DOES THE DENOISING TRAJECTORY REVEAL THE SOURCE OF UNCERTAINTY IN DIFFUSION LANGUAGE MODELS?

Anonymous authors
Paper under double-blind review

ABSTRACT

For a large language model, the source of uncertainty determines the appropriate response. An ambiguous prompt calls for a clarifying question, whereas missing knowledge calls for retrieval or abstention. In autoregressive generation, the model decides one token at a time. During a single generation we therefore observe only the distribution over the next position, conditioned on a prefix that is already set. Whether the uncertainty stems from many valid answers or from missing knowledge only becomes visible across repeated generations. Diffusion large language models instead generate a distribution over all positions at every denoising step from a single generation. We ask whether this denoising trajectory reveals the source of uncertainty, and make three contributions. First, we construct source-of-uncertainty-labels (SOUL), a benchmark of 16,224 prompts from seven corpora and an authored set, labeled according to established definitions of different sources of uncertainty. Second, we show that the denoising trajectory contains source-specific information across four diffusion language models. Statistical signals summarizing the trajectory identify the source of a prompt well above chance, and outperform the same signals taken from tokenizer-matched autoregressive models, by 0.08–0.22 F1 on task underspecification. Third, we demonstrate that a convolutional probe applied to the full position $\times$ denoising-time trajectory performs within 0.03 F1 of statistical signals, indicating that the information relevant to the source is largely captured by simple, interpretable properties of the denoising process. Because these distinctions are available from the generation a model already produces, they offer a practical basis for deciding when a system should ask, abstain, or answer.

1 INTRODUCTION

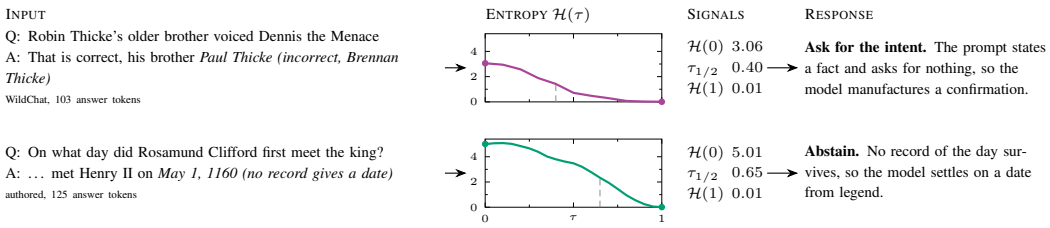


Figure 1: **Entropy trajectories for two sources of uncertainty.** Entropy $H(\tau)$ of a dLLM's posterior over denoising time τ for a prompt with no task (↑) and a prompt without recorded answer (↓).

Tell a language model "Robin Thicke's older brother voiced Dennis the Menace", and it confirms a claim you never asked it to check. Ask it on what day Rosamund Clifford first met the king, and it settles on a date from legend (Figure 1). Both answers look confident, and yet the first prompt should call for a question back to the user, whereas the second should call for an abstention. For a large language model (LLM), the source of its uncertainty therefore decides the appropriate response, which may be to clarify an underspecified request (Zhang et al., 2024), to retrieve a missing

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.

fact (Mallen et al., 2023), to abstain when no answer can be determined today (Amayuelas et al., 2024), or to proceed when several answers are valid (Min et al., 2020; Kirchhof et al., 2025). This makes it important to determine *why* a prompt is uncertain. Uncertainty quantification returns a scalar that measures *how* uncertain the model is (Kuhn et al., 2023; Farquhar et al., 2024), and cannot tell these cases apart, as it collapses toward chance when several answers are valid (Tomov et al., 2025). Thus, identifying the sources of uncertainty is a question of its own. Recent work answers it by intervening on the input or by generating several responses and reading their disagreement (Hou et al., 2024; Nan et al., 2025), and the traditional split into aleatoric and epistemic uncertainty leaves out the cases in which the task or the context is underspecified (Kirchhof et al., 2025). Whether a single generation already carries the source of uncertainty, before any second query and without an auxiliary model, remains open.

Diffusion language models (dLLMs) offer a new way to ask this question. An autoregressive language model (AR–LLM) commits one token at a time, so within a single generation it exposes only the distribution over the next position, conditioned on a prefix that is already fixed. A dLLM instead refines all positions jointly over a sequence of denoising steps and exposes a distribution over the vocabulary at every position and every step (Nie et al., 2025; Ye et al., 2025). A single generation therefore yields a trajectory over positions and denoising time that records how the answer takes shape. Recent work reads this trajectory to estimate how uncertain the model is and to detect whether its answer is wrong (Vazhentsev et al., 2026; Chang et al., 2026; Zhang et al., 2026), but it has not asked whether the trajectory shows *why* the model is uncertain. Figure 1 suggests that it might, as both answers look settled while the two trajectories differ.

This work asks whether the trajectory of a single dLLM generation suffices to recover the source of uncertainty. Answering it requires prompts whose source of uncertainty is known independently of any model. Existing corpora each cover a single source, so we build a benchmark that covers common sources of uncertainty under one label scheme (Section 3). We start from the signals that prior work derives from denoising dynamics (Vazhentsev et al., 2026), each of which reduces the generation to one interpretable number, such as how undecided the model is when denoising starts or how much of a guess each token is when it commits, and identify which of them carry information about the source (Section 5.1). We then remove the denoising steps from these signals by computing them for an AR–LLM that shares the tokenizer and outputs a single posterior per position, which measures what the time dimension contributes (Section 5.2). Finally, we skip the aggregation with a convolutional probe that reads the full trajectory over positions and denoising steps, and quantify whether reducing it to signals discards information about the source (Section 5.3). Our contribution is threefold:

- **SOUL**, a benchmark of 16,224 prompts from seven existing corpora and our own authored prompts, each labeled for four sources of uncertainty by a fixed questionnaire that is blind to the corpus annotations and to any model output, together with a model-agnostic rarity score (Section 3).
- **Identification of the source from the trajectory.** Interpretable signals of four dLLMs recover every source of uncertainty above chance, and they improve on the same signals taken from a tokenizer-matched AR–LLM, most clearly for task underspecification, which shows that the denoising trajectory adds information that a single posterior per position does not hold (Sections 5.1 and 5.2).
- **Sufficiency of the signals.** A convolutional probe over the full position $\times$ time trajectory changes the result by a margin within the fold variance and leaves the lead of the dLLMs intact, which shows that the source can be read from a handful of interpretable numbers (Section 5.3).

2 Related Work

Uncertainty quantification for LLMs typically reduces a generation to a scalar estimate of uncertainty, derived from likelihoods, predictive entropy, or the spread of semantically distinct outputs (Kuhn et al., 2023; Farquhar et al., 2024; Shorinwa et al., 2025). Such estimates describe *how* uncertain the model is, but not *why* it is uncertain. This limitation becomes apparent when multiple answers are valid, where conventional uncertainty estimators can degrade toward random

performance (Tomov et al., 2025). To characterize where uncertainty comes from, traditional approaches distinguish between aleatoric and epistemic uncertainty (Depeweg et al., 2018; Hüllermeier & Waegeman, 2021). For interactive systems, Kirchhof et al. (2025) argue that this distinction is too coarse, since different sources of uncertainty can call for different actions. They instead frame uncertainty in terms of the action it calls for, namely asking, retrieving, or answering. We follow this view and define sources of uncertainty by the intervention needed to resolve them (Section 3.1).

Identifying the source of uncertainty for a given prompt requires prompts with known sources, but existing resources typically isolate individual sources. AmbigQA, CAmbigNQ, and CLAMBER focus on ambiguity that can be resolved through clarification (Min et al., 2020; Lee et al., 2023; Zhang et al., 2024), while KUQ targets questions for which the required knowledge is unavailable (Amayuelas et al., 2024). PopQA and SimpleQA provide complementary evaluations of factual knowledge (Mallen et al., 2023; Wei et al., 2024). Section 3 combines these corpora under a common set of source labels. For individual sources, recent work attempts to identify the source of uncertainty from multiple model responses. Hou et al. (2024) use clarification ensembling, generating several clarified versions of an input to separate uncertainty associated with the input from uncertainty that remains after clarification. Nan et al. (2025) distinguish ambiguity, knowledge gaps, and their combination using multiple responses with an auxiliary language model. Both approaches require repeated model evaluations, and Nan et al. additionally require a second model to interpret the responses. This leaves open whether information about the source is already present in a single generation, without additional queries or auxiliary models.

dLLMs provide a natural setting to study this question, as they generate text by iteratively denoising while producing a distribution over every open position at every step (Austin et al., 2021; Nie et al., 2025; Ye et al., 2025; Wu et al., 2025). A single generation yields a sequence of predictions for each position, forming a position $\times$ time trajectory that is not exposed by a single AR–LLM generation. Several studies have shown that these trajectories are informative: denoising entropy can predict generation quality and guide decoding order (Chen et al., 2025), confidence-based decoding reveals how uncertainty evolves as context is resolved (Ni et al., 2026), and trajectory probes recover linguistic information before exact token identities (Lu, 2026). Vazhentsev et al. (2026) further derive uncertainty scores from intermediate generations, remasking dynamics, and denoising complexity without additional sampling, which motivates our signal catalog (Section 4.2). Denoising trajectories have also been used for hallucination detection (Chang et al., 2026; Qian et al., 2026; Hemmat et al., 2026; Zhang et al., 2026), modeling temporal patterns to assess the reliability of generated answers. Existing work therefore uses denoising trajectories primarily to characterize uncertainty, generation dynamics, or the reliability of the resulting output. Our focus is instead on the source of uncertainty itself, which is distinct from the correctness of the resulting answer: a wrong answer can arise from an underspecified prompt as well as from missing knowledge.

3 SOUL: A BENCHMARK OF LABELED SOURCES OF UNCERTAINTY

Existing corpora each label a single source of uncertainty. For any given prompt they cannot tell whether other sources are present as well. We therefore build a labeled prompt dataset, termed source-of-uncertainty labels (SOUL), in which every prompt carries a label for each of four sources of uncertainty. Section 3.1 defines the four sources. Section 3.2 describes which corpora we combine and how we extend them. Section 3.3 explains how we label the prompts. The resulting dataset allows asking whether the dLLMs' denoising trajectory reveals the source of uncertainty (Section 4).

3.1 SOURCES OF UNCERTAINTY

Most uncertainty estimators target total predictive uncertainty and, if they decompose it at all, split it into an aleatoric and an epistemic term (Depeweg et al., 2018; Hüllermeier & Waegeman, 2021). For an LLM that talks to a user, this split is incomplete in two ways (Kirchhof et al., 2025). First, whether uncertainty is reducible depends on who could reduce it, e.g., the user, a lookup, or no one. Second, the prompt may leave the task or a detail the answer depends on open. We therefore label each prompt by *what would settle its uncertainty*. Each answer to this question corresponds to a source of uncertainty and to a response (Table 1). All four sources are properties of the prompt and independent of any model.

Table 1: **Sources of uncertainty.** Each source is a model-agnostic property, defined by what would settle its uncertainty. The response column names the action the source calls for.

Source of uncertainty	Description	What settles it?	Response
TASK	task underspecification	to name the task	ask for the intent
CONTEXT	context underspecification	to ask the user	ask for the detail
UNRESOLVED	world-side epistemic	nothing, today	abstain
ALEATORIC	aleatoric	nothing, ever	answer with one or the set
RARITY	model-side epistemic	to look it up	retrieve

Task underspecification (TASK). The prompt does not determine the task. TASK holds when a competent answerer could not name the one action the user asks for, because several tasks are plausible and none is the default (Kirchhof et al., 2025). *"It's my mom's birthday."* may ask for a greeting, a reminder or a gift idea.

Context underspecification (CONTEXT). The task is clear, but the answer depends on a detail that only the user can supply. *"When did the first Harry Potter movie come out?"* depends on the country (Min et al., 2020). The answer remains uncertain until the user provides the missing detail.

Aleatoric multiplicity (ALEATORIC). Several materially different answers remain correct after everything has been looked up and the user has been asked. This is data uncertainty as a property of the prompt (Lahlou et al., 2023; Hüllermeier & Waegeman, 2021), not the residual a model class cannot reduce (Schweighofer et al., 2025). Creative requests, preferences and value judgements are ALEATORIC, as are pick-some requests, e.g., *"Name three sports teams in Boston"*.

Unresolved in the world (UNRESOLVED). A true answer exists or will exist, but nobody can observe it today. The event has not happened, the problem is open, or the record is lost. This is epistemic uncertainty (Gruber et al., 2026). We follow the convention that UNRESOLVED is what no one can reduce today and ALEATORIC what no one could reduce ever (Der Kiureghian & Ditlevsen, 2009). The death toll of the Wilhelm Gustloff is UNRESOLVED, not ALEATORIC, because a single true value existed.

Rare representation (RARITY). Whether a model knows a retrievable answer is model-side epistemic uncertainty. It is a property of the model and its training data, not of the prompt, so it is not a label in SOUL. As a model-agnostic proxy we score each prompt by the rarity (ϱ) of its rarest entity-like span in an open pre-training corpus, following the finding that fact frequency predicts recall (Kandpal et al., 2023; Mallen et al., 2023). We use rarity to balance the authored set (Section 3.2) and study its effects in detail as an ablation in Section A.4.

3.2 PROMPTS

The SOUL dataset contains 16,224 prompts, 10,003 taken from seven public corpora and 6,221 as authored dataset. *AmbigQA* (Min et al., 2020) and *CLAMBER* (Zhang et al., 2024) supply questions that need a detail only the user holds, *KUQ* (Amayuelas et al., 2024) supplies questions that no one can answer today, and *WildChat* (Zhao et al., 2024) supplies real conversation openers that often not contain a task. *PopQA* (Mallen et al., 2023) spans settled facts from common to rare, and *SimpleQA* (Wei et al., 2024) and *HaluEval* (Li et al., 2023) add rare questions. Every prompt carries a label for each of the four sources and a rarity score ϱ (Table 2). We label a first pass drawn round-robin across corpora, and a second pass that selects further prompts so that each source can be contrasted with CONTROL prompts of the same corpus, matched on first word and prompt length.

In these corpora the labels correlate with corpus, since each corpus was built around one source. Thus, we author every source in the style of every corpus. Each authored prompt rewrites a fact from a CONTROL pool item of the imitated corpus. These facts are half anchored on common and half on rare facts using the median rarity as cut (Section A).

3.3 LABELS

Each label is annotated by a fixed set of yes/no questions (Section A.1). Labelers do not see a corpus's own annotation or any model output. LLMs worked through the questionnaire in batches.

Table 2: **Source labels by corpus.** Number of prompts per label and share of the corpus in parentheses. A prompt can carry several labels, so shares do not sum to one. CONTROL counts prompts with no label. ϱ is the median rarity score of the corpus (Section A.3).

Corpus	prompts	ϱ	CONTROL	TASK	CONTEXT	UNRESOLVED	ALEATORIC
AmbigQA	2297	-10.4	1631 (71%)	0 (0%)	656 (29%)	5 (0%)	5 (0%)
CLAMBER	1597	-12.0	948 (59%)	0 (0%)	555 (35%)	1 (0%)	107 (7%)
HaluEval	797	-4.2	785 (98%)	0 (0%)	11 (1%)	0 (0%)	1 (0%)
KUQ	2000	-10.7	914 (46%)	2 (0%)	332 (17%)	412 (21%)	364 (18%)
PopQA	1835	-10.9	1415 (77%)	0 (0%)	413 (23%)	4 (0%)	3 (0%)
SimpleQA	893	-4.1	893 (100%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)
WildChat	584	-9.6	286 (49%)	58 (10%)	44 (8%)	6 (1%)	207 (35%)
authored	6221	-7.2	1336 (21%)	1238 (20%)	1201 (19%)	1210 (19%)	1254 (20%)
total	16224	-8.6	8208 (51%)	1298 (8%)	3212 (20%)	1638 (10%)	1941 (12%)

On a 125-item calibration batch, two independent LLMs agreed on all labels for 94.4% of prompts (Cohen's κ between 0.88 and 1.00 per label). The disagreements and unsure labels go to adjudication (no majority vote). Authored prompts are labeled the same way. For the authored prompts, the labels agree with the intended source of uncertainty for 96.8% of prompts. On 120 prompts drawn at random from the labeled set, a human walked the same questionnaire without seeing the LLM labelers' answers, agreeing on all labels for 97% of prompts. Thus, every prompt in SOUL carries its corpus, the four labels indicating the sources of uncertainty, and its rarity score. Corpus and rarity correlate with the labels, and so does the length of the answer a model produces. Any of the three could let a classifier separate the sources without reading the trajectory.

4 Reading the Source of Uncertainty from the Trajectory

This section describes how we read the source of uncertainty from a single generation. We first recap how a dLLM generates a response and represent it as a posterior cube over denoising steps, answer positions and tokens (Section 4.1). We then chain aggregations along the axes of the cube until a single scalar remains. Different operators yield different signals (Section 4.2). Finally, we describe a convolutional probe that reads the cube directly instead of collapsing it (Section 4.3). Section 5 then tests whether signals and probe separate the four sources of uncertainty defined in Section 3.1.

4.1 Posterior cube

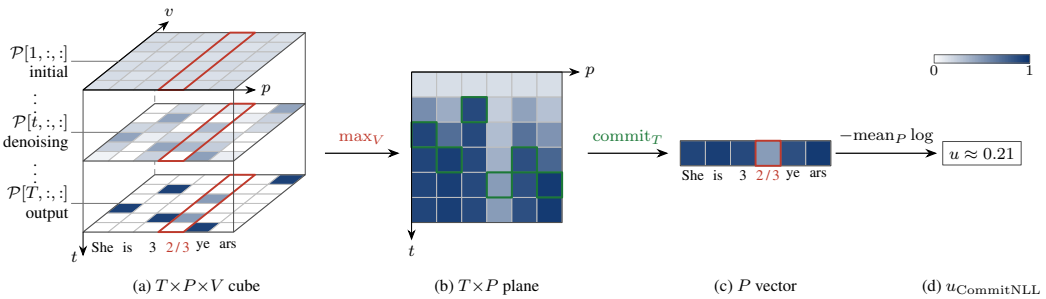
Given a prompt x, a dLLM (Nie et al., 2025) starts with an answer of P positions and refines all of them over T denoising steps. At every step t the network outputs a posterior $\mathcal{P}[t, p, \cdot]$ over the tokens $\mathcal{V}$ at every position p, from which the sampler commits tokens. We define the resulting $T \times P \times |\mathcal{V}|$ *posterior cube* as:

$$\mathcal{P}[t, p, \cdot] \in \Delta^{|\mathcal{V}|-1}, \qquad t = 1, \ldots, T, \quad p = 1, \ldots, P, \tag{1}$$

Figure 2 (a) shows a posterior cube, with the step t running downward, the position p to the right, and the token v into the page. Each horizontal slice $\mathcal{P}[t, :, :]$ is the grid of posteriors at one step, shaded from white (probability zero) to dark blue (one). In the first steps t the mass at each position p is spread over many tokens. As t grows the distributions sharpen until each position commits to a single token. The number of denoising steps T differs between generations, so we normalize time to $\tau = t/T \in [0, 1]$. For an AR–LLM, $\mathcal{P}$ loses the time axis, because the model outputs a single posterior per position.

4.2 Denoising Trajectory signals

A trajectory signal is a scalar computed from a single generation, without extra samples or a second forward pass. Our signals extend the measures of (Vazhentsev et al., 2026). Every signal is a chain of aggregations of the posterior cube $\mathcal{P}$ defined in Section 4.1. The first aggregation collapses the tokens and turns the posterior $\mathcal{P}[t, p, \cdot]$ of one position p at one step t into a single number, which



(a) $T \times P \times V$ cube (b) $T \times P$ plane (c) P vector (d) $u_{\text{CommitNLL}}$

Figure 2: **Computing signals from the posterior cube.** (a) The posterior cube over steps t, positions p and tokens v for the prompt "It is the ICLR paper deadline 25th of September 2026. How old is Lisa born 1993?". The answer is 32 or 33 depending on Lisa's birthday, a detail only the user holds (CONTEXT), so position p_4 splits its mass between the tokens 2 and 3. (b) to (d) chained aggregations collapse the cube axis by axis, from a plane over steps and positions to a vector over positions to the scalar signal `CommitNLL` (cf. Section B).

gives a $T \times P$ plane per generation. We use five such planes, all standard in uncertainty quantification for language models (Shorinwa et al., 2025; Vazhentsev et al., 2026): the log confidence (log of the top probability), the Shannon entropy, the Rényi-2 entropy (how concentrated the posterior is on its top tokens), the top token, and the held state, which records whether a position keeps its token from one step to the next.[1]

Two collapses remain, over the step axis and over the position axis. For the scalar signals we aggregate each plane along the step axis t. We then collapse the resulting P vector into the scalar signal u by its mean, a quantile or its spread across positions. We provide two concrete examples here and point to Section B for the complete signals catalog used in Sections 5.1 and 5.3: (i) Averaging a plane along t gives its mean state during denoising. Averaged again over positions, yields TRAJENTROPY, how undecided the model was while it wrote the answer. (ii) If instead we aggregate the $T \times P$ plane by identifying the commit step t_p of a position[2], we get the state at the moment its token was fixed. The negative log of the top probability at that step, averaged over positions, yields COMMITNLL, how much of a guess each committed token was (cf. Figure 2).

An AR–LLM outputs one posterior per position, so its cube has no step axis (Section 4.1). For the AR–LLMs we therefore drop the signals built on the held state and the argmax flips, which need more than one step per position, and compute the remaining signals on the single posterior of each position, averaged over positions.

4.3 CONVOLUTIONAL PROBE

Each signal collapses the posterior cube one axis at a time. A pattern that spans steps and positions together, such as uncertainty moving from one part of the answer to another during denoising, is averaged away before reaching the scalar (Zhang et al., 2026). To learn such patterns directly from the posterior cube, we train a convolutional probe on the five $T \times P$ planes of Section 4.2, replacing the top token by the argmax flip, which marks every step at which the top token changes. Each plane is resampled to 21 points of normalized time τ and 32 position bins, giving every generation a $5 \times 21 \times 32$ input. The probe is a small convolutional neural network (CNN) with two convolutional blocks and a linear head, trained and evaluated as the signal classifiers of Section C. The gap between probe and signals therefore measures how much source information the chained aggregations discard. For the AR–LLMs each position holds a single value per plane, so we repeat it over τ and set the held and flip planes to zero. Section B.1 contains the architecture and training details.

[1] For masked dLLMs the held state marks the unmasking event; for DiffusionGemma it marks acceptance by the sampler.

[2] We compute commit signals only for masked dLLMs and not for DiffusionGemma, which rewrites tokens instead of committing them.

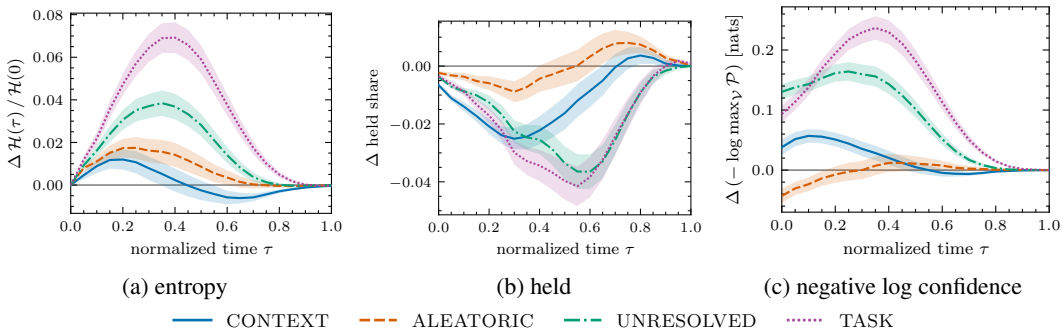


Figure 3: **DiffusionGemma.** Deviations from the CONTROL prompts of the same corpus and answer length, over normalized denoising time τ. Bands are 95% intervals.

5 Experiments

We answer three questions about the posterior cube and the signals derived from it. Do the trajectory signals of a dLLM identify the source of uncertainty (Section 5.1)? How much of that is due to the denoising trajectory, which an AR–LLM does not have (Section 5.2)? And does the posterior cube hold information that the scalar signals miss (Section 5.3)? Table 3 summarizes the main results. Each block pairs a dLLM with its matched AR–LLM, and within a block the rows give the signal classifier (Section 4.2) and the probe (Section 4.3) for each model. Each question is answered by comparing rows of this table.

We run four dLLMs on all 16,224 prompts of SOUL (Section 3). `LLaDA2.1-mini`, `Dream-v0-Instruct-7B` and `Nemotron-Labs-Diffusion-14B` are masked dLLMs. `DiffusionGemma-26B-A4B-it` starts from random tokens instead of masks and rewrites them step by step. For the second question we pair three of them with an AR–LLM that shares their tokenizer and vocabulary: `Gemma 4 26B-A4B-it` (thinking disabled) for DiffusionGemma, `Qwen2.5-7B-Instruct` for Dream and `Ministral-3-14B-Instruct-2512-BF16` for Nemotron. `LLaDA2.1` has no AR–LLM with a shared tokenizer. All models decode greedily with a maximum generation length of 512 tokens.

5.1 Identifying the source of uncertainty from trajectory signals

We first ask whether dLLM trajectories differ across sources of uncertainty. Figure 3 plots, for DiffusionGemma, how the entropy, the held share and the log confidence deviate over normalized time τ from CONTROL prompts matched by corpus and answer length. The trajectories separate by source: TASK and UNRESOLVED show more early entropy and decay more slowly, while CONTEXT stays near CONTROL throughout. The other dLLMs show the same pattern (Section D.2).

We then ask whether single signals identify sources of uncertainty, treating each as a score in a univariate analysis (Section D.1). The strongest single signals identify UNRESOLVED and TASK on three of the four dLLMs, with top AUROCs of 0.79–0.84 on Nemotron, 0.73–0.75 on DiffusionGemma and 0.69–0.73 on LLaDA2.1; Dream does not exceed 0.63. The entropy at the start of denoising is the best or near-best signal on every dLLM. On Nemotron also the surprisal at the commit step matches it.

To use all signals at once, we train one classifier per model that assigns each prompt to one of the four sources or CONTROL (cf. Section C). We evaluate different signal sets and report the best under 5-fold cross-validation. We report F1 per class and the F1 across the five classes (Table 3). Answer length and corpus both correlate with the labels. A classifier on answer length alone reaches 0.25 to 0.34 across models (Section D.3).

The trajectory signals identify sources of uncertainty. The first row in each model block, *dLLM and (a) Signals*, in Table 3 answers the first question. The classifiers achieve an F1 score of 0.42–0.47 across the five classes and recover every source with an F1 score between 0.31 and 0.49. For

Table 3: **Classification of the source of uncertainty.** One classifier per model assigns each prompt to one of the four sources or CONTROL by the argmax over the five classes. Columns give the F1 per class and the unweighted mean over the five classes, as mean ± standard deviation over folds. Each block pairs a dLLM with its tokenizer-matched AR–LLM; rows (a) and (c) use the trajectory signals, rows (b) and (d) the convolutional probe. Within a block and column, **bold** marks the highest value and underline the second highest.

		F1 per class					overall
		TASK	CONT.	UNRES.	ALEAT.	CONTROL	F1
DiffusionGemma	(a) Signals	**.38** ±.02	.43 ±.01	.40 ±.01	.46 ±.03	.64 ±.01	.46 ±.01
	(b) Probe	.37 ±.04	**.46** ±.02	**.47** ±.03	**.52** ±.01	.64 ±.03	**.49** ±.02
Gemma 4 (AR)	(c) Signals	.22 ±.01	.44 ±.02	.41 ±.01	.42 ±.01	**.64** ±.00	.43 ±.00
	(d) Probe	.31 ±.04	.45 ±.02	.39 ±.02	.27 ±.05	.63 ±.03	.41 ±.02
Dream	(a) Signals	.34 ±.02	.32 ±.03	.39 ±.02	.35 ±.02	**.68** ±.01	**.42** ±.01
	(b) Probe	**.34** ±.01	.31 ±.03	.42 ±.02	.35 ±.03	.65 ±.03	.41 ±.01
Qwen2.5 (AR)	(c) Signals	.26 ±.03	.36 ±.01	**.43** ±.02	.38 ±.04	.65 ±.01	.41 ±.01
	(d) Probe	.25 ±.02	**.37** ±.02	.40 ±.03	**.38** ±.03	.62 ±.07	.40 ±.01
Nemotron	(a) Signals	.44 ±.03	.36 ±.02	**.49** ±.01	**.42** ±.02	.65 ±.01	.47 ±.01
	(b) Probe	**.50** ±.03	.36 ±.06	.47 ±.03	.42 ±.03	.65 ±.05	**.48** ±.01
Ministral 3 (AR)	(c) Signals	.22 ±.02	.29 ±.02	.43 ±.01	.42 ±.02	**.68** ±.01	.41 ±.01
	(d) Probe	.27 ±.02	**.37** ±.01	.40 ±.03	.35 ±.03	.63 ±.05	.40 ±.02
LLaDA2.1	(a) Signals	.31 ±.02	.34 ±.01	.40 ±.02	.43 ±.01	**.64** ±.02	.42 ±.01
	(b) Probe	**.32** ±.04	**.36** ±.02	**.41** ±.03	**.45** ±.01	.64 ±.03	**.44** ±.00

comparison, a classifier that always predicts CONTROL (the majority class) achieves an F1 score of 0.13 across the five classes, and one that guesses each class based on its frequency achieves 0.20. The breakdown over sources of uncertainty shows CONTROL is the easiest class for every model, with an F1 score of 0.64–0.68. TASK is the hardest for DiffusionGemma and LLaDA2.1, CONTEXT for Dream and Nemotron. Thus, signals from a single denoising trajectory carry enough information to assign a prompt to its source of uncertainty well above chance.

5.2 Comparing the denoising trajectories with autoregressive baselines

To measure what the denoising trajectory adds, we compute the signals that a single posterior per position allows for the three AR–LLMs (Section 4.2) and train the classifiers identically. We compare *dLLM and (a) Signals* against *AR–LLMs and (c) Signals* in Table 3. The AR–LLMs reach F1 scores of 0.41–0.43 across all classes. The dLLMs improve on this by 0.03 for DiffusionGemma, 0.01 for Dream and 0.06 for Nemotron. Answer length does not explain this gap (Section D.3). Nemotron improves on Ministral 3 by 0.22 on TASK, 0.07 on CONTEXT and 0.06 on UNRESOLVED. DiffusionGemma improves on Gemma 4 by 0.16 on TASK and 0.04 on ALEATORIC, staying within one standard deviation on the other classes. Dream improves on Qwen2.5 by 0.08 on TASK and falls 0.03 to 0.04 behind on the other three. TASK is the only source on which every dLLM improves on its AR–LLM (Table 15). Overall, the denoising trajectory adds source information beyond a single posterior per position.

5.3 Probing the posterior cube with convolutions

Finally, we ask whether the aggregation of the posterior cube into a few signals discards information that a probe over the full position × time trajectory could recover. The comparison is row (b) against row (a) for the dLLMs and row (d) against row (c) for the AR–LLMs in Table 3. On the dLLMs the probe changes the F1 score across the five classes by at most 0.03. Row (b) improves on row (a) by 0.03 for DiffusionGemma, 0.02 for LLaDA2.1 and 0.01 for Nemotron, and falls 0.01 behind for Dream. On the AR–LLMs the input to the probe is constant over τ. Row (d) falls 0.01 to 0.02 behind row (c) across models. In other words, the patterns across positions and steps that the convolutional probe can learn add little to what the chained aggregations already capture in the signals (cf. Table 16).

6 DISCUSSION

We show that the denoising trajectory carries source-specific information that is not fully captured by a single posterior. This effect is clearest for TASK: a prompt that leaves the task unspecified stays undecided through the middle of denoising (Figure 3), and every dLLM improves over its tokenizer-matched AR–LLM there. UNRESOLVED shows in initial entropy and commit surprisal, but much of that signal is already present in the AR–LLMs. Therefore, the trajectory does not provide a uniformly stronger measure of uncertainty, it exposes a temporal structure in uncertainty that can distinguish sources that look similar at the endpoint. Unlike approaches that infer source from repeated generations or an auxiliary model (Hou et al., 2024; Nan et al., 2025), a single dLLM generation already contains it, which gives the action-based view of Kirchhof et al. (2025) an empirical footing.

CONTEXT is the one source for which the trajectory adds nothing beyond a single posterior. No single trajectory signal separates it within corpus $\times$ length (Section D.1), its curves stay at CONTROL (Section D.2), and its multivariate F1 is matched by the AR–LLM and for DiffusionGemma by answer length alone (Table 14). A prompt that lacks a detail the user holds can invite a plausible guess, and that guess can follow much the same denoising path as an answer to a prompt that lacks nothing (Table 15). The denoising trajectory can therefore reveal the source of uncertainty when that source shapes the process of resolution, but it cannot reveal a source that leaves that process unchanged. What determines whether a source leaves such a signature remains an open question for SOUL.

The practical appeal of these signals is that the relevant source information survives a simple reduction of the trajectory. Probing the full $\tau \times P$ planes changes F1 by at most 0.03 and leaves relative source performance and dLLM–AR comparison unchanged, indicating that source attribution does not depend on complex interactions across particular positions and denoising steps. Instead, a few interpretable temporal statistics capture the useful signal, at least relative to a 5.8k-parameter probe (Section B.1), and extend uncertainty estimation from whether an answer is wrong (Vazhentsev et al., 2026) toward why.

This information comes from a single generation without additional sampling or a second forward pass. Thus, it can inform the response, e.g., an UNRESOLVED prompt to abstention, an ALEATORIC prompt to an answer that keeps its alternatives. An F1 of 0.42–0.49 is well above the 0.13–0.20 of naive baselines but not enough to take these decisions unattended. Table 15 shows that TASK is often mistaken for UNRESOLVED or ALEATORIC, and CONTEXT for CONTROL, so a system acting on these signals would over-answer rather than over-ask. We read the result as a floor for routing, not a routing policy.

Limitations. Several limitations qualify these conclusions. Only one of the three dLLM/AR pairs, DiffusionGemma and Gemma 4, shares an architecture, so the other two confound the time axis with differences in pretraining and scale; the within-model ablation of Section D.5, which strips the trajectory from the same dLLM, does not share this confound and shows the same ordering. All models decode greedily, and confidence-ordered decoding changes when and at what entropy a position commits (Ni et al., 2026), so the signals may shift under other decoding policies. In the public corpora TASK and UNRESOLVED each concentrate in one corpus, however, the authored prompts and the stratified evaluation counter this.

7 CONCLUSION

We asked whether the denoising trajectory of a dLLM reveals why the model is uncertain. On SOUL, a benchmark of 16,224 prompts labeled with four sources of uncertainty, interpretable signals of the trajectory identify every source well above chance for all four dLLMs, improve on a tokenizer-matched AR–LLM wherever one exists, most clearly on task underspecification, where the gain is 0.08 to 0.22 F1, and come within 0.03 F1 of a convolutional probe over the full posterior cube. The source of uncertainty therefore shows in *how* the model's uncertainty resolves over denoising, and a few statistics of that resolution capture most of it. The exception is a prompt that lacks a detail the user holds, which produces a trajectory much like a prompt that lacks nothing. The source of uncertainty decides the response, and our results show that a single generation already carries enough of it to inform that choice.

AI USE STATEMENT

In this work, we use generative AI tools to label the prompts of SOUL. Two LLM agents answer the fixed questionnaire of Section A.1 for every prompt. The labels are validated against the intended source of every authored prompt (96.8% agreement) and against a human annotator on 120 randomly drawn prompts (97% agreement, Section 3.3). We further use generative AI tools to generate the authored prompts of Section 3.2 from templates that we specify. Additionally, we use generative AI tools to edit the text for readability. We also use generative AI tools to write and edit code. The authors test the code and verify its outputs against the reported results. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The dataset SOUL and the code for all experiments are available at `https://anonymous.4open.science/r/soul-2E6D`. Section 3 and Section A.1 describe how the prompts are selected and labeled, Section D.1 defines every signal, Section C gives the evaluation protocol, and Section B.1 the probe. The repository includes the inference scripts and out-of-fold predictions for every model, so the tables and figures can be recomputed.

REFERENCES

Alfonso Amayuelas, Kyle Wong, Liangming Pan, Wenhu Chen, and William Yang Wang. Knowledge of knowledge: Exploring known-unknowns uncertainty with large language models. In *Association for Computational Linguistics*, 2024.

Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. In *Advances in Neural Information Processing Systems*, 2021.

Shenxu Chang, Junchi Yu, Weixing Wang, Yongqiang Chen, Jialin Yu, Philip Torr, and Jindong Gu. TraceDet: Hallucination detection from the decoding trace of diffusion large language models. In *International Conference on Learning Representations*, 2026.

Ziyu Chen, Xinbei Jiang, Peng Sun, and Tao Lin. Optimizing decoding paths in masked diffusion models by quantifying uncertainty. *arXiv*, 2025.

Stefan Depeweg, Jose Miguel Hernandez-Lobato, Finale Doshi-Velez, and Steffen Udluft. Decomposition of uncertainty in Bayesian deep learning for efficient and risk-sensitive learning. In *Proceedings of the 35th International Conference on Machine Learning*. PMLR, 2018.

Armen Der Kiureghian and Ove Ditlevsen. Aleatory or epistemic? Does it matter? *Structural Safety*, 2009.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 2024.

Cornelia Gruber, Patrick Oliver Schenk, Malte Schierholz, Frauke Kreuter, and Göran Kauermann. Sources of uncertainty in supervised machine learning—a statisticians' view. *Statistical Science*, 2026.

Arshia Hemmat, Philip Torr, Yongqiang Chen, and Junchi Yu. TDGNet: Hallucination detection in diffusion language models via temporal dynamic graphs. *arXiv*, 2026.

Bairu Hou, Yujian Liu, Kaizhi Qian, Jacob Andreas, Shiyu Chang, and Yang Zhang. Decomposing uncertainty for large language models through input clarification ensembling. In *Proceedings of the 41st International Conference on Machine Learning*. PMLR, 2024.

Eyke Hüllermeier and Willem Waegeman. Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Machine Learning*, 2021.

Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. Large language models struggle to learn long-tail knowledge. In *International Conference on Machine Learning (ICML)*, Proceedings of Machine Learning Research, 2023.

Michael Kirchhof, Gjergji Kasneci, and Enkelejda Kasneci. Position: Uncertainty quantification needs reassessment for large-language model agents. In *Proceedings of the 42nd International Conference on Machine Learning*. PMLR, 2025.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *International Conference on Learning Representations*, 2023.

Salem Lahlou, Moksh Jain, Hadi Nekoei, Victor Ion Butoi, Paul Bertin, Jarrid Rector-Brooks, Maksym Korablyov, and Yoshua Bengio. DEUP: Direct epistemic uncertainty prediction. *Transactions on Machine Learning Research*, 2023.

Dongryeol Lee, Segwang Kim, Minwoo Lee, Hwanhee Lee, Joonsuk Park, Sang-Woo Lee, and Kyomin Jung. Asking clarification questions to handle ambiguity in open-domain QA. Association for Computational Linguistics, 2023.

Junyi Li, Xiaoxue Cheng, Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. HaluEval: A large-scale hallucination evaluation benchmark for large language models. In *Conference on Empirical Methods in Natural Language Processing*, 2023.

Jiacheng Liu, Sewon Min, Luke Zettlemoyer, Yejin Choi, and Hannaneh Hajishirzi. Infini-gram: Scaling unbounded n-gram language models to a trillion tokens. *arXiv*, 2025.

Harry Lu. Measuring temporal linguistic emergence in diffusion language models. *arXiv*, 2026.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the Association for Computational Linguistics*, 2023.

Sewon Min, Julian Michael, Hannaneh Hajishirzi, and Luke Zettlemoyer. AmbigQA: Answering ambiguous open-domain questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 2020.

Yang Nan, Pengfei He, Ravi Tandon, and Han Xu. Can multiple responses from an LLM reveal the sources of its uncertainty? In *Findings of the Association for Computational Linguistics: EMNLP 2025*. Association for Computational Linguistics, 2025.

Zanlin Ni, Shenzhi Wang, Yang Yue, Tianyu Yu, Weilin Zhao, Yeguo Hua, Tianyi Chen, Jun Song, Cheng Yu, Bo Zheng, and Gao Huang. The flexibility trap: Why arbitrary order limits reasoning potential in diffusion language models. *arXiv*, 2026.

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models. In *Advances in Neural Information Processing Systems*, 2025.

Yanyu Qian, Yue Tan, Yixin Liu, Wang Yu, and Shirui Pan. DynHD: Hallucination detection for diffusion large language models via denoising dynamics deviation learning. *arXiv*, 2026.

Kajetan Schweighofer, Lukas Aichberger, Mykyta Ielanskyi, and Sepp Hochreiter. On information-theoretic measures of predictive uncertainty. In *Proceedings of the 41st Conference on Uncertainty in Artificial Intelligence (UAI)*, Proceedings of Machine Learning Research, 2025.

Ola Shorinwa, Zhiting Mei, Justin Lidard, Allen Z. Ren, and Anirudha Majumdar. A survey on uncertainty quantification of large language models: Taxonomy, open research challenges, and future directions. *ACM Computing Surveys*, 2025.

Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, Valentin Hofmann, Ananya Harsh Jha, Sachin Kumar, Li Lucy, Xinxi Lyu, Nathan Lambert, Ian Magnusson, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew E. Peters, Abhilasha Ravichander, Kyle Richardson, Zejiang Shen, Emma Strubell, Nishant Subramani, Oyvind Tafjord, Pete Walsh, Luke Zettlemoyer, Noah A. Smith, Hannaneh Hajishirzi, Iz Beltagy, Dirk Groeneveld, Jesse Dodge, and Kyle Lo. Dolma: an open corpus of three trillion tokens for language model pretraining research. *arXiv*, 2024.

Tim Tomov, Dominik Fuchsgruber, Tom Wollschläger, and Stephan Günnemann. Entropy is not enough: Uncertainty quantification for llms fails under aleatoric uncertainty. In *NeurIPS 2025 Workshop on SPIGM*, 2025.

Artem Vazhentsev, Vladislav Smirnov, David Li, Maxim Panov, Timothy Baldwin, and Artem Shelmanov. Uncertainty quantification for large language diffusion models. *arXiv*, 2026.

Jason Wei et al. SimpleQA: A simple, challenging, and highly-controllable factuality benchmark. In *Advances in Neural Information Processing Systems*, 2024.

Chengyue Wu, Hao Zhang, Shuchen Xue, Zhijian Liu, Shizhe Diao, Ligeng Zhu, Ping Luo, Song Han, and Enze Xie. Fast-dLLM: Training-free acceleration of diffusion llm by enabling kv cache and parallel decoding. *arXiv*, 2025.

Jiacheng Ye, Zhihui Xie, Lin Zheng, Jiahui Gao, Zirui Wu, Xin Jiang, Zhenguo Li, and Lingpeng Kong. Dream 7B: Diffusion large language models. *arXiv*, 2025.

Tong Zhang, Peixin Qin, Yang Deng, Chen Huang, Wenqiang Lei, Junhong Liu, Dingnan Jin, Hongru Liang, and Tat-Seng Chua. CLAMBER: A benchmark of identifying and clarifying ambiguous information needs in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, 2024.

Xin Zhang, Yili Wang, Yue Tan, Xin He, Yanyu Qian, Yixin Liu, Yi Chang, Shirui Pan, and Xin Wang. Demts: Denoising trajectories as multivariate time series for hallucination detection in diffusion language models. *arXiv*, 2026.

Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. Wildchat: 1m chatGPT interaction logs in the wild. In *The Twelfth International Conference on Learning Representations*, 2024.

A SOUL DATASET

A.1 LABELING QUESTIONNAIRE AND CONVENTIONS

Every prompt is put through the same five questions, in the same order, and two follow-ups that are asked only when the question above them was answered in a particular way. A label is never skipped because an earlier answer made it unlikely. Labelers see the prompt, its corpus and, where they exist, the corpus's gold answers. The corpus's own annotation is withheld, as is any model output. Answers are recorded as a trail. The source of uncertainty is derived from that trail in code, so a change of mapping re-derives every label without re-annotation.

The questionnaire

Q1 (TASK). *From the prompt alone, can you name the ONE action the user is asking for?*
Polarity: **no** sets TASK.

Q2 (CONTEXT). *Taking the task as identified, is a detail the USER must supply still missing?*
Polarity: **yes** sets CONTEXT.

Q3 (ALEATORIC). *Even after looking everything up AND asking the user what they meant, would several materially different answers still be correct?*
Polarity: **yes** sets ALEATORIC.

Q4 (UNRESOLVED). *Is what is needed to answer missing from the world today?*
Polarity: **yes** opens Q4a.

Q4a (asked only if Q4 = yes). *Is it missing because the event has not happened, the problem is open, or the record is lost?*
Polarity: **yes** sets UNRESOLVED; **no** opens Q4b.

Q4b (asked only if Q4a = no). *Is the prompt's presupposition false?*
Polarity: **yes** drops the item as a false premise; **no** leaves UNRESOLVED unset.

Q5 (*Additional label*: SET). *As posed, is more than one materially different answer correct?*
Polarity: **yes** sets SET.

Conventions. Prompts with a *salient default* are not labeled with CONTEXT, as a competent answerer would take one reading without asking (nothing is missing), e.g., "What is the plot of *The Conjuring*?" carries no label while "When did the first Harry Potter movie come out?" is labeled with CONTEXT due to the release date differing by country. *Time-relative* prompts are anchored at the time of asking, thus backward-looking words ("last", "current") are a lookup, while forward-looking ones ("next", "new") leave which event is meant open and are labeled with CONTEXT. *Pick-some* requests ("name three sports teams in Boston") are considered ALEATORIC, whereas a request for the complete set is one answer.

Example 1 CLAMBER/BENCHMARK/00690
"When did the original toy story come out?"

Q1 (TASK)	yes	the action is nameable: answer a factual question
Q2 (CONTEXT)	yes	the release date differs by country, and no country is the default
Q3 (ALEATORIC)	no	naming the country settles it, and hence the gap is the user's
Q4 (UNRESOLVED)	no	the dates are on record
Q5 (SET)	yes	the readings give materially different dates

Labels: CONTEXT, SET.

Example 2 KUQ/KNOWNS_UNKNOWNS/01582
"How to predict unanticipated transportation disasters accurately?"

Q1 (TASK)	yes	the action is nameable: explain how to do something
Q2 (CONTEXT)	no	no detail the user holds would settle it
Q3 (ALEATORIC)	no	the multiplicity is not in the question; the method is simply not known
Q4 (UNRESOLVED)	yes	what is needed is missing from the world
Q4a	yes	the problem is open
Q5 (SET)	no	competing proposals are guesses at an unsettled answer

Labels: UNRESOLVED.

This item also illustrates adjudication. A second labeler answered Q3 = yes and Q5 = yes, which the adjudication process rejects: UNRESOLVED $\Rightarrow \neg$ ALEATORIC. The item went to adjudication and was resolved to UNRESOLVED alone, as it is an open problem, which has one answer that nobody holds yet.

A.2 CORPUS STATISTICS

The corpora differ in what they contribute as well as in how much: SimpleQA and HaluEval are almost entirely CONTROL and sit at the rare end of ϱ, while KUQ and WildChat carry a source on about half of their prompts. The authored set is the only one that carries every source, and it is balanced by construction: within each imitated style it holds between 208 and 257 prompts of each source, and 57% of it lies above the rarity cut.

A prompt may carry more than one source, and 73 of 16,224 do: 59 carry CONTEXT together with ALEATORIC, 12 CONTEXT with UNRESOLVED, and 2 TASK with CONTEXT. Prompts carrying a source are answered at several times the length of CONTROL prompts by every model (Table 4). On DiffusionGemma the median CONTROL answer is 73 tokens against 497 to 512 for TASK, UNRESOLVED and ALEATORIC, and about half of the latter reach the 512-token limit. The signals follow that length, with a median $|\rho|$ of up to 0.57 across a model's catalog. A classifier could therefore separate the sources by reading length alone, which is what the corpus $\times$ length stratification of Section C.2 removes and what the length-only baseline measures. Two models are exceptions: Dream answers in about a dozen tokens whatever the prompt, and Nemotron's signals track length weakly ($|\rho| = 0.11$), so for those two models the stratification changes less.

Table 4: **Answer length by source.** Median answer tokens per model and source, with the share of answers that reach the 512-token limit in gray; those answers are a length floor and form a bin of their own. The last column is the median $|\rho|$ (Spearman) between signals and answer length.

	CONTR.	TASK	CONT.	UNRES.	ALEAT.	median $\lvert\rho\rvert$
DiffusionGemma	73 (9%)	497 (48%)	148 (11%)	510 (50%)	512 (59%)	0.53
Gemma 4	85 (12%)	512 (55%)	176 (15%)	512 (61%)	512 (72%)	0.50
Dream	10 (0%)	12 (0%)	12 (0%)	13 (0%)	14 (0%)	0.33
Nemotron	196 (19%)	512 (74%)	240 (29%)	512 (80%)	512 (69%)	0.11
LLaDA2.1	72 (7%)	226 (26%)	91 (9%)	256 (31%)	281 (34%)	0.57
Qwen2.5	85 (3%)	246 (13%)	94 (3%)	244 (9%)	313 (16%)	0.50
Ministral 3	111 (10%)	512 (53%)	139 (12%)	512 (56%)	512 (63%)	0.09

A.3 RARITY SCORE

The score counts entity-like spans of the prompt in an open pre-training corpus, without linking them to a knowledge base. We take spaCy noun chunks without determiners and possessives, named-entity spans, and numeric tokens of at least four characters; contiguous capitalized runs with link words allowed inside (*Under the Sign of Hell*); and, for lowercased corpora such as AmbigQA, spans from a title-cased reparse that the index contains predominantly capitalized. Spans shorter than three characters and a short list of generic nouns are dropped. Each span is counted exactly in an infini-gram index of Dolma v1.7 (Liu et al., 2025; Soldaini et al., 2024), taking the larger count of the span as written and of its capitalized form. The score is $\varrho = -\log(1 + \min_s \mathrm{count}(s))$ over

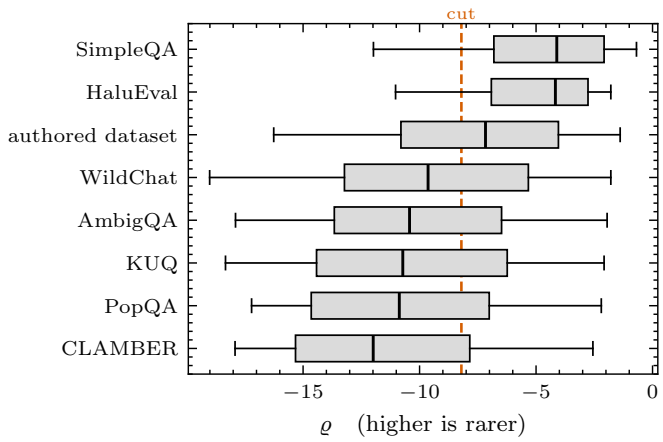


Figure 4: **Rarity by corpus.** Distribution of ϱ within each corpus, ordered by median. Box: interquartile range and median; whiskers: 5th to 95th percentile. The dashed line is the cut at $\varrho = -8.205$, the median of the pool, which is used only to balance the anchors of the authored set.

the spans with a non-zero count, so a prompt is as rare as its rarest span and a higher ϱ means rarer. A prompt with no occurring span falls back to its bare nouns.

Figure 4 shows where each corpus falls. The corpora separate as their construction implies: SimpleQA and HaluEval were built from rare facts and sit at the rare end, while CLAMBER and PopQA spread across the axis. The dashed line is the median of the eligible pool, $\varrho = -8.205$, frozen before any model was run. This cut is applied once in the paper. When we author prompts (Section 3.2), each one rewrites a fact taken from a CONTROL item of the corpus it imitates, and we draw half of those anchors from either side of the cut. Without it, an authored source would inherit the rarity of whatever facts happened to be convenient, and rarity would then correlate with the source we wrote. No experiment conditions on the cut, and every result in Section 5 treats ϱ as a continuous covariate or not at all.

A.4 RARITY ABLATION

Whether a model can retrieve a fact depends on what that model was trained on, so the same prompt is rare for one model and common for another, and a blind labeler cannot decide it. We therefore measure a proxy instead of labeling the source. The proxy follows the finding that the frequency of a fact in the pre-training corpus predicts whether a model recalls it (Kandpal et al., 2023; Mallen et al., 2023), which is also what makes PopQA a rarity benchmark. As we have no access to the training data of seven models, we count in an open corpus and treat the count as a stand-in for all of them. This section tests that assumption and finds it wanting, which is why ϱ appears in this appendix rather than in the results.

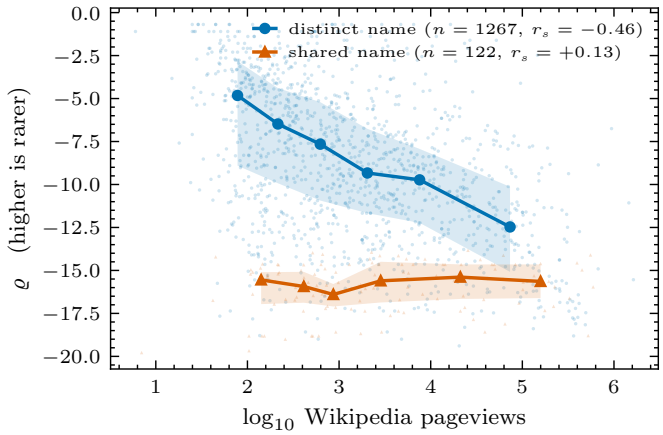


Figure 5: **Rarity score.** Within PopQA templates, where the wording is fixed and only the entity changes, ϱ is plotted against the entity's Wikipedia pageviews. Points are prompts, lines join the medians of six pageview bins, and bands are the interquartile range within a bin. For entities with a distinct name the two measures agree ($r_s = -0.46$); for the 8.8% whose name is shared with other entities ϱ is flat and slightly inverted ($r_s = +0.13$), as the index counts the name and not the entity.

A PopQA template fixes the wording and varies only the entity, so within one template the prompts differ in the entity. Figure 5 uses this aspect to compare ϱ with an external measure of how well known its Wikipedia pageviews are. For the 1,267 prompts whose entity name is distinct, the two agree: ϱ falls from -5 to -12.5 across the pageview range, at Spearman $r_s = -0.46$. For the 122 prompts (8.8%) whose entity shares its name with other entities, ϱ is flat at about -15.5 and slightly the wrong way ($r_s = +0.13$). This is the failure mode of counting strings: *Scooter Barry* is a rare person with a common name, and hence the score reports a well-known string where a rarity measure should report an obscure person. Accordingly, the string count cannot see the difference, and 9% of PopQA already matters for our analysis.

The acceptance test is whether ϱ predicts what a model knows. Across the seven models and the 6,721 prompts that carry a gold answer, ϱ correlates with answering correctly at -0.05 to $+0.02$ within corpus $\times$ length bins, while pageviews on the same PopQA prompts reach $+0.15$ to $+0.34$. The trajectory tells the same story from the other side. Table 5 gives the rank correlation of every signal with ϱ for every model, again within bins. Of the 139 signal-model pairs, 124 fall below $|\rho| = 0.10$, the median is 0.02 and the largest anywhere is 0.15. The exceptions are not a discovery: on Gemma 4 the whole catalog sits near $+0.11$, which is one quantity counted eleven times (Section D.1), and LLaDA2.1 has five entries between 0.12 and 0.15, all of which read how much the answer changes during denoising. Nothing in the denoising trajectory tracks the rarity of the prompt's rarest string.

We therefore report ϱ as the corpus frequency of a prompt's rarest string, and hence not as evidence of what a model knows. RARITY is sound enough to balance the authored anchors, and it is not reliable enough to be a fifth label or a regression target. Two routes would fix it, and both are outside the scope of this paper. A measured per-model score, obtained by asking each model the same fact repeatedly and recording how often it answers correctly, would make rarity model-side as its definition requires, at the cost of being a different quantity for every model and no longer a

property of the benchmark. Alternatively, linking spans to a knowledge base and counting entities rather than strings would remove the homonym failure of Figure 5, at the cost of an entity linker in the labeling path. Until one of those exists, the epistemic axis of SOUL rests on UNRESOLVED, which is labeled, and not on ϱ, which is only counted.

Table 5: **Rarity scores for all signals.** Rank correlation of each signal with ϱ, counted only between prompts of the same corpus and answer-length bin, so the corpora's different rarity profiles (Figure 4) cannot produce it. Each dLLM is followed by its tokenizer-matched AR–LLM. **Bold** marks $|\rho| \geq 0.10$; a dash marks a signal the model does not define. The last row is the largest $|\rho|$ anywhere in the model's catalog.

	DiffusionGemma	Gemma 4	Dream	Qwen2.5	Nemotron	Ministral 3	LLaDA2.1
from the literature							
commit_nll	–	+.**11**	−.03	−.00	+.00	+.04	−.00
dissim_avg	−.02	–	−.03	–	−.01	–	+.**12**
dissim_prog	−.01	–	−.02	–	+.01	–	+.**15**
flip_count	−.06	–	−.03	–	−.02	–	+.**12**
nfe	−.00	−.01	–	+.02	−.05	+.01	−.00
traj_entropy	−.01	+.**11**	−.00	−.00	−.02	+.04	+.**12**
traj_nll	+.02	−.**11**	+.01	+.00	+.03	−.04	−.**12**
level							
traj_renyi2	−.01	+.**11**	−.01	−.00	−.03	+.04	+.**11**
traj_tail_gap	+.01	+.**10**	+.01	+.00	+.01	+.03	+.**11**
final state							
final_entropy	+.00	+.**11**	−.02	−.00	+.00	+.04	−.00
final_entropy_q90	−.02	+.**11**	−.01	+.02	+.01	+.04	−.02
final_nll_max	−.00	+.06	−.01	+.05	+.04	+.05	−.01
shape over τ							
ent_auc_norm	−.00	–	+.01	–	−.02	–	+.03
ent_convexity	+.00	–	−.02	–	+.03	–	−.00
ent_drop_frac	+.00	–	+.02	–	−.05	–	−.05
ent_end	–	–	−.02	–	+.05	–	+.06
ent_half_time	−.01	–	−.02	–	−.03	–	−.01
ent_rise_frac	−.02	–	−.02	–	+.01	–	+.07
ent_start	−.03	–	+.01	–	−.01	–	−.01
ent_steepest_time	−.02	–	−.03	–	−.01	–	−.02
dynamics							
accept_half_time	−.02	–	−.00	–	−.04	–	−.03
flip_rate	−.05	–	−.03	–	−.00	–	+.**10**
late_flip_frac	+.02	–	+.04	–	−.02	–	+.05
settle_time	−.02	–	−.02	–	−.02	–	+.02
settle_time_q90	−.01	–	+.00	–	−.00	–	+.03
unaccept_rate	−.01	–	–	–	–	–	–
position							
pos_entropy_std	+.01	+.10	+.04	+.05	+.01	+.04	+.06
pos_entropy_top10	+.03	−.09	+.05	+.06	+.03	−.01	−.05
$\max_s \lvert\rho\rvert$	.06	.11	.05	.06	.05	.05	.15

B TRAJECTORY SIGNALS AND PROBE

Every signal of Section 4.2 is a chain of aggregations of the posterior cube, and the catalog groups them by what the chain reads. We distinguish between signals taken *from the literature*, the *level* of the posterior, its *final state*, its *shape* over τ, the *dynamics* of commitment, and the spread across *positions*. Depending on the generation mechanism, a dLLM defines 26 or 27 of them and an AR–LLM defines 11, as a single posterior vector per position leaves the shape and dynamics families undefined. Section D.1 reads each signal on its own against every source.

Table 6 lists every signal of the catalog with its aggregation chain. The plane column names the $T \times P$ plane of Section 4.2, which is the collapse of the token axis: the log confidence $\ell[t, p]$, the entropy $H[t, p]$, the Rényi-2 entropy $H_2[t, p]$, the top token $y[t, p]$ and the held state $h[t, p]$. The next two columns name the two remaining collapses in the order in which they are applied. For the scalar families the step axis is collapsed first and the position axis second, each with its own operator, so FLIP_COUNT sums flips over steps and averages over positions, which counts changes of mind per position rather than per step. Three operators read a single step instead of collapsing the axis: commit_t reads the cell at the step t_p at which position p locks in, last_t reads the normalized time of the last argmax flip, and $|T|$ counts the steps. The shape family reverses the order. Each position's window is resampled onto normalized time $\tau \in [0, 1]$, the curves are averaged over positions into $H(\tau)$, and the signal is a functional of that single curve, such as its value at $\tau = 0$ or the time at which it has dropped halfway. Two signals pool all valid cells of a plane at once, marked as $\mathrm{mean}_{t,p}$. An AR–LLM outputs one posterior per position, so its plane has a single row per position; the shape and dynamics families are undefined for it, commit_t reads that row, and COMMIT_NLL equals $-$TRAJ_NLL. Signals that read the commit step exist only for masked dLLMs, as DiffusionGemma rewrites tokens rather than committing them.

B.1 PROBE ARCHITECTURE AND TRAINING

Input. Each generation is mapped to one $5 \times 21 \times 32$ array. The five channels are the negative log confidence $-\ell[t, p] = -\log \max_v \mathcal{P}[t, p, v]$, the Shannon entropy $H[t, p]$, the Rényi-2 entropy $H_2[t, p] = -\log \sum_v \mathcal{P}[t, p, v]^2$, the held indicator $h[t, p]$ and the argmax flip $\mathbb{1}[y[t, p] \neq y[t-1, p]]$. We exclude the argmax token itself, as it is categorical over a vocabulary of tokens and a convolution over token IDs would be arithmetic on arbitrary integers.

Normalization. Time follows the convention of Section 4.2. We rescale each column's window to $\tau \in [0, 1]$, interpolate onto 21 points and drop columns whose window is a single step. The position is resampled onto 32 bins. We average when a generation has more than 32 answer columns and interpolate linearly when it has fewer. This removes answer length from the array. Length spans 10 to 512 tokens and predicts the corpus almost deterministically and hence an array that preserved it would give the probe the corpus. The length-only baseline of Section C.2 measures that quantity explicitly instead. For an AR–LLM each position is decided in one step, thus its array is constant along τ and its held and flip channels are zero.

Architecture. The network has two convolutional blocks. Each block is a 3×3 convolution with padding 1, batch normalization, ReLU and 2×2 max pooling, and the width grows from 5 to 16 to 32 channels. A global pooling layer then concatenates the mean and the maximum over (τ, p), followed by dropout of 0.3 and a linear softmax head. The mean measures how much of a pattern a generation contains and the maximum whether it occurs anywhere, so a single filter can serve a diffuse and a localized signature. The network has 5,797 parameters for the five-way task and 5,602 for a binary task.

Training. We train with Adam at learning rate 3×10^{-3}, weight decay 10^{-4}, batch size 64 and at most 60 epochs. The cross-entropy loss is weighted by inverse class frequency. We hold out 20% of each training fold as an inner validation split, stop after 10 epochs without improvement and restore the weights of the best epoch. The test fold therefore affects neither the weights nor the stopping point. Channel means and standard deviations are computed on training rows only. The probe uses the classes, folds and metrics of Section C, so a probe result and a signal result differ only in the input.

Table 6: **Signal catalog with chained aggregations.** Every signal is a plane of the posterior cube followed by two collapses, applied in the order of the columns. $\ell = \log \max_v \mathcal{P}$ is the log confidence, H the entropy, $H_2 = -\log \sum_v \mathcal{P}^2$ the Rényi-2 entropy, y the top token and h the held state. commit_t reads the step at which a position locks in, last_t the normalized time of its last argmax flip, and $\mathrm{mean}_{t,p}$ pools all valid cells. $H(\tau)$ and $h(\tau)$ are the entropy and held planes resampled onto normalized time τ and averaged over positions. The last column gives some intuition.

Signal	Plane	First collapse	Second collapse	Intuition
from the literature				
NFE	–	$\lvert T \rvert$	–	more denoising steps used
TRAJ_NLL	ℓ	mean_t	mean_p	more confident top token over the trajectory (lower is more uncertain)
TRAJ_ENTROPY	H	mean_t	mean_p	more undecided while writing the answer
COMMIT_NLL	$-\ell$	commit_t	mean_p	each token more of a guess when it locks in
FLIP_COUNT	$\mathbb{1}[y_t \neq y_{t-1}]$	sum_t	mean_p	more changes of mind per position
DISSIM_AVG	draft $\tilde{y}_t$	$\mathrm{mean}_t\, D(\tilde{y}_t, y)$	–	intermediate drafts further from the final answer
DISSIM_PROG	draft $\tilde{y}_t$	$\mathrm{mean}_t\, \frac{t}{T}\, D(\tilde{y}_t, y)$	–	late drafts still far from the final answer
level				
TRAJ_RENYI2	H_2	mean_t	mean_p	mass less concentrated on the top tokens
TRAJ_TAIL_GAP	$H - H_2$	mean_t	mean_p	heavier tail of small-probability tokens
final state				
FINAL_ENTROPY	H	commit_t	mean_p	positions lock in while still undecided
FINAL_ENTROPY_Q90	H	commit_t	$\mathrm{q90}_p$	the least settled positions lock in undecided
FINAL_NLL_MAX	$-\ell$	commit_t	$\max_p$	the weakest committed token is weaker
shape over τ				
ENT_START	H	resample to τ, $\mathrm{mean}_p \to H(\tau)$	$H(0)$	more undecided when denoising starts
ENT_END	H	same	$H(1)$	more undecided when denoising ends
ENT_DROP_FRAC	H	same	$\frac{H(0) - H(1)}{H(0)}$	larger share of the initial entropy resolved
ENT_AUC_NORM	H	same	$\int_0^1 \frac{H(\tau)}{H(0)}\, d\tau$	uncertainty persists longer
ENT_HALF_TIME	H	same	τ at which $H(\tau)$ crosses the midpoint of $H(0)$ and $H(1)$	resolves later
ENT_CONVEXITY	H	same	$\int_0^1 \frac{\mathrm{chord}(\tau) - H(\tau)}{H(0) - H(1)}\, d\tau$	collapse early (positive) rather than late (negative)
ENT_STEEPEST_TIME	H	same	$\arg\max_\tau\, (-dH/d\tau)$	the steepest drop comes later
ENT_RISE_FRAC	$\mathbb{1}[H_{t+1} > H_t]$	$\mathrm{mean}_{t,p}$	–	entropy goes back up more often
dynamics				
FLIP_RATE	$\mathbb{1}[y_t \neq y_{t-1}]$	$\mathrm{mean}_{t,p}$	–	more changes of mind per step
SETTLE_TIME	$\mathbb{1}[y_t \neq y_{t-1}]$	last_t	mean_p	the top token settles later
SETTLE_TIME_Q90	$\mathbb{1}[y_t \neq y_{t-1}]$	last_t	$\mathrm{q90}_p$	the slowest positions settle later
LATE_FLIP_FRAC	$\mathbb{1}[y_t \neq y_{t-1}]$	share of flips at $\tau > 0.5$	–	changes of mind concentrated in the second half
ACCEPT_HALF_TIME	h	resample to τ, $\mathrm{mean}_p \to h(\tau)$	τ at which $h(\tau)$ crosses 0.5	half of the answer is kept later
UNACCEPTRATE	$1[\neg h_t]$	$\mathrm{mean}_{t,p}$	–	sampler rejects the proposed token more often
position				
POS_ENTROPY_STD	H	mean_t	std_p	uncertainty uneven across the answer
POS_ENTROPY_TOP10	H	mean_t	share of sum_p in the top 10% of positions	uncertainty concentrated in a few tokens

C Evaluation protocol

C.1 Classification of source of uncertainty

One classifier per model assigns each prompt to one of the four sources or CONTROL. For every model we train three classifiers (logistic regression, gradient-boosted trees, random forest) on three feature sets (answer length only, signals, signals with normalized-time curves) under 5-fold cross-validation. Predictions are the argmax of the out-of-fold probabilities. We report the F1 of each class and the unweighted mean over the five classes, both as the mean over folds with its standard deviation, and select the configuration with the highest F1 across the five classes. Two naive baselines set the floor. A classifier that always predicts CONTROL reaches an F1 of 0.13 across the five classes, and one that draws each class with its frequency reaches 0.20. Each class is an exact combination of labels: CONTROL is a prompt with no label at all, TASK and UNRESOLVED carry their source alone, and CONTEXT and ALEATORIC carry theirs together with SET, the set-valued marker of Section A.1. A prompt therefore enters the five-way task only if its labels match one class, which excludes 116 of the 16,224 prompts and leaves 16,108. 73 carry two sources (Section A.2), 37 carry SET without any source and are therefore not CONTROL, and 6 carry a source in a combination that no class covers. The excluded prompts remain in the dataset and in the label counts of Table 2.

C.2 Control for corpus and answer length

Corpus and answer length both correlate with the labels (Section A.2), and the F1 of Table 3 pools over both. We control for them in two ways. The univariate analysis of Section D.1 compares only prompts of the same corpus and answer-length bin, using the stratified AUROC defined below, and the length-only classifier of Section D.3 measures what the confounders achieve on their own.

Stratified AUROC. Answer length is binned into deciles of the answers that finished within the 512-token limit; answers that hit the limit form a stratum of their own, as their length is a floor rather than a length. Let s index the bins formed by corpus $\times$ length stratum, computed for each model on its own answers, and let n_s^+ and n_s^- count the labeled and CONTROL prompts in bin s. We compute AUROC_s within each bin and pool

$$\mathrm{AUROC}_{\mathrm{w}} = \sum_s w_s \,\mathrm{AUROC}_s, \qquad w_s \propto n_s^+ n_s^-, \tag{2}$$

so that each bin counts in proportion to the number of labeled-CONTROL pairs it contributes. Bins with fewer than two labeled or two CONTROL prompts are dropped. Section D.1 reports this AUROC for every signal, source and model.

Intervals and tests. Each bin receives a 95% DeLong interval, and the pooled interval follows from the pooled standard error $\sqrt{\sum_s w_s^2 \sigma_s^2}$, with σ_s the DeLong standard error of bin s. For every model and task we train the classifiers and feature sets of Section C.1 and report the configuration with the highest pooled AUROC; the length-only baseline is the best configuration on answer length alone. To compare a classifier with its baseline we draw 1,000 paired bootstrap samples with Poisson(1) weights per prompt, shared between the two classifiers, and compute a two-sided p-value from the bootstrap standard error of the paired difference under a normal reference. Holm correction is applied over all task $\times$ model comparisons at once.

Table 7: **DiffusionGemma.** Every signal of the catalog against each source of uncertainty (CONTEXT, ALEATORIC, UNRESOLVED and TASK, abbreviated in the header), and CONTROL against every labeled prompt at once. The left half counts only pairs inside one corpus and one answer-length bin; the right half pools every pair. **Bold** marks an AUROC whose 95% interval excludes chance.

	within corpus × length					pooled				
	CONT.	ALEAT.	UNRES.	TASK	CTRL.	CONT.	ALEAT.	UNRES.	TASK	CTRL.
from the literature										
dissim_avg	**.53**	**.46**	**.55**	**.68**	**.47**	**.64**	**.73**	**.78**	**.82**	**.28**
dissim_prog	**.52**	**.45**	**.53**	**.65**	**.48**	**.61**	**.80**	**.81**	**.85**	**.26**
flip_count	**.55**	**.47**	**.58**	**.64**	**.46**	**.66**	**.79**	**.82**	**.85**	**.25**
nfe	.50	**.54**	**.56**	**.69**	**.46**	**.62**	**.87**	**.84**	**.89**	**.23**
traj_entropy	**.52**	.50	**.62**	**.72**	**.44**	**.63**	**.78**	**.84**	**.87**	**.25**
traj_nll	**.46**	**.55**	**.39**	**.31**	**.55**	**.36**	**.21**	**.15**	**.13**	**.76**
level										
traj_renyi2	**.54**	**.47**	**.61**	**.70**	**.45**	**.64**	**.79**	**.85**	**.87**	**.25**
traj_tail_gap	.49	**.57**	**.62**	**.75**	**.43**	**.60**	**.76**	**.79**	**.83**	**.29**
final state										
final_entropy	.50	**.53**	.51	**.45**	.50	**.44**	**.34**	**.35**	**.33**	**.62**
final_entropy_q90	.49	.52	.51	**.41**	.50	**.44**	**.47**	**.48**	**.44**	**.54**
final_nll_max	**.52**	.53	.51	.49	.49	**.57**	**.60**	**.58**	**.61**	**.41**
shape over τ										
ent_auc_norm	.50	.48	**.55**	**.66**	**.48**	**.54**	**.58**	**.63**	**.69**	**.41**
ent_convexity	.50	.52	**.45**	**.34**	**.52**	**.45**	**.41**	**.36**	**.30**	**.60**
ent_drop_frac	.51	.48	.52	**.59**	.49	**.60**	**.73**	**.73**	**.76**	**.32**
ent_half_time	.51	.48	**.54**	**.66**	**.48**	**.57**	**.61**	**.65**	**.71**	**.38**
ent_rise_frac	.49	.51	.50	**.64**	.49	**.59**	**.69**	**.68**	**.75**	**.34**
ent_start	**.53**	**.53**	**.73**	**.72**	**.41**	**.64**	**.85**	**.89**	**.92**	**.21**
ent_steepest_time	**.52**	.48	.51	**.60**	.49	**.58**	**.64**	**.66**	**.70**	**.37**
dynamics										
accept_half_time	**.54**	**.46**	**.54**	**.59**	**.48**	**.61**	**.64**	**.70**	**.71**	**.35**
flip_rate	**.55**	**.44**	.50	**.45**	.50	**.60**	**.55**	**.62**	**.54**	**.42**
late_flip_frac	**.48**	.48	**.53**	**.64**	.49	**.48**	.50	**.55**	**.62**	**.48**
settle_time	**.53**	**.46**	**.55**	**.67**	**.47**	**.63**	**.71**	**.77**	**.81**	**.29**
settle_time_q90	.49	.48	.53	**.63**	.49	**.56**	**.61**	**.65**	**.70**	**.39**
unaccept_rate	**.46**	.51	.48	.47	**.52**	**.55**	**.71**	**.70**	**.70**	**.36**
position										
pos_entropy_std	**.47**	.52	**.60**	**.67**	**.47**	**.54**	**.69**	**.74**	**.78**	**.34**
pos_entropy_top10	**.43**	**.55**	.48	**.41**	**.53**	**.33**	**.33**	**.28**	**.27**	**.69**

D ADDITIONAL RESULTS

D.1 UNIVARIATE SIGNALS

The tables below read each signal of Section B on its own as a score, with the stratified AUROC of Section C.2. This is the univariate counterpart of Table 3, which allows us to answer different questions. Three readings cut across all seven tables. An AUROC below 0.5 is a direction: TRAJ_NLL reaches .20 on TASK for Nemotron, which separates as sharply as .80 would. Single signals do not identify CONTEXT in any model, as across all signal-model pairs the largest departure from chance is 0.073. The rank correlations with ϱ in Table 5 are flat everywhere with $|\rho| \leq 0.15$ within corpus $\times$ length bin.

DiffusionGemma spreads its separation across families: ENT_START reaches .73 on UNRESOLVED and TRAJ_TAIL_GAP .75 on TASK, while four signals sit below .42 and carry the same information inverted. NFE is the clearest example of why the two halves are printed side by side. It results in an AUROC of .69 on TASK within corpus $\times$ length bin against .89 pooled, because the number of function evaluations is largely a readout of how long the answer is.

Table 8: **Dream.** Every signal of the catalog against each source of uncertainty (CONTEXT, ALEATORIC, UNRESOLVED and TASK, abbreviated in the header), and CONTROL against every labeled prompt at once. The left half counts only pairs inside one corpus and one answer-length bin; the right half pools every pair. **Bold** marks an AUROC whose 95% interval excludes chance.

	within corpus × length					pooled				
	CONT.	ALEAT.	UNRES.	TASK	CTRL.	CONT.	ALEAT.	UNRES.	TASK	CTRL.
from the literature										
commit_nll	**.55**	**.56**	.50	**.43**	**.47**	.51	**.55**	**.52**	**.45**	**.49**
dissim_avg	**.53**	**.57**	**.54**	**.45**	**.47**	**.53**	**.59**	**.56**	**.47**	**.46**
dissim_prog	**.52**	**.57**	**.54**	**.47**	**.47**	**.53**	**.59**	**.55**	**.47**	**.46**
flip_count	**.56**	**.58**	**.61**	.50	**.44**	**.56**	**.63**	**.63**	.51	**.42**
traj_entropy	.51	**.57**	**.46**	**.44**	.49	**.48**	**.55**	**.48**	**.47**	.51
traj_nll	**.47**	**.43**	.49	**.56**	**.52**	.50	**.43**	**.47**	**.53**	**.52**
level										
traj_renyi2	**.53**	**.58**	.50	**.44**	**.48**	.50	**.56**	**.52**	**.47**	**.49**
traj_tail_gap	.50	**.57**	**.43**	**.44**	.51	**.46**	**.53**	**.44**	**.47**	**.53**
final state										
final_entropy	**.53**	**.58**	**.47**	**.45**	**.48**	.49	**.55**	.49	**.48**	.50
final_entropy_q90	**.52**	**.55**	**.45**	**.43**	.50	.49	**.53**	**.45**	**.44**	**.52**
final_nll_max	**.54**	**.53**	.51	**.45**	**.48**	**.52**	**.55**	.51	**.45**	**.49**
shape over τ										
ent_auc_norm	**.47**	.51	**.41**	**.40**	**.54**	**.45**	**.47**	**.43**	**.44**	**.55**
ent_convexity	**.52**	.49	**.59**	**.59**	**.46**	**.55**	**.52**	**.58**	**.54**	**.45**
ent_drop_frac	.51	.51	.50	**.56**	**.48**	**.51**	**.55**	**.52**	**.62**	**.46**
ent_end	**.52**	**.54**	**.54**	**.47**	**.48**	**.51**	**.52**	**.53**	**.43**	.49
ent_half_time	.51	**.55**	.48	**.46**	.49	**.48**	**.52**	**.48**	.50	.50
ent_rise_frac	**.52**	**.57**	**.56**	**.45**	**.47**	.50	**.52**	.51	**.41**	**.51**
ent_start	**.57**	**.69**	**.63**	**.62**	**.38**	**.56**	**.70**	**.64**	**.60**	**.38**
ent_steepest_time	.50	**.53**	.49	**.47**	.50	**.48**	.50	**.48**	.51	**.51**
dynamics										
accept_half_time	.51	.50	.51	.50	.49	.50	.50	.49	.51	.50
flip_rate	**.56**	**.58**	**.61**	.50	**.44**	**.56**	**.63**	**.63**	.51	**.42**
late_flip_frac	.49	.50	.48	**.45**	**.52**	.49	.50	**.48**	**.43**	**.52**
settle_time	**.53**	**.58**	**.57**	**.47**	**.47**	**.53**	**.60**	**.58**	**.48**	**.45**
settle_time_q90	**.52**	**.54**	**.55**	**.47**	**.48**	**.54**	**.59**	**.59**	.49	**.45**
position										
pos_entropy_std	**.46**	.49	**.38**	**.36**	**.56**	**.43**	**.44**	**.35**	**.33**	**.60**
pos_entropy_top10	**.45**	**.44**	**.38**	.51	**.56**	**.43**	**.39**	**.41**	.51	**.57**

Dream has the weakest signal catalog of the four dLLMs. ENT_START scores an AUROC of .69 on ALEATORIC, which is the only value anywhere in the table beyond .65. Moreover, the signals do not depart from chance by 0.20 on any source.

Nemotron is the opposite case, and the reason it leads Table 3. In total 21 of its 27 signals depart from chance by at least 0.20 on TASK alone, with FINAL_ENTROPY and COMMIT_NLL at .84; ENT_START at .79 on UNRESOLVED. Nonetheless, the CONTEXT column is the flattest of any model with no signal exceeding .53.

LLaDA2.1 sits between the two, with ENT_START at .73 on TASK. It is also the only model whose signals track rarity at all: DISSIMPROG at $+.15$, DISSIMAVG, FLIPCOUNT and TRAJENTROPY at $+.12$ and TRAJNLL at $-.12$ within corpus × length bin (Table 5). These are the largest $|\rho|$ in the catalog and they are still small.

The three tables of the AR–LLMs share a different structure compared to the dLLMs. As they have one posterior per position, the signals COMMIT_NLL, TRAJ_ENTROPY, TRAJ_RENYI2 and FINAL_ENTROPY are monotone transforms of the same quantity, and their AUROCs agree to within 0.017 in every column of all three tables. Thus, the 11 signals of an AR–LLM hold roughly four independent quantities.

Table 9: **Nemotron.** Every signal of the catalog against each source of uncertainty (CONTEXT, ALEATORIC, UNRESOLVED and TASK, abbreviated in the header), and CONTROL against every labeled prompt at once. The left half counts only pairs inside one corpus and one answer-length bin; the right half pools every pair. **Bold** marks an AUROC whose 95% interval excludes chance.

	within corpus × length					pooled				
	CONT.	ALEAT.	UNRES.	TASK	CTRL.	CONT.	ALEAT.	UNRES.	TASK	CTRL.
from the literature										
commit_nll	.50	**.75**	**.77**	**.84**	**.31**	**.49**	**.71**	**.70**	**.75**	**.37**
dissim_avg	**.48**	**.68**	**.76**	**.79**	**.35**	.49	**.69**	**.75**	**.74**	**.37**
dissim_prog	**.48**	**.66**	**.75**	**.76**	**.36**	**.49**	**.70**	**.78**	**.75**	**.36**
flip_count	.50	.52	**.66**	**.65**	**.43**	.51	.49	**.62**	**.57**	**.46**
nfe	**.52**	**.64**	**.66**	**.62**	**.40**	**.58**	**.83**	**.91**	**.88**	**.25**
traj_entropy	.49	**.71**	**.78**	**.83**	**.33**	**.48**	**.68**	**.71**	**.75**	**.38**
traj_nll	.51	**.35**	**.24**	**.20**	**.65**	.51	**.38**	**.29**	**.27**	**.60**
level										
traj_renyi2	.49	**.68**	**.78**	**.82**	**.34**	.49	**.65**	**.72**	**.74**	**.38**
traj_tail_gap	**.48**	**.75**	**.76**	**.84**	**.32**	**.46**	**.72**	**.67**	**.75**	**.39**
final state										
final_entropy	.50	**.76**	**.76**	**.84**	**.31**	**.49**	**.73**	**.69**	**.76**	**.37**
final_entropy_q90	.50	**.76**	**.72**	**.83**	**.32**	**.49**	**.74**	**.66**	**.76**	**.37**
final_nll_max	.50	**.61**	**.59**	**.69**	**.42**	.49	**.64**	**.62**	**.70**	**.41**
shape over τ										
ent_auc_norm	**.47**	**.68**	**.73**	**.80**	**.36**	**.47**	**.65**	**.66**	**.70**	**.41**
ent_convexity	**.52**	**.34**	**.28**	**.20**	**.64**	**.53**	**.38**	**.37**	**.31**	**.58**
ent_drop_frac	**.52**	**.31**	**.34**	**.38**	**.60**	**.51**	**.25**	**.28**	**.34**	**.62**
ent_end	**.48**	**.72**	**.71**	**.70**	**.36**	**.48**	**.76**	**.74**	**.70**	**.36**
ent_half_time	**.48**	**.66**	**.71**	**.79**	**.37**	**.47**	**.61**	**.62**	**.68**	**.43**
ent_rise_frac	**.48**	**.69**	**.73**	**.76**	**.36**	**.47**	**.68**	**.69**	**.68**	**.40**
ent_start	.50	**.72**	**.79**	**.83**	**.32**	.49	**.67**	**.72**	**.76**	**.38**
ent_steepest_time	**.47**	.52	**.53**	**.60**	.48	**.45**	**.47**	**.47**	.52	**.53**
dynamics										
accept_half_time	.50	**.65**	**.72**	**.78**	**.36**	.50	**.62**	**.67**	**.68**	**.41**
flip_rate	.50	.50	**.63**	**.64**	**.44**	.50	**.47**	**.60**	**.54**	**.48**
late_flip_frac	.49	**.72**	**.74**	**.79**	**.34**	**.48**	**.68**	**.68**	**.69**	**.40**
settle_time	.49	**.68**	**.76**	**.77**	**.35**	**.48**	**.64**	**.71**	**.69**	**.40**
settle_time_q90	.49	**.68**	**.74**	**.77**	**.35**	.49	**.66**	**.70**	**.69**	**.40**
position										
pos_entropy_std	**.48**	**.70**	**.75**	**.82**	**.34**	**.47**	**.62**	**.63**	**.70**	**.43**
pos_entropy_top10	**.52**	**.32**	**.27**	**.23**	**.64**	.50	**.30**	**.26**	**.25**	**.64**

Table 10: **LLaDA2.1.** Every signal of the catalog against each source of uncertainty (CONTEXT, ALEATORIC, UNRESOLVED and TASK, abbreviated in the header), and CONTROL against every labeled prompt at once. The left half counts only pairs inside one corpus and one answer-length bin; the right half pools every pair. **Bold** marks an AUROC whose 95% interval excludes chance.

	within corpus × length					pooled				
	CONT.	ALEAT.	UNRES.	TASK	CTRL.	CONT.	ALEAT.	UNRES.	TASK	CTRL.
from the literature										
commit_nll	**.53**	**.66**	**.62**	**.65**	**.42**	**.56**	**.81**	**.79**	**.77**	**.30**
dissim_avg	**.48**	**.47**	**.43**	**.40**	**.54**	**.40**	**.21**	**.17**	**.19**	**.72**
dissim_prog	**.48**	**.44**	**.40**	**.35**	**.56**	**.41**	**.20**	**.16**	**.18**	**.73**
flip_count	.51	.52	.53	.51	**.49**	**.53**	**.55**	**.62**	**.59**	**.44**
nfe	**.52**	**.56**	**.58**	.49	**.46**	**.60**	**.84**	**.84**	**.81**	**.26**
traj_entropy	**.52**	**.58**	**.58**	.52	**.46**	.50	**.38**	**.39**	**.40**	**.57**
traj_nll	**.48**	**.44**	**.44**	.51	**.53**	**.48**	**.61**	**.58**	**.58**	**.45**
level										
traj_renyi2	**.52**	**.56**	**.57**	.50	**.46**	**.51**	**.39**	**.41**	**.42**	**.55**
traj_tail_gap	.51	**.59**	**.58**	**.55**	**.46**	**.48**	**.38**	**.35**	**.39**	**.59**
final state										
final_entropy	.51	**.61**	**.59**	**.60**	**.41**	**.54**	**.65**	**.64**	**.59**	**.38**
final_entropy_q90	.53	**.61**	**.60**	**.62**	**.41**	**.55**	**.64**	**.63**	**.58**	**.39**
final_nll_max	.53	**.59**	**.58**	**.62**	**.42**	**.54**	**.59**	**.58**	**.55**	**.43**
shape over τ										
ent_auc_norm	**.47**	**.56**	**.54**	**.58**	.49	**.42**	**.38**	**.31**	**.34**	**.63**
ent_convexity	**.48**	**.41**	**.43**	**.39**	**.56**	**.48**	**.28**	**.29**	**.32**	**.64**
ent_drop_frac	**.56**	.49	.52	**.55**	**.46**	**.58**	**.75**	**.80**	**.76**	**.31**
ent_end	**.48**	**.57**	**.53**	**.54**	.49	**.45**	**.31**	**.27**	**.30**	**.65**
ent_half_time	**.52**	**.59**	**.58**	**.62**	**.44**	**.52**	**.73**	**.73**	**.70**	**.35**
ent_rise_frac	.50	**.58**	**.62**	**.64**	**.45**	**.52**	**.70**	**.76**	**.71**	**.36**
ent_start	**.56**	**.64**	**.69**	**.73**	**.39**	**.60**	**.83**	**.87**	**.86**	**.24**
ent_steepest_time	.50	**.52**	.52	**.53**	**.48**	.50	**.61**	**.61**	**.59**	**.43**
dynamics										
accept_half_time	**.55**	**.63**	**.63**	**.69**	**.40**	**.59**	**.80**	**.83**	**.81**	**.27**
flip_rate	.49	.50	.49	.52	.51	**.46**	**.41**	**.45**	**.42**	**.56**
late_flip_frac	**.47**	**.54**	**.53**	**.54**	.49	**.42**	**.34**	**.29**	**.30**	**.65**
settle_time	.50	**.58**	**.59**	**.65**	**.45**	**.48**	**.48**	**.47**	**.48**	**.52**
settle_time_q90	.49	**.58**	**.55**	**.59**	**.47**	**.45**	**.38**	**.34**	**.35**	**.61**
position										
pos_entropy_std	.50	**.60**	**.55**	**.59**	**.46**	**.46**	**.66**	**.56**	**.58**	**.45**
pos_entropy_top10	.50	.50	**.46**	**.54**	.51	**.46**	**.62**	**.53**	**.54**	**.47**

Table 11: **Gemma 4.** Every signal of the catalog against each source of uncertainty (CONTEXT, ALEATORIC, UNRESOLVED and TASK, abbreviated in the header), and CONTROL against every labeled prompt at once. The left half counts only pairs inside one corpus and one answer-length bin; the right half pools every pair. **Bold** marks an AUROC whose 95% interval excludes chance.

	within corpus × length					pooled				
	CONT.	ALEAT.	UNRES.	TASK	CTRL.	CONT.	ALEAT.	UNRES.	TASK	CTRL.
from the literature										
commit_nll	**.46**	**.66**	**.75**	**.74**	**.39**	**.54**	**.77**	**.82**	**.78**	**.31**
nfe	.51	.50	**.51**	**.52**	**.49**	**.64**	**.87**	**.86**	**.86**	**.22**
traj_entropy	**.45**	**.66**	**.75**	**.74**	**.39**	**.54**	**.76**	**.82**	**.78**	**.31**
traj_nll	**.54**	**.34**	**.25**	**.26**	**.61**	**.46**	**.23**	**.18**	**.22**	**.69**
level										
traj_renyi2	**.46**	**.66**	**.75**	**.74**	**.39**	**.54**	**.77**	**.82**	**.79**	**.31**
traj_tail_gap	**.45**	**.65**	**.74**	**.74**	**.39**	**.53**	**.75**	**.80**	**.77**	**.32**
final state										
final_entropy	**.45**	**.66**	**.75**	**.74**	**.39**	**.54**	**.76**	**.82**	**.78**	**.31**
final_entropy_q90	**.46**	**.67**	**.74**	**.75**	**.38**	**.55**	**.79**	**.82**	**.80**	**.30**
final_nll_max	.49	**.63**	**.61**	**.64**	**.43**	**.57**	**.78**	**.76**	**.77**	**.31**
position										
pos_entropy_std	**.46**	**.68**	**.72**	**.74**	**.39**	**.52**	**.73**	**.74**	**.73**	**.35**
pos_entropy_top10	**.53**	**.41**	**.26**	**.31**	**.59**	**.42**	**.24**	**.14**	**.19**	**.72**

Table 12: **Qwen2.5.** Every signal of the catalog against each source of uncertainty (CONTEXT, ALEATORIC, UNRESOLVED and TASK, abbreviated in the header), and CONTROL against every labeled prompt at once. The left half counts only pairs inside one corpus and one answer-length bin; the right half pools every pair. **Bold** marks an AUROC whose 95% interval excludes chance.

	within corpus × length					pooled				
	CONT.	ALEAT.	UNRES.	TASK	CTRL.	CONT.	ALEAT.	UNRES.	TASK	CTRL.
from the literature										
commit_nll	.50	**.68**	**.69**	**.67**	**.42**	**.52**	**.79**	**.81**	**.74**	**.32**
nfe	.51	.51	.52	.52	**.49**	**.56**	**.81**	**.86**	**.82**	**.28**
traj_entropy	**.49**	**.69**	**.68**	**.66**	**.42**	.51	**.78**	**.79**	**.73**	**.33**
traj_nll	.50	**.32**	**.31**	**.33**	**.58**	**.48**	**.21**	**.19**	**.26**	**.68**
level										
traj_renyi2	.49	**.68**	**.69**	**.67**	**.42**	**.51**	**.79**	**.81**	**.74**	**.32**
traj_tail_gap	**.47**	**.69**	**.64**	**.64**	**.44**	.49	**.77**	**.74**	**.70**	**.36**
final state										
final_entropy	**.49**	**.69**	**.68**	**.66**	**.42**	.51	**.78**	**.79**	**.73**	**.33**
final_entropy_q90	**.46**	**.69**	**.64**	**.64**	**.44**	**.48**	**.78**	**.75**	**.71**	**.35**
final_nll_max	**.46**	**.59**	.51	**.55**	.50	**.47**	**.71**	**.66**	**.68**	**.40**
position										
pos_entropy_std	**.44**	**.65**	**.55**	**.59**	**.48**	**.45**	**.70**	**.62**	**.63**	**.43**
pos_entropy_top10	**.43**	**.38**	**.27**	**.37**	**.61**	**.41**	**.25**	**.13**	**.26**	**.71**

Table 13: **Ministral 3.** Every signal of the catalog against each source of uncertainty (CONTEXT, ALEATORIC, UNRESOLVED and TASK, abbreviated in the header), and CONTROL against every labeled prompt at once. The left half counts only pairs inside one corpus and one answer-length bin; the right half pools every pair. **Bold** marks an AUROC whose 95% interval excludes chance.

	within corpus × length					pooled				
	CONT.	ALEAT.	UNRES.	TASK	CTRL.	CONT.	ALEAT.	UNRES.	TASK	CTRL.
from the literature										
`commit_nll`	**.52**	**.65**	**.66**	**.70**	**.40**	**.53**	**.64**	**.61**	**.65**	**.41**
`nfe`	.51	**.51**	**.52**	**.53**	**.49**	**.57**	**.87**	**.88**	**.87**	**.24**
`traj_entropy`	**.52**	**.66**	**.65**	**.71**	**.40**	**.52**	**.65**	**.61**	**.65**	**.41**
`traj_nll`	**.48**	**.35**	**.34**	**.30**	**.60**	**.47**	**.36**	**.39**	**.35**	**.59**
level										
`traj_renyi2`	**.52**	**.65**	**.66**	**.71**	**.40**	**.53**	**.64**	**.61**	**.65**	**.41**
`traj_tail_gap`	**.52**	**.66**	**.65**	**.72**	**.40**	**.52**	**.66**	**.59**	**.66**	**.41**
final state										
`final_entropy`	**.52**	**.66**	**.65**	**.71**	**.40**	**.52**	**.65**	**.61**	**.65**	**.41**
`final_entropy_q90`	**.52**	**.68**	**.63**	**.72**	**.40**	**.52**	**.71**	**.64**	**.70**	**.38**
`final_nll_max`	**.51**	**.63**	**.59**	**.68**	**.42**	**.52**	**.72**	**.68**	**.73**	**.36**
position										
`pos_entropy_std`	**.52**	**.67**	**.62**	**.71**	**.40**	.51	**.67**	**.59**	**.67**	**.41**
`pos_entropy_top10`	**.48**	**.46**	**.35**	**.43**	**.56**	**.46**	**.46**	**.35**	**.44**	**.57**

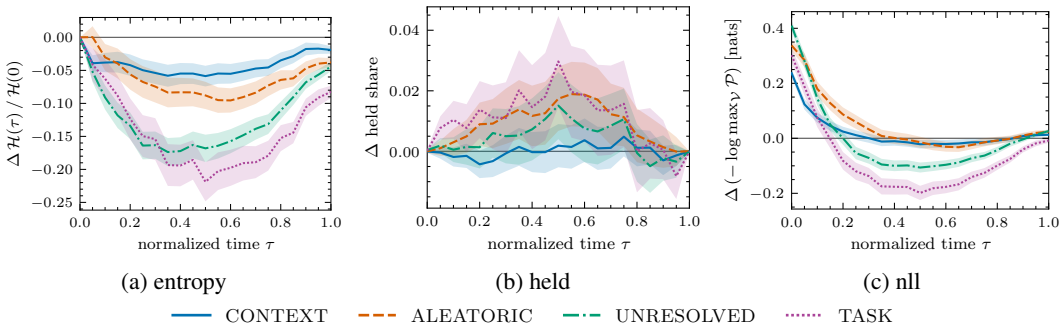


Figure 6: **Dream.** Departure from the CONTROL prompts of the same corpus and answer length, over normalized denoising time τ. Bands are 95% intervals.

D.2 Temporal signature

The tables above read each signal as one number per generation. The figures below unroll the same generations along τ. For every prompt carrying a source we subtract the mean curve of the CONTROL prompts in the same corpus and the same answer-length bin. Accordingly, a non-zero value is the addition of the source after the two confounders of Section C.2 are held fixed, and the CONTROL baseline is flat at zero by construction. The curves are averaged over prompts and the bands are 95% intervals of that mean. Because the models' block schedules differ, each position's active window is rescaled to $\tau \in [0, 1]$. As a result, $\tau = 0$ is the step a position first receives a posterior and $\tau = 1$ the step it freezes. The entropy panel is normalized by $\mathcal{H}(0)$. Overall, the label TASK departs furthest from the CONTROL baseline, followed by the label UNRESOLVED, then label ALEATORIC, and with label CONTEXT closest to zero.

DiffusionGemma (Figure 3) separates the signals in the middle of the trajectory. The TASK label peaks at $+0.069$ in relative entropy at $\tau = 0.40$, at -0.042 in held share at $\tau = 0.55$, and at $+0.237$ nats in the commit surprisal at $\tau = 0.35$. The output distribution therefore carries almost nothing, which is why FINAL_ENTROPY sits at .45 to .53 in Table 7 while the shape signals reach .73.

Dream (Figure 6) departs in the opposite direction and at the opposite end. Its relative entropy falls to -0.219 for the label TASK at $\tau = 0.50$, and its commit surprisal is largest at $\tau = 0$ ($+0.410$ nats for label UNRESOLVED). Dream's separation is thus concentrated in the initial posterior, which is the trajectory-level explanation for ENT_START being the one signal of Table 8 that leaves chance. Its held panel barely moves ($\leq +0.03$) and hence the reveal order is almost a random schedule carrying no source information.

For Nemotron (Figure 7), the departure of $+0.068$ in relative entropy at $\tau = 0.45$, and the departure of -0.079 in held share at $\tau = 0.40$, and the departure of $+0.290$ nats in commit surprisal at $\tau = 0.10$ demonstrate the largest effects of the four dLLMs. The prompts labeled with CONTEXT are the exception being flat on all three panels ($|\Delta| \leq 0.034$). This displays the curve-level form of the flattest CONTEXT column in the catalog (Table 9 with no signal above .53).

The dLLM LLaDA2.1 (Figure 8) raises all four sources of uncertainty together to about $+0.05$ by $\tau \approx 0.3$ and carries them below zero thereafter. Its entropy panel separates the labeled prompts from the CONTROL ones. It does not separate the sources from each other. The held and commit figures follow Nemotron's shape at roughly two thirds the amplitude (-0.075 and $+0.215$ for the labeled prompts of TASK).

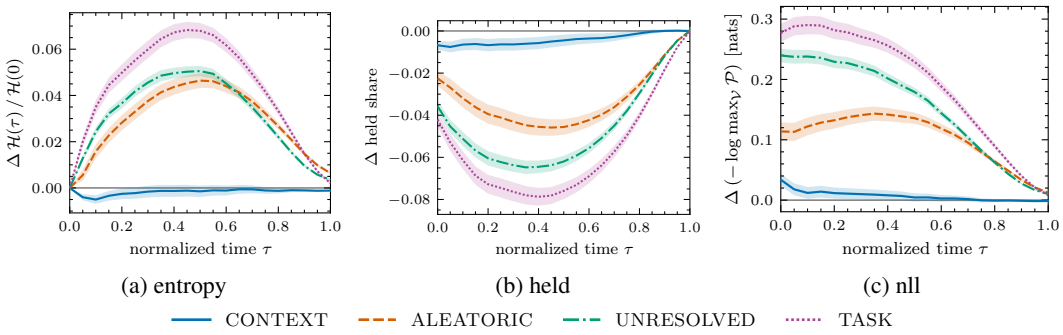


Figure 7: **Nemotron.** Departure from the CONTROL prompts of the same corpus and answer length, over normalized denoising time τ. Bands are 95% intervals.

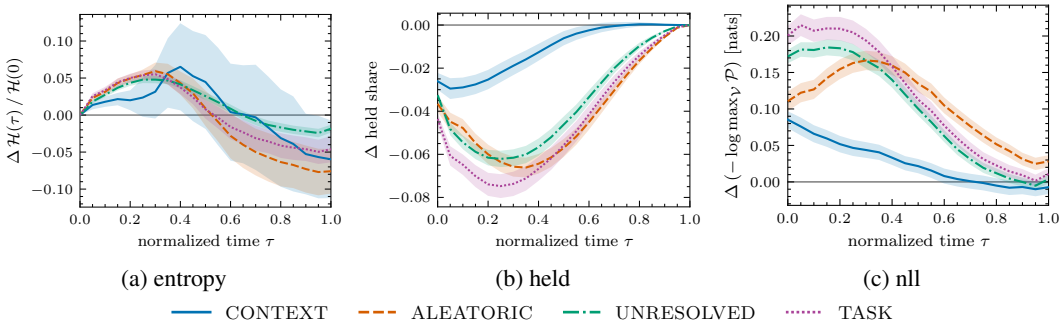


Figure 8: **LLaDA2.1.** Departure from the CONTROL prompts of the same corpus and answer length, over normalized denoising time τ. Bands are 95% intervals.

D.3 CONTROL FOR ANSWER LENGTH

Answer length is the strongest confounder in SOUL. Prompts that carry a source are answered at several times the length of CONTROL prompts by every model (Table 4), and about half of the TASK, UNRESOLVED and ALEATORIC answers reach the 512-token limit on DiffusionGemma, Gemma 4, Nemotron and Ministral 3. Table 14 displays the classification task of Table 3 with a classifier that sees only the answer length.

Length alone reaches an F1 of 0.25 to 0.34 across the five classes, above the random baseline of 0.20 and below every signal classifier. In other words, the source of uncertainty is visible in the trajectory (Section 5.2) beyond what the length of the answer reveals, most clearly for UNRESOLVED and TASK, and least for CONTROL and, on three models, ALEATORIC.

Table 14: **Classification of the source of uncertainty from answer length alone.** This is the baseline of Table 3 with a classifier that solely sees the answer length. The baseline displays what the confound of Section C.2 achieves unaided. Within a block and column, **bold** marks the higher value. Gray values mark the standard deviation over folds.

	F1 per class					**overall**
	TASK	CONT.	UNRES.	ALEAT.	CTRL.	F1
DiffusionGemma	.20 $\pm$.02	.36 $\pm$.01	.09 $\pm$.02	.40 $\pm$.01	.63 $\pm$.01	**.34** $\pm$.01
Gemma 4 (AR)	.21 $\pm$.01	**.37** $\pm$.03	.00 $\pm$.00	**.42** $\pm$.00	**.65** $\pm$.01	.33 $\pm$.01
Dream	**.27** $\pm$.01	.20 $\pm$.03	.21 $\pm$.01	.21 $\pm$.03	.56 $\pm$.03	.29 $\pm$.01
Qwen2.5 (AR)	.07 $\pm$.03	.23 $\pm$.01	.32 $\pm$.02	.21 $\pm$.02	.63 $\pm$.01	.29 $\pm$.01
Nemotron	.07 $\pm$.02	.19 $\pm$.03	**.33** $\pm$.01	.06 $\pm$.01	.60 $\pm$.02	.25 $\pm$.00
Ministral 3 (AR)	.10 $\pm$.03	.28 $\pm$.01	.19 $\pm$.04	.40 $\pm$.01	.63 $\pm$.01	.32 $\pm$.00
LLaDA2.1	.17 $\pm$.02	.24 $\pm$.01	.23 $\pm$.01	.31 $\pm$.02	.61 $\pm$.01	.31 $\pm$.01

D.4 CONFUSION MATRICES

Tables 15 and 16 give the confusion matrices behind rows (a) and (b) of Table 3. Two confusions carry most of the error. TASK prompts are assigned to UNRESOLVED or ALEATORIC, and CONTEXT prompts are assigned to CONTROL, which is the confusion that Section 6 describes: a confident guess looks like a prompt that lacks nothing.

Table 15: **Confusion matrices of the classification task with the signals.** Predictions are the argmax of the out-of-fold probabilities, pooled over corpora and answer lengths. Entries are row percentages; n is the number of prompts per true class.

		predicted					
	true	TASK	CONT.	UNRES.	ALEAT.	CTRL.	n
DiffusionGemma	TASK	**47**	7	21	22	3	1,295
	CONT.	8	**49**	12	10	21	3,134
	UNRES.	19	11	**51**	15	5	1,626
	ALEAT.	16	8	18	**52**	6	1,882
	CTRL.	6	26	9	7	**53**	8,171
Gemma 4 (AR)	TASK	**23**	10	33	30	4	1,295
	CONT.	6	**52**	8	10	24	3,134
	UNRES.	15	7	**55**	17	5	1,626
	ALEAT.	16	8	22	**48**	7	1,882
	CTRL.	5	27	9	7	**53**	8,171
Dream	TASK	**36**	12	16	18	19	1,295
	CONT.	7	**32**	16	10	35	3,134
	UNRES.	11	17	**49**	15	9	1,626
	ALEAT.	13	12	20	**38**	17	1,882
	CTRL.	4	18	7	8	**63**	8,171
Qwen2.5 (AR)	TASK	**31**	10	26	21	11	1,295
	CONT.	10	**34**	14	5	37	3,134
	UNRES.	11	7	**65**	13	4	1,626
	ALEAT.	17	3	27	**37**	16	1,882
	CTRL.	8	19	11	5	**58**	8,171
Nemotron	TASK	**54**	7	21	15	4	1,295
	CONT.	7	**39**	11	9	34	3,134
	UNRES.	14	7	**63**	13	2	1,626
	ALEAT.	22	7	21	**44**	7	1,882
	CTRL.	4	27	6	6	**56**	8,171
Ministral 3 (AR)	TASK	**22**	7	31	32	9	1,295
	CONT.	5	**27**	13	7	48	3,134
	UNRES.	15	7	**61**	12	5	1,626
	ALEAT.	17	5	25	**43**	9	1,882
	CTRL.	4	19	9	5	**63**	8,171
LLaDA2.1	TASK	**38**	11	28	21	3	1,295
	CONT.	10	**36**	12	11	30	3,134
	UNRES.	21	8	**51**	18	2	1,626
	ALEAT.	16	7	19	**52**	6	1,882
	CTRL.	6	25	7	9	**53**	8,171

Table 16: **Confusion matrices of the classification task with the probe.** Predictions are the argmax of the out-of-fold probabilities, pooled over corpora and answer lengths. Entries are row percentages; n is the number of prompts per true class.

		predicted					
	true	TASK	CONT.	UNRES.	ALEAT.	CTRL.	n
DiffusionGemma	TASK	**40**	11	23	24	3	1,295
	CONT.	5	**56**	10	8	20	3,134
	UNRES.	12	10	**62**	13	3	1,626
	ALEAT.	11	7	18	**58**	6	1,882
	CTRL.	4	29	9	5	**52**	8,171
Gemma 4 (AR)	TASK	**36**	14	30	17	3	1,295
	CONT.	5	**58**	6	9	22	3,134
	UNRES.	17	10	**52**	17	4	1,626
	ALEAT.	21	13	35	**27**	5	1,882
	CTRL.	4	31	8	7	**51**	8,171
Dream	TASK	**46**	8	17	18	11	1,295
	CONT.	11	**30**	21	10	28	3,134
	UNRES.	14	11	**58**	13	4	1,626
	ALEAT.	20	9	24	**38**	9	1,882
	CTRL.	9	19	7	10	**55**	8,171
Qwen2.5 (AR)	TASK	**37**	13	24	20	6	1,295
	CONT.	15	**39**	12	5	29	3,134
	UNRES.	18	7	**56**	16	3	1,626
	ALEAT.	16	6	28	**38**	12	1,882
	CTRL.	11	22	9	5	**52**	8,171
Nemotron	TASK	**64**	4	13	16	4	1,295
	CONT.	5	**37**	12	10	36	3,134
	UNRES.	16	6	**60**	15	3	1,626
	ALEAT.	21	4	19	**47**	9	1,882
	CTRL.	4	22	8	9	**57**	8,171
Ministral 3 (AR)	TASK	**32**	9	29	24	5	1,295
	CONT.	9	**39**	15	7	30	3,134
	UNRES.	13	7	**60**	16	3	1,626
	ALEAT.	20	5	31	**37**	7	1,882
	CTRL.	5	24	11	6	**53**	8,171
LLaDA2.1	TASK	**40**	11	25	21	3	1,295
	CONT.	11	**38**	11	10	30	3,134
	UNRES.	17	10	**52**	19	2	1,626
	ALEAT.	13	6	17	**54**	10	1,882
	CTRL.	6	23	8	9	**54**	8,171

D.5 Ablation Study: Final Posterior Baseline for dLLMs

Removing the time axis inside the same dLLM drops the overall F1 from .46 to .31 (DiffusionGemma), .42 to .35 (Dream), .47 to .29 (Nemotron) and .42 to .31 (LLaDA2.1). Nemotron's final posterior fails on CONTROL (.21), which drives its overall .29. This comparison holds architecture, pretraining and scale fixed and hence isolates what the denoising steps contribute. The cross-architecture pairs of section 5.2 corroborate it but do not carry it.

Table 17: **Final posterior baseline.** The classification of Table 3 for dLLMs compared with the *final posterior*, which gives the classifier only the distribution each token was drawn from. The gap from *signals* is what the time axis contributes inside the dLLMs. Every number here is pooled over corpora and answer lengths, because a threshold cannot be applied inside a bin, so these values carry the confound of Section D.3. Within a block and column, **bold** marks the highest value. Gray values mark the standard deviation over folds.

		F1 per class					**overall**
		TASK	CONT.	UNRES.	ALEAT.	CTRL.	F1
DiffusionGemma	Signals	**.38** $\pm.02$	.43 $\pm.01$	.40 $\pm.01$	.46 $\pm.03$	.64 $\pm.01$	.46 $\pm.01$
	Probe	.37 $\pm.04$	**.46** $\pm.02$	**.47** $\pm.03$	**.52** $\pm.01$	.64 $\pm.03$	**.49** $\pm.02$
	Final posterior	.19 $\pm.01$	.27 $\pm.02$	.22 $\pm.01$	.23 $\pm.01$	**.65** $\pm.01$	.31 $\pm.00$
Dream	Signals	**.34** $\pm.02$	**.32** $\pm.03$	.39 $\pm.02$	**.35** $\pm.02$	**.68** $\pm.01$	**.42** $\pm.01$
	Probe	.34 $\pm.01$	.31 $\pm.03$	**.42** $\pm.02$	.35 $\pm.03$	.65 $\pm.03$	.41 $\pm.01$
	Final posterior	.28 $\pm.02$	.20 $\pm.02$	.36 $\pm.01$	.28 $\pm.01$	.62 $\pm.01$	.35 $\pm.01$
Nemotron	Signals	.44 $\pm.03$	.36 $\pm.02$	**.49** $\pm.01$	**.42** $\pm.02$	.65 $\pm.01$	.47 $\pm.01$
	Probe	**.50** $\pm.03$	**.36** $\pm.06$	.47 $\pm.03$	.42 $\pm.03$	**.65** $\pm.05$	**.48** $\pm.01$
	Final posterior	.27 $\pm.01$	.33 $\pm.01$	.31 $\pm.03$	.32 $\pm.02$	.21 $\pm.02$	.29 $\pm.01$
LLaDA2.1	Signals	.31 $\pm.02$	.34 $\pm.01$	.40 $\pm.02$	.43 $\pm.01$	.64 $\pm.02$	.42 $\pm.01$
	Probe	**.32** $\pm.04$	**.36** $\pm.02$	**.41** $\pm.03$	**.45** $\pm.01$	**.64** $\pm.03$	**.44** $\pm.00$
	Final posterior	.18 $\pm.02$	.27 $\pm.01$	.28 $\pm.01$	.30 $\pm.03$	.55 $\pm.01$	.31 $\pm.01$