

Méta-calibration : quand les agents IA connaissent leurs propres limites

Adrien Cros¹

Ava^{1,*}

¹Avalon Research, Indépendant

*Agent IA — Claude Opus 4.6

contact@avalon-network.com

Février 2026

Résumé

Les grands modèles de langage (LLM) hallucinent parce qu'ils ne disposent d'aucune carte de leurs propres compétences : leur confiance est uniforme quel que soit le domaine interrogé. Nous proposons la **méta-calibration** : un profil empirique de fiabilité par domaine, construit par accumulation de feedback (réponses correctes, incorrectes, partielles), stocké dans un fichier structuré (`CALIBRATION.md`) et chargé en contexte actif à chaque session. Ce profil permet à l'agent d'adapter son comportement — émettre des disclaimers, suggérer une vérification externe, ou refuser de répondre — en fonction de sa fiabilité mesurée dans le domaine concerné. Contrairement aux approches par calibration de température, RLHF ou guardrails binaires, la méta-calibration est lisible, auditable, domain-specific, et ne nécessite aucun coût computationnel supplémentaire. Nous décrivons l'architecture complète, proposons un protocole expérimental comparant un agent baseline à un agent méta-calibré, et formulons l'hypothèse que l'agent méta-calibré hallucine significativement moins tout en signalant mieux ses incertitudes.

Mots-clés : méta-calibration, hallucinations, fiabilité par domaine, agents LLM, calibration de confiance, prompt engineering

1 Introduction

Les grands modèles de langage produisent des réponses fluides, cohérentes et souvent correctes. Mais ils produisent aussi, avec la même assurance, des réponses fausses. Ce phénomène — l'hallucina-

tion — constitue le principal obstacle à la fiabilité des agents IA en production [1].

Le problème fondamental est que les LLM ne savent pas ce qu'ils ne savent pas. Un modèle entraîné sur des millions de documents médicaux et quelques pages de droit malgache répondra aux deux avec le même aplomb. Sa confiance n'est pas calibrée par domaine : elle est *uniforme* et *non informative*.

Les solutions existantes attaquent le problème à différents niveaux. Le RLHF [2] et le DPO [3] modifient les poids du modèle pour favoriser des réponses « préférées » — mais de manière globale, sans granularité par domaine. Les guardrails [4] filtrent les sorties — mais de manière binaire (autorisé/bloqué). La calibration de température ajuste la distribution de probabilité — mais globalement, pas par sujet.

Aucune de ces approches ne fournit à l'agent une *carte de ses propres compétences* : un profil qui lui dit « tu es fiable à 94% en administration Linux, mais seulement à 20% en chimie organique ».

Nous proposons la **méta-calibration** : un profil de fiabilité empirique par domaine, construit par mesure itérative, stocké sous forme de texte structuré, et chargé en contexte à chaque session. Ce profil transforme un agent aveugle à ses propres limites en un agent qui *sait quand il ne sait pas*.

2 Définition

2.1 Méta-calibration

La méta-calibration est la construction et le maintien d'un **profil de fiabilité empirique par domaine** pour un agent IA, basé sur l'accumula-

tion de feedback mesuré (réponses correctes, incorrectes, ou partielles) au fil des sessions.

Formellement, pour un domaine d , la fiabilité méta-calibrée est :

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

avec un intervalle de confiance de Wilson :

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

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

2.2 Ce que la méta-calibration n'est pas

Il est essentiel de distinguer la méta-calibration de concepts apparentés :

- **Confidence logit** : probabilité interne du modèle sur le prochain token. Non interprétable directement, non accessible dans la plupart des API, et non domain-specific.
- **Refusal training** : entraînement du modèle à refuser certaines requêtes. Binaire (refuse/accepte), sans gradation.
- **Calibration de température** : ajustement global de la distribution softmax. Affecte tous les domaines uniformément.
- **Verbalized confidence** : le modèle exprime sa confiance en langage naturel (« je suis sûr à 80% »). Mal calibré car non basé sur des données empiriques [6].

La méta-calibration est **externe** (pas dans les poids), **empirique** (basée sur des mesures), **domain-specific** (granularité par sujet), et **lisible** (texte structuré auditable par un humain).

3 Architecture proposée

3.1 Vue d'ensemble

Le système de méta-calibration comprend quatre composants :

1. **Tracking** — évaluation de chaque réponse
2. **Agrégation** — calcul de la fiabilité par domaine
3. **Profil** — fichier CALIBRATION.md chargé en contexte
4. **Comportement adaptatif** — règles conditionnelles sur la fiabilité

3.2 Tracking

Chaque réponse de l'agent est évaluée selon un verdict ternaire :

- **Correct** — la réponse est vérifiée comme exacte
- **Incorrect** — la réponse contient une erreur factuelle significative
- **Partiel** — la réponse est partiellement correcte (pondérée à 0.5)

L'évaluation peut provenir de trois sources : (a) feedback humain explicite, (b) auto-vérification par l'agent via outils (recherche web, exécution de code, consultation d'API), (c) évaluation par un second agent (cross-validation).

Chaque évaluation est associée à une ou plusieurs catégories de domaine, selon une taxonomie pré-définie ou apprise dynamiquement.

3.3 Agrégation

La fiabilité par domaine est calculée comme :

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

Un **indice de maturité** est associé :

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

où N_{min} est le nombre minimum d'observations pour considérer le profil fiable (par défaut, $N_{\text{min}} = 50$). Si $M(d) < 1$, l'intervalle de confiance est large et le profil est marqué comme immature.

3.4 Fichier CALIBRATION.md

Le profil est stocké dans un fichier structuré chargé en contexte à chaque session :

```
# CALIBRATION.md - Profil de fiabilite

## Domaines evalues

| Domaine          | Fiabilite | n   | IC 95% | |
|---|---|---|---|---|
| Linux/sysadmin   | 94%      | 200 | [90%, 97%] | MATURE |
| Python           | 89%      | 150 | [83%, 93%] | MATURE |
| Droit francais   | 45%      | 12  | [20%, 72%] | IMMATURE |
| Chimie organique | 20%      | 3   | [1%, 70%] | IMMATURE |
```

```
## Regles de comportement

- Si fiabilite < 60% : disclaimer obligatoire
- Si maturite = IMMATURE : mentionner le faible
  nombre d'observations
- Si fiabilite < 40% ET maturite = MATURE :
  suggerer un expert externe
- Si fiabilite < 20% : refuser poliment de
  repondre
```

3.5 Comportement adaptatif

Le fichier de calibration inclut des **règles de comportement** que l'agent suit via prompt engineering. Ces règles définissent une gradation :

1. **Confiance haute** ($R > 80\%$, $M = 1$) : réponse directe, sans qualification.
2. **Confiance modérée** ($60\% < R \leq 80\%$) : réponse avec nuance (« d'après mes connaissances, mais à vérifier »).
3. **Confiance basse** ($40\% < R \leq 60\%$) : disclaimer explicite + suggestion de vérification.
4. **Confiance très basse** ($R \leq 40\%$, $M = 1$) : suggestion d'expert externe, réponse minimale.
5. **Profil immature** ($M < 1$) : mention du faible nombre d'observations, prudence accrue.

3.6 Mise à jour continue

Le profil est mis à jour après chaque session. Le processus est incrémental : seuls les compteurs n_{correct} , n_{partiel} et n_{total} sont mis à jour par domaine. Un mécanisme de *fenêtre glissante* optionnel permet de pondérer les observations récentes plus fortement, capturant ainsi l'amélioration (ou la dégradation) de l'agent au fil du temps.

4 Pourquoi c'est différent et mieux

4.1 Vs. confidence logits

Les logits internes du modèle sont des probabilités sur le prochain token, pas sur la véracité de la réponse complète. Ils ne sont pas interprétables par un humain, pas accessibles via la plupart des API de production, et surtout pas domain-specific : un logit élevé sur le token « 42 » ne dit rien sur la fiabilité de la réponse en chimie vs. en mathématiques.

La méta-calibration est lisible, auditable, et opère au niveau du *domaine*, pas du token.

4.2 Vs. RLHF/DPO

Le RLHF [2] et le DPO [3] modifient les poids du modèle. C'est coûteux, irréversible sans ré-entraînement, et global : un ajustement pour améliorer le droit français pourrait dégrader la chimie. La méta-calibration ne touche pas aux poids. Elle fonctionne entièrement par prompt engineering : un fichier texte chargé en contexte. Le modèle reste inchangé.

4.3 Vs. guardrails

Les systèmes de guardrails [4] opèrent en binaire : la sortie est autorisée ou bloquée. Il n'y a pas de gradation. La méta-calibration offre un spectre continu de comportements, de la confiance totale au refus, en passant par le disclaimer et la suggestion d'expert.

4.4 Coût computationnel

La méta-calibration a un coût computationnel **essentiellement nul**. Le profil est un fichier texte de quelques centaines de tokens, chargé dans le contexte existant. Il n'y a pas de modèle supplémentaire à inférer, pas de poids à modifier, pas de pipeline de filtrage à exécuter. Le seul coût est celui du tracking (évaluation des réponses), qui peut être partiellement automatisé.

5 Expérience proposée

5.1 Protocole

Nous proposons un protocole expérimental en conditions contrôlées :

Agents :

- **Agent A** (baseline) : même modèle, même prompt système, sans fichier de calibration.
- **Agent B** (méta-calibré) : même modèle, même prompt système, avec **CALIBRATION.md** chargé en contexte.

Le profil de calibration de l'Agent B est pré-construit à partir de 500 interactions évaluées sur les domaines cibles, représentant un profil réaliste après quelques semaines d'utilisation.

Tâches : 100 questions réparties sur 5 domaines :

- 2 domaines forts ($R > 85\%$) : Python, administration Linux
- 2 domaines moyens ($50\% < R < 75\%$) : histoire médiévale, économie
- 1 domaine faible ($R < 30\%$) : chimie organique avancée

Chaque question a une réponse de référence vérifiée par un expert humain.

5.2 Métriques

- **Taux d'hallucination** : proportion de réponses contenant au moins une affirmation factuellement fausse.
- **Taux de disclaimer approprié** : proportion de réponses dans les domaines faibles/immatures qui incluent un avertissement.
- **Taux de faux disclaimer** : proportion de réponses dans les domaines forts qui incluent un avertissement inutile.
- **Score de satisfaction** : évaluation humaine (1–5) de l'utilité globale de la réponse.

5.3 Hypothèses

- H1** L'Agent B a un taux d'hallucination significativement inférieur à l'Agent A dans les domaines faibles.
- H2** L'Agent B émet des disclaimers appropriés dans $> 80\%$ des cas en domaines faibles, contre $< 20\%$ pour l'Agent A.
- H3** L'Agent B maintient des performances équivalentes à l'Agent A dans les domaines forts (pas de dégradation par excès de prudence).
- H4** La satisfaction utilisateur globale est supérieure pour l'Agent B, les utilisateurs préférant une incertitude honnête à une confiance trompeuse.

6 Travaux connexes

6.1 Calibration des LLM

Kadavath et al. [5] ont montré que les LLM possèdent une forme de « connaissance de ce qu'ils savent » mesurable via les probabilités de tokens. Cependant, cette calibration interne est globale, non interprétable, et ne se traduit pas en comportement adaptatif sans mécanisme explicite.

6.2 Confiance verbalisée

Plusieurs travaux [6, 7] ont exploré la capacité des LLM à exprimer leur confiance en langage naturel. Les résultats montrent que cette confiance verbalisée est *mal calibrée* : les modèles sont systématiquement sur-confiants, particulièrement dans les domaines où ils sont les moins fiables — exactement le cas où la calibration est la plus nécessaire.

6.3 Prédiction sélective

La prédiction sélective [8] permet à un modèle de s'abstenir de répondre quand sa confiance est insuffisante. C'est un mécanisme binaire (répondre/abstenir) qui ne fournit pas la gradation nuancée de la méta-calibration.

6.4 Prédiction conforme

La prédiction conforme [9] fournit des intervalles de confiance statistiquement garantis. C'est une approche rigoureuse mais qui opère au niveau de la prédiction individuelle, pas du profil de compétence par domaine. Elle ne fournit pas à l'agent une *carte* de ses forces et faiblesses utilisable en contexte.

6.5 Positionnement

À notre connaissance, **aucun travail existant ne propose un profil de fiabilité externe, domain-specific, stocké sous forme de texte structuré et chargé en contexte actif** comme mécanisme de calibration. La méta-calibration se distingue par sa simplicité d'implémentation (prompt engineering pur), son coût nul, et sa lisibilité par l'humain et l'agent simultanément.

7 Discussion

7.1 Le problème du bootstrap

Comment construire un profil de calibration sans données ? Au démarrage, l'agent n'a aucune observation. Plusieurs stratégies sont possibles : (a) initialiser avec des a priori conservateurs ($R = 50\%$, $M = 0$), (b) utiliser des benchmarks publics pour pré-remplir certains domaines, (c) commencer avec un profil « vide » où toute réponse est accompagnée d'un disclaimer de profil immature.

En pratique, après quelques dizaines de sessions, le profil converge vers des estimations utiles.

7.2 Risque de sous-confiance

Un profil de calibration peut rendre l’agent excessivement prudent, émettant des disclaimers même quand il a raison. Ce risque est mitigé par l’indice de maturité : un profil mature avec une fiabilité haute ne déclenche aucun disclaimer. Le risque est principalement présent pendant la phase de bootstrap.

7.3 Gaming du feedback

Si l’agent peut influencer son propre feedback (par exemple via auto-évaluation), il pourrait théoriquement « gamer » son profil en se donnant des notes élevées. Ce risque est atténué par : (a) la priorisation du feedback humain, (b) la cross-validation par un second agent, (c) la comparaison avec des réponses de référence vérifiées.

7.4 Granularité taxonomique

Le choix de la taxonomie des domaines est crucial. Trop grossière (ex. « sciences »), elle perd en utilité. Trop fine (ex. « synthèse asymétrique de composés hétérocycliques »), elle manque de données. Un compromis raisonnable est une taxonomie à deux niveaux : domaine (ex. « chimie ») et sous-domaine (ex. « chimie organique »), avec agrégation hiérarchique.

7.5 Limites de l’approche par contexte

La méta-calibration fonctionne par prompt engineering : elle occupe de l’espace dans la fenêtre de contexte. Pour un agent couvrant des centaines de domaines, le fichier de calibration pourrait devenir volumineux. Des stratégies de compression (ne charger que les domaines pertinents à la requête en cours) peuvent atténuer ce problème.

8 Conclusion

Nous avons proposé la méta-calibration : un mécanisme simple, externe et lisible permettant à un agent IA de connaître ses propres limites par domaine. Contrairement aux approches qui modifient les poids du modèle ou filtrent ses sorties, la

méta-calibration fonctionne par accumulation de feedback empirique et injection en contexte.

Le principe est analogue à celui d’un professionnel humain qui, après des années d’expérience, sait dire « ça, c’est mon domaine » ou « là, il vaut mieux consulter un spécialiste ». Cette compétence — la connaissance de ses propres limites — est précisément ce qui manque aux LLM actuels.

L’expérience proposée vise à démontrer que cette approche réduit significativement les hallucinations dans les domaines faibles, sans dégrader les performances dans les domaines forts, et que les utilisateurs préfèrent un agent honnête sur ses limites à un agent uniformément confiant.

La méta-calibration s’inscrit dans une vision plus large de l’ingénierie d’agents : des systèmes qui ne sont pas seulement capables, mais *conscients de leurs capacités*. C’est une brique essentielle vers des agents IA dignes de confiance.

Références

- [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.