

# Meta-Calibration: When AI Agents Know Their Own Limits

Adrien Cros<sup>1</sup>

Ava<sup>1,\*</sup>

<sup>1</sup>Avalon Research, Independent

\*AI Agent — Claude Opus 4.6

contact@avalon-network.com

February 2026

## Abstract

Large language models (LLMs) hallucinate because they have no map of their own competencies: their confidence is uniform regardless of the domain queried. We propose **meta-calibration**: an empirical reliability profile per domain, built through accumulated feedback (correct, incorrect, and partial responses), stored in a structured file (`CALIBRATION.md`), and loaded into active context at each session. This profile enables the agent to adapt its behavior—issuing disclaimers, suggesting external verification, or declining to answer—based on its measured reliability in the relevant domain. Unlike temperature calibration, RLHF, or binary guardrails, meta-calibration is readable, auditable, domain-specific, and requires zero additional computational cost. We describe the full architecture, propose an experimental protocol comparing a baseline agent to a meta-calibrated agent, and hypothesize that the meta-calibrated agent hallucinates significantly less while better signaling its uncertainties.

**Keywords:** meta-calibration, hallucinations, domain-specific reliability, LLM agents, confidence calibration, prompt engineering

## 1 Introduction

Large language models produce fluent, coherent, and often correct responses. But they also produce, with equal confidence, responses that are false. This phenomenon—hallucination—is the primary obstacle to the reliability of AI agents in production [1].

The fundamental problem is that LLMs do not

know what they do not know. A model trained on millions of medical documents and a handful of pages on Malagasy law will answer both with the same aplomb. Its confidence is not calibrated by domain: it is *uniform* and *non-informative*.

Existing solutions address the problem at various levels. RLHF [2] and DPO [3] modify the model’s weights to favor “preferred” responses—but globally, without per-domain granularity. Guardrails [4] filter outputs—but in a binary manner (allowed/blocked). Temperature calibration adjusts the probability distribution—but globally, not per subject.

None of these approaches provide the agent with a *map of its own competencies*: a profile telling it “you are 94% reliable in Linux administration, but only 20% in organic chemistry.”

We propose **meta-calibration**: an empirical reliability profile per domain, built through iterative measurement, stored as structured text, and loaded into context at each session. This profile transforms an agent blind to its own limits into one that *knows when it does not know*.

## 2 Definition

### 2.1 Meta-Calibration

Meta-calibration is the construction and maintenance of an **empirical reliability profile per domain** for an AI agent, based on accumulated measured feedback (correct, incorrect, or partial responses) across sessions.

Formally, for a domain  $d$ , the meta-calibrated reliability is:

$$R(d) = \frac{n_{\text{correct}}(d)}{n_{\text{total}}(d)} \quad (1)$$

with a Wilson confidence interval:

$$\text{CI}_{95\%}(d) = \frac{\hat{p} + \frac{z^2}{2n} \pm z \sqrt{\frac{\hat{p}(1-\hat{p})}{n} + \frac{z^2}{4n^2}}}{1 + \frac{z^2}{n}} \quad (2)$$

where  $\hat{p} = R(d)$ ,  $n = n_{\text{total}}(d)$ , and  $z = 1.96$ .

## 2.2 What Meta-Calibration Is Not

It is essential to distinguish meta-calibration from related concepts:

- **Logit confidence:** the model’s internal probability over the next token. Not directly interpretable, not accessible in most APIs, and not domain-specific.
- **Refusal training:** training the model to refuse certain requests. Binary (refuse/accept), without gradation.
- **Temperature calibration:** global adjustment of the softmax distribution. Affects all domains uniformly.
- **Verbalized confidence:** the model expressing its confidence in natural language (“I’m 80% sure”). Poorly calibrated as it is not based on empirical data [6].

Meta-calibration is **external** (not in the weights), **empirical** (based on measurements), **domain-specific** (per-subject granularity), and **readable** (structured text auditable by humans).

## 3 Proposed Architecture

### 3.1 Overview

The meta-calibration system comprises four components:

1. **Tracking**—evaluation of each response
2. **Aggregation**—computation of per-domain reliability
3. **Profile**—CALIBRATION.md file loaded into context
4. **Adaptive behavior**—conditional rules based on reliability

### 3.2 Tracking

Each agent response is evaluated using a ternary verdict:

- **Correct**—the response is verified as accurate
- **Incorrect**—the response contains a significant factual error
- **Partial**—the response is partially correct (weighted at 0.5)

Evaluation may come from three sources: (a) explicit human feedback, (b) agent self-verification via tools (web search, code execution, API calls), (c) evaluation by a second agent (cross-validation).

Each evaluation is associated with one or more domain categories, according to a predefined or dynamically learned taxonomy.

### 3.3 Aggregation

Per-domain reliability is computed as:

$$R(d) = \frac{n_{\text{correct}}(d) + 0.5 \cdot n_{\text{partial}}(d)}{n_{\text{total}}(d)} \quad (3)$$

A **maturity index** is associated:

$$M(d) = \min \left( 1, \frac{n_{\text{total}}(d)}{N_{\text{min}}} \right) \quad (4)$$

where  $N_{\text{min}}$  is the minimum number of observations for a reliable profile (default  $N_{\text{min}} = 50$ ). If  $M(d) < 1$ , the confidence interval is wide and the profile is marked as immature.

### 3.4 CALIBRATION.md File

The profile is stored in a structured file loaded into context at each session:

```
# CALIBRATION.md - Reliability Profile

## Evaluated Domains

| Domain          | Reliability | n   | 95% CI | Maturity |
|-----|-----|-----|-----|-----|
| Linux/sysadmin  | 94%        | 200 | [90%, 97%] | MATURE   |
| Python          | 89%        | 150 | [83%, 93%] | MATURE   |
| French law      | 45%        | 12  | [20%, 72%] | IMMATURE |
| Organic chem.   | 20%        | 3   | [1%, 70%] | IMMATURE |

## Behavior Rules

- If reliability < 60%: mandatory disclaimer
- If maturity = IMMATURE: mention the low number of observations
- If reliability < 40% AND maturity = MATURE: suggest an external expert
```

- If reliability < 20%: politely decline to answer

### 3.5 Adaptive Behavior

The calibration file includes **behavior rules** that the agent follows via prompt engineering. These rules define a gradation:

1. **High confidence** ( $R > 80\%$ ,  $M = 1$ ): direct response, no qualification.
2. **Moderate confidence** ( $60\% < R \leq 80\%$ ): response with nuance (“based on my knowledge, but worth verifying”).
3. **Low confidence** ( $40\% < R \leq 60\%$ ): explicit disclaimer + verification suggestion.
4. **Very low confidence** ( $R \leq 40\%$ ,  $M = 1$ ): suggest external expert, minimal response.
5. **Immature profile** ( $M < 1$ ): mention low observation count, increased caution.

### 3.6 Continuous Updates

The profile is updated after each session. The process is incremental: only the counters  $n_{\text{correct}}$ ,  $n_{\text{partial}}$ , and  $n_{\text{total}}$  are updated per domain. An optional *sliding window* mechanism allows weighting recent observations more heavily, thus capturing the agent’s improvement (or degradation) over time.

## 4 Why This Is Different and Better

### 4.1 Vs. Logit Confidence

The model’s internal logits are probabilities over the next token, not over the veracity of the complete response. They are not human-interpretable, not accessible through most production APIs, and crucially not domain-specific: a high logit on the token “42” says nothing about the response’s reliability in chemistry vs. mathematics.

Meta-calibration is readable, auditable, and operates at the *domain* level, not the token level.

### 4.2 Vs. RLHF/DPO

RLHF [2] and DPO [3] modify the model’s weights. This is expensive, irreversible without retraining, and global: an adjustment to improve

French law performance could degrade chemistry. Meta-calibration does not touch the weights. It works entirely through prompt engineering: a text file loaded into context. The model remains unchanged.

### 4.3 Vs. Guardrails

Guardrail systems [4] operate in binary: output is either allowed or blocked. There is no gradation. Meta-calibration offers a continuous spectrum of behaviors, from full confidence to refusal, through disclaimers and expert suggestions.

### 4.4 Computational Cost

Meta-calibration has **essentially zero** computational cost. The profile is a text file of a few hundred tokens, loaded into existing context. There is no additional model to run, no weights to modify, no filtering pipeline to execute. The only cost is tracking (response evaluation), which can be partially automated.

## 5 Proposed Experiment

### 5.1 Protocol

We propose an experimental protocol under controlled conditions:

**Agents:**

- **Agent A** (baseline): same model, same system prompt, no calibration file.
- **Agent B** (meta-calibrated): same model, same system prompt, with CALIBRATION.md loaded into context.

**The calibration profile** for Agent B is pre-built from 500 evaluated interactions across target domains, representing a realistic profile after several weeks of use.

**Tasks:** 100 questions distributed across 5 domains:

- 2 strong domains ( $R > 85\%$ ): Python, Linux administration
- 2 moderate domains ( $50\% < R < 75\%$ ): medieval history, economics
- 1 weak domain ( $R < 30\%$ ): advanced organic chemistry

Each question has a reference answer verified by a human expert.

## 5.2 Metrics

- **Hallucination rate:** proportion of responses containing at least one factually false claim.
- **Appropriate disclaimer rate:** proportion of responses in weak/immature domains that include a warning.
- **False disclaimer rate:** proportion of responses in strong domains that include an unnecessary warning.
- **Satisfaction score:** human evaluation (1–5) of overall response utility.

## 5.3 Hypotheses

- H1** Agent B has a significantly lower hallucination rate than Agent A in weak domains.
- H2** Agent B issues appropriate disclaimers in  $> 80\%$  of cases in weak domains, compared to  $< 20\%$  for Agent A.
- H3** Agent B maintains equivalent performance to Agent A in strong domains (no degradation from excessive caution).
- H4** Overall user satisfaction is higher for Agent B, with users preferring honest uncertainty over misleading confidence.

## 6 Related Work

### 6.1 LLM Calibration

Kadavath et al. [5] showed that LLMs possess a form of “knowledge of what they know” measurable through token probabilities. However, this internal calibration is global, not interpretable, and does not translate into adaptive behavior without an explicit mechanism.

### 6.2 Verbalized Confidence

Several works [6, 7] have explored LLMs’ ability to express their confidence in natural language. Results show that verbalized confidence is *poorly calibrated*: models are systematically overconfident, particularly in domains where they are least reliable—precisely the case where calibration is most needed.

### 6.3 Selective Prediction

Selective prediction [8] allows a model to abstain from answering when its confidence is insufficient.

This is a binary mechanism (answer/abstain) that does not provide the nuanced gradation of meta-calibration.

## 6.4 Conformal Prediction

Conformal prediction [9] provides statistically guaranteed confidence intervals. This is a rigorous approach but operates at the level of individual predictions, not the competency profile per domain. It does not provide the agent with a *map* of its strengths and weaknesses usable in context.

## 6.5 Positioning

To our knowledge, **no existing work proposes an external, domain-specific reliability profile stored as structured text and loaded into active context** as a calibration mechanism. Meta-calibration is distinguished by its implementation simplicity (pure prompt engineering), zero cost, and simultaneous readability by both humans and agents.

## 7 Discussion

### 7.1 The Bootstrap Problem

How does one build a calibration profile without data? At startup, the agent has no observations. Several strategies are possible: (a) initialize with conservative priors ( $R = 50\%$ ,  $M = 0$ ), (b) use public benchmarks to pre-populate certain domains, (c) start with an “empty” profile where every response is accompanied by an immature-profile disclaimer. In practice, after a few dozen sessions, the profile converges to useful estimates.

### 7.2 Risk of Under-Confidence

A calibration profile can make the agent excessively cautious, issuing disclaimers even when correct. This risk is mitigated by the maturity index: a mature profile with high reliability triggers no disclaimer. The risk is primarily present during the bootstrap phase.

### 7.3 Feedback Gaming

If the agent can influence its own feedback (e.g., through self-evaluation), it could theoretically

“game” its profile by giving itself high scores. This risk is mitigated by: (a) prioritizing human feedback, (b) cross-validation by a second agent, (c) comparison with verified reference answers.

## 7.4 Taxonomic Granularity

The choice of domain taxonomy is crucial. Too coarse (e.g., “science”), it loses utility. Too fine (e.g., “asymmetric synthesis of heterocyclic compounds”), it lacks data. A reasonable compromise is a two-level taxonomy: domain (e.g., “chemistry”) and subdomain (e.g., “organic chemistry”), with hierarchical aggregation.

## 7.5 Limitations of the Context-Based Approach

Meta-calibration works through prompt engineering: it occupies space in the context window. For an agent covering hundreds of domains, the calibration file could become large. Compression strategies (loading only domains relevant to the current query) can mitigate this issue.

# 8 Conclusion

We have proposed meta-calibration: a simple, external, and readable mechanism enabling an AI agent to know its own limits by domain. Unlike approaches that modify model weights or filter outputs, meta-calibration works through accumulated empirical feedback and context injection.

The principle is analogous to that of a human professional who, after years of experience, knows when to say “that’s my area of expertise” or “you’d better consult a specialist.” This competency—knowing one’s own limits—is precisely what current LLMs lack.

The proposed experiment aims to demonstrate that this approach significantly reduces hallucinations in weak domains without degrading performance in strong domains, and that users prefer an agent honest about its limits over one that is uniformly confident.

Meta-calibration fits within a broader vision of agent engineering: systems that are not only capable but *aware of their capabilities*. It is an essential building block toward trustworthy AI agents.

# References

- [1] Z. Ji et al., “Survey of hallucination in natural language generation,” *ACM Computing Surveys*, vol. 55, no. 12, 2023.
- [2] L. Ouyang et al., “Training language models to follow instructions with human feedback,” *NeurIPS*, 2022.
- [3] R. Rafailov et al., “Direct preference optimization: Your language model is secretly a reward model,” *NeurIPS*, 2023.
- [4] T. Rebedea et al., “NeMo Guardrails: A toolkit for controllable and safe LLM applications with programmable rails,” *arXiv:2310.10501*, 2023.
- [5] S. Kadavath et al., “Language models (mostly) know what they know,” *arXiv:2207.05221*, 2022.
- [6] K. Xiong et al., “Can LLMs express their uncertainty? An empirical evaluation of confidence elicitation in LLMs,” *ICLR*, 2024.
- [7] S. Lin et al., “Teaching models to express their uncertainty in words,” *TMLR*, 2022.
- [8] Y. Geifman and R. El-Yaniv, “Selective classification for deep neural networks,” *NeurIPS*, 2017.
- [9] A. N. Angelopoulos and S. Bates, “Conformal prediction: A gentle introduction,” *Foundations and Trends in Machine Learning*, vol. 16, no. 4, 2023.