MEASURING INTERNAL UNCERTAINTY IN LANGUAGE MODEL AGENTS: A MULTI-METHOD TRIANGULATION APPROACH

Anonymous authors
Paper under double-blind review

ABSTRACT

Adaptive information acquisition requires an agent to assess uncertainty about its current judgement. For large language models (LLMs), however, it remains unclear which observable signals provide useful measures of this internal uncertainty. We use a controlled probabilistic reasoning task with evidence that is either ambiguous, strongly supports the correct answer, or strongly supports an incorrect answer. This design allows us to test whether candidate uncertainty measures respond to how conclusive the evidence is, even when it is misleading. We compare candidate measures derived from verbal reports, answer-option distributions, opt-out behaviour, reasoning traces, and internal representations, measured in GPT-OSS-20B and Qwen3.5-27B. We evaluate the ability of these candidate indicators to distinguish evidence conditions, retain a set of effective indicators, and assess their convergence across measurement methods. We construct two composite indices that consistently distinguish evidence conditions in both models. In Qwen3.5-27B, higher measured uncertainty is further associated with sampling beyond the model's currently preferred cue. These findings provide a measurement basis for investigating how internal uncertainty relates to information acquisition and decision-making.

1 INTRODUCTION

An effective agent must decide not only what to do, but also when it has enough information to act. A conversational assistant may need to ask a clarifying question before answering an ambiguous request; a coding agent may need to inspect additional files or run checks before proposing a change. Such decisions require an assessment of what remains uncertain and whether further information could change the appropriate response. Because information gathering incurs costs, an agent should continue sampling when its expected benefit outweighs its cost and act when further sampling is no longer worthwhile. Internal uncertainty is therefore a potential signal for regulating information acquisition and action.

How to endow large language models (LLMs) with such adaptive information-sampling behaviour remains an open problem. In human cognition, deciding whether to gather more information or act on existing evidence involves metacognitive control: individuals assess their own uncertainty and use that assessment to guide further information acquisition. Recent evidence suggests that LLMs can likewise use internal confidence signals to regulate answer-versus-abstention decisions (Kumaran et al., 2026), raising the possibility that such signals also inform their sampling choices. To investigate this possibility, we first need to establish how such internal signals can be measured.

We study internal uncertainty, defined as the model's internally generated assessment, given the information acquired so far, of whether its representation of the task-relevant world is sufficiently stable to support a response or action. High internal uncertainty indicates that the model considers additional information likely to substantially revise its current representation; low internal uncertainty indicates that it considers its current representation sufficiently stable to act upon.

Existing work has proposed diverse candidate indicators of internal uncertainty (Section 2), but these are typically validated by their ability to predict answer correctness, which is not a direct ground

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.

truth for internal uncertainty. In particular, low internal uncertainty can coexist with an incorrect answer when the available evidence is misleading.

We therefore treat internal uncertainty as a latent construct and triangulate across measurement approaches without direct ground truth. We assess candidate indicators in terms of discriminability (separating ambiguous from concentrated evidence), discriminant validity (distinguishing uncertainty from support for the true state), and convergent validity (agreement across measurement methods). An ideal Bayesian observer provides external reference conditions rather than ground truth for the model's internal assessment. We then combine retained indicators into two composite indices and examine their behaviour across evidence levels and model-directed sampling, including their relationship with subsequent choices.

2 Uncertainty in Language Model Agents

Uncertainty in machine learning is most commonly divided into *aleatoric* uncertainty, arising from randomness in the data-generating process, and *epistemic* uncertainty, arising from limited knowledge and reducible in principle (Kiureghian & Ditlevsen, 2009; Kendall & Gal, 2017; Hüllermeier & Waegeman, 2021). The aleatoric–epistemic distinction primarily concerns the *source* of uncertainty. Recent work questions this dichotomy: Smith et al. (2025) argue that it cannot express all quantities of interest and propose a decision-theoretic alternative, and Kirchhof et al. (2025) highlight conflicting definitions in interactive LLM settings, where the reducibility of uncertainty depends on information and interaction, making the boundary a modelling choice.

Our emphasis is on uncertainty as a model-internal assessment, irrespective of source: the extent to which the available information permits the model to form a stable representation of an unknown state of the world (in our task, the identity of the biased cue). Because this assessment is not directly observable, and neither correctness nor the available evidence provides its ground truth, we treat it as a latent construct.

Candidate indicators of this internal assessment include verbal reports (Lin et al., 2022; Kadavath et al., 2022; Tian et al., 2023; Xiong et al., 2024), abstention decisions (Kamath et al., 2020; Feng et al., 2024; Wen et al., 2025), output distributions (Malinin & Gales, 2021; Kuhn et al., 2023; Farquhar et al., 2024), reasoning traces (Yoon et al., 2025; Mei et al., 2025), and internal representations (Azaria & Mitchell, 2023; Kossen et al., 2024; Liu et al., 2024; Gurnee et al., 2026; Patel et al., 2026) , which we triangulate rather than validate individually against answer correctness (Section 4.2).

3 Experimental Setup

3.1 Active Probabilistic Reasoning Task

We adapted the active probabilistic reasoning task of Guiomar et al. (2026). Each trial contains four cues, $\mathcal{C} = \{A, B, C, D\}$, exactly one of which is biased. The biased cue, denoted by z^*, produces red outcomes with probability 0.9, whereas the other cues produce red and green outcomes with equal probability. The agent's goal is to identify z^* by sequentially sampling cue outcomes.

Each trial comprises 15 sampling rounds followed by a final inference round. At each sampling round $t \in \{1, \ldots, 15\}$, zero to three cues are temporarily occluded. The agent selects an available cue $X_t \in \mathcal{A}_t \subseteq \mathcal{C}$ and observes an outcome $Y_t \in \{\mathrm{RED}, \mathrm{GREEN}\}$. We denote the accumulated cue selections and outcomes by h_t. Sampling yields no reward; in the final inference round, the agent identifies the biased cue using the accumulated evidence.

3.2 Bayesian Ideal Observer

To evaluate whether candidate uncertainty measures track the evidence available to the model, we require an external characterisation of the evidential state at each round. We obtain this using an ideal Bayesian observer. Starting from a uniform prior over the four cues, the observer updates its beliefs using the task's outcome probabilities to obtain $p_{t,k} = P(Z = k \mid h_t)$, the posterior probability that cue k is biased given the observation history h_t.

We summarise this posterior using two complementary quantities:

$$\mathrm{M1}_t = p_{t,z^*}, \qquad \mathrm{M2}_t = 1 - \frac{\mathrm{H}(p_t)}{\ln 4}, \tag{1}$$

where $\mathrm{H}(p_t) = -\sum_{k\in\mathcal{C}} p_{t,k} \ln p_{t,k}$. The true-cue posterior probability, $\mathrm{M1}_t$, quantifies the extent to which the evidence supports the actual biased cue. The normalised inverse entropy, $\mathrm{M2}_t$, measures posterior concentration irrespective of whether the favoured cue is correct: it is zero for a uniform posterior and approaches one as the posterior concentrates on a single cue.

Comparing concentrated and diffuse posteriors alone can conflate posterior concentration with support for the true state if concentrated posteriors also tend to favour the true cue. By considering $\mathrm{M1}$ alongside $\mathrm{M2}$, we can explicitly include misleading evidence: histories that produce a highly concentrated posterior favouring an incorrect cue. This provides a critical contrast for assessing whether candidate uncertainty measures respond to posterior concentration even when the evidence points away from the true state. The full Bayesian update is provided in Appendix A.

3.3 Simulated Trials

To evaluate candidate measures of internal uncertainty, we construct simulated trial histories with controlled evidence profiles. Each history contains 15 sampling rounds and is presented retrospectively as a single text prompt, including cue availability, selected cues, and observed outcomes. We select and categorise trial histories using $\mathrm{M1}_{15}$ and $\mathrm{M2}_{15}$, computed from the ideal observer's posterior after all 15 sampling rounds:

$$\begin{aligned}
\textsc{Veridical}: &\quad \mathrm{M1}_{15} > 0.8,\ \mathrm{M2}_{15} > 0.8 \quad \text{(concentrated on the true cue)},\\
\textsc{Ambiguous}: &\quad \mathrm{M1}_{15} < 0.2,\ \mathrm{M2}_{15} < 0.2 \quad \text{(diffuse, with low true-cue support)},\\
\textsc{Misleading}: &\quad \mathrm{M1}_{15} < 0.2,\ \mathrm{M2}_{15} > 0.8 \quad \text{(concentrated on an incorrect cue)}.
\end{aligned} \tag{2}$$

These conditions test sensitivity to posterior concentration (Ambiguous versus Veridical and Misleading) separately from support for the true cue (Veridical versus Misleading).

We construct a fixed validation dataset of 300 trials by randomly sampling 100 trials per condition from a pool of 60,000 simulated trials. This same dataset is used to evaluate all candidate uncertainty measures, ensuring that comparisons across measures are based on identical trial histories. Details of trial generation, sampling, and prompts are provided in Appendix B.

3.4 Language Models

We evaluate `openai/gpt-oss-20b` and `Qwen/Qwen3.5-27B` without additional quantisation. GPT-OSS-20B is evaluated under no-, low-, and high-reasoning settings, while Qwen3.5-27B is evaluated with thinking disabled or enabled. Reasoning traces are generated at temperature 1.0, while Qwen verbal-report elicitation uses temperature 0.7. Direct-answer readouts use temperature 0 or non-sampling forward passes. Measurement-specific decoding settings and implementation details are provided in Appendix C, and data coverage is summarised in Table 4.

4 Uncertainty Estimation and Triangulation

4.1 Uncertainty measures

We assess internal uncertainty through candidate measures derived from answer distributions, verbal reports, reasoning traces, opt-out behaviour, and internal representations. We describe their operational definitions below; implementation details are provided in Appendix D.

4.1.1 Behavioural Measures

Verbal reports. For both models, we elicit verbal uncertainty reports under non-thinking and thinking conditions. After providing a cue choice, the model reports its uncertainty about that answer as an integer from 0 ("completely certain") to 100 ("a complete guess, no better than random"),

with higher values indicating greater reported uncertainty. For each trial and condition, we perform one repetition for GPT and ten repetitions for Qwen (Appendix D.1).

Opt-out behaviour. Inspired by the sure-reward opt-out paradigm of Kiani & Shadlen (2009), we construct a variant of the inference task that offers a fifth option, E, providing a guaranteed reward R_{out} instead of identifying the biased cue. Choosing A–D yields R_{correct} for a correct answer and R_{wrong} otherwise. Under linear utility, the probability of being correct at which answering and opting out have equal expected reward is

$$p^* = \frac{R_{\text{out}} - R_{\text{wrong}}}{R_{\text{correct}} - R_{\text{wrong}}}. \tag{3}$$

Thus, answering is preferable when the estimated probability of correctness exceeds p^*, whereas opting out is preferable below this threshold. We record an opt-out response when E has the highest extracted answer-token score among A–E, using an argmax readout rather than sampling the final answer. We use the opt-out response as a behavioural indicator of internal uncertainty (Appendix D.2).

4.1.2 Answer-Option Distribution

Answer-option entropy. The first uncertainty measure is based on the model's next-token distribution over the answer options. We extract the model's next-token probabilities for the tokens `A`, `B`, `C` and `D` (with and without a leading space) at the answer position following the elicitation question (Appendix B.1), either directly or after replaying a generated reasoning trace. For each repetition r, we normalise the extracted option probabilities to obtain $q_k^{(r)}$. We average these distributions over valid repetitions and compute the normalised entropy:

$$\bar{q}_k = \frac{1}{|\mathcal{V}|} \sum_{r \in \mathcal{V}} q_k^{(r)}, \qquad U_{\text{ans}} = -\frac{1}{\ln 4} \sum_{k \in \mathcal{C}} \bar{q}_k \ln \bar{q}_k, \tag{4}$$

where $\mathcal{V}$ is the set of valid repetitions. Higher values indicate greater dispersion. Computing entropy after averaging captures both dispersion within individual answer distributions and disagreement across reasoning traces, including cases where individual traces favour different answers with high confidence (Appendix D.3).

4.1.3 Reasoning-Trace Measures

We examine reasoning length and explicit uncertainty language as candidate indicators of internal uncertainty.

Chain-of-thought length. We measure reasoning length in model-tokenizer tokens for both GPT-OSS-20B and Qwen3.5-27B, excluding the final answer.

Uncertainty words in chain-of-thought. We identify uncertainty expressions using a shared lexicon of 36 words and phrases, applied to lowercased reasoning traces. These measures characterise observable properties of generated reasoning (Appendix D.4).

4.1.4 Internal-Representation Measures

We extract hidden-state representations while each model processes the task instructions and sampling history, before any elicitation question, reasoning trace, or answer. At each sampling round, we extract representations at three positions: the selected cue-letter token (*choice*), the observed colour token (*result*), and the final token of the round's feedback line (*boundary*). We use hidden states indexed 1–24 for GPT and 1–64 for Qwen (Appendix D.5). We investigate these representations using supervised probes, a Jacobian lens, and sparse autoencoder features.

Supervised probes. We train separate probes for each layer–position combination using hidden states from the fifteenth sampling round, extracted before explicit reasoning or answering. Targets comprise uncertainty-language density, uncertainty-language presence or rate, verbal uncertainty reports, and mean post-reasoning answer entropy. We evaluate support vector machines (SVMs) and

multilayer perceptrons (MLPs), using classification for binary targets and regression for continuous targets. Standardisation and PCA are fitted within each training fold.

Five-fold cross-validation assesses whether these uncertainty-related outputs are predictable from internal representations (Appendix D.6). The targets are operational measurements rather than ground-truth labels of internal uncertainty.

Jacobian lens. For GPT-OSS-20B, we use a Jacobian lens to map intermediate representations into the final-layer representation space and decode them through the model's unembedding. The lens is estimated using WikiText 2 without task-specific labels. At the final-round result and boundary positions, we extract the projected logits for A–D and apply a softmax over these options. We quantify the concentration of this distribution as one minus its normalised entropy and examine its variation across evidence conditions (Appendix D.7).

Sparse autoencoder features. We apply pretrained sparse autoencoders (SAEs) to result-position representations obtained during passive processing of the trial history. Within development data, we select features that distinguish AMBIGUOUS from the two concentrated evidence conditions while satisfying an equivalence criterion between VERIDICAL and MISLEADING. Selected activations are standardised, oriented towards greater activation in the ambiguous condition, and averaged to form a candidate uncertainty score. Feature selection and score construction use development data only; the resulting mapping is frozen before evaluation on a separate confirmation set (Appendix D.8).

4.2 VALIDATION CRITERIA

4.2.1 DISCRIMINABILITY AND DISCRIMINANT VALIDITY

Discriminability is a measure's ability to distinguish conditions relative to within-condition variability; here, uncertainty scores should be higher for AMBIGUOUS than for both VERIDICAL and MISLEADING trials. Discriminant validity concerns whether a measure distinguishes the intended construct from related quantities; here, scores should be similar for VERIDICAL and MISLEADING, despite their differing support for the true cue.

We assess the two directional contrasts using one-sided Welch's t tests with Holm correction within each measure, and the VERIDICAL–MISLEADING contrast using a two-sided Welch's t test. Treating trials as the units of analysis, we report Cohen's d, p values, and confidence intervals (90% for directional contrasts; 95% for the two-sided contrast).

4.2.2 CONVERGENT VALIDITY

Convergent validity concerns agreement between different measures of the same construct. We assess it separately for each model using pairwise binary agreement and correlations between original scores. We include measures satisfying both discriminability contrasts and showing no significant VERIDICAL–MISLEADING difference, retaining the variant with the largest minimum effect size across the two discriminability contrasts within each method.

We first convert non-binary scores into uncertainty flags using thresholds that maximise Youden's J (sensitivity plus specificity minus one), treating AMBIGUOUS as positive and the pooled VERIDICAL and MISLEADING conditions as negative. Pairwise agreement is the proportion of trials on which two measures produce the same flag. We then compute pairwise Pearson correlations between the original scores across all evidence conditions.

4.3 COMPOSITE UNCERTAINTY INDICES

We combine the retained measures to integrate complementary signals into a single estimate of internal uncertainty, potentially reducing reliance on the limitations of any individual measure. We construct two composite indices. The first counts the binary indicators that flag a trial as uncertain, using the held-out flags defined above. We examine the distribution of this count across evidence conditions. The second combines the original scores. We apply measure-specific transformations, align their directions so that higher values indicate greater uncertainty, and standardise them to z scores. The composite is the equally weighted mean of these standardised scores.

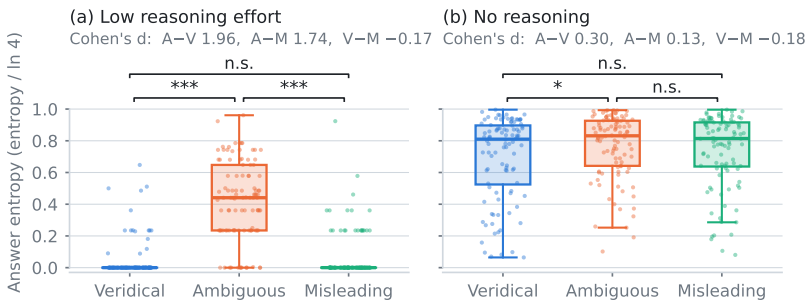

Figure 1: **Answer entropy depends on the reasoning setting.** GPT-OSS-20B answer-entropy distributions under (a) low reasoning effort and (b) no reasoning. Points represent individual trials. Low-effort entropy separates AMBIGUOUS from both concentrated conditions more strongly than direct-answer entropy. Annotated significance levels follow the tests described in the main text; n.s. denotes a non-significant contrast.

4.4 UNCERTAINTY ACROSS EVIDENCE LEVELS AND ACTIVE REASONING

Intermediate evidence levels. For Qwen3.5-27B, we measure candidate indicators at every sampling round of the 300 fixed trial histories. Although these trials are selected by their final-round evidence profiles, earlier rounds also contain intermediate evidence states. We partition the resulting 4,500 trial–round observations into ten equal-width bins of posterior concentration, $\mathrm{M2}$, spanning $[0, 1]$. We randomly sample 147 observations without replacement from each bin, matching the smallest bin and yielding 1,470 observations with equal representation across concentration intervals. We use this sample to examine indicator behaviour across evidence levels. These observations remain drawn from the endpoint-selected trials, and multiple rounds from the same trial are repeated measurements.

Active reasoning task. We additionally conduct 30 active reasoning trials with Qwen3.5-27B. Unlike the fixed histories, sampling choices in these trials are made by the model based on the available cues and accumulated observations. We convert the resulting interaction records into the same history format used for the fixed trials and measure the indicators at every sampling round.

5 RESULTS

5.1 DISCRIMINABILITY AND DISCRIMINANT VALIDITY

A suitable indicator should score AMBIGUOUS above both concentrated conditions, with similar scores for VERIDICAL and MISLEADING. Figure 1 illustrates this pattern for GPT-OSS-20B answer entropy under low reasoning effort: the concentrated conditions cluster near zero, while AMBIGUOUS spans much of the entropy range ($d_{A-V} = 1.96$, $d_{A-M} = 1.74$, both $p < .001$; $d_{V-M} = -0.17$, non-significant). Without reasoning, the distributions overlap at high entropy ($d_{A-V} = 0.30$, $d_{A-M} = 0.13$; the latter non-significant).

Figure 2 summarises all candidates, with purple markers indicating those passing the screen. Discrimination is strongest for low-effort answer entropy in GPT-OSS-20B and CoT length in Qwen3.5-27B ($d_{A-V} = 1.61$, $d_{A-M} = 1.28$, both $p < .001$). Verbal reports pass with and without reasoning in both models, whereas opt-out passes only with reasoning, highlighting the need for setting-specific validation. All four probe targets and SAE pass in GPT-OSS-20B, but Jacobian-lens entropy does not; three probe targets pass in Qwen3.5-27B, while the verbal-report probe and SAE do not. Selecting the strongest qualifying variant per method yields seven GPT-OSS-20B and five Qwen3.5-27B indicators, listed with their optimal binarisation thresholds in Table 1. Full validation results are provided in Appendices E.1 and E.2.

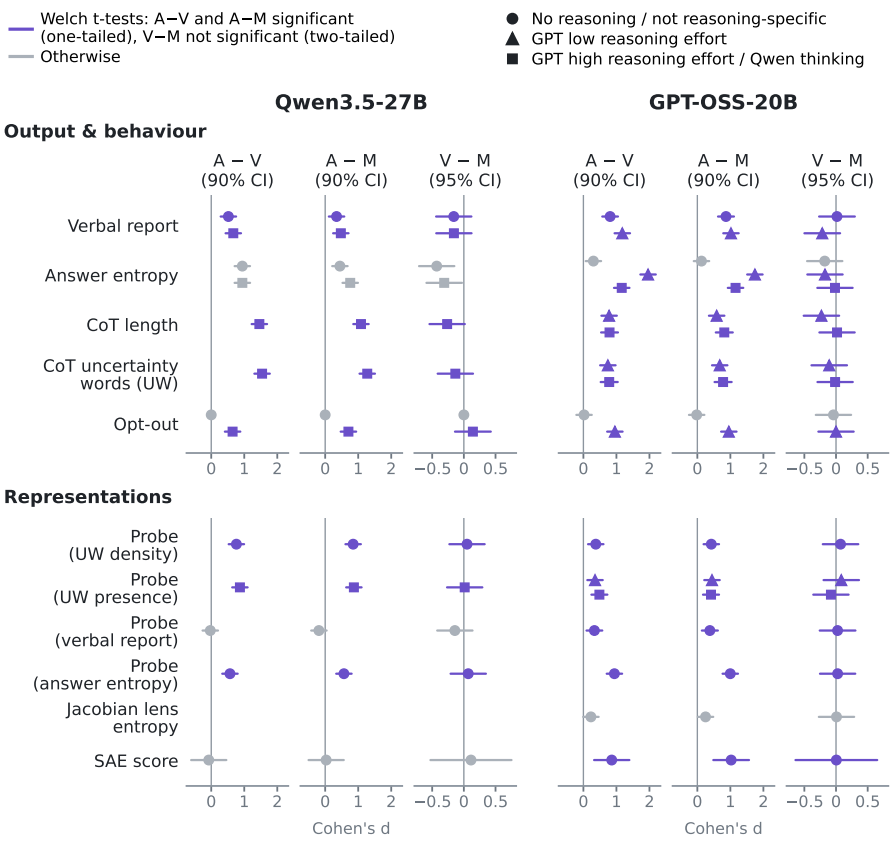

Figure 2: **Evidence-condition contrasts for candidate uncertainty measures.** A, V, and M denote AMBIGUOUS, VERIDICAL, and MISLEADING. Points show Cohen's d, with 90% confidence intervals for A–V and A–M and 95% confidence intervals for V–M. Purple indicates significant directional contrasts for A–V and A–M and a non-significant two-sided V–M contrast; grey indicates other outcomes. Non-significance does not establish equivalence. UW denotes uncertainty words and phrases. For probes, marker shapes indicate the reasoning setting used to construct the target labels. SAE results use the confirmation sets.

Table 1: Retained uncertainty indicators, with their setting and Youden-J-optimal firing threshold.

	GPT-OSS-20B		Qwen3.5-27B	
Indicator	Setting	Threshold	Setting	Threshold
Answer entropy	low effort	$\geq$0.23	–	–
Verbal report	low effort	$\geq$12.5	thinking	$\geq$22.25
CoT length	high effort	$\geq$1974	thinking	$\geq$6303
CoT uncertainty words (UW)	high effort	$\geq$2.6	thinking	$\geq$1.25
Opt-out	low effort	$\geq$1/3 reps	thinking	$\geq$1/3 reps
Probe	answer entropy (low)	$\geq$0.12	UW presence	$\geq$0.61
SAE score	–	$\geq$0.085	–	–

5.2 CONVERGENCE ACROSS MEASURES

The seven retained GPT-OSS-20B indicators show partial convergence, with binary agreement ranging from 0.60 to 0.89 and score correlations from 0.14 to 0.79 (Figure 3). Associations partly cluster by measurement source: the two internal-representation indicators—the answer-entropy probe and SAE score—are closely related (agreement = 0.75, r = 0.67), as are the two CoT-derived indicators (0.89, 0.79), whereas correlations between these pairs are weaker (r = 0.14–0.21). Answer entropy and verbal reports show broader associations across sources. This pattern suggests shared

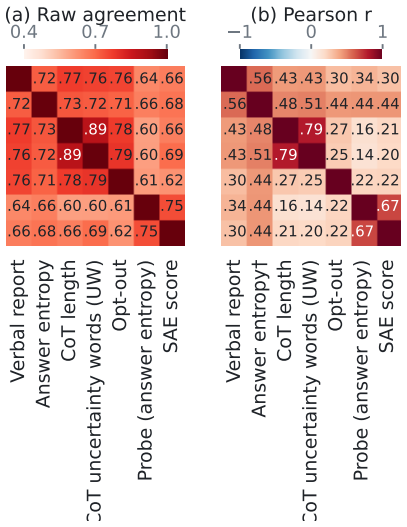

Figure 3: **Inter-indicator agreement, GPT-OSS-20B.** (a) Raw agreement between binary fire flags (Youden-optimal thresholds, positive class = AMBIGUOUS). (b) Pearson r across all trials, corrected for attenuation where estimable. Rows unlabeled: the matrix is symmetric, same order as the columns.

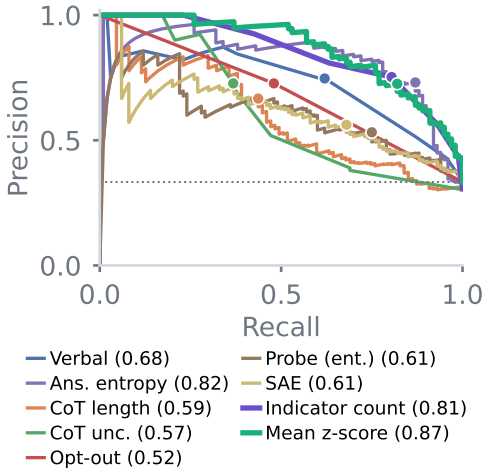

Figure 4: **Precision–recall, GPT-OSS-20B.** Positive class = AMBIGUOUS vs. {VERIDICAL, MISLEADING}; each line sweeps one retained indicator or one of the two composites. Legend values in parentheses are average precision. Dots mark each score's own optimal operating point.

uncertainty-related variation alongside source-dependent structure, rather than interchangeable measurements. Corresponding Qwen3.5-27B results and error-pattern clustering for both models are provided in Appendix E.3.

5.3 COMPOSITE UNCERTAINTY INDICES

We integrate the retained measures into two composites: the *uncertainty indicator count*, which counts threshold crossings, and the *mean standardised uncertainty score*, which averages z scores after log-transforming CoT length and log1p-transforming uncertainty-word scores, using synthetic-trial means and SDs.

Both composites outperform most individual indicators in average precision (AP). For GPT-OSS-20B (Figure 4), the indicator count and mean standardised score achieve 0.81 and 0.87, respectively, versus 0.82 for the best individual indicator. For Qwen3.5-27B, the corresponding scores are 0.73 and 0.74, comparable to the best individual score of 0.74 (Appendix E.4).

Indicator-count distributions differ visibly across conditions, with AMBIGUOUS shifted higher (Figure 5a,c). Mean counts for AMBIGUOUS, VERIDICAL, and MISLEADING are 3.95, 1.18, and 1.21 for GPT-OSS-20B, and 3.23, 1.19, and 1.34 for Qwen3.5-27B. The mean standardised score shows the same separation (panels b,d), supporting robustness to the aggregation choice.

5.4 UNCERTAINTY ACROSS EVIDENCE LEVELS AND ACTIVE REASONING

For Qwen3.5-27B, both composites decrease monotonically with ideal-observer posterior concentration in the synthetic histories (Figure 6a). The indicator count falls from approximately 3.5 of 5 in the lowest-concentration bin to fewer than one in the highest. The mean standardised score follows a similar trajectory, with intermediate evidence levels yielding intermediate scores. These trends extend across intermediate evidence levels and are also observed in the constituent indicators (Appendix E.5).

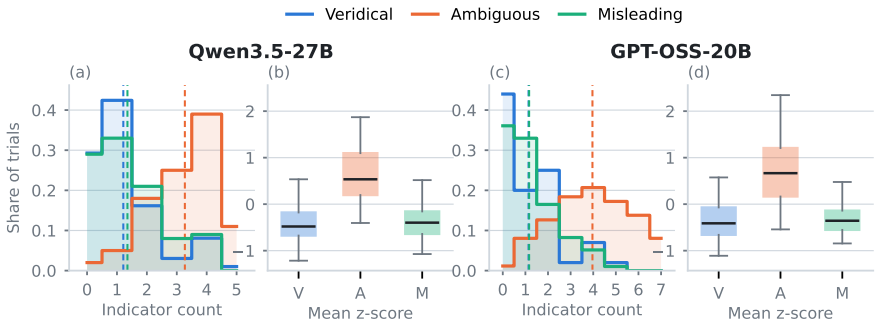

Figure 5: **Fire count and mean z-score by evidence condition.** (a),(c) Indicator count (of 5 for Qwen3.5-27B, 7 for GPT-OSS-20B, incl. SAE) per trial, by condition; dashed lines are condition means. (b),(d) Mean z-score of the same indicators (CoT length log-, CoT uncertainty words log1p-transformed, then z-scored and averaged), by condition. AMBIGUOUS trials score higher on both, for both models.

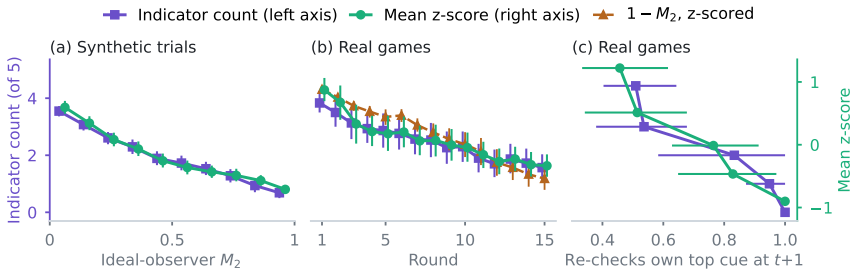

Figure 6: **Composite uncertainty across evidence levels, sampling rounds, and subsequent choices.** Results are for Qwen3.5-27B. Purple squares: indicator count (left axis); green circles: mean z-score (right axis). (a) Scores vs. ideal-observer posterior concentration. (b) Scores vs. sampling round; orange triangles show $1 - M_2$, z-scored. (c) Next-round own-top-cue re-check rate vs. uncertainty at round t.

We next apply the composites to 30 trials in which the model chooses which cues to sample, thereby influencing the evidence it receives. Both scores decline approximately monotonically across rounds (Figure 6b), with the indicator count falling from approximately 3.8 to 1.6. Standardised ideal-observer uncertainty, $1 - \mathrm{M2}$, shows a similar downward trajectory, indicating that the composite scores broadly track the reduction in evidence uncertainty during model-directed sampling.

Both composites also relate to subsequent choices (Figure 6c). The model selects its currently preferred cue on nearly all next-round choices at low measured uncertainty, compared with approximately 0.45–0.50 at high uncertainty. The similar profiles across aggregation methods support robustness to the scoring procedure, while their associations with subsequent choices motivate further investigation of how internal uncertainty relates to information-sampling strategies.

6 CONCLUSION

We evaluate internal uncertainty as a latent construct in GPT-OSS-20B and Qwen3.5-27B using controlled ambiguous, veridical, and misleading evidence, without treating answer correctness as ground truth. Retained indicators show partial, source-dependent convergence. Two composite indices consistently distinguish evidence conditions and, in Qwen3.5-27B, decrease with evidence concentration and sampling round. Higher measured uncertainty is associated with sampling beyond the model's currently preferred cue. These findings provide a measurement basis for investigating how internal uncertainty relates to information acquisition and action.

AI USE STATEMENT

Generative AI tools were used to critique the research ideas and experimental designs developed by the authors, suggest statistical tests and effect-size measures, and generate code for synthetic dataset generation, method implementation, data processing, analysis and visualization. The authors developed the core research idea, evidence-condition design, and candidate uncertainty measures. Synthetic datasets were produced by randomly generating candidate samples and filtering them according to two evaluation metrics defined by the authors; the generation and filtering procedures were implemented in AI-written code. AI-generated code was debugged jointly by the authors and AI tools, executed by the authors, and checked through inspection of visual outputs — including summary statistics and visualizations of the synthetic datasets — for anomalies. AI tools also helped organise, translate, and polish author-written points describing the results, and recommended potentially relevant literature. The authors manually reviewed conceptual suggestions and manuscript revisions, independently checked each recommended reference consulted for the paper, and iteratively reviewed and revised the figures. Code execution and visual inspection do not constitute exhaustive verification. The authors take responsibility for the final text, methods, analyses, results, and research artifacts.

REFERENCES

Amos Azaria and Tom Mitchell. The internal state of an llm knows when it's lying, 2023. URL https://arxiv.org/abs/2304.13734.

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.

Shangbin Feng, Weijia Shi, Yike Wang, Wenxuan Ding, Vidhisha Balachandran, and Yulia Tsvetkov. Don't hallucinate, abstain: Identifying LLM knowledge gaps via multi-LLM collaboration. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14664–14690, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.786. URL https://aclanthology.org/2024.acl-long.786/.

Gonçalo Guiomar, Elia Torre, Pehuen Moure, Victoria Shavina, Mario Giulianelli, Shih-Chii Liu, and Valerio Mante. Reasoning aligns language models to human inference. In *Mechanistic Interpretability Workshop at ICML 2026*, 2026. URL https://openreview.net/forum?id=SszVEekJTW.

Wes Gurnee, Nicholas Sofroniew, Adam Pearce, Mateusz Piotrowski, Isaac Kauvar, Runjin Chen, Anna Soligo, Paul Bogdan, Euan Ong, Rowan Wang, Ben Thompson, David Abrahams, Subhash Kantamneni, Emmanuel Ameisen, Joshua Batson, and Jack Lindsey. Verbalizable representations form a global workspace in language models, 2026. URL https://arxiv.org/abs/2607.15495.

Eyke Hüllermeier and Willem Waegeman. Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods. *Machine Learning*, 110(3):457–506, March 2021. ISSN 1573-0565. doi: 10.1007/s10994-021-05946-3. URL http://dx.doi.org/10.1007/s10994-021-05946-3.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. Language models (mostly) know what they know, 2022. URL https://arxiv.org/abs/2207.05221.

Amita Kamath, Robin Jia, and Percy Liang. Selective question answering under domain shift. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (eds.), *Proceedings of the 58th*

Annual Meeting of the Association for Computational Linguistics, pp. 5684–5696, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.503. URL https://aclanthology.org/2020.acl-main.503/.

Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision?, 2017. URL https://arxiv.org/abs/1703.04977.

Roozbeh Kiani and Michael N. Shadlen. Representation of confidence associated with a decision by neurons in the parietal cortex. *Science*, 324(5928):759–764, 2009. doi: 10.1126/science.1169405.

Michael Kirchhof, Gjergji Kasneci, and Enkelejda Kasneci. Position: Uncertainty quantification needs reassessment for large-language model agents, 2025. URL https://arxiv.org/abs/2505.22655.

Armen Der Kiureghian and Ove Ditlevsen. Aleatory or epistemic? does it matter? *Structural Safety*, 31(2):105–112, 2009. ISSN 0167-4730. doi: https://doi.org/10.1016/j.strusafe.2008.06.020. URL https://www.sciencedirect.com/science/article/pii/S0167473008000556. Risk Acceptance and Risk Communication.

Jannik Kossen, Jiatong Han, Muhammed Razzak, Lisa Schut, Shreshth Malik, and Yarin Gal. Semantic entropy probes: Robust and cheap hallucination detection in llms, 2024. URL https://arxiv.org/abs/2406.15927.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation, 2023. URL https://arxiv.org/abs/2302.09664.

Dharshan Kumaran, Nathaniel Daw, Simon Osindero, Petar Veličković, and Viorica Patraucean. Causal evidence that language models use confidence to drive behaviour. *Nature Machine Intelligence*, 2026. doi: 10.1038/s42256-026-01293-x. URL https://doi.org/10.1038/s42256-026-01293-x.

Stephanie Lin, Jacob Hilton, and Owain Evans. Teaching models to express their uncertainty in words, 2022. URL https://arxiv.org/abs/2205.14334.

Linyu Liu, Yu Pan, Xiaocheng Li, and Guanting Chen. Uncertainty estimation and quantification for llms: A simple supervised approach, 2024. URL https://arxiv.org/abs/2404.15993.

Andrey Malinin and Mark Gales. Uncertainty estimation in autoregressive structured prediction, 2021. URL https://arxiv.org/abs/2002.07650.

Zhiting Mei, Christina Zhang, Tenny Yin, Justin Lidard, Ola Shorinwa, and Anirudha Majumdar. Reasoning about uncertainty: Do reasoning models know when they don't know?, 2025. URL https://arxiv.org/abs/2506.18183.

Het Patel, Tiejin Chen, Hua Wei, Evangelos E. Papalexakis, and Jia Chen. Are llm uncertainty and correctness encoded by the same features? a functional dissociation via sparse autoencoders, 2026. URL https://arxiv.org/abs/2604.19974.

Freddie Bickford Smith, Jannik Kossen, Eleanor Trollope, Mark van der Wilk, Adam Foster, and Tom Rainforth. Rethinking aleatoric and epistemic uncertainty, 2025. URL https://arxiv.org/abs/2412.20892.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D. Manning. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback, 2023. URL https://arxiv.org/abs/2305.14975.

Bingbing Wen, Jihan Yao, Shangbin Feng, Chenjun Xu, Yulia Tsvetkov, Bill Howe, and Lucy Lu Wang. Know your limits: A survey of abstention in large language models. *Transactions of the Association for Computational Linguistics*, 13:529–556, 2025. doi: 10.1162/tacl_a_00754. URL https://aclanthology.org/2025.tacl-1.26/.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms, 2024. URL https://arxiv.org/abs/2306.13063.

Dongkeun Yoon, Seungone Kim, Sohee Yang, Sunkyoung Kim, Soyeon Kim, Yongil Kim, Eunbi Choi, Yireun Kim, and Minjoon Seo. Reasoning models better express their confidence, 2025. URL https://arxiv.org/abs/2505.14489.

A IDEAL BAYESIAN OBSERVER

This appendix derives the ideal-observer posterior update used to define M1 and M2 in Section 3.2, and shows that an equivalent count-based formula (Eq. 12–13) can be used for efficient computation.

Prior and observation model. The ideal observer considers four hypotheses, $Z = k$ for $k \in \mathcal{C} = \{A, B, C, D\}$, each specifying which cue is biased. It assigns a uniform prior, $P(Z = k) = 1/4$. Given the selected cue X_t and its observed outcome Y_t, the likelihood under hypothesis $Z = k$ is

$$\ell_t(k) = P(Y_t \mid X_t, Z = k) = \begin{cases} p, & X_t = k,\ Y_t = \text{RED}, \\ 1 - p, & X_t = k,\ Y_t = \text{GREEN}, \\ 1/2, & X_t \neq k, \end{cases} \tag{5}$$

where $p = 0.9$. Outcomes are generated independently across rounds conditional on the selected cues and the biased cue.

Cue availability and sampling choices. The observer uses the selected cues and their observed outcomes, without treating cue availability or sampling choices as additional evidence about Z. To justify this, let $D_t = (\mathcal{A}_s, X_s, Y_s)_{s=1}^t$ denote the complete interaction history, including the available cue sets. Its likelihood factorises as

$$P(D_t \mid Z = k) = \prod_{s=1}^{t} P(\mathcal{A}_s \mid D_{s-1}, Z = k) P(X_s \mid \mathcal{A}_s, D_{s-1}, Z = k) P(Y_s \mid X_s, Z = k). \tag{6}$$

In the generated trials, availability is generated independently of the biased-cue identity, and sampling policies depend only on the available cues, previous observations, and state-independent randomisation. Consequently, for a fixed observed history, the availability and sampling-policy factors are identical across hypotheses and cancel when normalising the posterior. Availability therefore affects which evidence is collected, but does not contribute an additional likelihood term to the posterior update. This argument requires sampling policies to have no access to the true biased-cue identity or unobserved outcomes.

Sequential posterior update. Writing h_t for the sequence of selected cues and observed outcomes up to round t, the posterior is

$$p_{t,k} = P(Z = k \mid h_t) = \frac{P(Z = k) \prod_{s=1}^{t} \ell_s(k)}{\sum_{j \in \mathcal{C}} P(Z = j) \prod_{s=1}^{t} \ell_s(j)}. \tag{7}$$

Equivalently, it can be computed recursively:

$$p_{0,k} = \frac{1}{4}, \qquad p_{t,k} = \frac{p_{t-1,k} \ell_t(k)}{\sum_{j \in \mathcal{C}} p_{t-1,j} \ell_t(j)}. \tag{8}$$

For numerical stability, the sequential implementation accumulates log-likelihoods. Defining

$$a_{t,k} = \ln P(Z = k) + \sum_{s=1}^{t} \ln \ell_s(k), \tag{9}$$

the posterior is obtained as

$$p_{t,k} = \frac{\exp(a_{t,k} - m_t)}{\sum_{j \in \mathcal{C}} \exp(a_{t,j} - m_t)}, \qquad m_t = \max_{j \in \mathcal{C}} a_{t,j}. \tag{10}$$

Each posterior vector incorporates the outcome observed at the corresponding round.

Equivalent count-based computation. The same posterior can be computed from cue-specific outcome counts. Let $r_{t,k}$ and $g_{t,k}$ denote the numbers of red and green outcomes observed from cue k up to round t. Relative to a reference model in which all cues are fair, the log-likelihood ratio for hypothesis $Z = k$ is

$$S_{t,k} = r_{t,k} \ln \frac{p}{1/2} + g_{t,k} \ln \frac{1-p}{1/2}. \tag{11}$$

The reference likelihood, $(1/2)^t$, is common to all hypotheses. Cancelling this factor and using the uniform prior gives

$$p_{t,k} = \frac{\exp(S_{t,k})}{\sum_{j \in \mathcal{C}} \exp(S_{t,j})}. \tag{12}$$

For $p = 0.9$, the score is

$$S_{t,k} = r_{t,k} \ln 1.8 + g_{t,k} \ln 0.2. \tag{13}$$

An unsampled cue has score zero. Thus, the sequential and count-based formulations yield identical posteriors when applied to the same valid cue–outcome observations.

B Validation Dataset Construction

Trial generation. Each simulated trial comprises 15 sampling rounds, followed by a separate final judgement. The outcome model uses a biased-cue red probability of $\theta = 0.9$ and a red probability of 0.5 for each unbiased cue. At each sampling round, each cue is independently made available with probability 0.5. If no cue is available, one is selected uniformly at random and made available. A uniform sampling policy then selects a cue from the available set, and an outcome is drawn according to the selected cue's identity relative to the biased cue. The available set, selected cue, and observed outcome are recorded at every round.

Candidate generation and stopping rule. A master pseudorandom number generator, initialised with seed 0, supplies a separate seed for each trial. Trials are generated in batches of 20,000. After each batch, we compute the ideal posterior after the fifteenth observation and assign qualifying histories to the three conditions defined in Section **??**. Histories satisfying none of these conditions are excluded from the candidate sets.

Generation stops once every condition contains at least 300 candidates, corresponding to three times the target evaluation sample of 100 histories per condition. This criterion was met after the third batch, yielding 60,000 generated trials. The AMBIGUOUS condition was the limiting condition. Table 2 reports the resulting candidate counts.

Table 2: Candidate histories and final evaluation samples. Percentages are relative to all 60,000 generated trials.

Condition	Candidates	Percentage	Selected
VERIDICAL	12,190	20.317%	100
AMBIGUOUS	340	0.567%	100
MISLEADING	395	0.658%	100

Balanced sampling. Using a separate pseudorandom number generator initialised with seed 1, we draw a simple random sample of 100 histories without replacement from each condition's candidate set. The selected trials are regenerated from their stored seeds to recover their complete histories. This produces a fixed, balanced evaluation dataset of 300 histories.

Presentation to the evaluated model. For each selected trial, the complete history after the fifteenth sampling round is presented as a single text prompt. It includes the available cues, the selected cue and observed outcome at each round, and the trial and round indices together with the total number of sampling rounds. The evaluated model does not select cues or alter outcomes in these retrospective histories. The same history is used across all measurement procedures, with a measurement-specific question or instruction appended where required. All indicator-screening comparisons use this final observation point.

B.1 PROMPTS

Input structure. For the fixed validation dataset described in Appendix B, each model input consists of a single user message concatenating the task instructions, 15 feedback lines describing the pre-generated sampling history, and a measurement-specific elicitation question. Each feedback line specifies the available cues, the selected cue, and the observed outcome. Although the task instructions describe an interactive sampling game, the evaluated model receives the history retrospectively and does not make the sampling choices. The wording "You chose" attributes the pre-generated choices to the agent within the task narrative.

The same 300 trial histories are used across models and measurement procedures. The task instructions differ between models as detailed below, and the appended elicitation question depends on the measurement procedure. The example in this section uses the baseline final-choice question.

Reward instructions. An unintended prompt-version mismatch resulted in different reward instructions across models. GPT used pv10:

```
Correct: +100 points, Wrong: -100 points.
```

Qwen used pv11:

```
Correct: +100 points, Wrong: 0 points.
Try to get as many points as possible. Your current accumulated score is
 0 points.
```

The task instructions are otherwise identical. Thus, the versions differ in the penalty for an incorrect answer and in the presence of an explicit score-maximisation instruction and an initial-score statement. The score statement in pv11 is fixed at zero in these validation prompts. For a given trial and measurement procedure, the history and elicitation question are unchanged across versions. Because model identity and reward instructions were not independently varied, between-model differences cannot be attributed to model identity alone.

History format. Each sampling round is represented by the following feedback template, where braces indicate placeholders:

```
Trial 1, Round {t}: Available cues {available}. You chose "{choice}" and
 saw {result}.
```

Here, {`t`} ranges from 1 to 15, {`available`} lists the available cues, {`choice`} is the selected cue, and {`result`} is either `RED` or `GREEN`.

Baseline final-choice question. For the baseline final judgement, the history is followed by a blank line and the following question:

```
Pause. The game ends now.
Make the final decision.
Which cue ['A', 'B', 'C', 'D'] would you choose?
Respond with exactly one letter: A, B, C, or D.
```

Complete example. The following is the complete pv11 input for trial `D_0000`, which belongs to the MISLEADING condition. The true biased cue is D, whereas the ideal posterior is concentrated on A, giving $\mathrm{M1}_{15} \approx 0.058$ and $\mathrm{M2}_{15} \approx 0.811$. The true cue, condition label, and evidence measures are metadata and are not included in the model input.

```
Task:
You will play a game with 15 rounds.
In each round you'll see available cues (choosable):
One cue has 90% one color / 10% other, others have 50/50 color
distribution.
```

```
Possible cues are: A, B, C, D.
Available cues disappear after random duration, at least one cue is
active per round.
Your task is to pick one of the available cues every round, by
responding with just the letter and nothing else. Don't use markup or
punctuation.
After 15 rounds, identify the biased cue.
Correct: +100 points, Wrong: 0 points.
Try to get as many points as possible. Your current accumulated score is
 0 points.
Trial 1, Round 1: Available cues A. You chose "A" and saw RED.
Trial 1, Round 2: Available cues A, B, C. You chose "B" and saw GREEN.
Trial 1, Round 3: Available cues B. You chose "B" and saw GREEN.
Trial 1, Round 4: Available cues B, C. You chose "C" and saw GREEN.
Trial 1, Round 5: Available cues B, D. You chose "D" and saw RED.
Trial 1, Round 6: Available cues C, D. You chose "D" and saw GREEN.
Trial 1, Round 7: Available cues A, B, D. You chose "B" and saw GREEN.
Trial 1, Round 8: Available cues C. You chose "C" and saw RED.
Trial 1, Round 9: Available cues C. You chose "C" and saw GREEN.
Trial 1, Round 10: Available cues B. You chose "B" and saw RED.
Trial 1, Round 11: Available cues A, C, D. You chose "A" and saw RED.
Trial 1, Round 12: Available cues A. You chose "A" and saw RED.
Trial 1, Round 13: Available cues A, B, C. You chose "B" and saw GREEN.
Trial 1, Round 14: Available cues B, D. You chose "D" and saw RED.
Trial 1, Round 15: Available cues A. You chose "A" and saw RED.

Pause. The game ends now.
Make the final decision.
Which cue ['A', 'B', 'C', 'D'] would you choose?
Respond with exactly one letter: A, B, C, or D.
```

The corresponding pv10 input replaces the two lines beginning with `Correct:` and `Try` with the single line `Correct: +100 points, Wrong: -100 points.`; all remaining text is unchanged.

C Language Models and Inference Settings

C.1 Deployment

Both models were deployed on RunPod GPU instances. We used the official `openai/gpt-oss-20b` weights in their released MXFP4 format, without additional quantisation. GPT-OSS was served through an OpenAI-compatible vLLM endpoint with a maximum context length of 8,192 tokens.

We used `Qwen/Qwen3.5-27B` in BF16 without quantisation. Text generation used vLLM with tensor parallelism set to one and a maximum context length of 16,384 tokens. Logits and internal activations were extracted separately through direct BF16 forward passes using Transformers.

C.2 GPT-OSS Inference

Direct-answer setting. The direct-answer setting used the completions endpoint with a raw Harmony-formatted input prefilled to the final channel. No system turn was included, and no explicit reasoning trace was generated. We used temperature 0 and a maximum output length of one token. For answer-distribution extraction, we requested the top-20 token log-probabilities and normalised the probabilities extracted for A–D over the four alternatives.

Reasoning settings. The low and high settings used the chat-completions endpoint with the corresponding reasoning-effort setting. Reasoning outputs were generated at temperature 1.0 with a maximum output length of 4,096 tokens.

Answer distributions were extracted in a separate readout step rather than estimated from sampled final answers. The generated reasoning trace was replayed as an analysis channel, followed by a final-channel prefill. This readout used temperature 0, a maximum output length of one token, and the same answer-distribution extraction procedure as the direct-answer setting.

Repetitions. Belief probes were evaluated after the fifteenth observation. For each trial and probe, we used one readout in the direct-answer setting and ten reasoning generations in each of the low and high settings.

Opt-out experiments used the direct-answer and low settings, with one and ten repetitions, respectively, at thresholds $p^* \in \{0, 0.30, 0.45, 0.60, 1\}$. No high-setting opt-out experiments were conducted.

Truncated reasoning. Reasoning outputs terminated by the generation-length limit were retained. Their available traces were replayed to extract answer distributions using the same readout procedure. The resulting measurements therefore include readouts following both completed and truncated reasoning traces.

Randomness. No request-level sampling seed was specified. A shuffle seed of 0 controlled request submission order only and did not fix the randomness of reasoning generation.

C.3 Qwen Inference

Thinking control and logit extraction. Thinking was enabled or disabled through the model's chat template. For direct-answer logit measurements, thinking was disabled and the full-vocabulary logits were obtained through a direct forward pass without sampling. One readout was performed per trial. For measurements following reasoning, each generated trace was replayed through the model to extract the subsequent answer-token distribution.

Verbal reports without thinking. The initial forced-choice response used thinking disabled, temperature 0, and a maximum output length of one token. Subsequent verbal-report elicitation also disabled thinking and used temperature 0.7, top-p 0.8, top-k 20, a presence penalty of 1.5, and a maximum output length of 64 tokens. Ten report samples were collected per trial for each elicitation.

Thinking and subsequent reports. Thinking traces were generated at temperature 1.0, top-p 0.95, top-k 20, and a presence penalty of 1.5, with a maximum output length of 8,192 tokens. Ten traces were generated per trial for measurements using repeated reasoning samples.

Subsequent verbal-report questions disabled thinking and used temperature 0.7, top-p 0.8, top-k 20, a presence penalty of 1.5, and a maximum output length of 128 tokens. For logit-based measurements, one answer-distribution readout was obtained for each generated trace.

Opt-out experiments. Thinking-enabled opt-out experiments used temperature 1.0, top-p 0.95, top-k 20, and a presence penalty of 1.5. One repetition was collected per trial at each sure-reward value in $\{0, 45, 60, 100\}$ points.

Randomness. The vLLM server seed was set to 0. No request-level sampling seeds were specified. Direct forward-pass measurements did not involve token sampling.

C.4 Prompt-Version Coverage

Table 3 summarises the prompt versions available for each measurement family. GPT pv10 runs cover all five families. Additional pv11 runs cover logit-based measures, opt-out behaviour, and reasoning-trace analysis. Qwen runs use pv11. The exact reward instructions and an example input are provided in Appendix B.1.

All measurement families use the same fixed trial histories. However, comparisons using different prompt versions cannot isolate model effects from differences in reward instructions. Comparisons

Table 3: Prompt-version coverage by model and measurement family.

Measurement family	GPT-OSS	Qwen
Verbal reports	pv10	pv11
Logit-based measures	pv10, pv11	pv11
Opt-out behaviour	pv10, pv11	pv11
Reasoning-trace analysis	pv10, pv11	pv11
Residual-stream analysis	pv10	pv11

Table 4: Data collected per model. *Rounds* lists the sampling rounds after which a measure was collected; all counts and all analyses refer to round 15, the final round of each of the 300 trials (100 per evidence condition). A record is one analysis unit, i.e. one trial × repetition (× payoff level) after de-duplication, not an API call. Validity criteria differ by measure and are listed below the table. Prompt versions are described in Appendix B.1.

Measure	Reasoning	Prompt	Rounds	Reps / trial	Valid / total records or coverage
Qwen3.5-27B					
Answer distribution	none	pv11, Q4	1–15	1 (determ.)	300 / 300
Verbal report	none	pv11	1–15	10	3,000 / 3,000
Reasoning trace	thinking	pv11	1–15	10	2,972 / 3,000[a]
Answer distribution	thinking	pv11, Q4	1–15	10	2,972 / 2,972[b]
Verbal report	thinking	pv11	1–15	10	2,971 / 2,972[b]
Opt-out (4 payoffs)	none	pv11	15	1 per payoff	1,200 / 1,200
Opt-out (4 payoffs)	thinking	pv11	1–15	1 per payoff	1,199 / 1,200
Residual activations	none	pv11	1–15	1 (determ.)	64 layers × 3 positions[c]
GPT-OSS-20B					
Answer distribution	none	pv11, Q4	15	1 (determ.)	300 / 300
Answer distribution	low	pv11, Q4	15	10	3,000 / 3,000
Answer distribution	high	pv11, Q4	15	10	3,000 / 3,000
Reasoning trace	low	pv11	15	10	3,000 / 3,000
Reasoning trace	high	pv11	15	10	2,570 / 3,000
Verbal report	none	pv10	15	1	284 / 300
Verbal report	low	pv10	15	1	297 / 297
Opt-out (5 payoffs)	none	pv11	15	1 per payoff	1,500 / 1,500
Opt-out (5 payoffs)	low	pv11	15	10 per payoff	15,000 / 15,000
Residual activations	none	pv10	1–15	1 (determ.)	24 layers × 3 positions[c]
Jacobian-lens readout	none	pv10	1–15	1 (determ.)	23 layers × 2 positions[c]

Validity criteria. *Answer distribution, no reasoning:* probabilities for A–D are returned by a single forward pass. *Answer distribution, reasoning:* for Qwen3.5-27B, each valid reasoning trace is re-scored; for GPT-OSS-20B, the final channel contains an A–D answer whose token distribution is read, and generations that reached the token limit after answering are retained (122 of 3,000 at high effort). *Reasoning trace:* for Qwen3.5-27B, the generation stopped normally with a thinking block; for GPT-OSS-20B, an answer is present and the reasoning budget was not exhausted. *Verbal report:* a value between 0 and 100 can be parsed from the reply (GPT-OSS-20B without reasoning: 14 follow-ups exhausted the reasoning budget and 2 were unparseable). *Opt-out:* an answer among A–E is obtained (Qwen3.5-27B thinking: parsed letter; otherwise the argmax of the A–E distribution).
[a]One planned generation per trial × repetition. Three generations failed and were regenerated under the same key; only the regenerated output is kept, so each pair contributes one record. Of the 28 remaining invalid records, 27 requests returned no output and 1 reached the 8,192-token limit. [b]Collected on the valid reasoning traces, so the total equals the number of valid traces. [c]Coverage per trial and round from one forward pass over the trial transcript (all 300 trials), not a record count.

using pv11 for both models share the task instructions but retain the model-specific inference settings described above.

C.5 Data Collection Overview

D Measurement Details

D.1 Verbal Reports

Experimental conditions and reporting scale. For both GPT and Qwen, we conduct verbal-report experiments under non-thinking and thinking conditions. After the model provides a cue choice, a follow-up question requests an integer uncertainty rating from 0 to 100. The endpoints are described as "completely certain" and "a complete guess, no better than random", respectively. The rating concerns the model's preceding answer, with higher values indicating greater reported uncertainty. Measurements use the history after the fifteenth sampling round.

GPT collection and scoring. GPT verbal-report experiments use pv10 and include the non-thinking and low-effort thinking conditions. The initial response is reused from the corresponding previously generated record with repetition index 0. We collect one report per evaluated trial and condition, using temperature 1.0 and a maximum output length of 2,048 tokens for report generation.

A numerical rating is successfully parsed for 284 of 300 trials in the non-thinking condition and all 297 evaluated trials in the thinking condition. Successfully parsed ratings provide the trial-level scores; records without a parsed numerical rating are treated as missing.

For the eight-indicator analysis, the rating from the low-effort thinking condition is additionally converted to a binary indicator:

$$U_{\text{verbal,bin}} = \mathbb{I}[U_{\text{verbal}} \geq 15]\,. \tag{14}$$

Qwen collection and scoring. Qwen verbal-report experiments use pv11 and include both non-thinking and thinking conditions, with ten repetitions per trial and condition. Report generation uses temperature 0.7, top-p 0.8, top-k 20, and a presence penalty of 1.5. Maximum report-generation lengths are 64 tokens in the non-thinking condition and 128 tokens in the thinking condition.

For each trial and condition, the score is the arithmetic mean of successfully parsed numerical ratings:

$$U_{\text{verbal}} = \frac{1}{N_{\text{valid}}} \sum_{r \in \mathcal{V}} u^{(r)}, \tag{15}$$

where $\mathcal{V}$ contains the valid repetitions and $R_{\text{valid}} = |\mathcal{V}|$. Unlike the entropy of the averaged distribution, this quantity measures dispersion within individual repetitions without incorporating disagreement between their answer distributions. We compare both aggregation rules in the indicator evaluation.

Cross-model interpretation. The models share the reporting scale but differ in prompt version, repetition count, and trial-level scoring. In particular, the GPT eight-indicator analysis uses a thresholded thinking-condition report, whereas Qwen uses mean numerical ratings. These derived scores are not treated as interchangeable measurements on a common numerical scale.

D.2 Opt-Out Behaviour

Task variant. We augment the cue-identification question with a fifth option, E, which provides a guaranteed reward R_{out}. Selecting A–D yields R_{correct} if the selected cue is the biased cue and R_{wrong} otherwise. The supplied sampling history is unchanged; the opt-out question replaces the belief question:

```
Trial 1: The game stops here, at round 15 of 15, and this is the last
decision of the trial.
Answering A, B, C or D names the cue you think had the highest ratio of
RED: {right_points} points if it is the right cue, {wrong_points} points
 if it is not. Answering E is worth {sure_points} points whichever cue
it turns out to be.
Respond with exactly one letter: A, B, C, D or E.
```

where {right_points}, {wrong_points} and {sure_points} are R_{correct}, R_{wrong} and R_{out}.

Let c denote the probability that the selected cue is correct. The offered rewards define the threshold

$$p^* = \frac{R_{\text{out}} - R_{\text{wrong}}}{R_{\text{correct}} - R_{\text{wrong}}}. \tag{16}$$

Answering has greater expected reward when $c > p^*$, whereas opting out has greater expected reward when $c < p^*$. This threshold characterises the offered rewards without assuming that the model follows an expected-reward-maximising policy.

Opt-out response. For repetition r, we record

$$O^{(r)} = \mathbb{I}\left[\arg\max_{k\in\{A,B,C,D,E\}} s_k^{(r)} = E\right], \tag{17}$$

where $s_k^{(r)}$ is the extracted score for option k in repetition r. For Qwen in the thinking condition, $O^{(r)}$ is instead determined by the answer letter the model generates after its reasoning trace. Different sampled reasoning traces can produce different responses across repetitions. We use the opt-out response as a behavioural indicator of uncertainty.

D.3 Answer-Option Entropy

Extraction point. We extract next-token probabilities at the position where the model is prompted to answer with a cue letter. In the direct-answer condition, the readout follows the supplied trial history and answer instruction without an explicit reasoning trace. In reasoning conditions, we replay the generated reasoning trace and extract the subsequent answer-token distribution. The baseline answer instruction is reproduced in Appendix B.1.

GPT token probabilities. We request the top-20 next-token log-probabilities from vLLM. Each returned token string is stripped of leading and trailing whitespace and retained only if the resulting string is exactly A, B, C, or D. Lowercase letters and strings containing additional punctuation are excluded. If multiple token strings map to the same letter, we retain their maximum probability rather than summing them.

Let s_k be the retained probability for cue k. If no matching token appears among the top-20 candidates, we set $s_k = 0$. We compute the retained option mass and normalised probabilities as

$$m = \sum_{k\in\mathcal{C}} s_k, \qquad q_k = \frac{s_k}{m}. \tag{18}$$

Records with $m = 0$ are marked invalid and excluded from aggregation. In the direct-answer runs, the observed retained mass has a minimum of 0.975 and a median of 0.995, indicating that the retained answer options account for most of the next-token probability mass in this setting.

Qwen token probabilities. For the main logit-based measurement, we obtain the full-vocabulary logits through a direct Transformers forward pass and apply a softmax over the vocabulary. For each cue, we retrieve the token IDs corresponding to its uppercase letter and its leading-space form. We retain the larger of these two probabilities and normalise the resulting scores over A–D. Unlike GPT's top-20 readout, this procedure uses the full-vocabulary distribution.

Repetition-level normalisation and aggregation. For each trial, elicitation prompt, and reasoning condition, we normalise the option probabilities separately within each valid repetition and compute its normalised entropy

$$U_{\text{ans}}^{(r)} = -\frac{1}{\ln 4}\sum_{k\in\mathcal{C}} q_k^{(r)} \ln q_k^{(r)}, \tag{19}$$

using the convention $0 \ln 0 = 0$. It ranges from zero for a point mass to one for a uniform distribution over the four cues. With R valid repetitions, the trial-level score is $U_{\text{ans}} = \frac{1}{R}\sum_{r=1}^{R} U_{\text{ans}}^{(r)}$. When $R = 1$, this reduces to the normalised entropy of a single answer distribution.

D.4 Reasoning-Trace Measures

Reasoning length. For GPT, we measure reasoning length using the model tokenizer, counting tokens generated before the transition to the final channel. The count excludes the final answer but includes a small number of channel-formatting tokens. For Qwen, we measure the character count of the extracted thinking text, excluding the answer following the closing thinking tag. The two models therefore use different length units, and their absolute lengths are not directly comparable.

Uncertainty lexicon. Both models use a shared lexicon of 36 uncertainty-related words and phrases:

> ambiguous; small sample; uncertain; not sure; not certain; unclear; not confident; no clear; not obvious; hard to tell; hard to say; can't be certain; cannot be certain; can't be sure; cannot be sure; not definitive; cannot definitively; inconclusive; coin flip; could be wrong; might be wrong; speculat*; tricky; random guess; questionable; could be either; can't tell; cannot tell; insufficient data; no strong; leaning; lean toward; lean towards; slightly favor; hard to differentiate; arbitrar*.

D.5 Hidden-State Extraction

Input and forward pass. For each trial, we provide the task instructions followed by all 15 sampling-feedback lines. No elicitation question, reasoning trace, or answer is appended. A single forward pass produces the hidden states used to extract representations for all rounds.

For GPT, the text is prefixed with the Harmony user header shown below, without a system message or a closing message delimiter:

```
<|start|>user<|message|>
```

For Qwen, the task instructions and history are supplied as plain text without applying a chat template.

Layers. We obtain representations from the hidden-state tuple returned by Hugging Face Transformers. GPT extraction uses indices 1–24. Qwen extraction uses indices 1–64. We retain the returned tensors without attempting to remove any final normalisation. The final tuple element is therefore used as returned by the model implementation rather than assumed to be an unnormalised residual-stream state.

Token positions. We locate three positions in each feedback line by mapping character locations to token indices. For example:

```
Trial 1, Round t: Available cues A, B. You chose "B" and saw RED.
```

The *result* position corresponds to the observed colour, `RED` or `GREEN`. The *choice* position corresponds to the selected cue letter inside quotation marks, here `B`. The *boundary* position is the final token of the feedback line, corresponding to its terminating newline. These positions are identified separately for each model's tokenisation. In the causal forward pass, the boundary representation has access to the completed current-round record but not to subsequent rounds.

D.6 Supervised Probes

Input representations. We train separate probes for each layer–position combination using the *result*, *choice*, and *boundary* positions from the fifteenth sampling round. Each trial contributes one representation per combination. GPT uses layers 1–24 and Qwen uses layers 1–64. Representations are extracted before any elicitation question, reasoning trace, or answer. GPT activation inputs use task wording identical to pv10; Qwen activation inputs use pv11.

Prediction targets. We consider four families of uncertainty-related targets:

- **Uncertainty-language density.** We sum lexicon-occurrence counts across valid reasoning traces, divide by their total character count, and multiply by 1,000. GPT uses ten attempted low-effort generations per trial under pv11. Qwen uses ten attempted thinking-enabled generations per trial under pv11.
- **Uncertainty-language presence or rate.** GPT uses separate binary targets for low- and high-effort reasoning under pv11. A target is positive if any valid generation contains a lexicon match. Low-effort and high-effort targets both use ten attempted generations per trial, across all 300 trials. Qwen uses the fraction of valid thinking-enabled generations containing a lexicon match. These targets use substring matching against the shared lexicon, without additional regular-expression patterns.
- **Verbal uncertainty.** The target is the uncertainty report in the non-thinking condition. GPT uses one report per trial under pv10; Qwen uses the mean of ten reports under pv11.
- **Post-reasoning answer entropy.** We average the normalised A–D answer distributions across valid repetitions and compute the normalised entropy of the resulting distribution:

$$\bar{q}_k = \frac{1}{R_{\text{valid}}} \sum_{r \in \mathcal{V}} q_k^{(r)}, \qquad y_{\text{ent}} = -\frac{1}{\ln 4} \sum_{k \in \mathcal{C}} \bar{q}_k \ln \bar{q}_k. \tag{20}$$

  Here, $\mathcal{V}$ is the set of valid repetitions and $R_{\text{valid}} = |\mathcal{V}|$. GPT uses low-effort reasoning and Qwen uses thinking-enabled generation, both under pv11 with ten attempted repetitions.

Entropy elicitation. For the entropy target, both models receive the following question after the 15 feedback lines, separated from the history by a blank line:

```
Pause. The game ends now.
Make the final decision.
Trial 1: Based on all observed colors, which cue ['A', 'B', 'C', 'D'] do
 you think had the highest ratio of RED?
Respond with exactly one letter: A, B, C, or D.
```

Valid observations and sample sizes. A reasoning generation is valid if it contains an answer and does not exhaust the reasoning budget. Trials without a valid target are excluded separately for each prediction task, without imputation. Qwen reasoning-derived targets use between four and ten valid repetitions per trial. Table 5 reports the resulting sample sizes.

Table 5: Available trials for each probe target. V, A, and M denote VERIDICAL, AMBIGUOUS, and MISLEADING conditions.

Model	Target	Total	V	A	M
GPT	Density	298	100	100	98
GPT	Presence, low	298	100	100	98
GPT	Presence, high	287	100	87	100
GPT	Verbal uncertainty	284	99	89	96
GPT	Answer entropy	300	100	100	100
Qwen	Density	300	100	100	100
Qwen	Language rate	300	100	100	100
Qwen	Verbal uncertainty	300	100	100	100
Qwen	Answer entropy	300	100	100	100

The GPT low-effort presence target contains 66 positive and 232 negative trials; the high-effort target contains 158 positive and 129 negative trials. The 13 trials excluded from the high-effort target all belong to the AMBIGUOUS condition and lack a valid answer after reaching the reasoning limit. Of the 16 trials excluded from the GPT verbal target, 11 belong to the AMBIGUOUS condition. Evaluations therefore describe the subset with observable labels; exclusions are not balanced across evidence conditions.

Probe models and preprocessing. Binary targets use a support vector classifier with a radial basis function kernel, $C = 1$, and balanced class weights. Continuous targets use support vector regression with the same kernel, $C = 1$, and $\epsilon = 0.1$. Both use the scikit-learn `scale` setting for the kernel coefficient. Hyperparameters are fixed rather than tuned.

Within each training fold, we fit feature standardisation followed by PCA with ten components and apply these transformations to the held-out fold. Regression targets are standardised using training-fold statistics, and predictions are transformed back to the original units.

Cross-validation and layer selection. We use five-fold stratified cross-validation for classification and five-fold cross-validation for regression, with shuffling and random seed 0. Classification folds are stratified by the prediction label. Each trial appears in exactly one held-out fold.

We pool out-of-fold predictions to calculate regression Pearson correlation and R^2, or classification accuracy, balanced accuracy, and ROC AUC. Within each position, the selected layer maximises overall out-of-fold Pearson correlation for regression or balanced accuracy for classification. Because selection and performance reporting use the same cross-validation results, selected-layer scores are exploratory rather than independent test-set estimates.

Condition-preserving permutation tests. For each layer–position combination, we perform 200 permutations, shuffling labels within each evidence condition. The original fold assignments remain fixed. Each permutation refits the probe using the shuffled labels; fold-specific feature standardisation and PCA are reused because they do not depend on the targets.

Classification tests use balanced accuracy. Regression tests use overall Pearson correlation and the mean of the three within-condition correlations. For an observed statistic s_{obs}, we calculate

$$p_{\text{perm}} = \frac{1 + \sum_{b=1}^{200} \mathbb{I}[s_b \geq s_{\text{obs}}]}{201}. \tag{21}$$

The minimum attainable value is $1/201 \approx 0.005$. For the within-condition statistic, undefined permutation correlations do not contribute to the exceedance count, while the denominator remains 201.

Permutation tests are conducted separately for each layer–position combination without reselecting the best layer. Their resulting p values do not adjust for layer selection or multiple comparisons across layers and positions.

D.7 Jacobian Lens

Lens estimation. We fit a Jacobian lens for GPT using Anthropic's open-source implementation.[1] For each source layer l, the method estimates an average Jacobian J_l mapping intermediate representations into the final-layer representation space. The projected representation is decoded using the model's own unembedding.

We construct the fitting corpus from the training split of WikiText-2 raw. We retain passages containing at least 400 characters, shuffle them with random seed 0, and select 100 passages. Each input is limited to 128 tokens and begins with a BOS token. The model is dequantised to BF16 for fitting. The resulting lens contains 23 matrices of size 2880×2880, stored in FP16, for source layers 0–22 and target layer 23. These indices follow the lens implementation's zero-based layer convention.

Task inputs and extraction positions. Readout inputs contain the task instructions and sampling-feedback lines, with wording identical to pv10. We prepend a BOS token without applying the Harmony user wrapper and append no elicitation question. We analyse the result and boundary positions of the fifteenth sampling round across all 23 fitted source layers.

Option distribution and concentration. For a representation $h_{l,s}$ at layer l and position s, we obtain vocabulary logits by applying the lens and the model's unembedding:

$$z_{l,s} = \text{Unembed}(J_l h_{l,s}). \tag{22}$$

[1] `https://github.com/anthropics/jacobian-lens`

We retain only the logits for the uppercase tokens A, B, C, and D, corresponding to token IDs 32–35. Leading-space variants are not included. The projected option distribution is

$$q_{l,s,k} = \frac{\exp(z_{l,s,k})}{\sum_{j \in \mathcal{C}} \exp(z_{l,s,j})}. \tag{23}$$

We calculate its concentration as

$$C_{\mathrm{JL}} = 1 - \frac{H(q_{l,s})}{\ln 4}, \qquad H(q_{l,s}) = -\sum_{k \in \mathcal{C}} q_{l,s,k} \ln q_{l,s,k}. \tag{24}$$

Higher values indicate a more concentrated projected option distribution. We examine all fitted layer–position combinations as an exploratory analysis; no single Jacobian-lens readout is included in the composite uncertainty index.

D.8 Sparse Autoencoder Features

Pretrained SAEs and inputs. We use pretrained SAEs without further training. For GPT-OSS-20B, we use the layer-15 BatchTopK SAE released by Andy Arditi, with 131,072 features, sparsity parameter 64, and input dimension 2,880.[2] For Qwen3.5-27B, we use Qwen-Scope TopK SAEs with 81,920 features, sparsity parameter 100, and input dimension 5,120.[3] Candidate Qwen SAE layers are 4, 12, 20, 28, 36, 44, 52, and 60. Qwen SAE layer L corresponds to hidden-state index $L + 1$.

Inputs are the final-round result-position representations extracted during passive processing of the task instructions and history, without an elicitation question. GPT inputs use wording identical to pv10; Qwen inputs use pv11. Each trial produces one feature vector, without averaging across rounds. GPT uses the full pretrained encoder. Qwen computes encoder scores using the encoder weights and bias, retains the top 100 entries for each trial, and sets the remaining entries to zero.

Development and confirmation sets. GPT analysis includes 297 trials with complete data for the accompanying verbal-report, opt-out, uncertainty-language, and logit-entropy measures. Qwen analysis includes all 300 trials. Within each evidence condition, we shuffle trials using seed 0 and reserve the rounded value of 20% of the condition's sample size for confirmation. This yields 238 development and 59 confirmation trials for GPT, and 240 development and 60 confirmation trials for Qwen. The split is performed once and saved. Confirmation data are not used for feature or layer selection.

Feature selection. All feature-selection statistics are computed on the current training set. For a feature with activation x, let $\bar{x}_g$ and s_g^2 denote its mean and sample variance in evidence condition g, where V, A, and M denote Veridical, Ambiguous, and Misleading. We define the standardised contrast

$$d(x) = \frac{\bar{x}_A - \frac{1}{2}(\bar{x}_V + \bar{x}_M)}{\sqrt{\frac{1}{3}(s_V^2 + s_A^2 + s_M^2)}}. \tag{25}$$

A feature is eligible only if it satisfies all three criteria:

1. Its activation is nonzero in at least 1% of training trials for GPT or 2% for Qwen.
2. Its standardised contrast satisfies $|d(x)| \geq 0.2$.
3. Its mean activation is equivalent between the Veridical and Misleading conditions under the equivalence test described below.

Eligible features are ranked by $|d(x)|$, and at most 18 are retained.

[2] https://huggingface.co/andyrdt/saes-gpt-oss-20b

[3] https://huggingface.co/Qwen/SAE-Res-Qwen3.5-27B-W80K-L0_100

Equivalence criterion. We use two one-sided tests (TOST) with an equivalence margin

$$\Delta = 0.3\sqrt{\frac{s_V^2 + s_M^2}{2}}. \tag{26}$$

The tested mean difference is $\bar{x}_V - \bar{x}_M$, with equivalence bounds $(-\Delta, \Delta)$. Standard errors and degrees of freedom use the Welch formulas. Equivalence is established when both one-sided tests reject at $\alpha = 0.05$. The same procedure is used to assess equivalence of the resulting candidate score between these two conditions.

Score construction. Let $\mathcal{F}$ be the selected feature set, and let μ_f and σ_f be the mean and standard deviation of feature f across training trials. We define the candidate score for trial i as

$$U_{\mathrm{SAE},i} = \frac{1}{|\mathcal{F}|} \sum_{f \in \mathcal{F}} \mathrm{sign}(d_f) \frac{x_{i,f} - \mu_f}{\sigma_f}. \tag{27}$$

Sign alignment makes higher scores correspond to the direction associated with the AMBIGUOUS condition. Selected features receive equal weight. Feature selection, directions, and standardisation parameters are determined exclusively from training data.

Development-set cross-validation. We use five-fold cross-validation stratified by evidence condition, repeated ten times with seed 1. For each training fold, we repeat feature selection and score construction, then apply the fitted mapping to the held-out fold. Each development trial's score is the mean of its ten out-of-fold scores.

For each candidate layer, we evaluate the standardised contrast, VERIDICAL–MISLEADING equivalence, and retention of the contrast after stratifying trials by answer correctness. All development diagnostics use out-of-fold scores.

Correctness-stratified retention. We calculate the same standardised contrast separately among correctly and incorrectly answered trials, denoted by d_{correct} and $d_{\mathrm{incorrect}}$. Relative to the contrast across all trials, d_{all}, retention is

$$\mathrm{retention} = \frac{\min(|d_{\mathrm{correct}}|, |d_{\mathrm{incorrect}}|)}{|d_{\mathrm{all}}|}. \tag{28}$$

This quantity is not computed if either correctness subset contains fewer than ten trials or lacks any evidence condition.

Correctness is based on final-round direct-answer readouts. For GPT, we take a majority vote over argmax cue choices across elicitation variants and pv10/pv11 readouts. For Qwen, we use the argmax cue from the local logit readout under pv11 and elicitation variant v4. A response is correct if the selected cue is the true biased cue. Evidence-condition membership is distinct from this model-response correctness label.

Layer selection and confirmation. For Qwen, eligible layers must pass score-level VERIDICAL–MISLEADING equivalence and achieve retention of at least 0.7 on the development set. Among eligible layers, we select the one with the largest absolute standardised contrast. GPT has a single candidate SAE layer, so no layer selection is performed.

We then repeat feature selection and score construction on the full development set at the retained layer. The feature set, directions, and standardisation parameters are frozen before application to confirmation trials. We evaluate the confirmation set once, reporting the standardised contrast, VERIDICAL–MISLEADING equivalence, and correctness-stratified retention where defined.

E SUPPLEMENTARY RESULTS

E.1 INDIVIDUAL-INDICATOR VALIDATION

Tables 6 and 7 report all candidate indicators and reasoning settings. Figures 7, 8, and 9 show their condition-wise distributions, including indicators that did not pass screening.

In the tables, n gives valid trial counts in VERIDICAL/AMBIGUOUS/MISLEADING (V/A/M) order. Effect sizes are Cohen's d using the pooled standard deviation, without small-sample bias correction. The A–V and A–M columns report effect sizes with 90% confidence intervals and one-sided Welch-test p values; V–M reports 95% intervals and two-sided Welch-test p values. Intervals use a Welch-based normal approximation. All reported p values are unadjusted. For the all-zero opt-out comparison, the implementation assigns $d = 0$, an interval of $[0, 0]$, and $p = 1$ by convention; these entries are not inferential estimates. P denotes passing both ambiguity contrasts without a detected V–M difference; FS denotes a failed ambiguity contrast; VD denotes a detected V–M difference after passing both ambiguity contrasts. An asterisk marks a retained indicator. Among qualifying variants within each method, selection maximises $\min(d_{A-V}, d_{A-M})$; probe targets compete within one method. SAE confirmation samples contain 59 GPT-OSS-20B and 60 Qwen3.5-27B trials.

HE, LE, and NR denote high effort, low effort, and no reasoning; Think denotes Qwen's thinking setting. The suffix "lbl" identifies the setting used to obtain a probe's target.

Table 6: GPT-OSS-20B: full statistics for every candidate indicator and reasoning version. Column definitions and screening rules are given in the accompanying text.

Indicator	Cond.	n	A–V	A–M	V–M	Screening
Answer entropy	HE	100/100/100	1.16 [.93,1.40] $p<.001$	1.16 [.92,1.39] $p<.001$	−.02 [−.30,.26] $p=0.908$	P
Answer entropy	LE	100/100/100	1.96 [1.72,2.19] $p<.001$	1.74 [1.51,1.98] $p<.001$	−.17 [−.45,.10] $p=0.219$	P*
Answer entropy	NR	100/100/100	.30 [.07,.54] $p=0.017$	.13 [−.11,.36] $p=0.189$	−.18 [−.46,.10] $p=0.209$	FS
CoT length	HE	100/87/100	.79 [.54,1.05] $p<.001$	.82 [.56,1.07] $p<.001$	.02 [−.26,.30] $p=0.909$	P*
CoT length	LE	100/100/98	.78 [.54,1.01] $p<.001$	.58 [.35,.82] $p<.001$	−.23 [−.51,.05] $p=0.106$	P
CoT uncertainty words (UW)	HE	100/87/100	.78 [.52,1.04] $p<.001$	.78 [.52,1.04] $p<.001$	−.01 [−.29,.26] $p=0.919$	P*
CoT uncertainty words (UW)	LE	100/100/98	.74 [.51,.98] $p<.001$	.68 [.44,.91] $p<.001$	−.11 [−.39,.17] $p=0.452$	P
Jacobian lens entropy	–	100/100/100	.23 [−.01,.46] $p=0.055$	.25 [.01,.48] $p=0.040$	.01 [−.27,.29] $p=0.957$	FS
Opt-out	LE	100/100/100	.95 [.72,1.19] $p<.001$	.95 [.72,1.19] $p<.001$	.00 [−.28,.28] $p=1.000$	P*
Opt-out	NR	100/100/100	.02 [−.21,.25] $p=0.443$	−.02 [−.25,.21] $p=0.557$	−.04 [−.32,.24] $p=0.775$	FS
Probe (UW density)	LE-lbl	100/100/98	.37 [.14,.61] $p=0.004$	.42 [.19,.66] $p=0.002$	.07 [−.21,.35] $p=0.616$	P
Probe (UW presence)	HE-lbl	100/87/100	.48 [.24,.73] $p<.001$	.41 [.17,.66] $p=0.003$	−.08 [−.36,.20] $p=0.571$	P
Probe (UW presence)	LE-lbl	100/100/98	.35 [.12,.59] $p=0.007$	.44 [.21,.68] $p=0.001$	.08 [−.20,.36] $p=0.560$	P
Probe (answer entropy)	LE-lbl	100/100/100	.94 [.71,1.17] $p<.001$	1.00 [.76,1.23] $p<.001$	.03 [−.25,.30] $p=0.857$	P*
Probe (verbal report)	NR-lbl	99/89/96	.33 [.09,.57] $p=0.011$	.38 [.14,.62] $p=0.005$	.02 [−.26,.31] $p=0.863$	P
SAE score	–	20/20/19	.86 [.32,1.39] $p=0.005$	1.02 [.48,1.56] $p=0.001$	.01 [−.64,.65] $p=0.987$	P*
Verbal report	LE	100/100/97	1.18 [.94,1.41] $p<.001$	1.02 [.78,1.25] $p<.001$	−.22 [−.50,.06] $p=0.130$	P*
Verbal report	NR	99/89/96	.81 [.57,1.05] $p<.001$	.87 [.63,1.11] $p<.001$	.02 [−.27,.30] $p=0.916$	P

Table 7: Qwen3.5-27B: full statistics for every candidate indicator and reasoning version. Column definitions and screening rules are given in the accompanying text.

Indicator	Cond.	n	A–V	A–M	V–M	Screening
Answer entropy	NR	100/100/100	.94 [.71,1.17] $p<.001$	.45 [.21,.68] $p<.001$	−.43 [−.71,−.15] $p=0.003$	VD
Answer entropy	Think	100/100/100	.94 [.71,1.18] $p<.001$	.76 [.52,.99] $p<.001$	−.31 [−.59,−.03] $p=0.031$	VD
CoT length	Think	100/100/100	1.61 [1.38,1.85] $p<.001$	1.28 [1.05,1.52] $p<.001$	−.26 [−.54,.02] $p=0.067$	P*
CoT uncertainty words (UW)	Think	100/100/100	1.54 [1.31,1.77] $p<.001$	1.27 [1.04,1.51] $p<.001$	−.13 [−.41,.14] $p=0.343$	P*
Opt-out	NR	100/100/100	.00 [.00,.00] $p=1.000$	.00 [.00,.00] $p=1.000$	.00 [.00,.00] $p=1.000$	FS

Indicator	Cond.	n	A–V	A–M	V–M	Screening
Opt-out	Think	99/100/100	.65 [.41,.88] $p<.001$	.70 [.47,.94] $p<.001$	.14 [−.14,.43] $p=0.320$	P*
Probe (UW density)	Think-lbl	100/100/100	.76 [.53,1.00] $p<.001$	.85 [.61,1.08] $p<.001$	.05 [−.23,.33] $p=0.725$	P
Probe (UW presence)	Think-lbl	100/100/100	.87 [.63,1.10] $p<.001$	.87 [.64,1.10] $p<.001$	.01 [−.27,.29] $p=0.925$	P*
Probe (answer entropy)	Think-lbl	100/100/100	.57 [.33,.80] $p<.001$	.57 [.33,.80] $p<.001$	.07 [−.21,.35] $p=0.625$	P
Probe (verbal report)	NR-lbl	100/100/100	−.03 [−.26,.20] $p=0.581$	−.19 [−.42,.04] $p=0.909$	−.14 [−.42,.14] $p=0.317$	FS
SAE score	–	20/20/20	−.08 [−.61,.46] $p=0.593$	.03 [−.50,.56] $p=0.463$	.11 [−.53,.75] $p=0.724$	FS
Verbal report	NR	100/100/100	.52 [.29,.75] $p<.001$	.34 [.11,.58] $p=0.008$	−.16 [−.44,.12] $p=0.261$	P
Verbal report	Think	100/100/100	.67 [.43,.90] $p<.001$	.47 [.24,.71] $p<.001$	−.16 [−.44,.12] $p=0.268$	P*

E.2 Representation-Based Analyses

Supervised probes. Figure 10 shows performance across layers and extraction positions. Performance depends on the target and location; the result position generally gives stronger predictions for verbal-report and answer-entropy targets, while uncertainty-word density shows a less consistent positional pattern. Stars mark the configurations used in the main text. These are not uniformly the highest-scoring configurations in the available sweeps: GPT-OSS-20B UW-presence (high-effort target) uses layer 16 at the result position (balanced accuracy 0.567), versus 0.582 at layer 4, boundary; Qwen3.5-27B UW-density uses layer 36, result ($r = 0.151$), versus $r = 0.187$ at layer 29, choice.

Jacobian lens. Jacobian-lens entropy shows modest discrimination across GPT-OSS-20B source layers (Figure 11), reaching AUCs of approximately 0.63 at the result position and 0.57 at the boundary position. The main-text configuration uses the result position at source layer 22, the last fitted source layer, and achieves approximately 0.56.

Sparse autoencoder features. Table 8 summarises SAE development and confirmation results; Figure 12 shows the corresponding distributions. The GPT-OSS-20B score separates Ambiguous from both concentrated conditions in the confirmation set. For Qwen3.5-27B, the development-set separation does not persist in the confirmation set.

E.3 Convergence and Error Patterns

Convergence in Qwen3.5-27B. Binary agreement among the five retained indicators ranges from 0.45 to 0.87, and Pearson correlations from $r = 0.11$ to 0.97 (Figure 13). CoT length and uncertainty-word scores are the most closely associated pair (agreement $= 0.87$, $r = 0.97$). The remaining correlations are weaker, indicating partial convergence.

Shared classification errors. We examine whether indicators misclassify the same trials, treating Ambiguous as positive and the concentrated conditions as negative. CoT length and uncertainty-word measures have correlated binary error flags in both models ($\phi = 0.68$ for GPT-OSS-20B and 0.56 for Qwen3.5-27B; Figure 14). The GPT-OSS-20B answer-entropy probe and SAE score also share errors ($\phi = 0.39$).

E.4 Composite Performance in Qwen3.5-27B

The indicator count and mean standardised score achieve average precision (AP) of 0.73 and 0.74, respectively (Figure 15). Both exceed most individual indicators and perform comparably to CoT length (0.73) and uncertainty-word scores (0.74).

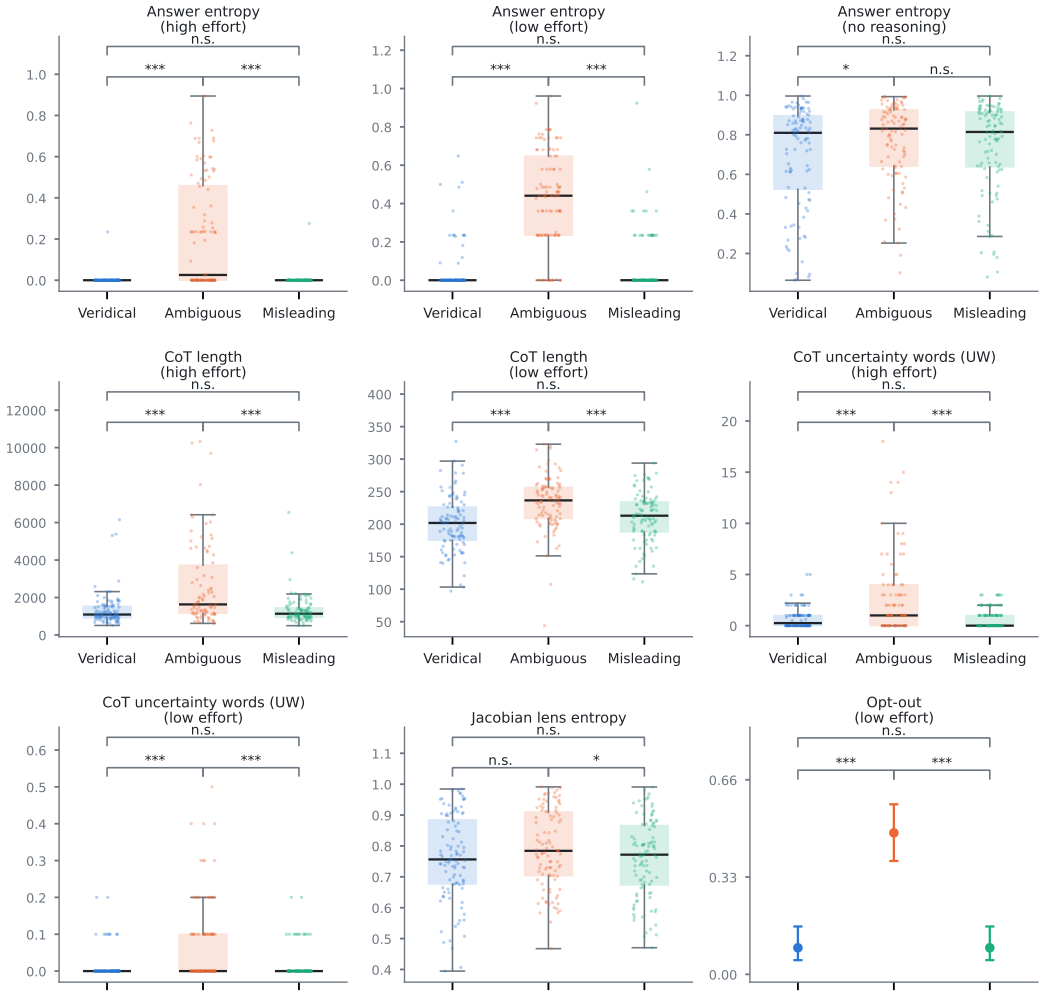

Figure 7: Candidate-indicator distributions for GPT-OSS-20B (part 1 of 2). Panels identify the indicator and reasoning setting or probe target. Boxes span the interquartile range (IQR), with medians marked and whiskers extending to the most extreme observations within $1.5 \times$ IQR of the box edges. Jittered points show all trial observations. Opt-out panels show proportions with 95% Wilson score intervals. Brackets indicate condition comparisons using unadjusted p values: $^{***}p < .001$, $^{**}p < .01$, $^{*}p < .05$; n.s., otherwise. Numerical results and sample sizes are reported in Table 6.

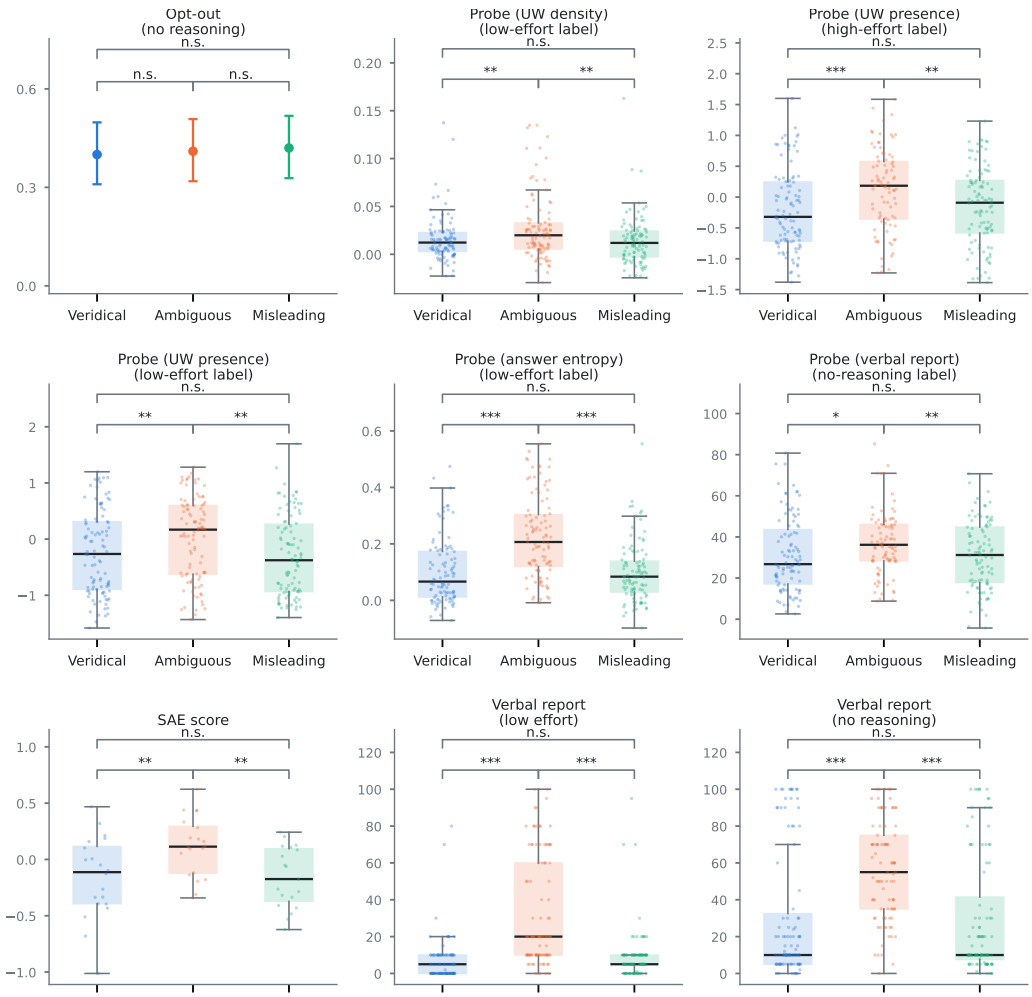

Figure 8: Candidate-indicator distributions for GPT-OSS-20B (part 2 of 2). Display conventions and significance annotations follow Figure 7; full statistics are reported in Table 6.

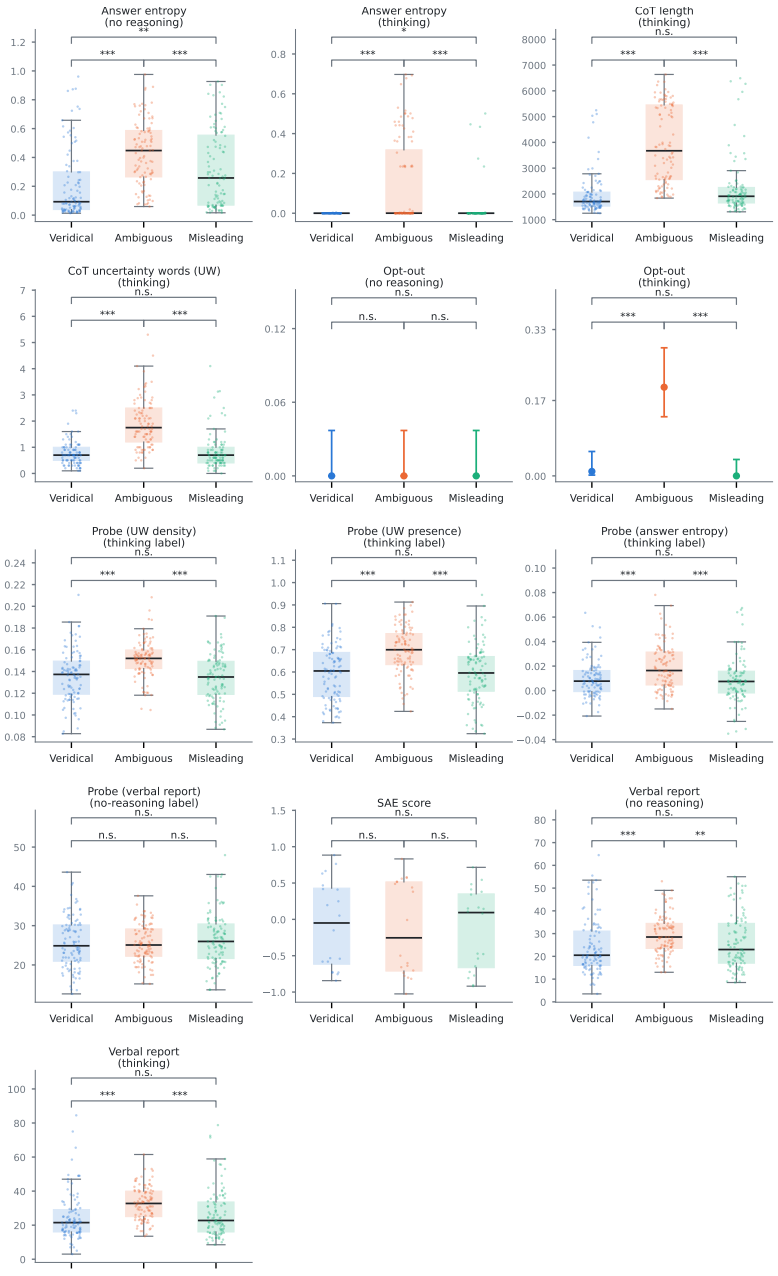

Figure 9: Candidate-indicator distributions for Qwen3.5-27B, using the display conventions of Figure 7. Numerical results and sample sizes are reported in Table 7.

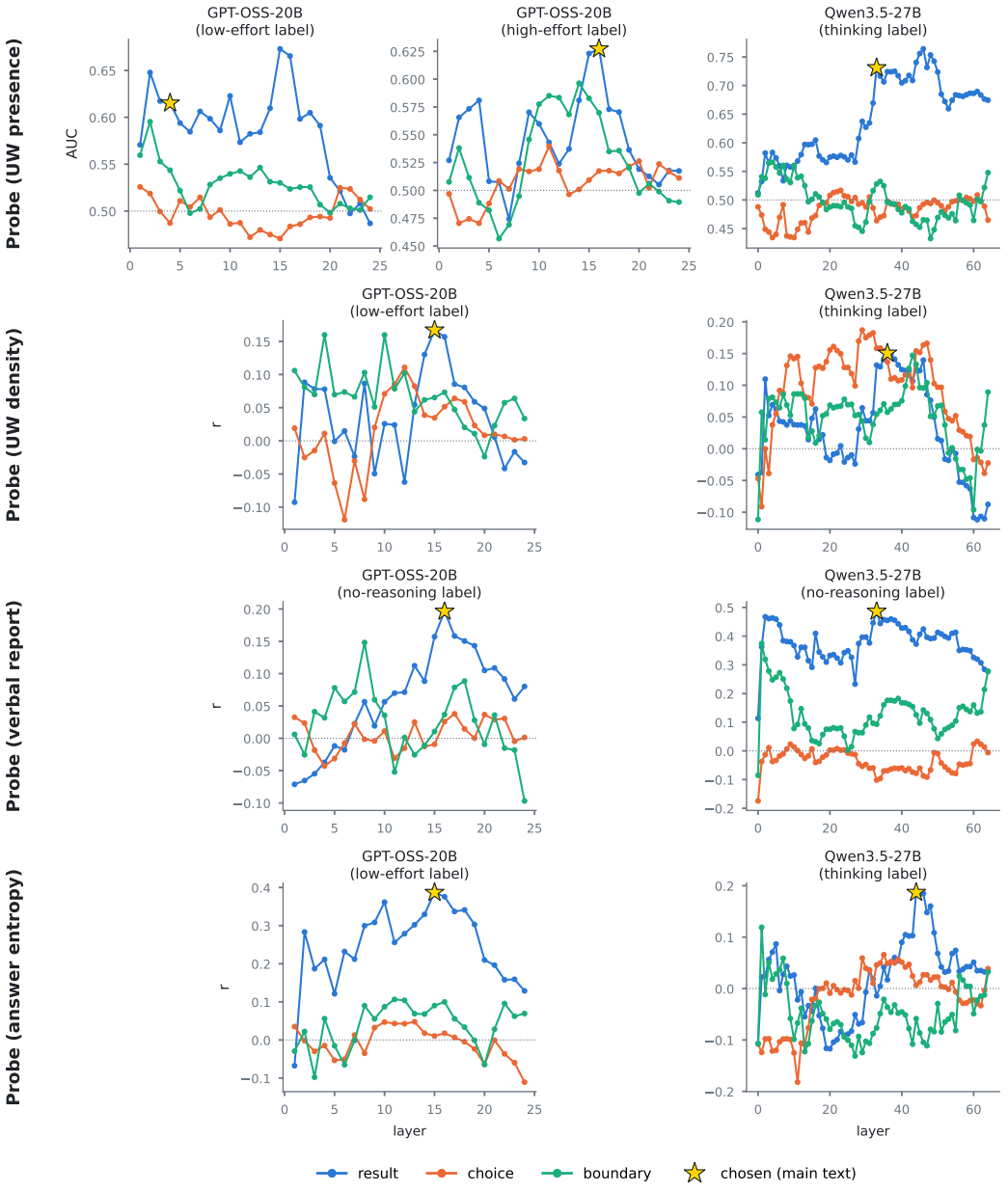

Figure 10: Probe performance across layers and extraction positions. The UW-presence row reports AUC for discriminating AMBIGUOUS from pooled VERIDICAL and MISLEADING trials using probe scores. The other rows report Pearson correlations between cross-validated readouts and their continuous targets. Colours identify result, choice, and boundary positions. Dotted lines mark AUC $= 0.5$ or $r = 0$; stars identify the configurations used in the main text. These rows therefore show different evaluation quantities.

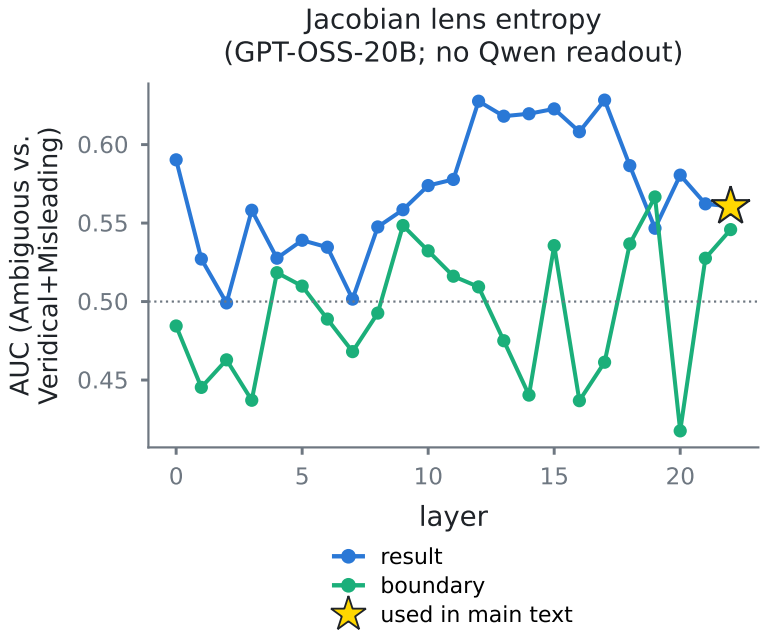

Figure 11: AUC of Jacobian-lens answer-option entropy across source layers and positions in GPT-OSS-20B. The positive condition is AMBIGUOUS; the negative conditions are VERIDICAL and MISLEADING. The dotted line marks AUC $= 0.5$. The star marks the main-text readout at source layer 22 and the result position; it is not the maximum-AUC configuration in this sweep.

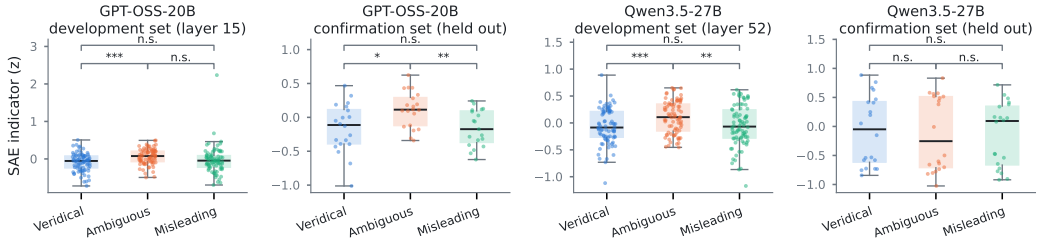

Figure 12: SAE scores in development and confirmation sets for both models. Confirmation scores use features, directions, and standardisation parameters frozen after development. Points represent trials; brackets indicate condition comparisons. Table 8 reports sample sizes, layer selection, and additional diagnostics.

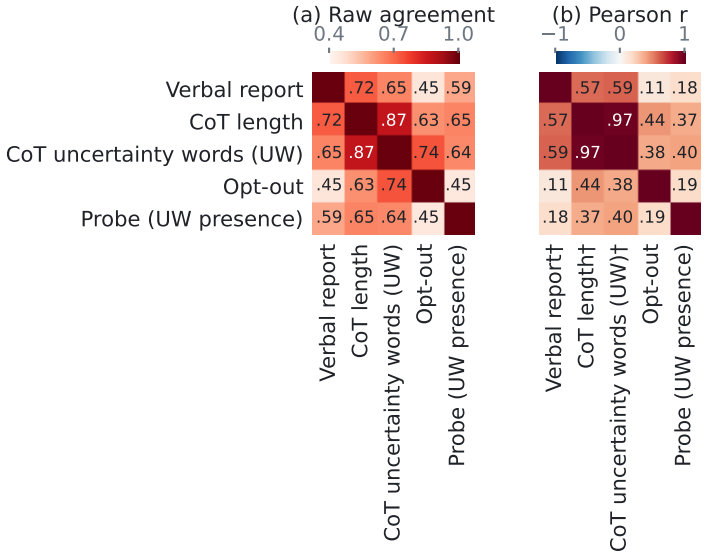

Figure 13: Convergence among retained Qwen3.5-27B indicators. (a) Raw agreement between binary uncertainty flags. (b) Pearson correlations between original scores. Rows and columns follow the same order. Analyses pool the three conditions and exclude missing observations pairwise.

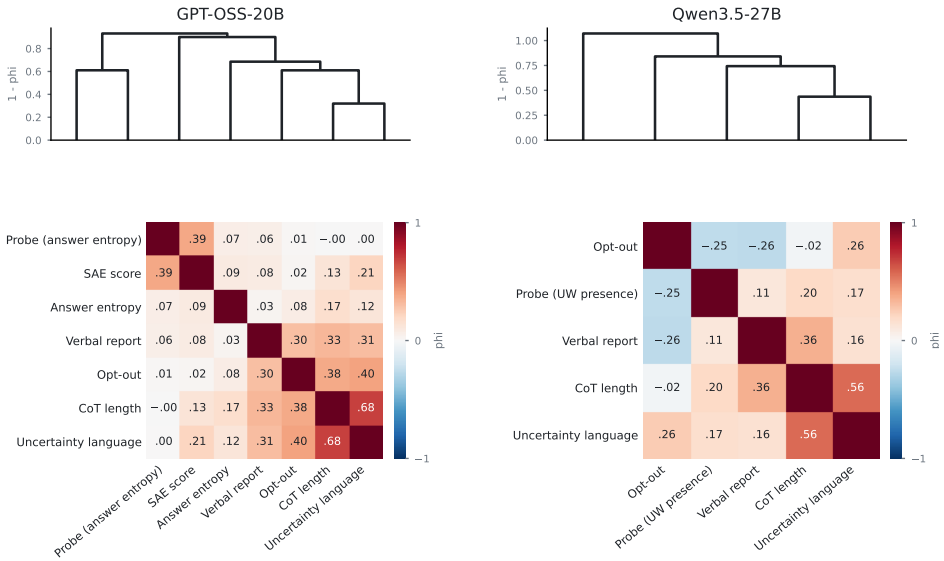

Figure 14: Classification-error correlations and hierarchical clustering. Error flags indicate disagreement with the AMBIGUOUS-versus-concentrated condition label, rather than incorrect cue identification by the model. Heatmaps show pairwise ϕ correlations; dendrograms use average linkage with dissimilarity $1 - \phi$. Positive correlations indicate shared classification errors.

Table 8: SAE development-set layer selection and held-out confirmation results. Arrows identify the retained layer. V–M equivalence refers to the TOST used in the SAE selection pipeline, distinct from the final Welch screening. Blank development-set entries indicate that equivalence was not established. The retention ratio compares the smaller absolute correctness-stratified effect with the absolute overall effect.

GPT-OSS-20B — development-set layer scan (layer 15 only, locally available, 16 features retained at chosen layer)

	Layer	n (dev)	d (A vs. V,M)	V–M TOST equiv.	Retention (dev)
→	15	238	0.425		0.78

Confirmation set (layer 15, frozen, never seen during selection):
$n = 59$ d(A vs. V,M)$= 0.903$ V–M TOST equiv.: no Retention: 0.92
restricted to trials where the model's own answer was correct: $d = 0.939$; restricted to trials where it was incorrect: $d = 0.833$; retention ratio 0.92
convergent validity, confirmation set: verbal report $r = 0.14$, opt-out prop. $r = 0.32$, unc. word prop. $r = 0.15$, answer entropy $r = 0.25$

Qwen3.5-27B — development-set layer scan (layers 4,12,. . . ,60, 18 features retained at chosen layer)

	Layer	n (dev)	d (A vs. V,M)	V–M TOST equiv.	Retention (dev)
	4	240	0.297		0.93
	12	240	0.407		0.99
	20	240	0.420	✓	1.01
	28	240	0.289		0.69
	36	240	0.625		0.51
	44	240	0.761		0.71
→	52	240	0.488	✓	0.74
	60	240	0.642		0.32

Confirmation set (layer 52, frozen, never seen during selection):
$n = 60$ d(A vs. V,M)$= -0.024$ V–M TOST equiv.: no Retention: –
restricted to trials where the model's own answer was correct: $d = -0.701$; restricted to trials where it was incorrect: $d =$ –; retention ratio –
convergent validity, confirmation set: verbal report $r = -0.15$, answer entropy $r = -0.02$, composite z $r = -0.10$

E.5 Individual Indicators Across Evidence Levels

The analysed sample contains 1,460 trial–round observations, with 146 in each of ten equal-width M2 bins. It spans 297 distinct trials: 100 VERIDICAL, 97 AMBIGUOUS, and 100 MISLEADING. All five retained Qwen3.5-27B indicators show an overall decline with increasing ideal-observer posterior concentration in the concentration-balanced sample (Figure 16). Opt-out declines steeply at low-to-intermediate concentration, whereas the other measures generally change more gradually. The composite trends thus accompany changes across multiple constituent indicators.

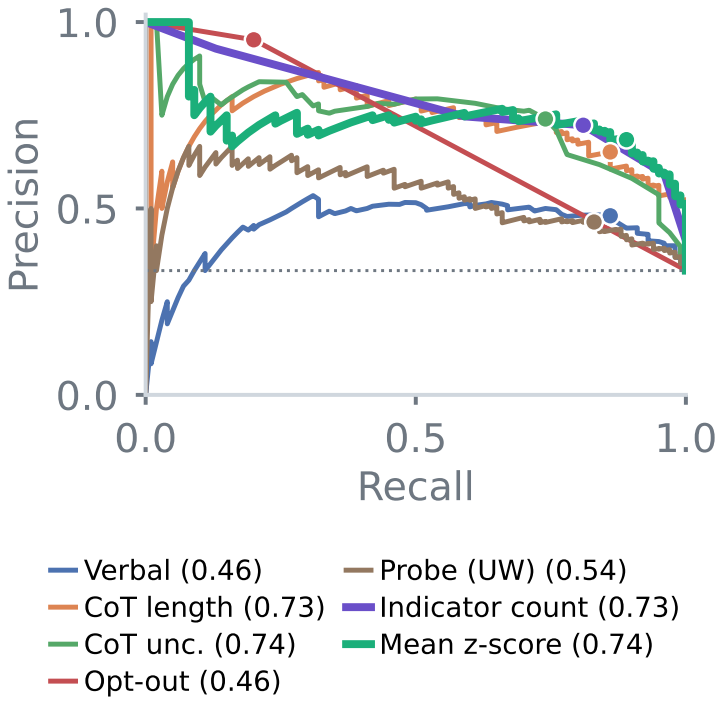

Figure 15: Precision–recall curves for Qwen3.5-27B, treating AMBIGUOUS as positive and pooled VERIDICAL and MISLEADING as negative. Legend values report AP. Thick curves denote the composites; the dotted line marks the positive-class proportion of one third.

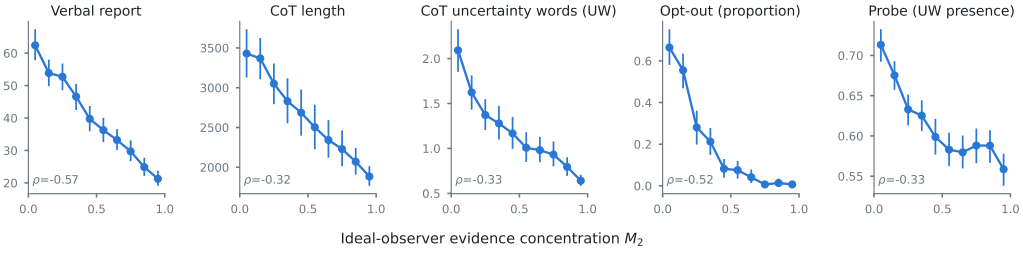

Figure 16: Individual Qwen3.5-27B indicators across ideal-observer posterior concentration, M2. Panels show binned means using the same concentration-balanced sample as the main-text composite analysis. Error bars are 95% percentile intervals from 2,000 trial-cluster bootstrap resamples, retaining observations from the same trial together. Panel annotations report Spearman correlations calculated across individual trial–round observations, rather than trial-level averages or bin means; multiple rounds from the same trial contribute separate observations. Higher M2 indicates more concentrated evidence.