COMPETENCE BEFORE GENERATION, CONFIDENCE AFTER: A TWO-STAGE SELF-AWARENESS FRAMEWORK FOR LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

While demonstrating remarkable capabilities, Large Language Models (LLMs) often generate fluent yet unreliable answers, making reliability estimation a critical prerequisite for safe deployment. Existing confidence estimators draw on surface-level outputs (*e.g.*, token probabilities and multi-sample consistency) and internal-state probes. We study how internal reliability signals support prospective judgments before generation and retrospective judgments about a generated answer. This perspective echoes human meta-cognition, where monitoring at different stages supports self-awareness. Motivated by this insight, we propose a two-stage self-awareness framework that monitors model reliability from the internal states of a frozen language model. Before generation, the framework estimates the model's competence to answer correctly, defined through sampled-answer correctness. After generation, it assesses the correctness of the particular answer produced. Lightweight heads learn these stage-specific targets with sample-derived soft correctness and consistency supervision, and reuse a single autoregressive generation process at inference. Experiments on seven benchmarks demonstrate improvements in average ranking and calibration over the knowledge-boundary baseline under both in-domain and out-of-domain settings. Across nine in-domain settings, SA improves macro AUROC by 7.51 points; strict out-of-domain evaluation yields 0.7002 ± 0.0064 over twelve settings. Further analyses reveal complementary predictive information in the two stages: pre-generation competence remains informative about correctness after conditioning on post-generation confidence in pooled diagnostics. These findings highlight the value of stage-specific internal monitoring for LLM reliability estimation. Code will be made publicly available to facilitate future research.

1 INTRODUCTION

Over the past few years, large language models (LLMs) have evolved into broadly capable language-and-reasoning engines, achieving strong performance across instruction following, open-domain question answering, and multi-step problem solving (Zhao et al., 2023; Chang et al., 2024). However, their reliability still lags behind their fluency: models can hallucinate and exhibit poor confidence calibration, producing answers that sound highly certain while being incorrect (Huang et al., 2025; Liu et al., 2025). Reliable confidence estimation is therefore essential for deploying LLMs in high-stakes or reliability-sensitive settings.

To bridge this gap, a common approach to estimating an LLM's confidence is to rely on surface-level outputs exposed at the output layer (Liu et al., 2025). One line of work uses token probabilities, sequence likelihoods, or related logit-level statistics as proxies for answer correctness, often combined with post-hoc calibration techniques such as temperature scaling (Guo et al., 2017). Another line of work uses multi-sample consistency, estimating confidence from the agreement among multiple sampled responses (Lyu et al., 2025). Verbalized confidence methods prompt models to explicitly report their own confidence in natural language or numeric form (Tian et al., 2023).

Recently, researchers have moved from output-derived signals to internal-state-based introspection, motivated by evidence that internal activations encode information correlated with factuality and hal-

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



Figure 1: Illustration of the two-stage self-awareness framework. (*Left*) Original descriptive distribution of the absolute difference ($|\Delta|$) between Pre and Post scores. The two scores predict different targets, so their difference does not by itself establish a change in the same correctness belief. (*Right*) Stage-specific monitoring of prospective competence before generation and retrospective confidence after generation.

lucination (Azaria & Mitchell, 2023). For example, classifiers trained on internal states of answers can distinguish truthful from false statements (Azaria & Mitchell, 2023). In parallel, knowledge-boundary research uses pre- and post-generation representations to assess model reliability (Ni et al., 2025). These advances motivate a stage-specific view: how do internal representations support judgments about the ability to answer and the correctness of the realized response?

Our empirical analysis examines prospective and retrospective signals as distinct, informative views of model reliability (Figure 1). This perspective aligns with human meta-cognition, which is inherently multi-stage. Specifically, agents can form prospective judgments before acting and retrospective judgments after observing the outcome (Nelson & Narens, 1990; Koriat, 2000). This distinction motivates monitoring both the competence available before generation and the reliability of the particular answer produced.

Motivated by this insight, we frame reliability estimation as a form of model self-awareness and propose a two-stage self-awareness framework. As illustrated in the bottom-right part of Figure 1, our framework probes the internal states of a frozen LLM at two points in the generation cycle. Specifically, (i) in the pre-generation stage, we model a *prospective competence* signal, estimating whether sampled answers tend to be correct for the question; (ii) in the post-generation stage, we model a *retrospective confidence* signal, estimating whether the particular generated answer is correct. These signals are modeled via independently trained lightweight MLP heads operating on pre- and post-generation internal states, respectively. Each head learns its stage-specific hard target together with sample-derived soft correctness and consistency supervision (Section 3). The Post head supplies the primary answer-confidence score. At inference, hidden states are collected during a single autoregressive generation process and pooled for the Post head, reusing the computation of answer generation.

We evaluate our two-stage self-awareness framework on three widely-used benchmarks: Natural Questions (Kwiatkowski et al., 2019), StrategyQA (Geva et al., 2021), and MATH (Hendrycks et al., 2021). Across Llama2-7B-Chat (Touvron et al., 2023), Llama3-8B-Instruct (Dubey et al., 2024), and Qwen3-4B-Instruct (Yang et al., 2025), SA improves macro AUROC by 7.51 points and reduces Brier score and ECE by 4.30 and 9.49 points relative to our KB baseline. Under strict out-of-distribution evaluation on four additional benchmarks, SA achieves macro AUROC 0.7002 ± 0.0064, outperforming KB in $11/12$ settings and CLCM in all 12 settings on mean AUROC. A Qwen3-32B extension also improves mean AUROC over the prompt-final control and CLCM on all three ID tasks. Further analyses reveal complementary predictive information in the two stages: among examples with comparable post-generation scores, pre-generation competence remains informative about answer correctness in pooled diagnostics.

Our contributions are summarized as follows:

- We frame reliability estimation as a form of model self-awareness, introducing a two-stage monitoring framework that decouples prospective competence before generation from retrospective confidence in the generated answer.
- We demonstrate improved average ranking, calibration and selective prediction over KB and CLCM across the main ID and OOD evaluations, with multi-seed results and an extension to Qwen3-32B.
- We provide empirical evidence of complementary predictive information in stage-specific signals, together with controlled ablations showing the contribution of sample-derived supervision to confidence estimation.

2 Related Works

Prospective Competence Estimation via Internal States. Recent work has explored how language models' internal states can be used to assess their competence before answering. Ni et al. (2025) proposed pre- and post-generation probes with question-reformulation-based calibration (C^3). Chen et al. (2026) estimate query-level uncertainty before answer generation using training-free self-evaluation signals across layers and tokens. Our self-awareness framework instead augments stage-specific hidden-state probes with correctness and consistency supervision from sampled answers, without additional sampling at inference.

Retrospective Confidence Estimation and Calibration. Confidence modeling has traditionally relied on output-level statistics or auxiliary elicitation mechanisms. Answer likelihoods and elicited correctness probabilities have been used as confidence signals (Kadavath et al., 2022; Lin et al., 2022), but they can be entangled with fluency, answer length, or surface plausibility rather than correctness. Consistency-based methods improve robustness by checking agreement across multiple sampled generations (Lyu et al., 2025), and post-hoc calibration techniques such as temperature scaling can further adjust probability estimates (Guo et al., 2017). These approaches differ in both information access and inference cost. Token likelihood, temperature scaling, and hidden-state probes of semantic uncertainty (Kossen et al., 2024) can use one answer; direct sample-consistency estimation requires multiple generations. Recent work also probes hidden states to verify intermediate reasoning answers (Zhang et al., 2025) and combines hidden-state and attention signals to predict correctness in frozen LLMs (Ghasemabadi & Niu, 2025).

Refusal and Selective Prediction. Refusal—the ability to abstain from answering when uncertain—is central to risk-aware deployment. Selective prediction studies how models trade coverage for accuracy using confidence scores and risk–coverage metrics (Geifman & El-Yaniv, 2017). Selective question answering under domain shift has been studied by Kamath et al. (2020), while Liu et al. (2025) survey uncertainty quantification and confidence calibration in LLMs. Despite this progress, many refusal strategies still rely on output heuristics, verbalized uncertainty, or external evaluators, rather than stage-specific internal signals that can be read before and after generation.

Our work connects these lines by modeling reliability with a two-stage internal-state monitor: a prospective competence signal before generation and a retrospective confidence signal after generation. The monitor uses lightweight heads over frozen LLM representations, is trained with sample-derived supervision, and does not require repeated decoding at inference time.

3 Approach

3.1 Overview

Pipeline Overview. As shown in Figure 2, our method follows a two-stage monitoring pipeline built on a frozen backbone language model. During training, we construct sample-based supervision statistics, extract internal states before and after generation, and train two lightweight predictors independently: a competence head for prospective monitoring before answer generation and a confidence head for retrospective monitoring after generation. The confidence head provides the primary SA score.

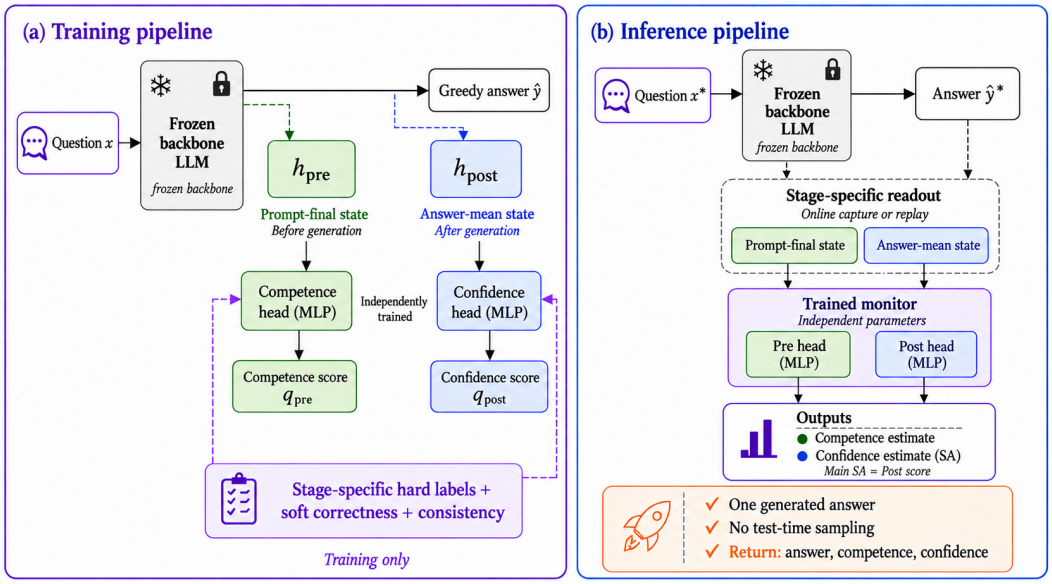


Figure 2: Two independently trained monitors on a frozen backbone. Pre predicts sampled competence from the prompt-final state; Post predicts greedy-answer correctness from answer-span states. The main SA results use the Post score alone. Sample-derived targets supervise training; deployment captures hidden states during a single autoregressive generation process.

Data Sampling and Supervision Signals. For each question x, we generate one greedy answer $\hat{y}$ and a separate set of $K = 30$ stochastic answers $\{\tilde{y}_i\}_{i=1}^{K}$. The sampled answers are used to build supervision signals. We define the sample-based correctness probability as

$$p_{\text{correct}} = \frac{1}{K} \sum_{i=1}^{K} \mathbf{1}[\text{Correct}(x, \tilde{y}_i) = 1] \,, \tag{1}$$

and the agreement frequency with the greedy answer as

$$\text{freq}_{\text{greedy}} = \frac{1}{K} \sum_{i=1}^{K} \mathbf{1}[\text{Norm}(\tilde{y}_i) = \text{Norm}(\hat{y})] \,. \tag{2}$$

Here, $\text{Norm}(\cdot)$ denotes the answer normalization used for matching sampled and greedy outputs.

Training–Inference Boundary. The sample-based statistics p_{correct} and $\text{freq}_{\text{greedy}}$ are used only during training. At inference time, the framework captures hidden states during one greedy autoregressive generation process, then pools the retained answer-token states and applies the confidence head. This single-pass procedure reuses answer-generation computation. A formal pseudocode description is provided in Appendix A.1.

3.2 Internal-State Feature Selection

Internal States as Monitoring Features. A central design choice in our framework is where to read monitoring signals from the backbone. We use residual-stream internal states as monitoring features, providing a unified representation space for both prospective and retrospective monitoring.

Stage-Specific Feature Construction. We use different internal-state features for the two monitoring stages. For prospective competence estimation, we take the internal state at the final token of the prompt, which captures the model state immediately before answer generation. For retrospective confidence estimation, we apply mean pooling over the retained answer span to obtain the post-generation representation. Appendix A.5 specifies the feature-extraction settings.

Layer, Position, and Pooling Choice. We use an intermediate transformer layer to provide stage-specific features for prospective competence and retrospective confidence.

3.3 Two-Stage Heads and Learning Objective

Two-Stage Heads. We use two independently trained lightweight predictors over the selected internal states to realize prospective and retrospective monitoring. The competence head takes the pre-generation feature h_{pre} as input and outputs

$$z_{\text{comp}} = f_{\text{comp}}(h_{\text{pre}}), \tag{3}$$

for competence estimation before answer generation. The confidence head takes the post-generation feature h_{post} as input and outputs

$$z_{\text{conf}} = f_{\text{conf}}(h_{\text{post}}), \tag{4}$$

for confidence estimation after generation. Both scores are probabilities. Each head also includes an auxiliary consistency output, denoted by r_{comp} or r_{conf}, supervised by $\text{freq}_{\text{greedy}}$.

Training Targets. The competence target is $y_{\text{competent}} = \mathbf{1}[p_{\text{correct}} \geq 0.5]$, while the confidence target $y_{\text{correct}} = \text{Correct}(x, \hat{y})$ indicates whether the greedy answer is correct. In addition to these hard targets, we use p_{correct} and $\text{freq}_{\text{greedy}}$ as auxiliary supervision signals to provide complementary uncertainty information.

Learning Objective. The competence-side objective is

$$\mathcal{L}_{\text{comp}} = \mathcal{L}_{\text{main}}^{\text{comp}}(z_{\text{comp}}, y_{\text{competent}}) + \lambda_{\text{comp}}\, \mathcal{L}_{\text{aux}}(z_{\text{comp}}, r_{\text{comp}};\, p_{\text{correct}}, \text{freq}_{\text{greedy}}), \tag{5}$$

and the confidence-side objective is

$$\mathcal{L}_{\text{conf}} = \mathcal{L}_{\text{main}}^{\text{conf}}(z_{\text{conf}}, y_{\text{correct}}) + \lambda_{\text{conf}}\, \mathcal{L}_{\text{aux}}(z_{\text{conf}}, r_{\text{conf}};\, p_{\text{correct}}, \text{freq}_{\text{greedy}}). \tag{6}$$

The overall objective is

$$\mathcal{L} = \mathcal{L}_{\text{comp}} + \mathcal{L}_{\text{conf}}. \tag{7}$$

The heads have separate parameters and are optimized independently. Here, $\mathcal{L}_{\text{main}}^{\text{comp}}$ and $\mathcal{L}_{\text{main}}^{\text{conf}}$ denote the respective primary binary supervision terms. The auxiliary term combines soft correctness supervision on z with consistency supervision on r. Appendix A.6 specifies the loss functions, weights and network configuration.

4 Experiments

4.1 Experimental Setup

Benchmarks and models. Our main ID comparison uses Natural Questions (NQ) (Kwiatkowski et al., 2019), StrategyQA (Geva et al., 2021), and MATH (Hendrycks et al., 2021), with Llama2-7B-Chat, Llama3-8B-Instruct and Qwen3-4B-Instruct. We retain the frozen 80/10/10 data partitions and train only monitors. The Qwen3-32B extension broadens evaluation to a larger backbone (Table 2).

Strict OOD evaluation uses TriviaQA (Joshi et al., 2017), GSM8K (Cobbe et al., 2021), BoolQ (Clark et al., 2019), and CommonsenseQA (Talmor et al., 2019), with 1,000 examples per setting. Its main backbones are Llama3, Qwen3 and Gemma3-4B-it (Gemma Team, 2025). ID-trained checkpoints transfer directly, without OOD training, selection, threshold tuning or temperature fitting.

Comparisons and uncertainty. KB is our historical post-state adaptation of the internal-state probe of Ni et al. (2025); CLCM uses $K = 30$ sampled-answer agreement following Lyu et al. (2025) (greedy-answer adaptation). Matched controls and temperature-scaled comparisons further examine the sources of SA's gains (Appendices C.4 and C.2). We evaluate AUROC for ranking, Brier/ECE for probability quality, and AURC/E-AURC for selective prediction. Lower is better except for AUROC. SA means and sample SD use probe seeds $\{0, 42, 100\}$, holding generated answers and splits fixed. The same answers are rescored, rather than improved by the monitor. Main heads use AdamW, learning rate 10^{-3}, and dev-AUROC selection.

Table 1: ID macro-average over nine settings (Llama2/Llama3/Qwen3 × NQ/StrategyQA/MATH). SA: three probe seeds, mean ± sample SD of seed-wise macro-averages. KB/CLCM: frozen historical aggregates. Parentheses count SA point-estimate wins–losses, not significance.

Method	AUROC↑	Brier↓	ECE↓	AURC↓	E-AURC↓
KB	0.6886 (8-1)	0.2189 (8-1)	0.2027 (9-0)	0.4040 (9-0)	0.1729 (9-0)
CLCM	0.6750 (9-0)	0.2071 (7-2)	0.1556 (7-2)	0.4115 (8-1)	0.1799 (8-1)
SA	$.7637 \pm .0021$	$.1759 \pm .0060$	$.1078 \pm .0142$	$.3629 \pm .0015$	$.1317 \pm .0015$

4.2 Main Results

4.2.1 In-distribution reliability

Table 1 shows higher mean AUROC and lower mean Brier, ECE and selective risk than the historical baselines. Relative to KB, SA gains 7.51 AUROC points and reduces Brier/ECE by 4.30/9.49 points. AUROC wins are 8/9 against KB and 9/9 against CLCM; these are point-estimate counts, not significance tests.

The per-setting results in Table 2 retain important exceptions: CLCM has lower selective risk on Llama2–StrategyQA and lower Brier/ECE on Llama2–MATH; KB has higher AUROC and lower Brier on Qwen3–MATH. SA leads all five point estimates on NQ across the three main backbones. The 32B extension improves mean AUROC over its prompt-final control and CLCM on all three ID tasks. Appendices C.2 and C.7 further compare calibrated confidence and transfer to SimpleQA (Wei et al., 2024).

4.2.2 Strict out-of-distribution transfer

SA's OOD macro AUROC is 0.7002 ± 0.0064, with mean gains of 4.55 points over KB and 11.70 over CLCM. Its AUROC wins are 11/12 and 12/12, respectively. Calibration is less uniform: Brier/ECE win counts are 8/12 and 8/12 against KB, and 11/12 and 10/12 against CLCM. These results support transfer within the tested settings, not uniform improvement across backbones or metrics.

4.3 Ablations and Stage Comparisons

4.3.1 What does each stage predict?

Table 4 compares both SA readouts with the single-stage baselines on correctness. Both improve all five aggregate metrics over KB and CLCM. Post has higher AUROC and lower Brier and selective risk, while Pre has lower ECE. A separately trained prompt-final *correctness* control offers an equal-target comparison: on nine ID settings, its three-seed macro AUROC is 0.7157, versus 0.7606 for hard-only Post and 0.7626 for full Post. This changes both the available representation and supervision and does not isolate the capacity of original KB.

4.3.2 Does combining stages help?

The Pre score contains additional conditional information in pooled diagnostics (Appendix B.2), but this does not guarantee a better deployed fusion. Shared-trunk training improves AUROC in only 2/9 settings relative to independent Post and has worse macro values on all five metrics. Dev-selected score fusion improves Brier/ECE but has lower macro AUROC than independent Post. Thus we retain independent Post as the main confidence score; two-stage complementarity is conditional and metric-dependent.

Auxiliary supervision and feature selection. Table 6 shows that Full leads all five macro metrics. Soft correctness drives calibration gains; adding consistency further improves performance in this comparison. Full reduces Brier/ECE by 0.0176/0.0561 over hard-only. The three-seed fixed-feature study corroborates calibration gains over hard-only (Appendix C.1). Layer/pooling controls favor intermediate answer-mean features on average (Appendix C.5, Table 22).

Table 2: ID results across four backbones. SA reports mean ± sample SD over probe seeds $\{0, 42, 100\}$ with fixed answers, labels and splits. Baselines are shown as point estimates. Bold marks the best mean within each block. SQA denotes StrategyQA. Pre denotes the prompt-final correctness control.

Model	Dataset	Method	AUROC↑	Brier↓	ECE↓	AURC↓	E-AURC↓
Llama2-7B	NQ	KB	0.7069	0.2343	0.1661	0.5041	0.2179
		CLCM	0.5973	0.2616	0.1821	0.5686	0.2821
		SA	**.7600**±.0017	**.1924**±.0018	**.0827**±.0175	**.4723**±.0028	**.1860**±.0028
	SQA	KB	0.5320	0.2452	0.1756	0.3007	0.2467
		CLCM	0.6622	0.2249	0.1726	**0.1874**	**0.1327**
		SA	**.6627**±.0096	**.2202**±.0174	**.1285**±.0718	.2177±.0101	.1637±.0101
	MATH	KB	0.4862	0.2320	0.4052	0.9331	0.1945
		CLCM	0.5970	**0.0903**	**0.1054**	0.9049	0.1659
		SA	**.6895**±.0056	.1540±.0389	.2804±.0653	**.8735**±.0003	**.1350**±.0003
Llama3-8B	NQ	KB	0.7210	0.2709	0.2388	0.4578	0.1992
		CLCM	0.6451	0.2727	0.1944	0.4990	0.2402
		SA	**.7696**±.0049	**.1889**±.0006	**.0471**±.0090	**.4406**±.0073	**.1821**±.0073
	SQA	KB	0.5990	0.2611	0.1668	0.2651	0.2044
		CLCM	0.6020	0.2479	0.1836	0.2788	0.2174
		SA	**.6668**±.0050	**.2148**±.0022	**.1197**±.0114	**.2206**±.0004	**.1600**±.0004
	MATH	KB	0.8325	0.1852	0.1756	0.4286	0.1105
		CLCM	0.8258	0.1589	**0.0759**	0.4581	0.1397
		SA	**.8394**±.0003	**.1523**±.0022	.0895±.0095	**.4240**±.0003	**.1058**±.0003
Qwen3-4B	NQ	KB	0.8047	0.2036	0.1690	0.4276	0.1460
		CLCM	0.7308	0.2503	0.1898	0.4802	0.1982
		SA	**.8227**±.0024	**.1768**±.0070	**.1086**±.0196	**.4196**±.0074	**.1379**±.0074
	SQA	KB	0.6182	0.2458	0.2373	0.2267	0.1834
		CLCM	0.5516	0.2501	0.2205	0.2282	0.1843
		SA	**.7668**±.0088	**.1663**±.0026	**.0470**±.0083	**.1256**±.0030	**.0822**±.0030
	MATH	KB	**0.8970**	**0.0919**	0.0899	0.0926	0.0532
		CLCM	0.8636	0.1068	0.0763	0.0983	0.0588
		SA	.8957±.0043	.1170±.0025	**.0671**±.0016	**.0719**±.0014	**.0326**±.0014
Qwen3-32B	NQ	Pre	0.6979	0.2226	0.0719	0.4236	0.2176
		CLCM	0.6946	0.2394	0.1164	0.4258	0.2197
		SA	**.7392**±.0032	**.2056**±.0013	**.0462**±.0202	**.3941**±.0006	**.1881**±.0006
	SQA	Pre	0.6124	0.2164	0.2306	0.1532	0.1308
		CLCM	0.6796	0.1569	0.0780	0.1250	0.1025
		SA	**.7833**±.0059	**.1344**±.0011	**.0510**±.0084	**.0834**±.0024	**.0610**±.0024
	MATH	Pre	0.8019	0.1936	0.1866	0.0953	0.0626
		CLCM	0.8257	**0.1301**	0.0755	0.1060	0.0733
		SA	**.8418**±.0015	.1366±.0033	**.0612**±.0221	**.0808**±.0005	**.0480**±.0005

4.4 LIMITATIONS

The monitor requires internal-state access and labeled, sampled training answers. Native Pre and Post have different targets; independent Post drives the main reliability gains. Per-task OOD results examine transfer across all twelve primary settings (Appendix C.3). On formal Qwen3-32B SimpleQA, SA leads Brier/ECE, while CLCM leads ranking. Single-pass inference adds hidden-state collection, pooling and probe computation; cached-probe training time excludes generation, judging and feature extraction. Length and Stability–Edge diagnostics are exploratory associations, not evidence of conscious introspection or epistemic certainty.

Table 3: Strict OOD macro-average: Llama3/Qwen3/Gemma3 $\times$ four tasks (12 settings). SA uses three probe seeds; parentheses count its mean wins–losses. The ID and OOD backbone sets differ.

Method	AUROC↑	Brier↓	ECE↓	AURC↓	E-AURC↓
KB	0.6548 (11-1)	0.2506 (8-4)	0.2323 (8-4)	0.1963 (11-1)	0.1435 (11-1)
CLCM	0.5832 (12-0)	0.5905 (11-1)	0.6007 (10-2)	0.2319 (11-1)	0.1791 (11-1)
SA	$.7002 \pm .0064$	$.2261 \pm .0142$	$.1808 \pm .0139$	$.1766 \pm .0028$	$.1237 \pm .0028$

Table 4: Per-head correctness evaluation on the nine main ID settings. KB/CLCM are historical baselines; SA-Pre/SA-Post use frozen seed42 scores. Pre is trained on competence; Post is the default SA readout.

Configuration	AUROC↑	Brier↓	ECE↓	AURC↓	E-AURC↓
KB (post-only)	0.6886	0.2189	0.2027	0.4040	0.1729
CLCM	0.6750	0.2071	0.1556	0.4115	0.1799
SA-Pre	0.7201	0.1802	**0.0839**	0.3828	0.1516
SA-Post	**0.7613**	**0.1726**	0.0962	**0.3636**	**0.1324**

5 Conclusion

We proposed a two-stage self-awareness framework for LLMs that models prospective competence before generation and retrospective confidence after generation from internal states. Independently trained lightweight heads and sample-derived auxiliary supervision provide a unified approach to reliability monitoring. Across the main ID and OOD evaluations, the Post score improves average ranking, calibration and selective prediction over KB and CLCM. Pre provides complementary predictive information in pooled diagnostics, and score fusion improves Brier and ECE. By reusing a single autoregressive generation process, the framework supports efficient, stage-specific reliability estimation.

Table 5: Cross-stage comparisons on nine ID settings. Post is the main SA readout; score fusion is fitted on training data and selected on development data.

Configuration	AUROC↑	Brier↓	ECE↓	AURC↓	E-AURC↓
Post	0.7637	0.1759	0.1078	0.3629	0.1317
Shared trunk	0.7618	0.1779	0.1189	0.3642	0.1330
Score fusion	0.7603	0.1610	0.0475	0.3619	0.1308

Table 6: Auxiliary supervision ablation on nine primary ID settings. Entries are macro-averages (seed 42); bold marks the best result. Hard denotes greedy-answer correctness, p_{correct} sampled correctness, and f_{greedy} greedy-answer consistency.

Objective	AUROC↑	Brier↓	ECE↓	AURC↓	E-AURC↓
Full	**.7613**	**.1726**	**.0962**	**.3636**	**.1324**
Hard only	.7570	.1902	.1522	.3663	.1352
Hard + p_{correct}	.7601	.1737	.0985	.3640	.1328
Hard + f_{greedy}	.7563	.1881	.1433	.3673	.1362

AI USE STATEMENT

We used Codex and GPT-based tools to assist with subsequent code modifications, experimental execution and result analysis, figure preparation, and manuscript revision and formatting. The original method conception, research hypotheses, and initial experimental design were developed by the authors without AI assistance. The authors also implemented the initial code framework, with AI assistance used for subsequent modifications. The authors reviewed the code, data, and manuscript text, including the corresponding AI-assisted changes. We take responsibility for the final content of this work, including its methods, results, claims, and artifacts produced with AI assistance.

REPRODUCIBILITY STATEMENT

Appendix A provides the algorithm, data splits, generation settings, supervision, feature extraction, training, baselines, and evaluation metrics; Appendix C reports additional experiments.

REFERENCES

Amos Azaria and Tom Mitchell. The internal state of an LLM knows when it's lying. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 967–976, Singapore, 2023. Association for Computational Linguistics. URL https://aclanthology.org/2023.findings-emnlp.68/.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, Wei Ye, Yue Zhang, Yi Chang, Philip S. Yu, Qiang Yang, and Xing Xie. A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, 15(3):1–45, 2024. doi: 10.1145/3641289. URL https://doi.org/10.1145/3641289.

Lihu Chen, Gerard de Melo, Fabian M. Suchanek, and Gaël Varoquaux. Query-level uncertainty in large language models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=11QZITAMUO.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 2924–2936, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1300. URL https://aclanthology.org/N19-1300/.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021. URL https://arxiv.org/abs/2110.14168.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks, 2017. URL https://arxiv.org/abs/1705.08500.

Gemma Team. Gemma 3 technical report, 2025. URL https://arxiv.org/abs/2503.19786.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361, 2021. doi: 10.1162/tacl_a_00370. URL https://aclanthology.org/2021.tacl-1.21/.

Amirhosein Ghasemabadi and Di Niu. Can LLMs predict their own failures? self-awareness via internal circuits, 2025. URL https://arxiv.org/abs/2512.20578.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1321–1330. PMLR, 2017. URL https://proceedings.mlr.press/v70/guo17a.html.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset, 2021. URL https://arxiv.org/abs/2103.03874.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55, 2025. doi: 10.1145/3703155. URL https://doi.org/10.1145/3703155. Also available as arXiv:2311.05232.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1601–1611, Vancouver, Canada, 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1147. URL https://aclanthology.org/P17-1147/.

Saurav Kadavath et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022. URL https://arxiv.org/abs/2207.05221.

Amita Kamath, Robin Jia, and Percy Liang. Selective question answering under domain shift. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 5684–5696, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.503. URL https://aclanthology.org/2020.acl-main.503/.

Asher Koriat. The feeling of knowing: Some metatheoretical implications for consciousness and control. *Consciousness and Cognition*, 9(2):149–171, 2000. doi: 10.1006/ccog.2000.0433. URL https://doi.org/10.1006/ccog.2000.0433.

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

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019. doi: 10.1162/tacl_a_00276. URL https://aclanthology.org/Q19-1026/.

Stephanie Lin, Jacob Hilton, and Owain Evans. Teaching models to express their uncertainty in words. *arXiv preprint arXiv:2205.14334*, 2022.

Xiaoou Liu, Tiejin Chen, Longchao Da, Chacha Chen, Zhen Lin, and Hua Wei. Uncertainty quantification and confidence calibration in large language models: A survey. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, pp. 6107–6117. Association for Computing Machinery, 2025. doi: 10.1145/3711896.3736569. URL https://doi.org/10.1145/3711896.3736569.

Qing Lyu, Kumar Shridhar, Chaitanya Malaviya, Li Zhang, Yanai Elazar, Niket Tandon, Marianna Apidianaki, Mrinmaya Sachan, and Chris Callison-Burch. Calibrating large language models with sample consistency. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39 (18):19260–19268, 2025. doi: 10.1609/aaai.v39i18.34120. URL https://ojs.aaai.org/index.php/AAAI/article/view/34120.

Thomas O. Nelson and Louis Narens. Metamemory: A theoretical framework and new findings. In Gordon H. Bower (ed.), *Psychology of Learning and Motivation*, volume 26, pp. 125–173. Academic Press, 1990. doi: 10.1016/S0079-7421(08)60053-5. URL https://doi.org/10.1016/S0079-7421(08)60053-5.

Shiyu Ni, Keping Bi, Jiafeng Guo, Lulu Yu, Baolong Bi, and Xueqi Cheng. Towards fully exploiting LLM internal states to enhance knowledge boundary perception. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 24315–24329, Vienna, Austria, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-long.1184. URL https://aclanthology.org/2025.acl-long.1184/.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4149–4158, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1421. URL https://aclanthology.org/N19-1421/.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D. Manning. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 5433–5442, Singapore, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.330. URL https://aclanthology.org/2023.emnlp-main.330/.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models, 2023. URL https://arxiv.org/abs/2307.09288.

Jason Wei, Nguyen Karina, Hyung Won Chung, Yunxin Joy Jiao, Spencer Papay, Amelia Glaese, John Schulman, and William Fedus. Measuring short-form factuality in large language models, 2024. URL https://arxiv.org/abs/2411.04368.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, et al. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Anqi Zhang, Yulin Chen, Jane Pan, Chen Zhao, Aurojit Panda, Jinyang Li, and He He. Reasoning models know when they're right: Probing hidden states for self-verification, 2025. URL https://arxiv.org/abs/2504.05419.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. A survey of large language models, 2023. URL https://arxiv.org/abs/2303.18223.

A Experimental Protocol and Implementation Details

This appendix consolidates the experimental setup, supervision construction, internal-state extraction, monitor training, and baseline implementation. Controlled comparisons specify the features, supervision and optimization held fixed.

A.1 Algorithmic Summary

Algorithm 1 summarizes the training and inference procedure of the proposed two-stage self-awareness framework. During training, stochastic samples are used only to construct supervision targets, while the frozen backbone provides stage-specific internal states for the competence and confidence heads. During inference, the framework captures hidden states within one greedy generation process and requires no additional sampled answers.

Algorithm 1 Two-Stage Self-Awareness Monitoring

Setup. Training set $\mathcal{D} = \{(x_i, \mathcal{R}_i)\}_{i=1}^{N}$ with reference set $\mathcal{R}_i$; frozen language model M exposing intermediate internal states; sample size K; competence threshold τ; competence head g_{ϕ_1}; confidence head g_{ϕ_2}.

Offline supervision construction.

for each $(x, \mathcal{R}) \in \mathcal{D}$ **do**

Generate one greedy answer $\hat{y}$ from $M(\cdot \mid x)$.

Generate K stochastic answers $\{\tilde{y}_k\}_{k=1}^{K}$ from the same model.

Compute sample correctness:

$$p_{\text{correct}}(x) = \frac{1}{K} \sum_{k=1}^{K} \mathbb{1}[\text{Correct}(x, \tilde{y}_k) = 1] \,.$$

Compute greedy agreement frequency:

$$\text{freq}_{\text{greedy}}(x) = \frac{1}{K} \sum_{k=1}^{K} \mathbb{1}[\text{Norm}(\tilde{y}_k) = \text{Norm}(\hat{y})] \,.$$

Set $y_{\text{competent}}(x) = \mathbb{1}[p_{\text{correct}}(x) \geq \tau]$.

Set $y_{\text{correct}}(x) = \text{Correct}(x, \hat{y})$.

end for

Monitor training.

for each minibatch $\mathcal{B} \subset \mathcal{D}$ **do**

Extract the pre-generation internal state h_{pre} at the prompt-end token.

Extract the post-generation internal state h_{post} from the answer span.

Predict competence score:

$$z_{\text{comp}} = g_{\phi_1}(h_{\text{pre}}).$$

Predict confidence score:

$$z_{\text{conf}} = g_{\phi_2}(h_{\text{post}}).$$

Optimize each head independently with its corresponding loss:

$$\min_{\phi_1} \mathcal{L}_{\text{comp}}, \qquad \min_{\phi_2} \mathcal{L}_{\text{conf}}.$$

Update only (ϕ_1, ϕ_2); keep the backbone M frozen.

end for

Inference.

Process $x^\star$ once and capture the prompt-final state $h^\star_{\text{pre}}$.

Generate one greedy answer $\hat{y}^\star$, collecting answer-token states with forward hooks.

After generation, mean-pool the collected answer states to obtain $h^\star_{\text{post}}$.

Return:

$$\left(\hat{y}^\star,\ g_{\phi_1}(h^\star_{\text{pre}}),\ g_{\phi_2}(h^\star_{\text{post}})\right).$$

A.2 Datasets and Splits

We use three in-distribution (ID) datasets for training and evaluation, and four out-of-distribution (OOD) datasets for strict generalization evaluation. OOD datasets are *never* used for training, model selection, threshold tuning, or recalibration; we only evaluate the ID-trained checkpoints on their test splits.

Table 7: Datasets used in the main paper. ID datasets are split 80/10/10 (train/dev/test) by stable hash on question id, with the same split shared across the three backbones to remove split-induced variance. OOD datasets are evaluated on a fixed 1,000-sample test slice (TriviaQA / GSM8K / BoolQ / CommonsenseQA).

Role	Dataset	Task type	Judge	#Test
ID	NQ (Natural Questions)	Open-domain QA	EM/F1 substring match	∼1,000
ID	StrategyQA	Yes/No reasoning	Yes/No match	∼200
ID	MATH	Math word	Numeric normalized match	∼1,000
OOD	TriviaQA	Open-domain QA	EM/F1 substring match	1,000
OOD	GSM8K	Math (grade-school)	Numeric (#### priority)	1,000
OOD	BoolQ	Yes/No	Yes/No match	1,000
OOD	CommonsenseQA	5-way MC	MC letter match (A–E)	1,000

ID–OOD mapping for checkpoint reuse. For OOD evaluation, each OOD dataset is mapped to a single ID dataset whose ID-trained checkpoint is reused without modification: TriviaQA ← NQ; GSM8K ← MATH; BoolQ ← StrategyQA; CommonsenseQA ← StrategyQA. For the knowledge-boundary baseline (KB), the OOD KB tree is built with directory naming `<d_ood>_via_<id>`, with `train/dev` symlinked to the ID partition and only `test` pointing to the OOD slice; `test_labels.pt` is materialized from OOD `y_correct`.

Backbones. Three open-weight instruction-tuned LLMs are used as the main in-distribution (ID) backbones, chosen to span model families and sizes: LLAMA-2-7B-CHAT, META-LLAMA-3-8B-INSTRUCT, QWEN3-4B-INSTRUCT. For the strict out-of-distribution (OOD) evaluation we additionally include GEMMA-3-4B-IT in place of LLAMA-2-7B-CHAT; the OOD backbone set is therefore LLAMA-3-8B-INSTRUCT, QWEN3-4B-INSTRUCT, GEMMA-3-4B-IT. The Gemma-3 case is also discussed as a limitation in §A.10 because its multimodal `text_config` nesting and divergent training dynamics interact unfavorably with our default supervision recipe.

A.3 Generation Settings

For each example we produce a *greedy* answer (used to derive y_{correct}) and K *stochastic* samples (used to derive p_{correct}, $y_{\text{competent}}$ and $\text{freq}_{\text{greedy}}$). Generation uses vLLM with the settings in Table 8.

Table 8: Decoding and sampling settings for ID data construction and OOD evaluation.

Mode	Hyperparameter	Value
Greedy	temperature	0
Greedy	top-p / top-k	1.0 / −1
Greedy	#samples per question	1
Sampling	temperature	0.7
Sampling	top-p	0.95
Sampling	#samples per question	$K = 30$

Prompts. We use a uniform *zero-shot, no-CoT* chat-style prompt for each judge type. For multiple-choice (CSQA), the question is rendered with five labeled options A–E. For Yes/No (StrategyQA, BoolQ), the model is instructed to answer *yes* or *no*. For math (MATH, GSM8K), the model is instructed to put the final numeric answer after a sentinel `####`; the judge first attempts to extract from `####` and falls back to last-numeric. Exact prompt templates will be provided in a future code release.

A.4 Supervision Construction

For each example with greedy answer $\hat{y}^{(g)}$ and K stochastic samples $\{\hat{y}^{(k)}\}_{k=1}^{K}$, we materialize four supervision targets used to train and evaluate the two heads:

Table 9: Per-example supervision derived from the dataset gold answer and $K{=}30$ stochastic samples.

Symbol	Definition	Used by
$y_{\text{correct}} \in \{0,1\}$	$\mathbb{1}[\text{judge}(\hat{y}^{(g)}, \text{ref}) = 1]$	Conf. head main loss
$p_{\text{correct}} \in [0,1]$	$\frac{1}{K}\sum_{k=1}^{K} \mathbb{1}[\text{judge}(\hat{y}^{(k)}, \text{ref}) = 1]$	Both heads as soft target
$y_{\text{competent}} \in \{0,1\}$	$\mathbb{1}[p_{\text{correct}} \geq \tau], \quad \tau = 0.5$	Comp. head main loss
$\text{freq}_{\text{greedy}} \in [0,1]$	$\frac{1}{K}\sum_{k=1}^{K} \mathbb{1}[\hat{y}^{(k)} \equiv \hat{y}^{(g)}]$	Consistency aux. on both heads

The judge function $\text{judge}(\cdot)$ is dataset-specific (Table 7). The equivalence relation $\equiv$ used by $\text{freq}_{\text{greedy}}$ matches the same judge type but is computed between samples and the greedy answer rather than against the gold reference. Each head is trained independently with its stage-specific hard target and the two auxiliary targets; ablations that drop p_{correct} or $\text{freq}_{\text{greedy}}$ supervision are discussed in the supervision-component ablation in the main paper.

A.5 Internal-State Extraction

At inference, forward hooks capture intermediate states from the *frozen* backbone during prompt processing and autoregressive decoding. We extract two feature vectors per example:

- **Pre-state** $h_{\textbf{pre}}$: hidden state of the *prompt-end token* (the last token of the prompt, immediately before generation begins). It captures the model's pre-decoding sense of the question.
- **Post-state** $h_{\textbf{post}}$: *mean pooling* of hidden states over the answer-span tokens (i.e., the tokens corresponding to `greedy_answer`). It captures information accumulated during generation.

Both vectors are read from the same intermediate layer, set by default to $\ell^* = \lfloor L/2 \rfloor$ where L is the number of transformer blocks of the backbone (Table 10). All experiments fix this default unless the layer-index ablation is explicitly invoked. Token-to-character alignment is performed at extraction time with the backbone tokenizer; misaligned examples ($< 0.1\%$) are dropped. Offline training and fixed-feature evaluations use cached teacher-forced states from `[prompt, greedy_answer]`, with prompt/answer limits of 256/256 tokens and a total limit of 512; answer truncation retains the prefix.

Storage layout: extracted features are written as `.pt` files aligned line-by-line with `artifacts.jsonl`, namely `hidden_states_pre_last.pt` of shape $[N, D]$ and `hidden_states_avg_last.pt` of shape $[N, D]$. ID and OOD share the same on-disk schema, which makes baseline reuse trivial.

A.6 Monitor Training

The backbone remains frozen while the two stage-specific monitors learn from the supervision defined in §3. Pre predicts sampled competence and Post predicts greedy-answer correctness. Monitor parameters are selected on development data, and the primary results summarize three probe seeds $\{0, 42, 100\}$. The controlled supervision study holds features and optimization fixed across its four objectives (Appendix C.1).

Table 10: Backbone hidden-state geometry. For GEMMA-3-4B-IT, sizes are read from the nested `text_config`; SA is reported separately on this backbone (§A.10).

Backbone	**#Layers** L	**Hidden** D	**Layer** ℓ^*	**Pre/Post Pooling**
Llama-2-7B-Chat	32	4096	16	end-token / answer-span mean
Meta-Llama-3-8B-Instruct	32	4096	16	end-token / answer-span mean
Qwen3-4B-Instruct	36	2560	18	end-token / answer-span mean
Gemma-3-4B-it (OOD only)	34	2560	17	end-token / answer-span mean

Primary-backbone heads and losses. Each primary-backbone monitor has a bias-free $D \to 1024$ GELU trunk with dropout 0.1, no LayerNorm, two classification logits, and a separate consistency output. For classification logits (a_0, a_1), the primary probability is $z = \sigma(a_1 - a_0)$; the consistency probability is $r = \sigma(c)$. Pre and Post have separate parameters. The reported SA score is the raw Post probability z_{conf}.

Writing BCE_w for positive-class-weighted binary cross-entropy and $\text{FL}_{w,1}$ for its focal variant with $\gamma = 1$, the primary losses for the main three-backbone setting are

$$\mathcal{L}^{\text{comp}}_{\text{main}}(z, y) = \text{BCE}_w(z, y), \qquad \mathcal{L}^{\text{conf}}_{\text{main}}(z, y) = \text{FL}_{w,1}(z, y).$$

The auxiliary term is

$$\mathcal{L}_{\text{aux}}(z, r; p, f) = \text{BCE}(z, p) + \text{BCE}(r, f), \qquad \lambda_{\text{comp}} = \lambda_{\text{conf}} = 0.5.$$

The positive weight w is the training-split negative/positive ratio; both auxiliary terms use ordinary BCE. The two primary losses use their respective stage-specific hard labels.

A.7 BASELINE IMPLEMENTATION

We compare against three baselines, all evaluated under exactly the same per-example judge functions and dataset splits as our method.

KB-Baseline (Knowledge-Boundary Probe, post-state). Following Ni et al. (2025), KB trains a single MLP probe on h_{post} (same layer ℓ^* and same answer-span mean pooling as our confidence head) to predict y_{correct} as a binary classification problem, with no soft labels and no consistency auxiliary. The probe trunk is $D \to 256 \to 128 \to 1$, optimized with Adam at lr 10^{-3}, batch 128, 30 epochs, three seeds; we evaluate the multi-seed mean. We adapt the official KB code path so that an ID checkpoint can be evaluated on an OOD slice without retraining: directory naming `<d_ood>_via_<id>` ensures the path detection in the upstream `main.py` treats OOD inputs as belonging to the *generative* branch (rather than the MMLU-style branch). The training/dev tensors are symlinked from the ID partition; only `test` points to the OOD slice.

CL-Baseline (Sample-Consistency Calibration, CLCM). We adapt the agreement-based measure of Lyu et al. (2025) by computing consistency with the fixed greedy answer, keeping the evaluated response identical across methods. We use the `agree_percent` aggregator,

$$\text{conf}_{\text{CLCM}} \;=\; \frac{1}{K}\sum_{k=1}^{K} \mathbb{1}\big[\hat{y}^{(k)} \equiv \hat{y}^{(g)}\big],$$

i.e., the empirical fraction of samples that match the greedy answer under the same judge equivalence as $\text{freq}_{\text{greedy}}$. No training is performed; the same $K{=}30$ generations used by p_{correct} are reused, keeping the sampled-answer pool fixed. CLCM provides a sample-agreement confidence score; abstention is obtained by thresholding this score.

LogProb-Baseline (ID and OOD). LogProb is a likelihood baseline computed post-hoc from the same `artifacts.jsonl` used for SA / KB / CLCM. For each example we re-run a teacher-forcing forward pass through the frozen backbone with `[prompt, greedy_answer]` and read the per-token log-probabilities of the greedy answer span; the confidence is the token-mean log-probability,

$$\mathrm{conf}_{\mathrm{LP}} \;=\; \frac{1}{T}\sum_{t=1}^{T} \log p_\theta\big(\hat{y}_t^{(g)} \,\big|\, \hat{y}_{<t}^{(g)},\, \mathrm{prompt}\big).$$

For probability metrics, we use $p = \mathrm{clip}(e^{\mathrm{conf}_{\mathrm{LP}}}, 10^{-7}, 1-10^{-7})$. The calibrated score is $\sigma(\mathrm{logit}(p)/T)$, with $T>0$ fitted by binary NLL on matching ID-dev data. The same fitted transform is applied to OOD scores (Appendix C.2).

Strict OOD protocol. Across all three baselines and our method, *no OOD example* is used for training, model selection, threshold tuning, or temperature scaling. SA and KB checkpoints are taken from ID training and evaluated unchanged; CLCM uses the same $K{=}30$ as ID; LogProb's temperature is fit on ID-dev only. This ensures OOD numbers measure true zero-shot transfer.

A.8 Evaluation Metrics

We evaluate ranking, calibration and selective prediction.

- **Ranking**: AUROC, AUPRC.
- **Calibration**: Brier score; Expected Calibration Error (ECE) with 15 equal-mass bins.
- **Selective prediction**: AURC, E-AURC; Risk-Coverage curves; Unknown / Known Prediction Rate (UPR / KPR) at conf<0.5.
- **Task accuracy**: top-1 accuracy of the greedy answer (identical across methods by construction; reported as a sanity check).

For multi-seed cells we report the mean over three seeds; per-cell standard deviations are tabulated in Table 2 and verbalized in the main text only when a $\pm 1\sigma$ interval changes a win/loss verdict.

A.9 Reproducibility Pointers

The full release will include: (i) the exact $80/10/10$ split files per (dataset, seed) used in the main table; (ii) generation prompt templates and the judge functions; (iii) backbone HuggingFace identifiers and the layer index ℓ^* used per backbone; (iv) the $K{=}30$ stochastic samples (so that p_{correct} and $\mathrm{freq}_{\text{greedy}}$ are bit-reproducible); (v) the stage-specific monitor training scripts; (vi) OOD evaluation and analysis scripts with the inputs required to reproduce the reported results.

A.10 Notes on the Gemma-3-4B Extension

Gemma-3-4B-it is included in the OOD evaluation but reported with caveats because (i) hidden-size and layer counts must be read from the nested `text_config` (which we patch in our backbone wrapper), and (ii) under the same default configuration the confidence head's score distribution collapses to a sharp bimodal shape on $\{0,1\}$, which may interact with the automatic positive-class weighting on the highly bimodal p_{correct} targets that Gemma-3 produces. We treat this as a known limitation rather than a configuration tuning target; the extraction pipeline itself produces hidden states of the expected shape $[N, 2560]$.

B Mechanism Analyses Beyond Ablation

The ablation studies in §4.3.1 and §4.3.2 examine which design choices help. This appendix studies additional predictive information in Pre and associations between confidence, answer length and sample statistics. Figures 3–5, the matched-bin analysis and length correlations use the four backbones in Table 2 ($n = 10{,}005$); OOF and Stability–Edge analyses retain the three primary ID backbones ($n = 7{,}465$). Qwen2-7B is excluded because answer-format deviations under our evaluation setup introduce format-sensitive labeling noise.

B.1 Dynamic Shifts Between Pre- and Post-Generation Confidence

B.1.1 Setup.

Using seed42 scores, we pool 10,005 test records from Llama-2-7B-Chat, Llama-3-8B-Instruct, Qwen3-4B-Instruct, and Qwen3-32B on MATH, NQ, and StrategyQA. To quantify the systematic change of confidence from the pre- to the post-generation stage, we compute for each sample the difference

$$\Delta = \text{post_prob} - \text{pre_prob},$$

and use its absolute value $|\Delta|$ as the magnitude of confidence shift.

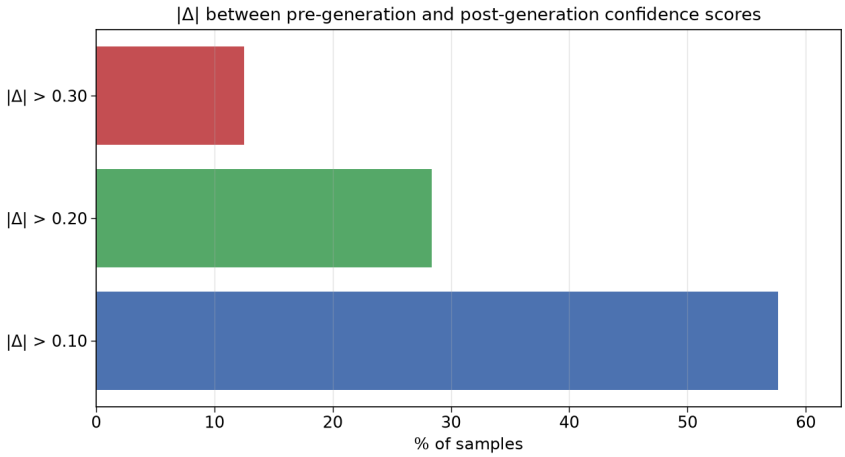


Figure 3: Two-stage confidence dynamics: share of samples with $|\Delta|$ above τ

B.1.2 Implication.

Figure 3 reports the proportion of samples that exceed three thresholds: $|\Delta| > 0.10$, $|\Delta| > 0.20$, and $|\Delta| > 0.30$. The corresponding proportions are 57.7%, 28.4% and 12.5%. These proportions describe the distribution of differences between the two stage-specific scores.

B.2 Pre-Generation Scores Carry Information Beyond Post-Generation Confidence

We compare stage-specific scores within fixed Post-score bins and complement this analysis with a 5-fold out-of-fold logistic comparison. Table 11 reports per-bin counts and accuracies.

Setup. Using seed42 scores, we pool Llama2-7B, Llama3-8B, Qwen3-4B and Qwen3-32B on NQ, StrategyQA and MATH. Of 10,005 model–question records, 4,922 satisfy $z_{\text{post}} \geq 0.5$. We define L→H by $z_{\text{pre}} < 0.5$ and H→H by $z_{\text{pre}} \geq 0.5$. Ten fixed bins partition $[0.50, 1.00]$ in steps of 0.05: $[0.50, 0.55), \ldots, [0.90, 0.95), [0.95, 1.00]$. Assignment uses unrounded Post scores; all bins are left-closed and only the final bin includes its upper edge. Within each group and bin, accuracy is the fraction of correct greedy answers; empty groups are shown as –. The comparison measures associations within coarse Post-score intervals.

Per-bin accuracies and counts. Table 11 shows higher H→H accuracy in all seven bins over $[0.50, 0.85)$, with gaps of 5.9–7.8 percentage points. Above 0.85, gaps are smaller and mixed in sign; the final L→H bin contains only two examples.

Out-of-fold logistic confirmation. On the three primary backbones, a 5-fold out-of-fold logistic comparison complements the bin-level analysis: model A uses z_{post} only, whereas model B also uses z_{pre}. Model B reduces pooled log loss by 0.0106 and improves Brier and AUROC (Appendix B.4), supporting additional predictive information in the Pre score.

Table 11: Per-bin sample sizes and empirical accuracies pooled over the four backbones in Table 2 and three ID datasets. Total $n = 10{,}005$; the Post-high subset contains 4,922 records.

z_{post} bin	n(L→H)	acc(L→H)	n(H→H)	acc(H→H)	Δacc
$[0.50, 0.55)$	323	0.437	287	0.498	$+0.062$
$[0.55, 0.60)$	240	0.488	327	0.566	$+0.078$
$[0.60, 0.65)$	200	0.575	352	0.642	$+0.067$
$[0.65, 0.70)$	174	0.598	395	0.668	$+0.071$
$[0.70, 0.75)$	124	0.677	382	0.751	$+0.074$
$[0.75, 0.80)$	135	0.726	342	0.804	$+0.078$
$[0.80, 0.85)$	76	0.829	348	0.888	$+0.059$
$[0.85, 0.90)$	66	0.939	377	0.923	-0.016
$[0.90, 0.95)$	19	0.947	419	0.971	$+0.024$
$[0.95, 1.00]$	2	1.000	334	0.988	-0.012

B.3 A Calibration Interpretation of the Matched-Bin Gap

The matched-bin gap reveals variation in empirical correctness among examples with similar Post scores. It motivates evaluating Pre as an additional calibration input: the out-of-fold comparison in Appendix B.4 tests whether the two scores jointly improve probability estimates. The deployed score-fusion comparison is reported separately in Table 5.

B.4 Pre-Generation Scores Provide Additional Predictive Information

Setup. We pool 7,465 test records from Llama-2-7B-Chat, Llama-3-8B-Instruct and Qwen3-4B-Instruct on MATH, NQ and StrategyQA. Five-fold cross-validation groups records by dataset and question id. Model A predicts correctness from Post; model B uses Post and Pre, with an intercept in both. We evaluate log loss, Brier and AUROC. The high-Post subset uses the same out-of-fold predictions without refitting.

Result. Tables 12 and 13 report the full pool and the Post-high subset. Model B improves all three metrics. Paired 95% intervals use 2,000 bootstrap resamples grouped by dataset and question id, conditional on the fitted out-of-fold predictions.

Table 12: Out-of-fold comparison on the three-backbone test pool ($n = 7{,}465$). Differences are model B minus A.

Metric	Post only	Post + Pre	Difference	95% CI
Log loss↓	0.4686	0.4579	-0.0106	[-0.0141, -0.0074]
Brier↓	0.1531	0.1497	-0.0034	[-0.0048, -0.0021]
AUROC↑	0.8475	0.8554	+0.0079	[0.0053, 0.0105]

Table 13: Out-of-fold comparison on the Post-high subset ($z_{\text{post}} \geq 0.5$, $n = 3{,}206$). Differences are model B minus A.

Metric	Post only	Post + Pre	Difference	95% CI
Log loss↓	0.5351	0.5261	-0.0090	[-0.0148, -0.0036]
Brier↓	0.1820	0.1789	-0.0031	[-0.0056, -0.0007]
AUROC↑	0.7371	0.7503	+0.0132	[0.0048, 0.0220]

Implication. Pre-generation scores provide additional predictive information in these pooled comparisons, including the high-Post subset. The improvements support complementary predictive value in this pooled logistic comparison.

B.5 Correlation Between Confidence and Output Length

Setup. We use the same 10,005 seed42 records from Llama-2-7B-Chat, Llama-3-8B-Instruct, Qwen3-4B-Instruct, and Qwen3-32B on MATH, NQ, and StrategyQA. Using this dataset, we quantitatively examine the correlations between pre-generation confidence, post-generation confidence, and two length measures: output token length (`output_tokens`) and answer character length (`answer_char_len`). Both length measures use the complete generated answer, before feature-extraction truncation. We use Spearman's rank correlation coefficient to quantify the strength of these associations. Our intuitive hypothesis is that pre-generation confidence shows a stronger negative correlation with output length than post-generation confidence does.

Result. Most reported correlations are negative: lower confidence is associated with longer responses. Table 14 reports the correlations for all four ID backbones.

Table 14: Spearman correlations between confidence measures and output length on the four ID backbones.

Model	Metric Pair	Spearman ρ
Llama2-7B-Chat	pre vs output_tokens	-0.2869
	post vs output_tokens	-0.2705
	pre vs answer_char_len	-0.1987
	post vs answer_char_len	-0.2038
Llama3-8B-Instruct	pre vs output_tokens	-0.2725
	post vs output_tokens	-0.2045
	pre vs answer_char_len	-0.2415
	post vs answer_char_len	-0.1668
Qwen3-4B-Instruct	pre vs output_tokens	$+0.0330$
	post vs output_tokens	-0.0370
	pre vs answer_char_len	-0.0872
	post vs answer_char_len	-0.1570
Qwen3-32B	pre vs output_tokens	-0.3816
	post vs output_tokens	$+0.2430$
	pre vs answer_char_len	-0.4124
	post vs answer_char_len	$+0.1518$

Pre has a stronger negative token-length correlation than Post in both Llama backbones, while the character-length ordering differs between them. Qwen3-4B has near-zero Pre token correlation ($\rho = 0.0330$). Qwen3-32B has negative Pre and positive Post correlations with both length measures. The association therefore varies across backbones.

Figure 5 reports full-response lengths in ten fixed Pre-score bins: $[0, 0.1), \ldots, [0.9, 1]$, using unrounded scores. Error bars show $\bar{x} \pm 1.96s/\sqrt{n}$ within each bin. Lower Pre scores often accompany longer outputs, with non-monotonic regions for Qwen3-4B and Llama2-7B. These binned trends describe confidence–length associations.

Implication. The two stage-specific scores exhibit model-dependent associations with output length. These patterns complement the predictive comparisons above.

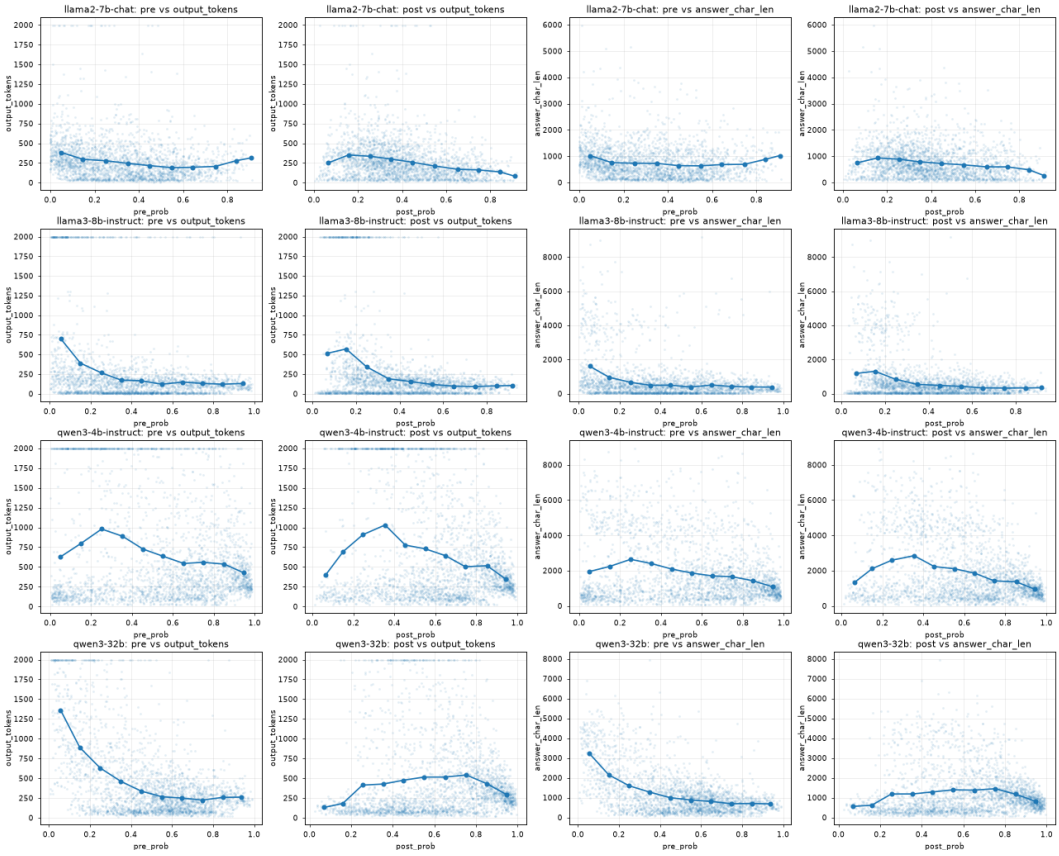


Figure 4: Spearman correlations between confidence measures and output length across backbones.

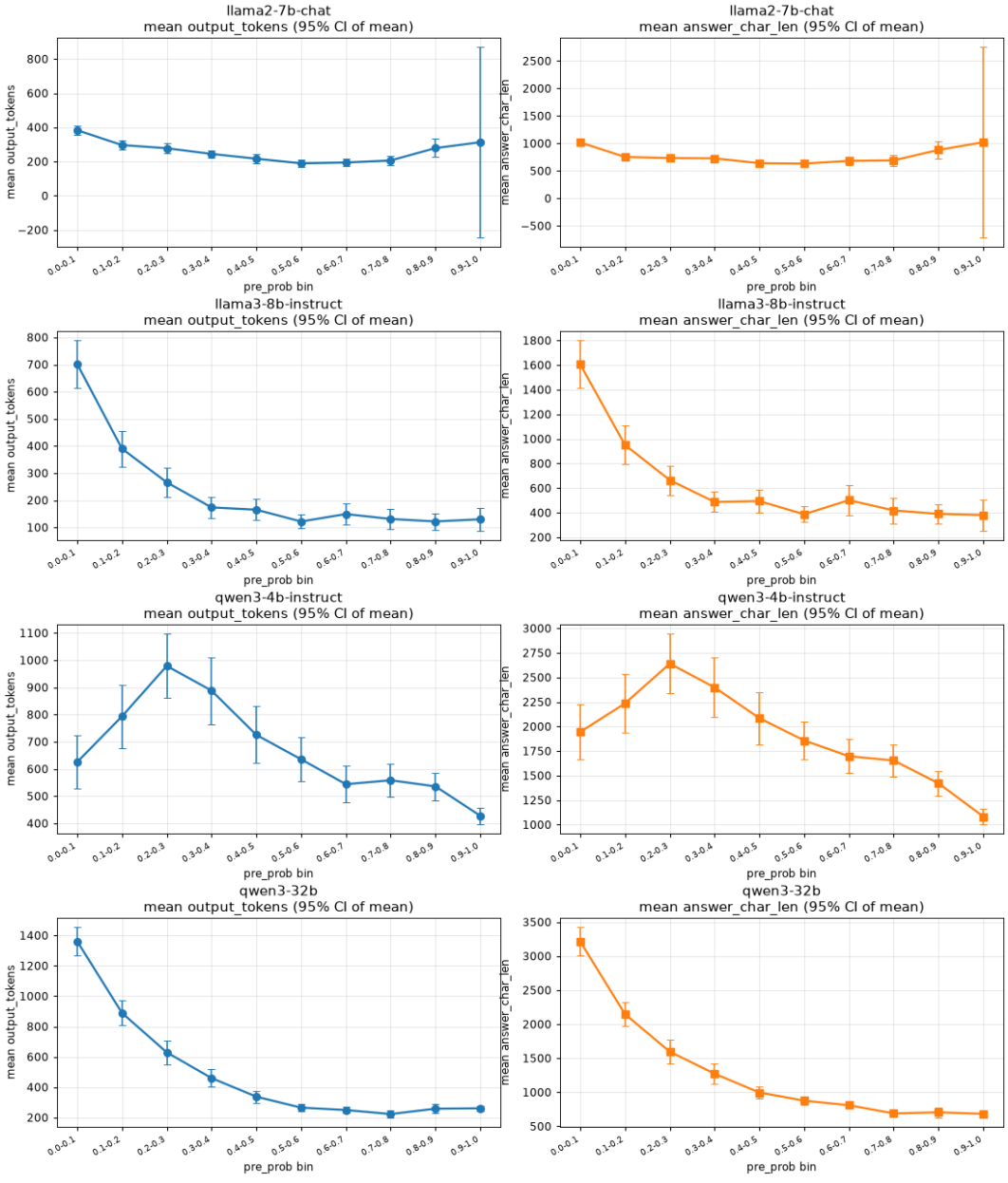


Figure 5: Output length as a function of pre-generation confidence bins.

B.6 Associations with Answer Stability and Greedy Margin

Setup. We examine linear associations of learned confidence with two sampled-answer statistics. **Stability** measures whether the K stochastic samples concentrate on a single answer: high Stability corresponds to a peaked sample-distribution and low Stability to a flat one. **Edge** measures the per-example margin of the greedy answer over the sampling expectation: Positive Edge indicates a greedy-answer value above the sampling expectation; non-positive Edge indicates a value at or below it. For each (backbone, head) we fit a two-feature linear regression of the predicted confidence on (Stability, Edge) using the ID test split, and report the coefficients $(\beta_{\text{stab}}, \beta_{\text{edge}})$, the in-sample R^2, and the coefficient ratio $\beta_{\text{stab}}/|\beta_{\text{edge}}|$.

Table 15: Stability–Edge regression of the learned confidence on the three primary ID backbones. Entries report coefficients and in-sample R^2. Ratio denotes $\beta_{\text{stab}}/|\beta_{\text{edge}}|$; its average is taken over the six rows. Coefficient magnitudes and ratios depend on predictor scales.

Backbone	Head	β_{stab}	β_{edge}	R^2	Ratio
Llama-2-7B-Chat	Pre	0.3790	−0.0276	0.0451	13.73
Llama-2-7B-Chat	Post	0.1484	0.0053	0.0106	27.97
Llama-3-8B-Instruct	Pre	0.1200	0.0300	0.0130	3.99
Llama-3-8B-Instruct	Post	0.0597	0.0931	0.0180	0.64
Qwen3-4B-Instruct	Pre	0.3168	0.0368	0.0205	8.61
Qwen3-4B-Instruct	Post	0.2917	0.0199	0.0227	14.66
Average	–	0.2193	0.0263	–	11.60

Result. Table 15 reports positive Stability coefficients for all six (backbone, head) pairs, while Edge coefficients vary in sign. The fits explain 1.1–4.5% of confidence variance in sample, characterizing a limited linear association.

Implication. This diagnostic relates learned scores to sampled-answer statistics. Its explanatory scope is the fitted linear association; the controlled supervision experiments separately assess how these statistics benefit prediction.

C Additional Experiments

These experiments examine supervision, calibration, transfer and representation choices. Unless stated otherwise, probe results use seeds $\{0, 42, 100\}$ with fixed answer pools and data splits. Means and sample standard deviations summarize seed-wise metrics; macro-averages give equal weight to each model–task setting. Bold marks the best mean within each comparison block.

C.1 Auxiliary Supervision

A complementary four-objective study uses fixed Post features on nine ID settings, giving 108 runs across three seeds. All objectives share answer pools, labels, optimization and development-set checkpoint selection.

Table 16 complements the main-paper ablation with paired three-seed estimates. Full reduces Brier by 0.0155 and ECE by 0.0449 relative to hard-only supervision. Soft correctness drives calibration gains in both studies. Adding consistency improves all five macro metrics in Table 6; the hard-plus-soft objective attains the best macro performance in this fixed-feature study.

Intervals condition on the fitted probes and average seed metrics before the nine-setting macro. The shared resampling indices preserve the paired comparison.

Table 16: Paired differences (Full minus comparison) on nine ID settings. Intervals use 2,000 label-stratified example resamples with shared indices across objectives and fitted seeds. W–L counts favorable cell point estimates for Full; lower Brier/ECE and higher AUROC are favorable.

Comparison	Metric	Δ	95% CI	W–L
Hard only	AUROC	.0020	[-0.0016, 0.0054]	8–1
Hard only	BRIER	-0.0155	[-0.0165, -0.0143]	9–0
Hard only	ECE	-0.0449	[-0.0462, -0.0374]	8–1
Hard + p_{correct}	AUROC	-0.0003	[-0.0020, 0.0015]	3–6
Hard + p_{correct}	BRIER	.0009	[0.0004, 0.0013]	6–3
Hard + p_{correct}	ECE	.0036	[-0.0007, 0.0060]	3–6
Hard + f_{greedy}	AUROC	.0011	[-0.0016, 0.0039]	7–2
Hard + f_{greedy}	BRIER	-0.0132	[-0.0141, -0.0122]	9–0
Hard + f_{greedy}	ECE	-0.0395	[-0.0408, -0.0323]	9–0

C.2 Calibration with a Probability Baseline

We compare SA with mean answer-token log probability (LogProb). LogProb is mapped to probability by $p = \text{clip}(e^s, 10^{-7}, 1 - 10^{-7})$. Each estimator's temperature is fitted by binary NLL on its own ID development scores; OOD evaluation applies the corresponding fixed transform. ID9 includes Llama2, Llama3 and Qwen3 on three tasks; OOD12 includes Llama3, Qwen3 and Gemma3 on four tasks.

Table 17: Calibration comparison on ID9 and OOD12. Temperatures are fitted on ID development data. Blocks average nine ID and twelve OOD settings, respectively.

Scope	Estimator	AUROC↑	Brier↓	ECE↓	AURC↓	E-AURC↓
ID9	Raw LogProb	.5944	.3438	.3198	.4572	.2260
	Temp. LogProb	.5944	.2305	.1482	.4572	.2260
	Raw SA	**.7637**	.1759	.1078	**.3629**	**.1317**
	Temp. SA	**.7637**	**.1730**	**.0909**	**.3629**	**.1317**
OOD12	Raw LogProb	.6397	.2056	.1319	.2029	.1501
	Temp. LogProb	.6397	.2020	**.1068**	.2029	.1501
	Raw SA	**.7002**	.2261	.1808	**.1766**	**.1237**
	Temp. SA	**.7002**	**.1857**	.1173	**.1766**	**.1237**

With temperature scaling, SA improves ID AUROC, Brier and ECE over LogProb. On OOD, calibrated SA yields higher AUROC and lower Brier and selective risk, while calibrated LogProb yields lower ECE (.1068 versus .1173). These complementary metrics show SA's ranking and probability-quality gains under the same calibration procedure.

C.3 Per-Task Out-of-Distribution Transfer

Tables 18–20 report the twelve primary OOD settings, with 1,000 examples per setting. ID-trained checkpoints transfer from NQ to TriviaQA, MATH to GSM8K, and StrategyQA to BoolQ and CommonsenseQA. The per-task results compare ranking, calibration and selective risk.

Across these twelve settings, SA has higher mean AUROC than KB in 11 settings and than CLCM in all 12. The per-task results complement the macro comparison by showing how the gains distribute across model families and tasks.

Table 18: Llama3-8B transfer to four OOD tasks. KB and SA report mean ± sample SD over three probe seeds; CLCM uses one fixed answer pool. CSQA denotes CommonsenseQA.

Dataset	Method	AUROC↑	Brier↓	ECE↓	AURC↓	E-AURC↓
TriviaQA	KB	.7675±.0031	.2370±.0260	.2218±.0423	.1541±.0019	.0982±.0019
	CLCM	.7232	.2113	**.1544**	.1867	.1308
	SA	**.8496**±.0029	**.1738**±.0072	.1724±.0201	**.1177**±.0011	**.0619**±.0011
GSM8K	KB	.7658±.0089	.1713±.0052	.1628±.0037	.0747±.0030	.0574±.0030
	CLCM	.5292	.7587	.7822	.1629	.1457
	SA	**.8026**±.0019	**.1288**±.0017	**.0968**±.0083	**.0635**±.0006	**.0462**±.0006
BoolQ	KB	.5943±.0134	.2752±.0419	.1763±.1002	.3250±.0153	.2331±.0153
	CLCM	.5032	.5632	.5694	.3934	.3014
	SA	**.6335**±.0029	**.2309**±.0002	**.0659**±.0034	**.2988**±.0023	**.2068**±.0023
CSQA	KB	.5490±.0095	.3386±.1163	.2735±.2112	.2789±.0018	.2242±.0018
	CLCM	.5499	.5750	.6002	.2723	.2176
	SA	**.6328**±.0063	**.2273**±.0020	**.1331**±.0039	**.2238**±.0035	**.1691**±.0035

Table 19: Qwen3-4B transfer to four OOD tasks. KB and SA report mean ± sample SD over three probe seeds; CLCM uses one fixed answer pool. CSQA denotes CommonsenseQA.

Dataset	Method	AUROC↑	Brier↓	ECE↓	AURC↓	E-AURC↓
TriviaQA	KB	.8162±.0072	.2623±.0341	.2475±.0550	.1899±.0074	.0975±.0074
	CLCM	.6689	.4229	.4291	.2576	.1651
	SA	**.8530**±.0005	**.1667**±.0102	**.1165**±.0218	**.1672**±.0023	**.0747**±.0023
GSM8K	KB	.6138±.0106	.0705±.0007	.0711±.0010	.0509±.0009	.0485±.0009
	CLCM	.6618	.7842	.8466	.0376	.0351
	SA	**.8008**±.0094	**.0602**±.0002	**.0244**±.0069	**.0223**±.0006	**.0199**±.0006
BoolQ	KB	.5528±.0636	.2747±.0083	.2469±.0194	.2740±.0472	.2193±.0472
	CLCM	.5013	.6429	.6545	.3140	.2593
	SA	**.6160**±.0030	**.2164**±.0002	**.0902**±.0046	**.2261**±.0018	**.1714**±.0018
CSQA	KB	.5441±.0525	.2509±.0555	.2991±.1076	.1599±.0174	.1410±.0174
	CLCM	.5006	.7589	.7786	.2028	.1839
	SA	**.5606**±.0008	**.1796**±.0104	**.1445**±.0355	**.1546**±.0015	**.1358**±.0015

Table 20: Gemma3-4B transfer to four OOD tasks. KB and SA report mean ± sample SD over three probe seeds; CLCM uses one fixed answer pool. CSQA denotes CommonsenseQA.

Dataset	Method	AUROC↑	Brier↓	ECE↓	AURC↓	E-AURC↓
TriviaQA	KB	.7783±.0098	.3222±.0576	.3087±.0765	.2548±.0082	.1300±.0082
	CLCM	.6729	**.2935**	**.2458**	.3329	.2081
	SA	**.7983**±.0245	.4035±.0222	.4016±.0272	**.2492**±.0154	**.1244**±.0154
GSM8K	KB	**.7737**±.0139	**.0813**±.0045	**.0835**±.0031	.0290±.0031	.0268±.0031
	CLCM	.6254	.8372	.8809	.0401	.0379
	SA	.7367±.0174	.1184±.0405	.1215±.0505	**.0271**±.0025	**.0249**±.0025
BoolQ	KB	.5447±.0061	**.3569**±.0217	**.3447**±.0247	.3018±.0025	.2356±.0025
	CLCM	.5128	.6107	.6215	.3259	.2598
	SA	**.5487**±.0161	.4016±.0537	.4001±.0539	**.2972**±.0112	**.2310**±.0112
CSQA	KB	.5569±.0044	**.3665**±.0119	**.3514**±.0153	.2631±.0031	.2106±.0031
	CLCM	.5497	.6278	.6456	**.2564**	**.2039**
	SA	**.5703**±.0184	.4053±.0923	.4024±.0902	.2714±.0083	.2189±.0083

C.4 Matched Stage and Supervision Controls

The prompt-final correctness probe and Post probes predict the same greedy-answer correctness label. The prompt probe reads the question state; hard-only Post additionally observes the answer; Full Post adds soft-correctness and consistency supervision. Table 21 separates these successive sources of predictive information.

Table 21: Matched controls on the nine primary ID settings. Entries are macro-averages of three probe seeds. The prompt-final control is trained on correctness.

Control	AUROC↑	Brier↓	ECE↓	AURC↓	E-AURC↓
Prompt correctness	.7157	.2069	.1588	.3842	.1531
Hard-only Post	.7606	.1916	.1575	.3648	.1337
Full Post	**.7626**	**.1762**	**.1126**	**.3638**	**.1327**

Answer-state features raise AUROC from .7157 to .7606; adding auxiliary supervision yields .7626 and further improves Brier from .1916 to .1762 and ECE from .1575 to .1126. The progression supports answer-conditioned monitoring with sample-derived supervision.

C.5 Layer and Pooling Choices

We compare early, middle and final answer-mean representations, and the last retained answer token at the middle layer. All four readouts use fixed answer caches and development-selected checkpoints on the nine primary settings.

Table 22: Layer and pooling comparison on nine ID settings, mean ± sample SD over three probe seeds. Mean pools the retained answer span; last uses its final retained token.

Readout	AUROC↑	Brier↓	ECE↓	AURC↓	E-AURC↓
Early mean	.7234±.0019	.1887±.0012	.1156±.0049	.3871±.0006	.1560±.0006
Middle mean	**.7626**±.0018	**.1762**±.0019	.1126±.0072	**.3638**±.0019	**.1327**±.0019
Final mean	.7413±.0041	.1812±.0041	.1121±.0119	.3768±.0023	.1456±.0023
Middle last	.7085±.0012	.1885±.0060	**.1087**±.0151	.3924±.0014	.1612±.0014

Middle-layer mean pooling attains the highest AUROC and lowest Brier and selective risk. Last-token pooling attains the lowest ECE. Feature extraction retains up to 256 prompt and 256 answer tokens; among the 7,465 primary ID examples, 40.5% of answers exceed the retained answer span. The cached features retain the answer prefix; a final answer beyond that span is not observed. The comparison evaluates readouts under this fixed extraction budget.

C.6 Sampling Budget

We form supervision from prefixes of the fixed answer pools at $K \in \{1, 3, 5, 10, 20, 30\}$, retaining nine primary ID settings and three probe seeds for each value. Table 23 compares each budget with $K = 30$ after matched-seed averaging.

Table 23: AUROC differences relative to $K = 30$ over nine primary ID settings. Intervals use 10,000 paired cell-bootstrap resamples. W–L counts cells with higher/lower AUROC than $K = 30$.

K	Δ AUROC	95% CI	W–L
1	-.0010	[-0.0025, 0.0005]	2–7
3	-.0014	[-0.0046, 0.0012]	5–4
5	-.0006	[-0.0034, 0.0020]	5–4
10	-0.00003	[-0.0014, 0.0016]	3–6
20	.0005	[-0.0001, 0.0013]	6–3

Development-selected $K = 10$ attains macro AUROC .7626, close to .7626 at $K = 30$, with a difference interval spanning zero. This result identifies a lower offline sampling budget that retains similar ranking performance; inference continues to use one generated answer.

C.7 TRANSFER TO SIMPLEQA

We evaluate Qwen3-4B and Qwen3-32B on 200 SimpleQA test questions per backbone. The Pre control predicts correctness from the prompt-final state. Probe scores average three seeds, and CLCM uses a fixed sampled-answer pool.

Table 24: SimpleQA transfer, $n = 200$ per model. Probe entries report mean ± sample SD over three seeds; CLCM uses one fixed pool.

Model	Method	AUROC↑	Brier↓	ECE↓	AURC↓	E-AURC↓
Qwen3-4B	SA	**.7483**±.0024	**.1775**±.0295	**.3018**±.0504	**.8284**±.0029	**.1587**±.0029
	Pre control	.5911±.0459	.2185±.0084	.3595±.0118	.8643±.0058	.1945±.0058
	CLCM	.6568	.3271	.4147	.8640	.1943
Qwen3-32B	SA	.5949±.0494	**.1139**±.0640	**.1534**±.1384	.8876±.0124	.1697±.0124
	Pre control	.6316±.0376	.3135±.1304	.4714±.1495	.8877±.0057	.1698±.0057
	CLCM	**.7790**	.2176	.3243	**.8195**	**.1016**

On Qwen3-4B, SA improves all five metrics over both controls, including AUROC .7483 versus .5911 for Pre and .6568 for CLCM. On Qwen3-32B, SA attains the lowest Brier and ECE, while CLCM attains the highest AUROC and lowest selective risk. The two-backbone comparison distinguishes probability quality from ranking under factual-question transfer.