SANITY CHECKS FOR REASONING BASED HALLUCINATION DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Hallucination detection methods for large language models increasingly operate on chain-of-thought reasoning traces, yet it remains unclear whether they evaluate the reasoning itself or merely exploit surface correlates of the final answer. We introduce a controlled-invariance methodology that exposes this distinction through two oracle tests: FORCE, which replaces each response's final answer with the ground truth while preserving the reasoning trace, and REMOVE, which strips answer-announcement steps while leaving the trajectory intact. This reveals if their predictive power derives from answer-level artifacts rather than from the structure or validity of intermediate reasoning. We further show that once these artifacts are controlled for, effective discriminative power does still exist within the trace itself. To verify, we introduce TRACT, a lightweight scorer built on explainable lexical trajectory features (hedging trends, step-length dynamics, and cross-response vocabulary convergence), verifying that traces still contain important discriminative signatures, while remaining competitive with or outperforming existing baselines on unperturbed traces. These findings suggest that the current central challenge in reasoning-aware hallucination detection is not the absence of signal in the trace, but the failure to isolate it from endpoint cues.

1 INTRODUCTION

As large language models (LLMs) are increasingly used for reasoning and decision support, their reliability depends on detecting when generated outputs are unsupported, inconsistent, or false. Hallucination detection addresses this problem, but a central question remains unresolved: do current detectors assess the reasoning process itself, or do they mainly exploit surface correlates of the final answer?

This question is especially important for reasoning based responses. Recent hallucination and uncertainty detectors increasingly operate on chain-of-thought traces, comparing sampled responses, measuring semantic agreement, or scoring reasoning-path consistency (Wang et al., 2026). Yet strong performance on unmodified traces does not by itself show that a detector is reasoning-faithful. A method may appear to evaluate intermediate reasoning while actually relying on endpoint cues, answer formatting, response length, or coarse agreement among final answers. In that case, reported gains may overstate progress toward genuine reasoning assessment.

We introduce a controlled-invariance framework for exposing this failure mode. The key idea is simple: if a detector claims to evaluate the reasoning trajectory, then transformations that preserve the reasoning body should not destroy its ability to distinguish correct from incorrect reasoning. We instantiate this idea with two oracle sanity checks, illustrated in Figure 1. In FORCE, we replace the final answer with the ground truth and canonicalize its presentation, while leaving the intermediate reasoning unchanged. In REMOVE, we delete explicit answer-announcement steps, again preserving the reasoning body. Neither intervention repairs a flawed derivation or corrupts a valid one. Thus, a trace-faithful detector should remain informative under both conditions; large shifts indicate dependence on answer-level artifacts rather than reasoning evidence.

Applying these tests across four benchmarks and five models reveals that many existing detectors are less trace-faithful than standard evaluations suggest. As shown in Figure 2, several methods move far from the diagonal under FORCE or REMOVE, meaning their discriminative behavior changes substantially even though the intermediate reasoning trajectory is preserved. This is not merely a

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.

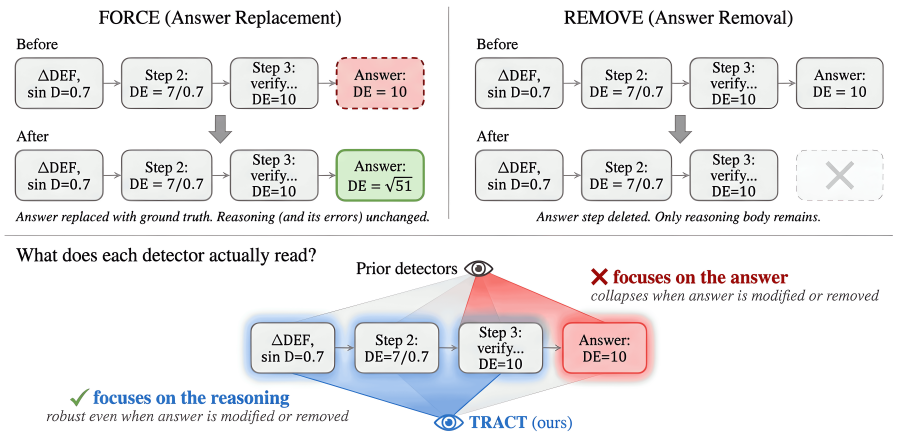


Figure 1: **Two sanity-check operations.** FORCE replaces only the final answer with the ground truth; REMOVE deletes explicit answer-announcement steps. Both preserve the reasoning body, so a trace-faithful detector should remain informative.

calibration issue: it indicates that some detectors obtain much of their signal from endpoint availability, answer standardization, or other artifacts that are orthogonal to reasoning quality.

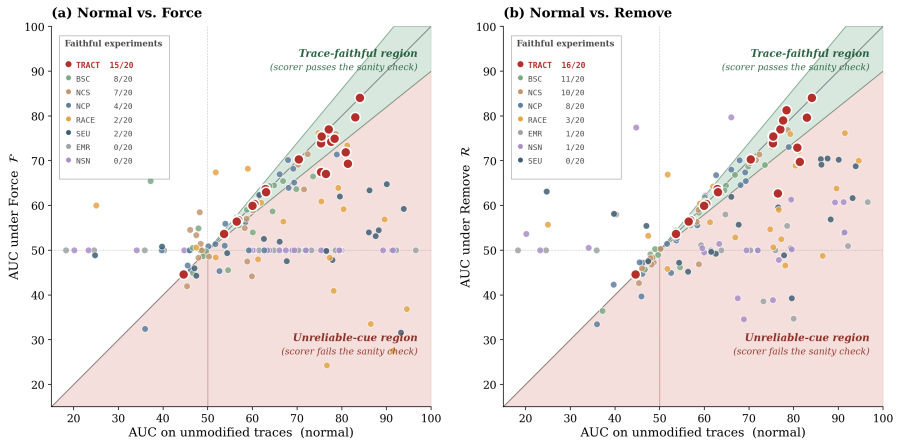


Figure 2: **Sanity-check results across four benchmarks and five models.** Each point is one scorer–model–benchmark experiment; x is AUC on original traces and y is AUC after FORCE or REMOVE. Trace-faithful scorers should remain near the diagonal because the reasoning body is preserved. TRACT has the highest number of faithful settings under both interventions.

We then ask whether robust trace-level detection requires complex learned representations and if these traces still contain discriminative features to determine an incorrect response. To do so, we propose TRACT, a lightweight scorer built from lexical trajectory features: local coherence cues, structural dynamics such as hedge and step-length trends, and cross-sample content convergence. TRACT is not a proof checker and does not verify each intermediate step. Instead, it targets observable symptoms of unsettled reasoning: traces that wander, hedge increasingly, become structurally irregular, or fail to converge across independent samples. Because these features are computed from the reasoning body rather than the endpoint string, TRACT is naturally suited to the FORCE/REMOVE setting, and verifies that effective detection can still be determined solely from the reasoning trace itself.

Our results support two conclusions. First, oracle robustness testing is a necessary sanity check for reasoning-aware hallucination detection: without it, a detector may appear successful while relying on answer-level artifacts. Second, useful trace-level signal exists in simple, interpretable trajectory statistics. TRACT is not meant to be a state of the art detector, but to validate that true reasoning

discrimination is possible; the challenge is therefore not only to design stronger detectors, but to evaluate whether their strength comes from the reasoning process they are meant to assess.

2 Background and Related Work

Sampling-based uncertainty. A common black-box approach to hallucination detection is to sample multiple responses and measure their disagreement: models that know the answer should produce mutually consistent outputs, while hallucinated or uncertain generations tend to diverge. Semantic entropy (Farquhar et al., 2024; Kuhn et al., 2023) formalizes this idea by clustering meaning-equivalent responses and computing entropy over the resulting semantic classes. Subsequent work refines either the representation or the disagreement measure: Kernel Language Entropy replaces hard clustering with a continuous similarity kernel (Nikitin et al., 2024), while embedding-based methods such as SINdex (Abdaljalil et al., 2025) and Semantic Embedding Uncertainty (SEU) (Grewal et al., 2024) use dense sentence representations to estimate inconsistency more efficiently. Perturbation-based variants further sample over input transformations rather than only model randomness (Gao et al., 2024). These methods provide strong black-box uncertainty signals, but their scores are usually computed over complete responses or final-answer semantics, making it difficult to tell whether they measure reasoning quality or endpoint agreement.

Reasoning-trace-aware detection. Chain-of-thought prompting (Wei et al., 2022) and long-form reasoning models make the intermediate trajectory observable, motivating detectors that evaluate not only what answer is produced but how the answer is reached. RACE (Wang et al., 2026) is representative of this direction, combining inter-sample reasoning-path consistency, answer uncertainty, reasoning–answer alignment, and intra-trace coherence. Such methods are closer to reasoning-aware hallucination detection than answer-only uncertainty estimators, but they also introduce a new evaluation problem: high performance may still come from endpoint cues, answer alignment, or coarse consistency rather than from trace-faithful assessment of the reasoning body. Our FORCE and REMOVE tests are designed to expose this distinction.

Unified UQ frameworks and baselines. Recent unified frameworks collect many of these signals into calibrated uncertainty pipelines. For example, UQLM (Bouchard et al., 2026) includes exact-match repetition and diversity scores (Cole et al., 2023), n-gram and BERTScore self-consistency (Manakul et al., 2023b; Zhang et al., 2020), NLI-based non-contradiction probability (Chen and Mueller, 2023), sentence-embedding similarity (Reimers and Gurevych, 2019), and semantic entropy variants (Farquhar et al., 2024; Kuhn et al., 2023). Surveys similarly emphasize uncertainty quantification as a central route to improving LLM reliability, while highlighting trade-offs among accuracy, cost, access requirements, and interpretability (Shorinwa et al., 2025; Kang et al., 2025). We use these families as black-box baselines because they cover the main operational signals in current hallucination detection: answer repetition, lexical overlap, embedding similarity, NLI agreement, semantic entropy, and reasoning-path consistency.

White-box versus black-box access. White-box detectors use token probabilities or hidden-state activations to estimate reliability in a single pass (Duan et al., 2024; Zollicoffer et al., 2025; Phukan et al., 2025; Binkowski et al., 2025; Fadeeva et al., 2024). These methods can be effective, but require model internals that are unavailable for many closed-source systems. We therefore focus on the black-box setting, where the detector observes only sampled text traces. Within this setting, our goal is not merely to improve AUC, but to test whether a detector's signal remains valid when answer-level artifacts are controlled.

3 TRACT: Trace Rhetorical and Coherence Trajectory

A correct reasoning trace tends to *settle*. As the model approaches a solution, its steps usually become more directed: the vocabulary stabilises, intermediate claims become more consistent, and independently sampled traces begin to resemble one another in how they progress and where they end. By contrast, an incorrect trace often *wanders*: it asks itself unnecessary questions, restarts or hedges, expands when it should compress, and disagrees with parallel samples about how many steps the problem requires.

Table 1: **TRACT feature inventory.** TRACT maps sampled reasoning traces into three interpretable feature blocks. The intuition column gives the diagnostic signature; the definition column gives the implementation-level statistic. Here T_k is the number of steps in trace k, $s_i^{(k)}$ is step i, $w_i^{(k)} = |s_i^{(k)}|$ is its word count, $q_i^{(k)}$ is its question-mark count, $h_i^{(k)}$ is the number of hedge words from lexicon $\mathcal{H}$ in the step, $\mathcal{U}_i^{(k)}$ is the lowercased unigram set, $\mathcal{E}_i^{(k)}$ is the set of capitalised tokens, $m_k = \lfloor T_k/2 \rfloor$, and $J(A,B) = |A \cap B|/|A \cup B|$ is Jaccard similarity. Sign indicates whether larger feature values increase $(+)$ or decrease $(-)$ the TRACT incorrectness score.

Block	Feature	Intuition	Sign	Definition
Coherence ϕ_{coh}	QUESTIONRATE	Frequent self-questioning indicates unresolved uncertainty.	$+$	$\frac{1}{K}\sum_k \frac{1}{T_k}\sum_i q_i^{(k)}$
	WORDSPERSTEP	Verbose steps often reflect hedging, qualification, or failure to compress.	$+$	$\frac{1}{K}\sum_k \frac{1}{T_k}\sum_i w_i^{(k)}$
	PLATEAUFRAC	Non-growing steps suggest the trace is stalling rather than developing.	$+$	$\frac{1}{K}\sum_k \frac{1}{T_k-1}\sum_{i=2}^{T_k} \mathbb{I}\left[w_i^{(k)} \le w_{i-1}^{(k)}\right]$
Structure ϕ_{struct}	HEDGESLOPE	Increasing hedge density suggests confidence is draining over time.	$+$	$\frac{1}{K}\sum_k \text{slope}_{i/T_k}\left(h_i^{(k)}\right)_{i=1}^{T_k}$
	COLONFRAC	Explicit delimiters mark organised decomposition, cases, or subclaims.	$-$	$\frac{1}{K}\sum_k \frac{1}{T_k}\sum_i \mathbb{I}\left[: \in s_i^{(k)}\right]$
	MAXSTEPWC	A focused long step can indicate concentrated reasoning depth.	$-$	$\frac{1}{K}\sum_k \max_i w_i^{(k)}$
	SCMAX	An outlier-length trace signals disagreement about the required reasoning path.	$+$	$\max_k T_k$
	WCVARSLOPE	Growing local length variance suggests the trace is losing its structural thread.	$+$	$\frac{1}{K}\sum_k \text{slope}_{i/T_k}\left(\text{Var}(w_{i-2:i}^{(k)})\right)_{i=3}^{T_k}$
Content ϕ_{cont}	MIDUNIGRAMDIV	Mid-trace lexical disagreement indicates early divergence across samples.	$+$	$\binom{K}{2}^{-1}\sum_{j<k}\left(1 - J(\mathcal{U}_{m_j}^{(j)}, \mathcal{U}_{m_k}^{(k)})\right)$
	FINALUNIGRAMDIV	Terminal lexical disagreement indicates that independent traces end in different places.	$+$	$\binom{K}{2}^{-1}\sum_{j<k}\left(1 - J(\mathcal{U}_{T_j}^{(j)}, \mathcal{U}_{T_k}^{(k)})\right)$
	ENTITYREPEAT	Repeatedly revisiting the same entity can indicate circular reasoning.	$+$	$\frac{1}{K}\sum_k \frac{1}{T_k}\sum_{i=2}^{T_k} \mathbb{I}\left[\mathcal{E}_i^{(k)} \cap \mathcal{E}_{i-1}^{(k)} \neq \varnothing\right]$

TRACT operationalises this observation in a fully black-box sampling setting. Given a prompt x, we sample K independent reasoning traces from the model, $\{\mathbf{r}^{(k)}\}_{k=1}^{K}$. Each trace is written as a sequence of textual steps, $\mathbf{r}^{(k)} = (s_1^{(k)}, \ldots, s_{T_k}^{(k)})$. A textual step is the smallest reasoning unit exposed in the response, such as a numbered line, bullet point, sentence-level inference, or explicitly separated intermediate statement. For example, in a chain-of-thought response, "First, compute the total cost" and "Therefore, the remaining amount is 12" would be treated as two separate steps. Answer-announcement steps such as "Final Answer:" are excluded before TRACT features are computed, so the scorer operates on the reasoning body rather than the endpoint string.

TRACT does not require access to logits, hidden states, answer labels, embedding models, or auxiliary entailment models. Instead, it reads the sampled traces themselves and extracts lightweight trajectory-level features that describe how the reasoning behaves. The features are grouped into three blocks, each corresponding to a different diagnostic question: i) **Coherence:** what does each trace look like on average? ii) **Structure:** how does the trace evolve as reasoning progresses? iii) **Content:** do independent traces converge on the same intermediate and final vocabulary?

Table 1 summarises the full TRACT feature set. The table is organised around the observable signature of each feature: what pattern it detects, whether that pattern increases the hallucination score, and how the statistic is computed. The prose below explains why these signatures are useful.

Coherence features: does each trace sound locally resolved? The coherence block captures whether a trace has the local rhetorical profile of a settled solution. When a model knows how to proceed, its steps tend to state intermediate claims directly and move on. When it is uncertain, the

trace often becomes interrogative, over-qualified, or repetitive. QUESTIONRATE captures explicit self-questioning, WORDSPERSTEP captures verbosity and over-explanation, and PLATEAUFRAC captures steps that fail to develop.

This block helps because many reasoning failures are visible before the final answer: the model circles around the problem, asks itself what to do, or spends words compensating for a missing solution path. Coherence features are therefore weak but useful symptoms of local unresolvedness.

Structure features: does the trace keep its trajectory? The structure block captures whether reasoning becomes more organised or more unstable over time. A correct trace can be long or short, but it usually maintains a coherent trajectory: decomposition, intermediate work, and convergence. An incorrect trace more often loses this trajectory. Hedging may increase, step lengths may become irregular, or one sampled trace may become much longer than the others because the model cannot settle on a path.

TRACT measures this with five structural signatures. HEDGESLOPE tracks whether uncertainty language grows over time (Lakoff, 1973; Katerenchuk and Levitan, 2024). COLONFRAC captures explicit organisation through cases, lists, or subclaims. MAXSTEPWC captures whether the trace contains a concentrated reasoning step. SCMAX flags an outlier-length trace among the K samples. WCVARSLOPE tracks whether the rhythm of step lengths becomes increasingly unstable (Jin et al., 2024).

This block helps because reasoning quality is often dynamic: the important signal is not only what the trace contains, but how it changes (Vanhoyweghen et al., 2025).

Content features: do independent traces converge? The content block captures whether independent samples appear to reason toward the same state. Correct traces need not be identical, but they often share key concepts at the midpoint and converge to similar terminal vocabulary. Incorrect traces are more likely to scatter: one sample pursues one interpretation, another pursues a different one, and their intermediate or final steps share little lexical overlap.

TRACT measures this with lightweight lexical agreement. MIDUNIGRAMDIV asks whether traces agree around the middle of the reasoning process. FINALUNIGRAMDIV asks whether they agree near the end. ENTITYREPEAT captures a different content failure: a single trace repeatedly revisiting the same named entity instead of advancing the reasoning state Yao et al. (2025); Duan et al. (2026). This block helps because uncertainty is often expressed across samples. Even when each individual trace looks fluent, disagreement among independently sampled trajectories can reveal that the model has not identified a stable solution. We note that previous works do not directly operationalize these properties as explicit trajectory-level measurements. TRACT fills this gap by measuring how these properties evolve over a reasoning trace.

Scoring TRACT maps the three feature blocks into a single incorrectness score. Each feature is robust-scaled using median centring and IQR normalisation, and then clipped to $[-3, 3]$ to limit the effect of extreme traces. Within each block, features use fixed equal-magnitude weights, with signs given in Table 1. The structure block always contributes to the score. This block captures trajectory dynamics—hedge trends, step-count outliers, and step-length irregularity—that remain meaningful across both terse and verbose reasoning styles. The coherence and content blocks are more style-dependent. They are most reliable when responses are explicitly segmented into step-wise reasoning; in prose-heavy traces, the same surface cue can change meaning. For example, self-questioning may signal confusion in terse chain-of-thought, but careful exposition in a long explanatory paragraph. TRACT therefore gates the coherence and content blocks using the raw WORDSPERSTEP value.

Formally, TRACT computes

$$\mathrm{TRACT}(x) = \mathbf{w}_{\mathrm{struct}}^{\top}\hat{\phi}_{\mathrm{struct}} + \big(1-\alpha(x)\big)\left(\mathbf{w}_{\mathrm{coh}}^{\top}\hat{\phi}_{\mathrm{coh}} + \mathbf{w}_{\mathrm{cont}}^{\top}\hat{\phi}_{\mathrm{cont}}\right),$$

where

$$\alpha(x) = \exp\!\left(-\frac{1}{2\sigma^2}\big(\bar{w}-\mu\big)^2\right).$$

Here $\hat{\phi}_b$ is the robust-scaled feature vector for block b, and $\bar{w}$ is the raw WORDSPERSTEP value before scaling. When $\bar{w}$ lies near the prose-heavy regime centred at μ, $\alpha(x)$ approaches one, suppressing

the coherence and content terms. When $\bar{w}$ is far from this regime, $\alpha(x)$ approaches zero and all three blocks contribute. Implementation details for step extraction, hedge counting, entity extraction, and slope computation are provided in Appendix C.

Table 2 summarises the external components required by each detector. This comparison is not the source of TRACT's accuracy; it clarifies its practical setting. Once sampled traces are available, TRACT is a text-only trajectory scorer and does not require an auxiliary NLI model, embedding model, answer parser, or access to model internals.

Table 2: **Method requirements.** A checkmark indicates that a detector requires the corresponding external component. TRACT operates directly on sampled reasoning trajectories and does not require external NLI or embedding models.

	EMR	NCP	BSC	NCS	NSN	RACE	SEU	TRACT
NLI Model		✓			✓	✓		
Embedding Model			✓	✓	✓	✓	✓	

Unlike NLI- and embedding-based baselines, TRACT computes all features directly from surface trajectory statistics. This makes the scorer inexpensive to run, transparent to inspect, and compatible with closed-source models where only generated text is available. Figure 3 evaluates whether TRACT's three feature blocks contribute complementary signal. The full scorer is strongest or near-strongest across benchmarks, indicating that coherence, structure, and content capture different failure signatures rather than redundant variants of the same cue. For this work, we restrict TRACT to lexical choices, although we note that performance and failure modes could be improved if layout signatures were to be used (see Appendix C.1 for details).

4 Experiments

Benchmarks We evaluate on four diverse reasoning benchmarks spanning distinct reasoning modalities and difficulty regimes: BBH-Tracking Suzgun et al. (2022), which requires multi-step state tracking under sequential object permutations; GPQA Diamond Rein et al. (2024), which tests graduate-level scientific reasoning resistant to surface-level retrieval; MATH-500 Lightman et al. (2023), which covers competition-level mathematics across seven subject areas where intermediate step quality is predictive of correctness; and CausalT5K Geng et al. (2026), which probes structural causal reasoning under adversarial narrative pressure across Pearl's three rungs; for CausalT5K we use the D4 (environment and climate) setting under all three rungs. Together, these benchmarks span symbolic, scientific, mathematical, and causal reasoning (details in Appendix A).

Models For each benchmark, we sample responses from a mixture of open-weight and proprietary large language models spanning several model families: Nemotron-30B (NVIDIA Corporation, 2025), GPT-OSS-120B (OpenAI, 2025), LLaMA-3-70B (Dubey et al., 2024), Amazon Nova Pro (Amazon Web Services, 2025), and Gemma-3-27B (Google DeepMind, 2025). Additional model details, prompting, sampling hyperparameters, and step-extraction procedures are provided in Appendix B.

Scorers We compare against seven black-box UQ scorers with recommended settings: Exact-Match Repetition (EMR) (Bouchard et al., 2026) , NLI-based Non-Contradiction Probability (NCP) (Manakul et al., 2023a), BERTScore Consistency (BSC) (Zhang et al., 2020), Normalized Cosine Similarity (NCS) (Reimers and Gurevych, 2019), Normalized Semantic Negentropy (NSN) (Farquhar et al., 2024; Bouchard et al., 2026), RACE (Wang et al., 2026), and Semantic Embedding Uncertainty (SEU) (Grewal et al., 2024). All methods receive the same 10 sampled traces. We evaluate TRACT with $\mu = 28$ words per step and $\sigma = 50$ (see Appendix C for high level analysis of TRACT with different settings). Following prior work (Wang et al., 2026; Kuhn et al., 2023), we report AUC as the primary metric.

In contrast to prior methods that rely heavily on final answer signatures to assess correctness, we design controlled interventions to isolate trajectory-level signals from answer-level artifacts (see Appendix B.5 for exact details). Specifically, we restrict evaluation to sets of sampled traces where

Table 3: AUC (%) across datasets, reported as mean$_{\pm SE}$ over 10 cross-validation folds, rounded to nearest .5. For each dataset and model, the best FORCE and REMOVE values across methods are bolded.

	Gemma3		GPT-OSS-120B		Llama3-70B		Nemo		Nova-Pro	
Method	FORCE	REMOVE	FORCE	REMOVE	FORCE	REMOVE	FORCE	REMOVE	FORCE	REMOVE
BBH-Tracking										
NCP	$67.0_{\pm 4.9}$	$68.0_{\pm 8.8}$	$51.0_{\pm 8.0}$	$52.0_{\pm 10.6}$	$46.0_{\pm 15.0}$	$47.0_{\pm 10.6}$	$56.0_{\pm 1.8}$	$53.0_{\pm 3.5}$	$45.5_{\pm 18.8}$	$45.0_{\pm 18.6}$
BSC	$53.5_{\pm 9.9}$	$54.5_{\pm 10.8}$	$65.5_{\pm 13.5}$	$\mathbf{67.0_{\pm 11.5}}$	$50.0_{\pm 8.4}$	$47.5_{\pm 7.7}$	$75.5_{\pm 4.6}$	$76.5_{\pm 4.2}$	$\mathbf{50.0_{\pm 22.1}}$	$\mathbf{50.0_{\pm 22.1}}$
NCS	$57.0_{\pm 12.8}$	$59.0_{\pm 10.2}$	$59.0_{\pm 9.7}$	$61.0_{\pm 9.3}$	$50.0_{\pm 7.5}$	$47.5_{\pm 9.1}$	$51.5_{\pm 2.9}$	$48.5_{\pm 4.6}$	$48.0_{\pm 21.2}$	$48.5_{\pm 21.4}$
NSN	$50.0_{\pm 0.0}$	$62.5_{\pm 5.5}$	$50.0_{\pm 0.0}$	$50.5_{\pm 0.2}$	$50.0_{\pm 0.0}$	$47.5_{\pm 8.2}$	$50.0_{\pm 0.0}$	$60.5_{\pm 13.0}$	$\mathbf{50.0_{\pm 0.0}}$	$34.5_{\pm 15.3}$
RACE	$76.0_{\pm 2.7}$	$55.0_{\pm 11.5}$	$\mathbf{68.5_{\pm 10.4}}$	$60.5_{\pm 6.9}$	$48.5_{\pm 1.5}$	$52.5_{\pm 8.2}$	$28.0_{\pm 0.4}$	$76.0_{\pm 8.8}$	$41.0_{\pm 8.4}$	$47.0_{\pm 20.8}$
SEU	$57.5_{\pm 9.9}$	$58.0_{\pm 10.2}$	$62.0_{\pm 9.3}$	$62.0_{\pm 9.3}$	$50.0_{\pm 8.8}$	$48.5_{\pm 8.6}$	$52.5_{\pm 4.2}$	$52.5_{\pm 4.2}$	$41.5_{\pm 18.3}$	$41.5_{\pm 18.3}$
TRACT	$\mathbf{77.0_{\pm 6.6}}$	$\mathbf{77.0_{\pm 6.6}}$	$53.5_{\pm 10.4}$	$53.5_{\pm 10.4}$	$\mathbf{60.5_{\pm 9.9}}$	$\mathbf{60.5_{\pm 9.9}}$	$\mathbf{84.0_{\pm 2.2}}$	$\mathbf{84.0_{\pm 2.2}}$	$45.0_{\pm 15.5}$	$45.0_{\pm 15.5}$
GPQA Diamond										
NCP	$56.5_{\pm 3.3}$	$56.0_{\pm 3.5}$	$47.0_{\pm 2.2}$	$47.5_{\pm 2.4}$	$51.0_{\pm 6.2}$	$50.0_{\pm 6.2}$	$32.5_{\pm 4.6}$	$33.5_{\pm 4.2}$	$51.5_{\pm 5.5}$	$51.5_{\pm 5.5}$
BSC	$64.5_{\pm 7.7}$	$64.5_{\pm 7.7}$	$55.5_{\pm 3.3}$	$55.5_{\pm 2.0}$	$48.5_{\pm 3.3}$	$50.5_{\pm 3.8}$	$45.5_{\pm 3.3}$	$46.5_{\pm 3.3}$	$54.0_{\pm 3.1}$	$53.5_{\pm 2.9}$
NCS	$61.0_{\pm 3.5}$	$60.5_{\pm 3.3}$	$55.0_{\pm 6.6}$	$56.5_{\pm 3.8}$	$48.5_{\pm 5.1}$	$48.5_{\pm 5.5}$	$47.5_{\pm 6.4}$	$46.5_{\pm 6.4}$	$56.5_{\pm 7.3}$	$56.5_{\pm 7.3}$
NSN	$50.0_{\pm 0.0}$	$50.0_{\pm 0.0}$	$50.0_{\pm 0.0}$	$49.5_{\pm 0.7}$	$50.0_{\pm 0.0}$	$50.5_{\pm 1.1}$	$50.0_{\pm 0.0}$	$50.0_{\pm 0.0}$	$50.0_{\pm 0.0}$	$50.0_{\pm 0.0}$
RACE	$24.5_{\pm 5.1}$	$59.0_{\pm 2.2}$	$33.5_{\pm 0.7}$	$48.5_{\pm 2.9}$	$48.5_{\pm 5.5}$	$46.0_{\pm 5.3}$	$61.0_{\pm 8.4}$	$51.0_{\pm 5.3}$	$48.0_{\pm 4.0}$	$56.0_{\pm 4.4}$
SEU	$50.0_{\pm 4.2}$	$59.5_{\pm 4.2}$	$54.5_{\pm 2.7}$	$57.0_{\pm 2.9}$	$49.0_{\pm 3.5}$	$47.5_{\pm 4.0}$	$52.5_{\pm 6.0}$	$49.0_{\pm 6.0}$	$47.5_{\pm 7.5}$	$55.5_{\pm 7.7}$
TRACT	$\mathbf{73.5_{\pm 3.3}}$	$\mathbf{73.5_{\pm 3.3}}$	$\mathbf{72.0_{\pm 4.0}}$	$\mathbf{73.0_{\pm 3.5}}$	$\mathbf{63.5_{\pm 1.1}}$	$\mathbf{64.0_{\pm 1.3}}$	$\mathbf{74.0_{\pm 4.4}}$	$\mathbf{79.0_{\pm 3.1}}$	$\mathbf{57.0_{\pm 5.7}}$	$\mathbf{57.0_{\pm 5.7}}$
Math 500										
NCP	$70.5_{\pm 10.8}$	$70.0_{\pm 6.6}$	$65.0_{\pm 0.9}$	$65.5_{\pm 2.9}$	$64.0_{\pm 2.7}$	$65.0_{\pm 2.2}$	$71.5_{\pm 1.1}$	$73.0_{\pm 2.2}$	$68.0_{\pm 2.2}$	$67.5_{\pm 2.4}$
BSC	$59.0_{\pm 5.7}$	$58.5_{\pm 3.8}$	$64.0_{\pm 1.8}$	$69.0_{\pm 3.1}$	$58.5_{\pm 3.3}$	$63.5_{\pm 2.9}$	$66.5_{\pm 2.0}$	$69.0_{\pm 1.3}$	$64.5_{\pm 1.5}$	$63.0_{\pm 1.3}$
NCS	$69.5_{\pm 0.7}$	$69.5_{\pm 4.6}$	$67.5_{\pm 2.0}$	$73.0_{\pm 1.3}$	$64.0_{\pm 4.4}$	$70.0_{\pm 4.0}$	$\mathbf{76.0_{\pm 3.5}}$	$77.0_{\pm 2.2}$	$71.5_{\pm 3.3}$	$71.5_{\pm 3.3}$
NSN	$50.0_{\pm 0.0}$	$61.0_{\pm 3.5}$	$50.0_{\pm 0.0}$	$\mathbf{77.5_{\pm 2.4}}$	$50.0_{\pm 0.0}$	$61.0_{\pm 3.1}$	$50.0_{\pm 0.0}$	$\mathbf{79.5_{\pm 6.0}}$	$50.0_{\pm 0.0}$	$54.0_{\pm 0.4}$
RACE	$36.5_{\pm 13.0}$	$70.0_{\pm 1.8}$	$57.0_{\pm 3.1}$	$64.0_{\pm 2.2}$	$64.0_{\pm 2.2}$	$\mathbf{76.0_{\pm 4.0}}$	$59.0_{\pm 1.3}$	$69.0_{\pm 4.4}$	$73.5_{\pm 2.4}$	$72.0_{\pm 2.2}$
SEU	$59.5_{\pm 6.0}$	$68.5_{\pm 5.1}$	$65.0_{\pm 2.7}$	$70.0_{\pm 2.7}$	$54.0_{\pm 1.3}$	$70.0_{\pm 3.5}$	$53.0_{\pm 5.3}$	$70.5_{\pm 4.2}$	$63.0_{\pm 3.5}$	$69.0_{\pm 2.2}$
TRACT	$\mathbf{80.0_{\pm 2.2}}$	$\mathbf{79.5_{\pm 1.5}}$	$\mathbf{69.5_{\pm 1.1}}$	$69.5_{\pm 2.4}$	$\mathbf{67.5_{\pm 0.7}}$	$\mathbf{76.0_{\pm 3.5}}$	$67.0_{\pm 4.9}$	$62.5_{\pm 7.3}$	$\mathbf{74.5_{\pm 2.4}}$	$\mathbf{81.0_{\pm 2.2}}$
CausalT5K-D4										
NCP	$50.0_{\pm 0.0}$	$45.0_{\pm 5.7}$	$50.0_{\pm 0.0}$	$59.5_{\pm 5.1}$	$50.0_{\pm 0.0}$	$42.0_{\pm 4.4}$	$50.0_{\pm 0.0}$	$67.0_{\pm 7.1}$	$50.0_{\pm 0.0}$	$39.5_{\pm 3.8}$
BSC	$45.0_{\pm 4.9}$	$46.0_{\pm 5.3}$	$54.0_{\pm 3.1}$	$62.0_{\pm 5.3}$	$55.0_{\pm 6.2}$	$49.0_{\pm 6.6}$	$65.5_{\pm 3.3}$	$36.5_{\pm 1.5}$	$50.5_{\pm 4.6}$	$45.5_{\pm 3.3}$
NCS	$49.0_{\pm 2.2}$	$50.5_{\pm 2.9}$	$44.0_{\pm 8.0}$	$60.0_{\pm 6.6}$	$54.5_{\pm 5.5}$	$46.0_{\pm 3.5}$	$58.5_{\pm 3.8}$	$46.5_{\pm 5.1}$	$53.5_{\pm 1.5}$	$47.5_{\pm 4.6}$
NSN	$50.0_{\pm 0.0}$	$51.5_{\pm 2.4}$	$50.0_{\pm 0.0}$	$39.5_{\pm 5.5}$	$50.0_{\pm 0.0}$	$50.5_{\pm 1.5}$	$50.0_{\pm 0.0}$	$39.0_{\pm 4.4}$	$50.0_{\pm 0.0}$	$53.5_{\pm 4.2}$
RACE	$61.0_{\pm 5.3}$	$\mathbf{64.0_{\pm 3.1}}$	$60.0_{\pm 4.4}$	$55.5_{\pm 6.4}$	$56.5_{\pm 6.0}$	$53.0_{\pm 4.4}$	$67.5_{\pm 4.2}$	$67.0_{\pm 5.3}$	$50.5_{\pm 3.3}$	$53.5_{\pm 2.4}$
SEU	$55.0_{\pm 6.2}$	$55.5_{\pm 6.4}$	$49.0_{\pm 1.3}$	$63.5_{\pm 6.4}$	$45.5_{\pm 3.8}$	$45.0_{\pm 4.0}$	$28.5_{\pm 6.0}$	$49.5_{\pm 2.9}$	$46.0_{\pm 3.1}$	$47.5_{\pm 3.8}$
TRACT	$\mathbf{63.0_{\pm 3.1}}$	$63.0_{\pm 3.1}$	$\mathbf{70.0_{\pm 5.3}}$	$\mathbf{70.0_{\pm 5.3}}$	$\mathbf{60.0_{\pm 7.5}}$	$\mathbf{60.0_{\pm 7.5}}$	$\mathbf{75.5_{\pm 4.6}}$	$\mathbf{75.5_{\pm 4.6}}$	$\mathbf{56.5_{\pm 3.8}}$	$\mathbf{56.5_{\pm 3.8}}$

the final answer is no longer a valid separator between incorrect and correct responses, ensuring that variation in scores arises from differences in reasoning structure, rhetorical coherence, and cross-trace consistency rather than answer disagreement. The FORCE and REMOVE interventions should be interpreted in light of each detector's intended objective. They reveal that the detector's signal depends on endpoint information in addition to, or instead of, the reasoning trajectory. We introduce two complementary conditions. In FORCE, we replace each response's final answer with the ground truth and canonicalize the announcement of the final answer, producing trajectories that have final answers that are correct and uniform; this isolates whether TRACT detects inconsistencies independent of answer correctness. Additionally, in REMOVE, we strip explicit answer-announcement steps, preserving natural reasoning trajectories while eliminating explicit answer signals. Responses maintain their original label and performance is based on the initial response. Under REMOVE, every detector, TRACT included, scores the identical answer-stripped traces, so these columns are the direct head-to-head comparison. Under FORCE, every detector receives the canonicalized ground-truth answer; because TRACT's preprocessing strips answer-announcement steps (Appendix B.5), its FORCE and REMOVE inputs coincide, and its identical scores across the two conditions follow from this design rather than being an empirical finding. Answer-level baselines such as EMR are not expected to pass FORCE.

4.1 RESULTS

Current detectors are susceptible to sanity checks A reliable detector should evaluate how the model reasons, not merely what answer it gives: it should remain informative under both interventions, and large shifts indicate sensitivity to answer presentation rather than reasoning.

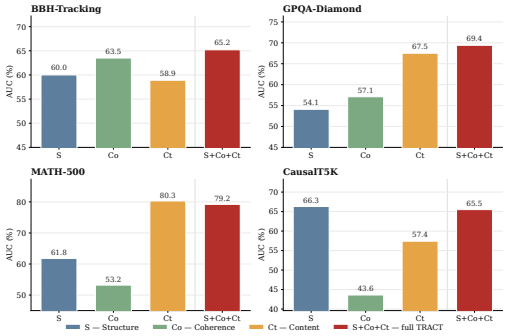


Figure 3: Ablating TRACT feature blocks across four benchmarks. **S** = Structure, **Co** = Coherence, and **Ct** = Content. The full S+Co+Ct scorer is strongest or near-strongest across benchmarks, indicating that the blocks capture complementary trace-level signals.

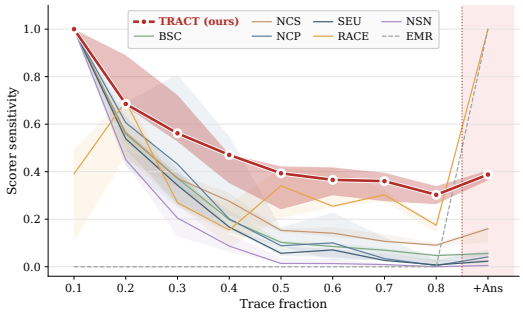


Figure 4: **Step-wise sensitivity on GPQA Diamond.** Normalised score changes as progressively more of the trace is revealed. Endpoint-reliant methods spike when the final answer appears; TRACT remains sensitive across the reasoning body.

Table 3 shows that several existing detectors are unstable under these sanity checks. For example, RACE achieves 76.0 AUC under FORCE on BBH-Tracking with Gemma-3-27B, but drops to 55.0 under REMOVE. On GPQA Diamond with the same model, it nearly inverts, scoring 24.5 under FORCE and 59.0 under REMOVE. NSN exhibits a different failure mode on MATH-500: it remains at chance under FORCE for every model, yet rises sharply under REMOVE, reaching 77.5 for GPT-OSS-120B and 79.5 for Nemotron-30B. These shifts suggest that the detectors are responding strongly to endpoint formatting or answer availability rather than to stable reasoning evidence.

TRACT is more stable across the same interventions. In Table 3, TRACT often obtains identical or near-identical FORCE/REMOVE scores because its features are computed from the reasoning body after answer-announcement removal. It also remains competitive in absolute AUC, especially on settings where surface answer cues are weak. On CausalT5K-D4, for instance, TRACT obtains 75.5/75.5 with Nemotron-30B and 70.0/70.0 with GPT-OSS-120B, while many baselines remain close to chance or vary substantially across intervention modes. All methods appear to struggle with Nova-Pro, however we note that performance and failure modes could be improved if layout signatures were to be used (see Appendix C.1 for details). Together with Figure 2, these results indicate that TRACT is less dependent on answer-level artifacts and more sensitive to trajectory-level signals.

Step-wise sensitivity analysis Figure 4 examines where each detector obtains its signal along the reasoning trajectory. For each response, we reveal progressively larger prefixes of the trace and measure the marginal change in detector score after each additional step. Scores are normalised to $[0, 1]$ within each method so that the figure compares where sensitivity occurs rather than raw score magnitude.

The methods separate into three qualitative patterns. First, answer-anchored methods such as EMR and RACE remain relatively insensitive throughout the reasoning body and respond most strongly when the final answer is revealed. This behaviour is undesirable for a reasoning-aware detector: it indicates that the method is primarily reading the endpoint. Second, methods such as NCS, NCP, BSC, SEU, and NSN are prone to saturating, and react strongly to early steps and then plateau. These methods extract useful information from the trace, but much of their signal is concentrated near the beginning rather than distributed across the full trajectory. TRACT behaves differently. Its sensitivity remains more evenly distributed across the reasoning process and does not spike primarily at the answer-reveal transition. This reflects the design of Table 1: TRACT combines static coherence, dynamic structural trends, and cross-sample content agreement, so no single step or endpoint dominates the score.

Dataset and feature relationships Figure 3 decomposes TRACT by feature block and shows that different benchmarks expose different trace-level failure modes. On MATH-500 and GPQA

Diamond, content agreement is especially informative: the Content block alone reaches 80.3 AUC on MATH-500 and 67.5 on GPQA Diamond. This suggests that, for mathematical and scientific reasoning tasks, incorrect sampled traces often diverge in the concepts or final vocabulary they converge to. CausalT5K-D4 shows the opposite pattern. There, the Structure block is strongest among individual blocks, reaching 66.3 AUC. This is consistent with the nature of causal reasoning under adversarial narrative pressure: correctness depends less on repeating the same lexical endpoint and more on maintaining a coherent reasoning trajectory.

The full block-combination heatmap in Appendix C (Figure 5) further shows that the blocks are complementary rather than redundant. Structure plus Content is the strongest two-block combination on GPQA Diamond and CausalT5K-D4, while on MATH-500 it nearly matches Content alone. The full TRACT scorer, combining Structure, Coherence, and Content, achieves the highest or near-highest AUC on every benchmark. TRACT thus relies on no single universal cue, but combines weak, interpretable signatures whose usefulness varies by task.

Complementarity with existing detectors Finally, we test whether TRACT captures information complementary to existing black-box scorers. As a diagnostic of complementarity, not a recommended deployment setting, Appendix D fuses TRACT with each baseline using 4-fold cross-validated logistic regression on unmodified traces (Table 4).

Table 4: **TRACT is complementary to existing detectors.** Average AUC across four benchmarks for standalone TRACT and the best TRACT+baseline supervised fusion.

Model	TRACT	Best TRACT+X	Gain
Nova Pro	59.05	79.05	+20.00
Gemma-3-27B	74.59	80.01	+5.42
GPT-OSS-120B	71.53	84.10	+12.57
LLaMA-3-70B	64.65	74.72	+10.07
Nemotron-30B	78.38	88.31	+9.93

Fusion improves over standalone TRACT for every model, with average gains from +5.42 to +20.00 AUC points. EMR and NSN are the most consistent partners, showing that TRACT complements both answer-repetition and semantic-entropy signals. RACE also helps for GPT-OSS-120B and Nemotron-30B, suggesting that TRACT's hedge, step-length, and lexical-convergence features are not fully captured by reasoning-consistency or answer-alignment scores.

5 Conclusion

We presented a controlled-invariance framework for testing whether hallucination detectors assess reasoning trajectories or rely on answer-level artifacts. Our Force and Remove oracle tests offer a simple, model-agnostic diagnostic: with the reasoning body preserved, a reasoning-faithful detector should stay informative when the final answer is forced to the ground truth or answer-announcement steps are removed. As reasoning defines frontier models and hallucination detection becomes a relied-upon safety mechanism, evaluation standards must keep pace. Strong performance on unperturbed traces is insufficient if it collapses, or persists for the wrong reasons. We urge the community to adopt oracle robustness testing as standard practice, so that progress reflects genuine reasoning assessment.

Limitations Force and Remove should be understood as necessary sanity checks, not sufficient guarantees of reasoning fidelity. Passing these tests does not prove that a detector verifies intermediate logical steps; it only rules out certain forms of answer-level dependence. TRACT inherits this limitation by design: its lexical and structural features are interpretable and efficient, but they are not a semantic proof of correctness. Our evaluation is limited to benchmark tasks with canonical answers; generalizing the framework to open-ended long-form generation, tool use, and domains where correctness is not reducible to final answers requires additional intervention designs.

REFERENCES

Samir Abdaljalil, Hasan Kurban, Parichit Sharma, Erchin Serpedin, and Rachad Atat. Sindex: Semantic inconsistency index for hallucination detection in llms, 03 2025.

Amazon Web Services. Amazon nova foundation models. https://aws.amazon.com/ai/generative-ai/nova/, 2025. Nova Pro multimodal foundation model.

Jakub Binkowski, Denis Janiak, Albert Sawczyn, Bogdan Gabrys, and Tomasz Jan Kajdanowicz. Hallucination detection in LLMs using spectral features of attention maps. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 24354–24385, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.1239. URL https://aclanthology.org/2025.emnlp-main.1239/.

Dylan Bouchard, Mohit Singh Chauhan, David Skarbrevik, Ho-Kyeong Ra, Viren Bajaj, and Zeya Ahmad. Uqlm: A python package for uncertainty quantification in large language models. *Journal of Machine Learning Research*, 27(13):1–10, 2026. URL http://jmlr.org/papers/v27/25-1557.html.

Jiuhai Chen and Jonas Mueller. Quantifying uncertainty in answers from any language model and enhancing their trustworthiness. *arXiv preprint arXiv:2308.16175*, 2023.

Jeremy R. Cole, Michael J. Q. Zhang, Daniel Gillick, Julian Martin Eisenschlos, Bhuwan Dhingra, and Jacob Eisenstein. Selectively answering ambiguous questions. *arXiv preprint arXiv:2305.14613*, 2023.

Jinhao Duan, Hao Cheng, Shiqi Wang, Alex Zavalny, Chenan Wang, Renjing Xu, Bhavya Kailkhura, and Kaidi Xu. Shifting attention to relevance: Towards the predictive uncertainty quantification of free-form large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5050–5063, 2024.

Zenghao Duan, Liang Pang, Zihao Wei, Wenbin Duan, Yuxin Tian, Shicheng Xu, Jingcheng Deng, Zhiyi Yin, and Xueqi Cheng. Circular reasoning: Understanding self-reinforcing loops in large reasoning models. *arXiv preprint arXiv:2601.05693*, 2026.

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

Ekaterina Fadeeva, Aleksandr Rubashevskii, Artem Shelmanov, Sergey Petrakov, Haonan Li, Hamdy Mubarak, Evgenii Tsymbalov, Gleb Kuzmin, Alexander Panchenko, Timothy Baldwin, Preslav Nakov, and Maxim Panov. Fact-checking the output of large language models via token-level uncertainty quantification. *arXiv preprint arXiv:2403.04696*, 2024.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630:625–630, 2024. doi: 10.1038/s41586-024-07421-0.

Xiang Gao, Jiaxin Zhang, Lalla Mouatadid, and Kamalika Das. Spuq: Perturbation-based uncertainty quantification for large language models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 2336–2346, 2024. doi: 10.18653/v1/2024.eacl-long.143.

Longling Geng, Andy Ouyang, Theodore Wu, et al. CausalT5K: Diagnosing and informing refusal for trustworthy causal reasoning of skepticism, sycophancy, detection-correction, and rung collapse. *arXiv preprint arXiv:2602.08939*, 2026.

Google DeepMind. Gemma 3 technical report. *arXiv preprint*, 2025.

Yashvir S. Grewal, Edwin V. Bonilla, and Thang D. Bui. Improving uncertainty quantification in large language models via semantic embeddings, 2024. URL https://arxiv.org/abs/2410.22685.

Mingyu Jin, Qinkai Yu, Dong Shu, Haiyan Zhao, Wenyue Hua, Yanda Meng, Yongfeng Zhang, and Mengnan Du. The impact of reasoning step length on large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Findings of the Association for Computational Linguistics: ACL 2024*, pages 1830–1842, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.108. URL https://aclanthology.org/2024.findings-acl.108/.

Sungmin Kang, Yavuz Faruk Bakman, Duygu Nur Yaldiz, Baturalp Buyukates, and Salman Avestimehr. Uncertainty quantification for hallucination detection in large language models: Foundations, methodology, and future directions. *arXiv preprint arXiv:2510.12040*, 2025.

Denys Katerenchuk and Rivka Levitan. "you should probably read this": Hedge detection in text, 2024. URL https://arxiv.org/abs/2405.13319.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *arXiv preprint arXiv:2302.09664*, 2023.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, pages 611–626, 2023.

George Lakoff. Hedges: A study in meaning criteria and the logic of fuzzy concepts. *Journal of Philosophical Logic*, 2(4):458–508, 1973. doi: 10.1007/BF00262952.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. *arXiv preprint arXiv:2305.20050*, 2023.

Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9004–9017. Association for Computational Linguistics, 2023a. doi: 10.48550/arxiv.2303.08896.

Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *arXiv preprint arXiv:2303.08896*, 2023b.

Alexander V Nikitin, Jannik Kossen, Yarin Gal, and Pekka Marttinen. Kernel language entropy: Fine-grained uncertainty quantification for LLMs from semantic similarities. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=j2wCrWmgMX.

NVIDIA Corporation. Nemotron-3-nano-30b-a3b model card. https://build.nvidia.com/nvidia/nemotron-3-nano-30b-a3b/modelcard, 2025. Hybrid Mamba-Transformer Mixture-of-Experts LLM.

OpenAI. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*, 2025.

Anirudh Phukan, Divyansh, Harshit Kumar Morj, Vaishnavi, Apoorv Saxena, and Koustava Goswami. Beyond logit lens: Contextual embeddings for robust hallucination detection & grounding in VLMs. In Luis Chiruzzo, Alan Ritter, and Lu Wang, editors, *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 9661–9675, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-189-6. doi: 10.18653/v1/2025.naacl-long.488. URL https://aclanthology.org/2025.naacl-long.488/.

Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=Ti67584b98.

Ola Shorinwa, Zhiting Mei, Justin Lidard, Allen Z. Ren, and Anirudha Majumdar. A survey on uncertainty quantification of large language models: Taxonomy, open research challenges, and future directions. *ACM Computing Surveys*, 58(3):63, 2025. doi: 10.1145/3744238.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V Le, Ed H Chi, Denny Zhou, , and Jason Wei. Challenging big-bench tasks and whether chain-of-thought can solve them. *arXiv preprint arXiv:2210.09261*, 2022.

Arne Vanhoyweghen, Brecht Verbeken, Andres Algaba, and Vincent Ginis. Lexical hints of accuracy in llm reasoning chains, 2025. URL https://arxiv.org/abs/2508.15842.

Changyue Wang, Weihang Su, Qingyao Ai, and Yiqun Liu. Joint evaluation of answer and reasoning consistency for hallucination detection in large reasoning models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(39):33377–33385, Mar. 2026. doi: 10.1609/aaai.v40i39.40624. URL https://ojs.aaai.org/index.php/AAAI/article/view/40624.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837, 2022. URL https://neurips.cc.

Junchi Yao, Shu Yang, Jianhua Xu, Lijie Hu, Mengdi Li, and Di Wang. Understanding the repeat curse in large language models from a feature perspective. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Findings of the Association for Computational Linguistics: ACL 2025*, pages 7787–7815, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.406. URL https://aclanthology.org/2025.findings-acl.406/.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*, 2020.

Geigh Zollicoffer, Minh Vu, and Manish Bhattarai. Mtre: Multi-token reliability estimation for hallucination detection in vlms, 2025. URL https://arxiv.org/abs/2505.11741.

A Datasets

We describe the four benchmarks used to evaluate TRACT and the other black-box scorers.

A.1 BBH-Tracking (BIG-Bench Hard)

BIG-Bench Hard (BBH; Suzgun et al. 2022) is a curated subset of 23 tasks drawn from the broader BIG-Bench evaluation suite, selected because prior language model evaluations failed to exceed average human-rater performance on them under standard few-shot prompting. BBH spans a diverse range of reasoning skills—multi-step algorithmic reasoning, logical deduction, natural language understanding, and commonsense inference—with 250 examples per task (6,511 total). Each task includes both answer-only and chain-of-thought (CoT) few-shot prompt sets. A central finding of the original paper is that CoT prompting substantially closes the human-model gap, enabling Codex to surpass human-rater performance on 17 of the 23 tasks.

We focus on the **Tracking Shuffled Objects** subtask. Each problem specifies N agents each initially holding one of N distinct objects, then narrates N pairwise swaps; the model must identify which object a designated agent holds at the end. Problems are constructed so that every agent participates in at least one swap and no two agents swap back-to-back, ruling out trivial surface shortcuts. Correct solutions require maintaining an explicit, incrementally updated state across multiple reasoning steps, which makes the coherence and rhetorical structure of the chain of thought directly diagnostic of solution quality.

A.2 GPQA DIAMOND

GPQA (Graduate-Level Google-Proof Q&A; Rein et al. 2024) is a benchmark of 448 four-way multiple-choice questions written by domain experts in biology, physics, and chemistry. Questions are deliberately designed to be "Google-proof": highly skilled non-expert validators reach only 34% accuracy despite an average of over 30 minutes of unrestricted web access. Domain experts holding or pursuing PhDs in the relevant fields achieve 65% accuracy (74% after discounting questions the experts themselves identified as flawed in retrospect).

GPQA Diamond is the hardest curated subset of 198 questions, retaining only those for which both independent expert annotators answered correctly while the majority of non-expert validators answered incorrectly. The random-chance baseline on Diamond is 25%. This subset is now a standard frontier evaluation: at the time of the benchmark's release, GPT-4 achieved 39%; subsequent models have improved substantially, but the dataset remains a meaningful discriminator for state-of-the-art systems. For TRACT evaluation, GPQA Diamond provides a pool of problems demanding deep, multi-step scientific reasoning where superficial lexical cues are insufficient and the internal structure of a reasoning trace is strongly predictive of correctness.

A.3 MATH-500

MATH-500 (Lightman et al., 2023) is a 500-problem subset of the MATH dataset, curated by OpenAI to support evaluation of process supervision methods for mathematical reasoning. The full MATH dataset comprises 12,500 competition-style problems drawn from AMC 10, AMC 12, AIME, and related contests; MATH-500 samples representatively from seven subject areas—Prealgebra, Algebra, Number Theory, Counting and Probability, Geometry, Intermediate Algebra, and Precalculus—across five difficulty levels (1–5). Each problem includes a complete step-by-step reference solution in LaTeX.

MATH-500 was introduced to compare outcome supervision (feedback on final answers only) against process supervision (feedback on each intermediate reasoning step), with process supervision found to significantly outperform outcome supervision. This is precisely the regime where the quality of intermediate reasoning steps, rather than the final answer alone, determines success. MATH-500 therefore provides a natural evaluation bed for TRACT's trace-level coherence features: problems are hard enough that flawed intermediate steps frequently co-occur with incorrect final answers, making correct step-by-step structure a reliable signal of correctness.

A.4 CAUSALT5K

CausalT5K (Geng et al., 2026) is a diagnostic benchmark of over 5,000 cases (5,147 total) spanning 10 domains, designed to stress-test three distinct failure modes in LLM causal reasoning: (1) *rung collapse*, where a model responds to an interventional or counterfactual query with associational evidence, conflating different levels of Pearl's causal hierarchy; (2) *sycophantic drift*, where a model abandons a correct causal claim under adversarial rhetorical pressure; and (3) *miscalibrated refusal*, where a model either refuses valid causal claims (the Skepticism Trap) or endorses underdetermined ones.

Unlike purely synthetic causal benchmarks, CausalT5K embeds causal traps in realistic narrative scenarios across domains including medicine, economics, history, sports, and daily life, developed through a rigorous human-machine collaborative pipeline involving 40 domain experts. Performance is decomposed into a two-axis scheme of Utility (sensitivity—correctly affirming valid causal claims) and Safety (specificity—correctly rejecting invalid ones), revealing failure modes invisible to aggregate accuracy. CausalT5K spans all three of Pearl's rungs (associational $\mathcal{L}_1$, interventional $\mathcal{L}_2$, counterfactual $\mathcal{L}_3$) and is unique among existing causal benchmarks in annotating trap type, applying adversarial pressure, and reporting the two-axis decomposition. For TRACT, this dataset provides a qualitatively distinct reasoning challenge relative to the other benchmarks: correct responses require not only multi-step deduction but also resistance to misleading narrative framing, making trace-level rhetorical and coherence features especially salient.

B Model Configurations and Sampling Procedures

B.1 Models

We evaluate across five models spanning four LLM families and a range of parameter scales, mixing open-weight and proprietary systems to reduce the risk of findings that are idiosyncratic to a single architecture or training pipeline. Table 5 summarises the key properties of each model.

Table 5: Models used in our evaluation. All models are instruction-tuned variants. Parameter counts are approximate and reflect the publicly reported size of each model.

Model	Family	Parameters	Access
Nemotron-30B (NVIDIA Corporation, 2025)	Nemotron (NVIDIA)	30B	Open weights
GPT-OSS-120B (OpenAI, 2025)	GPT (OpenAI)	120B	Open weights
LLaMA-3-70B (Dubey et al., 2024)	LLaMA (Meta)	70B	Open weights
Amazon Nova Pro (Amazon Web Services, 2025)	Nova (Amazon)	—	API
Gemma-3-27B (Google DeepMind, 2025)	Gemma (Google)	27B	Open weights

B.2 Prompting

All models are prompted with a standardised step-by-step instruction that elicits chain-of-thought reasoning before a final answer. The system prompt is:

```
Solve the following problem step by step. Show your
reasoning at each step, then provide your final answer on
the last line in the format: "Final Answer: <answer>".
```

We do not use few-shot exemplars; all evaluations are zero-shot. The prompt is identical across all models for BBH-Tracking, Math-500, and GPQA-Diamond to ensure that variation in reasoning traces reflects model behaviour rather than prompt engineering.

For CausalT5k, we utilize the prompt:

```
You are evaluating a causal claim. Read the scenario and
claim carefully, then reason through your answer step by
step.
Scenario: {scenario}
Claim: {claim}
Variables:
- X (exposure): {variables.X.name}
- Y (outcome): {variables.Y.name}
- Z (confounders/other factors): {variables.Z}
Your task: Evaluate whether the claim is causally valid.
Think through your reasoning step by step, then provide your
answer.
Step 1: Identify the causal relationship being claimed.
Step 2: Consider potential confounders or alternative
explanations.
Step 3: Evaluate whether the evidence supports a causal
relationship.
Step 4: Determine if additional information is needed to
make a reliable judgment.
Step 5: State your conclusion.
Final Answer: [YES, NO, or REFUSE]
Explanation:
```

B.3 SAMPLING

For each (model, benchmark) pair, we sample K independent reasoning traces per prompt using nucleus sampling. Table 6 reports the sampling hyperparameters.

Table 6: Sampling hyperparameters. All models use the same configuration except where noted.

Parameter	Value
Responses per prompt (K)	10
Temperature	0.3
Top-p (nucleus)	0.95
Max output tokens	2048

B.4 STEP EXTRACTION

Reasoning traces are segmented into steps by splitting on double newlines. When a model produces unstructured output (e.g. a single block of prose with no paragraph breaks), we fall back to splitting on numbered or bulleted list boundaries and, as a last resort, single newlines. Steps shorter than 5 characters or consisting entirely of punctuation and markdown formatting are discarded. Steps matching answer-announcement patterns (e.g. "Final Answer:", "The answer is") are removed from the trace body before feature extraction to prevent answer-level artifacts from leaking into trajectory features.

B.5 ANSWER EXTRACTION PROCEDURE

To explicitly discuss the construction of the REMOVE condition, we first process all model responses and reasoning steps using GPT-OSS-120B. The model is prompted to identify a broad span in which the target model explicitly announces its final answer. We use these outputs to construct a comprehensive set of regular-expression patterns covering common answer-announcement formats. The resulting patterns are then manually reviewed and validated to ensure that they correspond to explicit answer announcements. Our procedure intentionally removes only explicit answer announcements, that exist in any step number.

By relying on clearly defined and unambiguous linguistic markers, we reduce ambiguity and improve reproducibility in the construction of the REMOVE condition.

B.6 INFRASTRUCTURE

Open-weight models are served via vLLM (Kwon et al., 2023) on NVIDIA A100 GPUs with tensor parallelism as needed (TP=2 for LLaMA-3-70B, TP=1 for all others). Amazon Nova Pro is accessed through the Amazon Bedrock API. All inference is run with `TRANSFORMERS_OFFLINE=1` using locally cached model weights.

C FEATURE INVENTORY

Table 1 in the main paper gives the complete TRACT feature inventory, including the diagnostic intuition, sign, and implementation-level definition for each feature. Here we provide additional implementation details.

C.1 TRACKING TYPOGRAPHIC STRUCTURE FOR FURTHER IMPROVEMENTS

We found that Nova-Pro does encode uncertainty in its reasoning traces, but primarily through typographic structure (e.g., bolding and numbering) rather than the prose-level cues measured by TRACT. To test this, we introduce two non-TRACT features capturing cross-sample variation in typographic markup: *bold-text dispersion* and *numbered-list dispersion*. We then evaluate their association with correctness.

We compute per-feature AUCs with bootstrap 95% confidence intervals on BBH-Tracking:

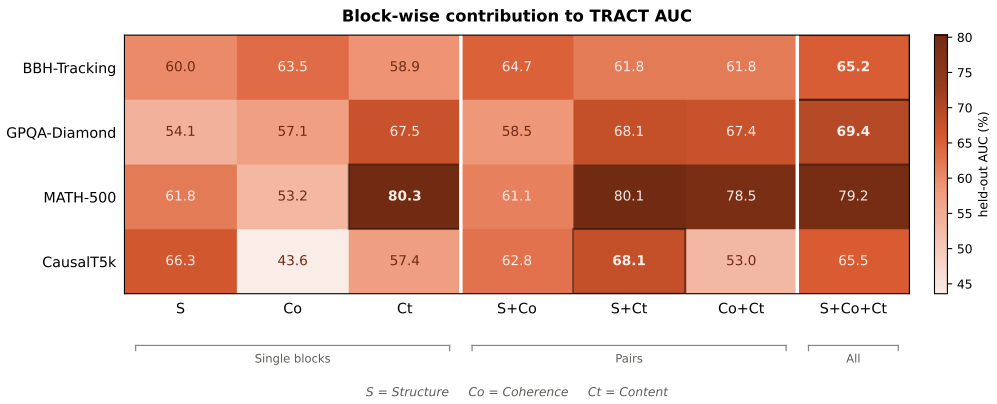


Figure 5: Block-wise contribution to TRACT AUC across four benchmarks, averaged over five models. S = Structure (cross-response rhetorical and structural signals), Co = Coherence, Ct = Content (cross-response lexical agreement). White dividers separate single-block ablations, pairwise combinations, and the full scorer. Bold outlines mark the highest-scoring subset per dataset.

Table 7: AUCs for typographic dispersion features on BBH-Tracking.

Model	Bold dispersion	Numbered-list dispersion
Nova-Pro	**60.1**	**60.6**

Both bold dispersion and numbered-list dispersion, outperform the TRACT features on this model. When Nova-Pro is uncertain, it varies how it numbers and bolds its responses across samples. Its deliberation is observable, but is encoded in layout rather than in lexical choice. For features defined independently within a trace, we compute the statistic for each of the K sampled traces and then average across traces. For cross-trace divergence features, we compute the average pairwise Jaccard distance across all unordered trace pairs.

The hedge count $h_i^{(k)}$ is computed using a fixed lexicon $\mathcal{H}$ containing uncertainty and contrast markers such as *however*, *although*, *maybe*, *perhaps*, *might*, *could*, *seems*, and *hmm*. The entity set $\mathcal{E}_i^{(k)}$ is approximated by capitalised tokens, excluding sentence-initial function words and answer-formatting tokens. Slopes are ordinary least-squares slopes against the normalised step position i/T_k within each trace.

Before scoring, each feature is robust-scaled using median centring and IQR normalisation, then clipped to $[-3, 3]$. The structure block always contributes to the final score. The coherence and content blocks are modulated by the verbosity gate described in Section 3.

C.2 Recall, Sampling Budget Performance, and Brier Score

We report precision at 60% recall and Brier score averaged across all benchmarks and models (normal setting) in Table 8. TRACT outperforms the best baseline on both metrics, with higher precision at the same recall and a lower Brier score. In terms of cost, TRACT degrades gracefully with fewer samples: $K = 3$ and $K = 5$ retain 88.1% and 93.8% of $K = 10$ performance, respectively, making it practical under tighter sampling, and efficient relative to single-pass methods.

C.3 Sweep over gates

For transparency, we evaluated TRACT across a broad grid of possible Gaussian-gate configurations while holding all other components fixed. The results show a broad region of comparable performance rather than a narrow isolated optimum, indicating that TRACT is not highly sensitive to the precise gate configuration.

Table 8: Performance averaged across all benchmarks and models in the normal setting.

Metric	Best baseline	TRACT
Precision at 60% recall	62.9%	**66.2%**
Brier score ↓	0.1779	**0.1678**

Table 9: Pooled mean AUC over 20 dataset–model cells for different Gaussian-gate configurations. Each cell reports mean AUC ± the half-width of the 95% confidence interval. Bold indicates the maximum within each row.

	$\sigma = 5$	$\sigma = 10$	$\sigma = 15$	$\sigma = 20$	$\sigma = 25$	$\sigma = 30$	$\sigma = 35$	$\sigma = 40$	$\sigma = 45$	$\sigma = 50$
$\mu = 10$	61.19 ± 3.2	60.02 ± 3.4	59.43 ± 3.5	59.21 ± 3.4	59.21 ± 3.4	59.11 ± 3.4	58.99 ± 3.4	58.90 ± 3.4	58.82 ± 3.4	$\mathbf{58.76 \pm 3.4}$
$\mu = 20$	$\mathbf{65.23 \pm 2.9}$	64.02 ± 3.2	62.72 ± 3.4	61.81 ± 3.7	61.23 ± 3.6	60.70 ± 3.7	60.33 ± 3.6	60.02 ± 3.7	59.80 ± 3.6	59.58 ± 3.5
$\mu = 30$	66.41 ± 2.9	$\mathbf{66.98 \pm 2.9}$	66.48 ± 2.7	65.49 ± 3.0	64.23 ± 3.3	63.23 ± 3.5	62.45 ± 3.9	61.94 ± 3.7	61.41 ± 3.7	60.95 ± 3.7
$\mu = 40$	65.50 ± 3.0	66.00 ± 2.9	66.64 ± 2.9	$\mathbf{66.90 \pm 2.7}$	66.50 ± 2.7	65.81 ± 2.8	64.81 ± 3.2	64.09 ± 3.3	63.27 ± 3.6	62.58 ± 3.8
$\mu = 50$	65.47 ± 3.1	65.54 ± 3.1	65.90 ± 2.9	66.33 ± 2.9	$\mathbf{66.65 \pm 2.9}$	66.64 ± 2.7	66.25 ± 2.8	65.83 ± 2.8	64.94 ± 3.2	64.35 ± 3.3
$\mu = 60$	65.38 ± 3.1	65.50 ± 3.1	65.64 ± 3.0	65.83 ± 3.0	66.11 ± 3.0	66.39 ± 2.9	$\mathbf{66.48 \pm 2.8}$	66.33 ± 2.6	66.07 ± 2.6	65.79 ± 2.7
$\mu = 70$	65.52 ± 3.0	65.51 ± 3.0	65.53 ± 3.0	65.63 ± 3.0	65.78 ± 3.0	66.01 ± 2.9	66.19 ± 2.9	66.37 ± 2.8	$\mathbf{66.45 \pm 2.7}$	66.30 ± 2.7
$\mu = 80$	65.46 ± 3.0	65.45 ± 3.0	65.43 ± 3.0	65.43 ± 3.0	65.54 ± 3.0	65.70 ± 2.9	65.95 ± 2.9	66.15 ± 2.9	66.31 ± 2.8	$\mathbf{66.43 \pm 2.7}$
$\mu = 90$	65.37 ± 3.1	65.28 ± 3.0	65.26 ± 3.0	65.31 ± 3.0	65.40 ± 3.0	65.49 ± 3.0	65.66 ± 3.0	65.88 ± 3.0	66.09 ± 3.0	$\mathbf{66.26 \pm 2.9}$
$\mu = 100$	65.27 ± 3.0	65.27 ± 3.0	65.28 ± 3.0	65.30 ± 3.0	65.34 ± 3.1	65.42 ± 3.1	65.49 ± 3.0	65.64 ± 3.0	65.84 ± 3.0	$\mathbf{66.04 \pm 2.9}$

The gate is the same $\alpha(x)$ defined in Section 3, written here as a function of its parameters:

$$\alpha(\mathrm{wps}; \mu, \sigma) = \exp\left(-\frac{(\mathrm{wps} - \mu)^2}{2\sigma^2}\right),$$

where wps denotes the mean number of words per reasoning step.

More importantly, performance remains similar across a broad region of the grid, particularly for $\mu \in [30, 80]$, supporting the conclusion that the method is not dependent on tuned gates. For our main results, the gate was chosen based on the terminal-punctuation rate, i.e., whether a step forms a complete sentence. We observed that this rate increased approximately monotonically for steps shorter than 25 words and largely saturated in the 25–30 word range. We therefore selected a gate near the middle of this transition region. The standard deviation ($\sigma = 50$) was chosen arbitrarily; as the sensitivity grid suggests, modest improvements may still be possible with additional tuning.

D TRACT Pairwise Complementarity

The main paper evaluates TRACT as a standalone detector. Here we ask a diagnostic question: does TRACT capture information that is complementary to existing black-box uncertainty scorers? To test this, we fuse TRACT with each baseline scorer using a 4-fold cross-validated logistic regression classifier. All fusion experiments use the unmodified trace setting, class-weighted logistic regression, standardised features, and regularisation parameter $C = 1.0$.

This analysis is not intended to propose an ensemble as the primary method. Instead, it tests whether TRACT contains signal that is not already captured by each baseline. Improvement over EMR indicates information beyond answer repetition; improvement over NCS, BSC, or SEU indicates information beyond embedding or semantic similarity; improvement over NCP or NSN indicates information beyond entailment or semantic-entropy-style uncertainty; and improvement over RACE indicates signal beyond reasoning-consistency and answer-alignment components.

Table 10 summarises the main result. Across all five models, the best TRACT+partner fusion improves over standalone TRACT, with average gains ranging from +5.42 to +20.00 AUC points across benchmarks. The most consistently helpful partners are EMR and NSN, indicating that TRACT complements both simple answer-repetition signals and semantic-entropy-style uncertainty. For GPT-OSS-120B and Nemotron-30B, RACE is also among the most useful partners, suggesting that TRACT's explicit step-level trajectory features provide signal not fully captured by aggregate reasoning-consistency or answer-alignment scores.

Table 10: **Summary of TRACT pairwise complementarity.** For each model, we report standalone TRACT AUC averaged over the four benchmarks, the best average TRACT+partner fusion, and the average gain over standalone TRACT. The final column lists the partners with the highest number of wins over standalone TRACT across benchmarks.

Model	TRACT avg.	Best avg. fusion	Avg. gain	Most consistent partners
Nova-Pro	59.05	79.05	+20.00	EMR, NSN
Gemma-3-27B	74.59	80.01	+5.42	EMR, NSN, NCP
GPT-OSS-120B	71.53	84.10	+12.57	EMR, NSN, RACE
LLaMA-3-70B	64.65	74.72	+10.07	EMR, NCP, BSC, NSN
Nemotron-30B	78.38	88.31	+9.93	EMR, NSN, RACE

The detailed results below show that complementarity is not confined to a single model or benchmark. Nova-Pro benefits strongly from fusion, suggesting that TRACT supplies missing trajectory information when standalone performance is modest. Gemma-3-27B and Nemotron-30B already have strong standalone TRACT scores, yet fusion still improves performance, indicating that TRACT is compatible with other uncertainty signals rather than simply replacing them. For GPT-OSS-120B and LLaMA-3-70B, the best partner varies by dataset, which further supports the view that no single baseline subsumes TRACT's trajectory features.

In Tables 11–15, **bold** entries indicate fusion scores that improve over both standalone TRACT and the partner scorer on the corresponding dataset. The final row, *Wins vs. T*, counts the number of benchmarks, out of four, where TRACT+partner improves over standalone TRACT, regardless of whether it also beats the partner alone.

D.1 NOVA-PRO

Table 11: **TRACT fusion results for Nova-Pro.** Nova-Pro shows the largest complementarity gains overall. EMR and NSN are the most consistent partners, indicating that TRACT contributes trajectory information that combines well with both endpoint repetition and semantic uncertainty.

Dataset	T	+EMR	+NCP	+BSC	+NCS	+NSN	+SEU	+RACE
MATH-500	78.37	**92.77**	**79.64**	78.19	**79.96**	**91.85**	**88.42**	**84.77**
GPQA Diamond	56.79	**70.51**	56.68	56.00	55.01	**70.10**	**68.73**	57.39
BBH-Tracking	44.59	71.65	27.43	26.34	20.08	61.30	68.41	64.78
CausalT5K-D4	56.45	**81.27**	54.70	52.66	50.30	**79.66**	53.41	51.91
Wins vs. T	–	4/4	1/4	0/4	1/4	4/4	2/4	2/4

D.2 GEMMA-3-27B

Table 12: **TRACT fusion results for Gemma-3-27B.** Standalone TRACT is already strong on Gemma-3-27B, but fusion still improves on most benchmarks. The broad usefulness of EMR, NSN, and NCP suggests that trajectory features remain complementary even when answer-level and entailment-style signals are available.

Dataset	T	+EMR	+NCP	+BSC	+NCS	+NSN	+SEU	+RACE
MATH-500	82.97	**86.30**	**83.40**	82.87	**83.10**	**87.03**	93.84	93.56
GPQA Diamond	75.36	**77.92**	**75.55**	74.66	75.23	**79.16**	**77.73**	75.29
BBH-Tracking	77.03	**77.21**	76.84	**78.51**	**78.42**	75.53	74.69	74.55
CausalT5K-D4	63.01	**68.16**	**63.68**	59.07	61.88	**68.54**	61.82	**63.27**
Wins vs. T	–	4/4	3/4	1/4	2/4	3/4	2/4	2/4

D.3 GPT-OSS-120B

Table 13: **TRACT fusion results for GPT-OSS-120B.** Fusion improves strongly on BBH-Tracking and CausalT5K-D4, where trajectory instability and endpoint uncertainty provide complementary evidence. The useful partners vary by dataset, indicating that TRACT is not subsumed by any single baseline family.

Dataset	T	+EMR	+NCP	+BSC	+NCS	+NSN	+SEU	+RACE
MATH-500	81.32	**83.09**	81.21	**81.62**	**82.44**	**81.45**	89.93	**90.23**
GPQA Diamond	80.82	**81.00**	79.17	80.32	80.36	80.71	86.99	85.55
BBH-Tracking	53.63	72.90	46.27	57.65	51.06	76.53	57.64	52.18
CausalT5K-D4	70.34	**81.18**	**78.22**	**71.59**	**71.78**	**79.14**	**82.66**	**73.76**
Wins vs. T	–	4/4	1/4	2/4	2/4	2/4	3/4	3/4

D.4 LLaMA-3-70B

Table 14: **TRACT fusion results for LLaMA-3-70B.** For LLaMA-3-70B, EMR, NCP, BSC, and NSN frequently improve over standalone TRACT. This pattern suggests that TRACT is most complementary to lexical, entailment, and semantic-entropy signals for this model, while SEU and RACE are less consistently helpful.

Dataset	T	+EMR	+NCP	+BSC	+NCS	+NSN	+SEU	+RACE
MATH-500	75.35	**91.59**	**77.11**	**75.38**	**77.66**	**89.32**	85.46	78.96
GPQA Diamond	62.90	62.83	**63.23**	**63.81**	62.42	62.63	60.97	62.09
BBH-Tracking	60.38	**79.46**	51.23	**60.70**	**63.18**	**78.32**	76.11	63.30
CausalT5K-D4	59.96	**61.39**	**62.93**	49.94	58.35	**63.66**	58.72	64.02
Wins vs. T	–	3/4	2/4	3/4	2/4	2/4	1/4	3/4

D.5 Nemotron-30B

Table 15: **TRACT fusion results for Nemotron-30B.** TRACT remains complementary despite strong standalone performance. EMR, NSN, and RACE improve on all four benchmarks, showing that TRACT's trajectory-level statistics combine well with both endpoint-based and reasoning-aware uncertainty signals.

Dataset	T	+EMR	+NCP	+BSC	+NCS	+NSN	+SEU	+RACE
MATH-500	76.42	**76.95**	**81.62**	**79.33**	**81.01**	**78.15**	87.34	**81.15**
GPQA Diamond	77.59	**87.72**	77.59	77.03	**78.22**	**87.75**	76.80	**81.24**
BBH-Tracking	84.05	**97.21**	83.58	**84.24**	82.98	**92.48**	**94.52**	**92.12**
CausalT5K-D4	75.44	**80.95**	**76.05**	**78.45**	**76.07**	**80.65**	**75.45**	**79.66**
Wins vs. T	–	4/4	2/4	3/4	3/4	4/4	2/4	4/4

Overall, the fusion analysis supports the interpretation of TRACT as a complementary trajectory scorer rather than a reparameterisation of existing uncertainty baselines. The strongest partners differ across models and datasets, but the consistent gains over standalone TRACT indicate that its coherence, structure, and content features capture trace-level information that can be combined with answer repetition, semantic similarity, entailment, semantic entropy, and reasoning-consistency signals.