression procedures. Experiments were conducted on four NVIDIA RTX A6000 GPUs. The implementation, experiment configurations, and scripts used to produce the reported results will be made publicly available upon publication.

REFERENCES

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, et al. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*, 2024.

Aishwarya Agrawal, Dhruv Batra, and Devi Parikh. Analyzing the behavior of visual question answering models. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pp. 1955–1960, 2016.

Dhananjay Ashok and Jonathan May. Language models can predict their own behavior. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=i8IqEzpHaJ.

Mathias Barthel. Speech planning interferes with language comprehension: Evidence from semantic illusions in question-response sequences. In *Proceedings of the 25th Workshop on the Semantics and Pragmatics of Dialogue*, pp. 16–29, 2021. URL https://www.semdial.org/anthology/Z21-Barthel_semdial_0005.pdf.

Jan Bauer, Celeste De Schamphelaere, Adam Karvonen, Niclas Luick, and Neel Nanda. Building better activation oracles, 2026. URL https://arxiv.org/abs/2606.02609.

Sara Bögels, Lilla Magyari, and Stephen C Levinson. Neural signatures of response planning occur midway through an incoming question in conversation. *Scientific reports*, 5(1):12881, 2015.

Oliver Bott, Petra Augurzky, Wolfgang Sternefeld, and Rolf Ulrich. Incremental generation of answers during the comprehension of questions with quantifiers. *Cognition*, 166:328–343, 2017.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. Boolq: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 2924–2936, 2019.

Arthur Conmy, Augustine Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards automated circuit discovery for mechanistic interpretability. *Advances in Neural Information Processing Systems*, 36:16318–16352, 2023.

Kyle Cox, Darius Kianersi, and Adrià Garriga-Alonso. Decoding answers before chain-of-thought: Evidence from pre-cot probes and activation steering. *arXiv preprint arXiv:2603.01437*, 2026.

Atticus Geiger, Hanson Lu, Thomas Icard, and Christopher Potts. Causal abstractions of neural networks. *Advances in neural information processing systems*, 34:9574–9586, 2021.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361, 2021.

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

Tal Haklay, Hadas Orgad, David Bau, Aaron Mueller, and Yonatan Belinkov. Position-aware automatic circuit discovery. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2792–2817, 2025.

Daniel Khashabi, Tushar Khot, and Ashish Sabharwal. More bang for your buck: Natural perturbation for robust question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 163–170, 2020.

Maximilian Li, Xander Davies, and Max Nadeau. Circuit breaking: Removing model behaviors with targeted ablation, 2024. URL https://arxiv.org/abs/2309.05973.

Samuel Marks, Can Rager, Eric Michaud, Yonatan Belinkov, David Bau, and Aaron Mueller. Sparse feature circuits: Discovering and editing interpretable causal graphs in language models. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 23888–23923, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/3ba4d47a83e498c2b1a0868cba20f6de-Paper-Conference.pdf.

Abhilasha Ravichander, Matt Gardner, and Ana Marasović. Condaqa: A contrastive reading comprehension dataset for reasoning about negation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 8729–8755, 2022.

Priyanka Sen and Amir Saffari. What do models learn from question answering datasets? In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pp. 2429–2438, 2020.

Saku Sugawara, Kentaro Inui, Satoshi Sekine, and Akiko Aizawa. What makes reading comprehension questions easier? In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pp. 4208–4219, 2018.

Naoya Sugiura, Kosuke Yamada, Ryohei Sasano, Koichi Takeda, and Katsuhiko Toyama. Building a buzzer-quiz answering system. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pp. 194–199, 2023.

Aaquib Syed, Can Rager, and Arthur Conmy. Attribution patching outperforms automated circuit discovery. In *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pp. 407–416, 2024.

Alon Talmor, Ori Yoran, Ronan Le Bras, Chandra Bhagavatula, Yoav Goldberg, Yejin Choi, and Jonathan Berant. Commonsenseqa 2.0: Exposing the limits of ai through gamification. *arXiv preprint arXiv:2201.05320*, 2022.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering language models with activation engineering, 2024. URL https://arxiv.org/abs/2308.10248.

Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: a circuit for indirect object identification in gpt-2 small. *arXiv preprint arXiv:2211.00593*, 2022.

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

Fred Zhang and Neel Nanda. Towards best practices of activation patching in language models: Metrics and methods. In *International Conference on Learning Representations*, volume 2024, pp. 1651–1678, 2024.

Jue Zhang, Qingwei Lin, Saravan Rajmohan, and Dongmei Zhang. From reasoning to answer: Empirical, attention-based and mechanistic insights into distilled deepseek r1 models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 3985–4002, 2025.

Long Zhang, Wei-neng Chen, Feng-feng Wei, and Zi-bo Qin. When does a language model commit? a finite-answer theory of pre-verbalization commitment. *arXiv preprint arXiv:2605.06723*, 2026.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate before use: Improving few-shot performance of language models. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 12697–12706. PMLR, 18–24 Jul 2021. URL `https://proceedings.mlr.press/v139/zhao21c.html`.